October 2025

Camera Guide

Blackmagicdesign

Blackmagic URSA Cine Immersive



Blackmagic URSA Cine Immersive

Languages

To go directly to your preferred language, simply click on the hyperlinks listed in the contents below.

English 3

日本語 154

Français 306

Deutsch 458

Español 610

中文 762

한국어 914

Русский 1066

Italiano 1218

Português 1370

Türkçe 1522

Polski 1674

Українська 1826

Contents

Blackmagic URSA Cine Immersive 5
What is Immersive? 5
Apple Immersive Video 5
URSA Cine Immersive 6
Getting Started 7
Powering your Camera 7
Setting the Record Settings 8
Monitoring your Image 10
Monitoring Options and Overlays 10
Framing your First Shot 11
Camera Height 11
Camera Tilt 12
Positioning 13
Audience Agency 14
Camera Movement 14
Data Rate Tables 15
Audio and Timecode 16
Apple Spatial Audio Format 16
Ambisonic 16
Lavaliere and Miniature Microphones 17
Handheld Microphones 17
Connecting Timecode 17
Shooting 19
Exposure Requirements 19
Shutter Speed 19
ISO and Dynamic Range 20
ND 21
Off Speed Recording 22
Histogram 23
LCD Monitor Options 23
Recording 24
Playback 25
Transferring Clips 26
Summary 27
DaVinci Resolve Studio 28
Workstation Recommendations 28
Project Setup 29
Master Settings 29
Video Monitoring 30
Image Scaling 30
Color Management 31
About DaVinci Wide Gamut 32
Apple Immersive Processing 33
Ingesting 33
Editing 35
Assembly 35
Backdrops 35
Using the Viewer 37
Transitions 39
Apple Immersive Motion Data 40
Titles 41
Creative Considerations 42
VFX Compositing 43
MediaOut nodes 43
Renderer3D 44
Compositing with the Immersive Patcher 45
Color Grading 46
Noise Reduction 46
Primary Grading 47
Secondaries 47
Sizing 48
3D Palette 48
CST Color Management 50
CST Timeline Mapping 50
CST Grading Monitor Mapping 51
CST Apple Vision Pro Monitoring 53
Audio 54
Apple Spatial Audio Format (ASAF) 54
Delivery 55
Bypassing Transitions 55
Streaming to the Apple Vision Pro 56
About Apple Immersive Video Utility 56
Developer Information 57
Camera Control REST API 57
Livestream Control API 59
Clips Control API 63
Media Pool Control API 64

Monitoring Control API 67
Event Control API 76
System Control API 76
Transport Control API 82
Timeline Control API 86
Media Control API 89
Slate Control API 92
Preset Control API 98
Audio Control API 100
Lens Control API 105
Video Control API 109
Camera Control API 117
Immersive Control API 120
Color Correction Control API 121
Notification websocket - 1.0.0 126
messages 126
Help 150
Regulatory Notices 151
Safety Information 152
Warranty 153

Blackmagic URSA Cine Immersive

Congratulations on your purchase or rental of the Blackmagic URSA Cine Immersive.

Blackmagic URSA Cine Immersive is designed to shoot 180 degree stereoscopic VR footage, which is commonly called immersive video. In particular, it has been optimized to shoot Apple Immersive Video for the Apple Vision Pro, the highest quality Immersive experience currently available.

This guide is designed to be an introduction to Blackmagic URSA Cine Immersive and how to shoot with it, focusing on the camera's basic operation and the unique format that is designed to deliver, Apple Immersive Video. An understanding of the format will help you to better understand the camera itself.

In the following pages, you will find a mix of technical and creative advice that will help you not just to use the camera, but to understand the benefits of the immersive format.

What is Immersive?

Immersive is a three dimensional spatial video format. You're likely aware that other similar technologies have existed for some time. To understand what makes immersive different, here's a broad overview of other technologies:

- **Traditional 3D Cinema** allows the audience to experience depth within the frame by capturing each scene with two cameras with an interaxial distance that mimics the distance between human eyes. While stereoscopic images allow the audience to experience depth, the audience's perspective is controlled by the cinematographer's composition.
- **Spherical Video** combines the principles of 3D and 360 degree cameras to allow audiences to look in any direction while experiencing depth. 360 degree images require very high resolutions to ensure the small portion of the image that the audience is looking at has sufficient fidelity. As human physiology only allows us to turn our head from shoulder to shoulder, to be able to look behind, the audience is typically standing or seated on a swivel chair.
- **Volumetric Video** allows the audience not just to experience depth and look around, but to move and change their perspective. While camera arrays can capture volumetric video most virtual reality or augmented reality makes use of dynamically rendered 3D assets and requires a safe environment for participants.

Immersive combines the best of traditional cinema and 3D interactive formats. In its simplest form, it is a high frame rate, high resolution 180 degree stereoscopic format.

Apple Immersive Video

Blackmagic URSA Cine Immersive is designed specifically to capture Apple Immersive Video. This format is used with Apple Vision Pro and supports high resolution, stereoscopic 3D video in a 180 degree field of view.

Apple Immersive Video has the following features:

- 90 frames per second playback.
- Can be distributed at resolutions up to 4320 x 4320 per eye.
- A unique metadata system for mapping the original camera image into the 180 environment.
- Apple Spatial Audio Format to provide 3D sound.

Apple Immersive Video requires hardware capable of high resolution stereoscopic 180 degree playback. Apple Vision Pro is currently the only device that supports this format.



Apple Vision Pro

The headset features dual 3660 x 3200 pixel micro OLED displays with a 7.5 micron pixel pitch. It includes 13 tracking cameras, a LiDAR scanner, internal measurement units, and eye tracking sensors to align imagery with head and eye movement.

An M2 processor with an 8 core CPU and 10 core GPU drives playback at up to 100 frames per second.

URSA Cine Immersive

Blackmagic URSA Cine Immersive is designed for capturing Apple Immersive Video for Apple Vision Pro.



Blackmagic URSA Cine Immersive

URSA Cine Immersive uses two matched 58 megapixel RGBW sensors. The sensors are nearly square to capture a full 180 degree field of view, both horizontally and vertically. Each sensor is paired with a 210 degree fisheye lens. The lens centers are placed 64mm apart.

Built in ND filters provide up to 8 stops of light control. The lens aperture is fixed at approximately f4.5.

The camera records two 8K image streams at 90 frames per second. This results in significantly higher data rates than standard 4K 24p recording.

Getting Started

Getting started with your Blackmagic URSA Cine Immersive is as simple as powering your camera and turning the camera on. Blackmagic Media Module 8TB is loaded into the camera's module bay, formatted and ready to record clips straight away!

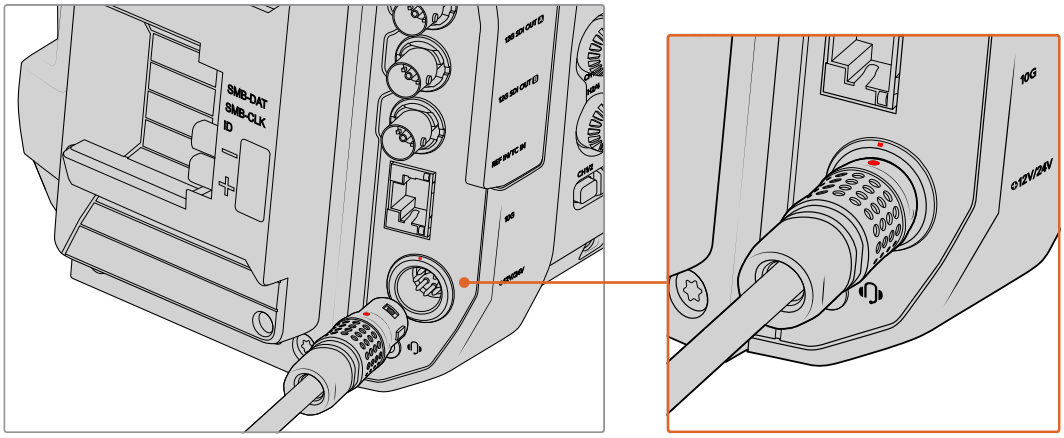
Powering your Camera

The quickest way to power your camera is to connect external power using the supplied power adapter.

To plug in external power:

1. First remove the rubber protective dust cap and store it somewhere safe. Connect the AC to 24V DC adapter plug to your mains power socket using a standard IEC C13 mains power cable.
2. Connect the adapter's 8 pin connector to the 24V DC power connector on the camera's rear panel.

If you have both external and battery power connected, the camera will draw from the external power source. Removing external power while a charged battery is connected will switch your camera to battery power without interruption.



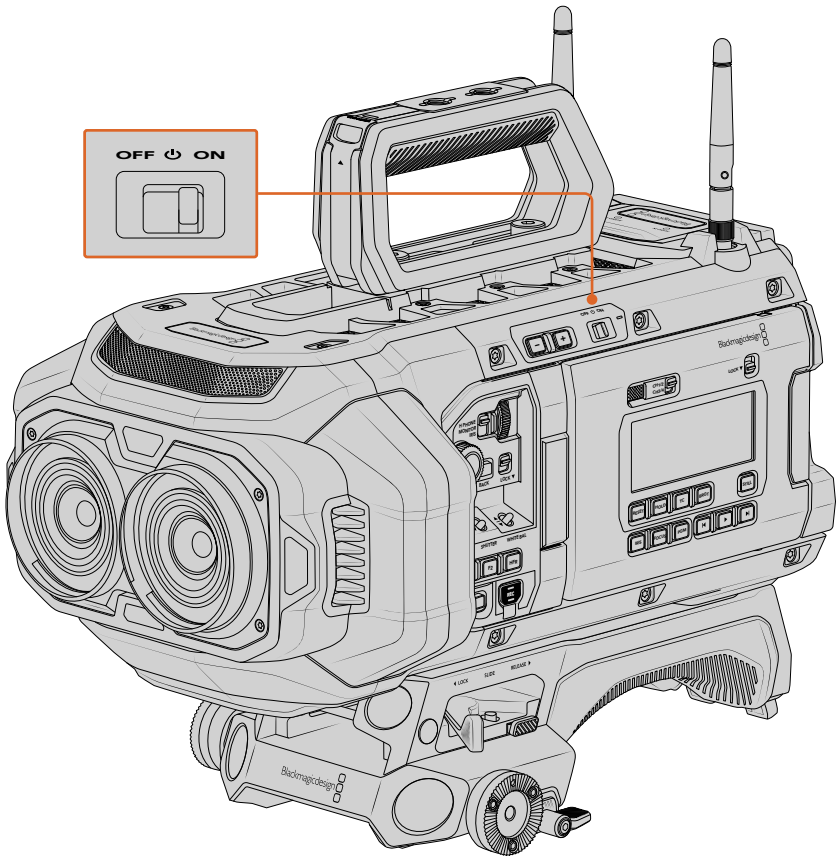
To plug in power, align the red dots and gently push the connector in until you feel it lock into place. To remove, hold the connector's metal sleeve and gently pull away from the camera. This will pull the sleeve back, releasing the connector from the socket.

The battery plate included on your camera is a B mount plate designed for 24 volt B mount batteries. Operating in the 24 to 34 voltage range is ideal for use with URSA Cine Immersive as the higher voltage reduces thermal losses allowing for high frame rate recording and powering multiple accessories from your camera.

Optional VLock and Gold Mount plates, compatible for 24v operation and Gold Mount Plus plates for 26v third party batteries are also available.

NOTE When running URSA Cine Immersive with 12 to 18 volt power supplies or batteries, powering accessories via the 3 pin RS connector or 2 pin lemo style connector is not possible.

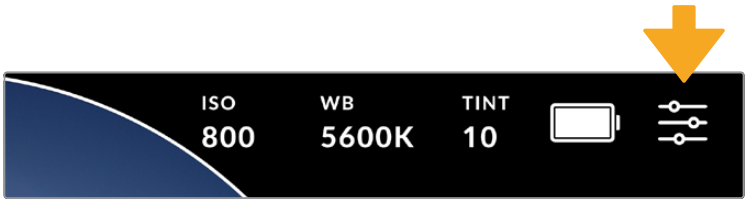
Once you have mounted a battery or connected external power, turn your URSA Cine Immersive on. Move the switch at the top of your camera to the ‘on’ position.



Move the power switch to ‘on’ to power your camera

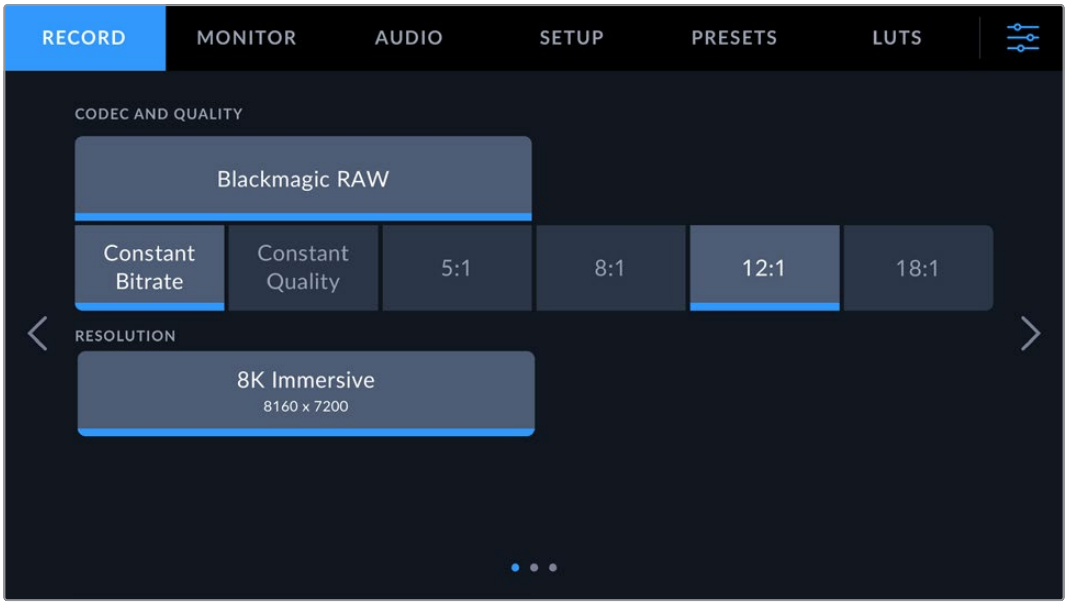
Setting the Record Settings

This section provides information on specific camera settings that are unique or important when shooting immersive content. To open your camera’s settings menu, tap the menu settings icon in the top right corner of the fold out or assist touchscreen.



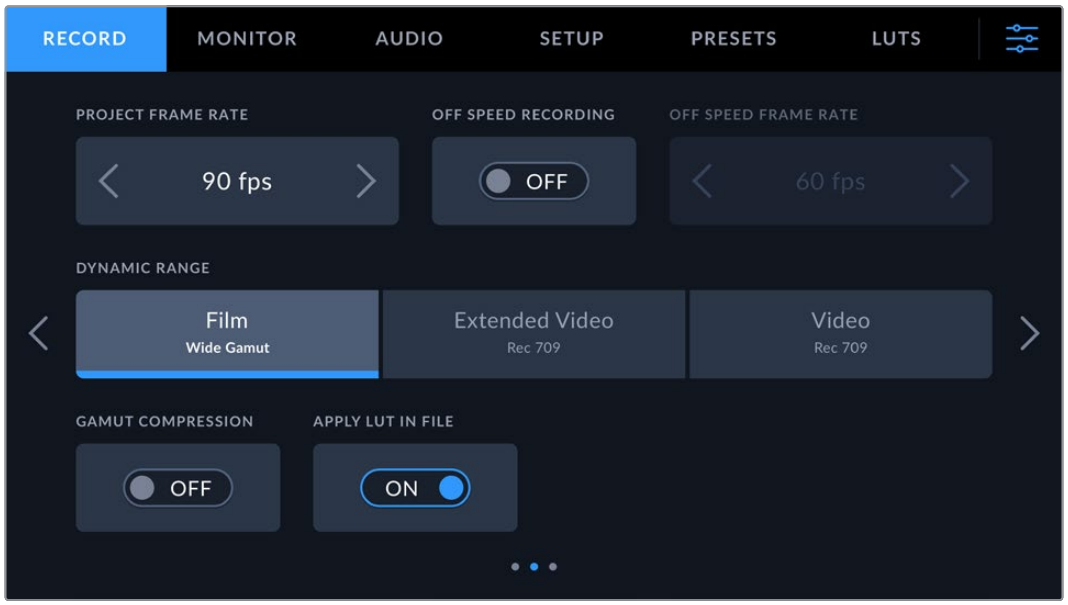
Record Settings 1

The ‘codec and quality’ options on this page lets you choose the Blackmagic RAW codec setting to suit the project you are working on. The options are divided between four ‘constant bitrate’ and ‘constant quality’ settings. We highly recommend using 12:1 or Q3 compression settings if recording 90fps content for Apple Vision Pro. These settings provide the sweet spot for maintaining manageable data rate and file sizes, with current computing hardware, while still retaining extremely high image quality.



Record Settings 2

The second page of your camera's 'record' settings includes the frame rate settings.



Your camera's project frame rate should be set to 90 frames per second to conform to Apple Vision Pro's display frame rate.

If you are shooting for other systems, lower frame rates such as 60fps or 50fps may be sufficient. Lower traditional film style capture frame rates such as 24fps or 25fps may be appropriate if you are shooting visual effects plates for 24p motion picture film content.

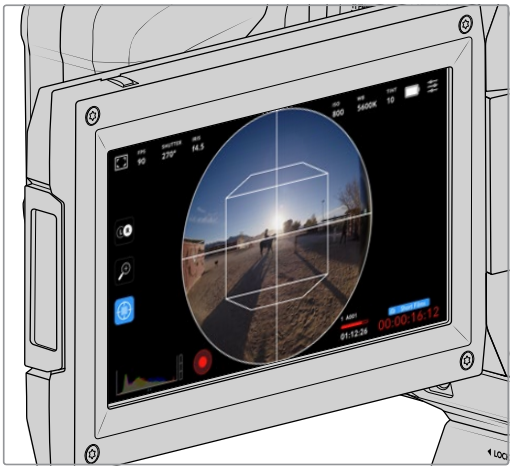
That's all there is to getting started with your URSA Cine Immersive! With the media module already formatted in your camera, you can start recording straight away.

Monitoring your Image

Your URSA Cine Immersive includes several options for monitoring your image while shooting.

The fold out touchscreen is intended for use by the camera operator. It can be reversed to reduce the camera's physical profile while still keeping the screen visible, or positioned forward to enable self monitoring when filming yourself.

The assist touchscreen allows another crew member to monitor the image without obstructing the operator's view. Due to the camera's extremely wide field of view, care should be taken when using the assist touchscreen.





Monitoring Options and Overlays

The camera displays lens space, with one eye viewable at a time. The fold out and assist touchscreens are masked to display a 180 degree view. The SDI outputs show the full 210 degree image captured by each sensor.



All monitoring outputs, display the same eye. You can switch between the left and right eye by tapping the 'L R' eye toggle button on either touchscreen.

An optional overlay can be enabled on the touchscreens by pressing the overlay button. This overlay includes the following:

- A 180 degree outer boundary
- A central crosshair marking the centre of the frame
- An inner guide representing the approximate field of view seen on an Apple Vision Pro when looking straight ahead.

These tools are designed to assist with accurate framing and positioning during immersive capture.

Framing your First Shot

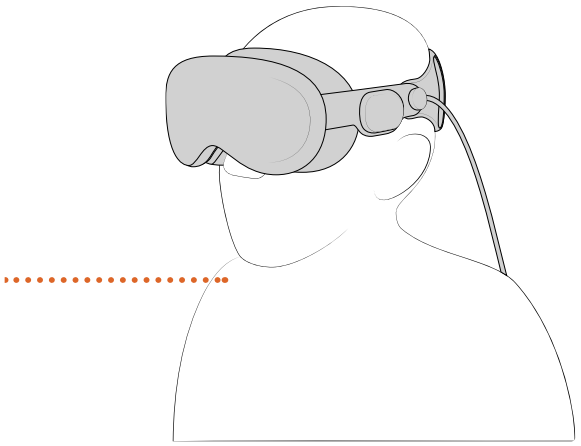
Immersive and traditional 2D content shares a lot in common but there are also distinct differences.

For example, have you ever got motion sickness in a car because you're too busy scrolling on your phone? It is caused by a sensory mismatch when your brain receives conflicting signals. Your inner ear is telling your body you are moving, but your eyes are telling it you are stationary.

It's easy to produce a similar experience with immersive. For example, if you pan the camera, your audience's eyes will tell them they are moving, but their neck and inner ear will tell them they are very much stationary. As a result, it's best to master stationary shots before learning how to make camera movement pleasant for your audience.

Camera Height

In traditional 2D cinema, you might place the camera high or low to signify the balance of power in a scene. When shooting Immersive, consider the impact of placing the camera on the ground. How might that affect the viewer? It could give the viewer the sensation of being buried up to their neck.



The perceived height of the floor can have an impact on your audience's viewing experience. A low camera will give your audience a perceived floor height that might not make sense!



When shooting immersive, low angles could make the audience feel uncomfortable

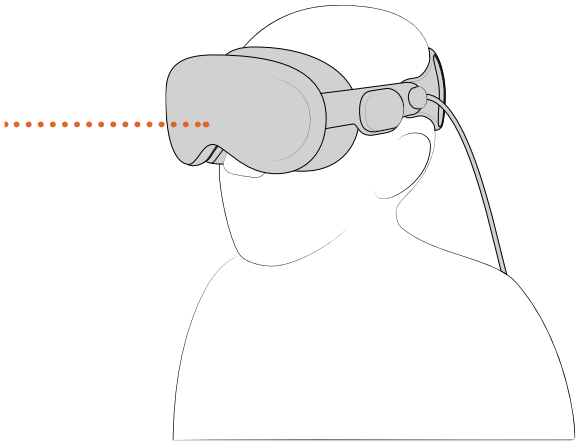
To help the viewer feel physically present, it's often best to choose a camera height and position that makes the viewer feel like they are standing or sitting in the scene. There is still room to express your creative intent. If there are people present in the scene, a slightly higher camera can imbue the viewer with power, allowing them to look down on others. Conversely, a slightly lower camera forces the viewer to look up at other people.

That's not to say that it's not possible to maintain audience comfort with more adventurous camera placement. Flying cameras, top down views, low angles and many other techniques are successfully being used in Immersive productions. When combined with the right editing, movement, and context, they can be used to entertain your audience.

Camera Tilt

When shooting traditional 2D content and placing the camera high or low, the cinematographer will often use camera tilt to maintain composition. Consider how that would affect an Immersive image.

If you were to angle the camera downwards, for your audience to view the horizon, they would need to angle their head upwards. This would be unlikely to cause motion sickness, but it could break their sense of presence because in real life, the horizon is always directly ahead.



Immersive shots will make most sense when the horizon is at zero degrees.



If you were to tilt the URSA Cine Immersive, there will be a disconnect for the audience between where they have to look to see the horizon and where they would normally look to see the horizon.

Therefore, most Immersive shots are captured with a zero degree, or near zero degree tilt. As the camera captures a 180+ degree field of view, both horizontally and vertically, the audience is still able to look up and down. But instead of the filmmaker forcing the audience to look in a particular direction, the audience is in control.

To encourage the audience to look at different parts of your image, you can use other techniques like lighting, aperture framing, and leading lines.

Positioning

Immersive is typically viewed when seated. As it is not an augmented reality or virtual reality format, viewers do not stand or move around. Are you seated presently? As an experiment, turn your head from side to side and move it up and down. How far can you comfortably turn your head without moving your body? The immersive format's field of view exceeds most people's comfortable range of rotation. That's a good thing because it means if you rotate to the left or right, the format still provides peripheral vision. This experiment helps you start to understand where your main action should be positioned within the field of view.

Typical head movement range:

Rotation 120 to 160 degrees
Flexion and Extension: 110 to 130 degrees

When head rotation is combined with eye movement, it is possible to look over your own shoulder, however it's uncomfortable for most people to sustain.

To aid composition, enable your camera's overlay. Keeping action within the inner marks that show typical field of view will aid your content's viewability.

The movement of points of interest within the frame should also be considered. Tracking a person as they walk from extreme left to extreme right requires extreme head movement. Similarly, cutting from a shot where the point of interest is at the edge of the image to a shot where the point of interest is on the opposite edge requires extreme head movement. Careful consideration of composition can aid audience comfort while still communicating creatively.

Audience Agency

When shooting traditional 2D content, you can use the frame, choice of lens and focal settings to direct the viewer's attention. When filming immersive, the 180 degree field of view makes it harder to exclude information and direct the audience's attention. Immersive gives the audience far greater agency to explore your images. This is one of the factors that makes immersive content so engaging. It's still possible to guide your audience's attention though careful use of directional sound, lighting, blocking and composition.

Camera Movement

An immersive camera does not need to remain static, but care must be taken as some camera movement can feel unnatural or uncomfortable. The car is a good analogy for helping to understand how to make immersive camera motion comfortable. A large vehicle has mass and inertia which resists sudden changes in speed or direction. As such, it will accelerate and decelerate over time and move in a predictable and controlled manner. Once you have mastered static shots and move onto moving immersive shots, giving consideration to these factors will help you to create pleasing motion:

- **Duration:** Allow your audience time to adjust to the camera motion and to enjoy it. Brief movements are more likely to cause disorientation.
- **Intensity:** Low intensity moves are easier to integrate into an immersive edit. High intensity moves are possible, but a warm up and cool down might aid audience comfort.
- **Visual Anchors:** These are parts of the scene that don't move. A shot without visual anchors can be disorienting. Foreground elements like the inside of a car or distant objects like the horizon will help ground your moving shots.

Data Rate Tables

The tables below contain data rates for 8K x 7K immersive content at 90 frames per second and 24 frames per second.

Maximum Data Rates at 90fps with Blackmagic RAW Constant Bitrate

8K x 7K Immersive			
Codec	**Pixel Dimensions**	**Frame Rate**	**Data Rate**
Blackmagic RAW 5:1	8,160 x 7,200 Stereo	90	3,203 MB/s
Blackmagic RAW 8:1	8,160 x 7,200 Stereo	90	2,002 MB/s
Blackmagic RAW 12:1	8,160 x 7,200 Stereo	90	1,335 MB/s
Blackmagic RAW 18:1	8,160 x 7,200 Stereo	90	889 MB/s

Maximum Data Rates at 24fps with Blackmagic RAW Constant Bitrate

8K x 7K Immersive			
Codec	**Pixel Dimensions**	**Frame Rate**	**Data Rate**
Blackmagic RAW 5:1	8,160 x 7,200 Stereo	24	854 MB/s
Blackmagic RAW 8:1	8,160 x 7,200 Stereo	24	534 MB/s
Blackmagic RAW 12:1	8,160 x 7,200 Stereo	24	356 MB/s
Blackmagic RAW 18:1	8,160 x 7,200 Stereo	24	237 MB/s

Audio and Timecode

Audio is a critical part of the immersive experience. It's important to pair your URSA Cine Immersive with dedicated audio capture equipment.

Apple Spatial Audio Format

Apple Spatial Audio Format, or ASAF, is used to provide 3D audio when watching Apple Immersive Video. ASAF is part of the wider Apple Positional Audio Framework and can store three different types of audio recording.

- **Channel based:** You can embed a traditional surround mix in your ASAF file. This is ideal for things like head locked narration and music.
- **Object based:** Sounds can be treated as individual objects with metadata that describes their position in 3D space, independent of individual speaker channels. These are typically mono recordings. Their 3D coordinates can be animated to follow objects that move within the scene. This is ideal for diegetic audio, which refers to objects that are visible in the scene.
- **Ambisonic:** A full sphere surround format that can render audio in all directions using spherical harmonics. It is ideal for environmental sound.

The rest of this section provides an overview of the equipment that you will need to build an effective ASAF file.

Ambisonic

An Ambisonic microphone contains four or more capsules arranged in a precise geometric pattern. Each capsule captures the same sounds with different arrival times and intensity depending on the direction that a sound comes from. Those capsules are recorded to discrete channels for processing by a computer later. By comparing the differences, software can estimate the direction and nature of the recorded sounds.



A first order ambisonic microphone

Ambisonic Microphones

Ambisonic microphones are classified by the number of capsules they contain. More capsules means the microphone can reproduce spatial detail more accurately.

First order:	4 channels
Second order:	9 channels
Third order:	16 channels
Fourth order:	25 channels and above

Whether the microphone is on the camera or by its side, it's critical that ambisonic microphones are correctly aligned with the camera's shooting direction. Additionally, because ambisonic microphones can be used in different configurations, such as endfire, broadside, upward or downward, it's important to make sure your audio technician knows which configuration was used. This information can be conveyed via metadata on the audio recorder or by sharing notes.

Ambisonic recordings can be used as an environmental bed. Other channel and object based sound effects can be layered on top. If correctly aligned with the camera, ambisonic recordings can also serve as a reference for the audio engineer as they add further sound design.

Lavaliere and Miniature Microphones

When you are filming with a 180 degree camera, it's impossible to use a boom pole as it will be visible in your scene! This means you'll need to use hidden microphones and lavalieres to record diegetic audio and dialogue.

For example, you could:

- Connect those audio sources directly to the URSA Cine Immersive's two XLR inputs using wireless transmitters and receivers.
- Connect them to an external audio recorder. That could be a large multi channel device or a miniature body worn style device.

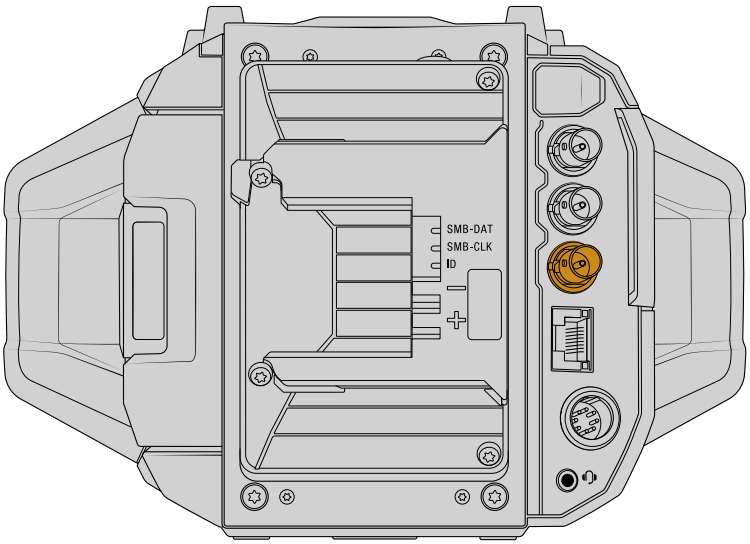
These recordings can then by synced and mixed as either channel or object based sounds.

Handheld Microphones

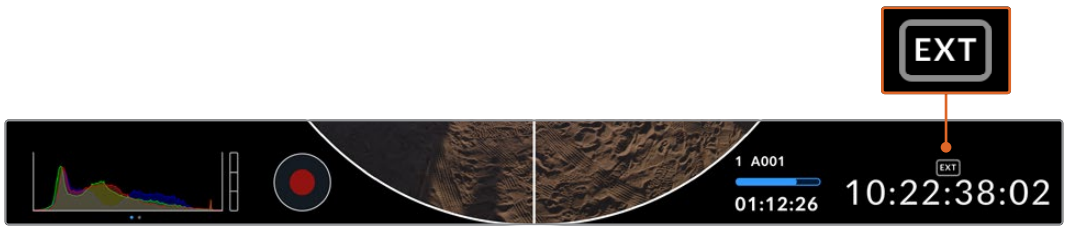
Recording additional sound effects on location after the camera has stopped rolling can not only save time in post production but may also yield a more authentic sound. Carrying a handheld hypercardioid or similar microphone on production is very beneficial, allowing actions to be re performed with the microphone positioned as closely as possible. These recordings can then be edited into the final mix as either channel or object based sounds.

Connecting Timecode

To connect external timecode to your URSA Cine Immersive, use a BNC cable to connect a timecode generator to the timecode input on your camera's rear panel. Connecting to syncronized timecode when shooting dual system ensures audio and picture can be easily synchronized during post production.



When external timecode with a matching frame rate is plugged into this connector the camera will lock to the incoming timecode automatically. The ‘EXT’ logo will appear next to the timecode counter on URSA Cine Immersive’s LCDs when external timecode is locked. If you unplug the cable, timecode will remain jammed and the ‘EXT’ logo will switch to ‘INT’ to let you know it is now running from your camera’s internal timecode clock.



The camera timecode will show ‘ext’ when connected to a timecode generator

NOTE SMPTE timecode standards exist for typical frame rates like 23.976, 24, 30, etc. There is not a SMPTE timecode standard for 90 frames per second. As such, when the URSA Cine Immersive’s Project Frame rate is set to 90 frames per second, the camera actually uses 30fps timecode. The SDI outputs also operate at 30p. Any timecode accessories or external audio recorders should be configured to 30fps.

Shooting

Now that you have a basic understanding of how to operate your Blackmagic URSA Cine Immersive camera and make the most of the Immersive format, this chapter will further explore the operation of the camera.

Exposure Requirements

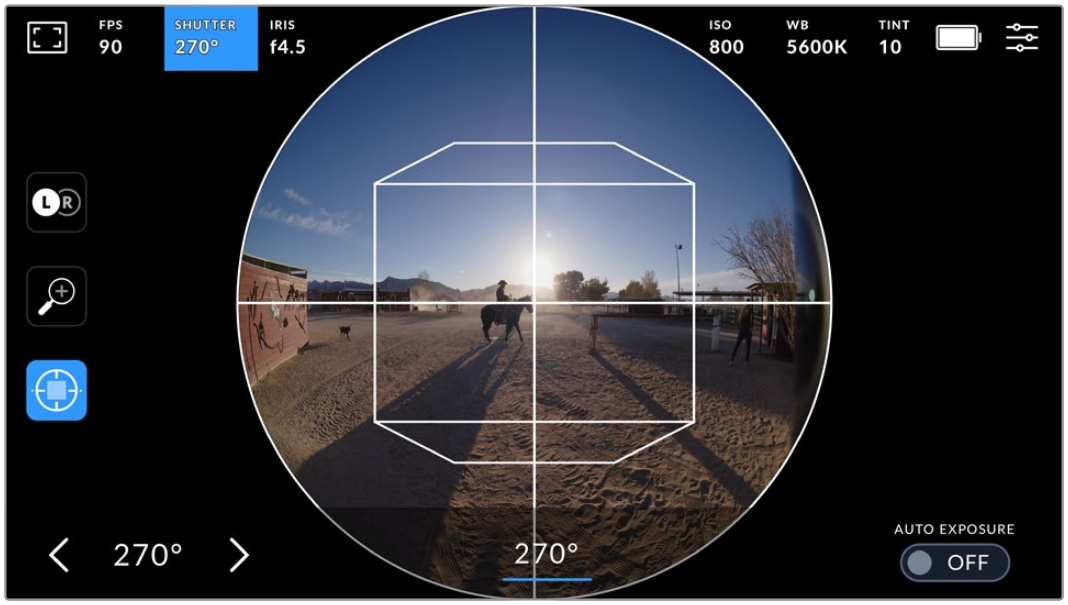
Apple Immersive Video has higher exposure requirements than traditional 24 fps capture and might require you to rethink your shooting and lighting approach. If you maintain a 180 degree shutter, 90 fps capture requires almost four times more light to expose to the same amount. In addition, the fixed iris of the lenses, which is around f4.5, is up to three stops darker than typical super speed lenses. This means you might need up to seven stops more light to achieve adequate exposure. However, if you are shooting at 90fps, using a wide shutter angle between 270° and 360° can help to compensate for high frame rate light loss while still achieving pleasing amounts of motion blur.

Shutter Speed

The ‘shutter’ indicator displays your shutter angle or shutter speed. The shutter measurement setting in the ‘setup’ menu can be used to select whether to display shutter information as ‘shutter angle’ or ‘shutter speed’.

To change the shutter speed:

1. Tap the ‘shutter’ icon in the upper toolbar.
2. Use the left or right arrows to cycle through common shutter speed presets. If you would like to choose a specific shutter value, you can do so by double tapping the current shutter indicator at the bottom left of your screen. This opens a keypad that allows you to type in any shutter value with up to two decimal places between 11.2 and 360 degrees.



Custom frame rates can be entered using the onscreen keypad.

Shutter speed has less of an impact on the appearance of high frame immersive content. While a 180 degree shutter is still favored by many for shooting at 24 or 30p, it’s more common to see shutter speed used to aid exposure when shooting for immersive.

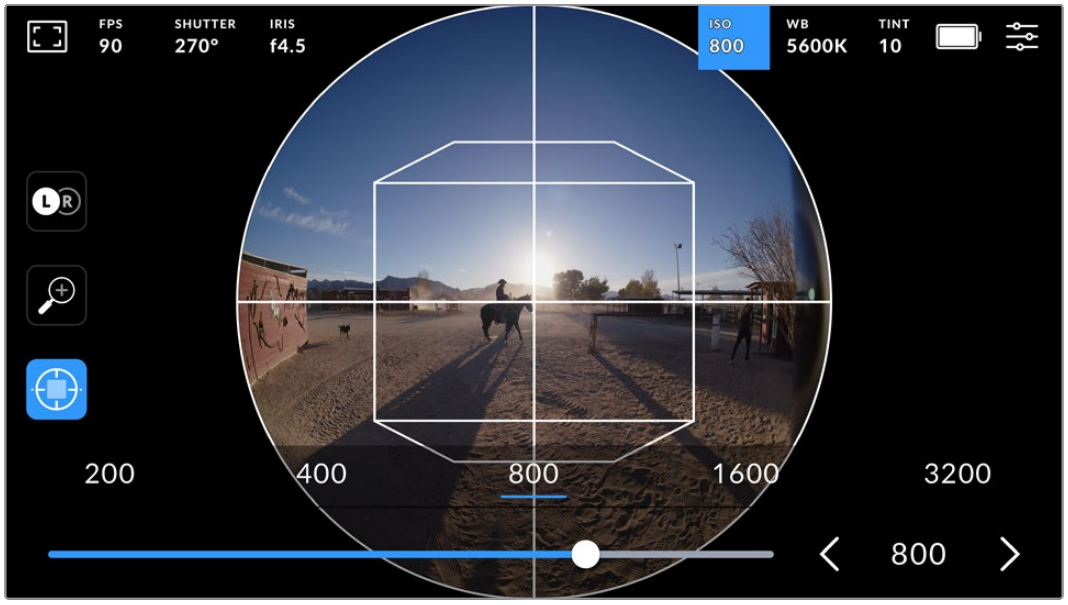
When shooting in low light, the shutter can be increased to 270 degrees or, in some exceptional circumstances, 360 degrees.

When shooting in bright conditions, the shutter speed can be lowered as far as 11.2 degrees. However, in bright conditions, it is more common to leave the shutter speed at your preferred setting for motion and use ND filters to control exposure.

It's worth noting that Apple Immersive Video's high frame rate of 90 frames per second can cause issues when shooting with flickering lights. A 270 degree shutter is suggested when working in 60hz environments and a 324 degree shutter is recommended when working in 50hz environments.

ISO and Dynamic Range

The 'ISO' indicator displays your camera's current ISO setting, or light sensitivity. Tapping this indicator lets you adjust your ISO to suit varying lighting conditions. The ISO range on URSA Cine Immersive is from ISO 200 to 3200. The optimum ISO is 800.



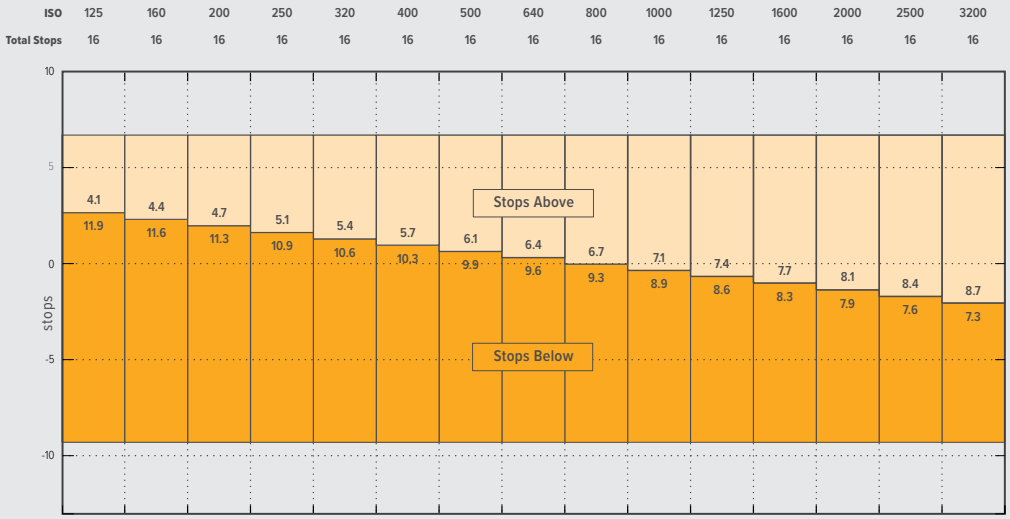
The iso can be configured by tapping the iso indicator.

Numerical presets provide the option to select an ISO setting in full stop increments by tapping on the number. The slider below the presets allows you to fine tune your ISO setting in between these presets in 1/3 stop increments.

Depending on your situation, you may choose a lower or higher ISO setting. For example, in low light conditions ISO 1600 can be suitable but may introduce some visible noise. In bright conditions ISO 200 can provide richer colors.

Blackmagic cameras are able to capture in wide dynamic range which gives you lots of flexibility when capturing images and also when color grading in DaVinci Resolve. This table shows the available dynamic range in your URSA Cine Immersive.

URSA Cine Immersive Dynamic Range



ND

The camera's ND filters are typically the most common way of controlling exposure, allowing you to reduce the amount of light reaching your camera's sensor by a preset number of exposure 'stops'.



The URSA Cine Immersive's ND are adjusted using the buttons on the left hand side above the fold out touchscreen. Together with a clear filter, the available options are '2', '4', '6' and '8' stops.

The current ND setting is displayed via the indicator in the top left of the touchscreen. When ND is engaged, this indicator remains on permanently. When ND is set to clear the indicator will disappear after 4 seconds.



NOTE You can change the ND filter terminology to reflect the conventions you're used to. Use the setup menu to select between ND number, stops and fractions.

Off Speed Recording

The project frame rate in the setup menu is URSA Cine Immersive's recording format frame rate and defines the speed at which your footage will play back. This is normally set to match your playback speed and audio sync used in your post production workflow and delivery requirements.



Tap your URSA Cine Immersive's frames per second indicator to access frame rate settings. URSA Cine Immersive has 8 project frame rate settings including 23.98, 24, 25, 29.97, 30, 50, 59.94, 60 and 90 frames per second. When shooting Immersive, the project frame rate should be set to 90 frames per second.

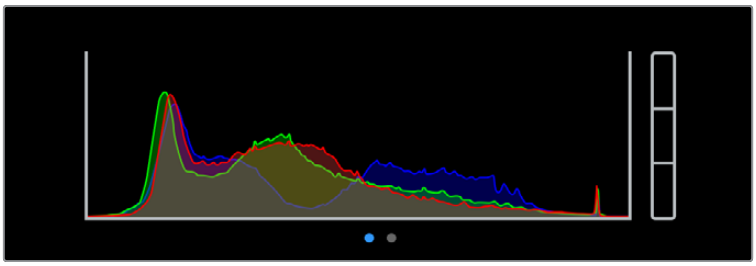
The sensor frame rate allows you to define the speed at which your sensor captures an image. By default, your URSA Cine Immersive's project and sensor frame rates are matched for a natural playback speed. However, by tapping the 'off speed frame rate' switch icon in the bottom right hand side of your camera's 'FPS' menu, you can independently set your sensor frame rate.

When the off speed frame rate switch is enabled, you can manually set the sensor frame rate. It's worth noting that 90 frames per second is the camera's highest frame rate and will match the project frame rate of 90 fps when filming for Apple Vision Pro. However, selecting a lower project frame rate like 30 or 60 frames and then shooting off speed at 90 frames per second will let you create beautiful slow motion footage.

While the Off Speed Frame Rate switch is engaged, the frame rate indicator in the top left of the screen will show the sensor frame rate followed by the project frame rate.

Histogram

At the bottom left of your URSA Cine Immersive touchscreen, you'll see the histogram. A histogram is a graph that shows the brightness distribution in an image. The left side represents shadows, the middle shows midtones, and the right shows highlights. The height of the graph indicates how much of the image is at that brightness level.



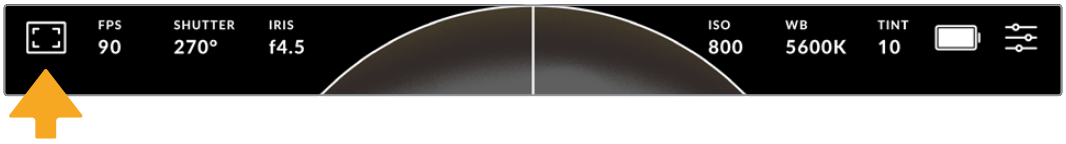
The histogram gives you an indication of the tonal distribution in your clip at a glance. An image where the graph sits on the left hand side is likely underexposed. Similarly, an image where the graph sits on the right hand side is likely overexposed. The histogram should always be interpreted in context with the image, but in general, a well exposed image has a balanced graph that sits in the middle. Increasing exposure will move the histogram to the right, reducing exposure will move it to the left.

To the right of the histogram is an indicator that shows when individual color channels clip. Typically, you should avoid clipping any channels. There are, however, scenarios when it's acceptable to clip your image if it helps you to preserve the image's midtones and shadows.

Expose to the Right is a technique where you deliberately overexpose an image by pushing the histogram as far to the right as possible without clipping color channels. This maximizes signal and minimizes noise. As long as the exposure is lowered in post, this is an excellent way of producing clean images!

LCD Monitor Options

Tap the monitor icon at the top left of your URSA Cine Immersive's LCD touchscreen to access the LCD monitor settings. This menu contains useful options like monitor brightness. It also contains powerful exposure tools.



Zebra

The 'zebra' setting toggles the appearance of zebra on the touchscreen, as well as setting the zebra level. Zebra displays diagonal lines over areas of your image that exceed a set exposure level. The percentage setting is used to show parts of your image that are overexposed. It's common to set this to a value like 95% rather than 100% as it allows you to identify parts of the image that are curable to over exposure. The second zebra setting lets you turn mid gray zebra zones on or off, or mid gray plus 1 stop.

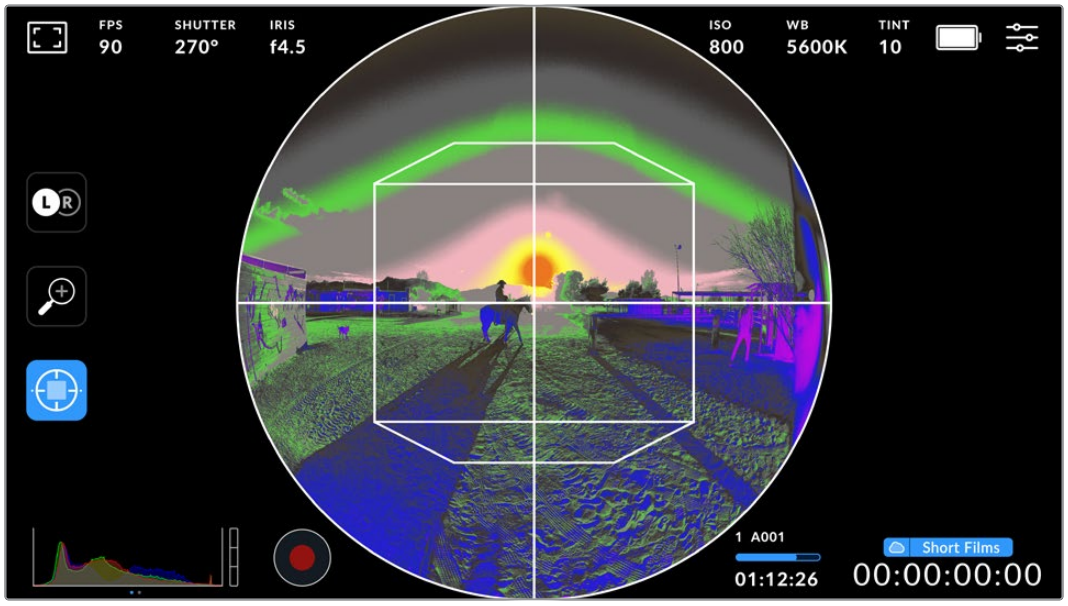
Grids

The ‘grids’ setting toggles the appearance of a rule of thirds grid, horizon meter, crosshair or center dot on the LCD touchscreen, as well as setting the overlay that will be visible on all URSA Cine Immersive’s outputs. When shooting immersive content the horizon meter is very useful. It displays a double crosshair in the centre of the image that is aligned as long as the camera is completely level, but diverges when the camera is not.

False Color

False Colour overlays different colors onto your image that represent exposure values for different elements in your image. For example, pink represents optimum exposure for lighter skin tones, while green is a good match to darker skin tones. By monitoring the pink or green false color when recording people, you can maintain consistent exposure for their skin tones.

Similarly, when elements in your image change from yellow to red, that means they are now overexposed. Similar to the histogram, this tool gives you a very quick and intuitive understanding of the image’s tonal distribution.



Recording

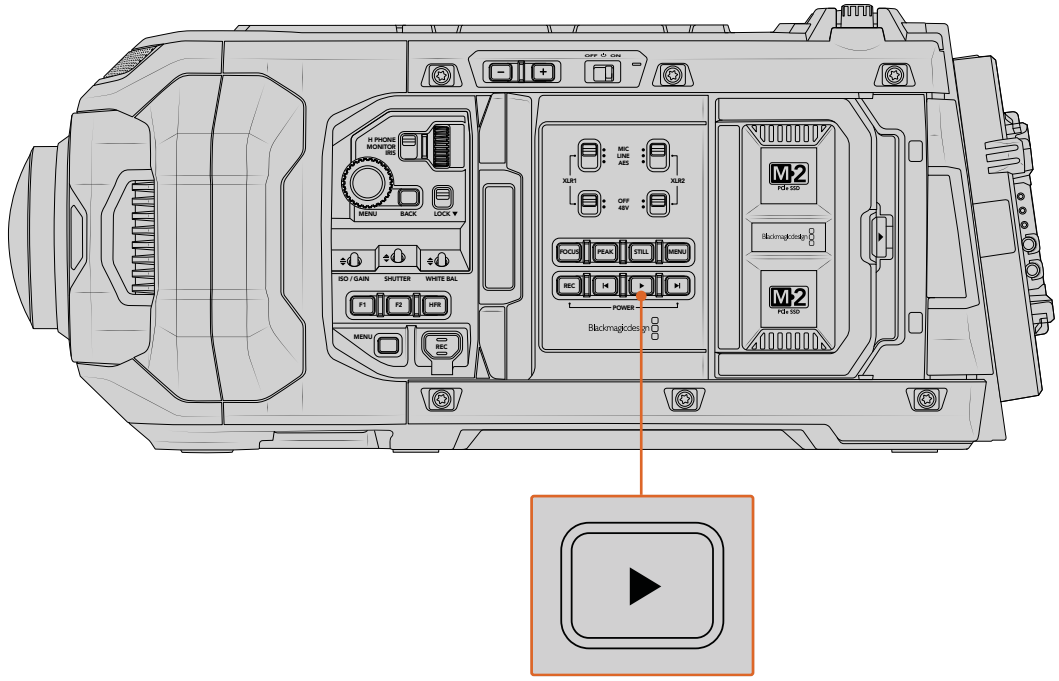
Start recording by pressing any of the red record buttons on your camera. Record buttons are located on the forward control panel, the internal control panel behind the fold out touchscreen, the assist station control panel and both LCD touchscreens. Press the record button again to stop recording.

The storage indicators show how much recording time is left. The time is displayed in hours:minutes:seconds based on your chosen frame rate. The media bar icon above the recording time will be either blue, white or red depending on its current status and will display the used space on the storage.



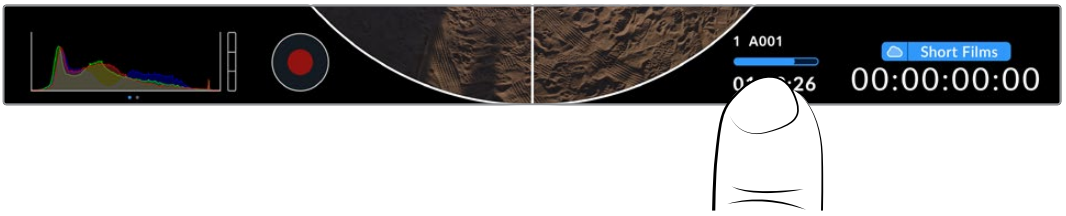
Playback

Once you have recorded your clips, you can use the transport control buttons to play them back. To play back the last clip that you recorded, push the physical play button on the left-hand side of the camera, located behind the fold out touch screen. The previous and next buttons can be used to navigate to other clips. To leave playback mode and ready the camera to continue recording, push the record button once.

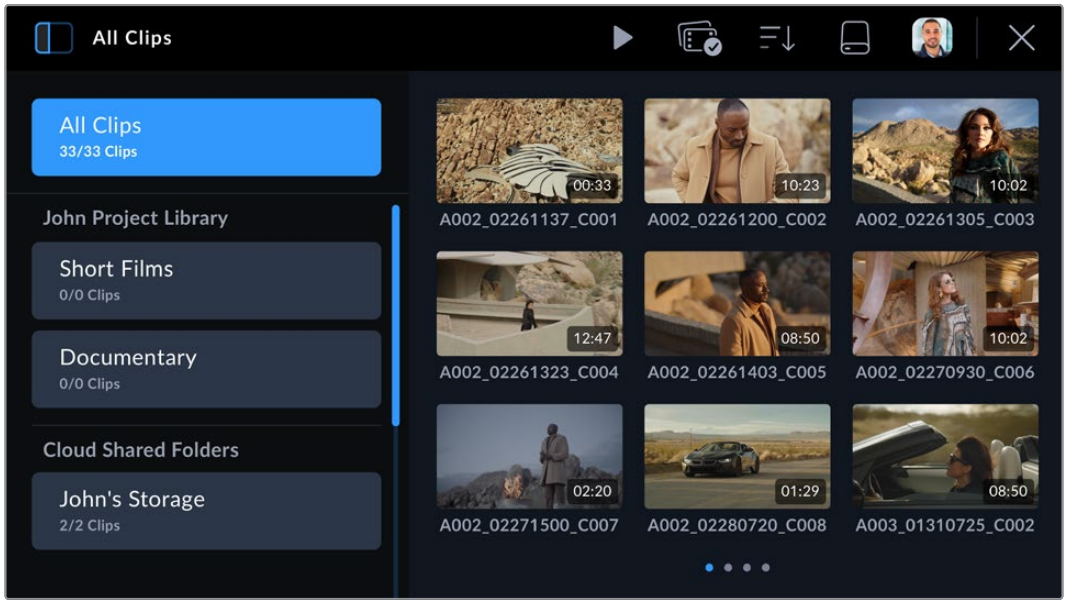


Press the forward and reverse buttons to skip to the start or end of clips. Press the ‘reverse skip’ button once to go to the start of the current clip or press twice to skip back to the start of the previous clip. Hold the ‘forward’ or ‘reverse skip’ button to play or reverse at 2x speed. Once shuttling forward or backwards, press the ‘fast forward’ or ‘reverse skip’ buttons twice for 4x, three times for x8 and four times for x16.

If a more extensive review of the footage you have shot is required, you will likely prefer to use the Media Pool. To open the media pool, tap one of the storage indicators at the bottom of your camera’s touchscreen display.



The main page browser displays thumbnails of all the recorded clips on all media connected to your camera. There are three rows of thumbnails and as more clips are added you can swipe to the next page of thumbnails using the touchscreen. Page indicators at the bottom of the display show how many pages of clips are available.



Connect the URSA Cine Immersive to a Blackmagic Cloud account and send footage directly to a DaVinci Resolve project.

If you wish to exit the media pool at any time, tap the X in the top right hand corner, or push any of the camera's record buttons once.

Transferring Clips

There are a number of ways you can transfer clips from your camera's storage. For example, you can plug URSA Cine Immersive into your computer or a network via a 10G Ethernet cable and copy clips from your camera's storage over Ethernet. You can even load the media module into a Blackmagic Media Dock connected to your network or computer, which lets you work with up to three media modules simultaneously.

When connecting to your computer over Ethernet, your computer will recognize the Blackmagic Media Module just like it would when plugging in a Blackmagic Cloud Store.

Your camera is set to use DHCP by default, so your computer or network will see the camera and assign an IP address automatically.

Enabling File Sharing

When you are setting your camera up for the first time and have connected to your computer or network via Ethernet, SMB file sharing will be disabled by default. This is because unlike a Blackmagic Cloud Store where you will likely want to share files with other users on the network straight away, you may not want to immediately allow the same level of access to the master camera files after an important day's shoot. For this reason, we decided to let you enable SMB file sharing yourself.

1. To enable file sharing, you will need to download and install Blackmagic Camera Setup on your computer. This setup utility is an administration software application that lets you update your camera and change admin settings. You can download Blackmagic Camera Setup from our website at: https://www.blackmagicdesign.com/support/family/professional-cameras
2. After downloading and installing the setup utility, connect your camera to the computer via USB and launch Blackmagic Camera Setup. Connect via the top panel USB port towards the rear of your camera.

3 To open the settings, click on the image of your URSA Cine Immersive or its settings icon.

4 Scroll down to the ‘network access’ settings and enable ‘file sharing (SMB)’. Then hit ‘save’ in the bottom right for these settings to take effect.

5 Now go to the URL setting underneath and click on the URL icon to automatically populate the address. The URL address will display your camera’s name as a local network location.

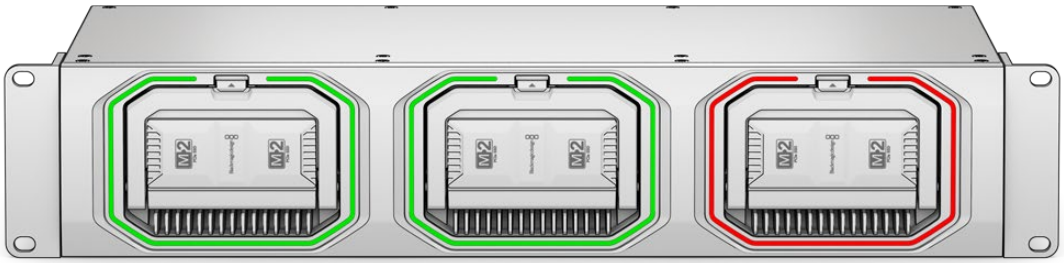
6 Click save.

To go straight to your clips, click on your camera’s URL name. A window will open asking you for a password or connect as a guest. Click on the ‘guest’ radio button and click ‘connect’.

7 Your computer will now automatically navigate to your media module in Finder on a Mac and Windows Explorer in Windows. The utility will remember your Ethernet setup so you only need to do this once when setting up for the first time and then the next time you plug in your camera via Ethernet, you will be able to access the media module straight away without touching any of these settings.

Using the Blackmagic Media Dock

The Blackmagic Media Module is designed to be quickly loaded and unloaded when you need to transfer clips to external storage or to swap to an empty module so you can keep recording. The Blackmagic Media Dock can be used to access footage on a Media Module without the camera.



Summary

Apple Immersive Video introduces a new approach to visual storytelling, and offers an alternative to traditional 2D cinema. It is not intended as a replacement but as an additional format that can be selected based on the needs of the project. As with any format, creative choices will vary depending on the story being told.

While new formats often involve complex workflows, URSA Cine Immersive is designed to simplify the process of capturing Apple Immersive Video. It provides a straightforward production path for immersive content and may also be used for visual effects and other applications.

This guide outlines the core features and workflows of the URSA Cine Immersive. It is intended as a starting point. As with any camera system, further exploration and testing are encouraged to achieve the best results for your specific production needs.

DaVinci Resolve Studio



DaVinci Resolve Studio is a comprehensive post production software that seamlessly integrates editing, color correction, visual effects, motion graphics, and audio post-production. It supports workflows tailored to the Blackmagic RAW Immersive format's unique characteristics – including ultra high resolution stereoscopic video, 90 fps frame rates, and Spatial Audio mixing. DaVinci Resolve Studio is fully equipped to handle the production, monitoring, and delivery of Apple Immersive Video, making it a powerful tool for immersive content creators.

Workstation Recommendations

Hardware recommendations

- Mac Studio M3 Ultra Desktop
- High refresh rate monitor (minimum 60Hz)

Minimum system requirements

- macOS 15 or later
- 64 GB of system memory
- For monitoring, Blackmagic Design Desktop Video 12.9 or later
- Apple Silicon based computer or GPU which supports Metal

Immersive monitoring requirements

- Apple Vision Pro
- High-speed wireless network (Wi-Fi 5 minimum; Wi-Fi 6 preferred)

Project Setup

Before ingesting Blackmagic URSA Cine Immersive raw media, it is good practice to go through DaVinci Resolve's project settings to ensure the media is correctly debayered and processed for optimal visual output and ease of editing.

Master Settings

It is crucial to indicate that you will be creating content for the Apple Vision Pro when setting up a project. This will ensure that all the correct settings and workflow options are visible to you as you prepare, assemble and deliver your project.

1. Launch DaVinci Resolve Studio.
2. In the lower right corner of the Project Manager window, click New Project.
3. Enter a project name, set the Media Location (if needed) and click Create.
4. Open the Project Settings by clicking the gear icon in the lower right corner of DaVinci Resolve Studio.

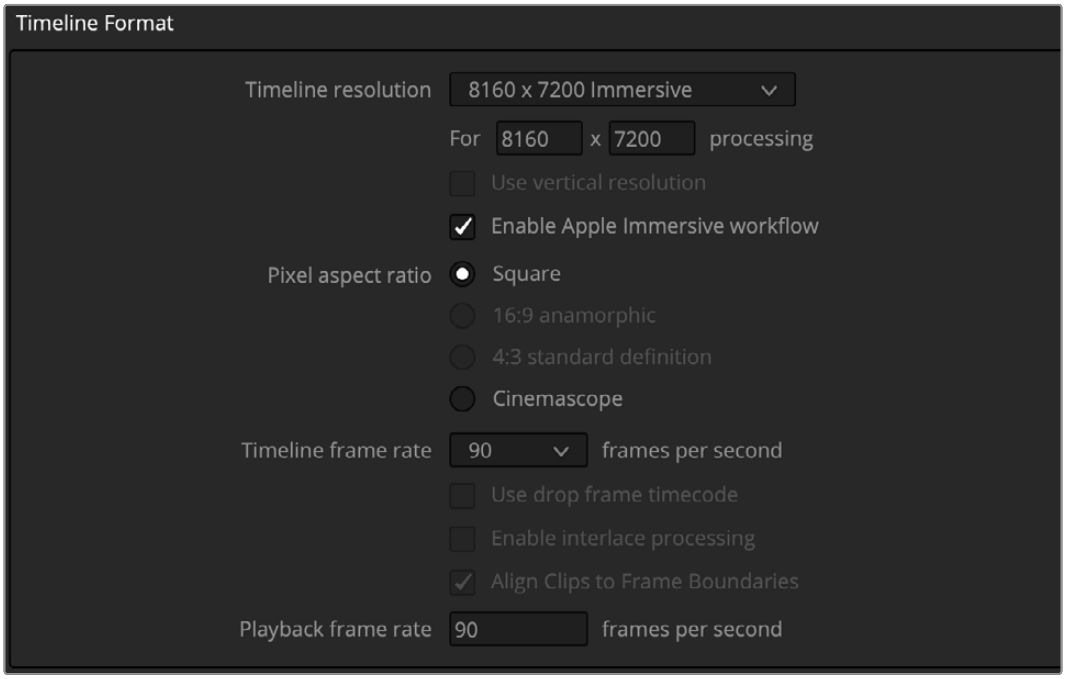
   First, you will set the resolution and frame rate of the project.
5. Select Master Settings in the sidebar.
6. In the Timeline Format section, select 'Enable Apple Immersive workflow'.



   This will configure multiple DaVinci Resolve settings to best accommodate immersive workflows. When creating an immersive project, ensure you always select this setting first.
7. Change the 'Timeline resolution' to '8160 x 7200 Immersive' to process the clips in their native sensor resolution.

   If working with limited processing power, choose '4080 x 3600 Immersive' to half the resolution for faster playback.
8. The 'Timeline frame rate' should be set to 90 frames per second. This parameter is automatically configured when you enable Apple Immersive workflows.
9. Set the 'Playback frame rate' to 90 frames per second for the most accurate reproduction of your Immersive video content.

   If working with limited processing power (or a monitor with a lower refresh rate), lower the frame rate for more stable playback. It is recommended you choose a frame rate that is a fraction of the native frame rate, such as 30fps or 60fps.



Video Monitoring

If you are using an external monitor (via a capture/playback device like Decklink or UltraStudio), configure the video monitoring settings based on your monitor's specifications.

1. In the Video Monitoring section, set the Video Resolution to match your monitor resolution. E.g. 1920 x 1080 HD or 3840 x 2160 Ultra HD.

   Though the native frame rate of immersive video is 90 fps, it is not a frame rate most broadcast monitors support.

2. Set the Format to match your monitor frame rate. E.g. 30p.

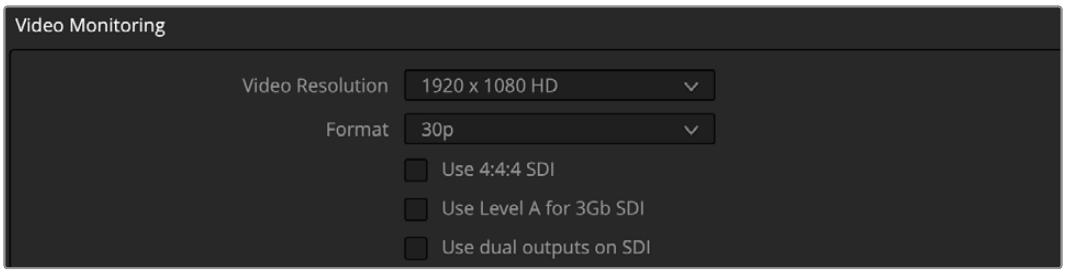


Image Scaling

The Image Scaling settings will ensure that the resolution and aspect ratio of the immersive media is correctly displayed in the viewer. These parameters would have automatically been configured after you enabled the Apple Immersive workflow option in the Master Settings, but you should still verify that they are correct.

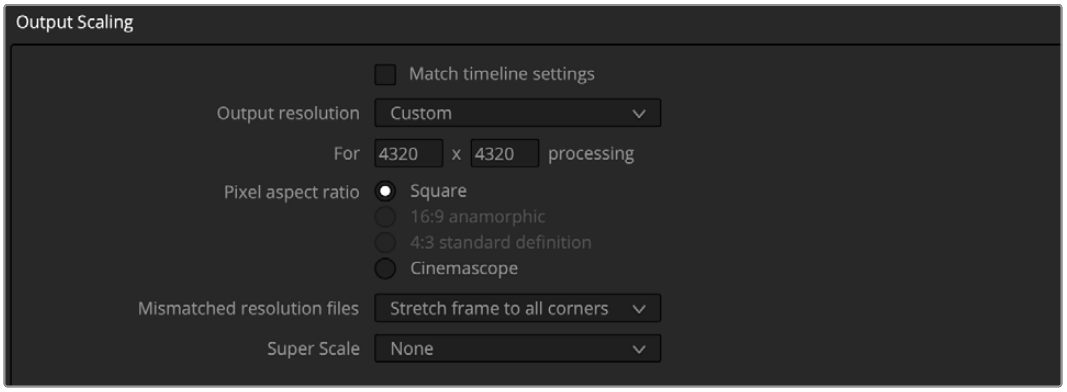
1. Select Image Scaling in the Project Settings sidebar.
2. In the Output Scaling section, ensure 'Match timeline settings' is deselected.

3 Ensure the ‘Output resolution’ is Custom and set to **4320 x 4320**.

Half the resolution (**2160 x 2160**) for smaller displays and/or faster playback.

These square aspect ratio resolutions are required for playing back Apple Immersive video content within DaVinci Resolve Studio and for streaming to the Apple Vision Pro (via wifi or developer strap).

4 Ensure ‘Mismatched resolution files’ is set to ‘Stretch frame to all corners’.



Color Management

Due to the need for custom control over the maximum output nit luminance, color management is encouraged with the use of Color Space Transform (CST) nodes in the color page. The goal of the color management settings in the Project Settings is to specify the timeline (or ‘working’) color space of the color page and to set the color space metadata that will be associated with the rendered file.

1 Select Color Management in the Project Settings sidebar.

2 Leave ‘Color science’ as DaVinci YRGB.

3 Set the ‘Timeline color space’ as DaVinciWG/Intermediate.

This is a future-proof wide gamut color space that supports HDR workflows, making it ideal for Blackmagic URSA Cine Immersive media and Apple Vision Pro deliverables.

4 Set the ‘Output color space’ as ‘P3-D65 ST2084 1000 nit.’

This color space features the appropriate gamut for monitoring the DaVinci Resolve viewers on a Mac Studio Desktop, which is typically Display P3. If you are using a grading monitor or a different Apple ICC profile, the output color space should be changed to match the color standard of your display.

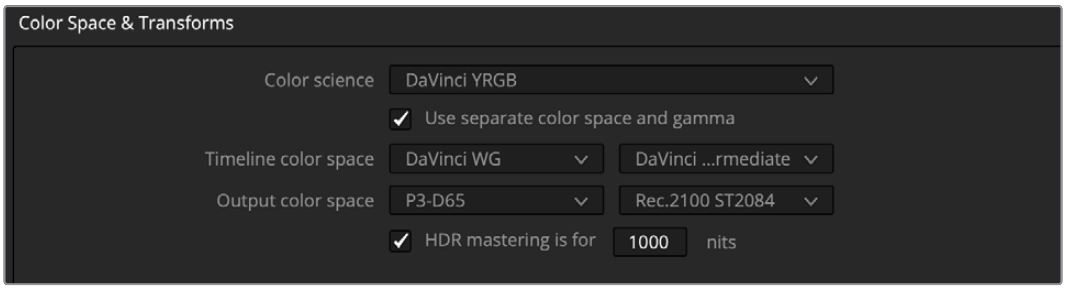
However, you will need to change the gamma (transfer function) to correctly map to the ICC profile.

5 Click ‘Use separate color space and gamma’ above the color space settings.

This splits the color space and gamma dropdown fields.

6 Set the output gamma (in the right column) as ‘Rec.2100 ST2084.’

7 Select ‘HDR mastering is for 1000 nits.’



To spend less time setting up your next Apple Immersive project, you can save the current settings as a preset.

8 Go to the three-dot options menu in the upper right corner of the Project Settings window and choose ‘Save Current Settings as Preset’.

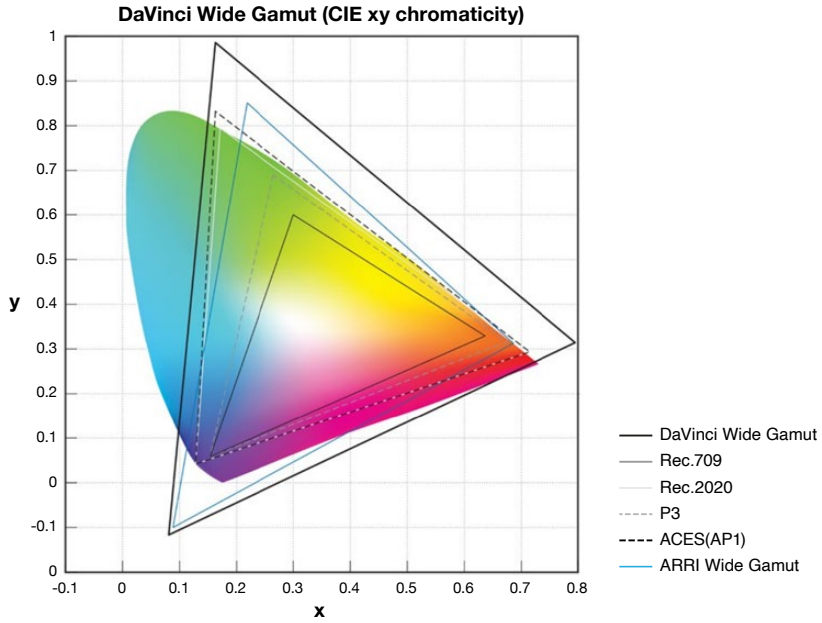
9 Name the preset and click Ok.

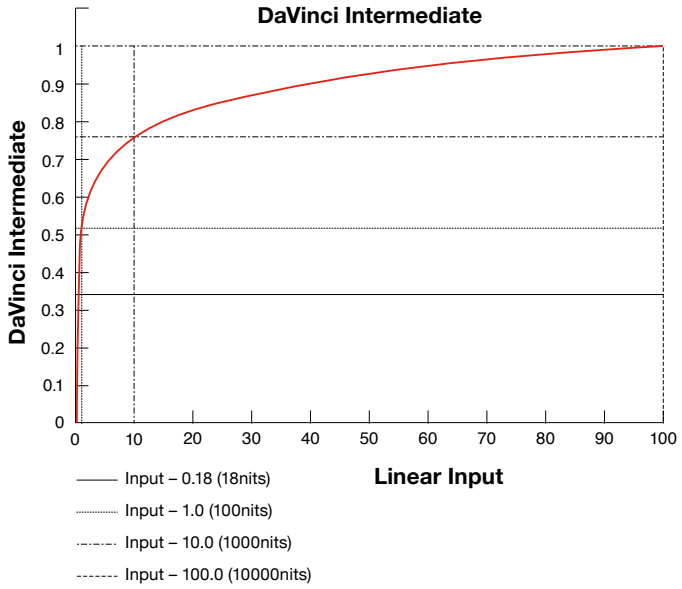
To quickly configure the project settings next time you work with Immersive media, open the Project Settings window, click the options menu and choose the listed preset.

10 Click Save to close the Project Settings.

About DaVinci Wide Gamut

DaVinci Wide Gamut is Blackmagic Design’s advanced internal color space developed for DaVinci Resolve, offering an exceptionally broad and precise environment for color grading and image processing. Paired with the DaVinci Intermediate gamma, it forms a unified working space that simplifies color management across mixed camera formats, ensuring consistent tone and saturation mapping from input to output. Ideal for working with HDR and Immersive media, this wide gamut allows colorists to make nuanced adjustments without risk of banding or tonal clipping when mastering for high-fidelity delivery formats.





Apple Immersive Processing

In the Project Settings sidebar, the Apple Immersive section allows you to set custom ILPD (Immersive Lens Profile Data) to modify how the immersive content is calibrated and projected.

1. Select Apple Immersive in the Project Settings sidebar.
2. Click Open Calibration Folder.
3. Add or update a custom ILPD file.
4. Click Update Lists to refresh the ILPD content accessible in the Media Pool.
5. Click Save to close the Project Settings.
6. In the Media Pool, right-click an immersive clip.
7. In the contextual menu, navigate to Apple Immersive Calibration and choose either the source ILPD (‘No Apple Immersive Calibration File Selected’), the custom ILPD, or the gap ILPD.

By default, the custom ILPD Calibration folder is located in the ProIM folder, alongside the default LUT folder.

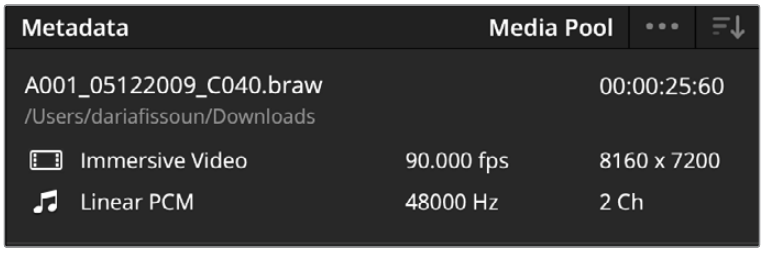
Ingesting

When ingesting raw immersive content, both eyes and all associated immersive metadata travel with the imported .braw clip.

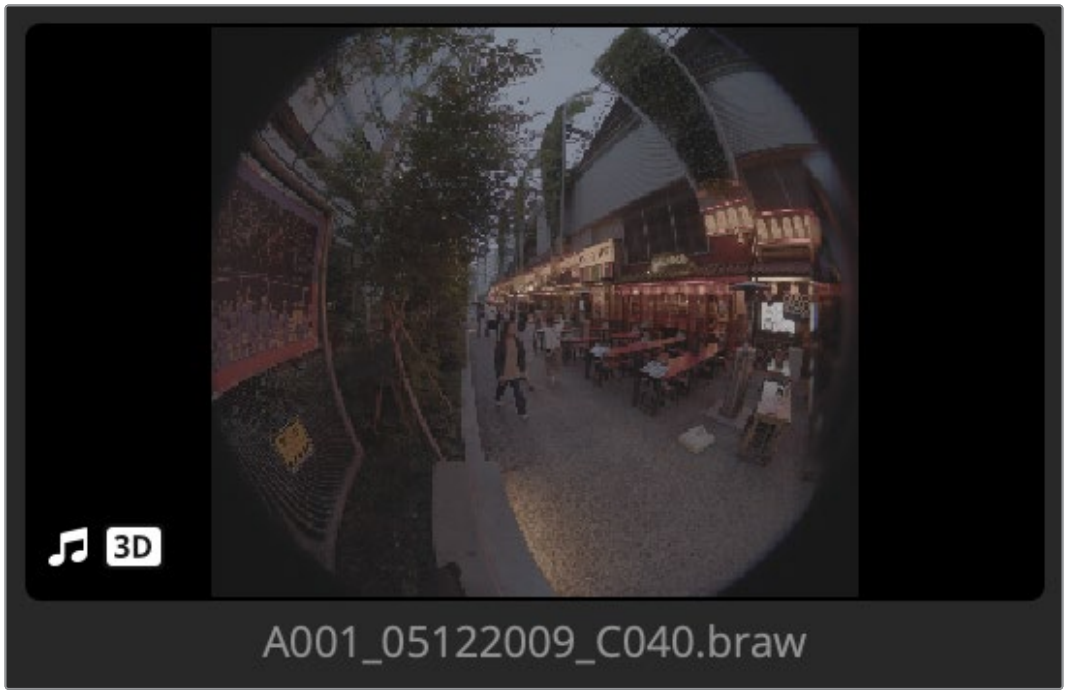
1. Go to the media page.
2. In the Media Storage panel, navigate to the location of the immersive video content.
3. Drag and drop the media into the Media Pool underneath.

TIP: To retain the original folder structure, drag folders from the Media Storage panel into the Media Pool sidebar. When dragged into the Master bin (or any bin you have created), folders and subfolders are retained.

When an Apple Immersive video clip is selected, the codec parameter in the header of the Metadata panel will indicate that it is an Immersive Video, alongside its native frame rate and resolution.



Additionally, a 3D icon in the lower left corner of the video thumbnail also confirms its status as a 3D clip.



NOTE: When you drag and and drop immersive media into the Media Pool for the first time, a dialogue window might prompt you to choose whether you wish to Keep Settings or Change Settings. Choosing Change Settings will set the Project Settings to automatically adopt the frame rate and resolution of the source media.

Editing

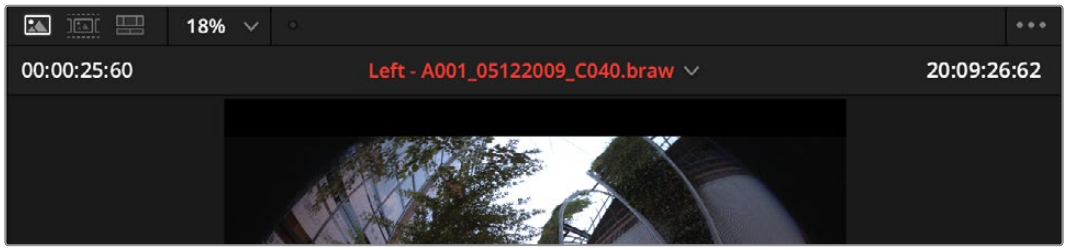
When working with immersive content, editing tools and actions remain mostly the same as when working with conventional 2D video formats. The following set of instructions describe edit features or actions that are unique to immersive video workflows.

Assembly

When previewing clips during timeline assembly, you have the ability to change which eye (left or right) you see in the source viewer. Otherwise, the process of assembling a cut on the timeline remains unchanged from standard NLE workflows.

1 Go to the edit page.

2 Double-click a clip to load it in the source viewer on the left. The label at the top of the viewer indicates which eye you are viewing.



3 To change the eye in the viewer, right-click the source clip in the Media Pool and choose Stereoscopic 3D > Stereoscopic 3D Mode > Left Eye/Right Eye/Both Eyes.

TIP: You can also hover over the thumbnails in the Media Pool to scrub clips in the source viewer if you have Live Media Preview enabled in the source viewer options.

4 Use I and O shortcuts to define a range in the source viewer.

5 Press Insert Clip in the toolbar (or use the F9 shortcut) to insert clips onto the timeline.

TIP: You can also drag and drop clips from the Media Pool into the timeline panel.

6 Use your mouse (in Selection or Trim Edit mode) to trim clips on the timeline.

Backdrops

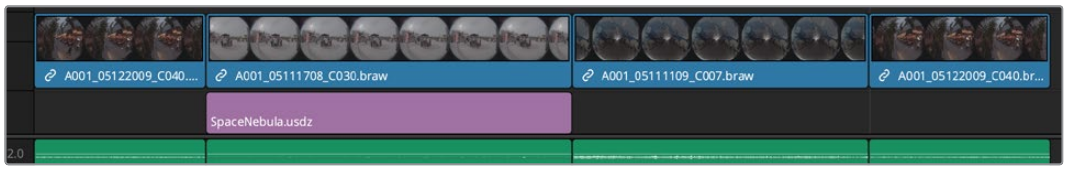
When you select ‘Enable Apple Immersive workflow’ in the Project Settings, a Backdrops track (‘BD’) automatically appears under the Video 1 track.

Backdrops refer to the area that the observer will see when they look beyond the perimeter of the immersive video content, which is typically limited to a 180–210 degree field of vision.

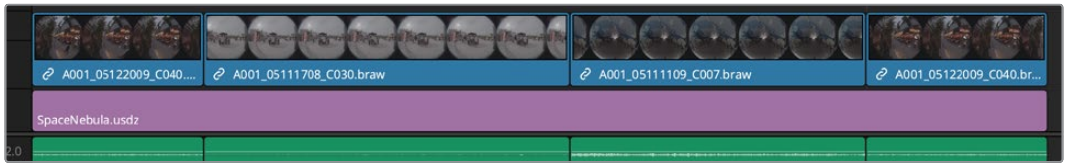
By default, this area is black, but you can insert custom background imagery (in the .usdz format) using the backdrops track.

1 Drag a .usdz backdrop file onto the Backdrops track to add it to the timeline.

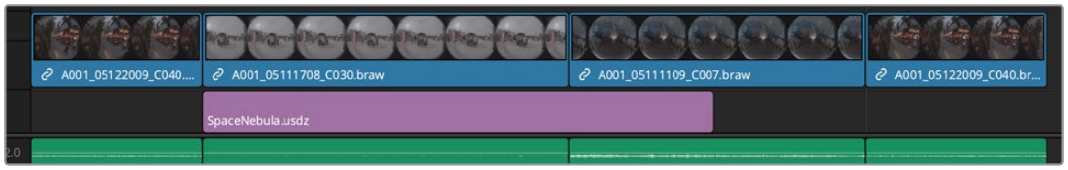
The duration of backdrop images must align to the clip boundaries in the video tracks. A single backdrop file can span one clip, multiple clips, or the entire timeline, but it cannot cut in the middle of an immersive .braw clip (aka 'L-cut').



One clip – correct ✓

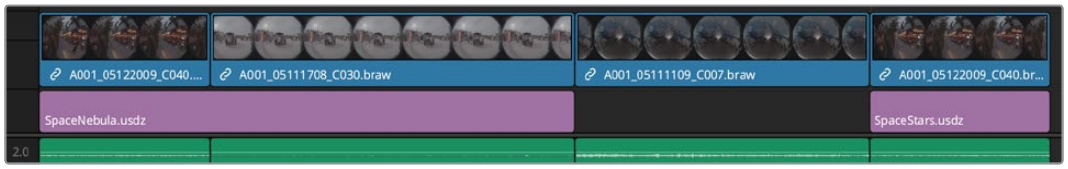


Multiple clips – correct ✓



L-cut – wrong ✗

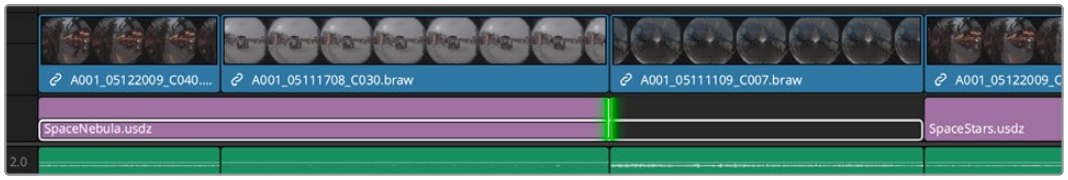
You can opt to have multiple backdrop files within a single timeline, or not add an image at all for the default black backdrop.



Different images and gaps – correct ✓

2 Use standard trimming controls to adjust the length of the backdrop file.

Drag the edge of the .usdz file (in Selection or Trim Edit mode) to adjust the length.



You may also use the , (comma) and . (period) commands to nudge the trim edge one frame at a time.

TIP: To ensure backdrops are always aligned to the Immersive Video edits, enable Snap mode in the edit page toolbar.

Using the Viewer

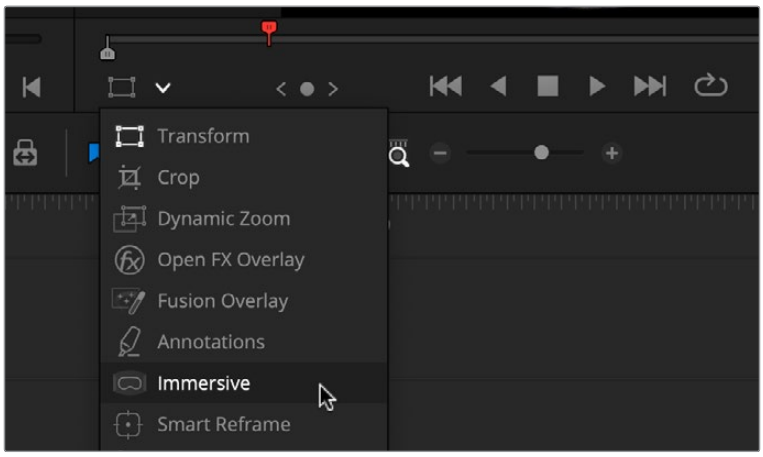
By default, immersive clips are displayed in the DaVinci Resolve page viewers in the circular ‘lens space’. The visual lens calibration used to project the image originates from the camera and is unique to every clip.



You also have the option to change the view mode to LatLong or Viewport.

1 In the lower left corner of the timeline viewer, click the dropdown arrow next to the onscreen controls button.

2 Select Immersive.



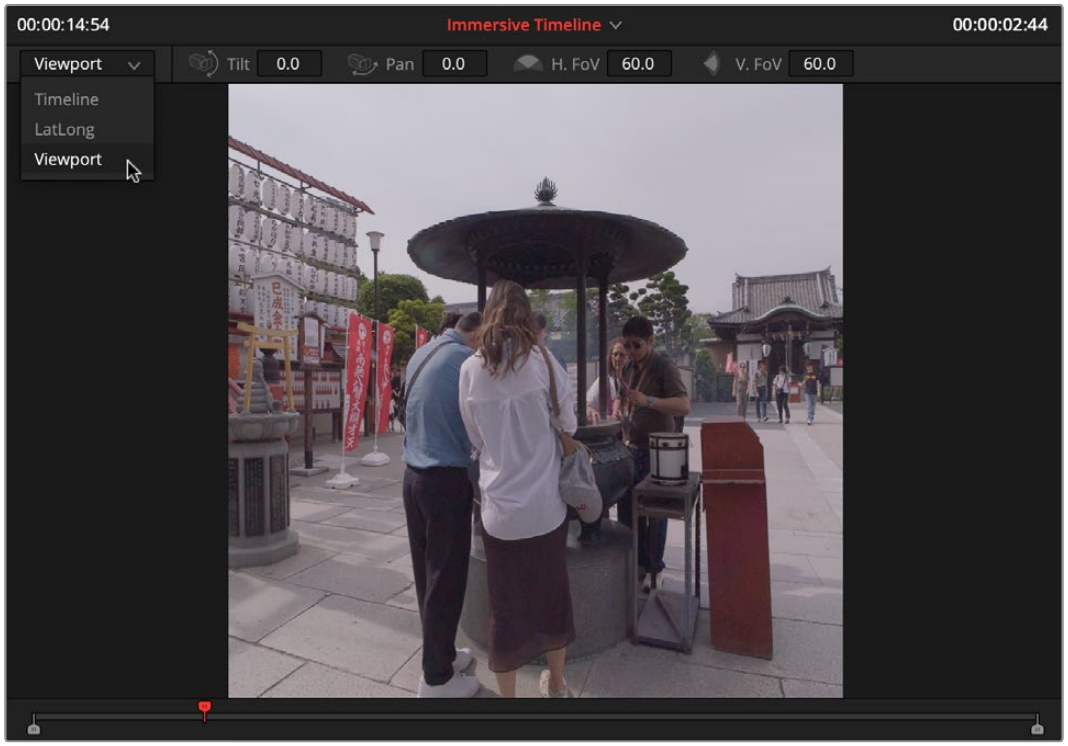
A new toolbar appears above the viewer with additional view options.

3 Click LatLong to see an equirectangular projection of the image. This mode warps the image to more closely resemble traditional 2D media proportions which eases certain post production processes, such as tracking and compositing.



4 Click Viewport to see a center cutout of the immersive scene in a square frame. This mode helps you clearly see the correctly-proportioned details of the scene on a standard monitor, though you will need to be mindful about not missing any off-frame action.

You can use the transform controls in the immersive toolbar at the top to tilt and pan the stereoscopic image in the frame, as well as to adjust the horizontal and vertical fields of view.



Transitions

To transition between two immersive video clips, you can either perform a direct cut, use the handles on either side of the clip to fade in or fade out, or use a Dip To Color Dissolve effect. Transitions should not be baked into the footage, but are rather rendered by the Apple Vision Pro itself during playback.

1 Open the Effects panel.

2 In the sidebar, navigate to Toolbox > Video Transitions.

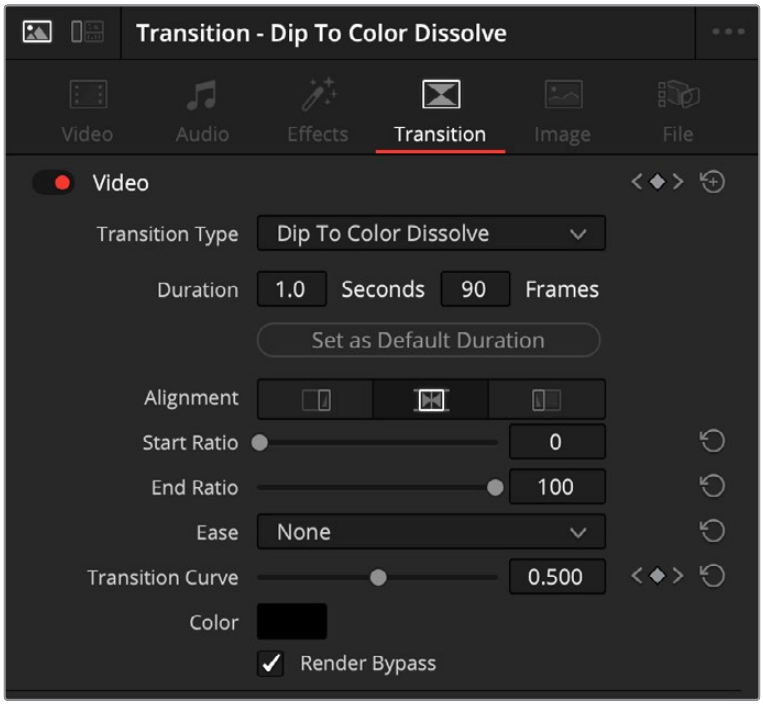
3 Drag Dip To Color Dissolve onto a cut point between two clips.



NOTE: You must have enough trim data beyond the cut points of both clips to apply a transition. If there is insufficient trim data ('handles'), the transition will not apply.

4 In the Inspector panel, navigate to the Transition tab and select Render Bypass. This will ensure that the dissolve is correctly rendered by the Apple Vision Pro.

5 You may then use the remaining parameters in the Dip To Color Dissolve Inspector to modify the duration, alignment, animation speed, and color of the transition.

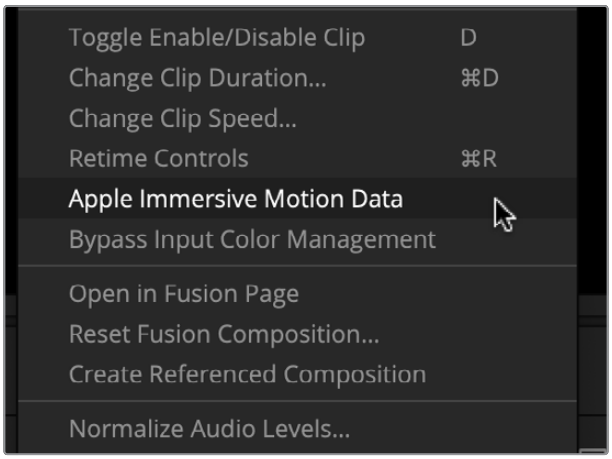


Apple Immersive Motion Data

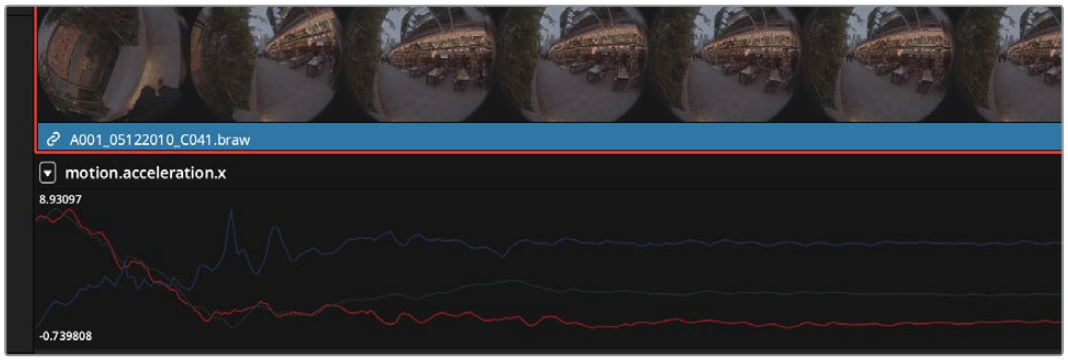
When working with immersive content, quick or sudden camera movement can be disorienting to the viewer. To ensure viewer comfort, avoid using takes with quick pans/tilts, rotating motion, or uneven horizons.

The Apple Immersive Motion Data feature graphs the x, y, and z acceleration and gyroscope position values to objectively demonstrate where camera motion and movement may be too strong. Editors can use this information to avoid using disorienting sections of footage.

1. Right-click an immersive clip on the timeline.
2. Choose Apple Immersive Motion Data.

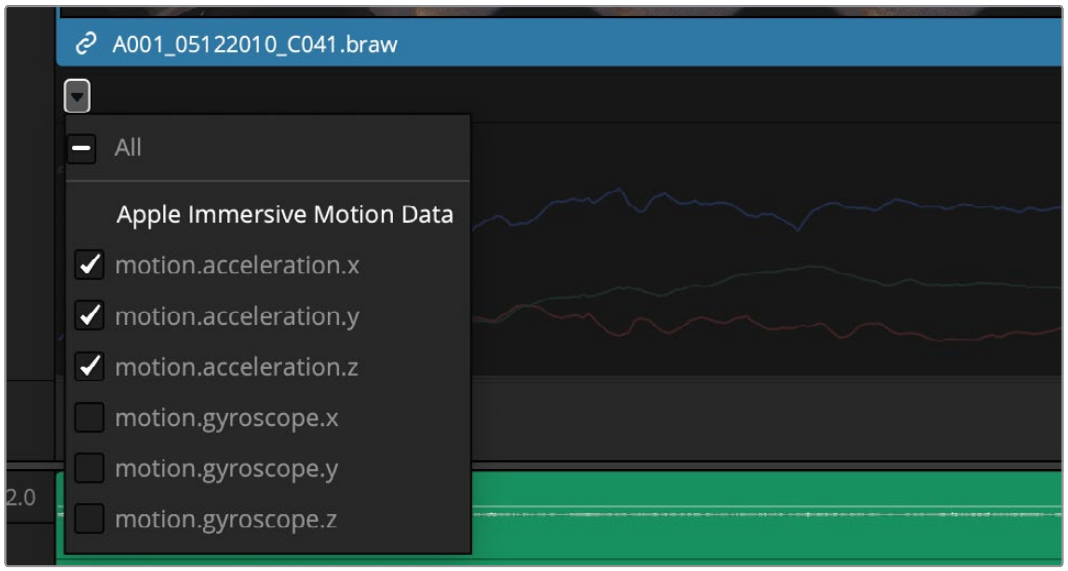


A graph appears in the video track showing the x, y, z acceleration rates.



In this example, the camera was recording as the camera operator was setting it up. The spikes in the graph indicate sudden movement as the camera was hoisted, rotated and placed on his shoulder. Ideally, you'd want to use the latter section of the video with the flatter readouts, after the camera has been steadied.

3 In the upper left corner of the graph, click the displacement arrow.



You can enable or disable readouts based on the content of the shot and the nature of the camera motion.

Titles

In the edit page, you can add text to an immersive scene using any of the title tools in the Effects panel – Text, Text+, MultiText, Scroll, and any of the lower third varieties.

When placed above an immersive clip, the title generator will use the clip's ILPD (Immersive Lens Profile Data) to accurately distort the text to match the lens warp of the scene.

If no ILPD is available (for example, when adding text over a black backdrop or immersive CGI scene), default lens distortion will be applied to ensure the text still appears appropriately warped when viewed in the Apple Vision Pro.

1 In the Effects panel, navigate to Toolbox > Titles.

2 Drag the Text generator into the timeline. Place it on a video track above an immersive clip to incorporate it into the scene.



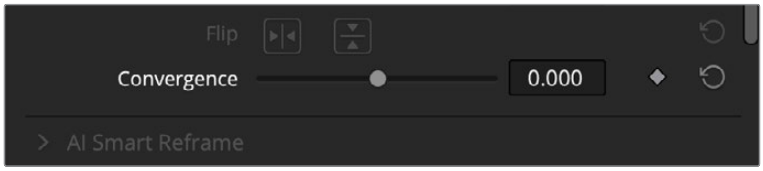
As with backdrops, Text generators cannot cross clip boundaries. As each clip is rendered with an individual lens space, any overlapping titles risk being warped between shots.

3 Trim the title to the video edit point(s).

4 Use the Inspector to write and format the title text.

5 Go to the Settings tab and use the Convergence slider to position the text in 3D space.

Moving the slider to the left pushes the title back in space, while moving it right brings it forward.



Convergence is best observed when monitoring on the Apple Vision Pro. If using other monitoring means, you can change how the text convergence will appear in the viewer by changing the Vision mode of the underlying video layer in the color page.

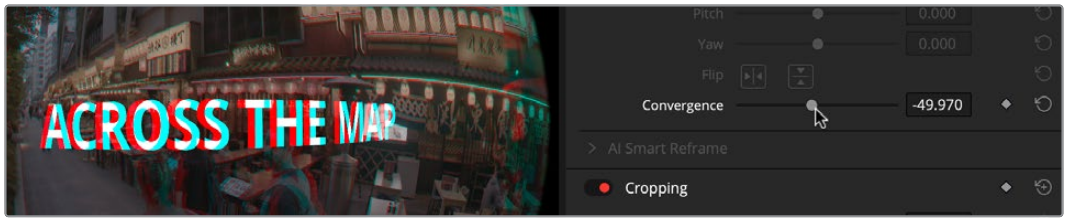
6 Go to the color page.

7 Open the 3D palette.

The Vision column in the centre of the palette allows you to change the stereoscopic standard used to project the immersive media.

8 Select the Stereo tab.

9 Change the Out mode to Anaglyph (Color).



Changing the text generator's Convergence parameter in the edit page's Inspector panel will now appear more intuitive if viewed through anaglyph (or anachrome) glasses.

Creative Considerations

During these early days of Blackmagic URSA Cine Immersive camera and Apple Vision Pro production, filmmakers are encouraged to experiment with the format to identify best practices, establish new rules and discover workflows that traditional filmmaking had not previously imagined.

With that in mind, some aspects of immersive production have already been flagged for special consideration to ensure maximum audience absorption and visual comfort.

1 Editing should begin in pre-production. Due to the special considerations for shot length and audience motion comfort, filmmakers need to capture scenes with the final edit in mind.

2 Avoid or limit using clips with rapid camera movements, as these may disorient the viewer. Instead, opt for locked shots, or slow pans and dollies.

3 Off-balance or extreme camera angles can also be disorienting – opt for takes that are level to the subject, and at a comfortable distance (not too close).

4 New environments take longer to absorb in immersive media than when viewing a 2D scene. Cuts should be slower to allow the viewer time to adapt to new surroundings.

5 When cutting between clips, be mindful of where the viewer would have been looking in the previous shots. Important action shouldn't be taking place on the left side of the frame if the audience's attention was previously towards the right.

6 Spatial Audio drives much of the storytelling – treat it as another character in immersive productions.

VFX Compositing

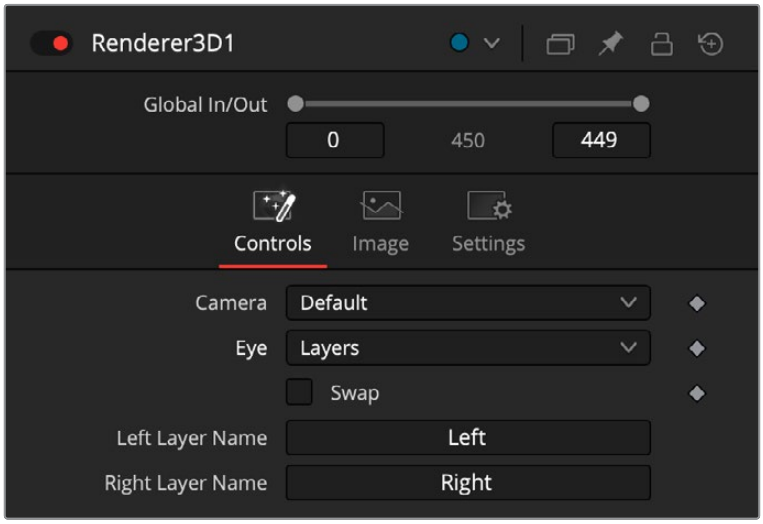
When ‘Apple Immersive workflow’ is activated in the Project Settings, dedicated Fusion parameters allow for seamless stereoscopic workflows and immersive-compliant compositing. MediaOut nodes feature left and right eye output options and the Immersive Patcher tool allows for equirectangular projection and composition.

MediaOut nodes

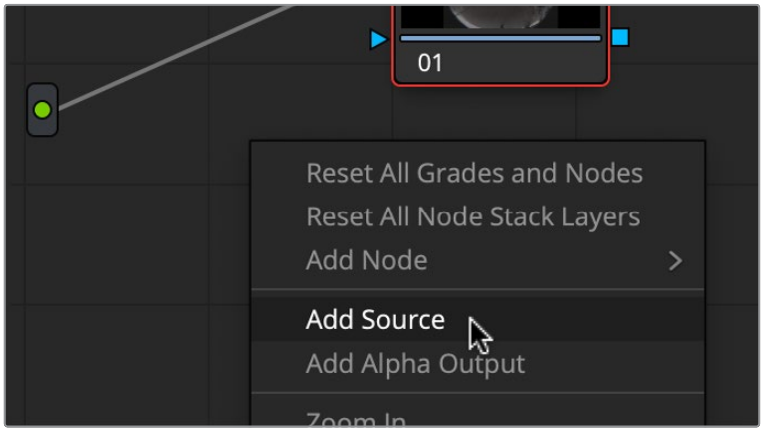
When you first enter the Fusion page while working on an immersive clip, the Nodes graph contains two MediaIn nodes connected to two corresponding MediaOut nodes.



In the MediaOut Inspector, the Stereo Eye parameter can be used to route the eye signal output.



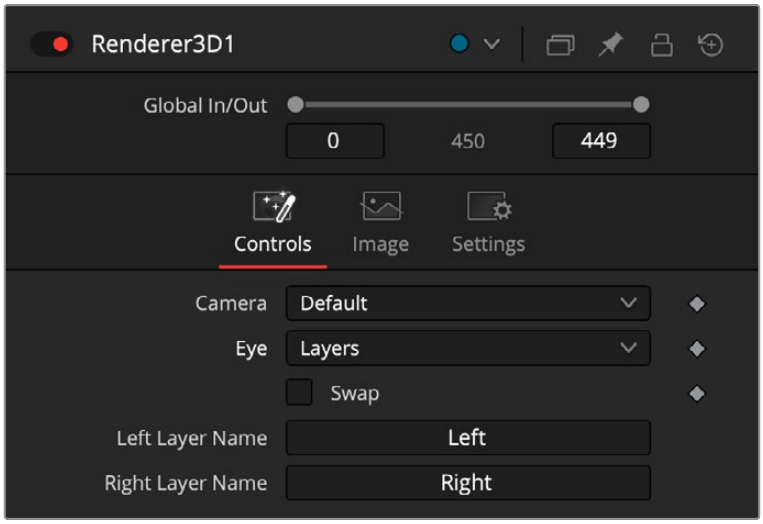
By default, MediaOut1 is set to the Left eye and MediaOut2 is set to Right. This configuration allows you to immediately start assembling stereo-compliant composites and graphic overlays. The dual MediaOut nodes also result in two signal variants being sent to the color page’s node editor for further image refinement. You can access the MediaOut2 signal in the color page by right-clicking in the node editor and choosing ‘Add Source’.



When the Stereo Eye parameter is set to Auto, the MediaOut node combines the signals and assigns them to the appropriate eyes on the timeline. This allows you to use Fusion's multilayer workflow to send left and right signals to a single MediaOut node.

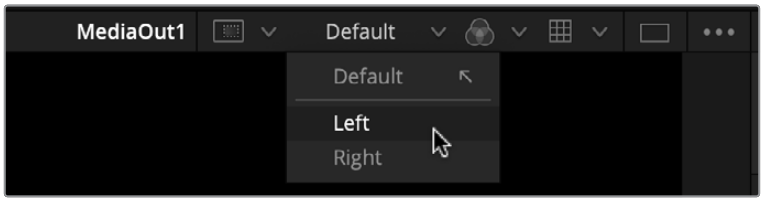
Renderer3D

When working with 3D graphics or CGI on an immersive timeline, the Renderer3D node features multiple eye setups, and the ability to process left and right layers.



This makes it possible to create complex 3D titles or entire CGI scenes and send them directly to a stereoscopic timeline in immersive projects.

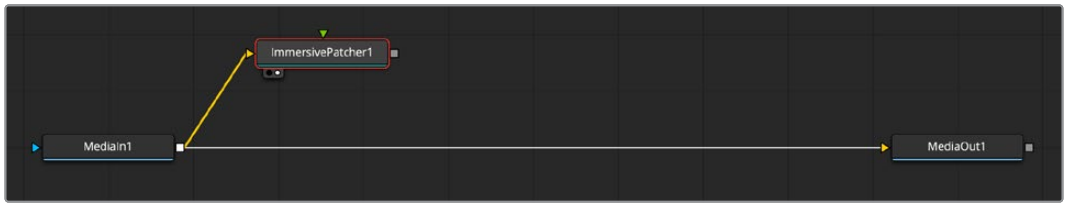
When working with this setup, you can use the drop down menu in the upper right corner of the Fusion viewers to alternative between viewing the left and right eye of your composition.



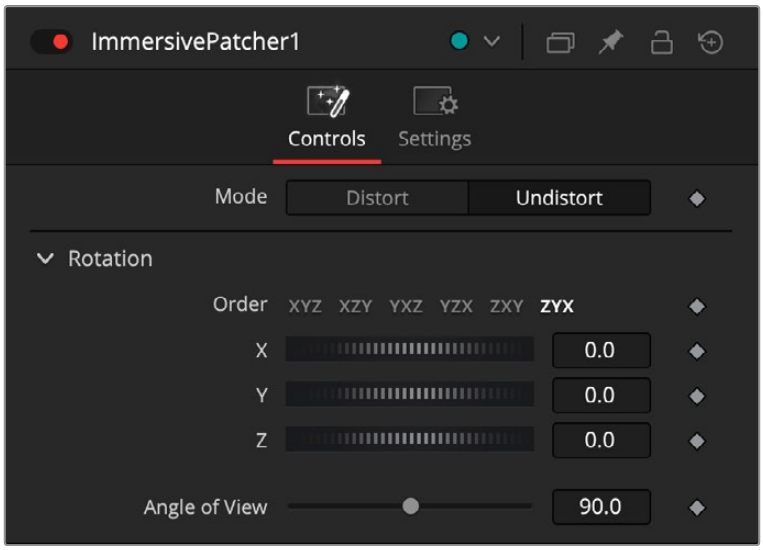
Compositing with the Immersive Patcher

The Immersive Patcher tool allows you to temporarily ‘undistort’ a section of an immersive clip to allow for easier tracking, painting and compositing on a flat plane. Once finished, you can use a second Immersive Patcher to re-distort the immersive signal and integrate the newly composited elements into the scene.

1 Add Immersive Patcher to the node pipeline.



2 In the Inspector, click Undistort.



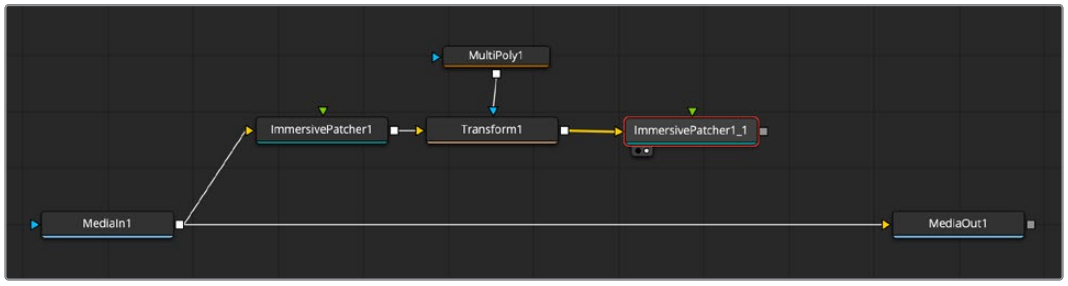
3 Use the X, Y, and Z rotation parameters to choose a section of the immersive image to flatten.

4 Proceed to composite as you would a standard 2D video signal. This includes (but is not limited to) using standard paint methods to remove unwanted objects, sampling sections of the frame with transforms and polygons for set extension, or applying mattes, text, and external images/video layers to replace signs or screens.

Once finished, you need to re-distort the flat image to match the original immersive warp.

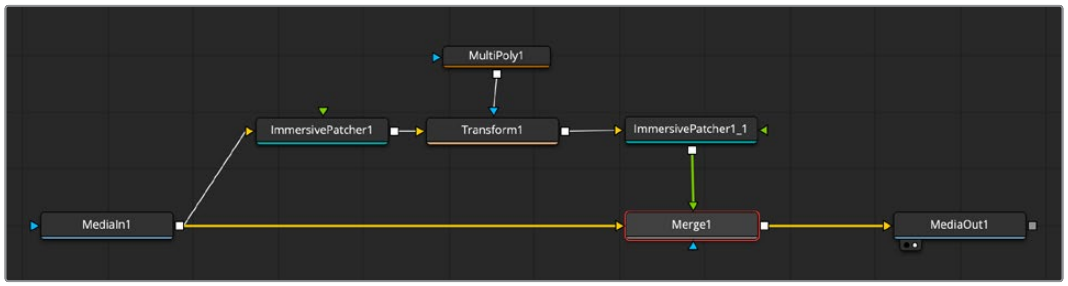
5 Copy the Immersive Patcher to retain the original rotation and angle-of-view information.

6 Paste the Immersive Patcher after the composite nodes and connect to the end of the composition pipeline.



7 In the Inspector, click Distort.

8 Merge the composited patch to the original clip to integrate the effect into the immersive scene.



If working collaboratively, you can exit the Fusion page and continue to edit, mix and grade the immersive timeline in DaVinci Resolve.

If performing immersive turnovers, render out the individual eyes as EXRs in ACES AP0 to retain maximum properties for immersive delivery.

Color Grading

This section covers the optical, color, and 3D tools that you can use when working with immersive media in DaVinci Resolve's color page.

Due to the unique nature of immersive content, not all tools will be possible to use during post-production. Other tools are currently unavailable, but are in the process of being adapted to support immersive content.

Noise Reduction

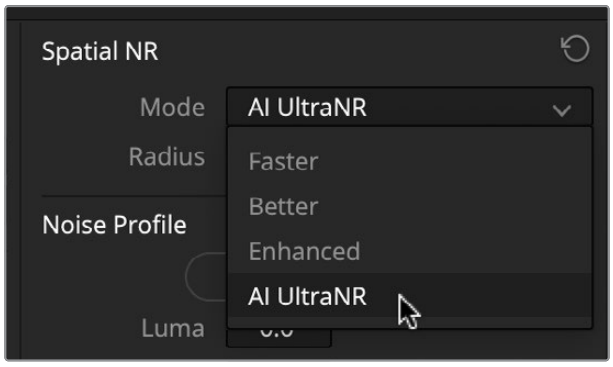
Noise reduction is possible when working with immersive video. In the Motion Effects palette, Spatial NR analyzes the noise pattern of the source media in order to maximally reduce it while retaining original image detail.

1 In the color page toolbar, open the Motion Effects palette.

In the Spatial NR column, you can select which noise reduction mode you wish to use.

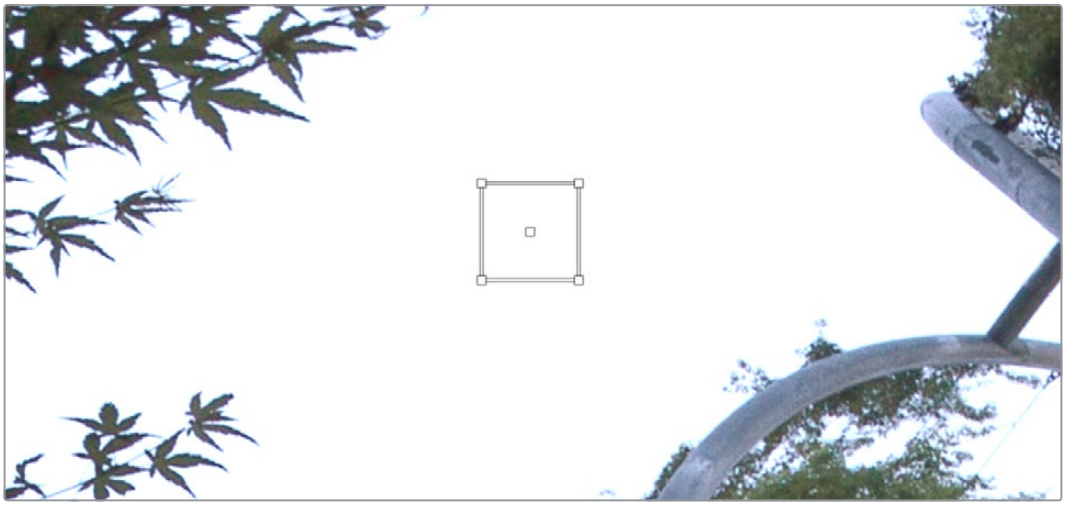
- *Faster*, *Better*, and *Enhanced* modes will all run their pattern analysis within a given radius size and configure their output based on the set threshold. They are listed in order of processing complexity.
- *AI UltraNR* is an AI-driven noise pattern identifier that will extract the noise pattern from the signal by locating and analyzing the "flattest" section of the frame for pure noise.

2 Set the spatial noise reduction mode to AI UltraNR.



3 Click Analyze.

A bounding box appears in the viewer, identifying the “emptiest” area of pixel data in the given frame. With no tangible visual content, the data within the square is ideal for extracting the digital noise produced by the camera sensor.



You can reposition and resize the bounding box in the viewer if you wish, but it is generally not advised—AI UltraNR is better at detecting optimal analysis regions than our eyes.

You can also manually adjust the Noise Profile Luma and Chroma Threshold settings, but again, AI UltraNR tends to automatically achieve the ideal mix of thresholds for clean noise reduction.

Primary Grading

Primary color grading operates the same as it does with regular video footage. Use the same considerations for normalization, balancing, and matching when using primaries color and log wheels, the HDR palette, and the HSL Curves. The Color Slice and Color Warper palettes also work as expected.

Secondaries

Pixel-based selection methods, like the Qualifier, work as expected with immersive media.

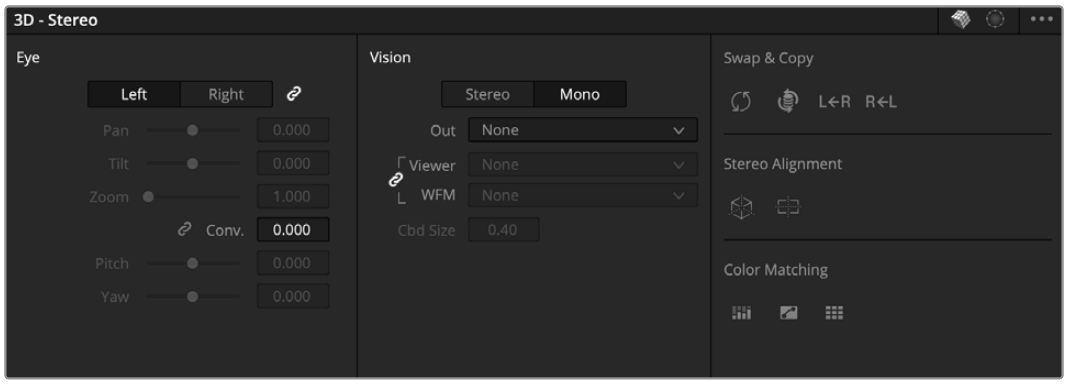
Object- or window-based selection can only work accurately in a flat equirectangular (or viewport) environment, meaning that most secondary methods that require precision in qualification shape are currently unavailable. This includes Power Windows (and, by extension, the Tracker palette) and the Magic Mask tool.

Sizing

There are no sizing controls (i.e. zooming and panning) available for immersive content.

3D Palette

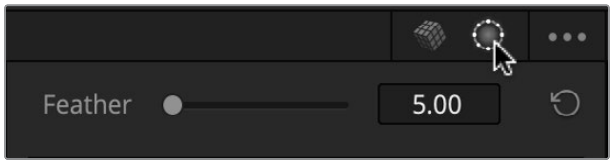
The color page's 3D palette features controls for managing a variety of 3D camera and graphics formats. Unlike with other stereoscopic formats, you do not need to adjust the alignment of the eyes, because the Blackmagic URSA Cine Immersive camera lenses, focal length and irises are fixed.



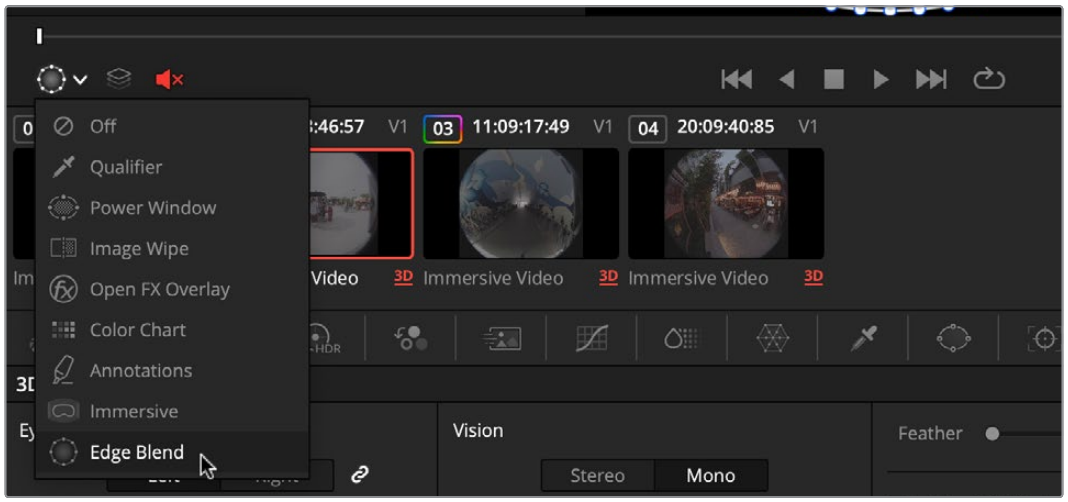
The Edge Blend mode is exclusive to immersive content and allows you to adjust and soften the perimeter of the immersive video frame. This can be used to hide visible film equipment and to limit the observer's field of vision. Beyond this edge, the observer will see the backdrop.

1 Open the 3D palette.

2 In the upper right corner, change the mode to Edge Blend.



3 Ensure the viewer's onscreen controls are set to Edge Blend.



You should see a circle of control points around the perimeter of the video frame, with an anchor handle in the center.

4 Drag a control point to change its position.



Or right-click to delete a point.

5 In the 3D palette's right column, use the Feather parameter to soften the edge of the frame.

6 Select 'visionOS Effect' to ensure the new edge data is correctly processed during playback on the Apple Vision Pro. If not selected, the edge will be baked into the rendered video file.

7 In the left column, use the Left and Right tabs to switch between eyes in the viewer.

Note that many of the controls in the 3D palette are greyed out. Parameters throughout DaVinci Resolve will appear grey when they are not applicable to the media you are actively working on, which in this case is the immersive content.

8 In the center column, use the Stereo and Mono tabs to switch between single and double lens viewers of the immersive video clip.

CST Color Management

Using the Color Space Transform (CST) effect on nodes in the color page's node editor will allow for accurate color mapping and custom control over the maximum output nit luminance of the timeline.

The recommended workflow involves four CST nodes:

1 INPUT CST – camera to working color space.

Color correction occurs between these two nodes.

2 OUTPUT CST – working color space to HDR Mac display.

3 MONITOR CST – 1000 nits to 250 nits for HDR monitoring.

4 AVP CST – 250 nits to 108 nits for Apple Vision Pro monitoring and delivery.



CST Timeline Mapping

First, you will map from the camera source input color space to the 'working' timeline color space. The Blackmagic URSA Cine Immersive camera operates in the Blackmagic Design Wide Gamut Gen 4/5 color gamut with the DaVinci Intermediate log curve.

1 Open the Effects panel.

2 Drag Color Space Transform onto an empty node (01) in the node editor.

When mapping from the input color space to the timeline color space, this node should be among the first nodes in the node tree, before any grading operations.

In cases where you are mapping from the input color space directly to the monitoring or delivery color space, this node should be among the last nodes in the node tree, after grading operations.

3 Label node 01 as **INPUT CST**.

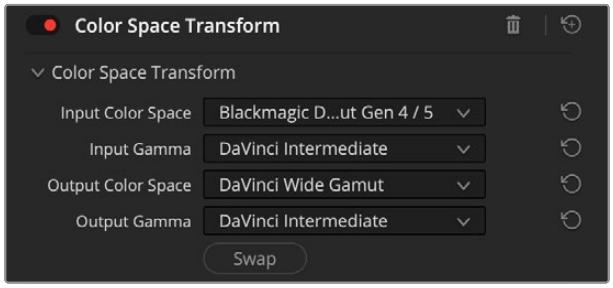
NOTE: The node labels used in these instructions are made for ease of reference to the documentation and screenshots. You may use whichever naming convention works best for you, or bypass labelling altogether.

4 Set the Input Color Space to Blackmagic Design Wide Gamut Gen 4/5.

5 Set the Input Gamma to DaVinci Intermediate.

6 Set the Output Color Space to DaVinci Wide Gamut.

7 Set the Output Gamma to DaVinci Intermediate.



8 Create a new serial node (02).

9 Copy and paste the first node (01) onto the new node (02).

10 Label node 02 as **OUTPUT CST**.

You will want the output color space and gamma to be the input in this new serial node to ensure that the timeline color space of the node tree is consistent.

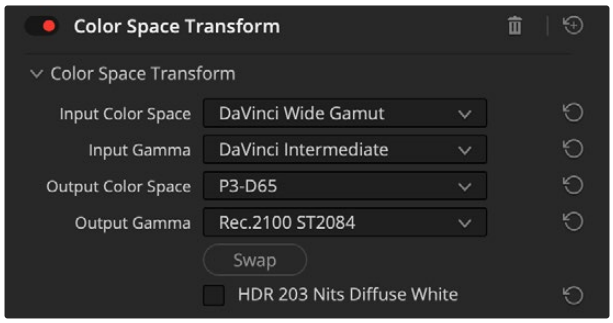
11 Press the Swap button to switch the input and output parameters.

The Input Color Space and Gamma are now appropriate for this CST node.

12 Set the Output Color Space to P3-D65. This is the current default gamut of Mac displays.

13 Set the Output Gamma to Rec.2100 ST2084, which is the HDR standard for a 1000-nit signal.

All grading operations will occur between these two nodes. But as you are working with node-based color management, you also have the unique ability to perform operations outside the perimeters of the timeline color management, which many colorists take advantage of to apply specific grading operations, effects or third-party elements.



ICC Profiles

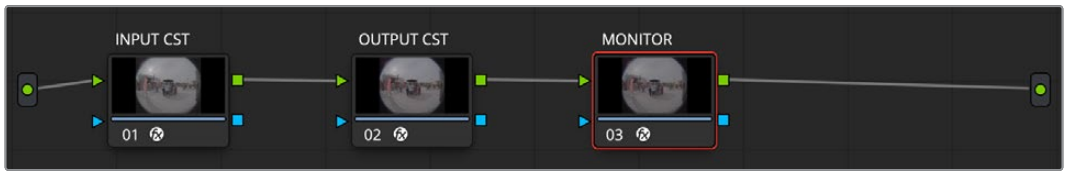
To further ensure that the Blackmagic URSA Cine Immersive signal is correctly mapped to your Mac display, go to the System Settings > Displays and verify that the color profile Preset is set to Apple XDR Display (P3-1600 nits).

CST Grading Monitor Mapping

You can use additional CSTs to temporarily limit the maximum nit output values for monitoring during grading.

1 Create a new serial node (03) after the color management OUTPUT CST (02).

2 Label node 03 as **MONITOR**.



3 Add the Color Space Transform effect to the MONITOR node (03).

The input values have to match the outgoing signal from the previous CST.

4 Set the Input Color Space to P3-D65.

5 Set the Input Gamma to Rec.2100 ST2084.

The purpose of this CST node is only to map the luminance of the image to a smaller nit value, which means the input and output parameters can remain the same.

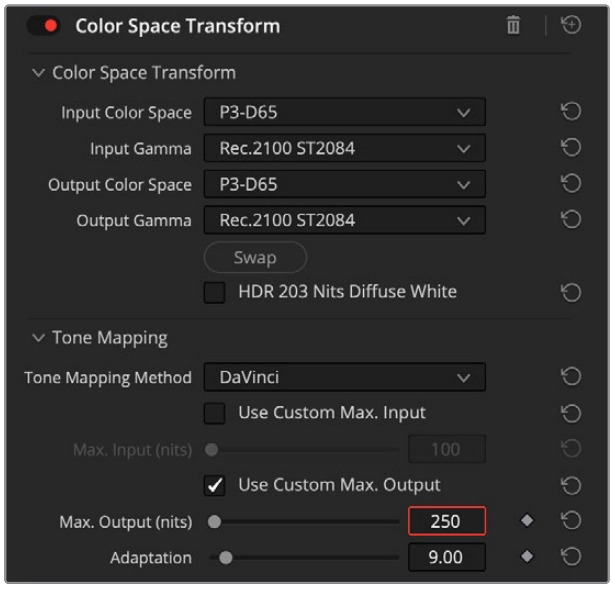
6 Set the Output Color Space to P3-D65.

7 Set the Output Gamma to Rec.2100 ST2084.

NOTE: In some circumstances, it is unnecessary to set the color space or gamma of a CST if the input and output are identical (as no mapping occurs). In this case, it is a necessity because the application of the custom nit mapping is affected by the value and shape of the signal's log curve.

8 Select 'Use Custom Max. Output'.

9 Set the Max. Output (nits) to **250**.



The 1000 nit signal will be now mapped (using the DaVinci Tone Mapping Method) to 250 nits.

This will result in an image that will closely match the rendered Apple Vision Pro signal on an HDR grading monitor (such as Flanders XMP310, Sony BVM-HX310, etc).

However, to monitor your project on an Apple Vision Pro, you will need to perform one more tone mapping operation.

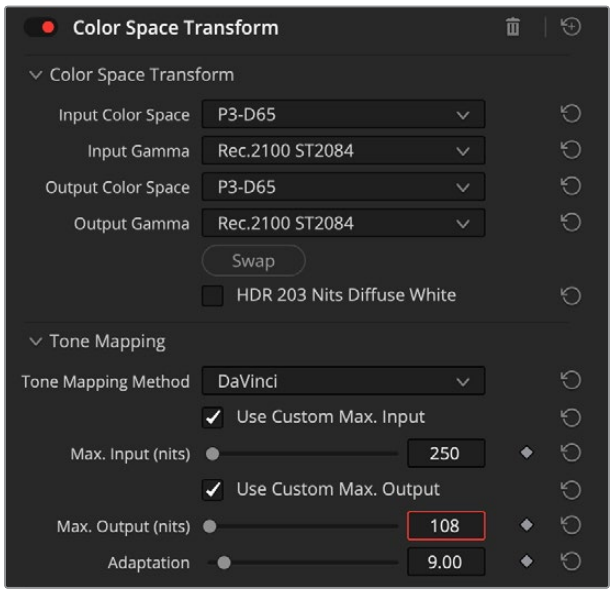
CST Apple Vision Pro Monitoring

A final CST node can be used to allow for live monitoring, preview and final delivery to Apple Vision Pro devices.

1. Create a new serial node (04) after the MONITOR node (03).
2. Copy and paste MONITOR node (03) onto the node 04. You will be using the same input and output parameters, and only changing the nit count.
3. Label node 04 as **AVP**.



4. Select ‘Use Custom Max. Input’.
5. Set the Max. Input (nits) to **250** to correctly interpret the mapped signal from the MONITOR node.
6. Set the Max. Output (nits) to **108**, which is the maximum nit value for monitoring and delivery to the Apple Vision Pro.



You may now proceed with the grade, disabling the CST nodes if (and when) needed based on your workflow. For example, during your initial grading pass, in which you will be using your external grading monitor, you should disable the AVP node (04) to ensure you are correctly seeing the 250-nit signal in your HDR grading monitor. When switching to previewing in the Apple Vision Pro or for final delivery, re-enable the AVP node (04).

Can the CSTs be placed in the timeline node editor?

No. At the moment, clip tonal mapping relies on the accurate application of the Color Space Transform Adaptation parameter, which must occur on a clip-by-clip basis.

Audio

The Apple Vision Pro and DaVinci Resolve support spatial and ambisonic audio, AVP (Apple Vision Pro) audio, and AIA (Apple Immersive Audio) formats.

Many aspects of audio post production remain unchanged when working with spatial audio. One of the major differences is that rather than monitoring with high-end speakers, audio engineers should instead use headphones or Airpods for true spatial immersion and for a better reproduction of the audience's listening environment.

Whereas in 2D mixes, the majority of dialogue is mixed to the center channels, spatial audio needs to be mixed to the area where the sound is originating. Ducking, equalization and balancing must be used to mix the audio stems in a way that does not overwhelm or confuse the listener.

Immersive spatial elements can be combined with a regular stereo stem, allowing certain sounds to remain constant no matter where the head is moved. This is helpful for keeping non-diegetic sound, such as the music score and voiceovers, clear and easy to interpret.

Environment sounds tend to be interpreted more quietly in spatial audio, so will often need to be mixed more aggressively to be picked up. However, this should not be done at the expense of disorienting the listener.

Reverb contributes greatly to immersion as it is something that listeners experience naturally every day – every sound produced within a given environment bounces off the surfaces that surround the listener, resulting in a 'blooming' reverb quality. Thus, reverb is encouraged in spatial audio design to ensure full immersion.

The Apple Immersive Audio (AIA) format is a container for several formats:

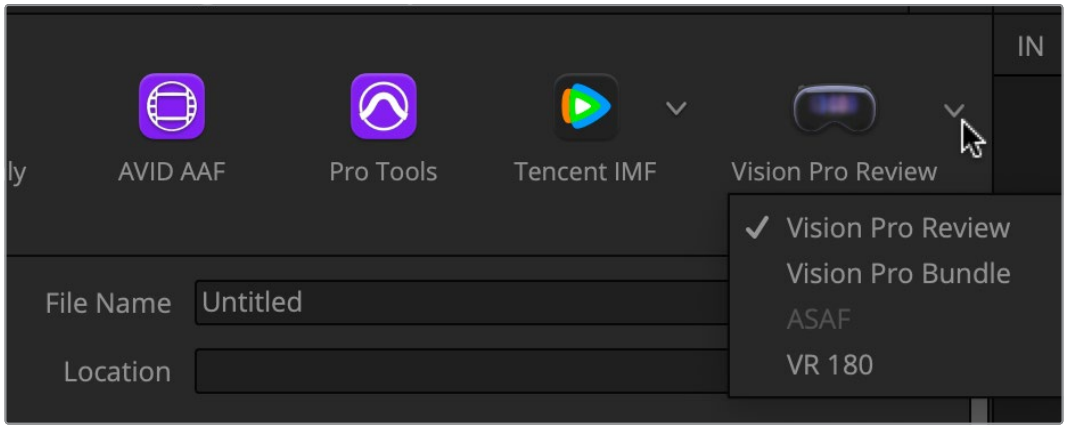
- **Apple Spatial Audio Format (ASAF)** is the raw working audio format.
- **Apple Positional Audio Codec (APAC)** is the encoded .mp4 format playable on APAC-supported devices, like Airpods, or from an AIVU file on the Apple Vision Pro.

Apple Spatial Audio Format (ASAF)

The Apple Spatial Audio Format (ASAF) is an immersive audio format developed primarily for VR and AR applications in the Apple Vision Pro. Exclusive to macOS, ASAF is supported through ProTools (via plugins) and DaVinci Resolve (built-in). ASAF features advanced technical capabilities, including built-in support for Ambisonics sources (called Scenes), and routing via bus assigns. Mixing is virtual, with no physical bus, and users can choose from various monitoring formats, including binaural with headtracking via direct Apple AirPod integration. Panning is handled through DaVinci Resolve's existing tools like the pan dialog, automation, SpaceView, and in-viewer pan tools, with secondary controls in the Inspector. These allow for the precise spatial placement of sources, including automatic tracking against lat-long video. ASAF mixes can be exported in WAV or MP4 formats (with accompanying metadata) via the Immersive Audio > Export Master File option, though they cannot currently be imported or played back within DaVinci Resolve.

Delivery

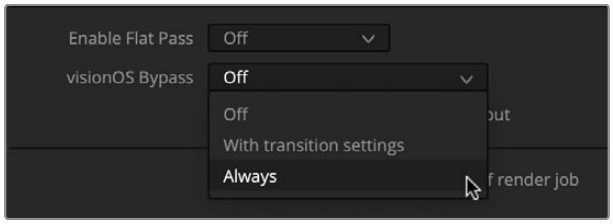
In the deliver page, two render settings allow you to render Apple Vision Pro files for review and final delivery.



- **Vision Pro Review** – creates a small MV-HEVC package for preview purposes. Apple 'share' the resulting file to the Apple Vision Pro via Wi-Fi and watch in the Apple Immersive Video Utility (AIVU) app, which is available in the App store.
- **Vision Pro Bundle** – creates a large ProRes file bundle which can be archived or processed in Apple Compressor to prepare it for playback on Apple TV.
- **ASAF** – renders an ASAF audio mix in WAV or MP4 formats.
- **VR180** – generates a VR180-compliant stereoscopic video file.

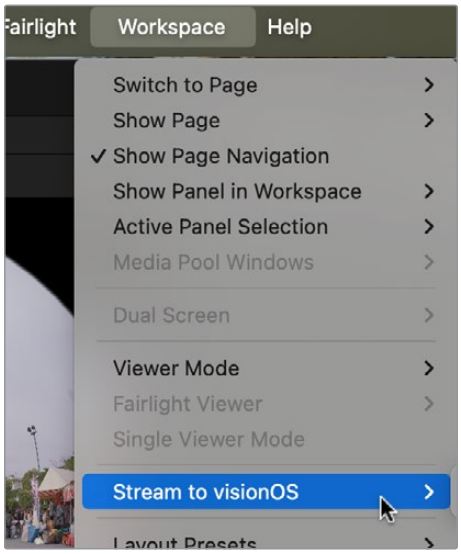
Bypassing Transitions

If a transition is added to an immersive timeline, it will be pre-rendered and turned into a single playback track. The XML generated for immersive content describes the process of switching between these rendered streams (footage, then pre-rendered transition, then back to footage). In the Advanced Settings of the Render Settings panel, you have the option to bypass visionOS transitions (Always), apply them according to their status in the timeline ('With transition settings'), or to include them as they appear in the timeline (Off).



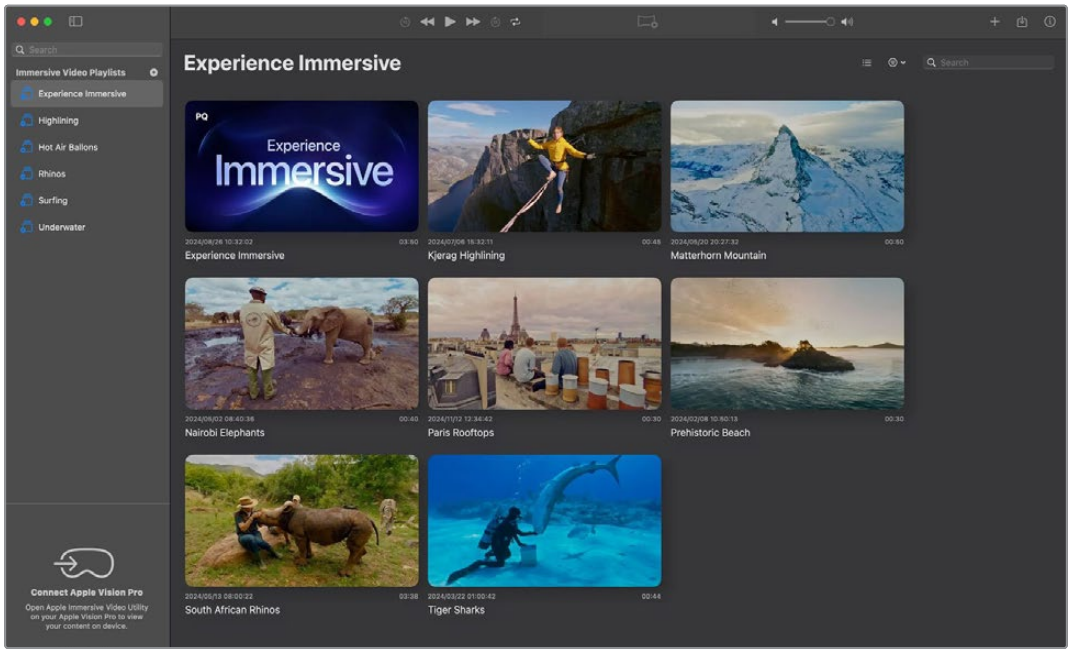
Streaming to the Apple Vision Pro

It is possible to stream to the Apple Vision Pro from the edit, color, and deliver pages in DaVinci Resolve by going to the Workspace menu and choosing ‘Stream to visionOS’ and selecting your AVP device. A fast wireless network (Wi-Fi 5 minimum; Wi-Fi 6 preferred) is recommended for smooth streaming.



About Apple Immersive Video Utility

The Apple Immersive Video Utility is a specialized macOS and visionOS app designed to streamline the creation, management, and playback of Apple Immersive Video content. Tailored for professional filmmakers and immersive media creators, the utility allows users to import, organize, package, and review 8K stereoscopic video files with a 180-degree field of view and Spatial Audio. It supports the creation of AIVU (Apple Immersive Video Universal) files, enabling metadata inspection and modification for post-production workflows. Users can build playlists, sort and search their libraries, and even stream content directly to one or more Apple Vision Pro devices.



Developer Information

Camera Control REST API

If you are a software or hardware developer you can build custom applications or leverage ready to use tools such as REST client or Postman to seamlessly control and interact with your compatible Blackmagic camera using Camera Control REST API. This API enables you to perform a wide range of operations, such as starting or stopping recordings, accessing disk information and much more. Whether you're developing a custom application tailored to your specific needs or utilizing existing tools, this API empowers you to unlock the full potential of your Blackmagic camera with ease. We look forward to seeing what you come up with!

NOTE It's important to mention that controlling Blackmagic cameras via REST API relies on the web manager being enabled on each compatible Blackmagic camera. Enable the web media manager in the Blackmagic Camera Setup 'network access' settings for each camera you are controlling.

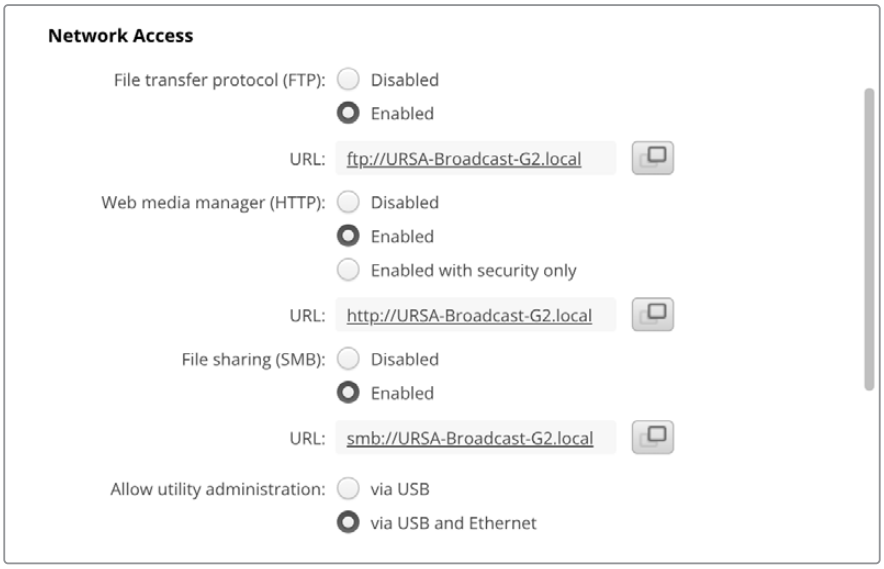
The following Blackmagic cameras are compatible with Camera Control REST API:

- Blackmagic URSA Cine 12K LF
- Blackmagic URSA Cine 17K 65
- Blackmagic URSA Cine Immersive
- Blackmagic PYXIS 6K
- Blackmagic Cinema Camera 6K
- Blackmagic URSA Broadcast G2
- Blackmagic Micro Studio Camera 4K G2
- Blackmagic Studio Camera 4K Plus
- Blackmagic Studio Camera 4K Pro
- Blackmagic Studio Camera 6K Pro
- Blackmagic Studio Camera 4K Plus G2
- Blackmagic Studio Camera 4K Pro G2

Sending API Commands

To send an API command to your camera from a third party application such as Postman, add /control/api/v1/ to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/api/v1/

You can find the Web media manager URL and IP address information in Blackmagic Camera Setup.



The Web media manager URL in Blackmagic Camera Setup

Downloading API's from your Camera

You can download REST API YAML documentation from your camera by adding /control/documentation.html to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/documentation.html

NOTE It's worth noting that changing the camera name in Blackmagic Camera Setup will also change the camera's Web media manager URL.

Livestream Control API

API for controlling Livestreams on Blackmagic Design products.

GET /livestreams/0

Get the livestream's current status.

Response

200 - Livestream's current status.

The response is JSON.

Name	Type	Description
status (required)	string	Possible values are: Idle, Connecting, Streaming, Flushing, Interrupted.
bitrate (required)	integer	Current bitrate (bps).
effectiveVideoFormat (required)	string	Effective video format for the livestream, serialised as a string.
duration	integer	Current stream duration in seconds. Absent if livestream is idle.
cache	integer	Current stream cache usage percentage.

GET /livestreams/0/start

Determine if the livestream is active.

Response

200 - Livestream active status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is active.

PUT /livestreams/0/start

Start the livestream.

Response

204 - Livestream started.

GET /livestreams/0/stop

Determine if the livestream is inactive.

Response

200 - Livestream inactive status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is inactive.

PUT /livestreams/0/stop

Stop the livestream.

Response

204 - Livestream stopped.

GET /livestreams/0/activePlatform

Get the currently selected platform configuration for the livestream.

Response

200 - Livestream active platform configuration.

The response is JSON.

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

PUT /livestreams/0/activePlatform

Set the currently selected platform configuration for the livestream.

Parameters

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

Response

204 - Livestream active platform configuration updated.

400 - Bad Request

GET /livestreams/platforms

Get the list of available platforms.

Response

200 - List of available platforms.

The response is JSON.

Name	Type	Description
	array	List of available platforms names.
[i]	string	Platform name.

GET /livestreams/platforms/{platformName}

Get the service configuration for a platform.

Parameters

Name	Type	Description
{platformName} (required)	string	Name of the platform.

Response

200 - Service configuration for specified platform.

The response is JSON.

Livestream platform service configuration.

Name	Type	Description
platform (required)	string	Corresponding platform name.
key	string	Default stream key.
servers (required)	array	List of server configurations.
servers[i]	object	Server configuration.
servers[i].server (required)	string	Server name.
servers[i].url (required)	string	Livestream destination.
servers[i].srtExtensions	array	Miscellaneous tags used for SRT livestreams.
servers[i].srtExtensions[i]	object	Dictionary object mapping SRT tag strings to values.
servers[i].srtExtensions[i][{key}]	string	SRT tag value.
servers[i].group	string	Logical grouping of the server.
profiles (required)	array	List of profile configurations.
profiles[i]	object	Quality configuration.
profiles[i].profile (required)	string	Quality level name.
profiles[i].configs (required)	array	List of video format configurations.
profiles[i].configs[i]	object	Video format configuration for profiles.
profiles[i].configs[i].resolution (required)	string	Video format serialised as a string.
profiles[i].configs[i].fps (required)	string	Frames per second.
profiles[i].configs[i].bitrate (required)	integer	Pixel bitrate (bps).
profiles[i].configs[i].audioBitrate	integer	Audio bitrate (bps).
profiles[i].configs[i].keyFrameInterval	integer	How often a key frame is sent, in seconds.
profiles[i].configs[i].videoCodecs	array	Supported video encoding algorithm/s.

Name	Type	Description
profiles[i].configs[i].videoCodecs[i]	string	Video encoding algorithm. Possible values are: H264, H265.
profiles[i].lowLatency (required)	boolean	If true, fewer frames will be buffered in the livestream.
defaultProfile	string	Quality level name.
credentials	object	Credientials used for RTMP streams.
credentials.username (required)	string	The username part of the creditials. Only used for RTMP streams.
credentials.password (required)	string	Used for RTMP streams, also used as Passphrase for SRT streams.
customizableUrlEnabled	boolean	True when the server URL is customizable.

400 - Bad Request

GET /livestreams/customPlatforms

Get a list of custom platform files.

Response

200 - List of custom platform files.

The response is JSON.

Name	Type	Description
	array	List of custom platform file names.
[i]	string	Custom platform file name.

DELETE /livestreams/customPlatforms

Remove all custom configuration files.

Response

204 - All custom configuration files removed.

GET /livestreams/customPlatforms/{filename}

Get a custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to get.

Response

200 - Custom platform file.

The response is XML.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

404 - Not Found

PUT /livestreams/customPlatforms/{filename}

Update a custom platform file if it exists, if not, create a new file with the given file name.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to update/create.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

Response

204 - Custom platform file created or updated.

400 - Bad Request

DELETE /livestreams/customPlatforms/{filename}

Remove the given custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to be removed.

Response

204 - Custom platform file removed.

404 - Not Found

Clips Control API

API for listing clips on disk.

GET /clips

Get the list of clips on the active disk.

Response

200 - List of clips on the active disk.

The response is JSON.

List of media clips.

Name	Type	Description
clips (required)	array	
clips[i]	object	Media clip.
clips[i].clipUniqueId (required)	integer	Unique ID used to identify this clip.
clips[i].filePath	string	Path to the file relative to the root of a mount.
clips[i].fileSize	integer	Size of file on disk in bytes.
clips[i].codecFormat	object	
clips[i].codecFormat.codec	string	Currently selected codec.

Name	Type	Description
clips[i].codecFormat.container	string	Multimedia container format.
clips[i].videoFormat	object	Video format configuration.
clips[i].videoFormat.name (required)	string	Video format serialised as a string.
clips[i].videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
clips[i].videoFormat.height	number	Height dimension of video format.
clips[i].videoFormat.width	number	Width dimension of video format.
clips[i].videoFormat.interlaced	boolean	Is the display format interlaced?
clips[i].startTimecode	string	Start timecode of the clip serialised as string.
clips[i].durationTimecode	string	Duration of the clip in timecode format serialised as string.
clips[i].frameCount	integer	Number of frames in clip; duration of the clip in frames.

404 - There is no active disk.

Media Pool Control API

API to manage media pool and handle uploads and project data.

GET /cloud/projects

List all projects within the media pool.

Response

200 - Successfully retrieved the list of all projects.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].libraryID	string	
[i].name	string	
[i].private	boolean	
[i].shared	boolean	
[i].clips	array	List of clips associated with the project.
[i].clips[i]	string	
[i].status	object	
[i].status.numClipsRequested	integer	
[i].status.numClipsComplete	integer	
[i].status.uploadPercent	integer	Percentage of upload completion.
[i].status.numClipsPaused	integer	
[i].status.outOfSpace	boolean	
[i].status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
[i].status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/active

Retrieve data of the actively uploading project.

Response

200 - Successfully retrieved the active project's data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/{projectID}

Retrieve specific project data by project ID.

Parameters

Name	Type	Description
{projectID} (required)	integer	Unique identifier of the project.

Response

200 - Successfully retrieved the project data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	

Name	Type	Description
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

404 - Project not found.

GET /cloud/clips

List all clips within the media pool.

Response

200 - Successfully retrieved the list of all clips.

The response is JSON.

Name	Type	Description
	array	
[i]	string	

GET /cloud/clips/activeUploading

Retrieve data of actively uploading clips.

Response

200 - Successfully retrieved the list of actively uploading clips.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].path	string	
[i].projectID	integer	
[i].status	object	
[i].status.projectID	integer	
[i].status.outOfSpace	boolean	
[i].status.proxyExtension	string	
[i].status.growingFile	boolean	
[i].status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.originalClipTotalSize	integer	
[i].status.proxyClipTotalSize	integer	
[i].status.originalClipCompletedSize	integer	
[i].status.proxyClipCompletedSize	integer	
[i].status.secsRemaining	integer	

GET /cloud/clips/{deviceName}/{path}

Retrieve specific clip data by device and path.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored.
{path} (required)	string	Path to the clip.

Response

200 - Successfully retrieved the clip data.

The response is JSON.

Name	Type	Description
path	string	
projectID	integer	
status	object	
status.projectID	integer	
status.outOfSpace	boolean	
status.proxyExtension	string	
status.growingFile	boolean	
status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.originalClipTotalSize	integer	
status.proxyClipTotalSize	integer	
status.originalClipCompletedSize	integer	
status.proxyClipCompletedSize	integer	
status.secsRemaining	integer	

404 - Clip not found.

Monitoring Control API

API for monitoring and controlling display settings in video equipment.

GET /monitoring/display

Retrieve a list of all display names.

Response

200 - Returns a list of display names.

The response is JSON.

Name	Type	Description
displays	array	List of display names available.
displays[i]	string	

GET /monitoring/{displayName}/cleanFeed

Get the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/cleanFeed

Set the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Clean feed enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/displayLUT

Get the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

400 - Invalid display name.

404 - Display name not found.

PUT /monitoring/{displayName}/displayLUT

Set the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Display LUT enabled/disabled successfully.

400 - Invalid input.

422 - Unprocessable Entity - Unable to process the contained instructions.

GET /monitoring/{displayName}/zebra

Get the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/zebra

Set the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Zebra enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/focusAssist

Get the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/focusAssist

Set the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
mode	string	Mode of focus assist, e.g., ‘Peak’ or ‘ColoredLines’. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/focusAssist

Get the focus assist settings.

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Mode of focus assist, e.g., ‘Peak’ or ‘ColoredLines’. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

404 - Display name not found.

PUT /monitoring/focusAssist

Set the focus assist settings.

Parameters

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/frameGuide

Get the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame guide enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGuide

Set the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame guide state updated successfully.

422 - Unable to update the frame guide state.

GET /monitoring/frameGuideRatio

Get the current frame guide ratio.

Response

200 - Returns the current frame guide ratio.

The response is JSON.

Name	Type	Description
ratio	string	The frame guide ratio.

PUT /monitoring/frameGuideRatio

Set the frame guide ratio.

Parameters

Name	Type	Description
ratio	string	The frame guide ratio.

Response

204 - Frame guide ratio updated successfully.

422 - Unable to update the frame guide ratio.

GET /monitoring/frameGuideRatio/presets

Get the presets for frame guide ratios.

Response

200 - Returns a list of preset frame guide ratios.

The response is JSON.

Name	Type	Description
presets	array	
presets[i]	string	A frame guide ratio.

GET /monitoring/{displayName}/frameGrids

Get the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame grids enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGrids

Set the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame grids state updated successfully.

422 - Unable to update the frame grids state.

GET /monitoring/frameGrids

Get the global frame grids settings.

Response

200 - Returns the current frame grids settings.

The response is JSON.

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

PUT /monitoring/frameGrids

Set the global frame grids settings.

Parameters

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

Response

204 - Frame grids settings updated successfully.

400 - Invalid input, check the number of frame grids or values.

422 - Unable to update the frame grids settings.

GET /monitoring/{displayName}/safeArea

Get the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the safe area enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/safeArea

Set the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Safe area state updated successfully.

422 - Unable to update the safe area state.

GET /monitoring/safeAreaPercent

Get the current safe area percentage.

Response

200 - Returns the current safe area percentage.

The response is JSON.

Name	Type	Description
percent	integer	Safe area coverage percentage.

PUT /monitoring/safeAreaPercent

Set the safe area percentage.

Parameters

Name	Type	Description
percent	integer	Safe area coverage percentage to set.

Response

204 - Safe area percentage updated successfully.

400 - Invalid percentage value.

422 - Unable to update the safe area percentage.

GET /monitoring/{displayName}/falseColor

Get the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the false color enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/falseColor

Set the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - False color state updated successfully.

422 - Unable to update the false color state.

Event Control API

API For working with built-in websocket.

GET /event/list

Get the list of events that can be subscribed to using the websocket API.

Response

200 - Websocket API events list.

The response is JSON.

Name	Type	Description
events	array	List of events that can be subscribed to using the websocket API.
events[i]	string	

System Control API

API for controlling the System Modes on Blackmagic Design products.

GET /system

Get device system information.

Response

200 - System summary.

The response is JSON.

The properties will be populated only with the values that are supported/implemented by the device in use.

Name	Type	Description
codecFormat	object	Codec format configuration.
codecFormat.codec	string	Codec serialised as string.
codecFormat.container	string	Multimedia container format.
videoFormat	object	Video format configuration.
videoFormat.name (required)	string	Video format serialised as a string.
videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
videoFormat.height	number	Height dimension of video format.
videoFormat.width	number	Width dimension of video format.
videoFormat.interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/product

Get device product information.

Response

200 - Device product information.

The response is JSON.

Product information.

Name	Type	Description
deviceName	string	Name of device as displayed in Setup.
productName	string	Device's product name.
softwareVersion	string	Software version running on device.

501 - This functionality is not implemented for the device in use.

GET /system/supportedCodecFormats

Get the list of supported codecs.

Response

200 - List of supported codec formats.

The response is JSON.

Name	Type	Description
codecs	array	
codecs[i]	object	Codec format configuration.
codecs[i].codec	string	Codec serialised as string.
codecs[i].container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

GET /system/codecFormat

Get the currently selected codec.

Response

200 - Current codec format.

The response is JSON.

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

PUT /system/codecFormat

Update the system codec.

Parameters

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

Response

204 - The codec updated successfully.

400 - The specified codec format is unsupported.

501 - This functionality is not implemented for the device in use.

GET /system/videoFormat

Get the currently selected video format.

Response

200 - Current system video format.

The response is JSON.

Video format configuration.

Name	Type	Description
name (required)	string	Video format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height	number	Height dimension of video format.
width	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

PUT /system/videoFormat

Set the system video format.

Parameters

This parameter can be one of the following types:

Name	Type	Description
name (required)	string	Video format serialised as a string.

Name	Type	Description
frameRate (required)	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height (required)	number	Height dimension of video format.
width (required)	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

Response

204 - The video format updated successfully.

400 - Invalid request.

409 - Operation unsupported in the current state.

501 - This functionality is not implemented for the device in use.

GET /system/supportedVideoFormats

Get the list of supported video formats for the current system state.

Response

200 - List of supported video formats.

The response is JSON.

List of supported video formats.

Name	Type	Description
formats	array	List of video formats.
formats[i]	object	Video format configuration.
formats[i].name (required)	string	Video format serialised as a string.
formats[i].frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
formats[i].height	number	Height dimension of video format.
formats[i].width	number	Width dimension of video format.
formats[i].interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/supportedFormats

Get supported formats.

Response

200 - List of supported formats.

The response is JSON.

Name	Type	Description
supportedFormats	array	
supportedFormats[i]	object	
supportedFormats[i].codecs	array	
supportedFormats[i].codecs[i]	string	
supportedFormats[i].frameRates	array	
supportedFormats[i].frameRates[i]	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
supportedFormats[i]. maxOffSpeedFrameRate	number	
supportedFormats[i]. minOffSpeedFrameRate	number	
supportedFormats[i]. recordResolution	object	
supportedFormats[i]. recordResolution.height	number	Height of the resolution.
supportedFormats[i]. recordResolution.width	number	Width of the resolution.
supportedFormats[i]. sensorResolution	object	
supportedFormats[i]. sensorResolution.height	number	Height of the resolution.
supportedFormats[i]. sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

GET /system/format

Get current format.

Response

200 - Current format.

The response is JSON.

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

PUT /system/format

Set the format.

Parameters

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

Response

204 - System format updated.

501 - This functionality is not implemented for the device in use.

Transport Control API

API for controlling Transport on Blackmagic Design products.

GET /transports/0

Get device's basic transport status.

Response

200 - Transport status.

The response is JSON.

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

PUT /transports/0

Set device's basic transport status.

Parameters

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, Output.

Response

204 - Transport mode was set.

400 - Failed to set transport mode.

GET /transports/0/stop

Determine if transport is stopped.

Response

200 - Transport stop response.

The response is JSON.

Name	Type	Description
	boolean	True when transport mode is InputPreview or when in Output mode and speed is 0.

PUT /transports/0/stop

Stop transport. Deprecated, use POST /transports/0/stop instead.

Response

204 - Transport stopped.

POST /transports/0/stop

Stop transport.

Response

204 - Transport stopped.

GET /transports/0/play

Determine if transport is playing.

Response

200 - Transport play response.

The response is JSON.

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

PUT /transports/0/play

Start playing on transport. Deprecated, use POST /transports/0/play instead.

Response

204 - Transport playing.

400 - Failed to set transport to play.

POST /transports/0/play

Start playing on transport.

Response

204 - Transport playing.

400 - Failed to set transport to play.

GET /transports/0/playback

Get playback state.

Response

200 - Transport playback state.

The response is JSON.

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

PUT /transports/0/playback

Set playback state.

Parameters

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

Response

204 - Updated transport playback state.

400 - Failed to set transport playback state.

GET /transports/0/record

Get record state.

Response

200 - Recording state.

The response is JSON.

Name	Type	Description
recording	boolean	If true, transport is in InputRecord mode.

PUT /transports/0/record

Set record state. Deprecated, use POST /transports/0/record instead.

Parameters

Name	Type	Description
recording (required)	boolean	If true, starts a recording, otherwise stops.
clipName	string	Optional, sets the requested clip name to record to, when “recording” attribute is set to true.

Response

204 - Recording state updated.

400 - Failed to update recording state.

POST /transports/0/record

Start recording.

Parameters

Name	Type	Description
clipName	string	Optional, provides a specific name of clip to record to.

Response

204 - Recording started.

400 - Failed to start recording.

GET /transports/0/clipIndex

Get the clip index of the currently playing clip on the timeline.

Response

200 - Clip index response.

The response is JSON.

Name	Type	Description
clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

GET /transports/0/timecode

Get device timecode.

Response

200 - Timecode response.

The response is JSON.

Name	Type	Description
display	string	The display timecode serialised as a string.
timeline	string	The timeline timecode serialised as a string.

GET /transports/0/timecode/source

Get timecode source selected on device.

Response

200 - Timecode source response.

The response is JSON.

Name	Type	Description
timecode	string	Possible values are: Timeline, Clip.

Timeline Control API

API for controlling playback timeline.

GET /timelines/0

Get the playback timeline.

Response

200 - Playback timeline.

The response is JSON.

Name	Type	Description
clips	array	
clips[i]	object	Timeline clip.
clips[i].clipUniqueId (required)	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).
clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

404 - No timeline / disk available.

DELETE /timelines/0

Clear the current playback timeline. Deprecated, prefer to use POST /timelines/0/clear

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

POST /timelines/0

Add a clip to the timeline.

Parameters

This parameter can be one of the following types:

Add multiple media clips to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips[i].clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips[i].clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips[i].frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add multiple media clips to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add a single clip to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips.clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips.clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips.frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add a single clip to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the timeline as specified.

501 - The operation is not supported on this device.

POST /timelines/0/add

Add a clip to the end of the timeline. Deprecated, use POST /timelines/0 to add clips within the timeline.

Parameters

This parameter can be one of the following types:

Add one clip to the end of the timeline.

Name	Type	Description
clips	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add many clips to the end of the timeline.

Name	Type	Description
clips	array	List of clipUniqueIds of clips to add to end of timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the end of the timeline.

501 - The operation is not supported on this device.

POST /timelines/0/clear

Clear the playback timeline.

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

DELETE /timelines/0/clips/{timelineClipIndex}

Remove the specified clip from the timeline.

Parameters

Name	Type	Description
{timelineClipIndex} (required)	integer	The (0-based) timeline clip index of the clip to remove from the timeline.

Response

204 - The specified clip was removed from the timeline.

501 - The operation is not supported on this device.

Media Control API

API for controlling media devices in Blackmagic Design products.

GET /media/workingset

Get the list of media devices currently in the working set.

Response

200 - The list of media devices in the working set.

The response is JSON.

Name	Type	Description
size	integer	The fixed size of this device's working set.
workingset	array	The device's working set.
workingset[i]	object | null	Device within the working set. null if no device is present within the given working set slot.
workingset[i].volume	string	Volume name.
workingset[i].deviceName	string	Internal device name of this media device.
workingset[i].remainingRecordTime	integer	Remaining record time using current codec and video format in seconds.
workingset[i].totalSpace	integer	Total space on media device in bytes.
workingset[i].remainingSpace	integer	Remaining space on media device in bytes.
workingset[i].clipCount	integer	Number of clips currently on the device.

GET /media/active

Get the currently active media device.

Response

200 - The current active media device.

The response is JSON.

The active media device, or null if there is no active media.

Name	Type	Description
workingsetIndex	integer	Working set index of the active media device.
deviceName	string	Device name of media device.

204 - No media is currently active.

PUT /media/active

Set the currently active media device.

Parameters

Name	Type	Description
workingsetIndex	integer	Working set index of the media to make active.

Response

204 - The active media device was set successfully.

400 - Setting the currently active media device is not possible in the current state.

GET /media/devices/doformatSupportedFilesystems

Get the list of filesystems available to format a media device.

Response

200 - The list of filesystems permitted for formatting.

The response is JSON.

Name	Type	Description
	array	List of filesystems permitted for formatting media.
[i]	string	Filesystem serialised as string.

GET /media/devices/{deviceName}

Get information about a requested device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Information about the requested device.

The response is JSON.

Media device state.

Name	Type	Description
state	string	The current state of the media device. Possible values are: None, Scanning, Mounted, Uninitialised, Formatting, RaidComponent.

400 - Invalid device name.

404 - Device not found.

GET /media/devices/{deviceName}/doformat

Get a format key, used to format the device with a PUT request.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Format prepared.

The response is JSON.

Name	Type	Description
deviceName	string	Device name of media device to format.
key	string	The key required to format this device, provide to PUT /media/devices/{deviceName}/doformat to perform format of media device.

400 - Cannot format the device.

404 - Device not found.

PUT /media/devices/{deviceName}/doformat

Perform a format of the specified media device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Name	Type	Description
key	string	The key used to format this device, retrieved from prepare format media request GET /media/devices/{deviceName}/doformat. Format key provided cannot be reused after successful format.
filesystem	string	Filesystem to format to. Supported filesystems can be retrieved with GET /media/devices/doFormatSupportedFilesystems.
volume	string	Volume name to set for the disk after format.

Response

204 - Format successful.

400 - Cannot format the device, invalid filesystem or key.

404 - Device not found.

Slate Control API

API to manage digital slate data.

GET /slates/nextClip

Retrieve the digital slate for the next clip.

Response

200 - Returns the slate data for the next clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	

Name	Type	Description
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Slate data is not available.

PUT /slates/nextClip

Update the slate data for the next clip.

Parameters

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

POST /slates/nextClip/resetProjectData

Reset the project data for the next clip's slate.

Response

200 - Project data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetProjectData

Reset the project data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Project data reset successfully.

POST /slates/nextClip/resetLensData

Reset the lens data for the next clip's slate.

Response

200 - Lens data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetLensData

Reset the lens data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Lens data reset successfully.

GET /slates/clips/{deviceName}/{path}

Retrieve slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Returns the slate data for the specified clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

404 - Clip not found.

PUT /slates/clips/{deviceName}/{path}

Update the slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.

Name	Type	Description
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

Preset Control API

API For controlling the presets on Blackmagic Design products

GET /presets

Get the list of the presets on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presets	array	List of the presets on the camera (.cset files)
presets[i]	string	

POST /presets

Send a preset file to the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presetAdded	string	Name of the preset uploaded (without .cset extension)

400 - Bad request - missing Content-Disposition header or filename

GET /presets/active

Get the currently active preset on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

PUT /presets/active

Set the active preset on the camera

Parameters

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

Response

204 - No Content

404 - Preset file not found

PARAMETERS /presets/{presetName}

GET /presets/{presetName}

Download the preset file

Response

200 - OK

The response is a binary file.

404 - File does not exist

PUT /presets/{presetName}

Save current camera state as a preset

Response

204 - No Content

DELETE /presets/{presetName}

Delete a preset from the camera

Response

204 - No Content

404 - Preset file not found

Audio Control API

API For controlling audio on Blackmagic Design Cameras

GET /audio/channels

Get the total number of audio channels available

Response

200 - Returns the total number of channels

The response is JSON.

Name	Type	Description
channels	integer	Total number of audio channels available

GET /audio/supportedInputs

Get the list of supported audio inputs

Response

200 - List of all supported audio inputs

The response is JSON.

Name	Type	Description
	array	
[i]	string	A supported audio input

GET /audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently selected input

The response is JSON.

Name	Type	Description
input	string	Audio input source and type

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/input

Set the audio input for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
input	string	Audio input source and type

Response

204 - No Content

400 - Invalid audio input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/input/description

Get the description of the current input of the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Description of the current input of the selected channel

The response is JSON.

Name	Type	Description
description	object	
description.gainRange	object	
description.gainRange.Min	number	The minimum gain value in dB
description.gainRange.Max	number	The maximum gain value in dB
description.capabilities	object	
description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter
description.capabilities.Padding	object	
description.capabilities.Padding. available	boolean	Input supports setting of padding
description.capabilities.Padding. forced	boolean	Padding is forced to be set for the input
description.capabilities.Padding. value	number	Value of the padding in dB

404 - Channel does not exist

GET /audio/channel/{channelIndex}/supportedInputs

Get the list of supported inputs and their availability to switch to for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its supported inputs are being queried. (Channels index from 0)

Response

200 - The list of supported inputs

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

404 - Channel does not exist

GET /audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set level for the selected channel

The response is JSON.

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/level

Set the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

Response

204 - No Content

400 - Invalid input or value out of range

404 - Channel does not exist

GET /audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set phantom power for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Phantom power enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/phantomPower

Set the audio phantom power for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Phantom power enabled state

Response

204 - No Content

400 - Phantom power is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set padding for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Padding enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/padding

Set the audio input padding for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Padding enabled state

Response

204 - No Content

400 - Padding is not supported or is forced for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set low cut filter for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Low cut filter enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/lowCutFilter

Set the audio input low cut filter for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Low cut filter enabled state

Response

204 - No Content

400 - Low cut filter is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set availability for the selected channel

The response is JSON.

Name	Type	Description
available	boolean	Whether the input is currently available

404 - Channel does not exist

Lens Control API

API For controlling the lens on Blackmagic Design products

GET /lens/iris

Get lens' aperture

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number

PUT /lens/iris

Set lens' aperture

Parameters

Name	Type	Description
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number
adjustmentStep	integer	Signed value for relative aperture adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens iris is not controllable or is controlled by auto exposure

GET /lens/zoom

Get lens' zoom

Response

200 - OK

The response is JSON.

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value

PUT /lens/zoom

Set lens' zoom

Parameters

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value
adjustmentFocalLength	integer	Signed value for relative focal length adjustment
adjustmentNormalised	number	Signed normalized value for relative zoom adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens zoom is not controllable

GET /lens/focus

Get lens' focus

Response

200 - OK

The response is JSON.

Name	Type	Description
normalised	number	Normalised value

PUT /lens/focus

Set lens' focus

Parameters

Name	Type	Description
normalised	number	Normalised value
focusDistance	integer	Focus distance value

Response

204 - No Content

400 - Bad Request if out of range value is provided

PUT /lens/focus/doAutoFocus

Perform auto focus

Parameters

Name	Type	Description
position	object	
position.x	number	Normalized x coordinate for autofocus ROI
position.y	number	Normalized y coordinate for autofocus ROI

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens focus is not controllable

GET /lens/opticalImageStabilization

Get optical image stabilization status

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Whether optical image stabilization is enabled

501 - Not Implemented if optical image stabilization is not supported on this product

PUT /lens/opticalImageStabilization

Enable or disable optical image stabilization

Parameters

Name	Type	Description
enabled	boolean	Enable or disable optical image stabilization

Response

204 - No Content

501 - Not Implemented if optical image stabilization is not supported on this product

GET /lens/iris/description

Get detailed description of lens' iris capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the iris can be controlled
apertureStop	object	
apertureStop.min	number	Minimum aperture stop
apertureStop.max	number	Maximum aperture stop

GET /lens/zoom/description

Get detailed description of lens' zoom capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the zoom can be controlled
focalLength	object	
focalLength.adjustable	boolean	If focal length is adjustable
focalLength.min	integer	Minimum focal length
focalLength.max	integer	Maximum focal length

GET /lens/focus/description

Get detailed description of lens' focus capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the focus can be controlled
focusDistance	object	
focusDistance.adjustable	boolean	If focus distance is adjustable
focusDistance.min	number	Minimum focus distance
focusDistance.max	number	Maximum focus distance

Video Control API

API For controlling the video on Blackmagic Design products

GET /video/iso

Get current ISO

Response

200 - OK

The response is JSON.

Name	Type	Description
iso	integer	Current ISO value

PUT /video/iso

Set current ISO

Parameters

Name	Type	Description
iso	integer	ISO value to set

Response

204 - No Content

403 - ISO cannot be changed in the current state

GET /video/supportedISOs

Get the list of supported ISO settings

Response

200 - List of supported ISO values

The response is JSON.

Name	Type	Description
supportedISOs	array	Array of supported ISO values
supportedISOs[i]	integer	

GET /video/gain

Get current gain value in decibels

Response

200 - OK

The response is JSON.

Name	Type	Description
gain	integer	Current gain value in decibels

PUT /video/gain

Set current gain value

Parameters

Name	Type	Description
gain	integer	Gain value in decibels to set

Response

204 - No Content

403 - Gain cannot be changed in the current state

GET /video/supportedGains

Get the list of supported gain settings in decibels

Response

200 - List of supported gain values in decibels

The response is JSON.

Name	Type	Description
supportedGains	array	Array of supported gain values in decibels
supportedGains[i]	integer	

GET /video/whiteBalance

Get current white balance

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	integer	Current white balance

PUT /video/whiteBalance

Set current white balance

Parameters

Name	Type	Description
whiteBalance	integer	White balance to set

Response

204 - No Content

400 - Invalid white balance temperature

GET /video/whiteBalance/description

Get white balance range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	object	
whiteBalance.min	integer	Minimum color temperature
whiteBalance.max	integer	Maximum color temperature

PUT /video/whiteBalance/doAuto

Set current white balance automatically

Response

204 - No Content

GET /video/whiteBalanceTint

Get white balance tint

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	integer	Current white balance tint

PUT /video/whiteBalanceTint

Set white balance tint

Parameters

Name	Type	Description
whiteBalanceTint	integer	White balance tint to set

Response

204 - No Content

400 - Invalid white balance tint

GET /video/whiteBalanceTint/description

Get white balance tint range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	object	
whiteBalanceTint.min	integer	Minimum white balance tint
whiteBalanceTint.max	integer	Maximum white balance tint

GET /video/ndFilter

Get ND filter stop

Response

200 - OK

The response is JSON.

Name	Type	Description
stop	number	Current filter power (fStop)

501 - Not implemented for this device

PUT /video/ndFilter

Set ND filter stop

Parameters

Name	Type	Description
stop	number	Filter power (fStop) to set

Response

204 - No Content

400 - Invalid ND filter stop

501 - Not implemented for this device

GET /video/supportedNDFilters

Get the list of available ND filter stops

Response

200 - List of available ND filter stops

The response is JSON.

Name	Type	Description
supportedStops	array	Array of available ND filter stops
supportedStops[i]	number	

501 - Not implemented for this device

GET /video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

Response

200 - List of supported display modes

The response is JSON.

Name	Type	Description
supportedDisplayModes	array	Array of supported display modes
supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

GET /video/ndFilter/displayMode

Get ND filter display mode on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

PUT /video/ndFilter/displayMode

Set ND filter display mode on the camera

Parameters

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

Response

204 - No Content

400 - Invalid display mode for ND filter

501 - Not implemented for this device

GET /video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

Response

200 - Indicates if ND filter is selectable

The response is JSON.

Name	Type	Description
selectable	boolean	True if ND filter adjustments are selectable via a slider

501 - Not implemented for this device

GET /video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

PUT /video/shutter

Set current shutter

Parameters

Name	Type	Description
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

Response

204 - No Content

GET /video/shutter/measurement

Get the current shutter measurement mode

Response

200 - OK

The response is JSON.

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

PUT /video/shutter/measurement

Set the shutter measurement mode

Parameters

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

Response

204 - No Content

400 - Invalid shutter measurement

GET /video/supportedShutters

Get supported shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of supported shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of supported shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of flicker-free shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of flicker-free shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/autoExposure

Get current auto exposure mode

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

PUT /video/autoExposure

Set auto exposure

Parameters

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

Response

204 - No Content

400 - Failed to set auto exposure mode

GET /video/detailSharpening

Get the current state of detail sharpening

Response

200 - Current detail sharpening state

The response is JSON.

Name	Type	Description
enabled	boolean	Whether detail sharpening is enabled

501 - Not implemented for this device

PUT /video/detailSharpening

Enable or disable detail sharpening

Parameters

Name	Type	Description
enabled	boolean	Enable or disable detail sharpening

Response

204 - Detail sharpening state updated

501 - Not implemented for this device

GET /video/detailSharpeningLevel

Get the current detail sharpening level

Response

200 - Current detail sharpening level

The response is JSON.

Name	Type	Description
level	string	Current detail sharpening level Possible values are: Low, Medium, High.

501 - Not implemented for this device

PUT /video/detailSharpeningLevel

Set the detail sharpening level

Parameters

Name	Type	Description
level	string	Desired level of detail sharpening Possible values are: Low, Medium, High.

Response

204 - Detail sharpening level updated

400 - Invalid detail sharpening level

501 - Not implemented for this device

Camera Control API

API For controlling the Camera specific features on Blackmagic Design products

GET /camera/colorBars

Get the status of color bars display

Response

200 - Returns the current status of color bars

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the color bars are currently enabled

PUT /camera/colorBars

Set the status of color bars display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the color bars

Response

204 - Color bars status updated successfully

GET /camera/programFeedDisplay

Get the status of program feed display

Response

200 - Returns the current status of program feed display

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the program feed display is currently enabled

PUT /camera/programFeedDisplay

Set the status of program feed display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the program feed display

Response

204 - Program feed display status updated successfully

GET /camera/tallyStatus

Get the tally status of the camera

Response

200 - Returns the current tally status of the camera

The response is JSON.

Name	Type	Description
status	string	Current tally status of the camera Possible values are: None, Preview, Program.

GET /camera/power

Get the power status of the camera

Response

200 - Returns the current power status

The response is JSON.

Name	Type	Description
source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
batteries	array	List of batteries currently connected to the camera
batteries[i]	object	
batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
batteries[i].chargeRemainingPercent	integer	Remaining battery charge percentage
batteries[i].statusFlags	array	List of battery status flags
batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

GET /camera/power/displayMode

Get the power display mode of the camera

Response

200 - Returns the current power display mode

The response is JSON.

Name	Type	Description
mode	string	Current power display mode Possible values are: Percentage, Voltage.

PUT /camera/power/displayMode

Set the power display mode of the camera

Parameters

Name	Type	Description
mode	string	Power display mode to set Possible values are: Percentage, Voltage.

Response

204 - Power display mode updated successfully

400 - Invalid power display mode

GET /camera/timingReferenceLock

Get the timing reference lock status

Response

200 - Returns the timing reference lock status

The response is JSON.

Name	Type	Description
locked	boolean	Indicates if timing reference is locked

Immersive Control API

API for controlling immersive camera settings on Blackmagic Design cameras

GET /immersive/display/{displayName}/eye

Get the current eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to query (from /monitoring/display endpoint)

Response

200 - OK

The response is JSON.

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

400 - Invalid display name format

404 - Display not found

422 - Failed to get eye view

PUT /immersive/display/{displayName}/eye

Set the eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to control (from /monitoring/display endpoint)

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

Response

204 - No Content

400 - Invalid input or display name format

404 - Display not found

422 - Failed to set eye view

Color Correction Control API

API For controlling the color correction on Blackmagic Design products based on DaVinci Resolve Color Corrector

GET /colorCorrection/lift

Get color correction lift

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

PUT /colorCorrection/lift

Set color correction lift

Parameters

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gamma

Get color correction gamma

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

PUT /colorCorrection/gamma

Set color correction gamma

Parameters

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gain

Get color correction gain

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

PUT /colorCorrection/gain

Set color correction gain

Parameters

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/offset

Get color correction offset

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

PUT /colorCorrection/offset

Set color correction offset

Parameters

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/contrast

Get color correction contrast

Response

200 - OK

The response is JSON.

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/contrast

Set color correction contrast

Parameters

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/color

Get color correction color properties

Response

200 - OK

The response is JSON.

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/color

Set color correction color properties

Parameters

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/lumaContribution

Get color correction luma contribution

Response

200 - OK

The response is JSON.

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

PUT /colorCorrection/lumaContribution

Set color correction luma contribution

Parameters

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

Response

204 - No Content

Notification websocket - 1.0.0

Service that notifies subscribers of device state changes.

messages

Subscribe (The messages from the server/device)

Websocket Opened Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: websocketOpened.
.type	string	Possible values are: event.

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*

Name	Type	Description
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Event Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: propertyValueChanged.

Name	Type	Description
.data.property	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.value	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.type	string	Possible values are: event.

Publish (The messages that user can send to the server/device)

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	

Name	Type	Description
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	

Name	Type	Description
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Device Properties

/media/workingset

The value JSON returned via the eventResponse when the `/media/workingset` property changes on the device:

Name	Type	Description
.size	integer	The fixed size of this device's working set.
.workingset	array	Array of devices within the working set. null if no device is present within the given working set slot.
.workingset[i]		

/media/active

The value JSON returned via the eventResponse when the `/media/active` property changes on the device:

Name	Type	Description
.workingsetIndex	integer	Working set index of the active media device.
.deviceName	string	Internal device name of this media device.

/system

The value JSON returned via the eventResponse when the `/system` property changes on the device:

Name	Type	Description
.codecFormat	object	Codec format configuration.
.codecFormat.codec	string	Codec format serialised as a string.
.codecFormat.container	string	Multimedia container format.
.videoFormat	object	Currently selected video format.
.videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.videoFormat.height	number	Height dimension of video format.
.videoFormat.width	number	Width dimension of video format.
.videoFormat.interlaced	boolean	Is the display format interlaced?.
.videoFormat.name	string	Video format serialised as a string.

/system/codecFormat

Codec format configuration.

The value JSON returned via the eventResponse when the `/system/codecFormat` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.container	string	Multimedia container format.

/system/videoFormat

Currently selected video format.

The value JSON returned via the eventResponse when the `/system/videoFormat` property changes on the device:

Name	Type	Description
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.height	number	Height dimension of video format.
.width	number	Width dimension of video format.
.interlaced	boolean	Is the display format interlaced?.
.name	string	Video format serialised as a string.

/system/format

The value JSON returned via the eventResponse when the `/system/format` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.maxOffSpeedFrameRate	number	
.minOffSpeedFrameRate	number	
.offSpeedEnabled	boolean	
.offSpeedFrameRate	number	
.recordResolution	object	
.recordResolution.height	number	Height of the resolution.
.recordResolution.width	number	Width of the resolution.
.sensorResolution	object	
.sensorResolution.height	number	Height of the resolution.
.sensorResolution.width	number	Width of the resolution.

/system/supportedFormats

The value JSON returned via the eventResponse when the `/system/supportedFormats` property changes on the device:

Name	Type	Description
.supportedFormats	array	
.supportedFormats[i]	object	
.supportedFormats[i].codecs	array	
.supportedFormats[i].codecs[i]	string	
.supportedFormats[i].frameRates	array	
.supportedFormats[i].frameRates[i]	string	Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.supportedFormats[i].maxOffSpeedFrameRate	number	
.supportedFormats[i].minOffSpeedFrameRate	number	
.supportedFormats[i].recordResolution	object	
.supportedFormats[i].recordResolution.height	number	Height of the resolution.
.supportedFormats[i].recordResolution.width	number	Width of the resolution.
.supportedFormats[i].sensorResolution	object	
.supportedFormats[i].sensorResolution.height	number	Height of the resolution.
.supportedFormats[i].sensorResolution.width	number	Width of the resolution.

/timelines/0

The value JSON returned via the eventResponse when the `/timelines/0` property changes on the device:

Name	Type	Description
.clips	array	
.clips[i]	object	Timeline clip.
.clips[i].clipUniqueId	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
.clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
.clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
.clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
.clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).

Name	Type	Description
.clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
.clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

/transports/0

The value JSON returned via the eventResponse when the `/transports/0` property changes on the device:

Name	Type	Description
.mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

/transports/0/stop

true when transport mode is InputPreview or when in Output mode and speed is 0.

The value JSON returned via the eventResponse when the `/transports/0/stop` property changes on the device:

Name	Type	Description
	boolean	true when transport mode is InputPreview or when in Output mode and speed is 0.

/transports/0/play

True when transport is in Output mode and speed is non-zero.

The value JSON returned via the eventResponse when the /transports/0/play property changes on the device:

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

/transports/0/playback

The value JSON returned via the eventResponse when the `/transports/0/playback` property changes on the device:

Name	Type	Description
.type	string	Possible values are: Play, Jog, Shuttle, Var.
.loop	boolean	When true playback loops from the end of the timeline to the beginning of the timeline.
.singleClip	boolean	When true playback loops from the end of the current clip to the beginning of the current clip.
.speed	number	Playback speed, 1.0 for normal forward playback
.position	integer	Playback position on the timeline in units of video frames.

/transports/0/record

The value JSON returned via the eventResponse when the `/transports/0/record` property changes on the device:

Name	Type	Description
.recording	boolean	Is transport in Input Record mode.

/transports/0/timecode

The value JSON returned via the eventResponse when the `/transports/0/timecode` property changes on the device:

Name	Type	Description
.display	string	The display timecode serialised as a string.
.timeline	string	The timeline timecode serialised as a string.

/transports/0/timecode/source

The value JSON returned via the eventResponse when the `/transports/0/timecode/source` property changes on the device:

Name	Type	Description
.timecode	string	Possible values are: Timeline, Clip.

/transports/0/clipIndex

The value JSON returned via the eventResponse when the `/transports/0/clipIndex` property changes on the device:

Name	Type	Description
.clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

/slates/nextClip

The value JSON returned via the eventResponse when the `/slates/nextClip` property changes on the device:

Name	Type	Description
.clip	object	
.clip.clipName	string	
.clip.reel	integer	
.clip.scene	string	
.clip.sceneLocation	string	Possible values are: Interior, Exterior.
.clip.sceneTime	string	Possible values are: Day, Night.
.clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
.clip.take	integer	
.clip.takeType	string	Possible values are: None, PU, VFX, SER.
.clip.goodTake	boolean	
.lens	object	
.lens.lensType	string	
.lens.iris	string	

Name	Type	Description
.lens.focalLength	string	
.lens.distance	string	
.lens.filter	string	
.project	object	
.project.projectName	string	
.project.director	string	
.project.camera	string	
.project.cameraOperator	string	

/monitoring/{displayName}/cleanFeed

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/cleanFeed` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/displayLUT

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/displayLUT` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/zebra

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/zebra` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/focusAssist` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGuide

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGuide` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGrids` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/safeArea

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/safeArea` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/falseColor

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/falseColor` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/focusAssist` property changes on the device:

Name	Type	Description
.mode	string	Possible values are: Peak, ColoredLines.
.color	string	Possible values are: Red, Green, Blue, White, Black.
.intensity	integer	

/monitoring/frameGuideRatio

The value JSON returned via the eventResponse when the `/monitoring/frameGuideRatio` property changes on the device:

Name	Type	Description
.ratio	string	

/monitoring/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/frameGrids` property changes on the device:

Name	Type	Description
.frameGrids	array	
.frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

/monitoring/safeAreaPercent

The value JSON returned via the eventResponse when the `/monitoring/safeAreaPercent` property changes on the device:

Name	Type	Description
.percent	integer	Safe area coverage percentage.

/audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input` property changes on the device:

Name	Type	Description
.input	string	Audio input source and type

/audio/channel/{channelIndex}/supportedInputs

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/supportedInputs` property changes on the device:

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

/audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/level` property changes on the device:

Name	Type	Description
.gain	number	Gain value in dB
.normalised	number	Normalised level value between 0.0 and 1.0

/audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/phantomPower` property changes on the device:

Name	Type	Description
.enabled	boolean	Phantom power enabled state

/audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/padding` property changes on the device:

Name	Type	Description
.enabled	boolean	Padding enabled state

/audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/lowCutFilter` property changes on the device:

Name	Type	Description
.enabled	boolean	Low cut filter enabled state

/audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/available` property changes on the device:

Name	Type	Description
.available	boolean	Whether the input is currently available

/audio/channel/{channelIndex}/input/description

Description of the current input of the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input/description` property changes on the device:

Name	Type	Description
.description	object	
.description.gainRange	object	
.description.gainRange.Min	number	The minimum gain value in dB
.description.gainRange.Max	number	The maximum gain value in dB
.description.capabilities	object	
.description.capabilities.PhantomPower	boolean	Input supports setting of phantom power
.description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter

Name	Type	Description
.description.capabilities.Padding	object	
.description.capabilities.Padding.available	boolean	Input supports setting of padding
.description.capabilities.Padding.forced	boolean	Padding is forced to be set for the input
.description.capabilities.Padding.value	number	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.

/colorCorrection/lift

Get color correction lift

The value JSON returned via the eventResponse when the `/colorCorrection/lift` property changes on the device:

Name	Type	Description
.red	number	Red lift component
.green	number	Green lift component
.blue	number	Blue lift component
.luma	number	Luma lift component

/colorCorrection/gamma

Get color correction gamma

The value JSON returned via the eventResponse when the `/colorCorrection/gamma` property changes on the device:

Name	Type	Description
.red	number	Red gamma component
.green	number	Green gamma component
.blue	number	Blue gamma component
.luma	number	Luma gamma component

/colorCorrection/gain

Get color correction gain

The value JSON returned via the eventResponse when the `/colorCorrection/gain` property changes on the device:

Name	Type	Description
.red	number	Red gain component
.green	number	Green gain component
.blue	number	Blue gain component
.luma	number	Luma gain component

/colorCorrection/offset

Get color correction offset

The value JSON returned via the eventResponse when the `/colorCorrection/offset` property changes on the device:

Name	Type	Description
.red	number	Red offset component
.green	number	Green offset component
.blue	number	Blue offset component
.luma	number	Luma offset component

/colorCorrection/contrast

Get color correction contrast

The value JSON returned via the eventResponse when the `/colorCorrection/contrast` property changes on the device:

Name	Type	Description
.pivot	number	Contrast pivot point
.adjust	number	Contrast adjustment

/colorCorrection/color

Get color correction color properties

The value JSON returned via the eventResponse when the `/colorCorrection/color` property changes on the device:

Name	Type	Description
.hue	number	Color hue adjustment
.saturation	number	Color saturation adjustment

/colorCorrection/lumaContribution

Get color correction luma contribution

The value JSON returned via the eventResponse when the `/colorCorrection/lumaContribution` property changes on the device:

Name	Type	Description
.lumaContribution	number	Luma contribution value

/lens/iris

Get lens' aperture

The value JSON returned via the eventResponse when the `/lens/iris` property changes on the device:

Name	Type	Description
.continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
.apertureStop	number	Aperture stop value
.normalised	number	Normalised value
.apertureNumber	integer	Aperture number

/lens/iris/description

Get detailed description of lens' iris capabilities

The value JSON returned via the eventResponse when the `/lens/iris/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the iris can be controlled
.apertureStop	object	
.apertureStop.min	number	Minimum aperture stop
.apertureStop.max	number	Maximum aperture stop

/lens/focus

Get lens' focus

The value JSON returned via the eventResponse when the `/lens/focus` property changes on the device:

Name	Type	Description
.normalised	number	Normalised value

/lens/focus/description

Get detailed description of lens' focus capabilities

The value JSON returned via the eventResponse when the `/lens/focus/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the focus can be controlled
.focusDistance	object	
.focusDistance.adjustable	boolean	If focus distance is adjustable
.focusDistance.min	number	Minimum focus distance
.focusDistance.max	number	Maximum focus distance

/lens/zoom

Get lens' zoom

The value JSON returned via the eventResponse when the `/lens/zoom` property changes on the device:

Name	Type	Description
.focalLength	integer	Focal length in mm
.normalised	number	Normalised value

/lens/zoom/description

Get detailed description of lens' zoom capabilities

The value JSON returned via the eventResponse when the `/lens/zoom/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the zoom can be controlled
.focalLength	object	
.focalLength.adjustable	boolean	If focal length is adjustable
.focalLength.min	integer	Minimum focal length
.focalLength.max	integer	Maximum focal length

/presets

Get the list of the presets on the camera

The value JSON returned via the eventResponse when the `/presets` property changes on the device:

Name	Type	Description
.presets	array	List of the presets on the camera (.cset files)
.presets[i]	string	

/presets/active

Get the currently active preset on the camera

The value JSON returned via the eventResponse when the `/presets/active` property changes on the device:

Name	Type	Description
.preset	string	Name of the active preset (with .cset extension, or 'default')

/camera/colorBars

Get the status of color bars display

The value JSON returned via the eventResponse when the `/camera/colorBars` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the color bars are currently enabled

/camera/programFeedDisplay

Get the status of program feed display

The value JSON returned via the eventResponse when the `/camera/programFeedDisplay` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the program feed display is currently enabled

/camera/tallyStatus

Get the tally status of the camera

The value JSON returned via the eventResponse when the `/camera/tallyStatus` property changes on the device:

Name	Type	Description
.status	string	Current tally status of the camera Possible values are: None, Preview, Program.

/camera/power

Get the power status of the camera

The value JSON returned via the eventResponse when the `/camera/power` property changes on the device:

Name	Type	Description
.source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
.milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
.batteries	array	
.batteries[i]	object	
.batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
.batteries[i]. chargeRemainingPercent	integer	Remaining battery charge percentage
.batteries[i].statusFlags	array	List of battery status flags
.batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

/camera/power/displayMode

Get the power display mode of the camera

The value JSON returned via the eventResponse when the `/camera/power/displayMode` property changes on the device:

Name	Type	Description
.mode	string	Current power display mode Possible values are: Percentage, Voltage.

/camera/timingReferenceLock

Get the timing reference lock status

The value JSON returned via the eventResponse when the `/camera/timingReferenceLock` property changes on the device:

Name	Type	Description
.locked	boolean	Indicates if timing reference is locked

/video/iso

Get current ISO

The value JSON returned via the eventResponse when the `/video/iso` property changes on the device:

Name	Type	Description
.iso	integer	Current ISO value

/video/supportedISOs

Get the list of supported ISO settings

The value JSON returned via the eventResponse when the `/video/supportedISOs` property changes on the device:

Name	Type	Description
.supportedISOs	array	Array of supported ISO values
.supportedISOs[i]	integer	

/video/gain

Get current gain value in decibels

The value JSON returned via the eventResponse when the `/video/gain` property changes on the device:

Name	Type	Description
.gain	integer	Current gain value in decibels

/video/supportedGains

Get the list of supported gain settings in decibels

The value JSON returned via the eventResponse when the `/video/supportedGains` property changes on the device:

Name	Type	Description
.supportedGains	array	Array of supported gain values in decibels
.supportedGains[i]	integer	

/video/whiteBalance

Get current white balance

The value JSON returned via the eventResponse when the `/video/whiteBalance` property changes on the device:

Name	Type	Description
.whiteBalance	integer	Current white balance

/video/whiteBalance/description

Get white balance range

The value JSON returned via the eventResponse when the `/video/whiteBalance/description` property changes on the device:

Name	Type	Description
.whiteBalance	object	
.whiteBalance.min	integer	Minimum color temperature
.whiteBalance.max	integer	Maximum color temperature

/video/whiteBalanceTint

Get white balance tint

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint` property changes on the device:

Name	Type	Description
.whiteBalanceTint	integer	Current white balance tint

/video/whiteBalanceTint/description

Get white balance tint range

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint/description` property changes on the device:

Name	Type	Description
.whiteBalanceTint	object	
.whiteBalanceTint.min	integer	Minimum white balance tint
.whiteBalanceTint.max	integer	Maximum white balance tint

/video/ndFilter

Get ND filter stop

The value JSON returned via the eventResponse when the `/video/ndFilter` property changes on the device:

Name	Type	Description
.stop	number	Current filter power (fStop)

/video/supportedNDFilters

Get the list of available ND filter stops

The value JSON returned via the eventResponse when the `/video/supportedNDFilters` property changes on the device:

Name	Type	Description
.supportedStops	array	Array of available ND filter stops
.supportedStops[i]	number	

/video/ndFilter/displayMode

Get ND filter display mode on the camera

The value JSON returned via the eventResponse when the `/video/ndFilter/displayMode` property changes on the device:

Name	Type	Description
.displayMode	string	ND filter display mode Possible values are: Stop, Number, Fraction.

/video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

The value JSON returned via the eventResponse when the `/video/supportedNDFilterDisplayModes` property changes on the device:

Name	Type	Description
.supportedDisplayModes	array	Array of supported display modes
.supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

/video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

The value JSON returned via the eventResponse when the `/video/ndFilterSelectable` property changes on the device:

Name	Type	Description
.selectable	boolean	True if ND filter adjustments are selectable via a slider

/video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

The value JSON returned via the eventResponse when the `/video/shutter` property changes on the device:

Name	Type	Description
.continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
.shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
.shutterAngle	number	Shutter angle

/video/shutter/measurement

Get the current shutter measurement mode

The value JSON returned via the eventResponse when the `/video/shutter/measurement` property changes on the device:

Name	Type	Description
.measurement	string	Current shutter measurement mode Possible values are: ShutterAngle, ShutterSpeed.

/video/supportedShutters

Get supported shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/supportedShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of supported shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array of flicker-free shutter speeds
.shutterSpeeds[i]	integer	

/video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/flickerFreeShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of flicker-free shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array
.shutterSpeeds[i]	integer	

/video/autoExposure

Get current auto exposure mode

The value JSON returned via the eventResponse when the `/video/autoExposure` property changes on the device:

Name	Type	Description
.mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
.type	string	Comma-separated list of device types in the auto exposure stack

/video/detailSharpening

Get the current state of detail sharpening

The value JSON returned via the eventResponse when the `/video/detailSharpening` property changes on the device:

Name	Type	Description
.enabled	boolean	Whether detail sharpening is enabled

/video/detailSharpeningLevel

Get the current detail sharpening level

The value JSON returned via the eventResponse when the `/video/detailSharpeningLevel` property changes on the device:

Name	Type	Description
.level	string	Current detail sharpening level of supported shutter speeds Possible values are: Low, Medium, High.

Help

The fastest way to obtain help is to go to the Blackmagic Design online support pages and check the latest support material available for your camera.

Blackmagic Design Online Support Pages

The latest manual, software and support notes can be found at the Blackmagic Design support center at www.blackmagicdesign.com/support.

Contacting Blackmagic Design Support

If you can't find the help you need in our support material, please use the 'Send us an email' button on the support page to email a support request. Alternatively, click on the 'Find your local support team' button on the support page and call your nearest Blackmagic Design support office.

Checking the Software Version Currently Installed

To check which version of Blackmagic Camera Setup utility software is installed on your computer, open the About Blackmagic Camera Setup utility window.

- On Mac, open Blackmagic Camera Setup utility from the Applications folder. Select About Blackmagic Camera Setup utility from the application menu to reveal the version number.
- On Windows, open Blackmagic Camera Setup utility from your Start menu or Start Screen. Click on the Help menu and select About Blackmagic Camera Setup utility to reveal the version number.

How to Get the Latest Software Updates

After checking the version of Blackmagic Camera Utility software installed on your computer, please visit the Blackmagic Design support center at www.blackmagicdesign.com/support to check for the latest updates. While it is usually a good idea to run the latest updates, it is wise to avoid updating any software if you are in the middle of an important project.

Regulatory Notices



Disposal of Waste of Electrical and Electronic Equipment Within the European Union.

The symbol on the product indicates that this equipment must not be disposed of with other waste materials. In order to dispose of your waste equipment, it must be handed over to a designated collection point for recycling. The separate collection and recycling of your waste equipment at the time of disposal will help conserve natural resources and ensure that it is recycled in a manner that protects human health and the environment. For more information about where you can drop off your waste equipment for recycling, please contact your local city recycling office or the dealer from whom you purchased the product.



This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. Operation of this product in a residential area is likely to cause harmful interference, in which case the user will be required to correct the interference at personal expense.

The operation of this equipment is subject to the following two conditions:

1. This equipment or device may not cause harmful interference.
2. This equipment or device must accept any interference received, including interference that may cause undesired operation.



R-R-BMD-20250213001



ISED Canada Statement

This device complies with Canadian standards for Class A digital apparatus.

Any modifications or use of this product outside its intended use could void compliance to these standards.

Connection to HDMI interfaces must be made with high quality shielded HDMI cables.

This equipment has been tested for compliance with the intended use in a commercial environment. If the equipment is used in a domestic environment, it may cause radio interference.

Safety Information

The supplied AC to 24V DC power supply for Blackmagic URSA Cine must be connected to a mains socket outlet with a protective earth connection.

No operator serviceable parts inside Blackmagic URSA Cine. Refer servicing to your local Blackmagic Design service center.

When connecting the input power connector to an external DC battery source, the supply wiring must include current limiting or fusing.

The wiring used should be marked VW-1 or comply with the relevant parts of IEC 60332 or IEC 60695".

We recommend that the DC source for this camera is 24V. If a 12V DC source is used, the wiring used should be sufficient for a current of 14A.

A B-Mount battery plate is provided with this product. We recommend the use of a 24V (28.8V) battery or dual voltage battery.

For a 12V (14.4V) battery, refer to your battery manual or markings to determine the Maximum Continuous Discharge Current. We recommend that this has a minimum rating of 12A.

State of California statement

This product can expose you to chemicals such as trace amounts of polybrominated biphenyls within plastic parts, which is known to the state of California to cause cancer and birth defects or other reproductive harm.

For more information go to www.P65Warnings.ca.gov.

European Office

Blackmagic Design Europe B.V.
Rijnlanderweg 766, Unit D
2132 NM Hoofddorp
NL

Blackmagic URSA Cine

During sunny conditions, consider shading of the camera to prevent exposure of the camera or Lithium battery to extended periods of sunlight. Keep Lithium batteries away from all sources of heat.

The 24V 'RS' output connector is suitable to provide power to the third party accessories. Please ensure the power consumption is less than 48W.

Warranty

Limited Warranty

Blackmagic Design warrants that this product will be free from defects in materials and workmanship for a period of 12 months from the date of purchase. If a product proves to be defective during this warranty period, Blackmagic Design, at its option, either will repair the defective product without charge for parts and labor, or will provide a replacement in exchange for the defective product.

In order to obtain service under this warranty, you the Customer, must notify Blackmagic Design of the defect before the expiration of the warranty period and make suitable arrangements for the performance of service. The Customer shall be responsible for packaging and shipping the defective product to a designated service center nominated by Blackmagic Design, with shipping charges pre paid. Customer shall be responsible for paying all shipping charges, insurance, duties, taxes, and any other charges for products returned to us for any reason.

This warranty shall not apply to any defect, failure or damage caused by improper use or improper or inadequate maintenance and care. Blackmagic Design shall not be obliged under this warranty: a) to repair damage resulting from attempts by personnel other than Blackmagic Design representatives to install, repair or service the product, b) to repair damage resulting from improper use or connection to incompatible equipment, c) to repair any damage or malfunction caused by the use of non Blackmagic Design parts or supplies, or d) to service a product that has been modified or integrated with other products when the effect of such a modification or integration increases the time or difficulty of servicing the product.

Exposing URSA Viewfinder to direct sunlight could damage the viewfinder display as the viewfinder optics act as a magnifying glass. Image retention or burn-in could happen on OLED panels when static or high contrast images, such as frame guides, are displayed on the panels for extended periods. To avoid this, ensure the IR sensor for face detection is not covered deliberately and disconnect the viewfinder when not in use for prolonged periods. Image retention is not covered by this product warranty.

THIS WARRANTY IS GIVEN BY BLACKMAGIC DESIGN IN LIEU OF ANY OTHER WARRANTIES, EXPRESS OR IMPLIED. BLACKMAGIC DESIGN AND ITS VENDORS DISCLAIM ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. BLACKMAGIC DESIGN'S RESPONSIBILITY TO REPAIR OR REPLACE DEFECTIVE PRODUCTS DURING THE WARRANTY PERIOD IS THE WHOLE AND EXCLUSIVE REMEDY PROVIDED TO THE CUSTOMER. BLACKMAGIC DESIGN WILL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES IRRESPECTIVE OF WHETHER BLACKMAGIC DESIGN OR THE VENDOR HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES. BLACKMAGIC DESIGN IS NOT LIABLE FOR ANY ILLEGAL USE OF EQUIPMENT BY CUSTOMER. BLACKMAGIC IS NOT LIABLE FOR ANY DAMAGES RESULTING FROM USE OF THIS PRODUCT. USER OPERATES THIS PRODUCT AT OWN RISK.

© Copyright 2025 Blackmagic Design. All rights reserved. 'Blackmagic Design', 'URSA', 'DeckLink', 'HDLink', 'Workgroup Videohub', 'Multibridge Pro', 'Multibridge Extreme', 'Intensity' and 'Leading the creative video revolution' are registered trademarks in the US and other countries. All other company and product names may be trademarks of the respective companies with which they are associated.

The Bluetooth® word mark and logos are registered trademarks owned by Bluetooth SIG, Inc. and any use of such marks by Blackmagic Design is under license. Other trademarks and trade names are those of the respective owners.

2025年10月
カメラガイド

Blackmagicdesign

Blackmagic

URSA Cine Immersive



Blackmagic URSA Cine Immersive

目次

Blackmagic URSA Cine Immersive 157
イマーシブとは 157
Apple Immersive Video 157
URSA Cine Immersive 158
はじめに 159
カメラの電源を入れる 159
収録設定を設定する 160
イメージをモニタリングする 162
モニタリングオプションおよびオーバーレイ 162
最初のショットをフレーミングする 163
カメラの高さ 163
カメラティルト 164
配置 165
視聴者の自由意志 166
カメラの動き 166
データレート表 167
オーディオおよびタイムコード 168
Apple Spatial Audio Format 168
アンビソニック 168
ピンマイクと小型マイク 169
手持ちマイク 169
タイムコードの接続 169
撮影 171
露出要件 171
シャッタースピード 171
ISOとダイナミックレンジ 172
ND 173
オフスピード収録 174
ヒストグラム 175
LCDモニターオプション 175
収録 176
再生 177
クリップの転送 178
概要 179
DaVinci Resolve Studio 180
ワークステーションの推奨事項 180
プロジェクトのセットアップ 181
マスター設定 181
ビデオモニタリング 182
イメージスケーリング 182
カラーマネージメント 183
DaVinci Wide Gamutについて 184
Apple Immersiveの処理 185
インジェスト 185
編集 187
タイムラインの構築 187
バックドロップ 187
ビューファインダーの使用 189
トランジション 191
Apple Immersive Motion Data 192
タイトル 193
創造面における考慮事項 194
VFX合成 195
メディア出力 (MediaOut) ノード 195
レンダラー3D (Renderder3D) 196
イマーシブパッチャーで合成 197
カラーグレーディング 198
ノイズ除去 198
プライマリーグレーディング 199
セカンダリーコレクション 199
サイズ調整 200
3Dパレット 200
CSTカラーマネージメント 202
CSTタイムラインマッピング 202
CSTグレーディングモニター・マッピング 203
CST Apple Vision Proモニタリング 205
オーディオ 206
Apple Spatial Audio Format (ASAF) 206
書き出し 207
トランジションのバイパス 207
Apple Vision Proへの配信 208
Apple Immersive Video Utilityについて 208
Developer Information 209
Camera Control REST API 209

Livestream Control API 211
Clips Control API 215
Media Pool Control API 216
Monitoring Control API 219
Event Control API 228
System Control API 228
Transport Control API 234
Timeline Control API 238
Media Control API 241
Slate Control API 244
Preset Control API 250
Audio Control API 252
Lens Control API 257
Video Control API 261
Camera Control API 269
Immersive Control API 272
Color Correction Control API 273
Notification websocket - 1.0.0 278
messages 278
ヘルプ 302
規制に関する警告 303
安全情報 304
保証 305

Blackmagic URSA Cine Immersive

Blackmagic URSA Cine Immersiveをご購入またはレンタルいただき、ありがとうございます。

Blackmagic URSA Cine Immersiveは、イマーシブビデオと呼ばれる180°のステレオスコピックVRフッテージを撮影できるように設計されています。特に、現在利用可能な中で最高品質のイマーシブ体験ができるApple Vision Pro用のApple Immersive Videoの撮影に最適化されています。

このガイドでは、Blackmagic URSA Cine Immersiveの基礎情報と撮影方法、特にカメラの基本的な操作方法とApple Immersive Videoの書き出し用の独自のフォーマットに焦点を絞って説明しています。このフォーマットを理解することで、カメラ自体もより深く理解できます。

以下には、カメラを使用する上で役立つだけでなく、イマーシブフォーマットの利点を理解する手助けとなる技術面と創造面におけるアドバイスが記載されています。

イマーシブとは

イマーシブは3次元空間ビデオフォーマットです。おそらく、他の同様のテクノロジーが以前から存在していたことはご存知でしょう。イマーシブとの違いを把握するために、他のテクノロジーの大まかな概要を説明します。

- **従来の3D映画**では、人間の目の間の距離を模倣した距離に2台のカメラを置いて各シーンを撮影することにより、観客はフレーム内の奥行きを体験できます。ステレオスコピックイメージでは、撮影監督の構図に基づき、視聴者の視点をコントロールすることで奥行きを体験できます。
- **全天球ビデオ**は3Dの原理と360°カメラを組み合わせることで、あらゆる方向を向いても、奥行きのある映像を見ることができます。360°の映像は、視聴者が見ている部分的な映像でも十分な忠実性が得られるように、極めて高い解像度が求められます。生理学上、人は片方の肩からもう一方の肩までしか首を回せないため、通常、後ろも見えるように、視聴者は立位か、回転椅子に座って映像を視聴します。
- **ボリューメトリックビデオ**では、視聴者が奥行きを感じられ、全方向の映像を見られるだけでなく、自由に動いて異なる視点からの映像を見られます。ボリューメトリックビデオの撮影はカメラアレイで行えますが、ほとんどのバーチャル・リアリティや拡張現実は、動的にレンダリングされた3Dアセットを使用し、視聴者が映像を見る際に安全な環境にいることが求められます。

イマーシブは、従来の映画と3Dのインタラクティブフォーマットの優れた点を組み合わせた映像です。最もシンプルな形式では、高フレームレート、高解像度の180°のステレオスコピック・フォーマットです。

Apple Immersive Video

Blackmagic URSA Cine Immersiveは、Apple Immersive Videoの撮影用に特別に設計されました。このフォーマットはApple Vision Proで使用され、180°の視野で高解像度のステレオスコピック3Dビデオに対応しています。

Apple Immersive Videoの特徴は以下の通りです。

- 90fpsの再生。
- 片目あたり4320 x 4320までの解像度。
- 元のカメラの映像を180°の環境にマッピングする為の独自のメタデータシステム。
- Apple Spatial Audio Formatによる3Dサウンド。

Apple Immersive Videoには、180°の高解像度ステレオスコピック再生に対応したハードウェアが必要です。このフォーマットに対応しているのは、現在Apple Vision Proのみです。



Apple Vision Pro

このヘッドセットは、3660 x 3200ピクセル（7.5ミクロンのピクセルピッチ）のデュアル・マイクロ有機ELディスプレイを搭載しています。映像を頭と目の動きに合わせるために、13のトラッキングカメラ、LiDARスキャナー、慣性計測装置、アイトラッキングセンサーが搭載されています。

8コアCPUと10コアGPUに対応したM2プロセッサーにより、100fpsまでの再生をサポートしています。

URSA Cine Immersive

Blackmagic URSA Cine Immersiveは、Apple Vision Pro用のApple Immersive Videoを撮影するために開発されました。



Blackmagic URSA Cine Immersive

URSA Cine Immersiveは、2つの一致する58メガピクセルのRGBWセンサーを使用します。このセンサーはほぼ正方形で、縦方向と横方向両方の完全な180°の視野をキャプチャーします。各センサーは210°の魚眼レンズとペアになっています。レンズ中心間距離は64mmです。

内蔵のNDフィルターでは、最大8ストップの光量をコントロールできます。レンズアパーチャーは、約f4.5に固定されています。

カメラは、2ストリームの8K映像を90fpsで収録します。これにより、通常の4K 24p収録より遥かに高いデータレートとなります。

はじめに

Blackmagic URSA Cine Immersiveは、カメラを電源に接続して、電源をオンにするだけで簡単に使用できます。また、Blackmagic Media Module 8TBがカメラのモジュールベイにロードされており、クリップの収録を即座に開始できます。

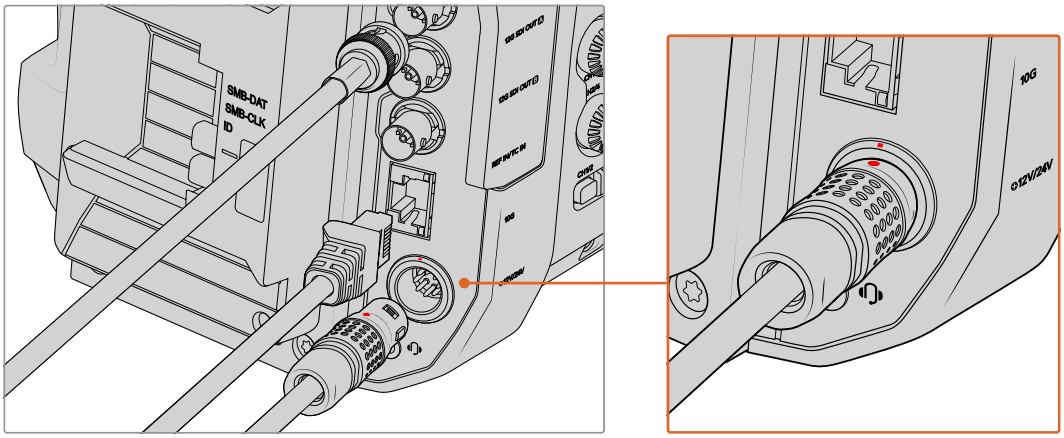
カメラの電源を入れる

カメラに電源を供給する一番簡単な方法は、同梱の電源アダプターで外部電源に接続することです。

外部電源に接続する：

1. 保護用のゴム製ダストキャップを外し、失くさないように保管してください。AC—24V DCアダプターのプラグをIEC C13主電源ケーブルで標準の主電源ソケットに接続します。
2. アダプターの8ピンコネクターを、カメラのリアパネルにある24V DC電源コネクターに接続します。

外部電源とバッテリー電源に接続している場合、外部電源のみが使用されます。充電されたバッテリーが接続されている時に外部電源を外した場合、カメラは動作を中断することなく、バッテリー電源に切り替えます。



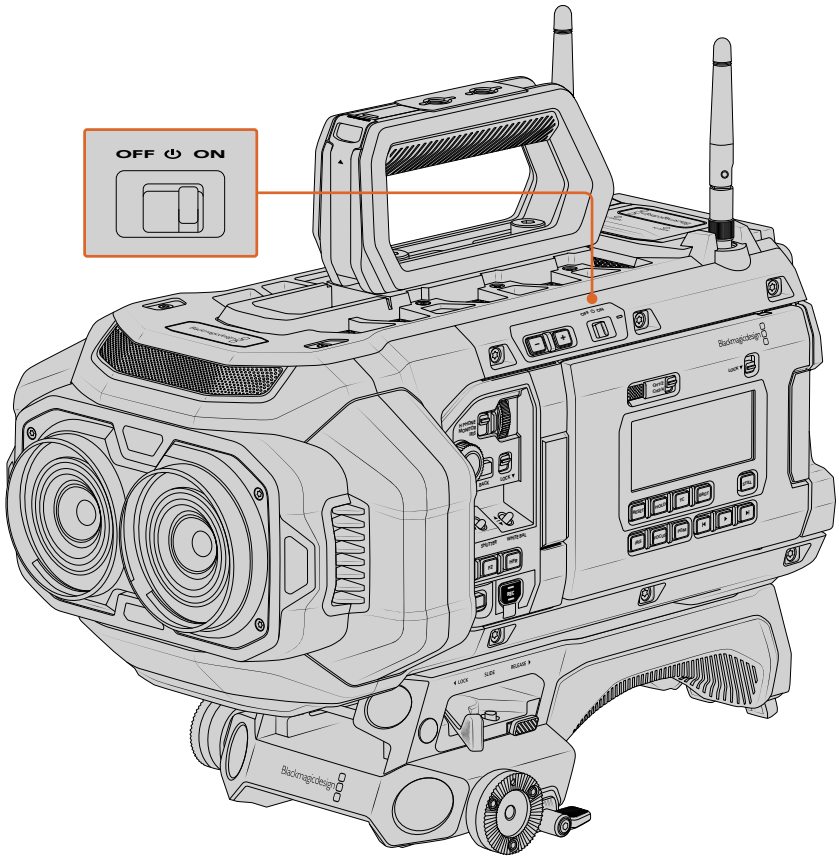
電源に接続するには、本体とコネクターの赤いドットを合わせ、適切な位置に固定されるまで、コネクターを押し込みます。接続を外すには、メタルスリーブを持ち、カメラからゆっくりと引き抜きます。これにより、スリーブが後ろにスライドし、ソケットからコネクターがリリースされます。

カメラに同梱されているバッテリープレートはBマウントプレートで、24VのBマウントのバッテリー用に作られています。URSA Cine Immersiveとの使用には、24〜34Vの電圧範囲で使用することが理想的です。電圧が高いほど熱損失が減少するため、高フレームレートでの収録や、カメラから多数のアクセサリへの給電が可能です。

また、24Vに対応したVLockおよびGold Mountプレート、26Vのサードパーティ製バッテリー用のGold Mount Plusプレートをオプションで取り付けて使用することも可能です。

メモ URSA Cine Immersiveを12〜18Vの電源またはバッテリーで駆動する場合、3ピンのRSコネクターまたは2ピンのLemo型コネクターでアクセサリに給電することはできません。

バッテリーをマウントするか、外部電源に接続したら、URSA Cine Immersiveの電源をオンにします。カメラ上部のスイッチを「ON」の位置にするとカメラがオンになります。



電源スイッチを「ON」の位置にするとカメラがオンになります

収録設定を設定する

このセクションは、イマーシブコンテンツの撮影における独自のカメラ設定または重要な設定について説明します。カメラの設定メニューを開くには、開閉式タッチスクリーンまたはアシスタント用タッチスクリーンの右上にあるメニュー設定アイコンをタップします。



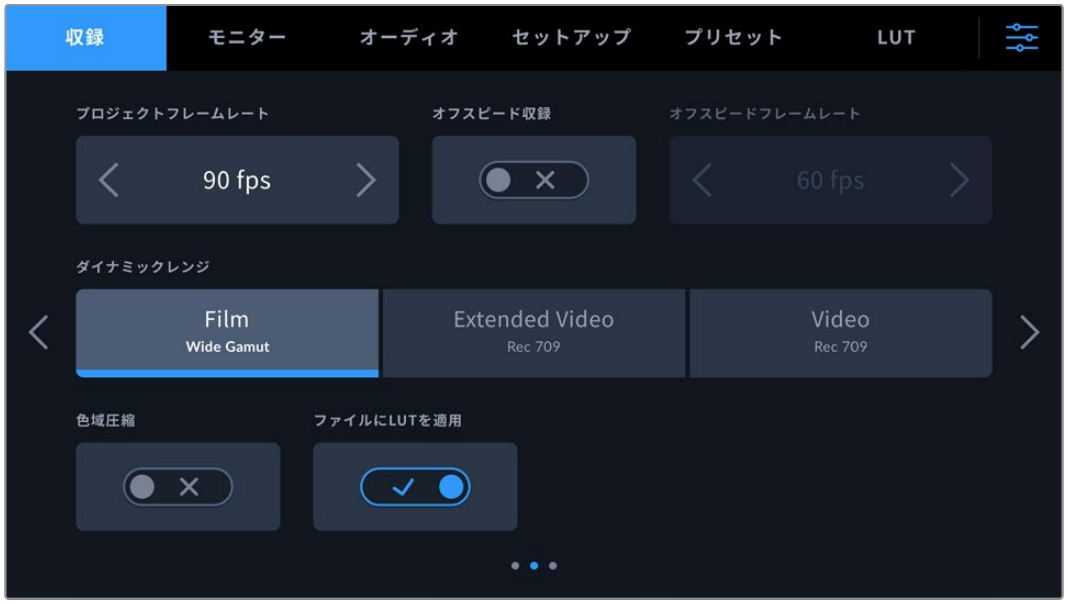
収録設定 ページ 1

このページの「コーデックと品質（CODEC AND QUALITY）」オプションでは、作業中のプロジェクトに適したBlackmagic RAWコーデック設定を選択できます。Blackmagic RAWの設定は、それぞれ4種類の「固定ビットレート（Constant Bitrate）」と「固定クオリティ（Constant Quality）」から選択できます。Apple Vision Pro用に90fpsのコンテンツを収録している場合は、12:1またはQ3の圧縮設定を強く推奨します。これらの設定により、現在のコンピューティング・ハードウェアで扱いやすいデータレートとファイルサイズを維持しながら、極めて高画質の映像が得られます。



収録設定 ページ 2

「収録 (RECORD)」設定タブの2ページ目には、フレームレートの設定が含まれます。



カメラのプロジェクトフレームレートは、Apple Vision Proの表示フレームレートに合わせて、90fpsに設定してください。

その他のシステム向けに撮影している場合は、60fpsや50fpsなどの低めのフレームレートで十分でしょう。24fpsや25fpsなどの従来の映画で使用される低いフレームレートは、24pの映画コンテンツ用にVFXのプレートを撮影している場合に適しています。

URSA Cine Immersiveを使い始めるにあたって必要な準備はこれですべてです！カメラのMedia Moduleはすでにフォーマット済みなので、すぐに収録を開始できます。

イメージをモニタリングする

URSA Cine Immersiveでは、撮影中にイメージをモニタリングする方法が複数あります。

開閉式タッチスクリーンは、カメラのオペレーターが使用することを意図して作られています。スクリーンは回転させて筐体に収めることで、物理的に幅を取らずに、映像を確認できます。もしくは、自分自身を撮影している場合に、前方に向けて自分でモニタリングすることも可能です。

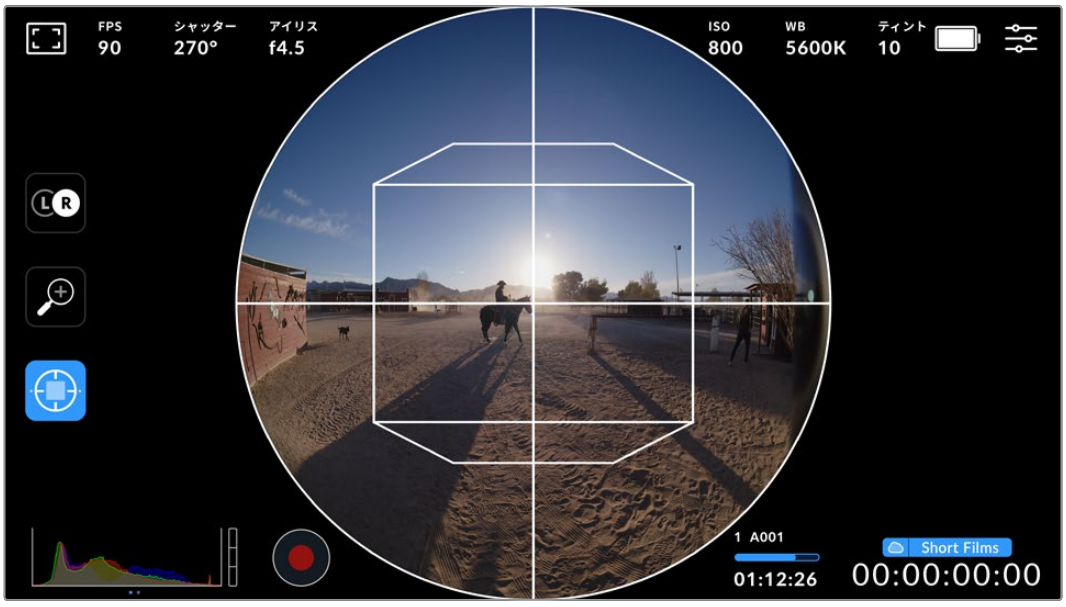
アシスタント用タッチスクリーンでは、カメラオペレーターの邪魔にならない位置で別のスタッフが映像をモニタリングできます。カメラの視野が極めて広いため、アシスタント用タッチスクリーンを使用する際には注意が必要です。





モニタリングオプションおよびオーバーレイ

カメラはレンズ空間を表示し、片目ずつ確認できます。開閉式タッチスクリーンおよびアシスタント用タッチスクリーンは、180°の視野を表示するようにマスキングされます。SDI出力は、各センサーでキャプチャーされたフルの210°の映像を表示します。



全モニタリング出力は同じ目の映像を表示します。左右の目を切り替えるには、いずれかのタッチスクリーンに表示されている「L R」ボタンをタップします。

オプションのオーバーレイは、オーバーレイボタンを押すとタッチスクリーンで有効になります。このオーバーレイには以下が含まれます。

- 180°の外側の境界線
- フレームの中央を示す中央十字線
- 真っ直ぐ正面を向いている場合に、Apple Vision Proで見えるおよその視野を示すガイド

これらのツールは、イマーシブ撮影において適切なフレーミングと配置を行う手助けになります。

最初のショットをフレーミングする

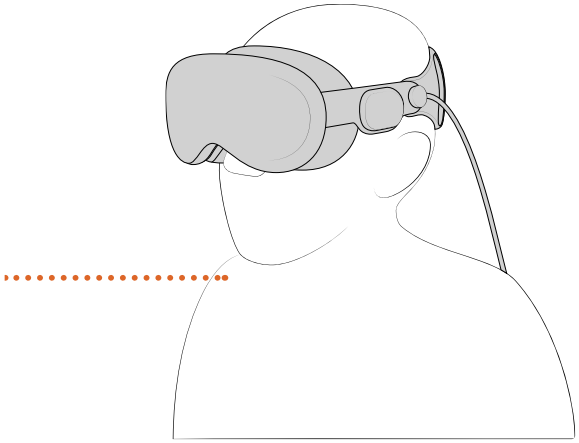
イマーシブと従来の2Dコンテンツは多くの点で似ていますが、明確な違いもあります。

例えば、電話を車内でスクロールしていたことが原因で車酔いしたことはありませんか？ これは、脳が異なる信号を受け取り、感覚のミスマッチが生じることが原因として起きます。内耳では動いていると感知されますが、目では静止していると認識するからです。

イマーシブでも同様のことを簡単に再現できます。例としては、カメラをパンした場合、視聴者の目は動いていると体に伝えますが、首と内耳は完全に静止していると認識します。このことから、視聴者が快適に視聴できるカメラの動きは、静止ショットをマスターしてから学ぶことをお勧めします。

カメラの高さ

従来の2D映画では、シーン内の力関係を表すためにカメラの高さを変える場合があります。イマーシブの撮影では、カメラを地面に配置する影響を考慮する必要があります。視聴者にどのような影響を与えるでしょうか？ あたかも、首まで埋められているような感覚を与える可能性があります。



地面の知覚的な高さは、視聴者の視聴体験に影響を与えかねません。低い位置に配置したカメラの映像では、視聴者が地面の高さに違和感を覚える可能性があります！



イマーシブの撮影においては、低いアングルでは視聴者が不快感を感じる場合があります。

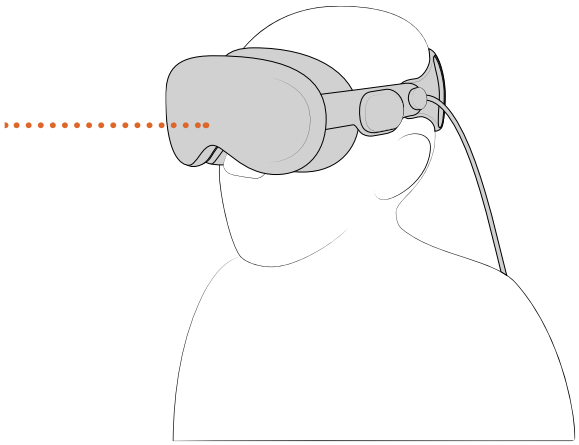
実際にその場にいるかのように感じさせるためには、多くの場合、そのシーンで視聴者が立っているか、座っているかと感じられるようなカメラの高さと配置を選択することが最善でしょう。クリエイティブな意図を表現する方法は他にもあります。シーンに登場人物がいる場合、カメラを若干高く配置すると、視聴者が他の人々を見下ろすことになるため、力関係が視聴者側にあることを感じさせることが可能です。逆に、若干低いカメラ位置では、視聴者は他の人々を見上げる視点になります。

しかし、視聴者が快適に視聴できて、かつ冒険度の高いカメラ配置が存在しないわけではありません。飛行カメラ、俯瞰ビュー、低アングルなど、多数のテクニックがイマーシブ制作で効果的に使用されています。適切な編集、動き、コンテクストによって、視聴者が楽しめる映像を作り出すことが可能です。

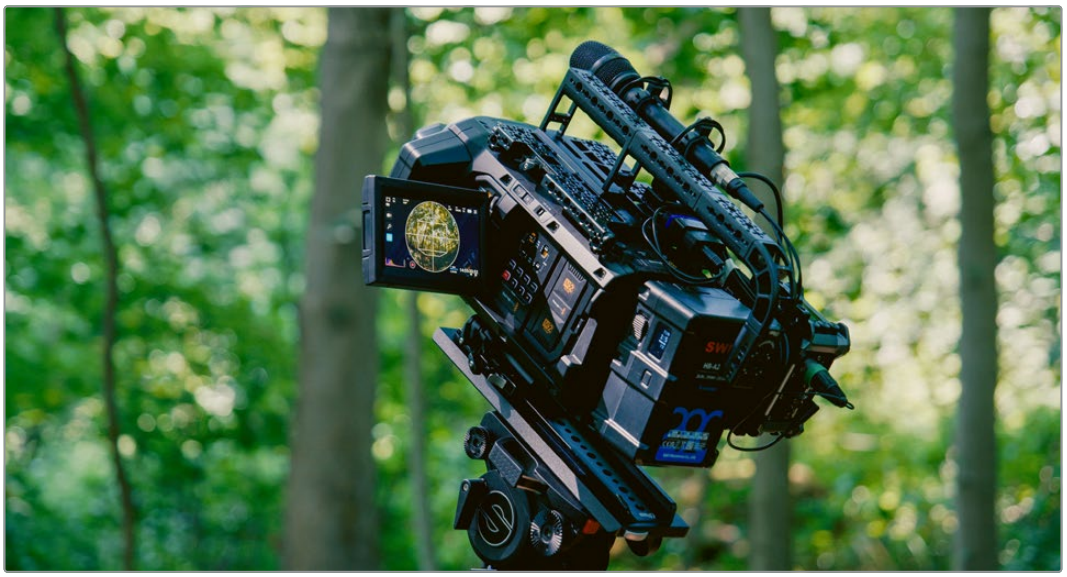
カメラティルト

従来の2Dコンテンツの撮影において、カメラを高くまたは低く配置する際、多くの場合、撮影監督は構図を維持するためにカメラティルトを用います。これがイマーシブ映像に与える影響を考えてみてください。

カメラを下向きにした場合、視聴者が水平線を見るためには首を上向きにする必要があります。これにより、画面酔いを起こすことはないと思われますが、実世界では水平線は通常は頭を真正面に向けた状態で見えるため、没入感が失われる可能性があります。



イマーシブのショットでは、水平線が0°にあると
最も高い没入感が得られます。



URSA Cine Immersiveを傾けると、映像で水平線が見える向きと、視聴者が実世界で水平線が見えると予測する向きとの違いにより、没入感が失われます。

こういった理由により、イマーシブショットのほとんどは0°またはほぼ0°のティルトで撮影されます。カメラは縦方向と横方向両方で180°以上の視野をキャプチャーするため、視聴者は見上げたり、見下げることができます。しかし、映像の制作者が視聴者に特定の方向を向かせるのではなく、視聴者が自身で任意の方向を見ることが可能です。

映像の異なる部分に視聴者の視線を誘導するためには、照明、アパーチャー、フレーミング、リーディングラインなど、その他のテクニックを使用できます。

配置

イマーシブは多くの場合、座った状態で視聴します。拡張現実やVRフォーマットではないため、視聴者は立位で見たり、動き回ることはありません。現在、座っていますか？ 試しに、頭を左右に回したり、上下に動かしたりしてください。体を動かさずに、首を痛めない範囲でどれくらい首を回せますか？ イマーシブフォーマットの視野は、ほとんどの人の可動域を超えています。これは、左右を向いても映像が周辺視野をカバーすることを意味するので良いことです。この実験は、視野のどこにメインのアクションを配置するべきかを理解する助けになります。

典型的な頭部の可動域：

120〜160°の回転
屈曲と伸展：110〜130°

首の動きと目の動きを組み合わせると、自分の肩越しに見ることもできますが、ほとんどの人にとって長時間維持するには不快な姿勢です。

構図を決める助けに、カメラのオーバーレイをオンにします。典型的な視野を示すマーク内にアクションを収めることで、コンテンツが見やすくなります。

フレーム内の関心ポイントの動きも考慮するべきです。人物が右端から左端を歩くのを追うには、極端な首の動きが必要になります。同様に、映像の端に関心ポイントがあるショットから、その真逆に関心ポイントがあるショットに切り替わると、極端な首の動きが必要になります。構図を慎重に検討することで、創造性を発揮しつつ、視聴者が快適に視聴できる映像を制作できます。

視聴者の自由意志

従来の2Dコンテンツの撮影では、視聴者の注意を向ける目的で、フレーム、レンズ、フォーカス設定を使用できます。イマーシブの視野は180°であるため、情報が映らないようにしたり、視聴者の注意を向けることが難しくなります。イマーシブでは、視聴者は遥かに自由に映像を見ることができます。これは、イマーシブコンテンツが人をひきつける要素の一つです。指向性のある音、照明、ブロッキング、構図などを慎重に用いることで、視聴者の注意を引くことが可能です。

カメラの動き

イマーシブカメラは常に静止している必要はありませんが、動きによっては不自然または不快に感じられることがあるので注意が必要です。イマーシブカメラで快適に視聴できる動きについて理解するにあたって、車が良い例えになるでしょう。大型車両は大きな質量と慣性を持つため、速度や進行方向を急に変えることが困難です。そのため、時間をかけて加速や減速するので、予測可能かつ制御された方法で動きます。静止ショットの撮影をマスターし、動くショットを撮影するにあたって、以下の要素を考慮に入れると、違和感のない動きの映像を制作する役に立ちます。

- **継続時間：**視聴者がカメラの動きに慣れ、楽しめるように、時間に余裕を持たせます。短い動きを繰り返すと、方向感覚を失う映像になる可能性が高くなります。
- **動きの激しさ：**緩やかな動きは、イマーシブ編集に統合しやすいです。激しい動きは統合することは可能ですが、視聴者のことを考えると、ウォームアップとクールダウンが必要かもしれません。
- **視覚的な指標：**これは、シーン内の動かない要素です。視覚的な指標が無いショットでは、方向感覚が失われる可能性があります。車内のインテリアや地平線などの遠く離れたものなど、前景の要素は動くショットで方向感覚を得る上で役に立ちます。

データレート表

以下の表は、90fpsと24fpsで撮影した8K x 7Kイマーシブコンテンツのデータレートです。

Blackmagic RAW固定ビットレートで90fps使用時の最大データレート

8K x 7Kイマーシブ			
コーデック	**サイズ**	**フレームレート**	**データレート**
Blackmagic RAW 5:1	8,160 x 7,200 ステレオ	90	3,203 MB/s
Blackmagic RAW 8:1	8,160 x 7,200 ステレオ	90	2,002 MB/s
Blackmagic RAW 12:1	8,160 x 7,200 ステレオ	90	1,335 MB/s
Blackmagic RAW 18:1	8,160 x 7,200 ステレオ	90	889 MB/s

Blackmagic RAW固定ビットレートで24fps使用時の最大データレート

8K x 7Kイマーシブ			
コーデック	**サイズ**	**フレームレート**	**データレート**
Blackmagic RAW 5:1	8,160 x 7,200 ステレオ	24	854 MB/s
Blackmagic RAW 8:1	8,160 x 7,200 ステレオ	24	534 MB/s
Blackmagic RAW 12:1	8,160 x 7,200 ステレオ	24	356 MB/s
Blackmagic RAW 18:1	8,160 x 7,200 ステレオ	24	237 MB/s

オーディオおよびタイムコード

オーディオは、イマーシブ体験の欠かせない要素のひとつです。URSA Cine Immersiveを専用のオーディオ録音機器と共に使用することが重要です。

Apple Spatial Audio Format

Apple Spatial Audio Format（ASAF）は、Apple Immersive Videoを視聴する際に使用する3Dオーディオです。ASAFはAppleの空間オーディオフレームワークの一環で、3種類のオーディオ収録を保存できます。

- **チャンネルベース：**ASAFファイルには、従来のサラウンドミックスをエンベッドできます。これは、位置が固定されたナレーションや音楽に最適です。
- **オブジェクトベース：**サウンドは、3D空間の位置を示すメタデータ付きの個別のオブジェクトとして、スピーカーの個別チャンネルから独立して扱うことが可能です。これらは、通常はモノ収録です。3D座標は、シーン内で移動するオブジェクトを追うようにアニメートすることが可能です。これは、物語世界の音に最適です。つまり、シーン内に表示されているオブジェクトから発せられるサウンドです。
- **アンビソニック：**球面調和関数を用いて、全方向にオーディオをレンダリングできる全天球サラウンドフォーマットです。これは、環境音に適しています。

このセクションの残りでは、効果的なASAFファイルを構築する上で必要となる機器の概要を説明します。

アンビソニック

アンビソニックマイクには、高精度の幾何学模様に配置された4つ以上のカプセルが含まれています。各カプセルは、音源の方向に基づいて、同じサウンドを異なる到着時間と強度でキャプチャーします。これらのカプセルは、後でコンピューターで処理するために個別のチャンネルに収録します。違いを比較することで、ソフトウェアは収録したサウンドの方向と性質を推測できます。



1次アンビソニックマイク

アンビソニックマイク

アンビソニックマイクは、カプセルの数によって分類されています。カプセルの数が多いほど、マイクが空間のディテールをより高精度に再現できます。

1次アンビソニック：4チャンネル

2次アンビソニック：9チャンネル

3次アンビソニック：16チャンネル

4次アンビソニック：25チャンネル以上

マイクがカメラに搭載されているものか、外付けかどうかに関わらず、アンビソニックマイクは必ずカメラの撮影方向と正確に一致している必要があります。さらに、アンビソニックマイクは、エンドファイア、ブロードサイド、アップワード、ダウンワードなど、異なるコンフィギュレーションで使用できるので、音響技術者がどのコンフィギュレーションを使用しているかを把握しておくことが重要です。この情報は、オーディオレコーダーのメタデータを使用するか、メモを残しておくことで記録できます。

アンビソニックのサウンドは、環境音ベッドとして使用できます。他のチャンネルとオブジェクトベースのサウンドエフェクトは、その上にレイヤーできます。カメラの向きと一致している場合、オーディオエンジニアがサウンドデザインをさらに行う際に、アンビソニックのサウンドはリファレンスとしても使用できます。

ピンマイクと小型マイク

180°のカメラで撮影している場合、ブームマイクのポールはシーンに映り込んでしまうので使用できません！つまり、物語世界の音や会話音声を録音するには、ピンマイクや隠しマイクを使用する必要があります。

例えば、以下のような方法が挙げられます。

- ワイヤレストランスミッターとレシーバーを使用して、オーディオソースをURSA Cine Immersiveの2つのXLR入力に直接接続する。
- 外部オーディオレコーダーに接続する。大型の複数チャンネルデバイス、または身体装着型の小型デバイスのいずれも可能。

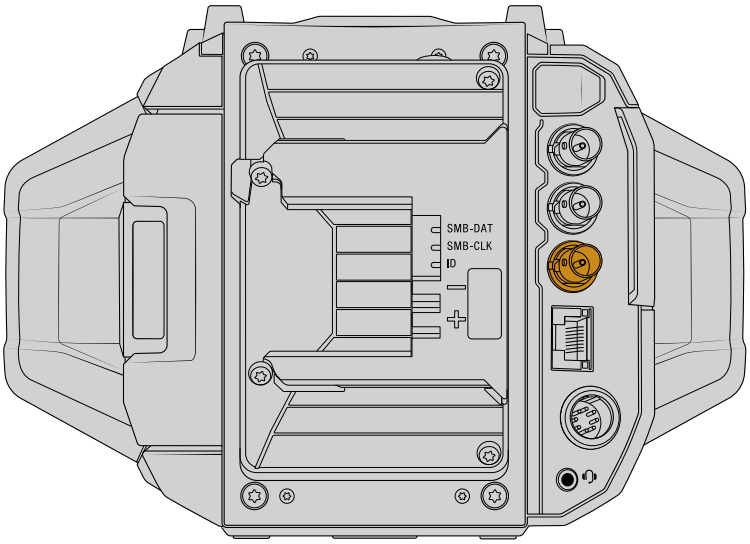
これらの収録は、いずれかのチャンネルまたはオブジェクトベースのサウンドと同期して、ミックスできます。

手持ちマイク

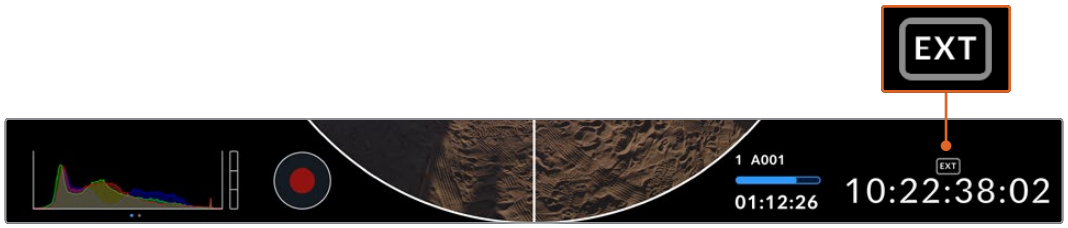
カメラでの撮影終了後に現場で追加のサウンドエフェクトを録音することで、ポストプロダクションの時間が節約できるだけでなく、よりリアルな音響が得られることがあります。手持ちのハイパーカーディオイドなどのマイクを制作に持ち込むのは非常に有益です。これにより、マイクをできる限り近くに配置してアクションを再び演じることができるからです。これらの収録は、いずれかのチャンネルまたはオブジェクトベースのサウンドとして、最終ミックスに編集できます。

タイムコードの接続

外部タイムコードをURSA Cine Immersiveに接続するには、BNCケーブルを使用して、タイムコードジェネレーターをカメラのリアパネルにあるタイムコード入力に接続します。デュアルシステムで撮影する際に、同期したタイムコードを接続することで、オーディオと映像をポストプロダクションで簡単に同期できます。



フレームレートがマッチした外部タイムコードがこのコネクターに接続されると、送信されてくるタイムコードにカメラは自動的にロックします。外部タイムコードがロックされるとURSA Cine ImmersiveのLCDのタイムコードカウンターの隣に「EXT」と表示されます。ケーブルを外すと、タイムコードはジャムシンクし続け、「EXT」から「INT」に変わり、カメラの内部タイムコードクロックに切り替わったことを示します。



タイムコードジェネレーターに接続されると、カメラのタイムコードは「EXT」と表示

メモ 23.976、24、30fpsなどの一般的なフレームレートでは、SMPTEタイムコードの規格が存在します。しかし、90fpsに対するSMPTEタイムコードの規格はありません。そのため、URSA Cine Immersiveのプロジェクトフレームレートが90fpsに設定されている場合、カメラは実際30fpsのタイムコードを使用します。SDI出力も30pで稼働します。タイムコードアクセサリや外部レコーダーは、30fpsに設定してください。

撮影

Blackmagic URSA Cine Immersiveカメラの基本的な操作方法と、イマーシブフォーマットを活用する方法を学んだので、本章ではカメラの操作方法についてさらに詳しく紹介します。

露出要件

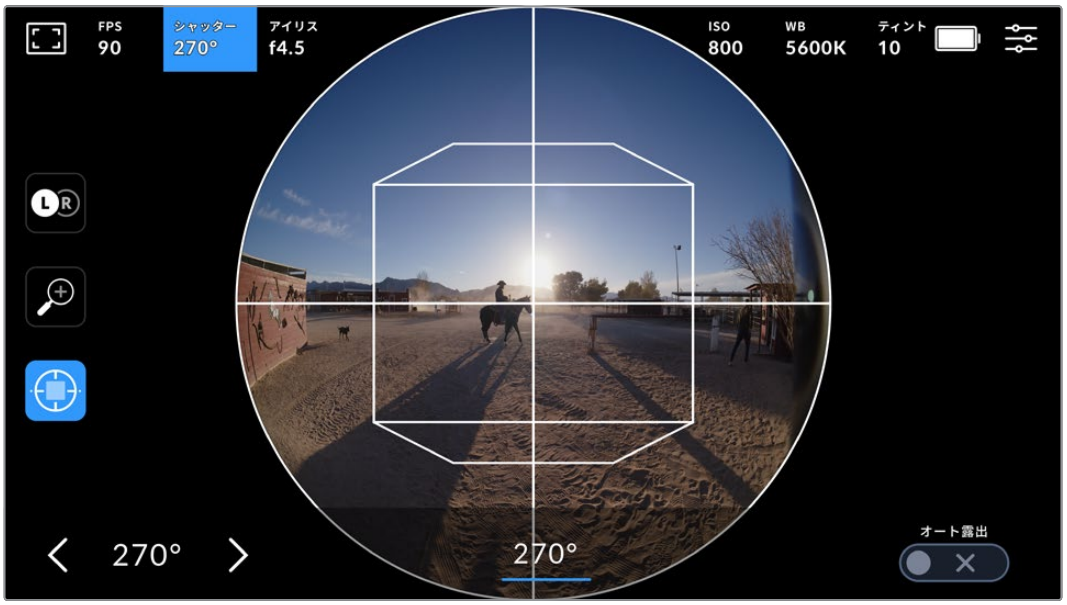
Apple Immersive Videoの露出要件は、従来型の24fpsでの撮影と比較して高く、撮影と照明のアプローチを考え直す必要があるかもしれません。180°のシャッターを維持する場合、90fpsでの撮影で同じ露光量を得るには、ほぼ4倍の光量が必要になります。さらに、レンズのアイリスはおよそf4.5に固定されているので、通常の超高速レンズより最大で3ストップ暗くなります。つまり、適切な露出を得るには、最大で7ストップ分多い光量が必要になる可能性があります。しかし、90fpsでの撮影では、270°から360°にシャッターアングルに設定することで、高フレームレートの光量のロスを補い、程よいモーションブラーを得る助けになります。

シャッタースピード

「シャッター」インジケーターは、シャッターアングルまたはシャッタースピードを表示します。「セットアップ（SETUP）」メニューの「シャッター表示（SHUTTER MEASUREMENT）」設定で、シャッター情報の表示方法を「アングル（Shutter Angle）」と「スピード（Shutter Speed）」で切り替えられます。

シャッタースピードを変更する：

1. 上部ツールバーで「シャッター（SHUTTER）」アイコンをタップします。
2. 左右の矢印を使用して、一般的なシャッタースピードのプリセットを切り替えます。特定のシャッターの値を使用したい場合、スクリーンの左下にある現在のシャッターインジケーターをダブルタップします。キーパッドが表示されるので、小数点以下2桁までのシャッターの値（11.2から360°まで）を入力できます。



画面上のキーパッドを使用すると、カスタムフレームレートを入力できます。

シャッタースピードが高フレームのイマーシブコンテンツの見た目に与える影響は多くありません。180°のシャッターは、24または30pでの撮影に多く使用されていますが、イマーシブ用の撮影では露出を助けるためにシャッタースピードの方がより一般的に使用されています。

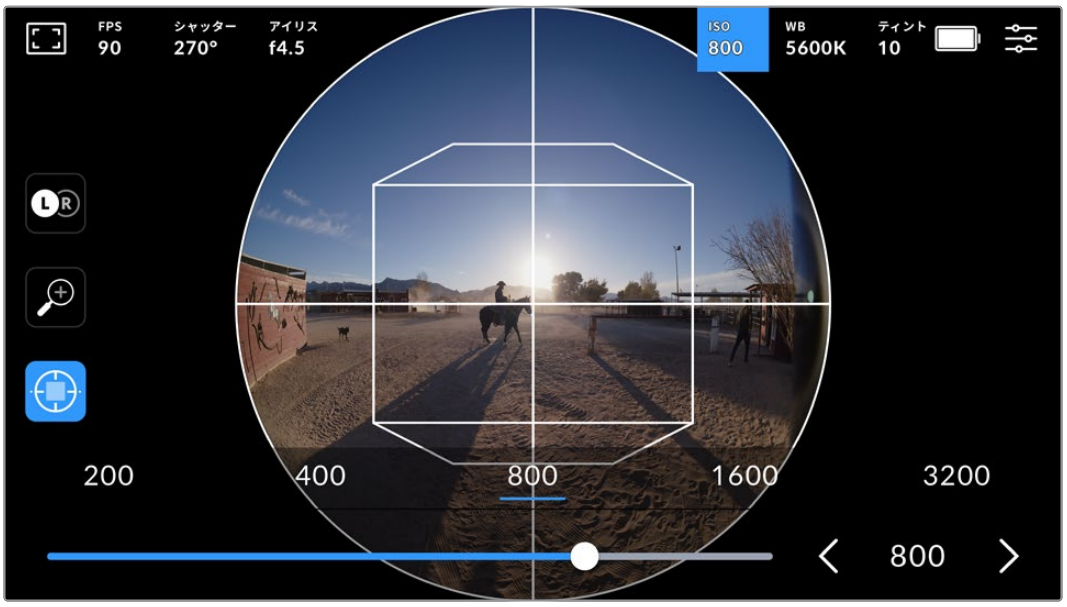
低照明での撮影では、シャッターは270°に上げることができます。一部の例外的な状況では、360°まで上げられます。

明るい条件での撮影では、シャッタースピードは11.2°まで下げられます。しかし、明るい条件では、モーションブラーを考慮してシャッタースピードを任意の設定のままにして、NDフィルターで露出をコントロールする方が一般的です。

Apple Immersive Videoの90fpsの高フレームレートでは、ちらつく光での撮影で問題が生じる可能性があることを理解することは重要です。60hzの環境では270°のシャッター、50hzでは324°を使用することが推奨されます。

ISOとダイナミックレンジ

「ISO」インジケーターは、カメラの現在のISO設定（光感度）を表示します。このインジケーターをタップすると、様々な照明条件に適合するようISOを調整できます。URSA Cine Immersiveは、ISO 200から3200に対応しています。最適な数値はISO 800です。



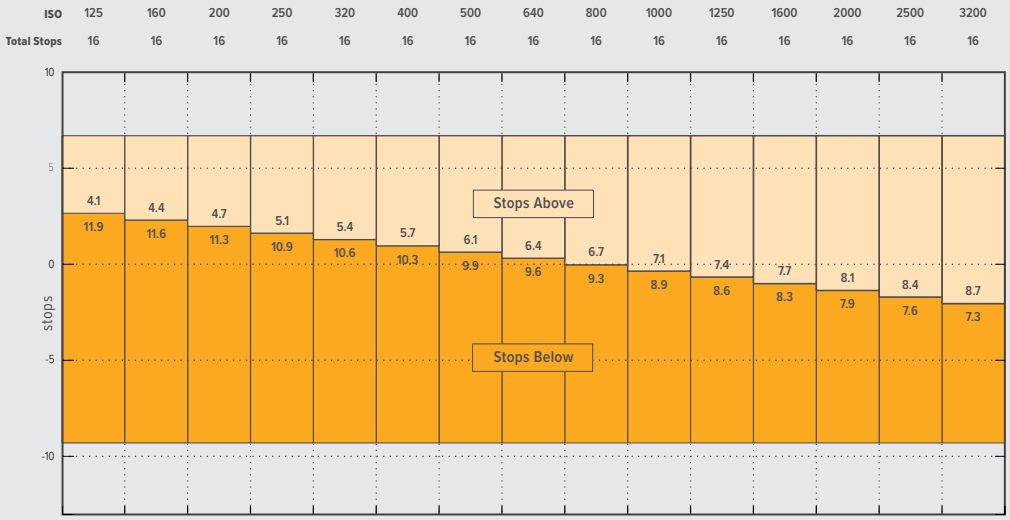
これはISOインジケーターをタップすることで調整できます。

数値のプリセットでISOを設定でき、また番号をタップすると1段ずつ調整できます。プリセットの下にあるスライダーで、これらのISO設定間を1/3段単位で微調整できます。

状況に応じて、ISOは高い/低い設定に変更できます。例えば、低照明条件ではISO 1600が適していますが、ノイズが発生することもあります。照明量が多く明るい場合は、ISO 200に設定すると、よりリッチなカラーが得られます。

Blackmagicのカメラでは幅広いダイナミックレンジが得られるため、撮影時およびDaVinci Resolveでのカラーグレーディングにおいて高い柔軟性が得られます。以下の表は、URSA Cine Immersiveで使用できるダイナミックレンジを示しています。

URSA Cine Immersiveのダイナミックレンジ



ND

カメラのNDフィルターは、露出をコントロールする最も一般的な方法で、カメラのセンサーに入る光量を露出（F値）のプリセットで削減できます。



URSA Cine ImmersiveのNDフィルターは、開閉式タッチスクリーンの左上にあるボタンで調整できます。クリアフィルターの他、F値を2、4、6、8から選択できます。

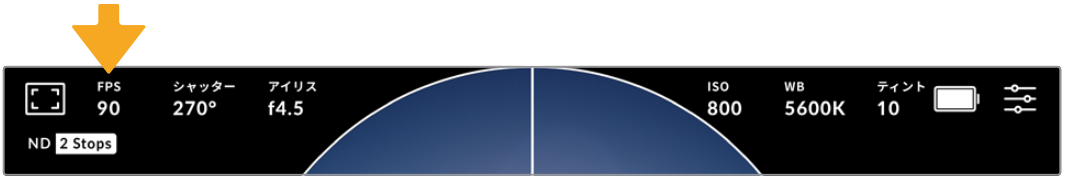
現在のNDフィルターの設定は、タッチスクリーン左上のインジケーターに表示されます。NDが使用されている場合、インジケーターは表示され続けます。NDがクリアに設定されている場合、インジケーターは4秒後に消えます。



メモ NDフィルターの表示は、使い慣れた単位に変更できます。「セットアップ（SETUP）」メニューで、「番号（Number）」、「F値（Stop）」、「分数（Fraction）」から選択できます。

オフスピード収録

「セットアップ（SETUP）」メニューの「プロジェクトフレームレート（PROJECT FRAME RATE）」は、URSA Cine Immersiveの収録フォーマットのフレームレートで、フッテージが再生される速度を定義します。通常、これは、ポストプロダクション・ワークフローと納品で求められる、再生速度およびオーディオ同期に合わせて設定します。



URSA Cine Immersiveの「FPS」インジケーターをタップして、フレームレート設定にアクセスします。URSA Cine Immersiveのプロジェクトフレームレート設定は、23.98、24、25、29.97、30、50、59.94、60、90fpsの8つに対応しています。イマーシブの撮影では、プロジェクトフレームレートは90fpsに設定してください。

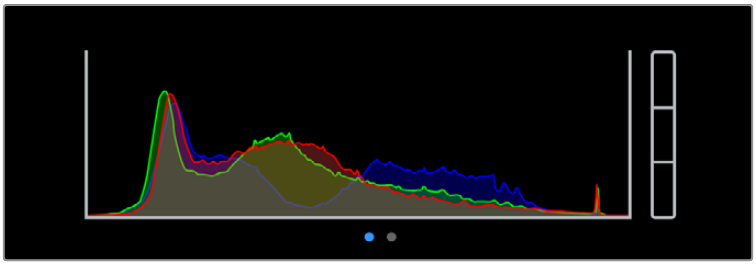
センサーフレームレートでは、イメージをセンサーがキャプチャーする速度を指定できます。デフォルトでは、URSA Cine Immersiveのプロジェクトフレームレートとセンサーフレームレートは、自然な再生速度で一致するようになっています。しかし、カメラの「FPS」メニューの右下にある「オフスピードフレームレート（OFF SPEED FRAME RATE）」スイッチをタップすると、センサーフレームレートを個別に設定できます。

「オフスピードフレームレート」スイッチが有効になっていると、センサーフレームレートを手動で設定できます。90fpsはこのカメラの最大フレームレートであり、Apple Vision Pro用に撮影している場合にプロジェクトフレームレートの90fpsと一致することを理解することが重要です。しかし、30や60fpsなどの低いフレームレートを選択し、90fpsのオフスピードで撮影することで、美しいスローモーションのフッテージを撮影できます。

「オフスピードフレームレート」スイッチがオンになっていると、左上の「FPS」インジケーターは、まずセンサーフレームレートを表示し、次にプロジェクトフレームレートを表示します。

ヒストグラム

URSA Cine Immersiveのタッチスクリーンの左下にはヒストグラムが表示されます。ヒストグラムは、イメージ内の明るさの分布を示すグラフです。左側はシャドウ、中央はミッドトーン、右側はハイライトを意味します。グラフの高さは、イメージのどの程度がその明るさレベルにあるかを示します。



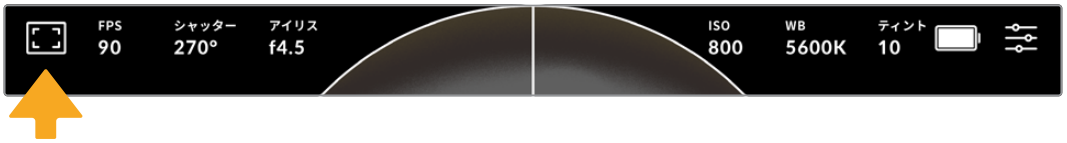
ヒストグラムでは、クリップのトーン分布を一目で確認できます。グラフが左側にあるイメージは露出不足である可能性があります。同様に、グラフが右側にあるイメージは露出過多である可能性があります。ヒストグラムは、常に映像のコンテクストを考慮に入れて読む必要があり、通常、適切な露出の映像は中央にバランスの取れたグラフが表示されます。露出を上げるとヒストグラムが右に動き、露出を下げると左に動きます。

ヒストグラムの右側には、各カラーチャンネルがクリッピングしているかどうかを確認できるインジケーターがあります。通常は、どのチャンネルにもクリッピングが生じないようにすることが一般的です。しかし、映像のミッドトーンやシャドウを維持するためにクリッピングが許容される場合もあります。

ETTR（Expose to the Right）は、カラーチャンネルをクリッピングさせない限界までヒストグラムの右側に露出を寄せて、意図的に露出過多の映像を撮影するテクニックです。これにより、信号が最大限になり、ノイズが最小限に抑えられます。ポストプロダクションで露出を下げることが可能であれば、これはクリーンな映像を撮影する優れた方法です。

LCDモニターオプション

URSA Cine ImmersiveのLCDタッチスクリーンの左上にあるモニターアイコンをタップすると、LCDモニター設定にアクセスできます。このメニューには、モニターの明るさなどの便利なオプションが含まれています。また、パワフルな露出ツールも搭載されています。



ゼブラ

タッチスクリーン上のゼブラ表示のオン/オフを切り替え、ゼブラレベルを設定できます。ゼブラは、イメージ内で、設定した露出レベルを越えた箇所に斜線を表示します。パーセンテージの設定は、映像内で露出過多の部分を表示するために使用します。露出を補正できる部分を特定できるように、この設定を100%ではなく、95%などの値にして使用することが一般的です。2つ目のゼブラ設定では、中間グレーのゼブラゾーンのオン/オフ、中間グレー +1ストップなどを設定できます。

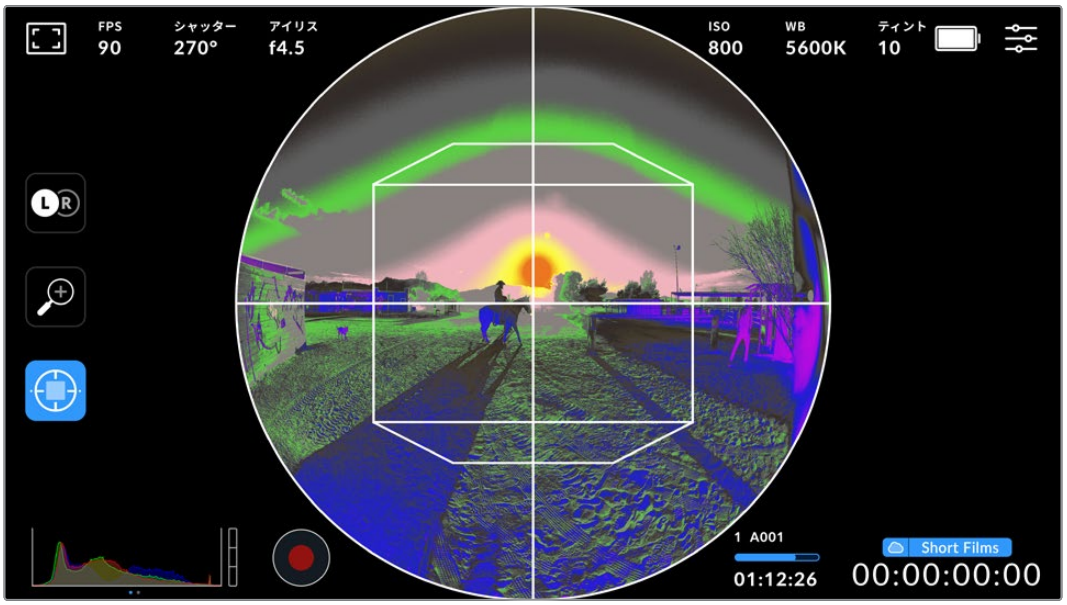
グリッド

LCDタッチスクリーン上の3x3グリッド、水平メーター、十字線、ドットの表示を切り替え、URSA Cine Immersiveの全出力で使用するオーバーレイを設定できます。イマーシブコンテンツの収録では、水平メーターは非常に役立ちます。このメーターでは、映像の中央に二重十字線が表示されます。これは、カメラが真っ直ぐで水平になっていると完全に重なって見えますが、カメラが傾くとずれて表示されます。

フォルスカラー

フォルスカラーは、イメージ内のエレメントに応じて、異なる露出値を様々なカラーでオーバーレイします。例えば、明るめのスキントーンに最適な露出はピンク、暗めのスキントーンはグリーンで表示されます。人物を撮影する際は、ピンク/グリーンのフォルスカラーをモニタリングすることで、スキントーンで一貫性のある露出を維持できます。

同様に、イメージ内のエレメントが黄色から赤に変わった場合は、露出過多になったことを意味します。ヒストグラムと同じく、このツールでは映像のトーン分布を極めてすばやく、直感的に理解できます。



収録

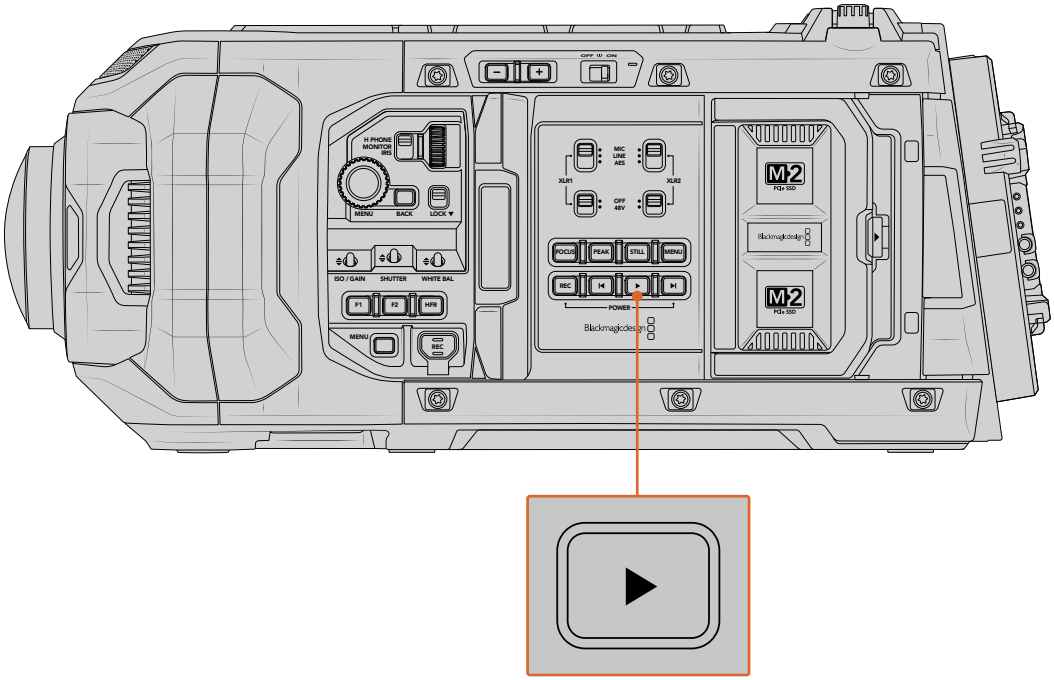
カメラの赤い録画ボタンのいずれかを押すと収録が開始されます。録画ボタンは、前方コントロールパネル、開閉式モニターの後ろにある内部コントロールパネル、アシスタント用ステーションのコントロールパネル、両方のLCDタッチスクリーンにあります。再度「録画（REC）」ボタンを押すと収録が停止します。

ストレージインジケーターでは、録画可能時間を確認できます。時間は、選択したフレームレートに応じて、「時間：分：秒」で表示されます。録画可能時間の上にあるメディアバーは、現在のステータスに応じて、青、白、赤でストレージの使用量を表示します。



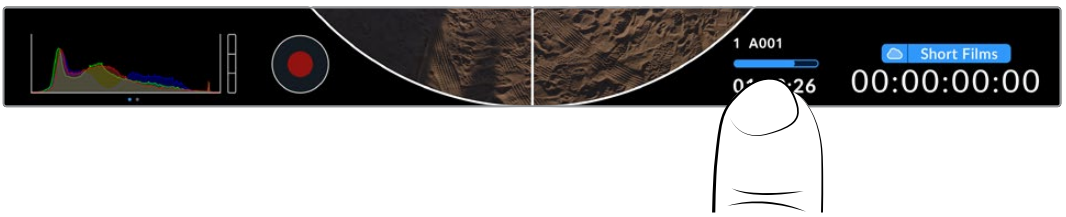
再生

撮影が終わったら、カメラのトランスポートコントロールボタンを使用すると、撮影したクリップを再生できます。最後に収録したクリップを再生するには、カメラ左側の物理的な再生ボタンを押します。このボタンには、開閉式タッチスクリーンを開くとアクセスできます。左/右三角ボタンは他のクリップにナビゲートするために使用できます。再生モードを出て、収録を続けたい場合は、もう一度録画ボタンを押します。

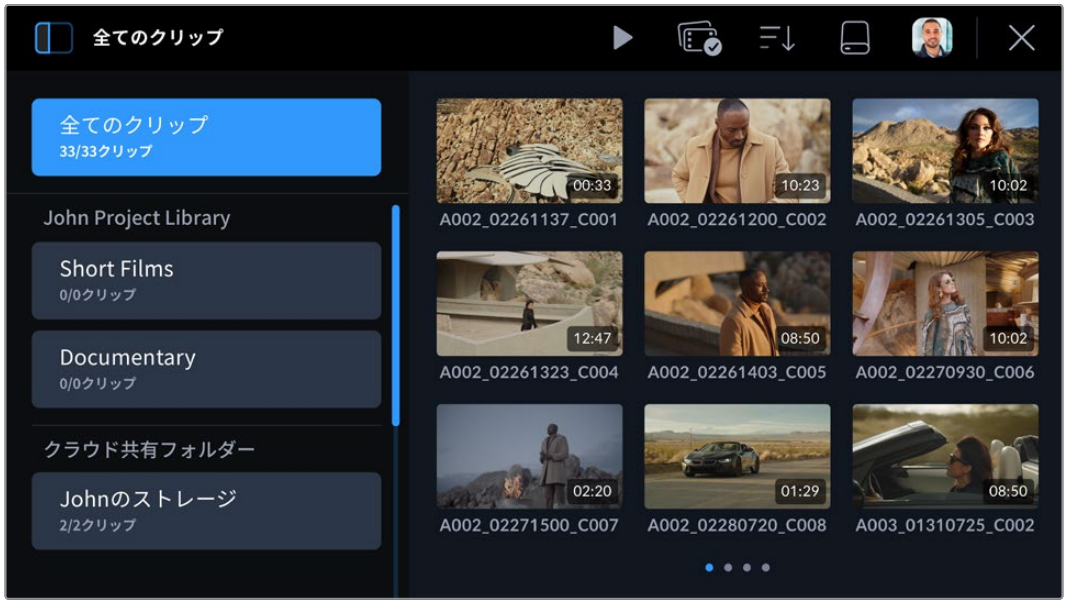


左/右三角ボタンを押すとクリップの最初/最後に飛びます。左三角ボタンを1回押すと、現在のクリップの先頭に戻り、2回押すと前のクリップの先頭に戻ります。左/右三角ボタンを長押しすると、2倍速で順方向/逆方向に再生します。倍速の再生や巻き戻し中に、左/右三角ボタンを2回押すと4倍速、3回押すと8倍速、4回押すと16倍速になります。

撮影したフッテージをより詳細に確認する必要がある場合、メディアプールを使用する方が良いでしょう。メディアプールを開くには、カメラのタッチスクリーン・ディスプレイ底面のストレージインジケーターのいずれかをタップします。



メインページのブラウザは、カメラに接続された全メディアに収録されているクリップすべてのサムネイルを表示します。サムネイルは3行ですが、それ以上の数のクリップが追加されると、タッチスクリーンをスワイプすることで次のページのサムネイルが表示されます。画面底部のページインジケーターでは、何ページのクリップがあるか確認できます。



URSA Cine ImmersiveをBlackmagic Cloudのアカウントに接続し、フッテージを直接DaVinci Resolveプロジェクトに送ります。

メディアプールから出るには、右上のXをタップするか、カメラの録画ボタンのいずれかを一回押します。

クリップの転送

カメラのストレージからクリップを転送する方法は多数あります。例えば、10Gイーサネットケーブルで、URSA Cine Immersiveをコンピューターやネットワークに接続し、イーサネットを介して、カメラのストレージからクリップをコピーできます。ネットワークやコンピューターに接続されたBlackmagic Media Dockに、Media Moduleをロードすることも可能です。Media Dockは、同時に3つのMedia Moduleを扱えます。

イーサネットでコンピューターに接続している場合、コンピューターはBlackmagic Cloud Storeを接続した際と同様に、Blackmagic Media Moduleを認識します。

カメラはデフォルトでDHCPを使用するように設定されているので、コンピューターまたはネットワークは自動的にカメラを見つけて、IPアドレスを割り当てます。

ファイル共有を有効にする

カメラを初めてセットアップする際に、イーサネットでURSA Cineをコンピューターまたはネットワークに接続している場合、SMBファイル共有はデフォルトで無効になっています。ネットワークで他のユーザーとファイルをすぐに共有するBlackmagic Cloud Storeとは異なり、重要な撮影の直後にカメラのマスターファイルに同様のアクセスを許可する必要はないでしょう。こういった理由から、SMBファイルの共有はユーザーが有効化できるようになっています。

1. ファイル共有を有効にするには、Blackmagic Camera Setupをダウンロードし、コンピューターにインストールする必要があります。Setup Utilityは管理ソフトウェアアプリケーションで、カメラのアップデートや設定の変更が実行できます。Blackmagic Camera Setupは以下よりダウンロードできます。
https://www.blackmagicdesign.com/jp/support/family/professional-cameras
2. Setup Utilityのダウンロードとインストールが完了したら、カメラをUSBでコンピューターに接続し、Blackmagic Camera Setupを起動します。カメラ後方のトップパネルにあるUSBポートに接続します。

3 設定を開くには、URSA Cine Immersiveのイメージまたは設定アイコンをクリックします。

4 「Network Access（ネットワークアクセス）」設定までスクロールし、「File Sharing (SMB)（SMBファイル共有）」を有効にします。設定の右下の「Save（保存）」を押すと、これらの設定が有効になります。

5 その下のURL設定に進み、URLアイコンをクリックすると、アドレスが自動的に入力されます。URLには、カメラの名前がローカルネットワークの場所として表示されます。

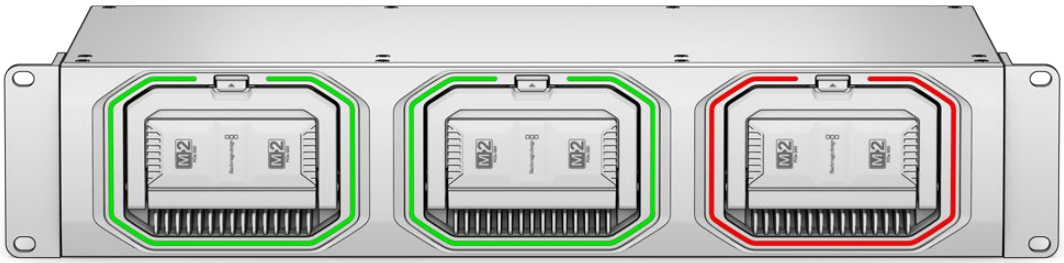
6 「Save（保存）」をクリックします。

クリップを表示するには、カメラのURLをクリックします。パスワードを入力するか、ゲストとして接続するかを確認するウィンドウが開きます。「Guest（ゲスト）」ボタンをクリックして、「Connect」をクリックします。

7 コンピューターは、MacではFinder、WindowsではWindows Explorerで、自動的にMedia Moduleまでナビゲートします。Setup Utilityはイーサネットセットアップを記憶するので、この作業は最初にセットアップする際に一度だけ行えば、次回はカメラをイーサネットで接続すると、設定を変更する必要なく、Media Moduleに即座にアクセスできます。

Blackmagic Media Dockの使用

Blackmagic Media Moduleは、外付けストレージにクリップを転送する必要がある場合や、収録を継続するために空のモジュールに交換する際に、すばやくロードおよびアンロードができるように作られています。Blackmagic Media Dockを使用すると、カメラを使用せずにBlackmagic Media Dockのフッテージにアクセスできます。



概要

Apple Immersive Videoは、視覚的に物語を伝える新しいアプローチで、これまでの2D映画に代わる選択肢を提供します。これは代替品として意図されているわけでなく、このフォーマットを追加することで、プロジェクトのニーズに合わせた選択肢を増やすことを目的として開発されました。どのフォーマットにも言えることですが、クリエイティブな選択は、物語の内容によって変わります。

多くの場合、新しいフォーマットは複雑なワークフローが必要になりますが、URSA Cine Immersiveは、Apple Immersive Videoの撮影プロセスをシンプルするように設計されています。イマーシブコンテンツの撮影を分かりやすい方法で実行でき、VFXなど他の用途にも使用できます。

このガイドは、URSA Cine Immersiveの中心となる機能とワークフローの概要を説明するもので、開始点を意図しています。どのカメラでも言えることですが、制作のニーズに合わせて、最高の結果を得るために、色々と試し、さらなるテストを行うことをお勧めします。

DaVinci Resolve Studio



DaVinci Resolve Studioは包括的なポストプロダクション・ソフトウェアであり、編集、カラーコレクション、VFX、モーショングラフィックス、オーディオ・ポストプロダクションをシームレスに統合します。また、超高解像度ステレオスコピックビデオ、90fpsのフレームレート、空間オーディオミキシングなど、Blackmagic RAW Immersiveフォーマット独自の特徴に合わせたワークフローに対応しています。DaVinci Resolve Studioは、Apple Immersive Videoの撮影、モニタリング、書き出しに完全に対応しているため、イマーシブコンテンツの作成においてパワフルなツールとなっています。

ワークステーションの推奨事項

ハードウェアの推奨事項

- Mac Studio M3 Ultraデスクトップ
- 高リフレッシュレートモニター（最低60Hz）

最小システム要件

- macOS 15以降
- 64GBのシステムメモリ
- モニタリングには、Blackmagic Design Desktop Video 12.9以降
- Appleシリコンベースのコンピューターまたは Metalに対応したGPU

イマーシブ・モニタリング要件

- Apple Vision Pro
- 高速ワイヤレスネットワーク（Wi-Fi 5以上、Wi-Fi 6を推奨）

プロジェクトのセットアップ

Blackmagic URSA Cine Immersiveの加工されていないメディアを取り込む前に、DaVinci Resolveのプロジェクト設定を開いて、メディアが適切にディベイヤーおよび処理されるように設定し、視覚的な出力を最適化して、編集を行いやすくすることをお勧めします。

マスター設定

プロジェクトのセットアップを行う際に、Apple Vision Proのコンテンツ作成用の設定にすることが重要です。これにより、プロジェクトの準備、構築、書き出しを行う際に、適切な設定とワークフローのオプションがすべて表示されます。

1. DaVinci Resolve Studioを起動します。
2. 「プロジェクトマネージャー」ウィンドウの右下で「新規プロジェクト」をクリックします。
3. プロジェクト名を入力し、必要に応じて「メディアの保存先」を設定して「作成」をクリックします。
4. DaVinci Resolve Studioの右下のギアアイコンをクリックして「プロジェクト設定」を開きます。

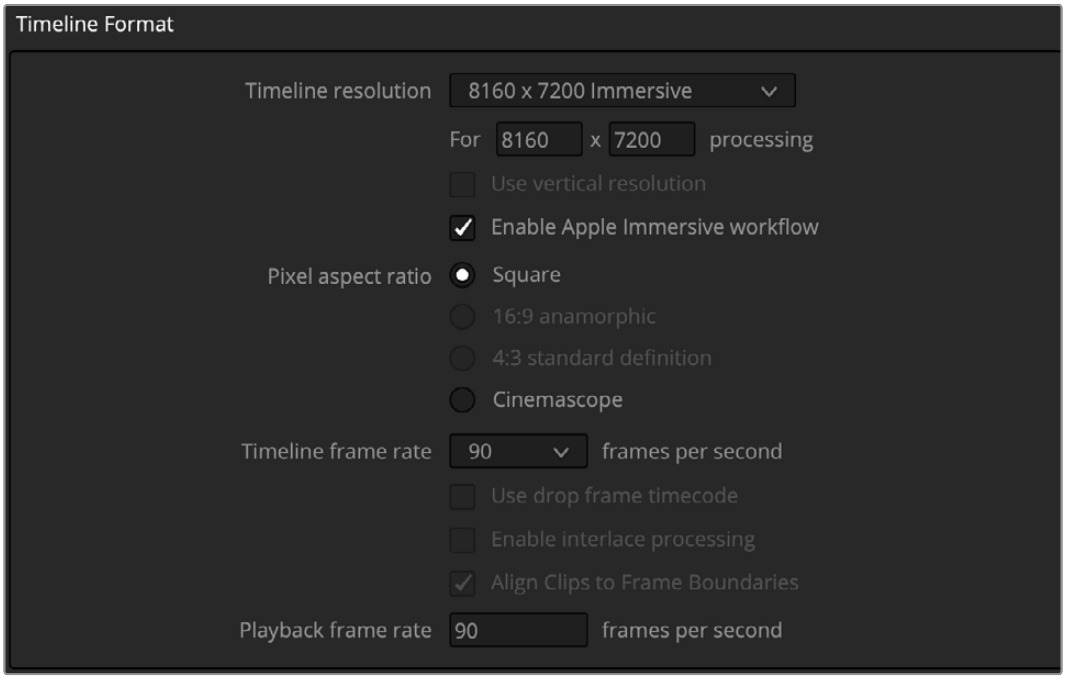
   まず、プロジェクトの解像度とフレームレートを設定します。
5. サイドバーの「マスター設定」を選択します。
6. 「タイムラインフォーマット」セクションで、「Apple Immersiveワークフローを有効化」を選択します。



   これにより、DaVinci Resolveの複数の設定がイマーシブワークフローに適した設定になります。イマーシブプロジェクトを作成する際は、必ずこの設定を最初に選択してください。
7. 「タイムライン解像度」を「8160 x 7200 イマーシブ」に変更して、センサーのネイティブ解像度でクリップを処理します。

   処理パワーが限られている場合、「4080 x 3600 イマーシブ」を選択すると、半分の解像度で再生がスムースになります。
8. 「タイムラインフレームレート」は90fpsに設定します。このパラメーターは、Apple Immersiveワークフローを有効化すると自動的に調整されます。
9. イマーシブビデオ・コンテンツを最も正確に再現できるように「再生フレームレート」を90fpsに設定します。

   処理能力が限られている（あるいは、リフレッシュレートの低いモニターを使用している）場合、フレームレートを下げると再生が安定化します。フレームレートを選択する際は、ネイティブフレームレートを割り切れる数、例えば30fpsや60fpsに設定することを推奨します。



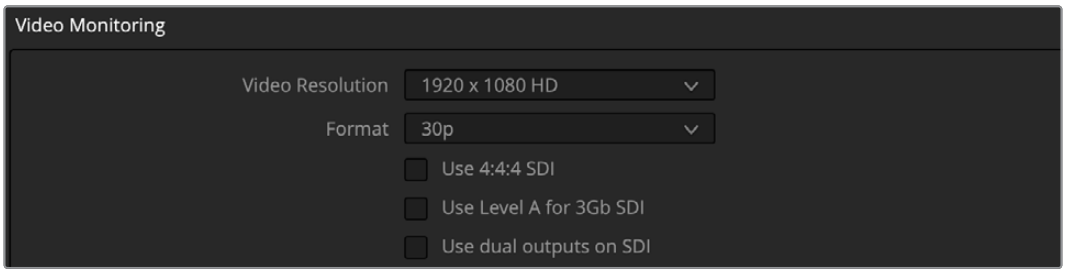
ビデオモニタリング

DeckLinkやUltraStudioなどのキャプチャー・再生デバイスを介して、外部モニターを使用している場合は、モニターの仕様に基づいてビデオモニタリング設定を行います。

1 「ビデオモニタリング」セクションで、モニターの解像度と一致する「ビデオ解像度」を設定します。例えば、1920 x 1080 HDや3840 x 2160 Ultra HDなどです。

イマーシブビデオのネイティブフレームレートは90fpsですが、これは多くの放送モニターが対応しているフレームレートではありません。

2 「フォーマット」をモニターのフレームレートと一致する設定にします。例えば、30pなどです。



イメージスケーリング

「イメージスケーリング」設定は、イマーシブメディアの解像度とアスペクトレシオがビューアに適切に表示されるようにするための設定です。これらのパラメーターは、「マスター設定」でApple Immersiveワークフローを有効にすると自動的に設定されますが、適切かどうか自分でチェックすることが重要です。

1 「プロジェクト設定」サイドバーで「イメージスケーリング」を選択します。

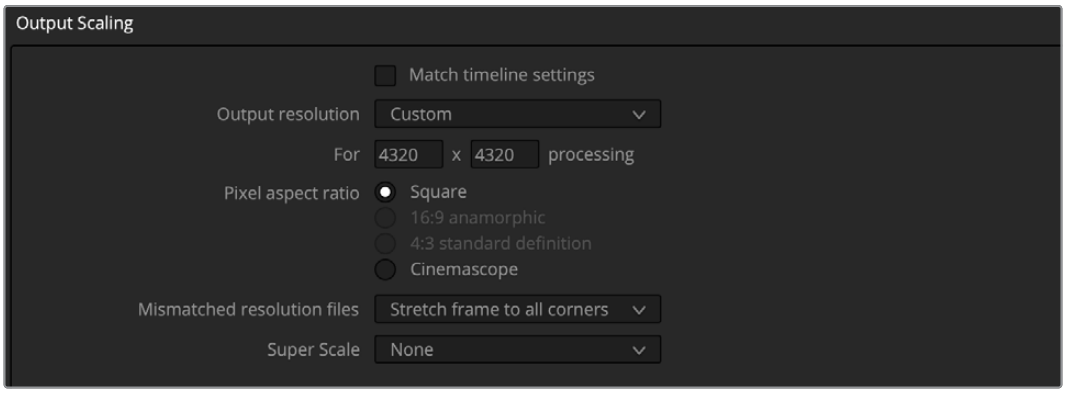
2 「出力スケーリング」セクションで「タイムライン設定と一致」の選択が解除されているか確認します。

3 「出力解像度」が「カスタム」で**4320 x 4320**に設定されているかチェックします。

小さなディスプレイや円滑な再生のために、半分の解像度（**2160 x 2160**）に設定することも可能です。

これらの正方形のアスペクトレシオの解像度は、DaVinci Resolve StudioでApple Immersive Videoのコンテンツを再生し、（WiFiまたはDeveloper Strapで）Apple Vision Proに配信するために必要です。

4 「解像度が一致しないファイル」が「全辺をマッチ: 縮小/拡大」になっていることを確認します。



カラーマネージメント

最大出力輝度（nit）でカスタムコントロールを行う必要があるため、カラーマネージメントには、カラーページでカラースペース変換（CST）ノードを使用することをお勧めします。プロジェクト設定におけるカラーマネージメント設定の目的は、カラーページのタイムライン（作業用）カラースペースを定義し、レンダリングしたファイルに関連付けられるカラースペースメタデータを設定することです。

1 「プロジェクト設定」サイドバーで「カラーマネージメント」を選択します。

2 「カラーサイエンス」はDaVinci YRGBのままにします。

3 「タイムラインカラースペース」をDaVinciWG/Intermediateに設定します。

これは、将来が保証された広色域カラースペースでHDRワークフローに対応しています。したがって、Blackmagic URSA Cine Immersiveメディアおよび Apple Vision Proの納品ファイルに最適な設定です。

4 「出力カラースペース」を「P3-D65 ST2084 1000 nits」に設定します。

このカラースペースは、Mac Studioデスクトップで DaVinci Resolveのビューアをモニタリングするために適した色域に対応しています。これは、多くの場合「Display P3」です。グレーディングモニターやAppleの異なるICCプロファイルを使用している場合は、出力カラースペースは、ディスプレイのカラー規格に一致する設定に変更します。

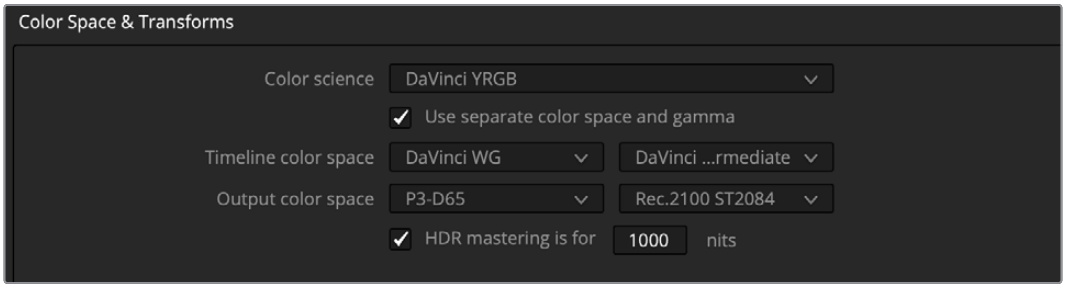
しかし、ガンマ（伝達関数）を変更して、ICCプロファイルを適切にマッピングする必要があります。

5 カラースペース設定の上にある「別々のカラースペースとガンマを使用」をクリックします。

これにより、カラースペースとガンマのドロップダウンフィールドが分割して表示されます。

6 右の出力ガンマは「Rec.2100 ST2084」に設定します。

7 「HDMIメタデータの最大輝度」を1000nitsに設定します。



次回Apple Immersiveのプロジェクトを設定する時間を削減するために、現在の設定をプリセットとして保存できます。

8 「プロジェクト設定」ウィンドウの右上にある3つのドットのオプションで、「現在の設定をプリセットとして保存」を選択します。

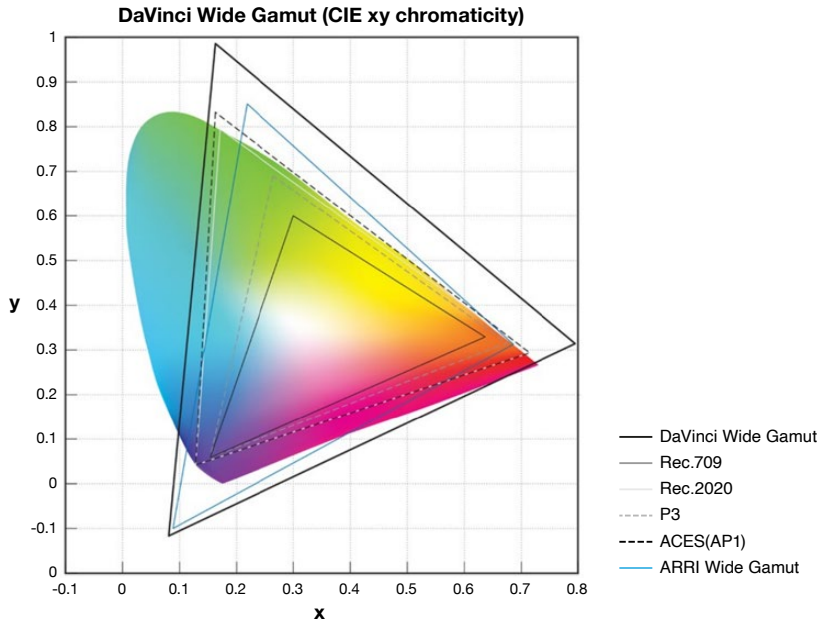
9 プリセットに名前を付け、「OK」をクリックします。

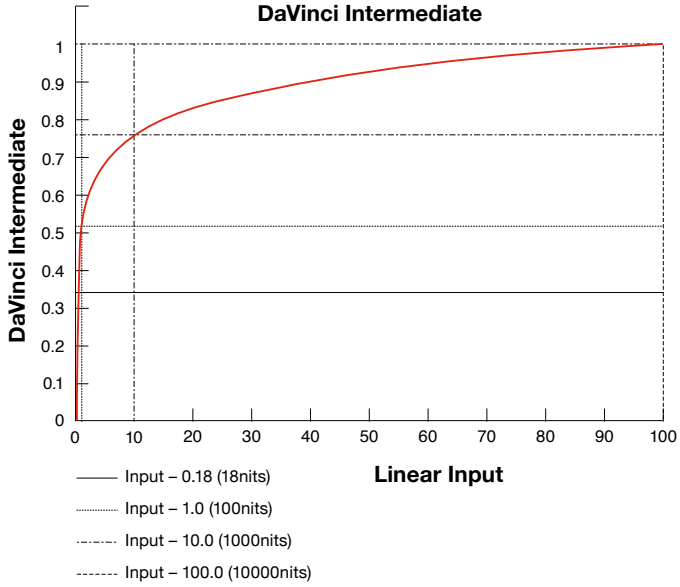
次回イマーシブメディアで作業する際に、プロジェクト設定をすばやく設定するには、「プロジェクト設定」ウィンドウを開き、オプションメンヒューをクリックして、表示されているプリセットから任意のものを選択します。

10 「保存」をクリックして、「プロジェクト設定」を閉じます。

DaVinci Wide Gamutについて

DaVinci Wide Gamutは、DaVinci Resolveのために開発されたBlackmagic Designの高度な内部カラースペースで、カラーグレーディングおよびイメージ処理において、極めて広く正確な環境を提供しています。DaVinci Intermediateガンマと組み合わせることで、異なるカメラフォーマットの混ざったプロジェクトを扱う上で、カラーマネージメントを簡素化する統一された作業スペースとなり、入力から出力において、一貫したトーンと彩度のマッピングが行われます。この広色域はHDRおよびイマーシブメディアでの作業に適しており、カラリストが忠実度の高い書き出しフォーマットでのマスタリングで、バンディングやトーンのクリッピングのリスクが生じることなく、ニュアンスのある調整を行えます。





Apple Immersiveの処理

「プロジェクト設定」サイドバーの「Apple Immersive」セクションでは、カスタムのILPD（Immersive Lens Profile Data）に設定することで、イマーシブコンテンツがどのようにキャリブレーションされ、投影されるかを調整できます。

1. 「プロジェクト設定」サイドバーで「Apple Immersive」を選択します。
2. 「キャリブレーションフォルダーを開く」をクリックします。
3. カスタムILPDファイルを追加またはアップデートします。
4. 「リストの更新」をクリックして、メディアプールでILPDコンテンツにアクセスできるようにします。
5. 「保存」をクリックして、「プロジェクト設定」を閉じます。
6. メディアプールでイマーシブクリップを右クリックします。
7. コンテクストメニューで「Apple Immersiveキャリブレーション」に進み、ソースILPD（No Apple Immersive Calibration File Selected [Apple Immersiveキャリブレーションファイルが選択されていません]）、カスタムILPD、ギャップILPDのいずれかを選択します。

デフォルトでは、カスタムILPDキャリブレーションフォルダーは、ProIMフォルダーにデフォルトのLUTフォルダーと共に存在します。

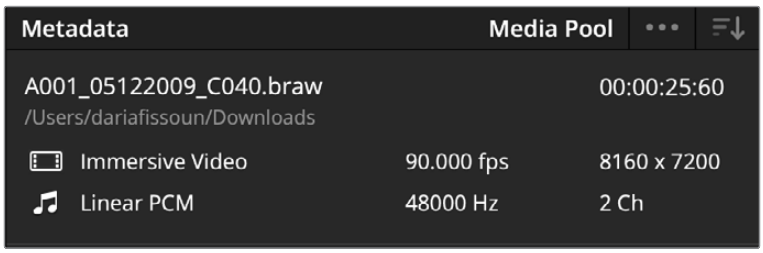
インジェスト

加工されていないイマーシブコンテンツをインジェストする際は、両目と関連するイマーシブメタデータはすべて、読み込まれた.brawクリップに付随します。

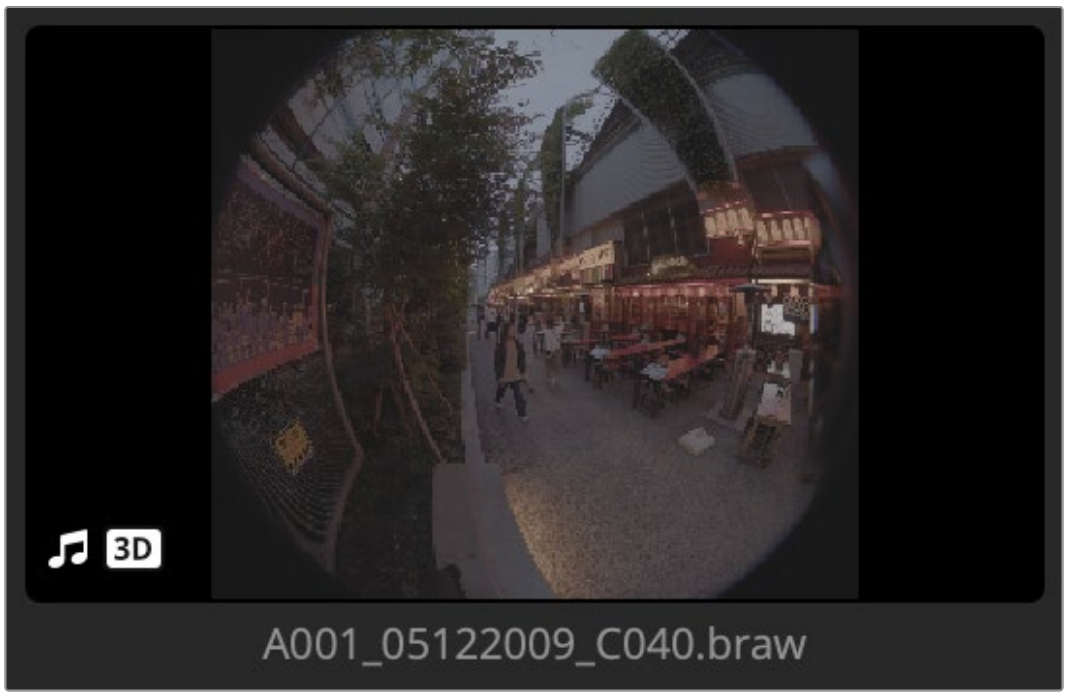
1. メディアページに進みます。
2. 「メディアストレージ」パネルで、イマーシブビデオ・コンテンツのある場所まで進みます。
3. メディアを下の「メディアプール」にドラッグ＆ドロップします。

作業のこつ：元のフォルダー構造を維持するには、「メディアストレージ」パネルから「メディアプール」サイドバーにドラッグします。「マスター」ビン（または作成したあらゆるビン）にドラッグする際に、フォルダーとサブフォルダーは維持されます。

Apple Immersiveビデオクリップが選択されると、「メタデータ」パネルのヘッダーでコーデックパラメーターにイマーシブビデオであることが表示され、ネイティブのフレームレートと解像度も合わせて表示されます。



さらに、ビデオサムネイルの左下にある3Dアイコンでも3Dクリップであることが確認できます。



メモ：初めてメディアプールにイマーシブメディアをドラッグ＆ドロップする際は、「設定を維持」するか「設定を変更」するかを選択するダイアログウィンドウが表示されることがあります。「設定を変更」を選択することで、「プロジェクト設定」がソースメディアのフレームレートと解像度を自動的に適用します。

編集

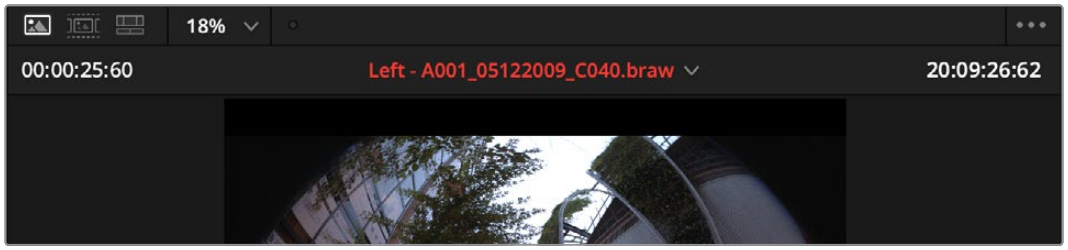
イマーシブコンテンツの作業においては、編集ツールとその挙動は従来の2Dビデオフォーマットで作業をする際とほとんどの場合で変わりません。以下の手順では、イマーシブビデオ・ワークフロー独自の編集機能と挙動について説明します。

タイムラインの構築

タイムラインの構築の際にクリップをプレビューする際、ソースビューアでどちらの目（左目か右目）を表示するか変更できます。それ以外の点では、タイムラインでのカット割り作業は標準的なNLEワークフローと変わりません。

1 エディットページに進みます。

2 任意のクリップをダブルクリックして、スクリーン左のソースビューアにロードします。ビューア上部でどちらの目が表示されているかを確認できます。



3 ビューアで表示する目を変更するには、メディアプールでソースクリップを右クリックして、「ステレオスコピック3D」＞「ステレオスコピック3Dモード」＞「左目」、「右目」、「両目」のいずれかを選択します。

> **作業のこつ**：ソースビューアのオプションで「ライブメディアプレビュー」を有効にしている場合、メディアプールでサムネイルにマウスを合わせると、ソースビューアでクリップをスクラブできます。

4 「I」と「O」のショートカットを使用すると、ソースビューアで範囲を指定できます。

5 ツールバーで「クリップを挿入」を押すか、F9ショートカットを使用すると、クリップをタイムラインに挿入できます。

> **作業のこつ**：メディアプールからもクリップをタイムラインパネルにドラッグ＆ドロップできます。

6 選択モードまたはトリム編集モードでは、マウスでタイムラインのクリップをトリムできます。

バックドロップ

「プロジェクト設定」で「Apple Immersiveワークフローを有効化」を選択すると、バックドロップ（BD）トラックが自動的に "ビデオ 1" トラックに表示されます。

バックドロップとは、通常180〜210°の視野であるイマーシブコンテンツの境界を超えて見える領域です。

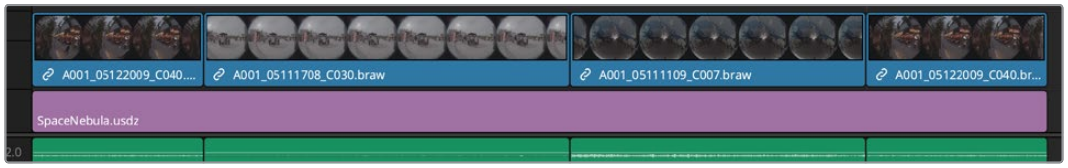
デフォルトでは、この領域はブラックになっていますが、バックドロップトラックを使用して、カスタマイズした背景画像（.usdzフォーマット）を挿入することも可能です。

1 バックドロップ用の.usdzファイルをバックドロップトラックにドラッグしてタイムラインに追加します。

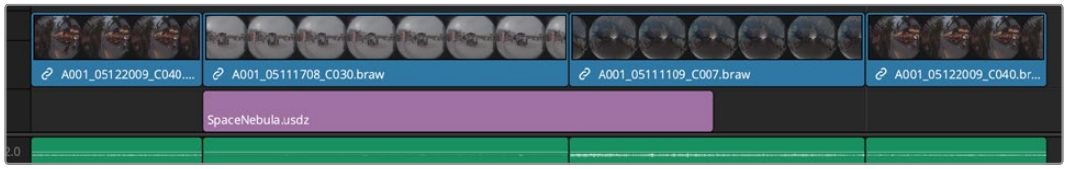
バックドロップ画像の長さは、ビデオトラックのクリップの長さと一致している必要があります。1つのバックドロップファイルは、1つのクリップ、複数のクリップ、タイムライン全体をカバーできますが、イマーシブの.brawクリップの間に入れることはできません。これは、Lカットと呼ばれます。



1つのクリップ – 適切 ✓

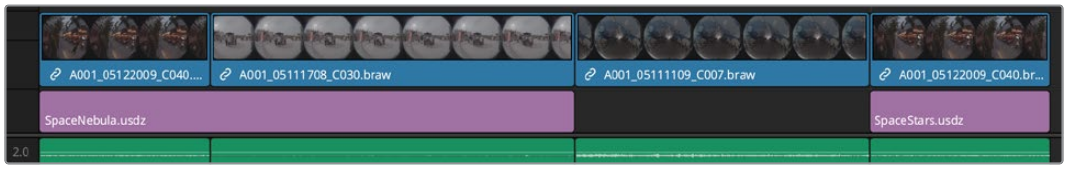


複数のクリップ – 適切 ✓



Lカット – 不適切 ✗

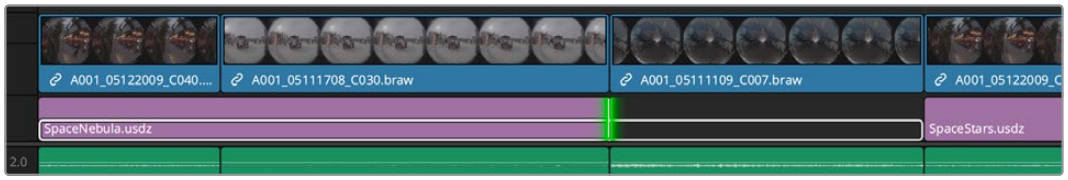
1つのタイムラインで複数のバックドロップファイルを使用したり、バックドロップに画像を追加せずにデフォルトのブラックを使用することも可能です。



異なる画像を使用し、間にギャップあり – 適切 ✓

2 標準的なトリムコントロールで、バックドロップファイルの長さを調整します。

選択モードまたはトリム編集モードで.usdzファイルの端をドラッグすると、長さを調整できます。



コンマ (,) およびピリオド (.) を使用すると、トリムを1フレームずつナッジできます。

作業のこつ：バックドロップがイマーシブビデオの編集と常に揃っていることを確実にするために、エディットページのツールバーで「スナップ」モードを有効にします。

ビューファインダーの使用

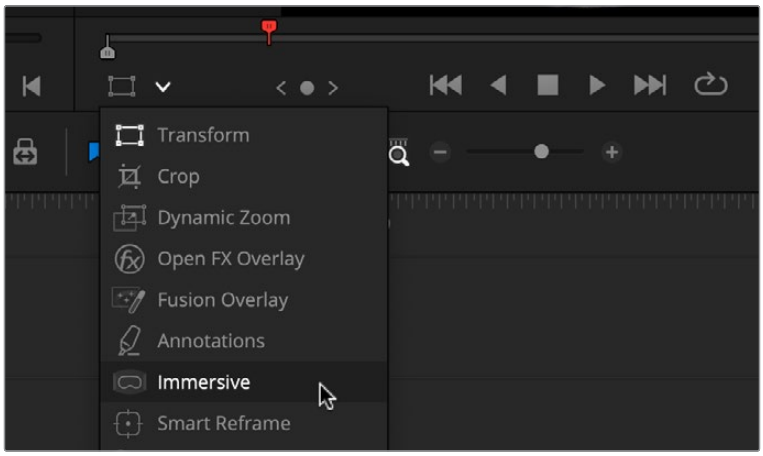
デフォルトでは、イマーシブクリップはDaVinci Resolveのビューアで円形のレンズ空間に表示されます。イメージを投影するために使用される視覚的なレンズキャリブレーションは、カメラを元にしており、クリップごとに異なります。



また、表示モードを「緯度/経度」または「ビューポート」に切り替えるオプションも搭載されています。

1 タイムラインビューアの左下で、オンスクリーンコントロール・ボタンの隣にあるドロップダウンメニューの矢印をクリックします。

2 「イマーシブ」を選択します。



ビューア上部に、追加の表示オプションが搭載された新しいツールバーが表示されます。

3 「緯度/経度」をクリックすると、映像がエクイレクタングラー形式（正距円筒図法）で表示されます。このモードは、イメージをワープさせて従来の2Dメディアの比率に近づけて表示するため、トラッキングや合成などの特定のポストプロダクションにおいて作業がしやすくなります。



4 「ビューポート」をクリックすると、正方形のフレームにイマーシブのシーンの中央部が切り抜かれた形で表示されます。このモードでは、標準のモニターでシーンのディテールを適切な比率ではっきりと確認する上で役立ちますが、フレーム外のアクションを逃さないように注意が必要です。

上部のイマーシブツールバーにある変形コントロールを使用して、フレーム内でステレオスコピックイメージをティルトおよびパンしたり、縦または横の視野を調整できます。



トランジション

2つのイマーシブビデオ・クリップをトランジションするには、ディレクトカットを実行するか、クリップの両側にあるハンドルを用いたフェードインまたはフェードアウトを使用するか、「カラーディップ」エフェクトを使用します。トランジションはフッテージに焼き付けず、再生中にApple Vision Proでレンダリングされるべきです。

1 「エフェクト」パネルを開きます。

2 サイドバーで「ツールボックス」>「ビデオトランジション」に進みます。

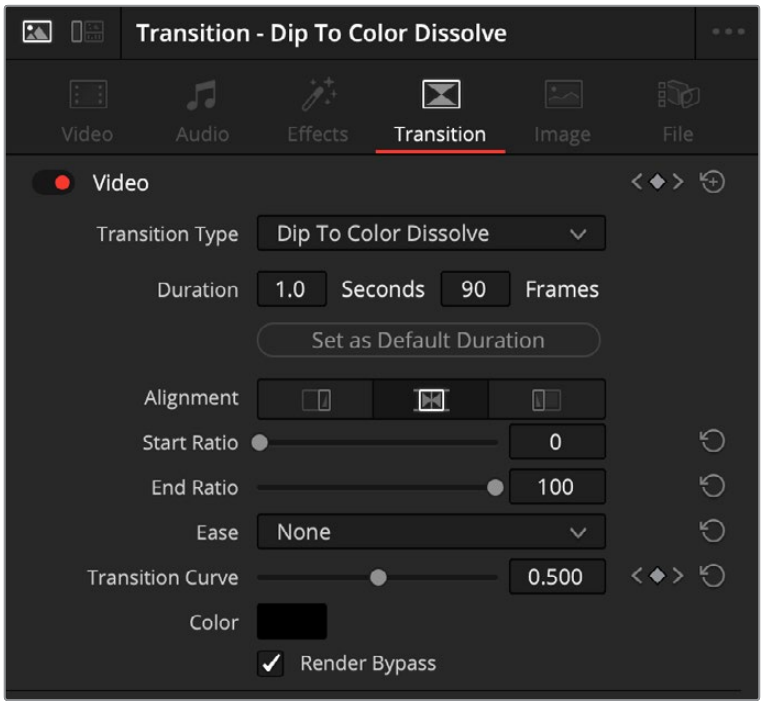
3 「カラーディップ」を2つのクリップの間にある編集点にドラッグします。



メモ：トランジションを適用するには、両方のクリップの編集点を超えて十分なトリムデータが存在する必要があります。十分なトリムデータ（ハンドル）が無い場合、トランジションは適用されません。

4 「インスペクタ」パネルで「トランジション」タブに進み、「レンダーをバイパス」を選択します。これにより、Apple Vision Proによりディゾルブが適切にレンダリングされます。

5 その後、「インスペクタ」の「カラーディップ」で残りのパラメーターを使用して、トランジションの長さ、配置、アニメーション速度、カラーを調整できます。



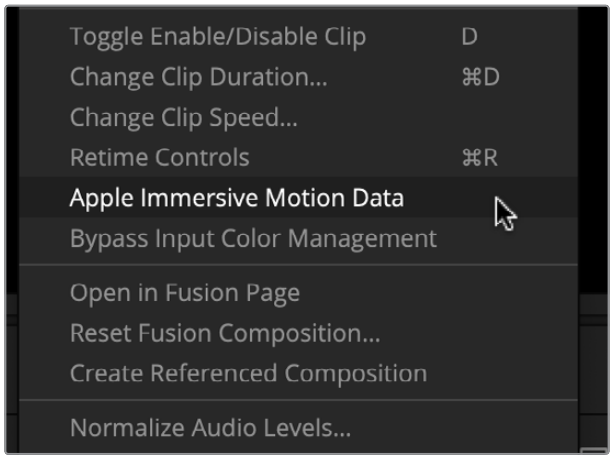
Apple Immersive Motion Data

イマーシブコンテンツの作業では、突然またはすばやいカメラの動きは、視聴者が方向感覚を失う原因になりかねません。視聴者の快適さを維持するために、すばやいパン/ティルト、回転、平坦でない地平線などが含まれるテイクの使用は避けます。

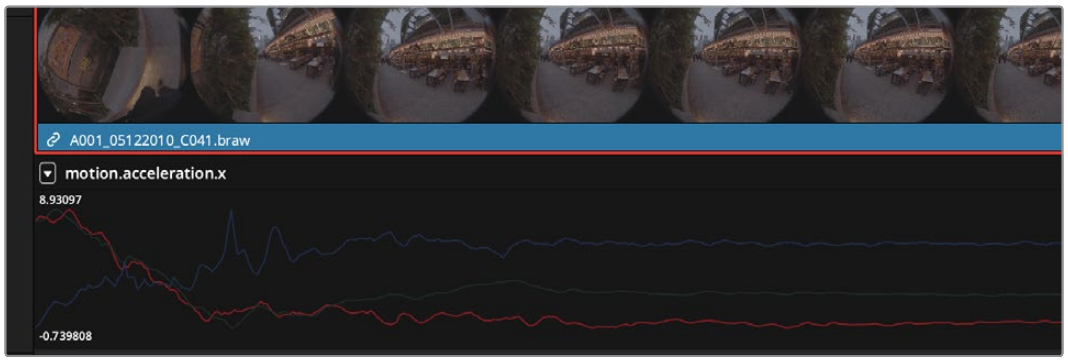
Apple Immersive Motion Dataは、加速度およびジャイロスコープの位置値をX軸、Y軸、Z軸で表示するグラフで、カメラの動きが強すぎるかどうかを客観的に証明します。エディターはこの情報を使用して、フッテージの方向感覚が失われるセクションの使用を避けることができます。

1 タイムラインでイマーシブクリップを右クリックします。

2 「Apple Immersive Motion Data」を選択します。

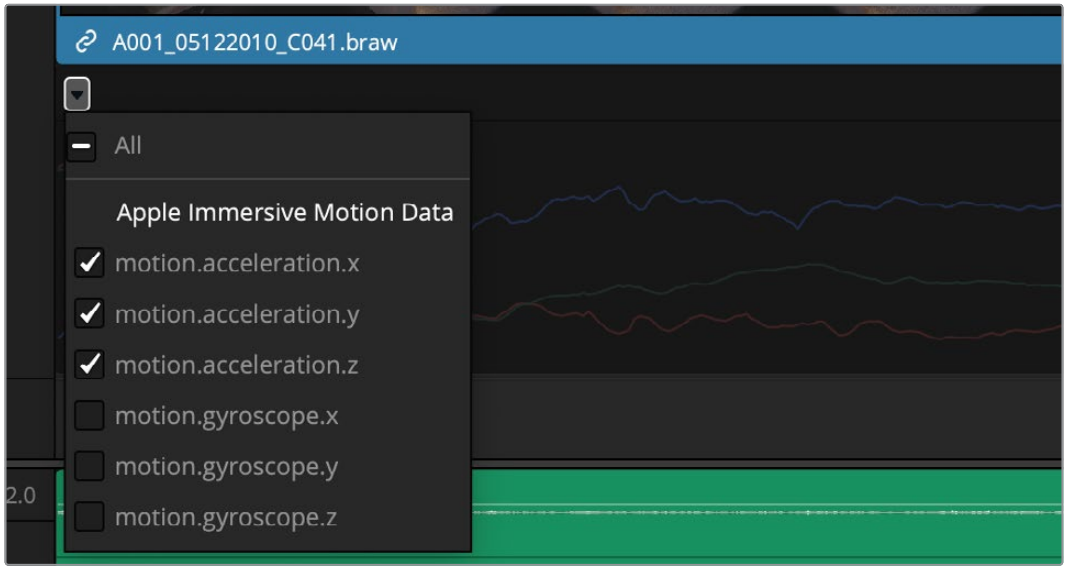


ビデオトラックにグラフが表示され、X軸、Y軸、Z軸の加速度が示されます。



この例では、カメラオペレーターがセットアップ中にカメラで収録が始まっていました。グラフの急激な山は、カメラが持ち上げられ、回転し、肩に乗せられた時の突然の動きを示しています。理想的には、カメラが安定し、グラフの値が平坦化するビデオ後半を使用することが勧められます。

3 グラフ左上の矢印をクリックします。



各軸の値は、ショットの内容やカメラの動きの性質に基づいて、有効または無効にできます。

タイトル

エディットページでは、「エフェクト」パネルのあらゆるタイトルツール（テキスト、テキスト+、マルチテキスト、スクロール）およびローワーサードを使用して、イマーシブシーンにテキストを追加できます。

イマーシブクリップの上にタイトルジェネレーターが配置された場合、クリップのILPD（Immersive Lens Profile Data）が使用され、シーンのレンズワープと一致するようにテキストが適切に変形されます。

ILPDが使用できない（例えば、ブラックのバックドロップ上やイマーシブCGシーンにテキストを追加した）場合、Apple Vision Proで視聴した際に適切にテキストがワープされるように、デフォルトのレンズのゆがみが適用されます。

1 「エフェクト」パネルで「ツールボックス」>「タイトル」に進みます。

2 「テキスト」ジェネレーターをタイムラインにドラッグします。イマーシブクリップの上のビデオトラックにドロップし、シーンに統合します。



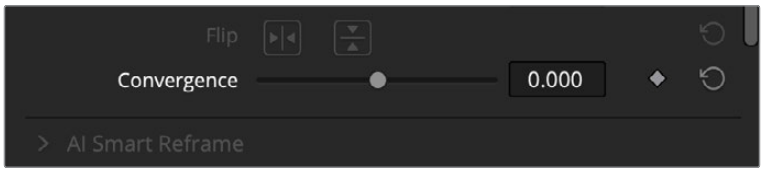
バックドロップと同様に「テキスト」ジェネレーターの途中に配置できません。各クリップが個別のレンズ空間でレンダリングされるため、重なっているタイトルはショット間でワープされるリスクがあります。

3 タイトルをビデオの編集点に合わせてトリムします。

4 「インスペクタ」でタイトルテキストを入力し、書式設定します。

5 「設定」タブで「コンバージェンス」スライダーを使用して、テキストを3D空間で任意の位置に配置します。

スライダーを左に動かすと空間でタイトルが奥に移動し、右に動かすと手前に移動します。



コンバージェンスの効果は、Apple Vision Proで確認するのが最も良いでしょう。他の方法でモニタリングする場合は、ビューアでテキストの奥行きの見え方を変更できます。これは、カラーページで、下のビデオレイヤーの「ビジョン」モードの調整で変更できます。

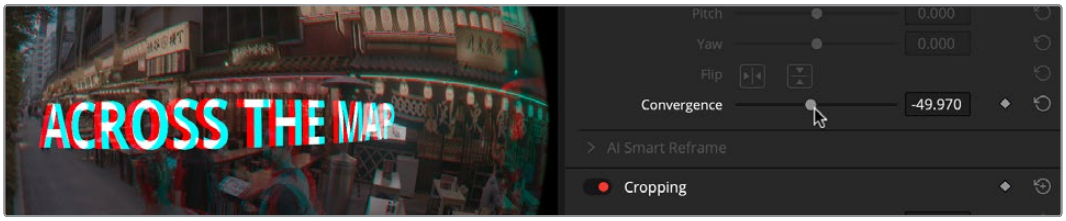
6 カラーページに進みます。

7 「3D」パレットを開きます。

パレット中央の「ビジョン」では、イマーシブメディアを投影するために使用されるステレオスコピックのフォーマットを変更できます。

8 「ステレオ」タブを選択します。

9 「アウト」モードを「アナグリフ（カラー）」に変更します。



エディットページの「インスペクタ」パネルで「テキスト」ジェネレーターの「コンバージェンス」パラメーターを変更すると、アナグリフ（またはアナクローム）メガネで見た際に、より直感的な形で表示されます。

創造面における考慮事項

Blackmagic URSA Cine ImmersiveカメラとApple Vision Proでの制作における初期段階である現在、映像作家はこのフォーマットで色々と試して、最善の方法を割り出し、新しいルールを確立し、これまでの映像制作では考えられなかったようなワークフローを見つけることが勧められます。

それを踏まえて、イマーシブ制作のいくつかの側面は、視聴者の没入感と視覚的な快適さを最大限にするために特別な注意が必要な事項として既に識別されています。

1 編集は、プリプロダクションの段階で始めるべきです。ショットの長さと視聴者の快適さにおいて特別な考慮を払う必要があるため、最終的な編集を念頭に置いてシーンを撮影する必要があります。

2 急速なカメラの動きのあるクリップは避けるか、最小限に抑えます。こういったショットでは、視聴者が方向感覚を失う可能性があるからです。代わりに、固定のショットやゆっくりとしたパンやドリーを使用します。

3 バランスの悪いカメラアングルや極端なアングルも方向感覚を失う可能性があるので、被写体に対して水平で、快適な距離にある（近すぎない）テイクを使用します。

4 イマーシブメディアでは、新しい環境を理解して馴染むまでに、2Dシーンを視聴する際より時間がかかります。カット割りはゆっくりめのペースで、新しい環境に視聴者が慣れるまでの時間を十分に持たせます。

5 クリップ間でカットする場合、視聴者が前のショットでどこを見ていたかを注意して行います。視聴者がフレームの右側を注意して見ていた場合、重要なアクションが左側で起きることは避けるべきです。

6 空間オーディオはストーリーテリングの多くを担います。イマーシブの制作では、空間オーディオを登場人物の一人として扱うくらいの重要性があります。

VFX合成

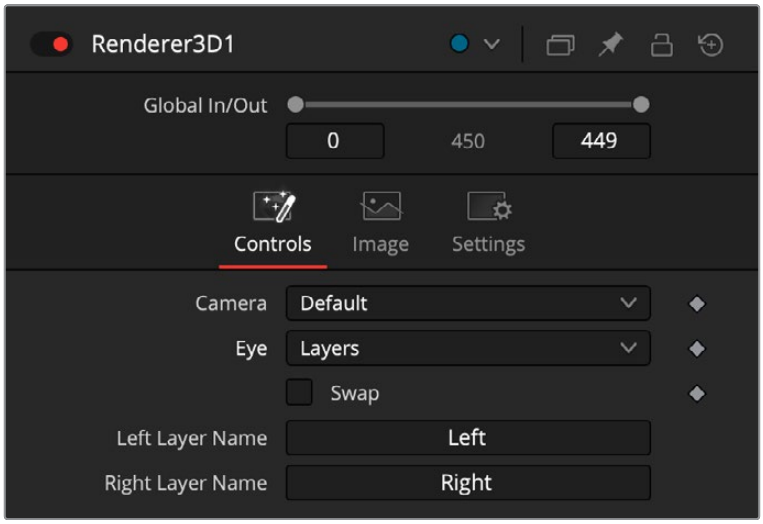
「プロジェクト設定」で「Apple Immersiveワークフローを有効化」を選択すると、専用のFusionパラメーターにより、シームレスなステレオスコピック・ワークフローとイマーシブに適した合成が行えるようになります。「メディア出力（MediaOut）」ノードには左右の目の出力オプションがあり、「イマーシブパッチャー」ツールではエクイレクタングラーの投影と合成が実行できます。

メディア出力（MediaOut）ノード

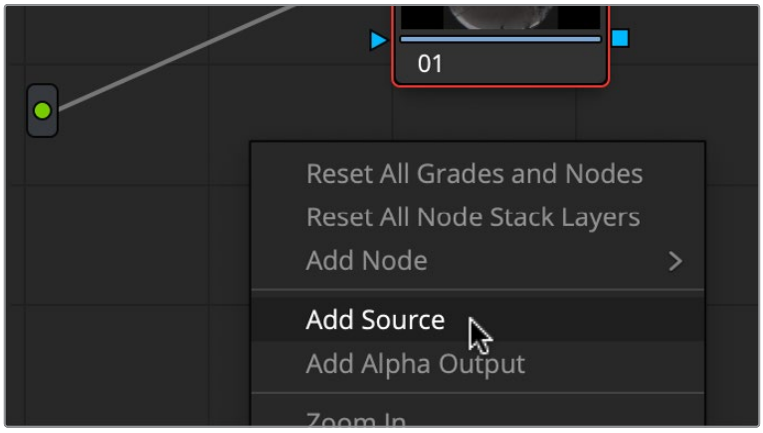
イマーシブクリップで作業をする際にFusionページに初めて行くと、ノードグラフには2つの「メディア入力（MediaIn）」ノードが2つの「メディア出力（MediaOut）」ノードに接続されています。



「メディア出力（MediaOut）」の「インスペクタ」で「ステレオの目」パラメーターを使用すると、目の出力信号のルーティングを行えます。



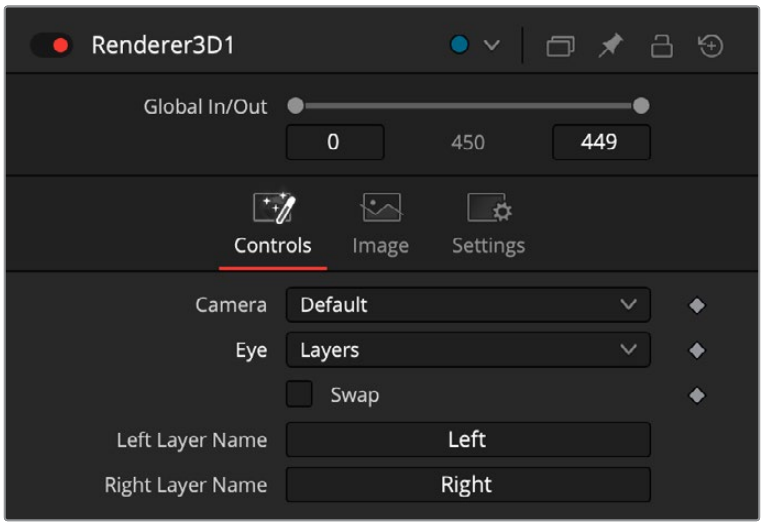
デフォルトでは「メディア入力1（MediaIn1）」は左目、「メディア入力2（MediaIn2）」は右目に設定されています。これにより、ステレオに対応した合成とグラフィックのオーバーレイの構築を即座に開始できます。また、「メディア出力（MediaOut）」ノードが2つあることで、映像をさらに調整するためにカラーページのノードエディターに送信される信号が2つの異なる信号になります。カラーページで「メディア入力2（MediaIn2）」の信号にアクセスするには、ノードエディターで右クリックして「ソースを追加」を選択します。



「ステレオの目」パラメーターが「自動」に設定されている場合、「メディア出力（MediaOut）」ノードは信号を結びつけ、タイムラインの適切な目に割り当てます。これにより、Fusionのマルチレイヤー・ワークフローを使用して、左右の信号を1つの「メディア出力（MediaOut）」ノードに送信できます。

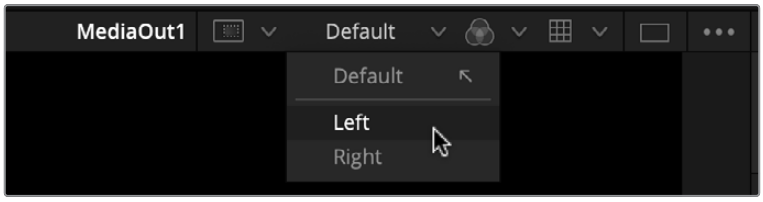
レンダラー3D (Renderder3D)

イマーシブタイムラインで3Dグラフィックや CGの作業を行う際に、「レンダラー3D（Renderder3D）」ノードでは複数の目のセットアップに対応しており、左右のレイヤーを処理できます。



これにより、複雑な3Dタイトルや CGシーン全体を作成し、イマーシブプロジェクトのステレオスコピックタイムラインに直接送信できます。

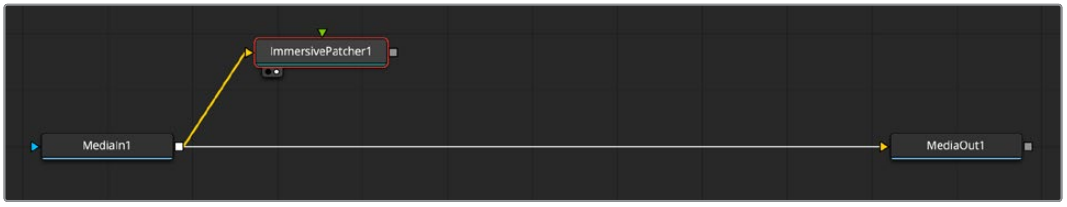
このセットアップで作業する際、Fusionビューアの右上にあるドロップダウンメニューで、合成の目の表示を左右で切り替えられます。



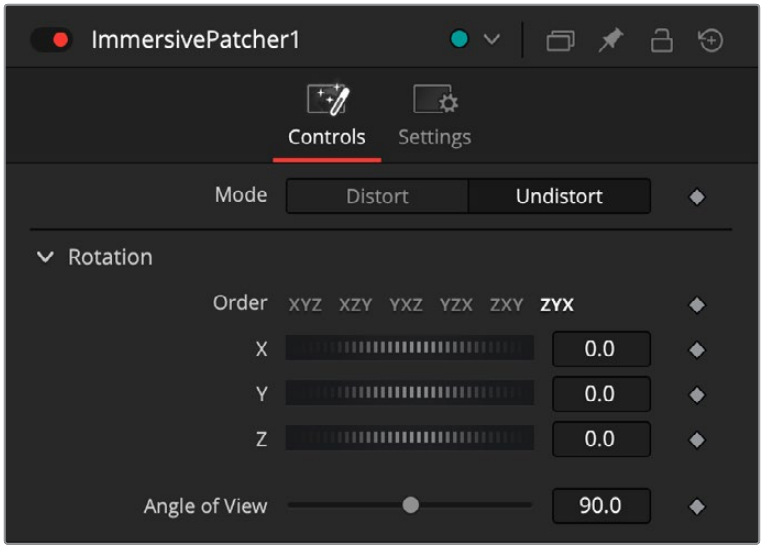
イマーシブパッチャーで合成

「イマーシブパッチャー」ツールでは、イマーシブクリップの一部の歪みを一時的に真っ直ぐにすることで、平面でトラッキング、ペインティング、合成が行えるため、作業が楽になります。終わったら、2つ目の「イマーシブパッチャー」でイマーシブ信号を再び歪め、シーンに新しく合成したエレメントを統合します。

1 「イマーシブパッチャー」をノードパイプラインに追加します。



2 「インスペクタ」で「歪みなし」をクリックします。



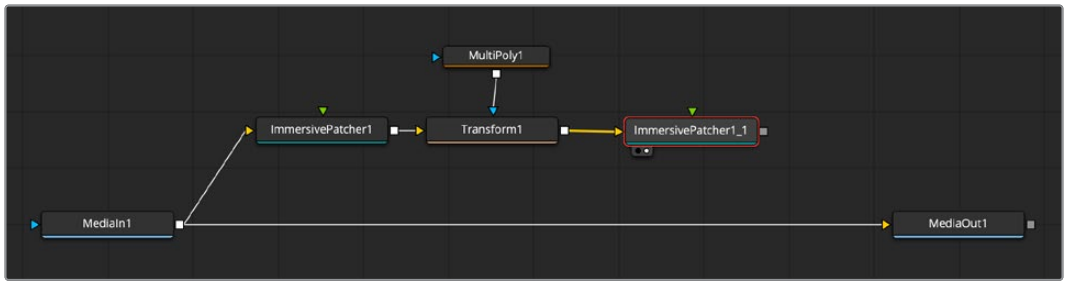
3 「回転」のX、Y、Zパラメーターで、イマーシブ映像の真っ直ぐにしたいセクションを選択します。

4 標準的な2Dビデオ信号と同様に合成の作業を行います。例えば、標準的なペイント方法を用いた不要なオブジェクトの削除、セット拡張のために変形やポリゴンを使用したフレームの一部のサンプリング、マット、テキスト、外部画像/ビデオレイヤーを適用した看板や画面の置き換え作業が含まれます。

終わったら、映像に再び歪みを適用して、元のイマーシブワープと一致するようにします。

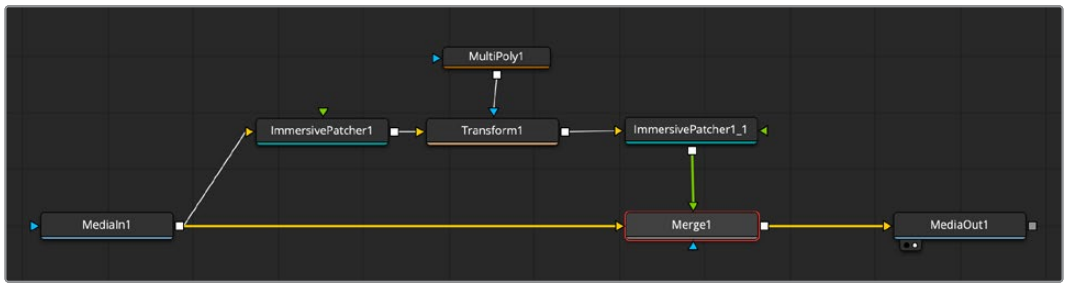
5 「イマーシブパッチャー」をコピーし、元の回転と視野角の情報を維持します。

6 合成ノードの後に「イマーシブパッチャー」をペーストし、合成パイプラインの最後に接続します。



7 「インスペクタ」で「歪み」をクリックします。

8 合成したパッチを元のクリップにマージし、エフェクトをイマーシブのシーンに統合します。



共同作業を行っている場合、Fusionを出て、DaVinci Resolveのイマーシブタイムラインで引き続き、編集、ミックス、グレーディングを実行できます。

イマーシブの反転を行う場合は、各目をACES AP0でEXRとしてレンダリングし、イマーシブの書き出しに必要な情報を最大限に維持します。

カラーグレーディング

このセクションでは、DaVinci Resolveのカラーページでイマーシブメディアを扱う際に使用できる光学、カラー、3Dツールについて説明します。

イマーシブコンテンツの性質により、ポストプロダクションですべてのツールを使用することはできません。その他のツールは現在使用できませんが、イマーシブコンテンツに対応できるように開発が行われています。

ノイズ除去

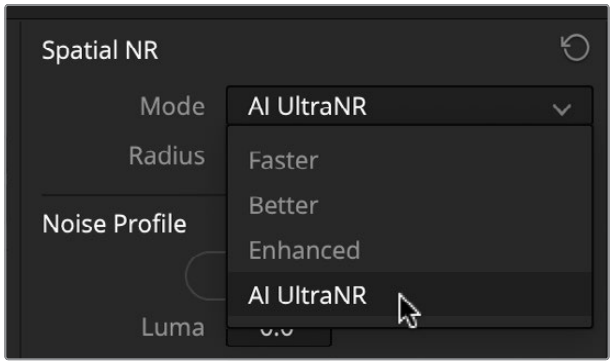
ノイズ除去は、イマーシブビデオで使用できます。「モーションエフェクト」パレットの「空間的ノイズ除去」は、ソースメディアのノイズのパターンを分析して、元のイメージのディテールを維持しながら、最大限にノイズを除去します。

1 カラーページのツールバーで「モーションエフェクト」パレットを開きます。

「空間的ノイズ除去」では任意のノイズ除去モードを選択できます。

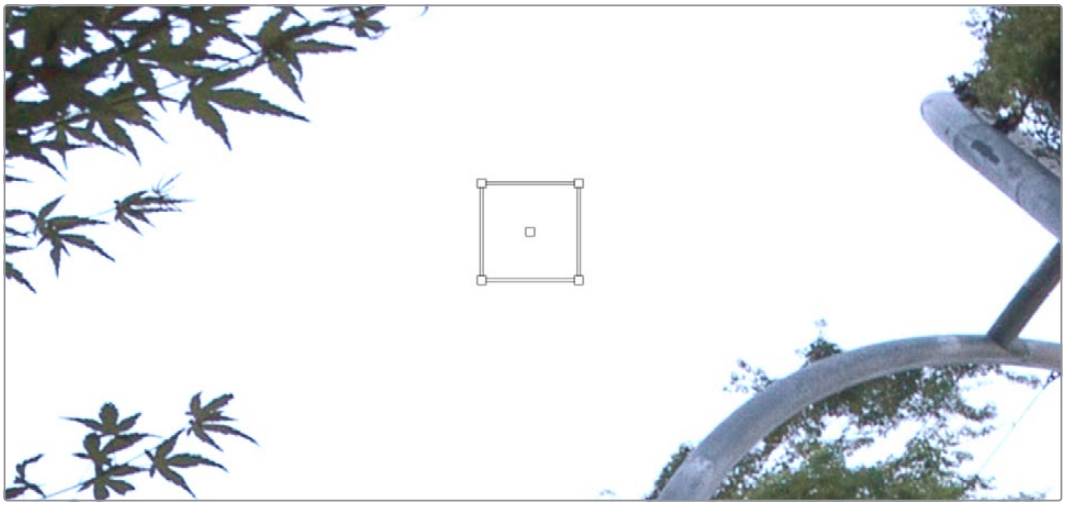
- 速度優先、品質優先、最高品質モードはすべて、定義した範囲内でパターンの分析を行い、設定されたしきい値に基づいて出力を構成します。各モードの記載順は、処理の複雑さに基づいています。
- *AI UltraNR*はAI駆動のノイズパターンの識別機能で、真のノイズを割り出すために、フレームの最も情報の少ないセクションを見つけ、分析することで、信号からノイズのパターンを抽出します。

2 「空間的ノイズ除去」を「AI UltraNR」モードに設定します。



3 「分析」をクリックします。

ビューアに境界ボックスが表示され、該当のフレームで最もピクセルデータが少ない領域を識別します。具体的な視覚コンテンツがないので、ボックス内のデータはカメラのセンサーによるデジタルノイズを抽出するのに最適です。



必要に応じて、ビューアの境界ボックスの位置やサイズは変更できますが、多くの場合は変更は勧められません。AI UltraNRは、人間の目より分析に適した領域を検出するのに優れているからです。

また、「ノイズプロファイル」の「輝度」および「クロマ」のしきい値はマニュアルで調整可能ですが、多くの場合、AI UltraNRはクリーンなノイズ除去を行うために最適なしきい値を自動的に設定します。

プライマリーグレーディング

プライマリーグレーディングでは、標準的なビデオフッテージと同じように操作します。プライマリーカラーホイールやLogホイール、HDRパレット、HSLカーブを使用して、ノーマライゼーション、バランス調整、マッチングを行う際に同じ考慮を払って作業を行います。「ColorSlice」と「カラーワーパー」パレットも同様に機能します。

セカンダリーコレクション

クオリファイアーなどのピクセルベースの選択は、イマーシブメディアで同様に機能します。

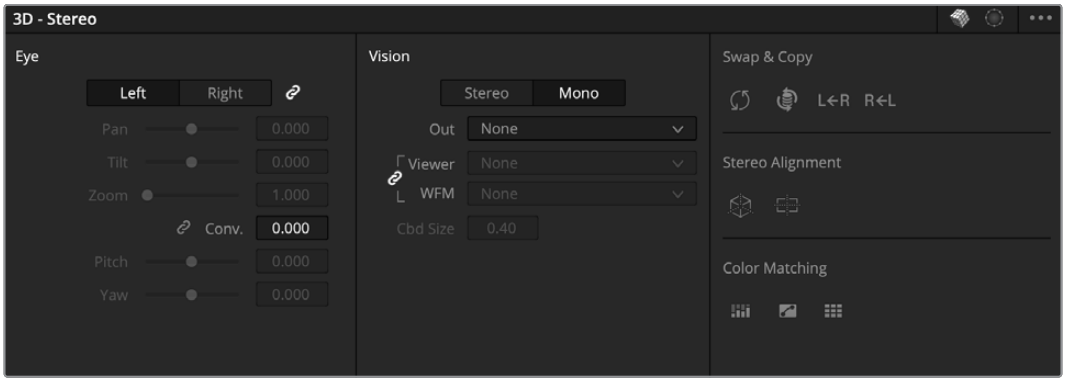
オブジェクトまたはウィンドウベースの選択は、真っ直ぐになったエクイレクタングラー（またはビューポート）の環境でのみ正確に機能します。つまり、高精度なクオリフィケーションの形状が要求されるセカンダリーの手法のほとんどは、現在使用できません。これには、Power Window（延長線上で考えると「トラッカー」パレット）およびMagic Maskツールが含まれます。

サイズ調整

イマーシブコンテンツでは、サイズ調整コントロール（ズームやパンなど）は使用できません。

3Dパレット

カラーページの「3D」パレットには、多様な3Dカメラとグラフィックフォーマットを管理するコントロールが搭載されています。他のステレオスコピックフォーマットとは異なり、目の並びを調整する必要はありません。その理由は、Blackmagic URSA Cine Immersiveカメラのレンズ、焦点距離、アイリスは固定されているからです。



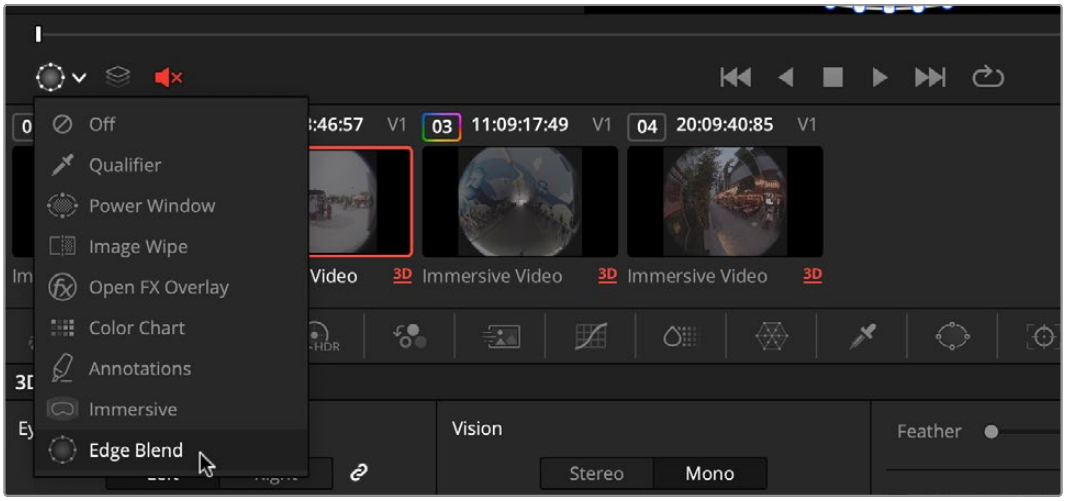
「エッジのブレンド」モードはイマーシブコンテンツ独自のもので、イマーシブビデオのフレームの境界を調整し、ソフトにできます。これは、映り込んだ撮影機器を隠したり、視聴者の視野を制限するために使用できます。このエッジを超えるとバックドロップが視聴者には見えます。

1 「3D」パレットを開きます。

2 右上でモードを「エッジのブレンド」に変更します。



3 ビューアのオンスクリーンコントロールが「エッジのブレンド」になっているか確認します。



ビデオフレームの境界に円形のコントロールポイントが配置され、中央にアンカーハンドルが表示されます。

4 コントロールポイントをドラッグすると配置を変えられます。



あるいは、右クリックするとポイントを削除できます。

5 「3D」パレットの右側の列にある「フェザー」でフレームのエッジをソフトにします。

6 「visionOSエフェクト」を選択して、Apple Vision Proでの再生において、新しいエッジのデータが適切に処理されるようにします。選択しない場合、エッジがレンダリングしたビデオファイルに焼き付けられます。

7 左側の列にある「左」と「右」のタブでビューアに表示する目を切り替えます。

「3D」パレットの多くのコントロールはグレーアウトしています。DaVinci Resolve内のパラメーターは、現在作業中のメディアで使用できない場合はグレーで表示されます。このケースでは、イマーシブコンテンツで使用できない機能がグレーになっています。

8 中央の列にある「ステレオ」と「モノ」のタブで、イマーシブビデオ・クリップのビューアでの表示をシングルレンズとダブルレンズで切り替えます。

CSTカラーマネージメント

カラーページのノードエディターでノードにカラースペース変換 (CST) エフェクトを使用すると、正確なカラーマッピングと、タイムラインの最大出力nit輝度のカスタムコントロールが可能です。

4つのCSTノードを使用するワークフローが推奨されます。

1 INPUT CST – カメラから作業カラースペース

カラーコレクションは、この2つのノードの間で行います。

2 OUTPUT CST – 作業カラースペースからHDRのMacディスプレイ

3 MONITOR CST – 1000 nitからHDRモニタリング用の250 nit

4 AVP CST – 250 nitsからApple Vision Proでのモニタリングおよび書き出し用の108 nit



CSTタイムラインマッピング

まず、カメラのソース入力カラースペースを作業用のタイムラインカラースペースにマッピングします。Blackmagic URSA Cine Immersiveカメラは、「Blackmagic Design Wide Gamut Gen 4/5」色域、「DaVinci Intermediate」Logカーブを用います。

1 「エフェクト」パネルを開きます。

2 「カラースペース変換」をノードエディターの空のノード (01) にドラッグします。

入力カラースペースからタイムラインカラースペースにマッピングする際は、このノードはノードツリーの最初のノードである必要があり、グレーディング用のノードはこの後に配置します。

入力カラースペースからモニタリングまたは書き出しカラースペースに直接マッピングしている場合、このノードはノードツリーの最後のノードである必要があり、グレーディング用のノードはこの前に配置します。

3 ノード (01) に**INPUT CST**と名前を付けます。

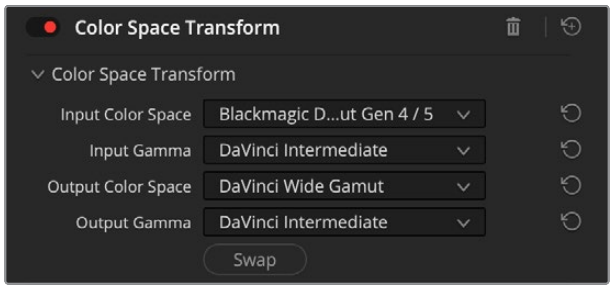
> **メモ**: このノード名は、このマニュアル内での表記とスクリーンショットを確認しやすくするために、全体を通して使用されています。自分で作業する際には任意の名前を付けてください。あるいは、名前を付けずに作業を進めることも可能です。

4 「入力カラースペース」を「Blackmagic Design Wide Gamut Gen 4 / 5」に設定します。

5 「入力ガンマ」を「DaVinci Intermediate」に設定します。

6 「出力カラースペース」を「DaVinci Wide Gamut」に設定します。

7 「出力ガンマ」を「DaVinci Intermediate」に設定します。



8 シリアルノード (02) を新規作成します。

9 最初のノード (01) を新しいノード (02) にコピー&ペーストします。

10 ノード (02) に**OUTPUT CST**と名前を付けます。

この出力カラースペースとガンマを新しいシリアルノードの入力にし、ノードツリーのタイムラインカラースペースに一貫性を持たせます。

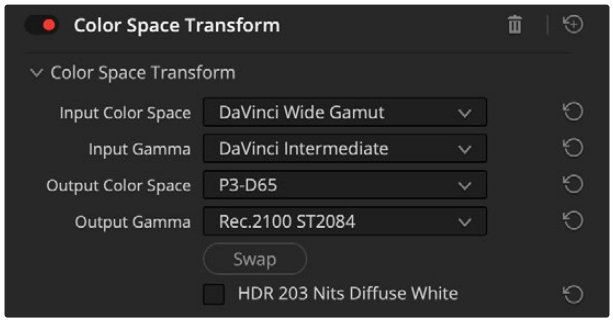
11 「反転」ボタンを押して、入力と出力のパラメーターを切り替えます。

このCSTノードの「入力カラースペース」と「入力ガンマ」は適切に設定されました。

12 「出力カラースペース」を「P3-D65」に設定します。これは、Macディスプレイのデフォルト色域です。

13 「出力ガンマ」を「Rec.2100 ST2084」に設定します。これは、1000 nitの信号のHDR規格です。

グレーディングは、この2つのノードの間で行います。しかし、ノードベースのカラーマネージメントで作業するため、タイムラインカラーマネージメント以上の作業も行えます。多くのカラリストはこの特徴を活かして、特定のカラーグレーディング、エフェクト、サードパーティのエレメントを適用しています。



ICCプロファイル

Blackmagic URSA Cine Immersiveの信号が適切にMacディスプレイにマッピングされることを確実にするために、「システム設定」>「ディスプレイ」に進み、カラープロファイルの「プリセット」が「Apple XDR Display (P3-1600 nits)」に設定されていることを確認します。

CSTグレーディングモニター・マッピング

追加のCSTを使用すると、グレーディング中のモニタリング用に最大nit出力値を一時的に下げることが可能です。

1 カラーマネージメント用の "OUTPUT CST" (02) の後に、シリアルノード (03) を新規作成します。

2 ノード (03) に**MONITOR**と名前を付けます。



3 "MONITOR" (03) に「カラースペース変換」エフェクトを追加します。

入力値は、前のCSTの出力信号と一致する必要があります。

4 「入力カラースペース」を「P3-D65」に設定します。

5 「入力ガンマ」を「Rec.2100 ST2084」に設定します。

このCSTノードの目的は、映像の輝度を低いnit値にマッピングすることのみです。つまり、入力と出力パラメーターは同じままで問題ありません。

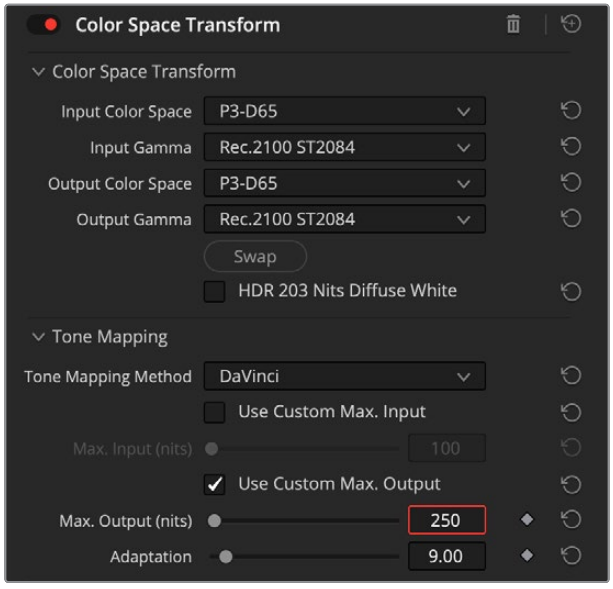
6 「出力カラースペース」を「P3-D65」に設定します。

7 「出力ガンマ」を「Rec.2100 ST2084」に設定します。

メモ: 特定のケースでは、入力と出力が同一の場合、マッピングは行われないため、CSTのカラースペースとガンマを設定する必要はありません。このケースでは、カスタムnitマッピングの適用が信号のLogカーブの値と形状の影響を受けるため、これらの設定を行う必要があります。

8 「カスタム最大出力を使用」を選択します。

9 「最大出力 (nit) 」を**250**に設定します。



これで1000 nitの信号は「トーンマッピング方法」に「DaVinci」を用いて、250 nitにマッピングされます。

その結果、映像はHDRグレーディングモニター (Flanders XMP310やSony BVM-HX310など) でレンダリングされたApple Vision Proの信号と極めて近くなります。

しかし、Apple Vision Proでプロジェクトをモニタリングするには、もう一つトーンマッピングを行う必要があります。

CST Apple Vision Proモニタリング

最後のCSTノードをApple Vision Proでのライブモニタリング、プレビュー、最終書き出しに使用することも可能です。

1 "MONITOR" (03) の後に、シリアルノード (04) を新規作成します。

2 "MONITOR" (03) を新しいノード (04) にコピー&ペーストします。同じ入力と出力パラメーターを使用し、nit値のみを変更します。

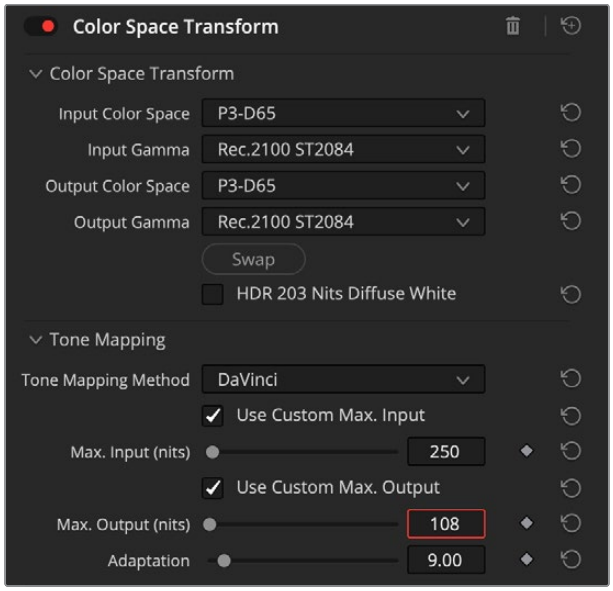
3 ノード (04) に**AVP**と名前を付けます。



4 「カスタム最大入力を使用」を選択します。

5 「最大入力 (nit)」を**250**に設定し、"MONITOR" ノードからのマッピングされた信号を適切に読み取られるようにします。

6 「最大出力 (nit)」を**108**に設定します。これは、Apple Vision Proでのモニタリングおよび書き出しに使用する最大nit値です。



これで、グレーディング作業を開始できます。ワークフローのニーズに応じて必要であれば、CSTノードは無効にしてください。例えば、最初のグレーディングパスで外部グレーディングモニターを使用している場合、"AVP" (04) を無効にして、HDRグレーディングモニターで250 nitの信号が適切に表示されているか確認してください。Apple Vision Proでのプレビューや最終書き出しの際は、再び "AVP" (04) を有効にします。

CSTはタイムラインのノードエディターに配置できる?

いいえ。現在、クリップのトーンマッピングは「カラースペース変換」パラメーターの正確な適用に基づいており、これはクリップベースで行われる必要があります。

オーディオ

Apple Vision ProとDaVinci Resolveは、空間オーディオとアンビソニック・オーディオ、AVP（Apple Vision Pro）オーディオ、AIA（Apple Immersive Audio）フォーマットに対応しています。

空間オーディオでの作業において、オーディオ・ポストプロダクションの多くの面は変わりません。大きな違いの一つは、ハイエンドのスピーカーでモニタリングするのではなく、オーディオエンジニアはヘッドフォンやAirPodsを使用して、真の空間没入を確認し、視聴者のリスニング環境を再現してモニタリングを行うべきです。

2Dのミックスでは、会話音声の大部分は中央チャンネルにミックスされますが、空間オーディオでは音声が発生している場所にミックスする必要があります。オーディオステムのミックスを行う際に使用するダッキング、EQ、バランス調整は、視聴者を圧倒させたり、混乱させないような方法で使用する必要があります。

イマーシブの空間エレメントは標準的なステレオステムと組み合わせられるため、頭を動かしても特定のサウンドが一定に聞こえるようにできます。これは、音楽やボイスオーバーなどの非物語世界の音を聞きやすくかつ把握しやすくするのに役立ちます。

空間オーディオでは、環境音は把握されにくいため、多くの場合、強めのミックスにする必要があります。しかし、これによって視聴者の方向感覚が失われないようにしてください。

リバーブは日々の生活で視聴者が自然に聞くものなので、没入感を作り出す上で非常に役立ちます。特定の環境で生成される全ての音は、聞く人を囲む表面で反射し、その結果として残響音を生み出します。したがって、リバーブは完全な没入感を生み出すために空間オーディオでの使用が推奨されています。

Apple Immersive Audio（AIA）は複数のフォーマットを含むコンテナです。

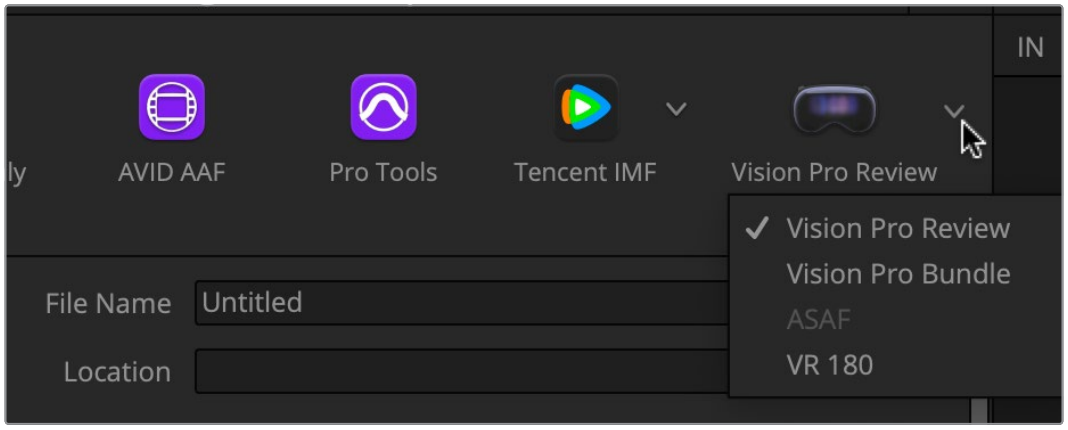
- **Apple Spatial Audio Format (ASAF)** は、RAWの作業オーディオフォーマットです。
- **Apple Positional Audio Codec (APAC)** は、AirPodsなどのAPACに対応したデバイスで再生可能なエンコードされた.mp4フォーマット、あるいはApple Vision ProのAIVUファイルからのものです。

Apple Spatial Audio Format (ASAF)

Apple Spatial Audio Format（ASAF）は、主にApple Vision ProのVRおよびARでの使用を意図して開発されたイマーシブオーディオ・フォーマットです。macOSのみに対応しているASAFは、ProTools（プラグイン経由）とDaVinci Resolve（内蔵）でサポートされています。ASAFは、Ambisonicsソース（呼称：シーン）の内蔵サポートとバスの割り当てを介したルーティングなど、高度な技術に対応しています。ミキシングは仮想で行われ、物理的なバスは使用されません。ユーザーは様々なモニタリングフォーマットから選択できます。これには、Apple AirPodsの直接統合を介したヘッドトラッキング対応のバイノーラルが含まれます。パンは、パンダイアログ、オートメーション、SpaceView、ビューア内のパンツールなどのDaVinci Resolveの既存のツールを「インスペクタ」のセカンダリーコントロールと組み合わせて使用することで適用します。これにより、ソースを空間内に高精度で配置できます。例えば、緯度経度ビデオに対する自動トラッキングなどが含まれます。ASAFミックスは、付随するメタデータ込みでWAVまたはMP4フォーマットで書き出せます。これは、「イマーシブオーディオ」>「マスターファイルの書き出し」で実行できます。しかし、現在、ASAFミックスはDaVinci Resolve内で読み込みや再生に対応していません。

書き出し

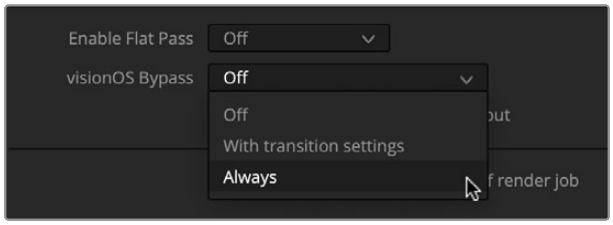
デリバーページに搭載されている2つのレンダリング設定で、レビューと最終書き出し用のApple Vision Proファイルをレンダリングできます。



- **Vision Proレビュー** – プレビュー用の小さなMV-HEVCパッケージを作成します。Appleは、Wi-Fiを介してApple Vision Proにファイルを共有し、Apple Immersive Video Utility（AIVU）アプリで視聴できるようにします。このアプリはApp Storeで入手できます。
- **Vision Proバンドル** – 大容量のProResファイルバンドルを作成します。これは、Apple Compressorでアーカイブまたは処理し、Apple TVで再生できる準備を行います。
- **ASAF** – WAVまたはMP4フォーマットでASAFオーディオミックスをレンダリングします。
- **VR180** – VR180に準拠したステレオスコピック・ビデオファイルを生成します。

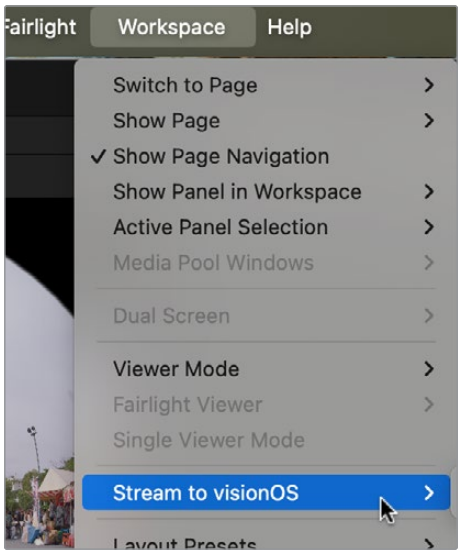
トランジションのバイパス

イマーシブタイムラインにトランジションが追加されている場合、プリレンダーされ、1つの再生トラックになります。イマーシブコンテンツ用に生成されたXMLには、これらのレンダリングされたストリームでの切り替えプロセス（フッテージ、次にプリレンダーしたトランジション、その後、フッテージに戻る）が記載されています。「レンダー設定」パネルの「詳細設定」で、「常に」でvisionOSトランジションをバイパスするか、「トランジション設定」でタイムラインでのステータスに基づいて適用するか、「オフ」でタイムラインでの表示に含めるかを選択できます。



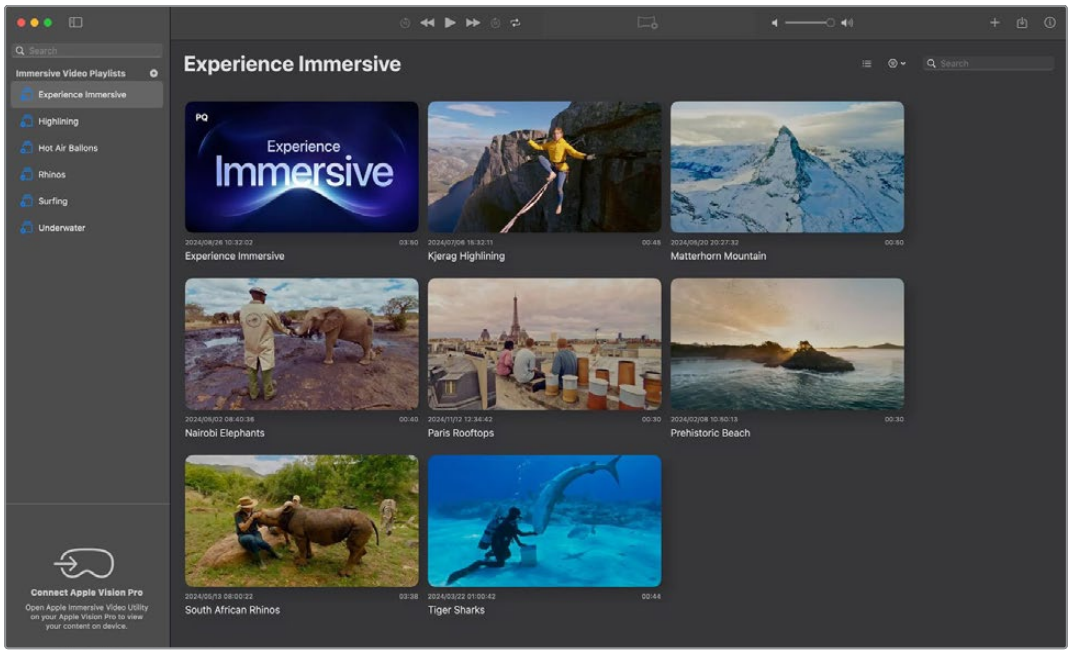
Apple Vision Proへの配信

DaVinci Resolveのエディット、カラー、デリバーページからApple Vision Proに配信できます。これは「ワークスペース」メニューで「visionOSに配信」を選択して、配信先のAVPデバイスを選択することで実行できます。円滑な配信には、高速ワイヤレスネットワーク（Wi-Fi 5以上、Wi-Fi 6を推奨）をお勧めします。



Apple Immersive Video Utilityについて

Apple Immersive Video Utilityは専用のmacOSおよびvisionOSアプリで、Apple Immersive Videoコンテンツの作成、管理、再生を簡素化するように作られています。プロの映像作家やイマーシブメディアのクリエイター向けに設計されている同ユーティリティでは、180°の視野と空間オーディオを使用して、8Kステレオスコピックビデオ・ファイルの読み込み、整理、パッケージ、レビューをユーザーが実行できます。AIVU（Apple Immersive Video Universal）ファイルの作成、メタデータ検査、ポストプロダクションのワークフロー用の調整に対応しています。ユーザーは、プレイリストの構築、ライブラリの並べ替えと検索、さらには複数のApple Vision Proデバイスへのコンテンツの配信が行えます。



Developer Information

Camera Control REST API

If you are a software or hardware developer you can build custom applications or leverage ready to use tools such as REST client or Postman to seamlessly control and interact with your compatible Blackmagic camera using Camera Control REST API. This API enables you to perform a wide range of operations, such as starting or stopping recordings, accessing disk information and much more. Whether you're developing a custom application tailored to your specific needs or utilizing existing tools, this API empowers you to unlock the full potential of your Blackmagic camera with ease. We look forward to seeing what you come up with!

NOTE It's important to mention that controlling Blackmagic cameras via REST API relies on the web manager being enabled on each compatible Blackmagic camera. Enable the web media manager in the Blackmagic Camera Setup 'network access' settings for each camera you are controlling.

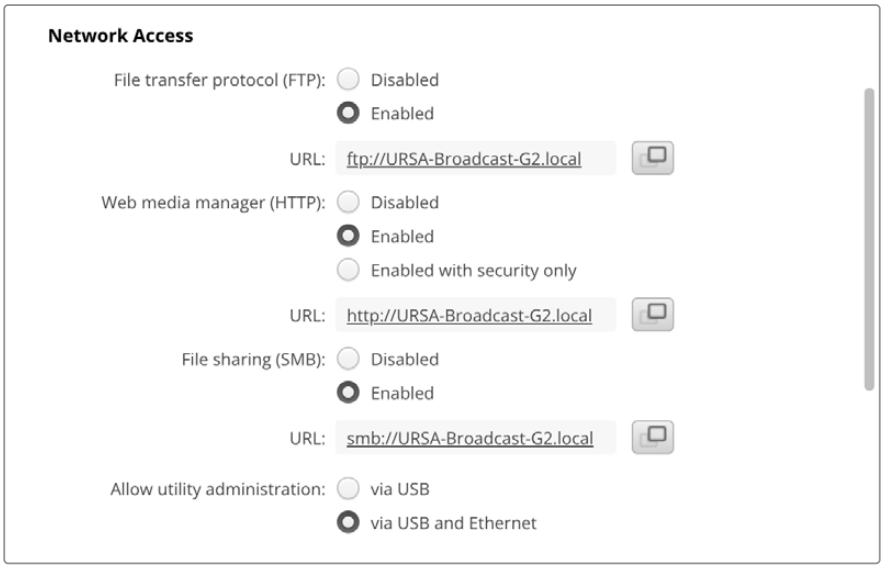
The following Blackmagic cameras are compatible with Camera Control REST API:

- Blackmagic URSA Cine 12K LF
- Blackmagic URSA Cine 17K 65
- Blackmagic URSA Cine Immersive
- Blackmagic PYXIS 6K
- Blackmagic Cinema Camera 6K
- Blackmagic URSA Broadcast G2
- Blackmagic Micro Studio Camera 4K G2
- Blackmagic Studio Camera 4K Plus
- Blackmagic Studio Camera 4K Pro
- Blackmagic Studio Camera 6K Pro
- Blackmagic Studio Camera 4K Plus G2
- Blackmagic Studio Camera 4K Pro G2

Sending API Commands

To send an API command to your camera from a third party application such as Postman, add /control/api/v1/ to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/api/v1/

You can find the Web media manager URL and IP address information in Blackmagic Camera Setup.



The Web media manager URL in Blackmagic Camera Setup

Downloading API's from your Camera

You can download REST API YAML documentation from your camera by adding /control/documentation.html to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/documentation.html

NOTE It's worth noting that changing the camera name in Blackmagic Camera Setup will also change the camera's Web media manager URL.

Livestream Control API

API for controlling Livestreams on Blackmagic Design products.

GET /livestreams/0

Get the livestream's current status.

Response

200 - Livestream's current status.

The response is JSON.

Name	Type	Description
status (required)	string	Possible values are: Idle, Connecting, Streaming, Flushing, Interrupted.
bitrate (required)	integer	Current bitrate (bps).
effectiveVideoFormat (required)	string	Effective video format for the livestream, serialised as a string.
duration	integer	Current stream duration in seconds. Absent if livestream is idle.
cache	integer	Current stream cache usage percentage.

GET /livestreams/0/start

Determine if the livestream is active.

Response

200 - Livestream active status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is active.

PUT /livestreams/0/start

Start the livestream.

Response

204 - Livestream started.

GET /livestreams/0/stop

Determine if the livestream is inactive.

Response

200 - Livestream inactive status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is inactive.

PUT /livestreams/0/stop

Stop the livestream.

Response

204 - Livestream stopped.

GET /livestreams/0/activePlatform

Get the currently selected platform configuration for the livestream.

Response

200 - Livestream active platform configuration.

The response is JSON.

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

PUT /livestreams/0/activePlatform

Set the currently selected platform configuration for the livestream.

Parameters

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

Response

204 - Livestream active platform configuration updated.

400 - Bad Request

GET /livestreams/platforms

Get the list of available platforms.

Response

200 - List of available platforms.

The response is JSON.

Name	Type	Description
	array	List of available platforms names.
[i]	string	Platform name.

GET /livestreams/platforms/{platformName}

Get the service configuration for a platform.

Parameters

Name	Type	Description
{platformName} (required)	string	Name of the platform.

Response

200 - Service configuration for specified platform.

The response is JSON.

Livestream platform service configuration.

Name	Type	Description
platform (required)	string	Corresponding platform name.
key	string	Default stream key.
servers (required)	array	List of server configurations.
servers[i]	object	Server configuration.
servers[i].server (required)	string	Server name.
servers[i].url (required)	string	Livestream destination.
servers[i].srtExtensions	array	Miscellaneous tags used for SRT livestreams.
servers[i].srtExtensions[i]	object	Dictionary object mapping SRT tag strings to values.
servers[i].srtExtensions[i][{key}]	string	SRT tag value.
servers[i].group	string	Logical grouping of the server.
profiles (required)	array	List of profile configurations.
profiles[i]	object	Quality configuration.
profiles[i].profile (required)	string	Quality level name.
profiles[i].configs (required)	array	List of video format configurations.
profiles[i].configs[i]	object	Video format configuration for profiles.
profiles[i].configs[i].resolution (required)	string	Video format serialised as a string.
profiles[i].configs[i].fps (required)	string	Frames per second.
profiles[i].configs[i].bitrate (required)	integer	Pixel bitrate (bps).
profiles[i].configs[i].audioBitrate	integer	Audio bitrate (bps).
profiles[i].configs[i].keyFrameInterval	integer	How often a key frame is sent, in seconds.
profiles[i].configs[i].videoCodecs	array	Supported video encoding algorithm/s.

Name	Type	Description
profiles[i].configs[i].videoCodecs[i]	string	Video encoding algorithm. Possible values are: H264, H265.
profiles[i].lowLatency (required)	boolean	If true, fewer frames will be buffered in the livestream.
defaultProfile	string	Quality level name.
credentials	object	Credientials used for RTMP streams.
credentials.username (required)	string	The username part of the creditials. Only used for RTMP streams.
credentials.password (required)	string	Used for RTMP streams, also used as Passphrase for SRT streams.
customizableUrlEnabled	boolean	True when the server URL is customizable.

400 - Bad Request

GET /livestreams/customPlatforms

Get a list of custom platform files.

Response

200 - List of custom platform files.

The response is JSON.

Name	Type	Description
	array	List of custom platform file names.
[i]	string	Custom platform file name.

DELETE /livestreams/customPlatforms

Remove all custom configuration files.

Response

204 - All custom configuration files removed.

GET /livestreams/customPlatforms/{filename}

Get a custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to get.

Response

200 - Custom platform file.

The response is XML.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

404 - Not Found

PUT /livestreams/customPlatforms/{filename}

Update a custom platform file if it exists, if not, create a new file with the given file name.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to update/create.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

Response

204 - Custom platform file created or updated.

400 - Bad Request

DELETE /livestreams/customPlatforms/{filename}

Remove the given custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to be removed.

Response

204 - Custom platform file removed.

404 - Not Found

Clips Control API

API for listing clips on disk.

GET /clips

Get the list of clips on the active disk.

Response

200 - List of clips on the active disk.

The response is JSON.

List of media clips.

Name	Type	Description
clips (required)	array	
clips[i]	object	Media clip.
clips[i].clipUniqueId (required)	integer	Unique ID used to identify this clip.
clips[i].filePath	string	Path to the file relative to the root of a mount.
clips[i].fileSize	integer	Size of file on disk in bytes.
clips[i].codecFormat	object	
clips[i].codecFormat.codec	string	Currently selected codec.

Name	Type	Description
clips[i].codecFormat.container	string	Multimedia container format.
clips[i].videoFormat	object	Video format configuration.
clips[i].videoFormat.name (required)	string	Video format serialised as a string.
clips[i].videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
clips[i].videoFormat.height	number	Height dimension of video format.
clips[i].videoFormat.width	number	Width dimension of video format.
clips[i].videoFormat.interlaced	boolean	Is the display format interlaced?
clips[i].startTimecode	string	Start timecode of the clip serialised as string.
clips[i].durationTimecode	string	Duration of the clip in timecode format serialised as string.
clips[i].frameCount	integer	Number of frames in clip; duration of the clip in frames.

404 - There is no active disk.

Media Pool Control API

API to manage media pool and handle uploads and project data.

GET /cloud/projects

List all projects within the media pool.

Response

200 - Successfully retrieved the list of all projects.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].libraryID	string	
[i].name	string	
[i].private	boolean	
[i].shared	boolean	
[i].clips	array	List of clips associated with the project.
[i].clips[i]	string	
[i].status	object	
[i].status.numClipsRequested	integer	
[i].status.numClipsComplete	integer	
[i].status.uploadPercent	integer	Percentage of upload completion.
[i].status.numClipsPaused	integer	
[i].status.outOfSpace	boolean	
[i].status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
[i].status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/active

Retrieve data of the actively uploading project.

Response

200 - Successfully retrieved the active project's data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/{projectID}

Retrieve specific project data by project ID.

Parameters

Name	Type	Description
{projectID} (required)	integer	Unique identifier of the project.

Response

200 - Successfully retrieved the project data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	

Name	Type	Description
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

404 - Project not found.

GET /cloud/clips

List all clips within the media pool.

Response

200 - Successfully retrieved the list of all clips.

The response is JSON.

Name	Type	Description
	array	
[i]	string	

GET /cloud/clips/activeUploading

Retrieve data of actively uploading clips.

Response

200 - Successfully retrieved the list of actively uploading clips.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].path	string	
[i].projectID	integer	
[i].status	object	
[i].status.projectID	integer	
[i].status.outOfSpace	boolean	
[i].status.proxyExtension	string	
[i].status.growingFile	boolean	
[i].status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.originalClipTotalSize	integer	
[i].status.proxyClipTotalSize	integer	
[i].status.originalClipCompletedSize	integer	
[i].status.proxyClipCompletedSize	integer	
[i].status.secsRemaining	integer	

GET /cloud/clips/{deviceName}/{path}

Retrieve specific clip data by device and path.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored.
{path} (required)	string	Path to the clip.

Response

200 - Successfully retrieved the clip data.

The response is JSON.

Name	Type	Description
path	string	
projectID	integer	
status	object	
status.projectID	integer	
status.outOfSpace	boolean	
status.proxyExtension	string	
status.growingFile	boolean	
status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.originalClipTotalSize	integer	
status.proxyClipTotalSize	integer	
status.originalClipCompletedSize	integer	
status.proxyClipCompletedSize	integer	
status.secsRemaining	integer	

404 - Clip not found.

Monitoring Control API

API for monitoring and controlling display settings in video equipment.

GET /monitoring/display

Retrieve a list of all display names.

Response

200 - Returns a list of display names.

The response is JSON.

Name	Type	Description
displays	array	List of display names available.
displays[i]	string	

GET /monitoring/{displayName}/cleanFeed

Get the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/cleanFeed

Set the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Clean feed enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/displayLUT

Get the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

400 - Invalid display name.

404 - Display name not found.

PUT /monitoring/{displayName}/displayLUT

Set the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Display LUT enabled/disabled successfully.

400 - Invalid input.

422 - Unprocessable Entity - Unable to process the contained instructions.

GET /monitoring/{displayName}/zebra

Get the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/zebra

Set the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Zebra enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/focusAssist

Get the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/focusAssist

Set the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/focusAssist

Get the focus assist settings.

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

404 - Display name not found.

PUT /monitoring/focusAssist

Set the focus assist settings.

Parameters

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/frameGuide

Get the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame guide enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGuide

Set the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame guide state updated successfully.

422 - Unable to update the frame guide state.

GET /monitoring/frameGuideRatio

Get the current frame guide ratio.

Response

200 - Returns the current frame guide ratio.

The response is JSON.

Name	Type	Description
ratio	string	The frame guide ratio.

PUT /monitoring/frameGuideRatio

Set the frame guide ratio.

Parameters

Name	Type	Description
ratio	string	The frame guide ratio.

Response

204 - Frame guide ratio updated successfully.

422 - Unable to update the frame guide ratio.

GET /monitoring/frameGuideRatio/presets

Get the presets for frame guide ratios.

Response

200 - Returns a list of preset frame guide ratios.

The response is JSON.

Name	Type	Description
presets	array	
presets[i]	string	A frame guide ratio.

GET /monitoring/{displayName}/frameGrids

Get the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame grids enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGrids

Set the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame grids state updated successfully.

422 - Unable to update the frame grids state.

GET /monitoring/frameGrids

Get the global frame grids settings.

Response

200 - Returns the current frame grids settings.

The response is JSON.

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

PUT /monitoring/frameGrids

Set the global frame grids settings.

Parameters

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

Response

204 - Frame grids settings updated successfully.

400 - Invalid input, check the number of frame grids or values.

422 - Unable to update the frame grids settings.

GET /monitoring/{displayName}/safeArea

Get the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the safe area enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/safeArea

Set the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Safe area state updated successfully.

422 - Unable to update the safe area state.

GET /monitoring/safeAreaPercent

Get the current safe area percentage.

Response

200 - Returns the current safe area percentage.

The response is JSON.

Name	Type	Description
percent	integer	Safe area coverage percentage.

PUT /monitoring/safeAreaPercent

Set the safe area percentage.

Parameters

Name	Type	Description
percent	integer	Safe area coverage percentage to set.

Response

204 - Safe area percentage updated successfully.

400 - Invalid percentage value.

422 - Unable to update the safe area percentage.

GET /monitoring/{displayName}/falseColor

Get the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the false color enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/falseColor

Set the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - False color state updated successfully.

422 - Unable to update the false color state.

Event Control API

API For working with built-in websocket.

GET /event/list

Get the list of events that can be subscribed to using the websocket API.

Response

200 - Websocket API events list.

The response is JSON.

Name	Type	Description
events	array	List of events that can be subscribed to using the websocket API.
events[i]	string	

System Control API

API for controlling the System Modes on Blackmagic Design products.

GET /system

Get device system information.

Response

200 - System summary.

The response is JSON.

The properties will be populated only with the values that are supported/implemented by the device in use.

Name	Type	Description
codecFormat	object	Codec format configuration.
codecFormat.codec	string	Codec serialised as string.
codecFormat.container	string	Multimedia container format.
videoFormat	object	Video format configuration.
videoFormat.name (required)	string	Video format serialised as a string.
videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
videoFormat.height	number	Height dimension of video format.
videoFormat.width	number	Width dimension of video format.
videoFormat.interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/product

Get device product information.

Response

200 - Device product information.

The response is JSON.

Product information.

Name	Type	Description
deviceName	string	Name of device as displayed in Setup.
productName	string	Device's product name.
softwareVersion	string	Software version running on device.

501 - This functionality is not implemented for the device in use.

GET /system/supportedCodecFormats

Get the list of supported codecs.

Response

200 - List of supported codec formats.

The response is JSON.

Name	Type	Description
codecs	array	
codecs[i]	object	Codec format configuration.
codecs[i].codec	string	Codec serialised as string.
codecs[i].container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

GET /system/codecFormat

Get the currently selected codec.

Response

200 - Current codec format.

The response is JSON.

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

PUT /system/codecFormat

Update the system codec.

Parameters

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

Response

204 - The codec updated successfully.

400 - The specified codec format is unsupported.

501 - This functionality is not implemented for the device in use.

GET /system/videoFormat

Get the currently selected video format.

Response

200 - Current system video format.

The response is JSON.

Video format configuration.

Name	Type	Description
name (required)	string	Video format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height	number	Height dimension of video format.
width	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

PUT /system/videoFormat

Set the system video format.

Parameters

This parameter can be one of the following types:

Name	Type	Description
name (required)	string	Video format serialised as a string.

Name	Type	Description
frameRate (required)	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height (required)	number	Height dimension of video format.
width (required)	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

Response

204 - The video format updated successfully.

400 - Invalid request.

409 - Operation unsupported in the current state.

501 - This functionality is not implemented for the device in use.

GET /system/supportedVideoFormats

Get the list of supported video formats for the current system state.

Response

200 - List of supported video formats.

The response is JSON.

List of supported video formats.

Name	Type	Description
formats	array	List of video formats.
formats[i]	object	Video format configuration.
formats[i].name (required)	string	Video format serialised as a string.
formats[i].frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
formats[i].height	number	Height dimension of video format.
formats[i].width	number	Width dimension of video format.
formats[i].interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/supportedFormats

Get supported formats.

Response

200 - List of supported formats.

The response is JSON.

Name	Type	Description
supportedFormats	array	
supportedFormats[i]	object	
supportedFormats[i].codecs	array	
supportedFormats[i].codecs[i]	string	
supportedFormats[i].frameRates	array	
supportedFormats[i].frameRates[i]	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
supportedFormats[i]. maxOffSpeedFrameRate	number	
supportedFormats[i]. minOffSpeedFrameRate	number	
supportedFormats[i]. recordResolution	object	
supportedFormats[i]. recordResolution.height	number	Height of the resolution.
supportedFormats[i]. recordResolution.width	number	Width of the resolution.
supportedFormats[i]. sensorResolution	object	
supportedFormats[i]. sensorResolution.height	number	Height of the resolution.
supportedFormats[i]. sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

GET /system/format

Get current format.

Response

200 - Current format.

The response is JSON.

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

PUT /system/format

Set the format.

Parameters

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

Response

204 - System format updated.

501 - This functionality is not implemented for the device in use.

Transport Control API

API for controlling Transport on Blackmagic Design products.

GET /transports/0

Get device's basic transport status.

Response

200 - Transport status.

The response is JSON.

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

PUT /transports/0

Set device's basic transport status.

Parameters

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, Output.

Response

204 - Transport mode was set.

400 - Failed to set transport mode.

GET /transports/0/stop

Determine if transport is stopped.

Response

200 - Transport stop response.

The response is JSON.

Name	Type	Description
	boolean	True when transport mode is InputPreview or when in Output mode and speed is 0.

PUT /transports/0/stop

Stop transport. Deprecated, use POST /transports/0/stop instead.

Response

204 - Transport stopped.

POST /transports/0/stop

Stop transport.

Response

204 - Transport stopped.

GET /transports/0/play

Determine if transport is playing.

Response

200 - Transport play response.

The response is JSON.

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

PUT /transports/0/play

Start playing on transport. Deprecated, use POST /transports/0/play instead.

Response

204 - Transport playing.

400 - Failed to set transport to play.

POST /transports/0/play

Start playing on transport.

Response

204 - Transport playing.

400 - Failed to set transport to play.

GET /transports/0/playback

Get playback state.

Response

200 - Transport playback state.

The response is JSON.

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

PUT /transports/0/playback

Set playback state.

Parameters

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

Response

204 - Updated transport playback state.

400 - Failed to set transport playback state.

GET /transports/0/record

Get record state.

Response

200 - Recording state.

The response is JSON.

Name	Type	Description
recording	boolean	If true, transport is in InputRecord mode.

PUT /transports/0/record

Set record state. Deprecated, use POST /transports/0/record instead.

Parameters

Name	Type	Description
recording (required)	boolean	If true, starts a recording, otherwise stops.
clipName	string	Optional, sets the requested clip name to record to, when “recording” attribute is set to true.

Response

204 - Recording state updated.

400 - Failed to update recording state.

POST /transports/0/record

Start recording.

Parameters

Name	Type	Description
clipName	string	Optional, provides a specific name of clip to record to.

Response

204 - Recording started.

400 - Failed to start recording.

GET /transports/0/clipIndex

Get the clip index of the currently playing clip on the timeline.

Response

200 - Clip index response.

The response is JSON.

Name	Type	Description
clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

GET /transports/0/timecode

Get device timecode.

Response

200 - Timecode response.

The response is JSON.

Name	Type	Description
display	string	The display timecode serialised as a string.
timeline	string	The timeline timecode serialised as a string.

GET /transports/0/timecode/source

Get timecode source selected on device.

Response

200 - Timecode source response.

The response is JSON.

Name	Type	Description
timecode	string	Possible values are: Timeline, Clip.

Timeline Control API

API for controlling playback timeline.

GET /timelines/0

Get the playback timeline.

Response

200 - Playback timeline.

The response is JSON.

Name	Type	Description
clips	array	
clips[i]	object	Timeline clip.
clips[i].clipUniqueId (required)	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).
clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

404 - No timeline / disk available.

DELETE /timelines/0

Clear the current playback timeline. Deprecated, prefer to use POST /timelines/0/clear

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

POST /timelines/0

Add a clip to the timeline.

Parameters

This parameter can be one of the following types:

Add multiple media clips to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips[i].clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips[i].clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips[i].frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add multiple media clips to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add a single clip to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips.clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips.clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips.frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add a single clip to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the timeline as specified.

501 - The operation is not supported on this device.

POST /timelines/0/add

Add a clip to the end of the timeline. Deprecated, use POST /timelines/0 to add clips within the timeline.

Parameters

This parameter can be one of the following types:

Add one clip to the end of the timeline.

Name	Type	Description
clips	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add many clips to the end of the timeline.

Name	Type	Description
clips	array	List of clipUniqueIds of clips to add to end of timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the end of the timeline.

501 - The operation is not supported on this device.

POST /timelines/0/clear

Clear the playback timeline.

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

DELETE /timelines/0/clips/{timelineClipIndex}

Remove the specified clip from the timeline.

Parameters

Name	Type	Description
{timelineClipIndex} (required)	integer	The (0-based) timeline clip index of the clip to remove from the timeline.

Response

204 - The specified clip was removed from the timeline.

501 - The operation is not supported on this device.

Media Control API

API for controlling media devices in Blackmagic Design products.

GET /media/workingset

Get the list of media devices currently in the working set.

Response

200 - The list of media devices in the working set.

The response is JSON.

Name	Type	Description
size	integer	The fixed size of this device's working set.
workingset	array	The device's working set.
workingset[i]	object | null	Device within the working set. null if no device is present within the given working set slot.
workingset[i].volume	string	Volume name.
workingset[i].deviceName	string	Internal device name of this media device.
workingset[i].remainingRecordTime	integer	Remaining record time using current codec and video format in seconds.
workingset[i].totalSpace	integer	Total space on media device in bytes.
workingset[i].remainingSpace	integer	Remaining space on media device in bytes.
workingset[i].clipCount	integer	Number of clips currently on the device.

GET /media/active

Get the currently active media device.

Response

200 - The current active media device.

The response is JSON.

The active media device, or null if there is no active media.

Name	Type	Description
workingsetIndex	integer	Working set index of the active media device.
deviceName	string	Device name of media device.

204 - No media is currently active.

PUT /media/active

Set the currently active media device.

Parameters

Name	Type	Description
workingsetIndex	integer	Working set index of the media to make active.

Response

204 - The active media device was set successfully.

400 - Setting the currently active media device is not possible in the current state.

GET /media/devices/doformatSupportedFilesystems

Get the list of filesystems available to format a media device.

Response

200 - The list of filesystems permitted for formatting.

The response is JSON.

Name	Type	Description
	array	List of filesystems permitted for formatting media.
[i]	string	Filesystem serialised as string.

GET /media/devices/{deviceName}

Get information about a requested device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Information about the requested device.

The response is JSON.

Media device state.

Name	Type	Description
state	string	The current state of the media device. Possible values are: None, Scanning, Mounted, Uninitialised, Formatting, RaidComponent.

400 - Invalid device name.

404 - Device not found.

GET /media/devices/{deviceName}/doformat

Get a format key, used to format the device with a PUT request.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Format prepared.

The response is JSON.

Name	Type	Description
deviceName	string	Device name of media device to format.
key	string	The key required to format this device, provide to PUT /media/devices/{deviceName}/doformat to perform format of media device.

400 - Cannot format the device.

404 - Device not found.

PUT /media/devices/{deviceName}/doformat

Perform a format of the specified media device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by "deviceName" member of GET /media/workingset or GET /media/active.

Name	Type	Description
key	string	The key used to format this device, retrieved from prepare format media request GET /media/devices/{deviceName}/doformat. Format key provided cannot be reused after successful format.
filesystem	string	Filesystem to format to. Supported filesystems can be retrieved with GET /media/devices/doFormatSupportedFilesystems.
volume	string	Volume name to set for the disk after format.

Response

204 - Format successful.

400 - Cannot format the device, invalid filesystem or key.

404 - Device not found.

Slate Control API

API to manage digital slate data.

GET /slates/nextClip

Retrieve the digital slate for the next clip.

Response

200 - Returns the slate data for the next clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	

Name	Type	Description
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Slate data is not available.

PUT /slates/nextClip

Update the slate data for the next clip.

Parameters

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

POST /slates/nextClip/resetProjectData

Reset the project data for the next clip's slate.

Response

200 - Project data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetProjectData

Reset the project data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Project data reset successfully.

POST /slates/nextClip/resetLensData

Reset the lens data for the next clip's slate.

Response

200 - Lens data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetLensData

Reset the lens data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Lens data reset successfully.

GET /slates/clips/{deviceName}/{path}

Retrieve slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Returns the slate data for the specified clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

404 - Clip not found.

PUT /slates/clips/{deviceName}/{path}

Update the slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.

Name	Type	Description
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

Preset Control API

API For controlling the presets on Blackmagic Design products

GET /presets

Get the list of the presets on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presets	array	List of the presets on the camera (.cset files)
presets[i]	string	

POST /presets

Send a preset file to the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presetAdded	string	Name of the preset uploaded (without .cset extension)

400 - Bad request - missing Content-Disposition header or filename

GET /presets/active

Get the currently active preset on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

PUT /presets/active

Set the active preset on the camera

Parameters

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

Response

204 - No Content

404 - Preset file not found

PARAMETERS /presets/{presetName}

GET /presets/{presetName}

Download the preset file

Response

200 - OK

The response is a binary file.

404 - File does not exist

PUT /presets/{presetName}

Save current camera state as a preset

Response

204 - No Content

DELETE /presets/{presetName}

Delete a preset from the camera

Response

204 - No Content

404 - Preset file not found

Audio Control API

API For controlling audio on Blackmagic Design Cameras

GET /audio/channels

Get the total number of audio channels available

Response

200 - Returns the total number of channels

The response is JSON.

Name	Type	Description
channels	integer	Total number of audio channels available

GET /audio/supportedInputs

Get the list of supported audio inputs

Response

200 - List of all supported audio inputs

The response is JSON.

Name	Type	Description
	array	
[i]	string	A supported audio input

GET /audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently selected input

The response is JSON.

Name	Type	Description
input	string	Audio input source and type

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/input

Set the audio input for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
input	string	Audio input source and type

Response

204 - No Content

400 - Invalid audio input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/input/description

Get the description of the current input of the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Description of the current input of the selected channel

The response is JSON.

Name	Type	Description
description	object	
description.gainRange	object	
description.gainRange.Min	number	The minimum gain value in dB
description.gainRange.Max	number	The maximum gain value in dB
description.capabilities	object	
description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter
description.capabilities.Padding	object	
description.capabilities.Padding. available	boolean	Input supports setting of padding
description.capabilities.Padding. forced	boolean	Padding is forced to be set for the input
description.capabilities.Padding. value	number	Value of the padding in dB

404 - Channel does not exist

GET /audio/channel/{channelIndex}/supportedInputs

Get the list of supported inputs and their availability to switch to for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its supported inputs are being queried. (Channels index from 0)

Response

200 - The list of supported inputs

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

404 - Channel does not exist

GET /audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set level for the selected channel

The response is JSON.

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/level

Set the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

Response

204 - No Content

400 - Invalid input or value out of range

404 - Channel does not exist

GET /audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set phantom power for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Phantom power enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/phantomPower

Set the audio phantom power for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Phantom power enabled state

Response

204 - No Content

400 - Phantom power is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set padding for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Padding enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/padding

Set the audio input padding for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Padding enabled state

Response

204 - No Content

400 - Padding is not supported or is forced for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set low cut filter for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Low cut filter enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/lowCutFilter

Set the audio input low cut filter for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Low cut filter enabled state

Response

204 - No Content

400 - Low cut filter is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set availability for the selected channel

The response is JSON.

Name	Type	Description
available	boolean	Whether the input is currently available

404 - Channel does not exist

Lens Control API

API For controlling the lens on Blackmagic Design products

GET /lens/iris

Get lens' aperture

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number

PUT /lens/iris

Set lens' aperture

Parameters

Name	Type	Description
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number
adjustmentStep	integer	Signed value for relative aperture adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens iris is not controllable or is controlled by auto exposure

GET /lens/zoom

Get lens' zoom

Response

200 - OK

The response is JSON.

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value

PUT /lens/zoom

Set lens' zoom

Parameters

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value
adjustmentFocalLength	integer	Signed value for relative focal length adjustment
adjustmentNormalised	number	Signed normalized value for relative zoom adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens zoom is not controllable

GET /lens/focus

Get lens' focus

Response

200 - OK

The response is JSON.

Name	Type	Description
normalised	number	Normalised value

PUT /lens/focus

Set lens' focus

Parameters

Name	Type	Description
normalised	number	Normalised value
focusDistance	integer	Focus distance value

Response

204 - No Content

400 - Bad Request if out of range value is provided

PUT /lens/focus/doAutoFocus

Perform auto focus

Parameters

Name	Type	Description
position	object	
position.x	number	Normalized x coordinate for autofocus ROI
position.y	number	Normalized y coordinate for autofocus ROI

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens focus is not controllable

GET /lens/opticalImageStabilization

Get optical image stabilization status

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Whether optical image stabilization is enabled

501 - Not Implemented if optical image stabilization is not supported on this product

PUT /lens/opticalImageStabilization

Enable or disable optical image stabilization

Parameters

Name	Type	Description
enabled	boolean	Enable or disable optical image stabilization

Response

204 - No Content

501 - Not Implemented if optical image stabilization is not supported on this product

GET /lens/iris/description

Get detailed description of lens' iris capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the iris can be controlled
apertureStop	object	
apertureStop.min	number	Minimum aperture stop
apertureStop.max	number	Maximum aperture stop

GET /lens/zoom/description

Get detailed description of lens' zoom capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the zoom can be controlled
focalLength	object	
focalLength.adjustable	boolean	If focal length is adjustable
focalLength.min	integer	Minimum focal length
focalLength.max	integer	Maximum focal length

GET /lens/focus/description

Get detailed description of lens' focus capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the focus can be controlled
focusDistance	object	
focusDistance.adjustable	boolean	If focus distance is adjustable
focusDistance.min	number	Minimum focus distance
focusDistance.max	number	Maximum focus distance

Video Control API

API For controlling the video on Blackmagic Design products

GET /video/iso

Get current ISO

Response

200 - OK

The response is JSON.

Name	Type	Description
iso	integer	Current ISO value

PUT /video/iso

Set current ISO

Parameters

Name	Type	Description
iso	integer	ISO value to set

Response

204 - No Content

403 - ISO cannot be changed in the current state

GET /video/supportedISOs

Get the list of supported ISO settings

Response

200 - List of supported ISO values

The response is JSON.

Name	Type	Description
supportedISOs	array	Array of supported ISO values
supportedISOs[i]	integer	

GET /video/gain

Get current gain value in decibels

Response

200 - OK

The response is JSON.

Name	Type	Description
gain	integer	Current gain value in decibels

PUT /video/gain

Set current gain value

Parameters

Name	Type	Description
gain	integer	Gain value in decibels to set

Response

204 - No Content

403 - Gain cannot be changed in the current state

GET /video/supportedGains

Get the list of supported gain settings in decibels

Response

200 - List of supported gain values in decibels

The response is JSON.

Name	Type	Description
supportedGains	array	Array of supported gain values in decibels
supportedGains[i]	integer	

GET /video/whiteBalance

Get current white balance

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	integer	Current white balance

PUT /video/whiteBalance

Set current white balance

Parameters

Name	Type	Description
whiteBalance	integer	White balance to set

Response

204 - No Content

400 - Invalid white balance temperature

GET /video/whiteBalance/description

Get white balance range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	object	
whiteBalance.min	integer	Minimum color temperature
whiteBalance.max	integer	Maximum color temperature

PUT /video/whiteBalance/doAuto

Set current white balance automatically

Response

204 - No Content

GET /video/whiteBalanceTint

Get white balance tint

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	integer	Current white balance tint

PUT /video/whiteBalanceTint

Set white balance tint

Parameters

Name	Type	Description
whiteBalanceTint	integer	White balance tint to set

Response

204 - No Content

400 - Invalid white balance tint

GET /video/whiteBalanceTint/description

Get white balance tint range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	object	
whiteBalanceTint.min	integer	Minimum white balance tint
whiteBalanceTint.max	integer	Maximum white balance tint

GET /video/ndFilter

Get ND filter stop

Response

200 - OK

The response is JSON.

Name	Type	Description
stop	number	Current filter power (fStop)

501 - Not implemented for this device

PUT /video/ndFilter

Set ND filter stop

Parameters

Name	Type	Description
stop	number	Filter power (fStop) to set

Response

204 - No Content

400 - Invalid ND filter stop

501 - Not implemented for this device

GET /video/supportedNDFilters

Get the list of available ND filter stops

Response

200 - List of available ND filter stops

The response is JSON.

Name	Type	Description
supportedStops	array	Array of available ND filter stops
supportedStops[i]	number	

501 - Not implemented for this device

GET /video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

Response

200 - List of supported display modes

The response is JSON.

Name	Type	Description
supportedDisplayModes	array	Array of supported display modes
supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

GET /video/ndFilter/displayMode

Get ND filter display mode on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

PUT /video/ndFilter/displayMode

Set ND filter display mode on the camera

Parameters

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

Response

204 - No Content

400 - Invalid display mode for ND filter

501 - Not implemented for this device

GET /video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

Response

200 - Indicates if ND filter is selectable

The response is JSON.

Name	Type	Description
selectable	boolean	True if ND filter adjustments are selectable via a slider

501 - Not implemented for this device

GET /video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

PUT /video/shutter

Set current shutter

Parameters

Name	Type	Description
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

Response

204 - No Content

GET /video/shutter/measurement

Get the current shutter measurement mode

Response

200 - OK

The response is JSON.

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

PUT /video/shutter/measurement

Set the shutter measurement mode

Parameters

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

Response

204 - No Content

400 - Invalid shutter measurement

GET /video/supportedShutters

Get supported shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of supported shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of supported shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of flicker-free shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of flicker-free shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/autoExposure

Get current auto exposure mode

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

PUT /video/autoExposure

Set auto exposure

Parameters

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

Response

204 - No Content

400 - Failed to set auto exposure mode

GET /video/detailSharpening

Get the current state of detail sharpening

Response

200 - Current detail sharpening state

The response is JSON.

Name	Type	Description
enabled	boolean	Whether detail sharpening is enabled

501 - Not implemented for this device

PUT /video/detailSharpening

Enable or disable detail sharpening

Parameters

Name	Type	Description
enabled	boolean	Enable or disable detail sharpening

Response

204 - Detail sharpening state updated

501 - Not implemented for this device

GET /video/detailSharpeningLevel

Get the current detail sharpening level

Response

200 - Current detail sharpening level

The response is JSON.

Name	Type	Description
level	string	Current detail sharpening level Possible values are: Low, Medium, High.

501 - Not implemented for this device

PUT /video/detailSharpeningLevel

Set the detail sharpening level

Parameters

Name	Type	Description
level	string	Desired level of detail sharpening Possible values are: Low, Medium, High.

Response

204 - Detail sharpening level updated

400 - Invalid detail sharpening level

501 - Not implemented for this device

Camera Control API

API For controlling the Camera specific features on Blackmagic Design products

GET /camera/colorBars

Get the status of color bars display

Response

200 - Returns the current status of color bars

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the color bars are currently enabled

PUT /camera/colorBars

Set the status of color bars display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the color bars

Response

204 - Color bars status updated successfully

GET /camera/programFeedDisplay

Get the status of program feed display

Response

200 - Returns the current status of program feed display

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the program feed display is currently enabled

PUT /camera/programFeedDisplay

Set the status of program feed display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the program feed display

Response

204 - Program feed display status updated successfully

GET /camera/tallyStatus

Get the tally status of the camera

Response

200 - Returns the current tally status of the camera

The response is JSON.

Name	Type	Description
status	string	Current tally status of the camera Possible values are: None, Preview, Program.

GET /camera/power

Get the power status of the camera

Response

200 - Returns the current power status

The response is JSON.

Name	Type	Description
source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
batteries	array	List of batteries currently connected to the camera
batteries[i]	object	
batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
batteries[i].chargeRemainingPercent	integer	Remaining battery charge percentage
batteries[i].statusFlags	array	List of battery status flags
batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

GET /camera/power/displayMode

Get the power display mode of the camera

Response

200 - Returns the current power display mode

The response is JSON.

Name	Type	Description
mode	string	Current power display mode Possible values are: Percentage, Voltage.

PUT /camera/power/displayMode

Set the power display mode of the camera

Parameters

Name	Type	Description
mode	string	Power display mode to set Possible values are: Percentage, Voltage.

Response

204 - Power display mode updated successfully

400 - Invalid power display mode

GET /camera/timingReferenceLock

Get the timing reference lock status

Response

200 - Returns the timing reference lock status

The response is JSON.

Name	Type	Description
locked	boolean	Indicates if timing reference is locked

Immersive Control API

API for controlling immersive camera settings on Blackmagic Design cameras

GET /immersive/display/{displayName}/eye

Get the current eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to query (from /monitoring/display endpoint)

Response

200 - OK

The response is JSON.

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

400 - Invalid display name format

404 - Display not found

422 - Failed to get eye view

PUT /immersive/display/{displayName}/eye

Set the eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to control (from /monitoring/display endpoint)

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

Response

204 - No Content

400 - Invalid input or display name format

404 - Display not found

422 - Failed to set eye view

Color Correction Control API

API For controlling the color correction on Blackmagic Design products based on DaVinci Resolve Color Corrector

GET /colorCorrection/lift

Get color correction lift

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

PUT /colorCorrection/lift

Set color correction lift

Parameters

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gamma

Get color correction gamma

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

PUT /colorCorrection/gamma

Set color correction gamma

Parameters

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gain

Get color correction gain

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

PUT /colorCorrection/gain

Set color correction gain

Parameters

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/offset

Get color correction offset

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

PUT /colorCorrection/offset

Set color correction offset

Parameters

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/contrast

Get color correction contrast

Response

200 - OK

The response is JSON.

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/contrast

Set color correction contrast

Parameters

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/color

Get color correction color properties

Response

200 - OK

The response is JSON.

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/color

Set color correction color properties

Parameters

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/lumaContribution

Get color correction luma contribution

Response

200 - OK

The response is JSON.

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

PUT /colorCorrection/lumaContribution

Set color correction luma contribution

Parameters

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

Response

204 - No Content

Notification websocket - 1.0.0

Service that notifies subscribers of device state changes.

messages

Subscribe (The messages from the server/device)

Websocket Opened Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: websocketOpened.
.type	string	Possible values are: event.

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*

Name	Type	Description
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Event Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: propertyValueChanged.

Name	Type	Description
.data.property	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.value	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.type	string	Possible values are: event.

Publish (The messages that user can send to the server/device)

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	

Name	Type	Description
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	

Name	Type	Description
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Device Properties

/media/workingset

The value JSON returned via the eventResponse when the `/media/workingset` property changes on the device:

Name	Type	Description
.size	integer	The fixed size of this device's working set.
.workingset	array	Array of devices within the working set. null if no device is present within the given working set slot.
.workingset[i]		

/media/active

The value JSON returned via the eventResponse when the `/media/active` property changes on the device:

Name	Type	Description
.workingsetIndex	integer	Working set index of the active media device.
.deviceName	string	Internal device name of this media device.

/system

The value JSON returned via the eventResponse when the `/system` property changes on the device:

Name	Type	Description
.codecFormat	object	Codec format configuration.
.codecFormat.codec	string	Codec format serialised as a string.
.codecFormat.container	string	Multimedia container format.
.videoFormat	object	Currently selected video format.
.videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.videoFormat.height	number	Height dimension of video format.
.videoFormat.width	number	Width dimension of video format.
.videoFormat.interlaced	boolean	Is the display format interlaced?.
.videoFormat.name	string	Video format serialised as a string.

/system/codecFormat

Codec format configuration.

The value JSON returned via the eventResponse when the `/system/codecFormat` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.container	string	Multimedia container format.

/system/videoFormat

Currently selected video format.

The value JSON returned via the eventResponse when the `/system/videoFormat` property changes on the device:

Name	Type	Description
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.height	number	Height dimension of video format.
.width	number	Width dimension of video format.
.interlaced	boolean	Is the display format interlaced?.
.name	string	Video format serialised as a string.

/system/format

The value JSON returned via the eventResponse when the `/system/format` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.maxOffSpeedFrameRate	number	
.minOffSpeedFrameRate	number	
.offSpeedEnabled	boolean	
.offSpeedFrameRate	number	
.recordResolution	object	
.recordResolution.height	number	Height of the resolution.
.recordResolution.width	number	Width of the resolution.
.sensorResolution	object	
.sensorResolution.height	number	Height of the resolution.
.sensorResolution.width	number	Width of the resolution.

/system/supportedFormats

The value JSON returned via the eventResponse when the `/system/supportedFormats` property changes on the device:

Name	Type	Description
.supportedFormats	array	
.supportedFormats[i]	object	
.supportedFormats[i].codecs	array	
.supportedFormats[i].codecs[i]	string	
.supportedFormats[i].frameRates	array	
.supportedFormats[i].frameRates[i]	string	Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.supportedFormats[i].maxOffSpeedFrameRate	number	
.supportedFormats[i].minOffSpeedFrameRate	number	
.supportedFormats[i].recordResolution	object	
.supportedFormats[i].recordResolution.height	number	Height of the resolution.
.supportedFormats[i].recordResolution.width	number	Width of the resolution.
.supportedFormats[i].sensorResolution	object	
.supportedFormats[i].sensorResolution.height	number	Height of the resolution.
.supportedFormats[i].sensorResolution.width	number	Width of the resolution.

/timelines/0

The value JSON returned via the eventResponse when the `/timelines/0` property changes on the device:

Name	Type	Description
.clips	array	
.clips[i]	object	Timeline clip.
.clips[i].clipUniqueId	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
.clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
.clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
.clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
.clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).

Name	Type	Description
.clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
.clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

/transports/0

The value JSON returned via the eventResponse when the `/transports/0` property changes on the device:

Name	Type	Description
.mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

/transports/0/stop

true when transport mode is InputPreview or when in Output mode and speed is 0.

The value JSON returned via the eventResponse when the `/transports/0/stop` property changes on the device:

Name	Type	Description
	boolean	true when transport mode is InputPreview or when in Output mode and speed is 0.

/transports/0/play

True when transport is in Output mode and speed is non-zero.

The value JSON returned via the eventResponse when the /transports/0/play property changes on the device:

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

/transports/0/playback

The value JSON returned via the eventResponse when the `/transports/0/playback` property changes on the device:

Name	Type	Description
.type	string	Possible values are: Play, Jog, Shuttle, Var.
.loop	boolean	When true playback loops from the end of the timeline to the beginning of the timeline.
.singleClip	boolean	When true playback loops from the end of the current clip to the beginning of the current clip.
.speed	number	Playback speed, 1.0 for normal forward playback
.position	integer	Playback position on the timeline in units of video frames.

/transports/0/record

The value JSON returned via the eventResponse when the `/transports/0/record` property changes on the device:

Name	Type	Description
.recording	boolean	Is transport in Input Record mode.

/transports/0/timecode

The value JSON returned via the eventResponse when the `/transports/0/timecode` property changes on the device:

Name	Type	Description
.display	string	The display timecode serialised as a string.
.timeline	string	The timeline timecode serialised as a string.

/transports/0/timecode/source

The value JSON returned via the eventResponse when the `/transports/0/timecode/source` property changes on the device:

Name	Type	Description
.timecode	string	Possible values are: Timeline, Clip.

/transports/0/clipIndex

The value JSON returned via the eventResponse when the `/transports/0/clipIndex` property changes on the device:

Name	Type	Description
.clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

/slates/nextClip

The value JSON returned via the eventResponse when the `/slates/nextClip` property changes on the device:

Name	Type	Description
.clip	object	
.clip.clipName	string	
.clip.reel	integer	
.clip.scene	string	
.clip.sceneLocation	string	Possible values are: Interior, Exterior.
.clip.sceneTime	string	Possible values are: Day, Night.
.clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
.clip.take	integer	
.clip.takeType	string	Possible values are: None, PU, VFX, SER.
.clip.goodTake	boolean	
.lens	object	
.lens.lensType	string	
.lens.iris	string	

Name	Type	Description
.lens.focalLength	string	
.lens.distance	string	
.lens.filter	string	
.project	object	
.project.projectName	string	
.project.director	string	
.project.camera	string	
.project.cameraOperator	string	

/monitoring/{displayName}/cleanFeed

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/cleanFeed` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/displayLUT

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/displayLUT` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/zebra

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/zebra` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/focusAssist` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGuide

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGuide` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGrids` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/safeArea

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/safeArea` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/falseColor

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/falseColor` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/focusAssist` property changes on the device:

Name	Type	Description
.mode	string	Possible values are: Peak, ColoredLines.
.color	string	Possible values are: Red, Green, Blue, White, Black.
.intensity	integer	

/monitoring/frameGuideRatio

The value JSON returned via the eventResponse when the `/monitoring/frameGuideRatio` property changes on the device:

Name	Type	Description
.ratio	string	

/monitoring/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/frameGrids` property changes on the device:

Name	Type	Description
.frameGrids	array	
.frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

/monitoring/safeAreaPercent

The value JSON returned via the eventResponse when the `/monitoring/safeAreaPercent` property changes on the device:

Name	Type	Description
.percent	integer	Safe area coverage percentage.

/audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input` property changes on the device:

Name	Type	Description
.input	string	Audio input source and type

/audio/channel/{channelIndex}/supportedInputs

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/supportedInputs` property changes on the device:

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

/audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/level` property changes on the device:

Name	Type	Description
.gain	number	Gain value in dB
.normalised	number	Normalised level value between 0.0 and 1.0

/audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/phantomPower` property changes on the device:

Name	Type	Description
.enabled	boolean	Phantom power enabled state

/audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/padding` property changes on the device:

Name	Type	Description
.enabled	boolean	Padding enabled state

/audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/lowCutFilter` property changes on the device:

Name	Type	Description
.enabled	boolean	Low cut filter enabled state

/audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/available` property changes on the device:

Name	Type	Description
.available	boolean	Whether the input is currently available

/audio/channel/{channelIndex}/input/description

Description of the current input of the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input/description` property changes on the device:

Name	Type	Description
.description	object	
.description.gainRange	object	
.description.gainRange.Min	number	The minimum gain value in dB
.description.gainRange.Max	number	The maximum gain value in dB
.description.capabilities	object	
.description.capabilities.PhantomPower	boolean	Input supports setting of phantom power
.description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter

Name	Type	Description
.description.capabilities.Padding	object	
.description.capabilities.Padding.available	boolean	Input supports setting of padding
.description.capabilities.Padding.forced	boolean	Padding is forced to be set for the input
.description.capabilities.Padding.value	number	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.

/colorCorrection/lift

Get color correction lift

The value JSON returned via the eventResponse when the `/colorCorrection/lift` property changes on the device:

Name	Type	Description
.red	number	Red lift component
.green	number	Green lift component
.blue	number	Blue lift component
.luma	number	Luma lift component

/colorCorrection/gamma

Get color correction gamma

The value JSON returned via the eventResponse when the `/colorCorrection/gamma` property changes on the device:

Name	Type	Description
.red	number	Red gamma component
.green	number	Green gamma component
.blue	number	Blue gamma component
.luma	number	Luma gamma component

/colorCorrection/gain

Get color correction gain

The value JSON returned via the eventResponse when the `/colorCorrection/gain` property changes on the device:

Name	Type	Description
.red	number	Red gain component
.green	number	Green gain component
.blue	number	Blue gain component
.luma	number	Luma gain component

/colorCorrection/offset

Get color correction offset

The value JSON returned via the eventResponse when the `/colorCorrection/offset` property changes on the device:

Name	Type	Description
.red	number	Red offset component
.green	number	Green offset component
.blue	number	Blue offset component
.luma	number	Luma offset component

/colorCorrection/contrast

Get color correction contrast

The value JSON returned via the eventResponse when the `/colorCorrection/contrast` property changes on the device:

Name	Type	Description
.pivot	number	Contrast pivot point
.adjust	number	Contrast adjustment

/colorCorrection/color

Get color correction color properties

The value JSON returned via the eventResponse when the `/colorCorrection/color` property changes on the device:

Name	Type	Description
.hue	number	Color hue adjustment
.saturation	number	Color saturation adjustment

/colorCorrection/lumaContribution

Get color correction luma contribution

The value JSON returned via the eventResponse when the `/colorCorrection/lumaContribution` property changes on the device:

Name	Type	Description
.lumaContribution	number	Luma contribution value

/lens/iris

Get lens' aperture

The value JSON returned via the eventResponse when the `/lens/iris` property changes on the device:

Name	Type	Description
.continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
.apertureStop	number	Aperture stop value
.normalised	number	Normalised value
.apertureNumber	integer	Aperture number

/lens/iris/description

Get detailed description of lens' iris capabilities

The value JSON returned via the eventResponse when the `/lens/iris/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the iris can be controlled
.apertureStop	object	
.apertureStop.min	number	Minimum aperture stop
.apertureStop.max	number	Maximum aperture stop

/lens/focus

Get lens' focus

The value JSON returned via the eventResponse when the `/lens/focus` property changes on the device:

Name	Type	Description
.normalised	number	Normalised value

/lens/focus/description

Get detailed description of lens' focus capabilities

The value JSON returned via the eventResponse when the `/lens/focus/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the focus can be controlled
.focusDistance	object	
.focusDistance.adjustable	boolean	If focus distance is adjustable
.focusDistance.min	number	Minimum focus distance
.focusDistance.max	number	Maximum focus distance

/lens/zoom

Get lens' zoom

The value JSON returned via the eventResponse when the `/lens/zoom` property changes on the device:

Name	Type	Description
.focalLength	integer	Focal length in mm
.normalised	number	Normalised value

/lens/zoom/description

Get detailed description of lens' zoom capabilities

The value JSON returned via the eventResponse when the `/lens/zoom/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the zoom can be controlled
.focalLength	object	
.focalLength.adjustable	boolean	If focal length is adjustable
.focalLength.min	integer	Minimum focal length
.focalLength.max	integer	Maximum focal length

/presets

Get the list of the presets on the camera

The value JSON returned via the eventResponse when the `/presets` property changes on the device:

Name	Type	Description
.presets	array	List of the presets on the camera (.cset files)
.presets[i]	string	

/presets/active

Get the currently active preset on the camera

The value JSON returned via the eventResponse when the `/presets/active` property changes on the device:

Name	Type	Description
.preset	string	Name of the active preset (with .cset extension, or 'default')

/camera/colorBars

Get the status of color bars display

The value JSON returned via the eventResponse when the `/camera/colorBars` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the color bars are currently enabled

/camera/programFeedDisplay

Get the status of program feed display

The value JSON returned via the eventResponse when the `/camera/programFeedDisplay` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the program feed display is currently enabled

/camera/tallyStatus

Get the tally status of the camera

The value JSON returned via the eventResponse when the `/camera/tallyStatus` property changes on the device:

Name	Type	Description
.status	string	Current tally status of the camera Possible values are: None, Preview, Program.

/camera/power

Get the power status of the camera

The value JSON returned via the eventResponse when the `/camera/power` property changes on the device:

Name	Type	Description
.source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
.milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
.batteries	array	
.batteries[i]	object	
.batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
.batteries[i]. chargeRemainingPercent	integer	Remaining battery charge percentage
.batteries[i].statusFlags	array	List of battery status flags
.batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

/camera/power/displayMode

Get the power display mode of the camera

The value JSON returned via the eventResponse when the `/camera/power/displayMode` property changes on the device:

Name	Type	Description
.mode	string	Current power display mode Possible values are: Percentage, Voltage.

/camera/timingReferenceLock

Get the timing reference lock status

The value JSON returned via the eventResponse when the `/camera/timingReferenceLock` property changes on the device:

Name	Type	Description
.locked	boolean	Indicates if timing reference is locked

/video/iso

Get current ISO

The value JSON returned via the eventResponse when the `/video/iso` property changes on the device:

Name	Type	Description
.iso	integer	Current ISO value

/video/supportedISOs

Get the list of supported ISO settings

The value JSON returned via the eventResponse when the `/video/supportedISOs` property changes on the device:

Name	Type	Description
.supportedISOs	array	Array of supported ISO values
.supportedISOs[i]	integer	

/video/gain

Get current gain value in decibels

The value JSON returned via the eventResponse when the `/video/gain` property changes on the device:

Name	Type	Description
.gain	integer	Current gain value in decibels

/video/supportedGains

Get the list of supported gain settings in decibels

The value JSON returned via the eventResponse when the `/video/supportedGains` property changes on the device:

Name	Type	Description
.supportedGains	array	Array of supported gain values in decibels
.supportedGains[i]	integer	

/video/whiteBalance

Get current white balance

The value JSON returned via the eventResponse when the `/video/whiteBalance` property changes on the device:

Name	Type	Description
.whiteBalance	integer	Current white balance

/video/whiteBalance/description

Get white balance range

The value JSON returned via the eventResponse when the `/video/whiteBalance/description` property changes on the device:

Name	Type	Description
.whiteBalance	object	
.whiteBalance.min	integer	Minimum color temperature
.whiteBalance.max	integer	Maximum color temperature

/video/whiteBalanceTint

Get white balance tint

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint` property changes on the device:

Name	Type	Description
.whiteBalanceTint	integer	Current white balance tint

/video/whiteBalanceTint/description

Get white balance tint range

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint/description` property changes on the device:

Name	Type	Description
.whiteBalanceTint	object	
.whiteBalanceTint.min	integer	Minimum white balance tint
.whiteBalanceTint.max	integer	Maximum white balance tint

/video/ndFilter

Get ND filter stop

The value JSON returned via the eventResponse when the `/video/ndFilter` property changes on the device:

Name	Type	Description
.stop	number	Current filter power (fStop)

/video/supportedNDFilters

Get the list of available ND filter stops

The value JSON returned via the eventResponse when the `/video/supportedNDFilters` property changes on the device:

Name	Type	Description
.supportedStops	array	Array of available ND filter stops
.supportedStops[i]	number	

/video/ndFilter/displayMode

Get ND filter display mode on the camera

The value JSON returned via the eventResponse when the `/video/ndFilter/displayMode` property changes on the device:

Name	Type	Description
.displayMode	string	ND filter display mode Possible values are: Stop, Number, Fraction.

/video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

The value JSON returned via the eventResponse when the `/video/supportedNDFilterDisplayModes` property changes on the device:

Name	Type	Description
.supportedDisplayModes	array	Array of supported display modes
.supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

/video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

The value JSON returned via the eventResponse when the `/video/ndFilterSelectable` property changes on the device:

Name	Type	Description
.selectable	boolean	True if ND filter adjustments are selectable via a slider

/video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

The value JSON returned via the eventResponse when the `/video/shutter` property changes on the device:

Name	Type	Description
.continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
.shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
.shutterAngle	number	Shutter angle

/video/shutter/measurement

Get the current shutter measurement mode

The value JSON returned via the eventResponse when the `/video/shutter/measurement` property changes on the device:

Name	Type	Description
.measurement	string	Current shutter measurement mode Possible values are: ShutterAngle, ShutterSpeed.

/video/supportedShutters

Get supported shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/supportedShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of supported shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array of flicker-free shutter speeds
.shutterSpeeds[i]	integer	

/video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/flickerFreeShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of flicker-free shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array
.shutterSpeeds[i]	integer	

/video/autoExposure

Get current auto exposure mode

The value JSON returned via the eventResponse when the `/video/autoExposure` property changes on the device:

Name	Type	Description
.mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
.type	string	Comma-separated list of device types in the auto exposure stack

/video/detailSharpening

Get the current state of detail sharpening

The value JSON returned via the eventResponse when the `/video/detailSharpening` property changes on the device:

Name	Type	Description
.enabled	boolean	Whether detail sharpening is enabled

/video/detailSharpeningLevel

Get the current detail sharpening level

The value JSON returned via the eventResponse when the `/video/detailSharpeningLevel` property changes on the device:

Name	Type	Description
.level	string	Current detail sharpening level of supported shutter speeds Possible values are: Low, Medium, High.

ヘルプ

すぐに情報が必要な方は、Blackmagic Designオンラインサポートページで、Blackmagicカメラの最新サポート情報を確認できます。

Blackmagic Designオンラインサポートページ

最新のマニュアル、ソフトウェア、サポートノートは、www.blackmagicdesign.com/jp/supportのBlackmagicサポートセンターで確認できます。

Blackmagic Designサポートへの連絡

サポートページで必要な情報を得られなかった場合は、サポートページの「メールを送信」ボタンを使用して、サポートのリクエストをメール送信してください。あるいは、サポートページの「お住まいの地域のサポートオフィス」をクリックして、最寄りのBlackmagic Designサポートオフィスに電話でお問い合わせください。

現在インストールされているソフトウェアのバージョンを確認する

コンピューターにインストールされているBlackmagic Camera Setup Utilityソフトウェアのバージョンを確認するには、「About Blackmagic Camera Setup Utility」ウィンドウを開きます。

- Macでは、アプリケーションフォルダーから「Blackmagic Camera Setup Utility」を開きます。アプリケーションメニューから「About Blackmagic Camera Setup Utility」を選択し、バージョンを確認します。
- Windowsでは、スタートメニューまたはスタート画面から「Blackmagic Camera Setup Utility」を開きます。ヘルプメニューをクリックして「About Blackmagic Camera Setup Utility」を選択し、バージョンを確認します。

最新のソフトウェアアップデートを入手する

コンピューターにインストールされているBlackmagic Camera Utilityのバージョンを確認した後、Blackmagicサポートセンター（www.blackmagicdesign.com/jp/support）で最新のソフトウェアアップデートをチェックしてください。常に最新のソフトウェアを使用することを推奨しますが、重要なプロジェクトの実行中は、ソフトウェアのアップデートは行わない方がよいでしょう。

規制に関する警告



欧州連合内での電気機器および電子機器の廃棄処分

製品に記載されている記号は、当該の機器を他の廃棄物と共に処分してはならないことを示しています。機器を廃棄するには、必ずリサイクルのために指定の回収場所に引き渡してください。機器の廃棄において個別回収とリサイクルが行われることで、天然資源の保護につながり、健康と環境を守る方法でリサイクルが確実に行われるようになります。廃棄する機器のリサイクルのための回収場所に関しては、お住まいの地方自治体のリサイクル部門、または製品を購入した販売業者にご連絡ください。



この機器は、FCC規定の第15部に準拠し、クラスAデジタル機器の制限に適合していることが確認されています。これらの制限は、商用環境で機器を使用している場合に有害な干渉に対する妥当な保護を提供するためのものです。この機器は無線周波エネルギーを生成、使用、放出する可能性があります。また、指示に従ってインストールおよび使用しない場合、無線通信に有害な干渉を引き起こす恐れがあります。住宅地域で当製品を使用すると有害な干渉を引き起こす可能性があり、その場合はユーザーが自己責任で干渉に対処する必要があります。

本機の動作は次の2つを条件とします：

1. 本機は、有害な干渉を起こさない。
2. 本機は希望しない動作を発生しかねない干渉を含む、いかなる受信干渉も受け入れる必要がある。



R-R-BMD-20250213001



ISED Canadaステートメント

本機は、カナダのクラスAデジタル機器の規格に準拠しています。

本機のいかなる改造、あるいは目的の用途以外での使用は、これらの規格への順守を無効にすることがあります。

HDMIインターフェースへの接続は、必ず高品質のシールドHDMIケーブルを使用する必要があります。

本機は、商用環境で目的の用途に順守した使用においてテストを行なっています。非商用環境で使用された場合、無線妨害を引き起こす可能性があります。

安全情報

Blackmagic URSA Cine用に同梱されたAC - 24V DC電源は、必ず保護接地のあるコンセントに接続してください。

Blackmagic URSA Cineの内部には、ユーザーが保守できる部品はありません。サービスに関しては、お近くのBlackmagic Designのサービスセンターにお問い合わせください。

入力電源コネクターを外付けDCバッテリーに接続する際は、供給配線に電流制限またはヒューズが必ず付いている必要があります。

使用するケーブルは、VW-1のマーキングが付いているか、関連するIEC 60332またはIEC 60695に準拠している必要があります。

当カメラのDCソースは24Vを使用することを推奨します。12VのDCソースを使用する場合、14Aの電流に十分に対応しているケーブルを使用してください。

本機には、Bマウントバッテリープレートが搭載されています。24V (28.8V) バッテリーまたはデュアルボルテージ・バッテリーを使用することを推奨します。

12V (14.4V) バッテリーでは、バッテリーのマニュアルまたはマーキングで、最大連続放電電流を確認してください。最低12Aであるものを使用することを推奨します。

カリフォルニア州ステートメント

この製品のユーザーは、プラスチック部品内の微量の多臭素化ビフェニルなどの化学物質にさらされる可能性があります。カリフォルニア州は、多臭素化ビフェニルは発がん性があり、先天異常や生殖機能へ危害を及ぼす物質であると認識しています。

詳細は、以下のウェブサイトをご確認ください。www.P65Warnings.ca.gov

ヨーロッパオフィス

Blackmagic Design Europe B.V.
Rijnlanderweg 766, Unit D
2132 NM Hoofddorp
NL

Blackmagic URSA Cine

日が当たる条件では、カメラやリチウム電池が長期にわたって太陽光にさらされないように、カメラに日よけを使用したり、日陰に移すなどの考慮をしてください。リチウム電池は、あらゆる熱源に近づけないでください。

24Vの「RS」出力コネクターは、サードパーティ製のアクセサリへの給電に適しています。消費電力が48W未満であることを確認してください。

保証

限定保証

Blackmagic Designは、お買い上げの日から12ヶ月間、本製品の部品および仕上がりについて瑕疵がないことを保証します。この保証期間内に製品に瑕疵が見つかった場合、Blackmagic Designは弊社の裁量において部品代および人件費無料で該当製品の修理、あるいは製品の交換のいずれかで対応いたします。

この保証に基づいたサービスを受ける際、お客様は必ず保証期限終了前にBlackmagic Designに瑕疵を通知し、保証サービスの手続きを行ってください。お客様の責任において不良品を梱包し、Blackmagic Designが指定するサポートセンターへ配送料前払で送付いただきますようお願い致します。理由の如何を問わず、Blackmagic Designへの製品返送のための配送料、保険、関税、税金、その他すべての費用はお客様の自己負担となります。

不適切な使用、または不十分なメンテナンスや取扱いによる不具合、故障、損傷に対しては、この保証は適用されません。Blackmagic Designはこの保証で、以下に関してサービス提供義務を負わないものとします。a) 製品のインストールや修理、サービスを行うBlackmagic Design販売代理人以外の者によって生じた損傷の修理、b) 不適切な使用や互換性のない機器への接続によって生じた損傷の修理、c) Blackmagic Designの部品や供給品ではない物を使用して生じたすべての損傷や故障の修理、d) 改造や他製品との統合により時間増加や製品の機能低下が生じた場合のサービス。

Blackmagic URSA Viewfinderを直射日光にさらすとレンズが拡大鏡と同じように作用し、ビューファインダーのディスプレイが損傷する恐れがあります。フレームガイドなど、静止画像や高コントラスト画像を長時間にわたって有機ELディスプレイに表示すると、残像や焼き付きが生じる原因となります。これらの現象を避けるため、ユーザーを検知するIRセンサーが何かで覆われていないことを確認し、カメラを長期間にわたって使用しない場合はビューファインダーを取り外してください。残像等の問題は製品保証に含まれません。

この保証はBLACKMAGIC DESIGNが保証するもので、明示または黙示を問わず他の保証すべてに代わるものです。BLACKMAGIC DESIGNとその販売社は、商品性と特定目的に対する適合性のあらゆる黙示保証を拒否します。製品保証期間は、BLACKMAGIC DESIGNの不良品の修理あるいは交換の責任が、お客様に提供される完全唯一の救済手段となります。BLACKMAGIC DESIGNは、特別に、間接的、偶発的、または結果的に生じる損害に対して、BLACKMAGIC DESIGNあるいは販売社がそのような損害の可能性についての事前通知を得ているか否かに関わらず、一切の責任を負いません。BLACKMAGIC DESIGNはお客様による機器のあらゆる不法使用に対して責任を負いません。BLACKMAGIC DESIGNは本製品の使用により生じるあらゆる損害に対して責任を負いません。使用者は自己の責任において本製品を使用するものとします。

© Copyright 2025 Blackmagic Design 著作権所有、無断複写・転載を禁じます。「Blackmagic Design」、「URSA」、「DeckLink」、「HDLink」、「Workgroup Videohub」、「Multibridge Pro」、「Multibridge Extreme」、「Intensity」、「Leading the creative video revolution」は、米国ならびにその他諸国での登録商標です。その他の企業名ならびに製品名全てはそれぞれ関連する会社の登録商標である可能性があります。

Bluetooth®のワードマークおよびロゴはBluetooth SIG, Inc.が所有する登録商標であり、Blackmagic Designはこれらのマークをライセンスに基づいて使用しています。その他の商標およびトレードネームは、それぞれの所有者に帰属します。

Octobre 2025

Guide de caméra

Blackmagicdesign

Blackmagic URSA Cine Immersive



Blackmagic URSA Cine Immersive

Contenu

Blackmagic URSA Cine Immersive 309
Qu'est-ce que le format Immersif ? 309
Apple Immersive Video 309
URSA Cine Immersive 310
Mise en route 311
Alimenter la caméra 311
Paramétrer les réglages d'enregistrement 312
Monitoring de l'image 314
Options de monitoring et affichage semi-transparent 314
Cadrer votre première prise 315
Hauteur de la caméra 315
Inclinaison de la caméra 316
Positionnement 317
Agentivité du spectateur 318
Mouvement de caméra 318
Tableaux des débits de données 319
Audio et timecode 320
Apple Spatial Audio Format 320
Ambisonique 320
Micros-cravates et micros miniatures 321
Micros à main 321
Connecter le timecode 321
Filmer 323
Exigences d'exposition 323
Vitesse d'obturation 323
ISO et plage dynamique 324
ND 325
Enregistrement Off speed 326
Histogramme 327
Options de l'écran LCD 327
Enregistrement 328
Lecture 329
Transférer les clips 330
Résumé 331
DaVinci Resolve Studio 332
Recommandations pour les stations de travail 332
Configurer le projet 333
Paramètres principaux 333
Monitoring vidéo 334
Mises à l'échelle 334
Gestion de la couleur 335
À propos du DaVinci Wide Gamut 336
Traitement du contenu Apple Immersive 337
Importation 337
Montage 339
Prémontage 339
Arrière-plans 339
Utiliser le viewer 341
Transitions 343
Apple Immersive Motion Data 344
Titres 345
Considérations créatives 346
Compositing VFX 347
Nœuds MediaOut 347
Renderer3D 348
Compositing avec l'Immersive Patcher 349
Étalonnage 350
Réduction de bruit 350
Étalonnage primaire 351
Corrections secondaires 351
Échelle 352
Palette 3D 352
Gestion des couleurs CST 354
Mappage de la timeline CST 354
Mappage CST pour le moniteur d'étalonnage 355
Mappage CST pour l'Apple Vision Pro 357
Audio 358
Apple Spatial Audio Format (ASAF) 358
Remise des projets 359
Ignorer les transitions 359
Streaming vers l'Apple Vision Pro 360
À propos de l'utilitaire Apple Immersive Video 360
Developer Information 361
Camera Control REST API 361
Livestream Control API 363
Clips Control API 367

Media Pool Control API 368
Monitoring Control API 371
Event Control API 380
System Control API 380
Transport Control API 386
Timeline Control API 390
Media Control API 393
Slate Control API 396
Preset Control API 402
Audio Control API 404
Lens Control API 409
Video Control API 413
Camera Control API 421
Immersive Control API 424
Color Correction Control API 425
Notification websocket - 1.0.0 430
messages 430
Assistance 454
Avis règlementaires 455
Informations de sécurité 456
Garantie 457

Blackmagic URSA Cine Immersive

Félicitations pour l'achat ou la location de votre Blackmagic URSA Cine Immersive.

La Blackmagic URSA Cine Immersive est conçue pour filmer des images en VR stéréoscopique à 180 degrés, que l'on appelle communément la vidéo immersive. Elle a été particulièrement optimisée pour filmer en Apple Immersive Video pour l'Apple Vision Pro, l'expérience immersive de la plus haute qualité actuellement disponible.

Ce guide d'introduction à la Blackmagic URSA Cine Immersive vous montrera comment filmer avec cette caméra, en se focalisant sur les opérations basiques de la caméra et sur l'unique format exportable, l'Apple Immersive Video. Comprendre le format vous aidera à mieux appréhender la caméra.

Dans les pages suivantes, vous trouverez aussi bien des conseils techniques que créatifs qui vous aideront à utiliser la caméra, mais aussi à apprécier les avantages du format immersif.

Qu'est-ce que le format Immersif ?

C'est un format de vidéo spatiale tridimensionnelle. D'autres technologies similaires existent depuis un certain temps. Pour comprendre ce qui rend le format immersif différent, vous trouverez ci-dessous un aperçu des autres technologies :

- **Le cinéma 3D traditionnel** Il permet aux spectateurs d'expérimenter la profondeur en capturant chaque scène à l'aide de deux caméras, avec une distance interaxiale qui recrée la distance entre les yeux humains. Tandis que les images stéréoscopiques permettent d'expérimenter la profondeur, la perspective est contrôlée par la composition du directeur de la photographie.
- **La vidéo sphérique** Elle combine les principes des caméras 3D et à 360 degrés pour permettre au public de regarder dans n'importe quelle direction tout en expérimentant la profondeur. Les images à 360 degrés nécessitent une très haute résolution pour veiller à ce que la petite partie de l'image que regarde le spectateur soit assez fidèle. Comme la physiologie humaine nous permet seulement de tourner notre tête d'une épaule à l'autre, afin de regarder derrière, le public est généralement debout ou assis sur une chaise pivotante.
- **La vidéo volumétrique** Elle permet au public d'expérimenter la profondeur et de regarder autour, mais aussi de bouger et de changer la perspective. Tandis que des matrices de caméras peuvent capturer la vidéo volumétrique, la réalité virtuelle ou la réalité augmentée utilise des rendus d'éléments 3D dynamiques et nécessite un environnement sécurisé pour les participants.

Le format immersif combine le meilleur des formats cinéma traditionnel et 3D interactif. C'est un format à fréquence d'images élevée stéréoscopique à 180 degrés haute résolution.

Apple Immersive Video

La Blackmagic URSA Cine Immersive est spécifiquement conçue pour filmer en Apple Immersive Video. Ce format est utilisé par l'Apple Vision Pro et supporte la vidéo stéréoscopique 3D haute résolution dans un champ de vision à 180 degrés.

L'Apple Immersive Video présente les caractéristiques suivantes :

- Lecture à 90 images par seconde.
- Peut être distribuée à des résolutions allant jusqu'à 4320 x 4320 par œil.
- Un système de métadonnées unique pour mapper l'image de la caméra originale à l'environnement à 180 degrés.
- Apple Spatial Audio Format pour fournir le son 3D.

L'Apple Immersive Video nécessite du matériel capable de lire de la vidéo haute résolution stéréoscopique à 180 degrés. L'Apple Vision Pro est actuellement le seul appareil qui supporte ce format.



Apple Vision Pro

Le casque comprend deux écrans Micro-Oled de 3660 x 3200 pixels avec un pas de pixel de 7,5 microns. Il inclut 13 caméras de suivi, un scanner LiDAR, des unités de mesure inertielle, et des capteurs de suivi du regard pour aligner l'imagerie avec le mouvement de la tête et des yeux.

Un processeur M2 avec une lecture disque CPU 8 cœurs et GPU 10 cœurs jusqu'à 100 images par seconde.

URSA Cine Immersive

La Blackmagic URSA Cine Immersive est conçue pour capturer du contenu Apple Immersive Video pour l'Apple Vision Pro.



Blackmagic URSA Cine Immersive

La URSA Cine Immersive utilise deux capteurs RGBW de 58 mégapixels appariés. Les capteurs sont presque carrés pour capturer un champ de vision à 180 degrés, horizontalement et verticalement. Chaque capteur est associé à un objectif fisheye à 210 degrés. Les centres des objectifs sont placés à 64mm les uns des autres.

Les filtres ND intégrés fournissent jusqu'à 8 diaphs de contrôle de la luminosité. L'ouverture de l'objectif est fixée à environ f4.5.

La caméra enregistre deux flux d'image 8K à 90 images par seconde. Le débit de données est bien supérieur que pour l'enregistrement standard en 4K 24p.

Mise en route

Pour commencer à utiliser votre Blackmagic URSA Cine Immersive, il suffit de charger votre caméra et de l'allumer. Le Blackmagic Media Module 8TB est chargé dans la baie de module de la caméra, formaté et prêt à enregistrer.

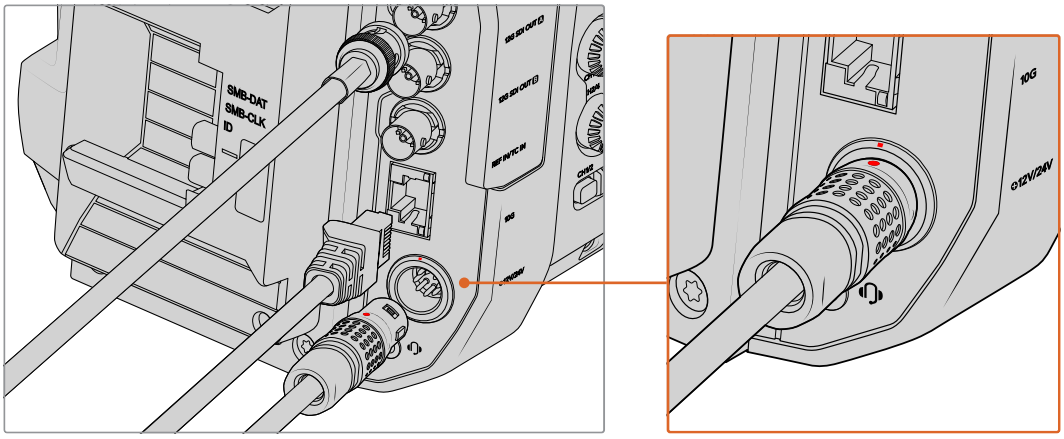
Alimenter la caméra

La façon la plus rapide d'alimenter votre caméra consiste à la connecter à une source d'alimentation externe à l'aide de l'adaptateur fourni.

Connecter la caméra à une source d'alimentation externe :

1. Comencez par retirer le cache de protection et rangez-le dans un endroit sûr. Connectez l'adaptateur AC vers DC 24V à une prise secteur à l'aide d'un câble d'alimentation secteur IEC C13 standard.
2. Connectez le connecteur à 8 broches de l'adaptateur au connecteur d'alimentation DC 24V situé sur le panneau arrière de la caméra.

Si la batterie et la source d'alimentation externe sont branchées, la caméra utilisera la source d'alimentation externe. Lorsque vous débranchez la source d'alimentation externe alors que la batterie est installée, la caméra passe automatiquement sur la batterie pour un enregistrement en continu.



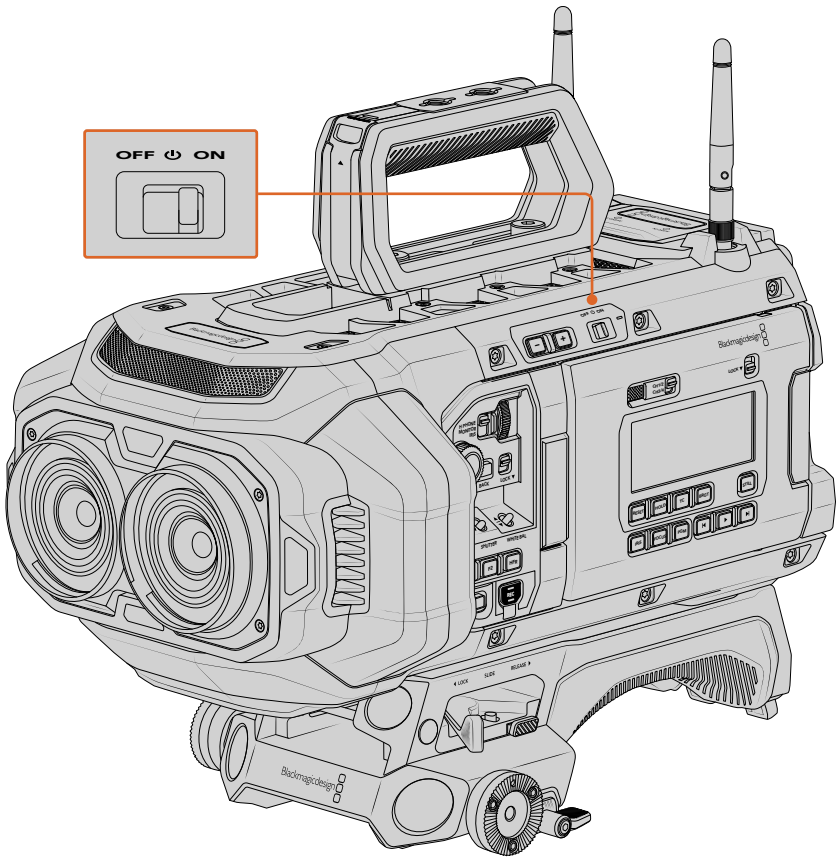
Pour brancher l'alimentation, alignez les points rouges et poussez doucement le connecteur jusqu'à ce qu'il s'enclenche. Pour le retirer, tenez le manchon métallique du connecteur et tirez doucement pour l'éloigner de la caméra. Le manchon se retire et le connecteur se détache.

Le support pour batterie inclus est un support B Mount conçu pour les batteries B Mount de 24 volts. La plage 24-34V est idéale pour l'utilisation de la URSA Cine Immersive, car la tension plus élevée réduit les pertes thermiques, ce qui permet d'enregistrer des fréquences d'images élevées et d'alimenter plusieurs accessoires à partir de votre caméra.

Les supports optionnels VLock et Gold Mount compatibles avec le 24V, et les supports Gold Mount Plus pour les batteries tierces 26V, sont également disponibles.

REMARQUE Lorsque vous utilisez la URSA Cine Immersive avec une alimentation ou des batteries de 12 à 18 volts, il est impossible d'alimenter des accessoires via le connecteur RS à 3 broches ou le connecteur de style lemo à 2 broches.

Une fois la batterie ou l'alimentation externe connectée, allumez votre URSA Cine Immersive. Déplacez l'interrupteur, situé sur le haut de la caméra, sur la position **On**.



Déplacez l'interrupteur sur **On** pour allumer la caméra.

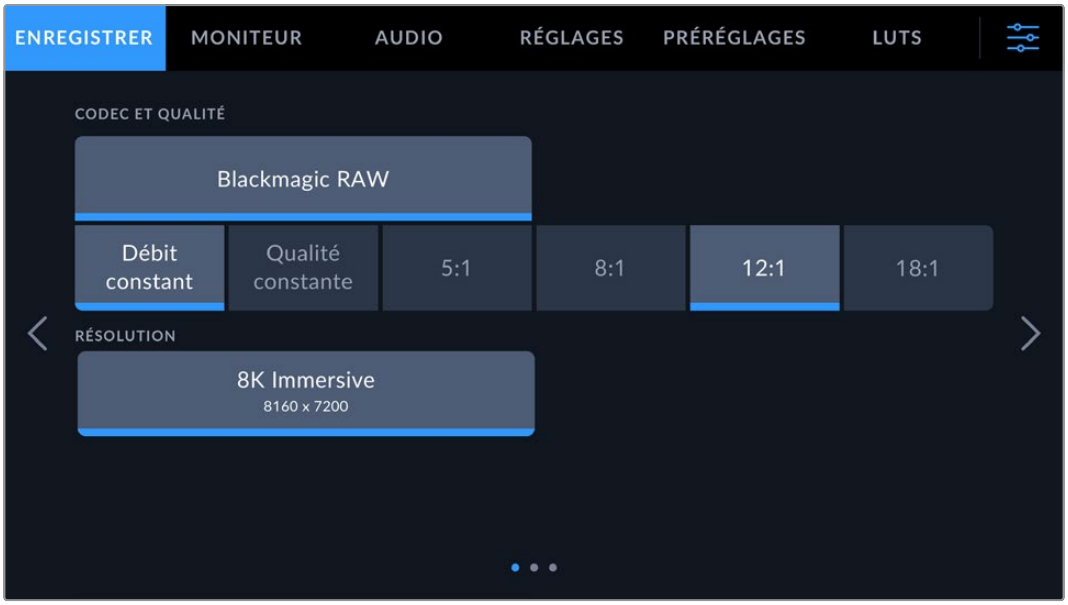
Paramétrer les réglages d'enregistrement

Cette section fournit des informations sur des réglages spécifiques de la caméra qui sont uniques ou importants lorsque vous filmez du contenu immersif. Pour ouvrir le menu de réglages de la caméra, touchez l'icône de réglages située en haut à droite de l'écran de l'assistant ou de l'écran rabattable.



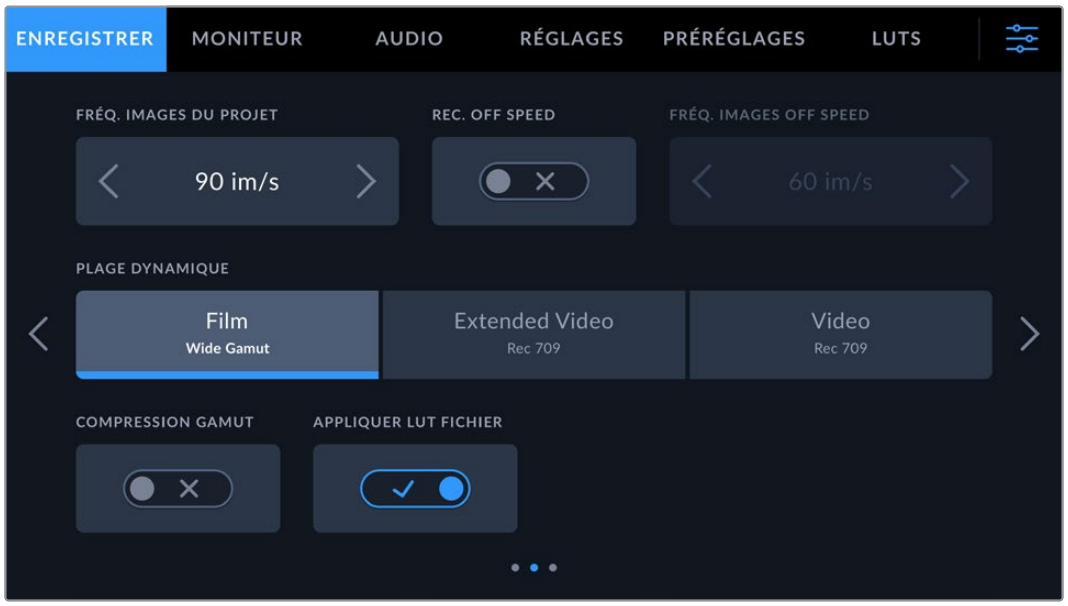
Onglet Enregistrer - Page 1

Les options **Codec et qualité** vous permettent de paramétrer le codec Blackmagic RAW selon le projet sur lequel vous travaillez. Les options sont divisées entre quatre paramètres de **Débit constant** et de **Qualité constante**. Il est hautement recommandé d'utiliser une compression 12:1 ou Q3 pour enregistrer du contenu à 90 im/s pour l'Apple Vision Pro. Ces réglages offrent l'équilibre idéal pour maintenir des tailles de fichiers gérables par les ordinateurs actuels, tout en conservant une qualité d'image extrêmement haute.



Onglet Enregistrer - Page 2

La deuxième page de réglages de la page **Enregistrer** comprend les fréquences d'images.



La fréquence d'images du projet de votre caméra doit être réglée sur 90 images par seconde pour être conforme à la fréquence d'images d'affichage de l'Apple Vision Pro.

Si vous filmez pour d'autres systèmes, des fréquences d'images plus basses, comme 60 im/s ou 50 im/s, peuvent être suffisantes. Des fréquences d'images de film traditionnel plus basses, comme 24 im/s ou 25 im/s, peuvent convenir si vous filmez des pelures d'effets visuels pour du contenu cinéma 24p.

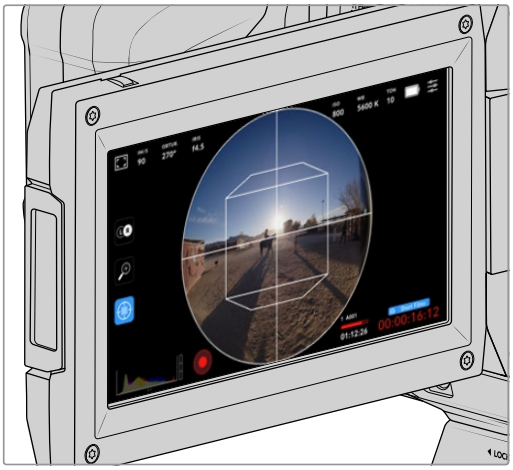
Vous avez toutes les cartes en main pour démarrer avec votre URSA Cine Immersive ! Le Media Module étant déjà formaté dans votre caméra, vous pouvez commencer à enregistrer immédiatement.

Monitoring de l’image

Votre URSA Cine Immersive inclut plusieurs options pour le monitoring de votre image pendant le tournage.

L’écran tactile rabattable a été conçu pour être utilisé par le cadreur. Il peut être inversé pour réduire le profil physique de la caméra tout en conservant la visibilité de l’écran, ou positionné vers l’avant pour l’auto-monitoring si vous vous filmez vous-même.

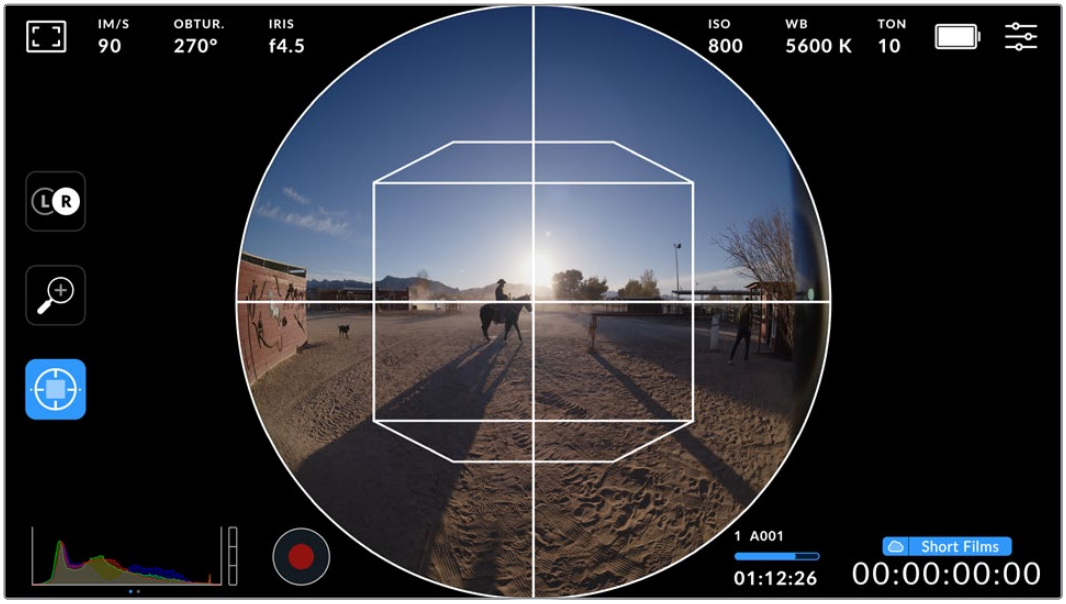
L’écran tactile de l’assistant permet à un autre membre de l’équipe de gérer le monitoring de l’image sans obstruer la vue du cadreur. Comme le champ de la caméra est extrêmement large, veuillez faire attention lorsque vous utilisez l’écran tactile de l’assistant.





Options de monitoring et affichage semi-transparent

La caméra affiche l’espace de l’objectif, avec un œil visionnable à la fois. Les écrans rabattable et de l’assistant sont cachés pour afficher une vue à 180 degrés. Les sorties SDI montrent l’image entière à 210 degrés capturée par chaque capteur.



Toutes les sorties de monitoring affichent le même œil. Vous pouvez basculer entre l’œil gauche et droit en touchant l’interrupteur **G D** sur chacun des écrans.

Un affichage optionnel peut être activé sur les écrans tactiles en appuyant sur le bouton d'affichage. Cela inclut :

- Une limite extérieure de 180 degrés
- Un réticule centrale marquant le centre de l'image
- Un repère interne représentant le champ de vision approximatif de l'Apple Vision Pro lorsque vous regardez droit devant.

Ces outils sont conçus pour vous aider à avoir un cadrage et un positionnement précis pendant la captation immersive.

Cadrer votre première prise

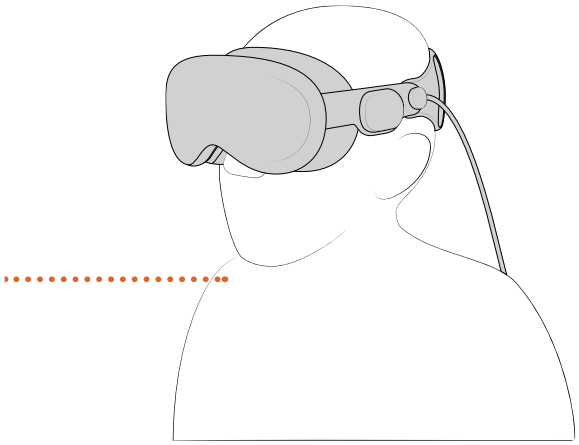
Le contenu immersif et traditionnel 2D partagent beaucoup de choses en commun, mais il y a également des différences.

Par exemple, avez-vous déjà eu la nausée pendant un trajet en voiture, car vous scrolliez sur votre téléphone ? Cela est dû à un décalage sensoriel, lorsque votre cerveau reçoit des signaux conflictuels. En effet, votre oreille interne indique à votre corps que vous êtes en mouvement, mais vos yeux lui communiquent que vous êtes à l'arrêt.

Il est aisé de reproduire une expérience similaire avec l'immersif. Par exemple, si vous faites un pan avec la caméra, les yeux des spectateurs vont leur indiquer qu'ils sont en mouvement, mais leur cou et oreille interne vont leur dire le contraire. C'est pourquoi il est préférable de maîtriser les plans stationnaires avant d'apprendre à créer des mouvements de caméra plaisants pour votre public.

Hauteur de la caméra

Dans le cinéma 2D traditionnel, vous pouvez placer la caméra vers le haut ou le bas pour signifier un rapport de force dans une scène. Lorsque vous filmez en immersif, pensez à l'impact que peut avoir une caméra placée au sol. Comment cela va affecter le spectateur ? Cela peut lui donner la sensation d'être enterré jusqu'au cou.



La hauteur perçue du sol peut impacter l'expérience de visionnage des spectateurs. Une caméra basse indiquera une hauteur qui ne fera sans doute aucun sens pour le public !



Lorsque vous filmez en immersif, les plans en contre-plongée peuvent être inconfortables pour le spectateur.

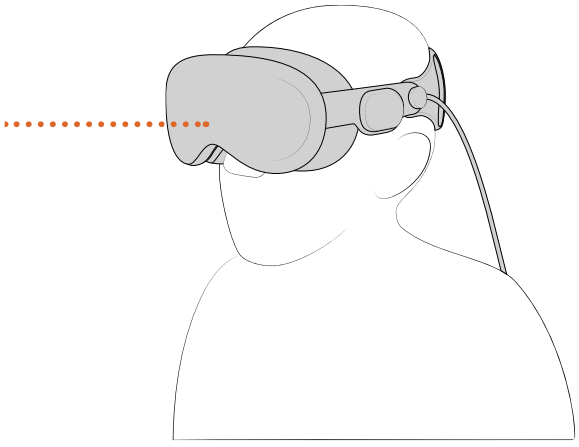
Pour l'aider à se sentir physiquement présent, il est préférable de choisir une hauteur et une position de caméra qui donnent l'impression d'être debout ou assis dans la scène. Mais il reste tout de même de la place pour exprimer votre intention créative. Si des gens sont présents dans la scène, une caméra légèrement plus haute peut donner du pouvoir au spectateur, lui permettant de regarder les autres de haut. À l'inverse, une caméra légèrement plus basse force le spectateur à lever les yeux vers les autres.

Tout cela pour dire qu'il n'est pas impossible de conserver le confort du public tout en utilisant des placements de caméra plus aventureux. Les caméras volantes, les vues de dessus, les contre-plongées et bien d'autres techniques ont été utilisées avec succès dans des productions immersives. Lorsqu'elles sont associées au bon montage, mouvement et contexte, elles peuvent être utiles pour divertir votre public.

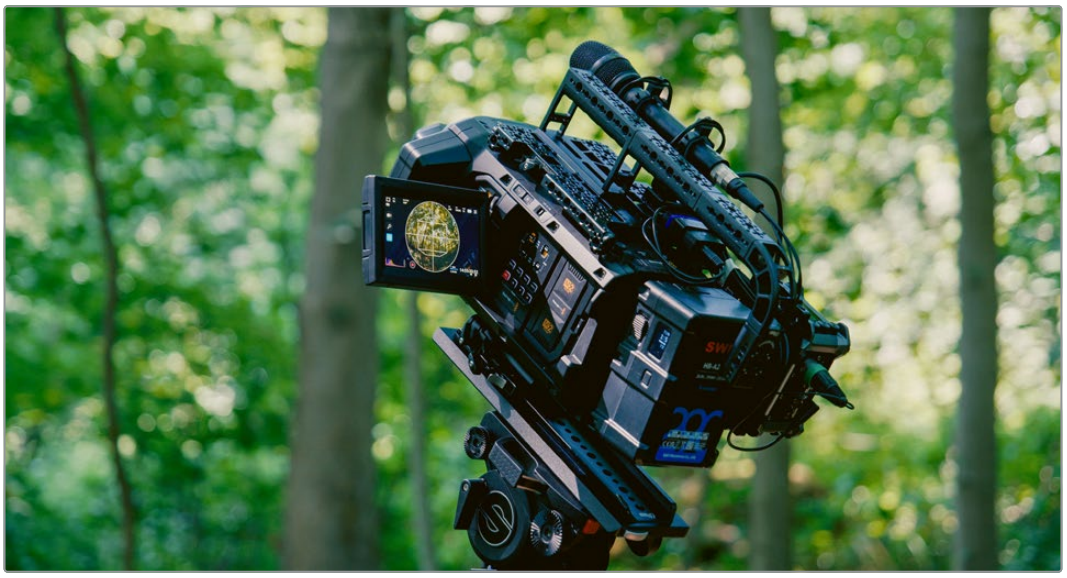
Inclinaison de la caméra

Lorsque vous filmez du contenu 2D traditionnel et que vous placez la caméra en plongée ou en contre-plongée, le directeur de la photographie utilisera souvent l'inclinaison de la caméra pour conserver la composition. Pensez à l'impact sur une image immersive.

Si vous inclinez la caméra vers le bas, afin que le public voie l'horizon, il devra lever la tête. Cela ne provoquera pas de cinétose, mais cela peut briser la sensation d'immersion, car dans la réalité, l'horizon est toujours directement devant.



Les prises immersives sont plus logiques lorsque l'horizon est à zéro degré.



Si vous inclinez la URSA Cine Immersive, cela provoquera une déconnexion pour le spectateur entre là où il doit regarder pour apercevoir l'horizon, et là où il regarderait normalement pour le voir.

C'est pour cela que la plupart des prises immersives sont capturées avec une inclinaison à zéro degré, ou proche de zéro. Comme la caméra capture un champ de vision à 180+, horizontalement et verticalement, le spectateur peut tout de même regarder de haut en bas. Mais cette fois, ce n'est pas le cinéaste qui force le spectateur à regarder dans une direction, car c'est le spectateur qui a le contrôle.

Pour l'encourager à regarder différentes parties de l'image, vous pouvez utiliser d'autres techniques comme l'éclairage, l'ouverture du cadre ou les lignes directrices.

Positionnement

Le format immersif est généralement visualisé en position assise. Comme ce n'est pas de la réalité augmentée ou de la réalité virtuelle, le spectateur n'a pas à se tenir debout ou à se déplacer. Êtes-vous assis en ce moment-même ? Pour faire une petite expérience, tournez votre tête d'un côté à l'autre, puis de haut en bas. Jusqu'où pouvez-vous confortablement tourner votre tête sans bouger votre corps ? Le champ de vision du format immersif dépasse la plage de confort de rotation pour la plupart des gens. C'est une bonne chose car, que vous tourniez la tête vers la gauche ou vers la droite, le format fournit tout de même une vision périphérique. Cette expérience vous aide à comprendre où votre action principale doit être positionnée dans le champ de vision.

Plage de mouvement standard de la tête :

Rotation de 120 à 160 degrés
Flexion et extension : 110 à 130 degrés

Lorsque la rotation de la tête est associée au mouvement de l'œil, il est possible de regarder par-dessus votre épaule, même si cela peut être une position inconfortable à tenir.

Pour vous accompagner avec la composition, activez l'affichage semi-transparent de votre caméra. Conserver l'action à l'intérieur des marques internes, qui indiquent le champ de vision standard, aidera la visibilité de votre contenu.

Le mouvement des points d'intérêt dans l'image doit également être pris en compte. Suivre une personne qui marche de l'extrême gauche à l'extrême droite nécessite un grand mouvement de tête. De la même manière, passer d'une prise où le point d'intérêt est au bord de l'image à une prise où le point d'intérêt est au bord opposé, nécessite le même grand mouvement de tête. Une prise en compte attentive de la composition facilitera le confort du public, tout en communiquant de manière créative.

Agentivité du spectateur

Lorsque vous filmez du contenu 2D, vous pouvez utiliser le cadrage, le choix de l'objectif et les réglages de mise au point pour diriger l'attention du public. Avec le format immersif, le champ de vision à 180 degrés complique l'exclusion d'informations et la direction de l'attention du public. Il offre au spectateur une autonomie bien plus grande pour explorer vos images. C'est un des facteurs qui rend l'expérience immersive si participative. Il est toujours possible de guider l'attention de votre public via une utilisation minutieuse du son directionnel, de l'éclairage, du blocking et de la composition.

Mouvement de caméra

Une caméra immersive n'a pas besoin de rester statique, mais il faut tout de même faire attention car certains mouvements de caméra peuvent paraître peu naturels ou inconfortables. Prenons l'analogie d'une voiture pour aider à comprendre comment rendre un mouvement de caméra immersive confortable. Un gros véhicule a une masse et une inertie résistant aux changements soudains de vitesse ou de direction. Il va accélérer et décélérer au fil du temps et bouger de manière contrôlée et prévisible. Une fois que vous avez maîtrisé les prises statiques et que vous passez aux prises immersives, prenez en compte ces facteurs pour offrir un mouvement plaisant :

- **La durée :** Donnez à votre public le temps de s'adapter au mouvement de la caméra et de l'apprécier. Les mouvements brusques vont plutôt créer de la confusion.
- **L'intensité :** Les mouvements de faible intensité sont plus faciles à intégrer dans un montage immersif. Les mouvements de haute intensité sont possibles, mais un échauffement et un radoucissement accompagneront le confort du public.
- **Ancrages visuels :** Ce sont des parties de la scène qui ne bougent pas. Une prise sans ancrage visuel va désorienter le public. Les éléments en avant-plan, comme l'intérieur d'une voiture ou des objets distincts comme l'horizon, aideront à ancrer vos plans.

Tableaux des débits de données

La tableau suivant affiche les débits de données pour le contenu immersif 8K x 7K à 90 images par seconde et à 24 images par seconde.

Débits de données maximaux à 90 im/s avec le débit constant Blackmagic RAW

8K x 7K Immersif			
Codec	**Dimensions des pixels**	**Fréquence d'images**	**Débit**
Blackmagic RAW 5:1	8160 x 7200 Stéreo	90	3,203 MB/s
Blackmagic RAW 8:1	8160 x 7200 Stéreo	90	2002 MB/s
Blackmagic RAW 12:1	8160 x 7200 Stéreo	90	1335 MB/s
Blackmagic RAW 18:1	8160 x 7200 Stéreo	90	889 MB/s

Débits de données maximaux à 24 im/s avec le débit constant Blackmagic RAW

8K x 7K Immersif			
Codec	**Dimensions des pixels**	**Fréquence d'images**	**Débit**
Blackmagic RAW 5:1	8160 x 7200 Stéreo	24	854 MB/s
Blackmagic RAW 8:1	8160 x 7200 Stéreo	24	534 MB/s
Blackmagic RAW 12:1	8160 x 7200 Stéreo	24	356 MB/s
Blackmagic RAW 18:1	8160 x 7200 Stéreo	24	237 MB/s

Audio et timecode

L'audio est une partie essentielle de l'expérience immersive. Il est important d'équiper votre URSA Cine Immersive avec du matériel d'acquisition audio dédié.

Apple Spatial Audio Format

L'Apple Spatial Audio Format, ou ASAF, est utilisé pour fournir de l'audio 3D lors du visionnage en Apple Immersive Video. L'ASAF fait partie du cadre de positionnement audio d'Apple et peut stocker trois types d'enregistrement audio.

- **Orienté canal** : Vous pouvez intégrer un mix surround traditionnel dans votre fichier ASAF. C'est idéal pour une narration ou une musique verrouillée.
- **Orienté objet** : Les sons peuvent être traités comme des objets individuels avec des métadonnées qui décrivent leur position dans l'espace 3D, indépendamment des canaux de haut-parleur individuels. Ce sont généralement des enregistrements mono. Leurs coordonnées 3D peuvent être animées pour suivre des objets qui se déplacent dans la scène. C'est idéal pour l'audio diégétique, qui se réfère aux objets qui sont visibles sur scène.
- **Ambisonique** : C'est un format surround entièrement sphérique qui peut rendre l'audio dans toutes les directions à l'aide d'une harmonie sphérique. C'est idéal pour le son d'ambiance.

La suite de cette section fournit un aperçu de l'équipement nécessaire pour construire un fichier ASAF efficace.

Ambisonique

Un microphone ambisonique contient quatre capsules ou plus, organisées en un motif géométrique précis. Chaque capsule capture les mêmes sons avec différents temps d'arrivée et d'intensité selon la direction d'où provient le son. Ces capsules sont enregistrées sur des canaux discrets, afin d'être traitées par un ordinateur ultérieurement. En comparant les différences, le logiciel peut estimer la direction et la nature des sons enregistrés.



Un microphone ambisonique de premier ordre

Microphones ambisoniques

Les microphones ambisoniques sont classifiés selon le nombre de capsules qu'ils contiennent. Plus il y a de capsules, plus le micro peut reproduire les détails spatiaux de manière détaillée.

Premier ordre :	4 canaux
Deuxième ordre :	9 canaux
Troisième ordre :	16 canaux
Quatrième ordre :	25 canaux et plus

Que le micro ambisonique soit sur la caméra ou à côté, il est essentiel qu'il soit correctement aligné avec la direction de captation de la caméra. De plus, comme les micros ambisoniques peuvent être utilisés dans différentes configurations (longitudinale, transversale, vers le haut ou vers le bas), il est important que votre technicien audio sache quelle configuration a été choisie. Cette information peut être transmise dans les métadonnées sur l'enregistreur audio ou via les notes partagées.

Les enregistrements ambisoniques peuvent être utilisés comme fond ambiant. Les autres effets sonores orientés canal ou orientés objet peuvent y être superposés. S'ils sont correctement alignés avec la caméra, les enregistrements ambisoniques peuvent aussi servir de référence pour l'ingénieur lors de la conception sonore.

Micros-cravates et micros miniatures

Lorsque vous filmez avec une caméra à 180 degrés, il est impossible d'utiliser une perche télescopique, car elle sera visible dans votre scène ! Vous devrez utiliser des micros cachés et des micros-cravates pour enregistrer l'audio diégétique et les dialogues.

Par exemple, vous pouvez :

- Connecter ces sources audio directement aux deux entrées XLR de la URSA Cine Immersive à l'aide de transmetteurs et de receveurs sans fil.
- Les connecter à un enregistreur audio externe. Cela peut être un grand appareil multi-canal ou un appareil miniature porté au corps.

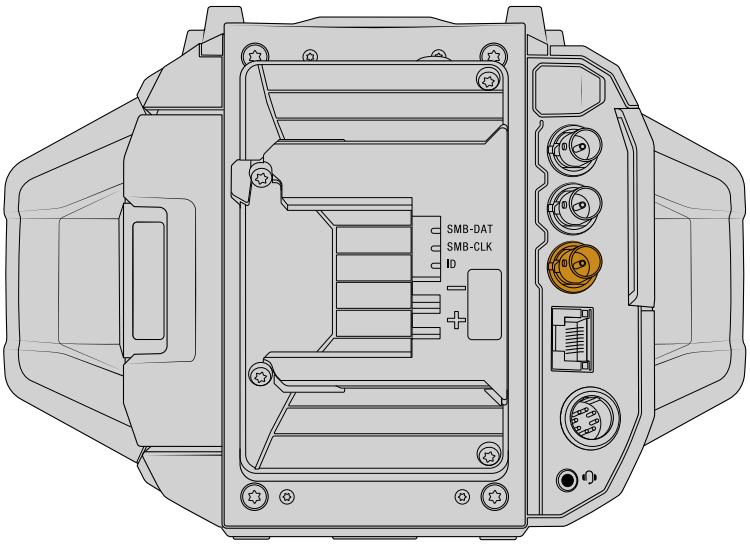
Ces enregistrements peuvent être synchronisés et mixés en tant que sons orientés canal ou orientés objet.

Micros à main

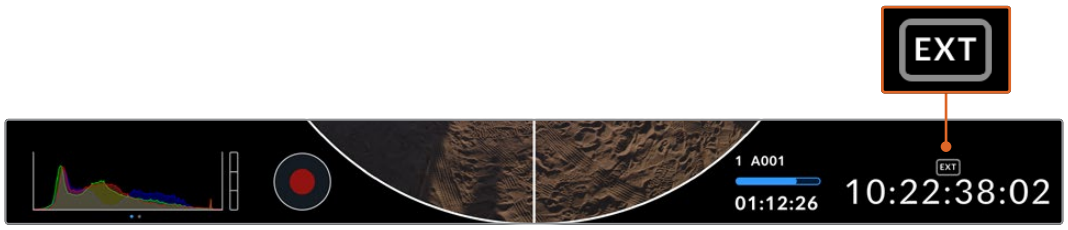
Enregistrer des effets sonores supplémentaires sur le lieu de tournage une fois que la caméra a cessé de tourner peut faire économiser du temps en post-production, mais aussi produire un son plus authentique. Apporter un micro hypercadioïde ou similaire sur la production peut être très bénéfique, car cela permet à l'action d'être recréée avec le micro positionné aussi près de l'action que possible. Ces enregistrements peuvent ensuite être montés dans le mix final en tant que sons orientés canal ou orientés objet.

Connecter le timecode

Pour connecter un timecode externe à votre URSA Cine Immersive, utilisez un câble BNC pour brancher un générateur de timecode à l'entrée du timecode sur le panneau arrière de votre caméra. Se connecter à un timecode synchronisé lors d'une prise en système double permet de faciliter la synchronisation de l'audio et de la vidéo durant la post-production.



Lorsque vous envoyez un timecode externe avec une fréquence d'images correspondante sur cette entrée, la caméra se synchronisera automatiquement sur le timecode entrant. Lorsque le signal de référence externe est verrouillé, l'indicateur **EXT** apparaît sur l'écran LCD de la URSA Cine Immersive, à côté du compteur de timecode. Si vous débranchez le câble, le timecode sera bloqué et l'indicateur **EXT** apparaîtra pour vous indiquer que vous utilisez désormais l'horloge du timecode interne de votre caméra.



EXT apparaîtra avec le timecode de la caméra si vous êtes connecté à un générateur de timecode

REMARQUE Les normes de timecode SMPTE existent pour les fréquences d'images standard telles que 23.976, 24, 30, etc. Il n'existe pas de norme de timecode SMPTE pour la fréquence de 90 images par seconde. Ainsi, lorsque la fréquence d'images du projet de la URSA Cine Immersive est réglée sur 90 images par seconde, la caméra utilise en fait le timecode de 30 im/s. Les sorties SDI opèrent également à 30p. Les accessoires de timecode ou les enregistreurs audio externes doivent être réglés sur 30 im/s.

Filmer

Maintenant que vous avez acquis les fondamentaux du fonctionnement de votre caméra Blackmagic URSA Cine Immersive et du format immersif, ce chapitre rentrera davantage en détail dans l'opération de la caméra.

Exigences d'exposition

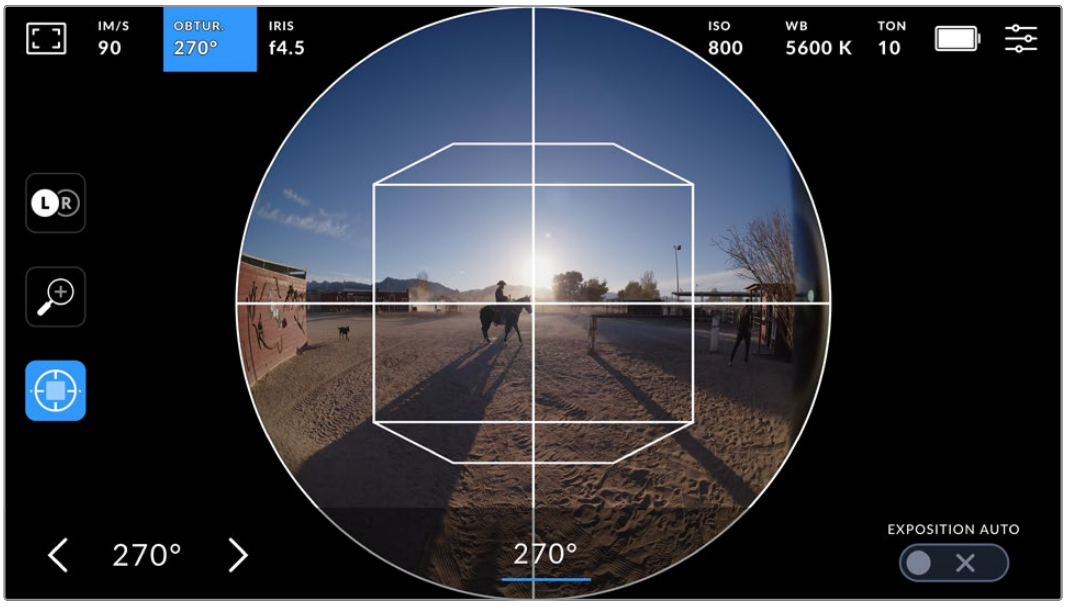
L'Apple Immersive Video a des exigences d'exposition plus élevées que la captation traditionnelle à 24 im/s. Il vous faudra peut-être repenser votre approche du tournage et de l'éclairage. Si vous conservez un obturateur à 180 degrés, l'acquisition à 90 im/s nécessite presque quatre fois plus de lumière pour la même quantité d'exposition. De plus, l'iris fixe des objectifs, autour de f4.5 est jusqu'à trois diaphs plus sombre que les objectifs Super Speed standard. Ainsi, vous devrez sûrement augmenter de sept diaphs de lumière afin d'obtenir l'exposition adéquate. Cependant, si vous filmez à 90 im/s, utiliser un angle d'obturation plus large, entre 270° et 360°, peut aider à compenser la perte lumineuse de la fréquence d'images élevée tout en obtenant un flou de bougé plaisant.

Vitesse d'obturation

L'indicateur **Obtur.** affiche l'angle ou la vitesse d'obturation. Vous pouvez choisir d'afficher l'angle ou la vitesse d'obturation depuis le menu **Réglages**.

Pour modifier la vitesse d'obturation :

1. Appuyez sur l'icône **Obtur.** dans la barre d'outils supérieure.
2. Utilisez les flèches gauche ou droite pour naviguer dans les préréglages de vitesses d'obturation les plus courantes. Si vous souhaitez utiliser une valeur d'obturation spécifique, touchez deux fois l'indicateur d'obturation situé en bas à gauche de l'écran. Un clavier numérique s'affichera et vous pourrez saisir la valeur d'obturation de votre choix jusqu'à deux décimales, entre 11.2 et 360 degrés.



Les fréquences d'images personnalisées peuvent être saisies via le clavier numérique.

La vitesse d'obturation a moins d'impact sur l'apparence du contenu immersif à fréquence d'images élevée. Tandis qu'une obturation à 180 degrés est souvent privilégiée pour filmer à 24 ou 30p, il est plus commun que la vitesse d'obturation soit utilisée pour accompagner la captation de contenu immersif.

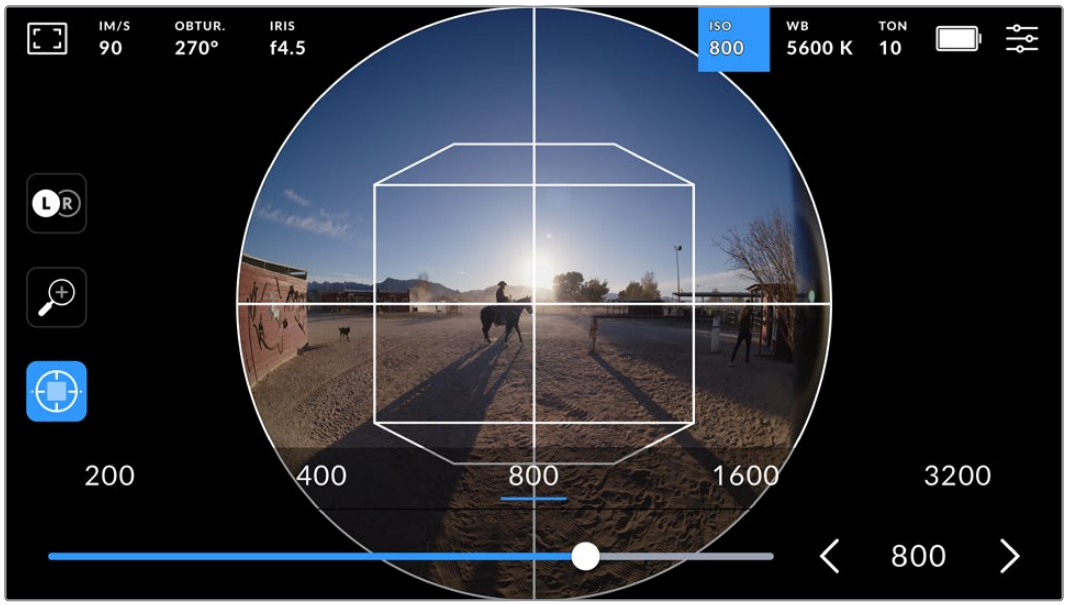
Lorsque vous filmez en faible condition de luminosité, l'obturateur peut être augmenté à 270 degrés ou, dans des circonstances exceptionnelles, à 360 degrés.

Dans des conditions lumineuses, la vitesse d'obturation peut être réduite jusqu'à 11,2 degrés. Cependant, si la luminosité est forte, il est plus usuel de laisser la vitesse d'obturation sur votre réglage favori pour le mouvement et d'utiliser les filtres ND afin de contrôler l'exposition.

Veuillez noter que la fréquence d'images élevée de 90 images par seconde de l'Apple Immersive Video peut causer des problèmes lorsque vous filmez avec des lumières clignotantes. Une obturation à 270 degrés est recommandée pour des environnements à 60hz, et à 324 degrés pour des environnements à 50hz.

ISO et plage dynamique

L'indicateur **ISO** affiche le paramètre ISO (sensibilité de la lumière) réglé sur la URSA Cine. Toucher l'indicateur vous permet de régler l'ISO pour répondre aux besoins de différentes conditions d'éclairage. La plage ISO de la URSA Cine Immersive va de 200 à 3 200. L'ISO optimum est 800.



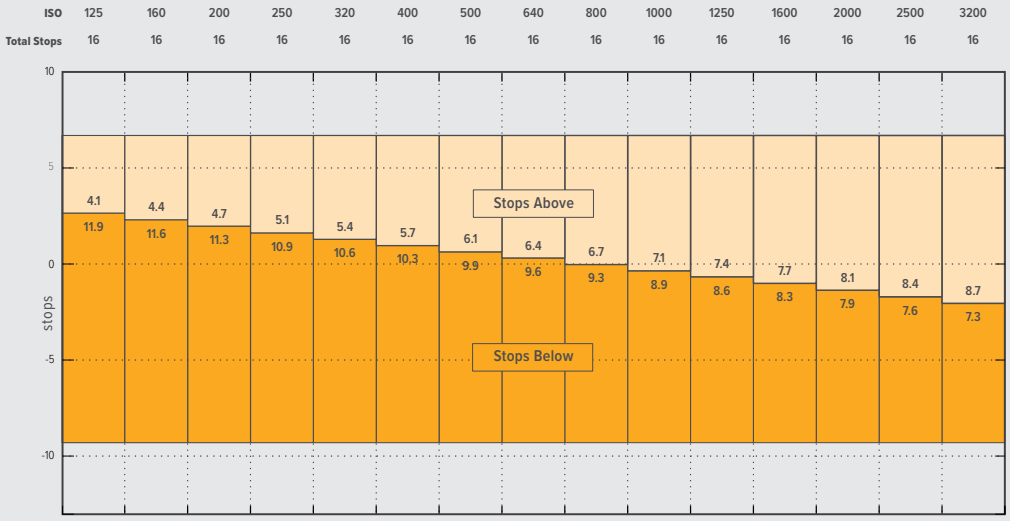
L'ISO peut être configuré en touchant l'indicateur ISO.

Les préréglages numériques vous permettent de régler l'ISO sur des valeurs entières, en les sélectionnant sur l'écran. Le curseur situé au-dessous des préréglages vous permet d'affiner vos réglages en incrémentant l'ISO d'1/3 de diaphragme.

Toutefois, en fonction de la situation, vous pouvez choisir une sensibilité ISO plus ou moins élevée. Par exemple, en conditions de faible éclairage, une sensibilité de 1600 ISO peut s'avérer nécessaire mais pourra introduire du bruit. Dans des conditions d'éclairage élevé, vous obtiendrez des couleurs plus riches en réglant l'ISO sur 200.

Les caméras Blackmagic sont dotées d'une large plage dynamique, ce qui vous offre une grande flexibilité lors de la capture d'images et de l'étalonnage dans DaVinci Resolve. Ce tableau montre la plage dynamique disponible de votre URSA Cine Immersive.

Plage dynamique de la URSA Cine Immersive



ND

Les filtres ND de la caméra sont le moyen le plus courant de contrôler l'exposition, vous permettant de réduire la quantité de lumière reçue par le capteur de la caméra par un nombre donné de repères de diaphs.



Les filtres ND de la URSA Cine Immersive sont ajustables via les boutons situés sur la gauche, au-dessus de l'écran tactile rabattable. Il est possible de désactiver le filtre ou de choisir entre les options **2**, **4**, **6** et **8** diaphs.

Le réglage du filtre ND actuel est affiché via l'indicateur situé en haut à gauche de l'écran tactile. Lorsque ND est actif, l'indicateur est affiché de manière permanente. Lorsque ND est désactivé, l'indicateur disparaît après 4 secondes.



REMARQUE Vous pouvez choisir la terminologie utilisée par l'indicateur du filtre ND afin qu'elle reflète la convention souhaitée. Utilisez le menu des réglages pour choisir entre un affichage en nombres de ND, en diaphs ou en fractions.

Enregistrement Off speed

La fréquence d'images du projet du menu des réglages de la URSA Cine Immersive est la fréquence d'images d'enregistrement et elle définit la vitesse de lecture de vos images. En général, on fait coïncider la fréquence d'images avec la vitesse de lecture et la synchronisation audio utilisées dans le workflow de post-production.



Touchez l'indicateur Im/s de votre URSA Cine Immersive pour accéder aux paramètres de fréquence d'images. La Blackmagic URSA Cine Immersive possède 8 paramètres de fréquence d'images dont 23.98, 24, 25, 29.97, 30, 50, 59.94, 60 et 90 im/s. Lorsque vous filmez en immersif, la fréquence d'images du projet doit être réglée sur 90 images par seconde.

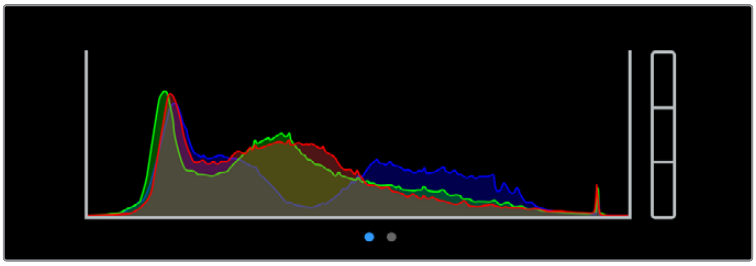
La fréquence d'images du capteur vous permet de définir la vitesse à laquelle votre capteur capture une image. Par défaut, les fréquences d'images du projet et du capteur de la URSA Cine Immersive correspondent à une vitesse de lecture naturelle. Cependant, en touchant l'interrupteur **Fréq. images off speed** situé en bas à droite du menu **Im/s** de la caméra, vous pouvez régler indépendamment la fréquence d'images du capteur.

Quand la fréquence d'images off speed est activée, vous pouvez régler manuellement la fréquence d'images du capteur. Veuillez noter que la fréquence de 90 images par seconde est la plus élevée de la caméra et qu'elle est adaptée à celle d'un projet pour Apple Vision Pro. Cependant, en sélectionnant une fréquence d'images plus basse, comme 30 ou 60 im/s, puis en filmant en off speed à 90 images par seconde, vous pourrez créer de magnifiques ralentis.

Lorsque l'option Freq. off speed est activée, l'indicateur des fréquences d'images en haut à gauche de l'écran affichera la fréquence du capteur, suivi par celle du projet.

Histogramme

L'histogramme est situé en bas à gauche de l'écran tactile de la URSA Cine Immersive. Un histogramme est un graphique qui montre la distribution de la luminosité dans une image. Le côté gauche représente les basses lumières, le milieu les tons moyens, et le côté droit les hautes lumières. Le haut du graphique montre la quantité d'image qui se trouve à ce niveau de luminosité.



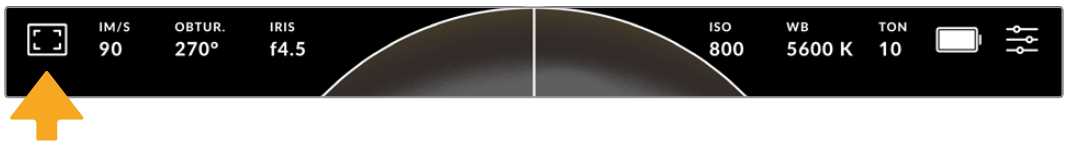
L'histogramme vous offre une indication de la distribution tonale dans votre plan en un clin d'œil. Une image pour laquelle le graphique repose sur le côté gauche est probablement sous-exposée. De même, une image pour laquelle le graphique repose sur le côté droit est probablement sur-exposée. L'histogramme doit toujours être interprété en contexte avec l'image, mais en général, une image correctement exposée est représentée par un graphique équilibré qui se situe au milieu. Augmenter l'exposition déplacera l'histogramme vers la droite, alors que réduire l'exposition la déplacera vers la gauche.

À droite de l'histogramme se trouve un indicateur qui montre lorsque des canaux de couleurs individuels sont écrêtés. Il faut en général éviter d'écrêter un canal. Cependant, dans certains cas, il est acceptable d'écrêter l'image si cela vous aide à préserver les tons moyens et les basses lumières de votre image.

« Exposer vers la droite » est une technique où vous sur-exposez délibérément une image en poussant l'histogramme le plus à droite possible sans écrêter les canaux de couleurs. Cela maximise le signal et minimise le bruit. Du moment que l'exposition est réduite en post, c'est un excellent moyen de produire des images propres !

Options de l'écran LCD

Touchez l'indicateur écran en haut à gauche de l'écran tactile de la URSA Cine Immersive pour ouvrir les options de l'écran LCD. Ce menu contient des options très utiles, comme le contrôle de la luminosité, ou les puissants outils d'exposition.



Zébra

Ce paramètre affiche le zébra sur l'écran tactile et permet de régler le niveau de zébra. Le zébra affiche des hachures diagonales sur les zones de l'image qui dépassent le niveau d'exposition préalablement réglé. Le paramètre de pourcentage est utilisé pour montrer les parties de votre image qui sont sur-exposées. Il est courant de le régler plutôt sur 95% que 100%, car cela permet d'identifier les parties de l'image dont la sur-exposition peut être corrigée. Le deuxième réglage permet d'activer ou de désactiver les zones de zébra gris moyen, ou gris moyen plus 1 diaph.

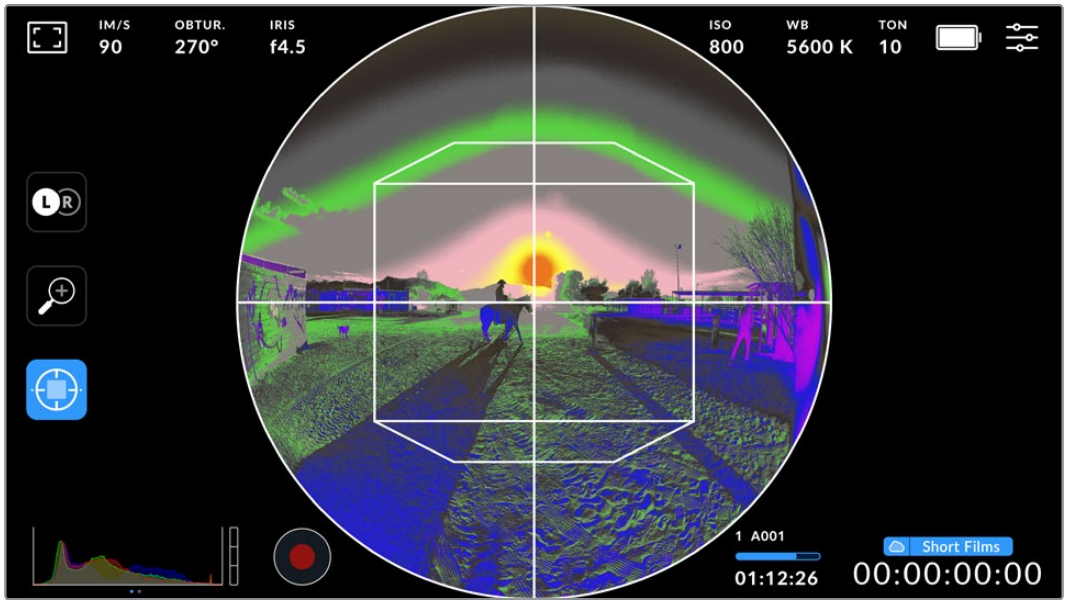
Grille

Ce paramètre affiche une grille selon la règle des tiers, un indicateur d'horizon, un réticule ou un point central sur l'écran tactile et permet de choisir les informations qui apparaîtront sur les sorties de la URSA Cine Immersive. Lorsque vous filmez du contenu immersif, l'indicateur d'horizon est très utile. Il affiche un double réticule au centre de l'image qui est aligné tant que la caméra est totalement à niveau, sinon, il divergera.

Fausses couleurs

La fonction **Fausses couleurs** applique différentes couleurs à différentes zones de l'image. Ces couleurs représentent les différentes valeurs d'exposition des éléments de votre image. Par exemple, la couleur rose représente une exposition optimale pour les peaux claires, alors que le vert correspond à une couleur de peau plus foncée. En vérifiant ces fausses couleurs lorsque vous filmez des êtres humains, vous maintiendrez une exposition stable de leur couleur de peau.

De même, lorsque la couleur des éléments de votre image passe du jaune au rouge, cela signifie qu'ils sont sur-exposés. Comme pour l'histogramme, cet outil vous offre un aperçu rapide et intuitif de la distribution tonale de l'image.



Enregistrement

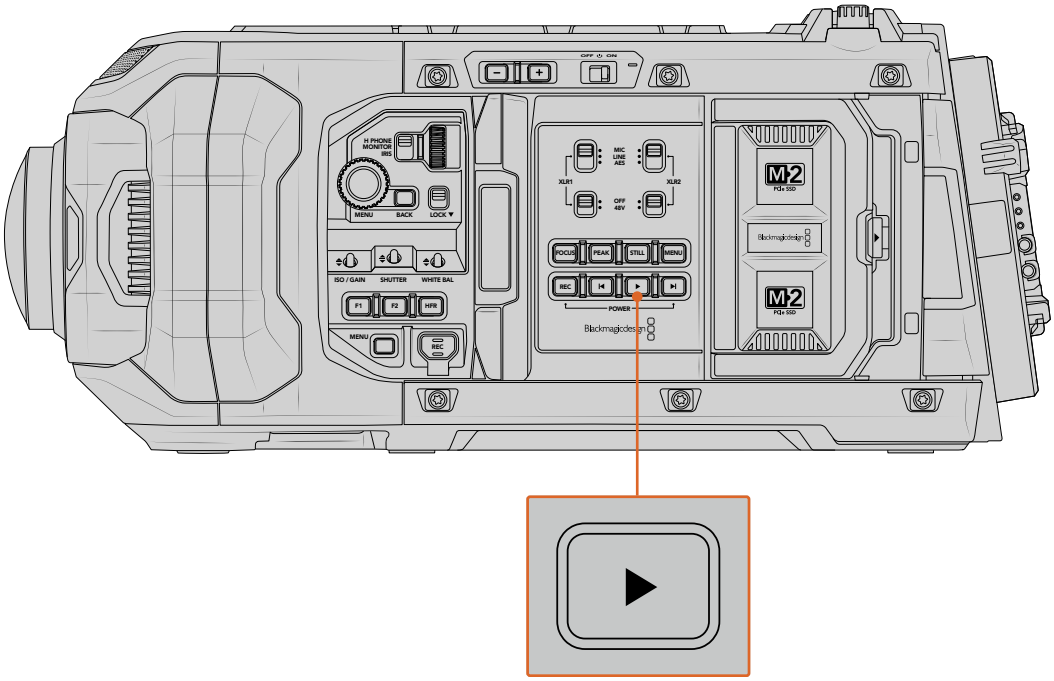
Appuyez sur l'un des boutons d'enregistrement rouges de votre caméra pour démarrer l'enregistrement. Les boutons d'enregistrement sont situés sur le panneau de contrôle avant, le panneau de contrôle interne derrière l'écran tactile rabattable, le panneau de contrôle de la station d'aide et les deux écrans tactiles LCD. Appuyez de nouveau sur ce bouton pour arrêter l'enregistrement.

Les indicateurs de stockage indiquent le temps d'enregistrement restant. Le temps est affiché en heures:minutes:secondes selon la fréquence d'images choisie. La barre du support, située au-dessus du temps d'enregistrement, s'affichera en bleu, en blanc ou en rouge selon son état. Elle affichera également l'espace utilisé sur l'espace de stockage.



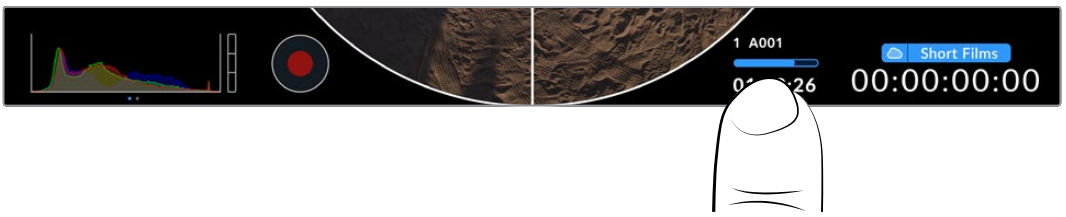
Lecture

Une fois vos clips enregistrés, vous pouvez utiliser les boutons de transport pour les lire. Pour lire le dernier clip que vous avez enregistré, appuyez sur le bouton physique de lecture situé sur le côté gauche de la caméra, sous l'écran tactile rabattable. Les boutons d'avance et de retour peuvent être utilisés pour naviguer dans d'autres clips. Pour quitter le mode de lecture et préparer la caméra à continuer d'enregistrer, appuyez une fois sur le bouton d'enregistrement.

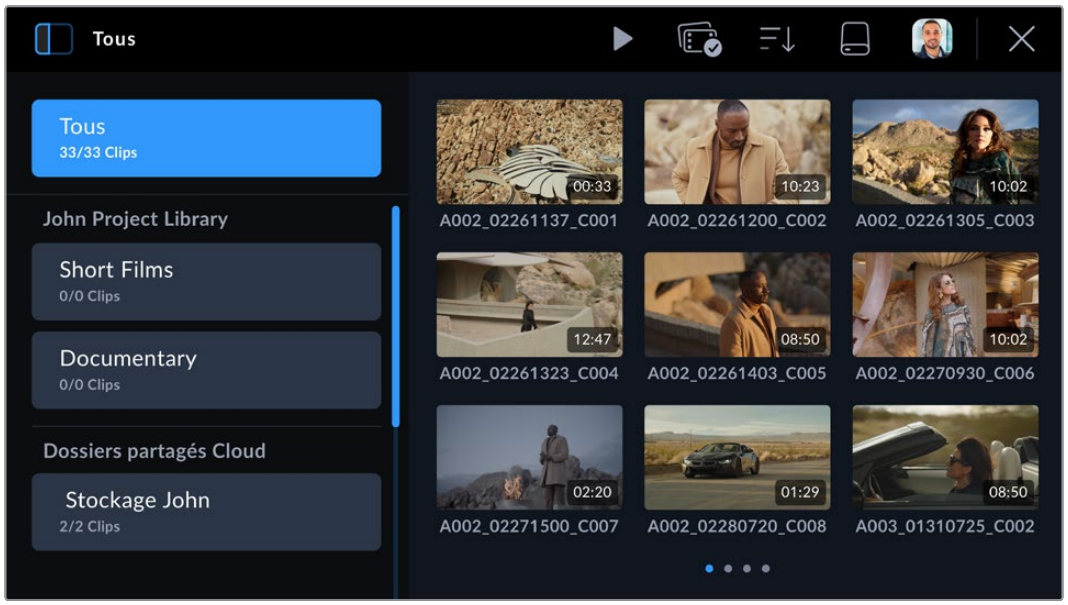


Vous pouvez vous servir des touches avant ou arrière pour passer directement au début ou à la fin des clips. Appuyez une fois sur l'icône de retour rapide pour retourner au début du clip en cours ou appuyez deux fois pour revenir au début du clip précédent. Maintenez ces boutons enfoncés pour lire en accéléré. Appuyez deux fois pour une vitesse x4, trois fois pour une vitesse x8 et quatre fois pour une vitesse x16.

Si un examen plus approfondi des images est exigé, il est préférable d'utiliser la bibliothèque de médias. Pour ouvrir la bibliothèque de médias, touchez un des indicateurs de stockage en bas de l'écran tactile de votre caméra.



Le navigateur de la page principale affiche les vignettes de tous les clips enregistrés sur tous les supports connectés à votre caméra. Il y a trois rangées de vignettes et, à mesure que vous ajoutez des clips, vous pouvez passer à la page de vignettes suivante en balayant l'écran tactile. Les indicateurs de page en bas de l'écran affichent combien de pages de clips sont disponibles.



Connectez la URSA Cine Immersive à un compte Blackmagic Cloud et envoyez les images directement vers un projet DaVinci Resolve.

Pour quitter la bibliothèque de médias à tout moment, appuyez sur la **X** située en haut à droite, ou appuyez une fois sur n'importe quel bouton d'enregistrement.

Transférer les clips

Il existe plusieurs moyens de transférer des clips depuis le stockage de votre caméra. Par exemple, vous pouvez brancher la URSA Cine Immersive sur votre ordinateur ou sur un réseau via un câble Ethernet 10G et copier les clips depuis le stockage de votre caméra via Ethernet. Vous pouvez même charger le Media Module dans un Blackmagic Media Dock connecté à votre réseau ou à votre ordinateur, ce qui vous permet de travailler avec jusqu'à trois modules simultanément.

Lors de la connexion à votre ordinateur par Ethernet, votre ordinateur reconnaîtra le Blackmagic Media Module comme il le ferait en branchant un Blackmagic Cloud Store.

Votre caméra est réglée par défaut sur DHCP, de sorte que votre ordinateur ou votre réseau verra la caméra et lui attribuera une adresse IP automatiquement.

Activer le partage de fichiers

Lorsque vous configurez votre caméra pour la première fois et que vous vous connectez à votre ordinateur ou à votre réseau via Ethernet, le partage de fichiers SMB est désactivé par défaut. Contrairement à un Blackmagic Cloud Store avec lequel vous souhaiterez probablement partager des fichiers avec d'autres utilisateurs du réseau immédiatement, vous ne voudrez peut-être pas autoriser immédiatement le même niveau d'accès aux fichiers master de la caméra après une journée de tournage importante. C'est pourquoi nous avons décidé de vous laisser activer vous-même le partage de fichiers SMB.

1. Pour activer le partage de fichiers, vous devez télécharger et installer l'utilitaire Blackmagic Camera Setup sur votre ordinateur. Il s'agit d'un logiciel d'administration qui vous permet de mettre à jour votre caméra et de modifier les paramètres d'administration. Vous pouvez le télécharger sur notre site web : https://www.blackmagicdesign.com/fr/support/family/professional-cameras
2. Après avoir téléchargé et installé l'utilitaire, connectez votre caméra à l'ordinateur via USB et lancez le Blackmagic Camera Setup. Connectez-vous via le port USB du panneau supérieur vers l'arrière de votre caméra.

3 Pour ouvrir les paramètres, cliquez sur l'image de votre URSA Cine Immersive ou sur l'icône de paramétrage.

4 Faites défiler vers le bas jusqu'aux paramètres **Network access** et activez **File sharing (SMB)**. Appuyez sur le bouton **Save** en bas à droite pour que ces réglages soient pris en compte.

5 Allez maintenant au paramètre URL en dessous et cliquez sur l'icône URL pour remplir automatiquement l'adresse. L'adresse URL affichera le nom de votre caméra en tant qu'emplacement sur le réseau local.

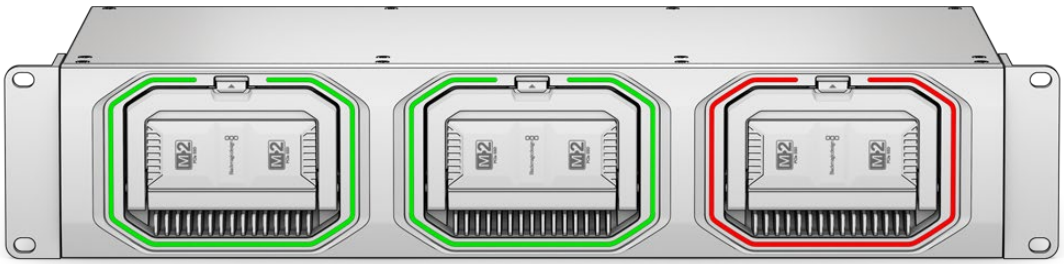
6 Cliquez sur **Save**.

Pour accéder directement à vos clips, cliquez sur le nom de l'URL de votre caméra. Une fenêtre s'ouvre alors pour vous demander un mot de passe ou de vous connecter en tant qu'invité. Cliquez sur la case d'option **Guest**, puis sur **Connect**.

7 Votre ordinateur naviguera maintenant automatiquement vers votre Media Module dans le Finder sur un Mac et dans l'explorateur Windows sous Windows. L'utilitaire se souviendra de votre configuration Ethernet, de sorte que vous n'aurez à effectuer cette opération qu'une seule fois lors de la première configuration. La prochaine fois que vous brancherez votre caméra via Ethernet, vous pourrez accéder directement au Media Module sans toucher à aucun de ces paramètres.

Utiliser le Blackmagic Media Dock

Le Blackmagic Media Module est conçu pour être chargé et retiré rapidement lorsque vous avez besoin de transférer des clips vers un stockage externe ou de passer à un module vide afin de pouvoir continuer à enregistrer. Le Blackmagic Media Dock peut être utilisé pour accéder aux images sur un Media Module sans la caméra.



Résumé

L'Apple Immersive Video introduit une nouvelle approche du storytelling visuel et offre une alternative au cinéma 2D traditionnel. Il n'a pas pour but de le remplacer, mais de présenter un format supplémentaire qui peut être adapté aux besoins du projet. Comme pour tout format, les choix créatifs peuvent varier selon le récit.

Tandis que les nouveaux formats impliquent souvent des workflows complexes, la URSA Cine Immersive est conçue pour simplifier le processus de captation en Apple Immersive Video. Elle propose un chemin de production simple pour le contenu immersif et peut également être utilisée pour les effets visuels ou d'autres applications.

Ce guide décrit les caractéristiques principales et les workflows de la URSA Cine Immersive. Il sert de point de départ. Comme pour n'importe quel système de caméra, de plus amples explorations et essais sont encouragés afin d'obtenir les meilleurs résultats pour vos besoins de production spécifiques.

DaVinci Resolve Studio



DaVinci Resolve Studio est un logiciel de post-production complet qui intègre le montage, l'étalonnage, les effets visuels, les animations graphiques et la post-production audio. Il prend en charge les workflows adaptés aux caractéristiques uniques du format Blackmagic RAW Immersive, notamment la vidéo stéréoscopique ultra haute résolution, les fréquences d'images de 90 im/s et le mixage audio spatial. Comme DaVinci Resolve Studio est équipé pour gérer la production, le monitoring et l'exportation de contenu Apple Immersive Video, c'est un outil puissant pour les créateurs de contenu immersif.

Recommandations pour les stations de travail

Recommandations matérielles

- Mac Studio M3 Ultra Desktop
- Moniteur à haute fréquence de rafraîchissement (minimum 60 Hz)

Configuration système minimale

- macOS 15 ou ultérieur
- 64 Go de mémoire système
- Pour le monitoring, Blackmagic Design Desktop Video 12.9 ou ultérieur
- Ordinateur ou GPU basé sur Apple Silicon, compatible avec Metal.

Configuration requise pour le monitoring immersif

- Apple Vision Pro
- Réseau sans fil haute vitesse (Wi-Fi 5 minimum, Wi-Fi 6 recommandé)

Configurer le projet

Avant d'importer les médias RAW provenant de la Blackmagic URSA Cine Immersive, il est recommandé de vérifier les paramètres du projet DaVinci Resolve afin de vous assurer que les médias sont correctement débayérisés et traités pour un rendu visuel optimal et un montage facilité.

Paramètres principaux

Il est essentiel d'indiquer que vous allez créer du contenu pour l'Apple Vision Pro lorsque vous configurez un projet. Ainsi, les paramètres et les options de workflow appropriés seront visibles lorsque vous préparez, assemblez et exportez votre projet.

1. Ouvrez DaVinci Resolve Studio.
2. Dans le coin inférieur droit de la fenêtre Gestionnaire de projet, cliquez sur Nouveau projet.
3. Saisissez un nom de projet, réglez l'emplacement média (si nécessaire) et cliquez sur Créer.
4. Ouvrez les paramètres du projet en cliquant sur la roue dentée en bas à droite de DaVinci Resolve Studio.

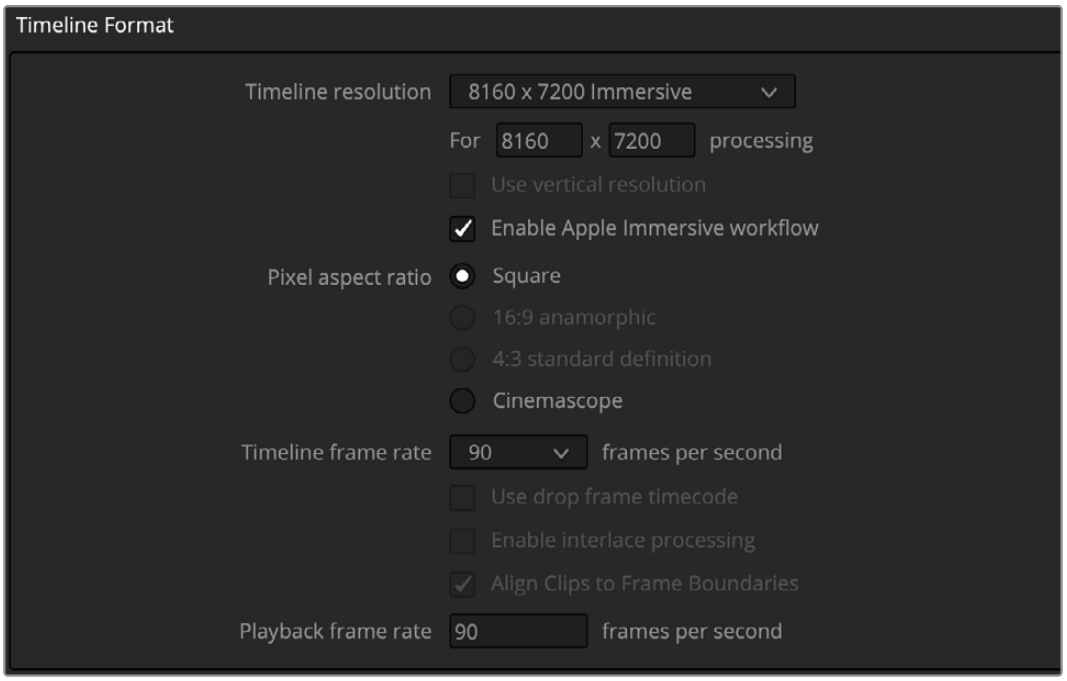
   Vous verrez d'abord la résolution et la fréquence d'images du projet.
5. Sélectionnez Configuration du projet dans la barre latérale.
6. Dans la section Format de la timeline, sélectionnez **Enable Apple Immersive workflow**.



   Cela configurera plusieurs paramètres DaVinci Resolve pour optimiser les workflows immersifs. Lorsque vous créez un projet immersif, assurez-vous de toujours sélectionner ce paramètre en premier.
7. Réglez la résolution de la timeline sur **8160 x 7200 Immersive** pour traiter les clips dans la résolution native du capteur.

   Si la puissance de traitement est limitée, choisissez **4080 x 3600 Immersive** afin de réduire la résolution de moitié pour une lecture plus rapide.
8. La **Fréquence d'images de la timeline** devrait être réglée sur 90 images par seconde. Ce paramètre est automatiquement configuré lorsque vous activez les workflows Apple Immersive.
9. Réglez la **Fréquence d'images de lecture** sur 90 images par seconde pour reproduire votre contenu vidéo immersif le plus fidèlement possible.

   Si vous travaillez avec une puissance de traitement limitée (ou un moniteur avec une plus faible fréquence de rafraîchissement), réduisez la fréquence d'images pour une lecture plus stable. Il est recommandé de choisir une fréquence d'images qui représente une fraction de la fréquence d'images native, par exemple 30 im/s ou 60 im/s.



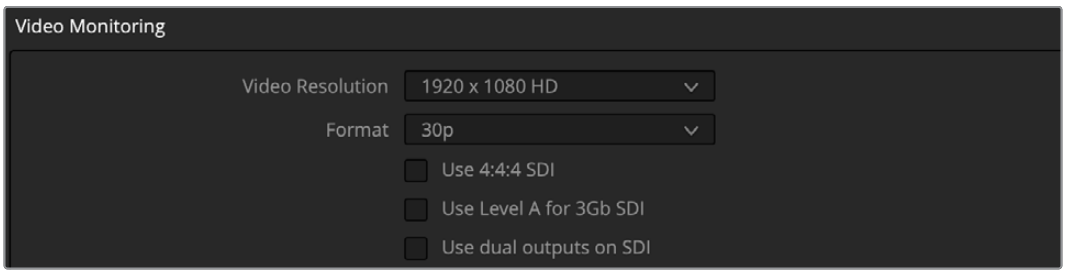
Monitoring vidéo

Si vous utilisez un moniteur externe (via un appareil d'acquisition et de lecture comme Decklink ou UltraStudio), configurez les paramètres de monitoring vidéo selon les spécifications de votre moniteur.

1 Dans la section de monitoring vidéo, réglez la résolution vidéo pour qu'elle corresponde à celle de votre moniteur. Par ex. 1920 x 1080 HD ou 3840 x 2160 Ultra HD.

Bien que la fréquence d'images native de la vidéo immersive soit de 90 im/s, ce n'est pas une fréquence d'images supportée par la plupart des moniteurs broadcast.

2 Réglez le format pour qu'il corresponde à la fréquence d'images de votre moniteur, par ex. 30p.



Mises à l'échelle

Le paramètre Mises à l'échelle vous assurera que la résolution et le rapport d'image des médias immersifs sont correctement affichés dans le viewer. Ces paramètres ont été automatiquement configurés lorsque vous avez activé l'option de workflow Apple Immersive dans la configuration du projet. Toutefois, il est judicieux de vérifier qu'ils sont corrects.

1 Sélectionnez Mises à l'échelle dans la barre latérale des paramètres du projet.

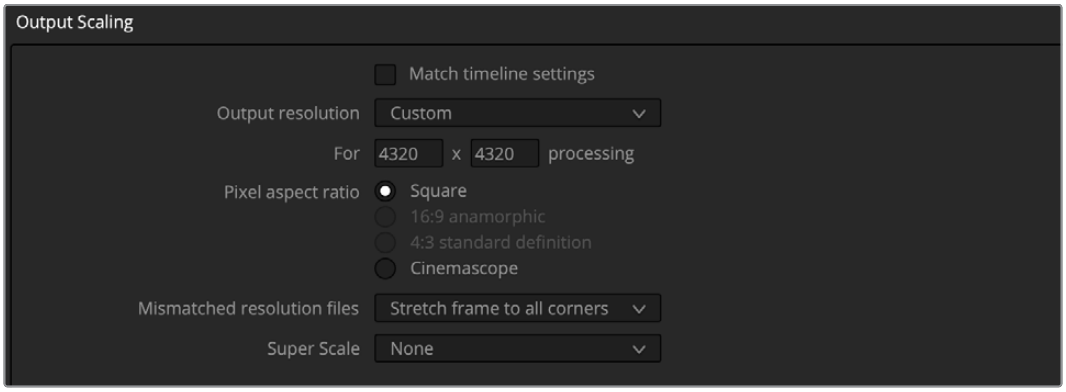
2 Dans la section Mise à l'échelle en Sortie, assurez-vous que l'option **Copier les paramètres de la timeline** n'est pas sélectionnée.

3 Assurez-vous que la **Résolution de sortie** est réglée sur Personnalisée et sur 4320 x 4320.

Réduisez la résolution de moitié (2160 x 2160) pour les écrans plus petits et/ou une lecture plus rapide.

Ces résolutions au rapport d'image carré sont requises pour lire du contenu vidéo Apple Immersive dans DaVinci Resolve Studio et pour la diffusion sur l'Apple Vision Pro (via wifi ou le bracelet développeur).

4 Assurez-vous que le paramètre **Résolutions différentes** est réglé sur **Déformer l'image avec mise à l'échelle**.



Gestion de la couleur

En raison de la nécessité d'un contrôle personnalisé sur la luminance maximale en nits, il est recommandé d'utiliser la gestion de la couleur à l'aide des nœuds Color Space Transform (CST) dans la page Étalonnage. L'objectif des paramètres de gestion de la couleur dans les paramètres du projet est de définir l'espace colorimétrique de la timeline (ou « de travail ») de la page Étalonnage et de régler les métadonnées de l'espace colorimétrique qui seront associées au fichier rendu.

1 Dans la barre latérale des paramètres du projet, sélectionnez Gestion de la couleur.

2 Laissez DaVinci YRGB dans **Colorimétrie**.

3 Réglez **Espace colorimétrique de la timeline** sur DaVinciWG/Intermediate.

Cet espace colorimétrique étendu et évolutif prend en charge les workflows HDR. Il est donc idéal pour les médias de la Blackmagic URSA Cine Immersive et les livrables pour l'Apple Vision Pro.

4 Réglez l'espace colorimétrique de destination sur **P3-D65 ST2084 1000 nits**.

Cet espace colorimétrique offre le gamut approprié pour le monitoring dans les viewers DaVinci Resolve sur un Mac Studio Desktop, habituellement en Display P3. Si vous utilisez un moniteur d'étalonnage ou un profile ICC Apple différent, il faudra changer l'espace espace colorimétrique de sortie afin qu'il corresponde à la norme de votre écran.

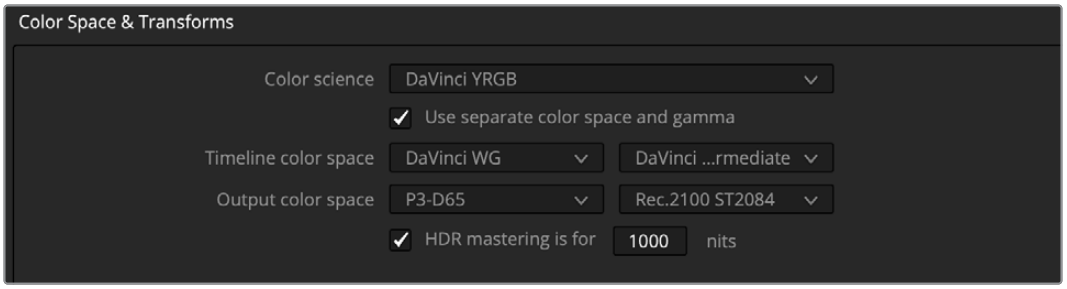
Toutefois, vous devrez changer le gamma (fonction de transfert) afin de mapper correctement le profil ICC.

5 Cliquez sur **Utiliser un gamut et un gamma séparé** au-dessus des paramètres de l'espace colorimétrique.

Cela sépare les menus déroulants de l'espace colorimétrique et du gamma.

6 Réglez le gamma de destination (dans la colonne de droite) sur **Rec.2100 ST2084**.

7 Sélectionnez **Mastering HDR pour 1000 nits**.



Pour passer moins de temps à configurer votre prochain projet Apple Immersive, vous pouvez sauvegarder les paramètres actuels en tant que préréglage.

8 Allez dans le menu Options (trois points) dans le coin supérieur droit de la fenêtre Paramètres du projet et choisissez **Sauvegarder les paramètres actuels comme préréglage**.

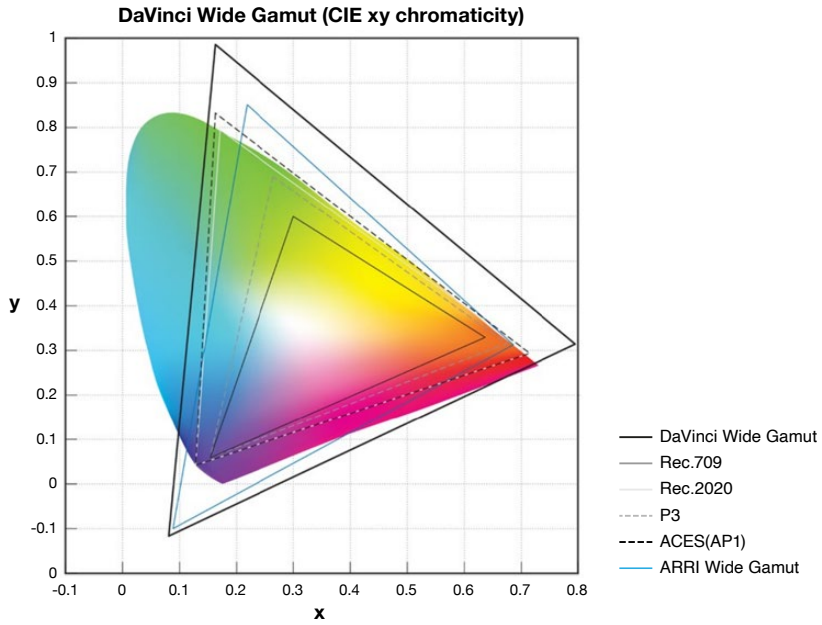
9 Nommez le préréglage et cliquez sur OK.

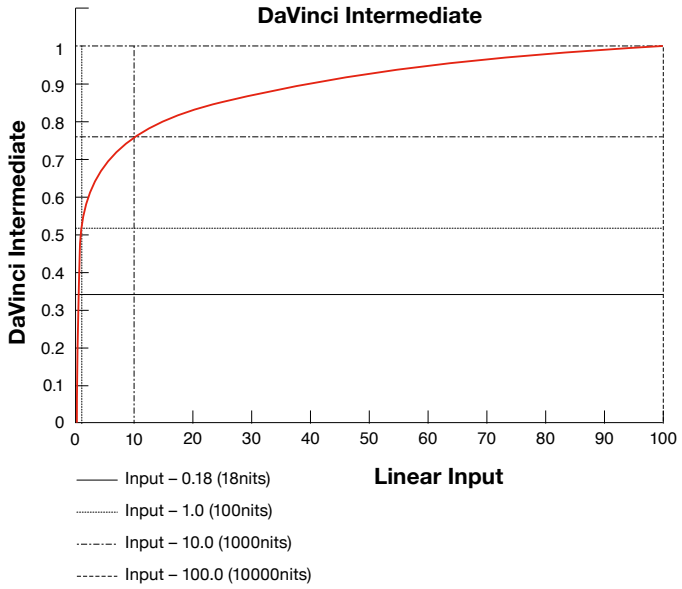
Pour configurer rapidement les paramètres du projet la prochaine fois que vous travaillerez avec des médias immersifs, ouvrez la fenêtre Paramètres du projet, cliquez sur le menu Options et choisissez le préréglage listé.

10 Cliquez sur Enregistrer pour fermer la fenêtre.

À propos du DaVinci Wide Gamut

Le DaVinci Wide Gamut est l'espace colorimétrique interne avancé de Blackmagic Design, qui a été développé pour DaVinci Resolve. Ce dernier offre un environnement extrêmement large et précis pour l'étalonnage et le traitement d'image. Associé au gamma DaVinci Intermediate, il forme un espace de travail unifié qui simplifie la gestion des couleurs entre différents formats de caméra, en garantissant un mappage cohérent des tons et de la saturation, de l'entrée à la sortie. Idéal pour travailler avec les médias HDR et immersifs, ce gamut étendu permet aux étalonneurs de faire des ajustements subtils sans risque de bandes ou d'écrêtage tonal lors du mastering pour des formats haute fidélité.





Traitement du contenu Apple Immersive

Dans la barre latérale des paramètres du projet, la section Apple Immersive vous permet de régler des ILPD (données de profil d'objectif immersif) personnalisées pour modifier la façon dont le contenu immersif est calibré et projeté.

1. Sélectionnez Apple Immersive dans la barre latérale des paramètres du projet.
2. Cliquez sur **Open Calibration Folder**.
3. Ajoutez ou mettez à jour un fichier ILPD personnalisé.
4. Cliquez sur **Update Lists** pour actualiser le contenu ILPD accessible dans la bibliothèque de médias.
5. Cliquez sur Save pour fermer la fenêtre.
6. Dans la bibliothèque de médias, faites un clic droit sur un clip immersif.
7. Dans le menu contextuel, naviguez jusqu'à Apple Immersive Calibration et choisissez la source ILPD (No Apple Immersive Calibration File Selected), Custom ILPD ou Gap ILPD.

Par défaut, le dossier de calibrage des ILPD personnalisées est situé dans le dossier ProIM, avec le dossier LUT par défaut.

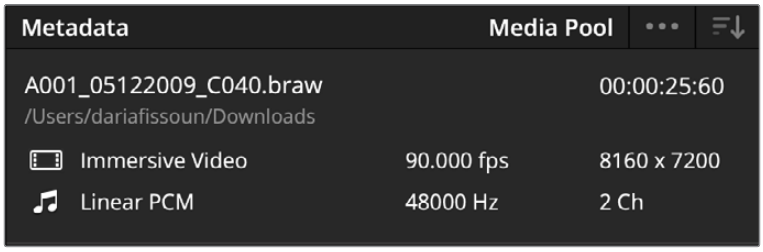
Importation

Lors de l'importation de contenu immersif RAW, les deux yeux et toutes les métadonnées immersives associées sont acheminés avec le clip .braw importé.

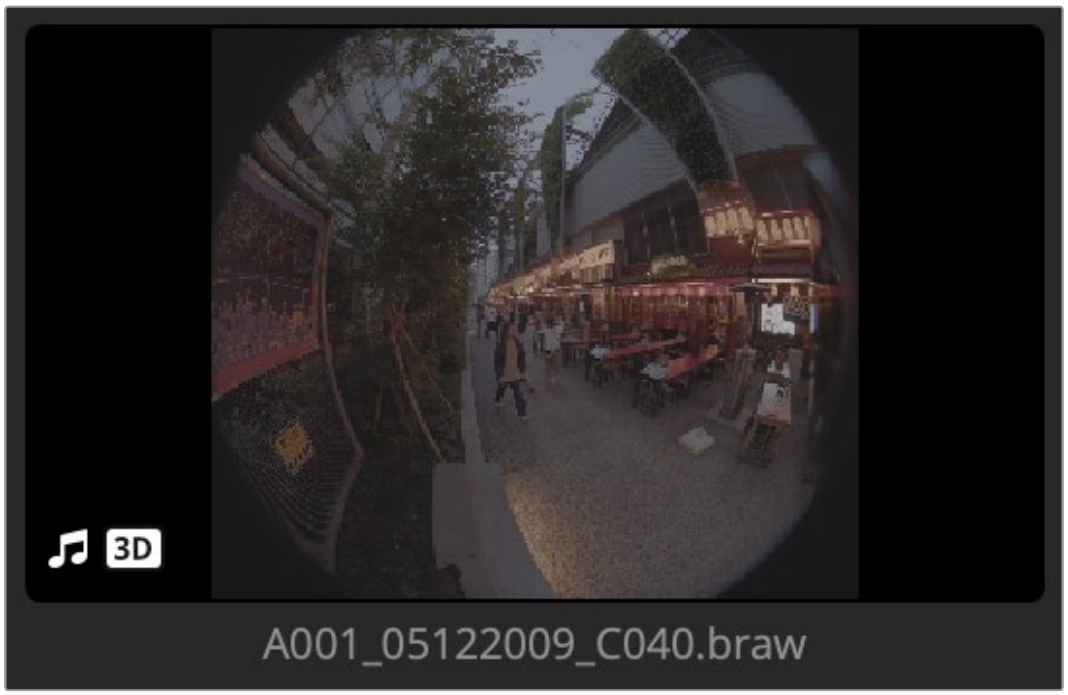
1. Allez sur la page Média.
2. Dans la fenêtre Espaces de stockage, allez à l'emplacement du contenu vidéo immersif.
3. Glissez-déposez les médias dans la bibliothèque de médias au-dessous.

CONSEIL : Pour conserver la structure originale des dossiers, faites glisser les dossiers de la fenêtre Espaces de stockage vers la barre latérale Bibliothèque de médias. Lorsqu'ils sont déplacés dans le Chutier Master ou dans n'importe quel chutier créé, les dossiers et les sous-dossiers sont conservés.

Lorsqu'un clip vidéo Apple Immersive est sélectionné, le paramètre du codec dans l'en-tête de la fenêtre Métadonnées indiquera qu'il s'agit de vidéo immersive, ainsi que sa fréquence d'images et sa résolution natives.



De plus, une icône 3D dans le coin inférieur droit de la vignette vidéo confirmera également qu'il s'agit d'un clip 3D.



REMARQUE: Lorsque vous faites glisser des médias immersifs dans la bibliothèque de médias pour la première fois, une fenêtre de dialogue vous demandera peut-être de choisir si vous souhaitez conserver les paramètres ou les modifier. Si vous choisissez de les modifier, les paramètres du projet adopteront automatiquement la fréquence d'images et la résolution des médias source.

Montage

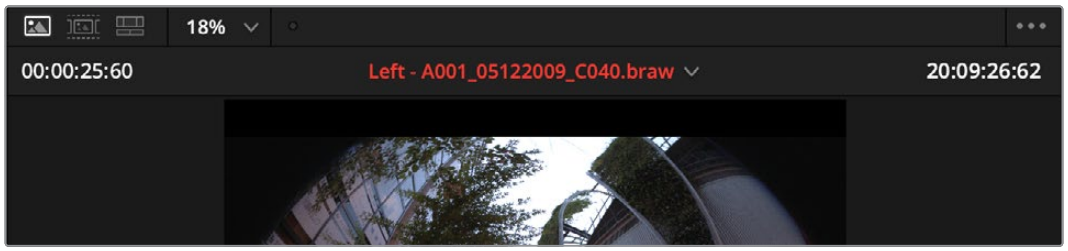
Lorsque vous travaillez avec du contenu immersif, les outils et les actions de montage sont principalement les mêmes que lorsque vous travaillez avec des formats vidéo 2D conventionnels. Les instructions suivantes décrivent les fonctionnalités ou actions de montage qui sont uniques aux workflows vidéo immersifs.

Prémontage

Lorsque vous prévisualisez des clips durant l'assemblage de la timeline, vous pouvez choisir l'œil (gauche ou droit) affiché dans le viewer source. Autrement, le processus de prémontage sur la timeline reste identique aux workflows de montage non linéaire standard.

1 Allez sur la page Montage.

2 Double-cliquez sur un clip pour le charger dans le viewer source à gauche. Le libellé en haut du viewer indique l'œil affiché.



3 Pour changer l'œil dans le viewer, faites un clic droit sur le clip source dans la bibliothèque de médias et choisissez 3D Stéréoscopique > Mode 3D Stéréoscopique > Œil gauche/Œil droit/Les deux yeux.

CONSEIL : Vous pouvez aussi survoler les vignettes de la bibliothèque de médias pour faire défiler les clips dans le viewer source lorsque la fonction Prévisualisation en direct est activée dans les options du viewer source.

4 Utilisez les raccourcis I et O pour définir une plage dans le viewer source.

5 Appuyez sur Insérer le plan dans la barre d'outils (ou utilisez le raccourci F9) pour insérer les clips dans la timeline.

CONSEIL : Vous pouvez aussi glisser-déposer des clips de la bibliothèque de médias à la fenêtre de la timeline.

6 Utilisez votre souris (en mode Sélection ou Trim) pour rogner des clips sur la timeline.

Arrière-plans

Lorsque vous sélectionnez **Enable Apple Immersive workflow** dans les paramètres du projet, une piste d'arrière-plan apparaît automatiquement sous la piste vidéo 1.

Les arrière-plans désignent la zone que l'observateur voit lorsqu'il regarde au-delà du périmètre du contenu vidéo immersif, généralement limité à un champ de vision de 180 à 210 degrés.

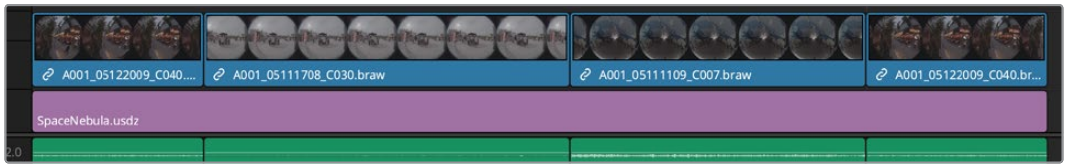
Par défaut, cette zone est noire, mais vous pouvez insérer un arrière-plan personnalisé (au format .usdz) à l'aide de la piste d'arrière-plan.

1 Faites glisser un fichier d'arrière-plan .usdz sur la piste Arrière-plans pour l'ajouter à la timeline.

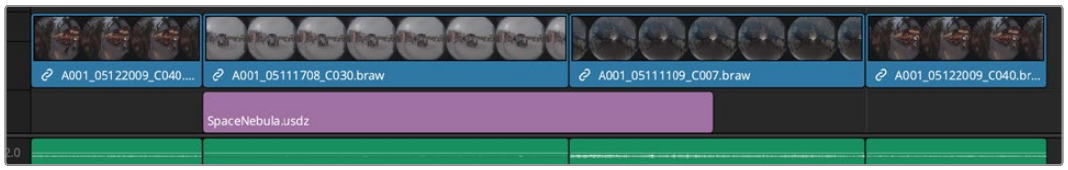
La durée des images d'arrière-plan doit correspondre aux limites des clips dans les pistes vidéo. Un seul fichier d'arrière-plan peut couvrir un clip, plusieurs clips ou la timeline entière. Toutefois, il ne peut pas s'insérer au milieu d'un clip immersif .braw (L-cut).



✓ **Un clip - correct**

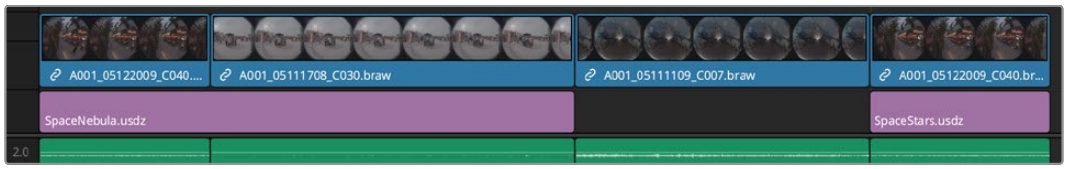


✓ **Plusieurs clips - correct**



✗ **L-cut - incorrect**

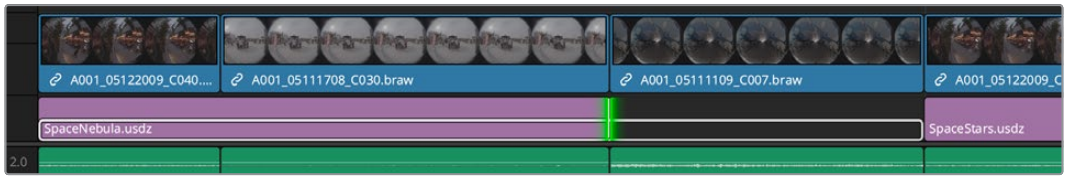
Vous pouvez choisir d'avoir plusieurs fichiers d'arrière-plan dans une seule timeline, ou de ne pas ajouter d'image du tout et d'utiliser l'arrière-plan noir par défaut.



✓ **Différentes images et espaces - correct**

2 Utilisez les commandes de rognage standard pour ajuster la longueur du fichier d'arrière-plan.

Faites glisser le bord du fichier .usdz (en mode Sélection ou Trim) pour ajuster sa longueur.



Vous pouvez utiliser les commandes , (virgule) et . (point) pour déplacer le bord rogné image par image.

CONSEIL : Pour que les arrière-plans soient toujours alignés avec les coupes de vidéo immersive, activez le mode Snap dans la barre d'outils de la page Montage.

Utiliser le viewer

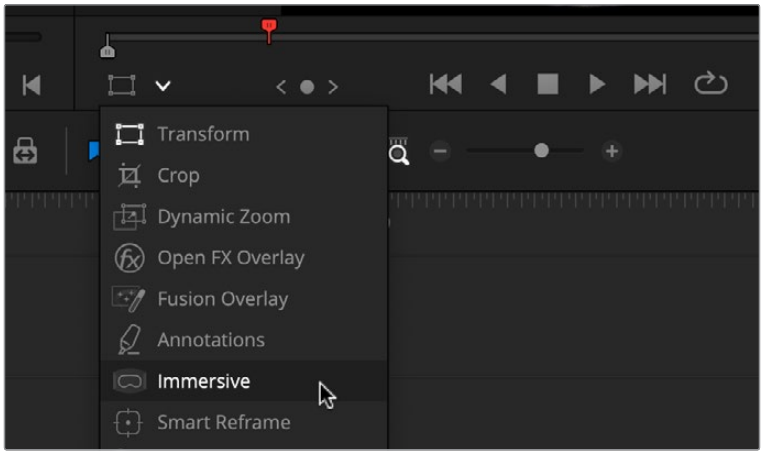
Par défaut, les clips immersifs sont affichés dans les viewers DaVinci Resolve dans l'espace d'objectif circulaire. Le calibrage de l'objectif utilisé pour projeter les images provient de la caméra et est unique à chaque clip.



Vous pouvez aussi changer le mode d'affichage en LatLong ou Viewport.

1 En bas à gauche du viewer timeline, cliquez sur la flèche du menu déroulant à côté du bouton des commandes à l'écran.

2 Sélectionnez Immersive.



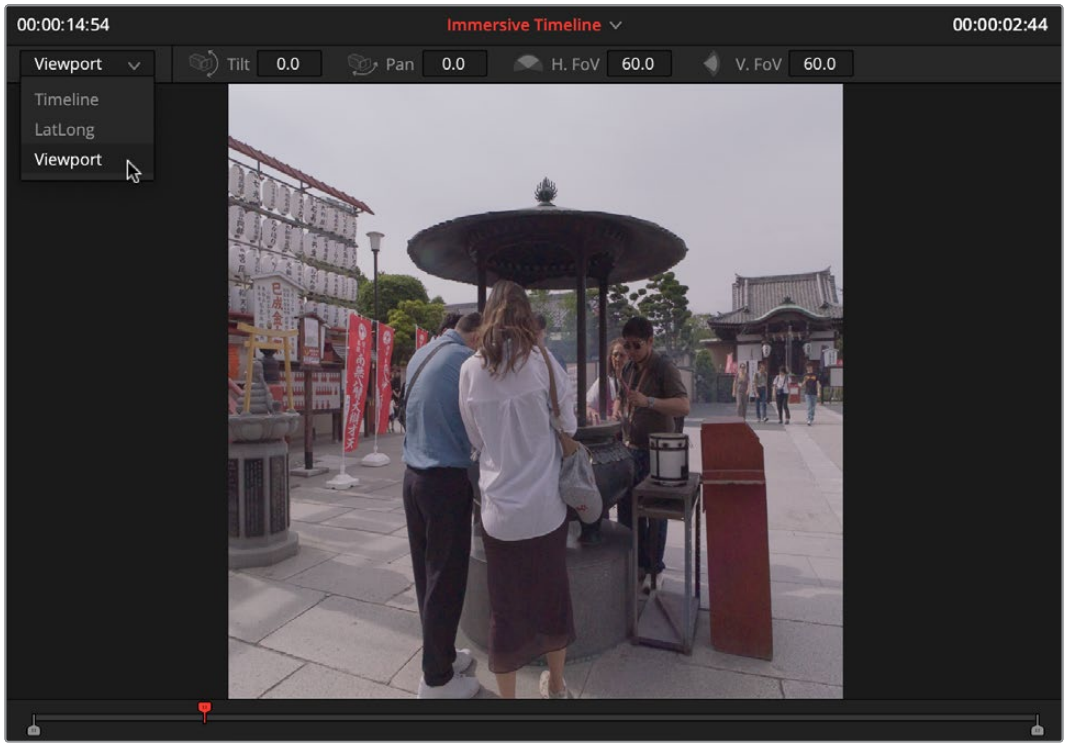
Une nouvelle barre d'outils apparaît au-dessus du viewer avec des options d'affichage supplémentaires.

3 Cliquez sur LatLong pour afficher une projection équirectangulaire de l'image. Ce mode déforme l'image afin que ses proportions ressemblent davantage à celles des médias 2D traditionnels, ce qui facilite certains processus de post-production, tels que le tracking et le compositing.



4 Cliquez sur Viewport pour afficher une découpe centrée de la scène immersive dans un cadre carré. Ce mode vous permet de visualiser clairement les détails correctement proportionnés de la scène sur un moniteur standard. Toutefois, il faudra être attentif à ne pas manquer des éléments hors cadre.

Vous pouvez utiliser les commandes de conversion dans la barre d'outils immersive en haut pour déplacer l'image stéréoscopique dans le cadre, ainsi que pour ajuster les champs de vision horizontaux et verticaux.



Transitions

Si vous souhaitez faire une transition entre deux clips vidéo immersifs, vous pouvez opter pour une coupe franche, utiliser les poignées de chaque côté du clip pour faire un fondu d'ouverture ou de fermeture, ou utiliser un fondu vers une couleur. Les transitions ne devraient pas être incrustées dans les séquences, mais rendues par l'Apple Vision Pro durant la lecture.

1 Ouvrez la fenêtre d'effets.

2 Dans la barre latérale, allez sur Boîte à outils > Transitions vidéo.

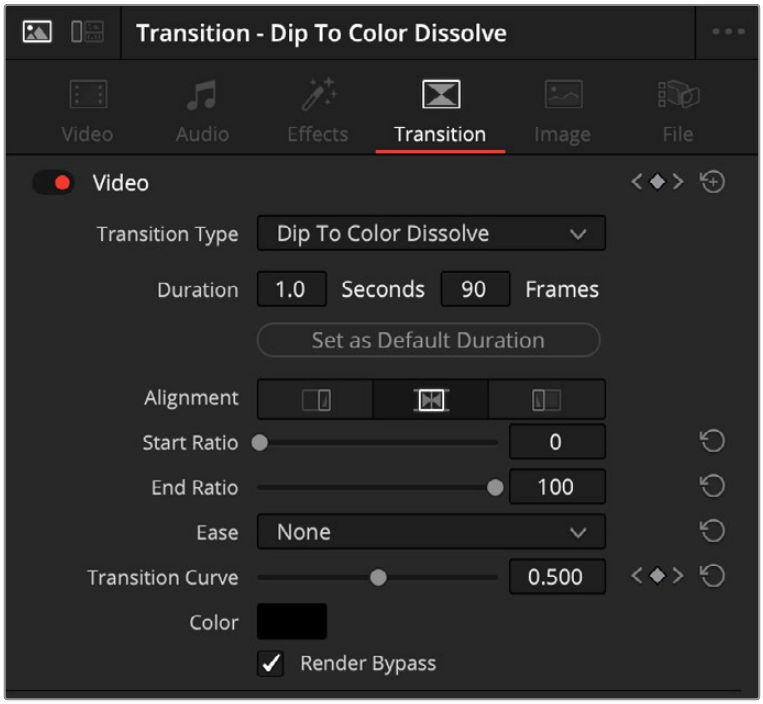
3 Faites glisser le Fondu vers une couleur sur un point de coupe entre deux clips.



REMARQUE : Vous devez disposer de suffisamment de données de trim au-delà des points de coupe des deux clips pour appliquer une transition. S'il n'y a pas suffisamment de données de trim (poignées), la transition ne sera pas appliquée.

4 Dans la fenêtre Inspecteur, allez sur l'onglet Transition et sélectionnez Render Bypass. Ainsi, le fondu sera correctement rendu par l'Apple Vision Pro.

5 Vous pouvez ensuite utiliser les paramètres de l'Inspecteur Fondu vers une couleur pour modifier la durée, l'alignement, l'animation, la vitesse et la couleur de la transition.



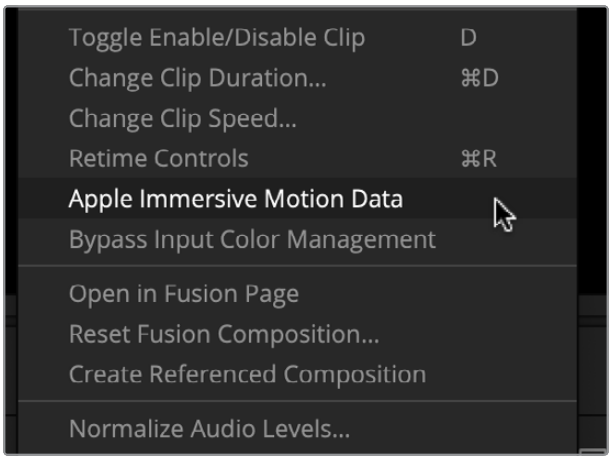
Apple Immersive Motion Data

Lorsque vous travaillez avec du contenu immersif, un mouvement de caméra rapide ou soudain peut désorienter le spectateur. Pour assurer son confort, évitez d'utiliser des prises avec des panoramiques/inclinaisons rapides, des mouvements de rotation ou des horizons inégaux.

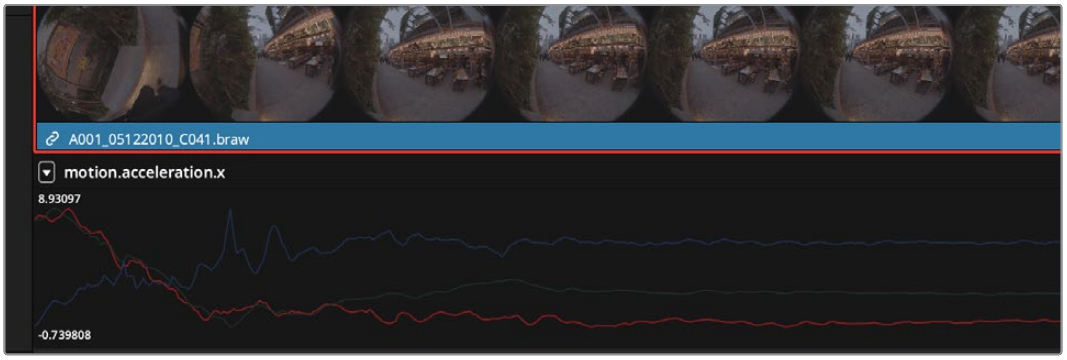
La fonctionnalité Apple Immersive Motion Data représente graphiquement les taux d'accélération x, y et z, ainsi que les valeurs de position du gyroscope, pour illustrer de manière objective les passages où les mouvements de caméra pourraient être trop brusques. Les monteurs peuvent utiliser ces informations pour éviter d'utiliser des passages susceptibles de désorienter le spectateur.

1 Faites un clic droit sur un plan immersif dans la timeline.

2 Choisissez Apple Immersive Motion Data.

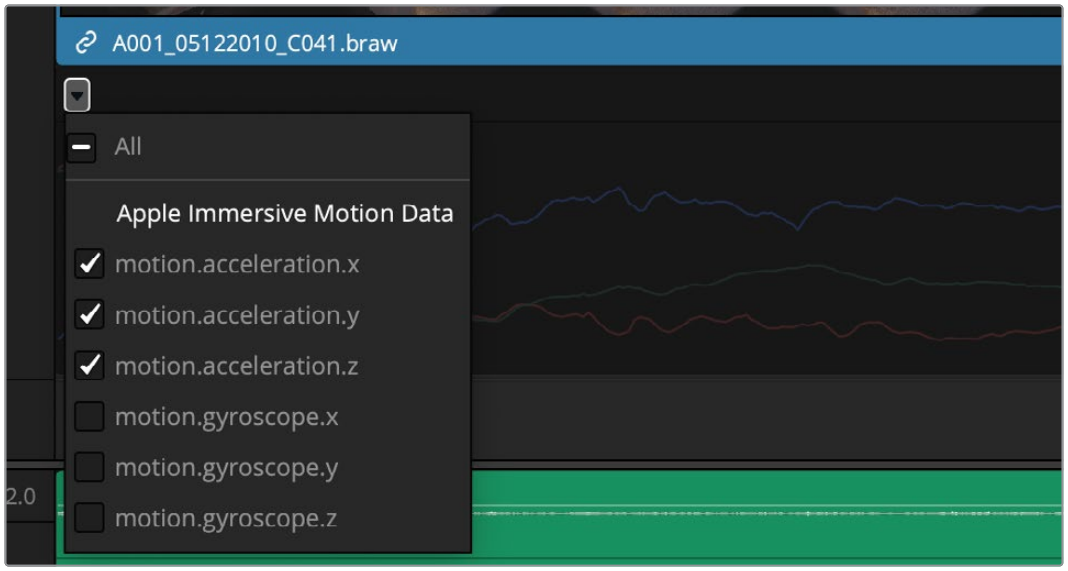


Un graphique apparaît dans la piste vidéo, montrant les taux d'accélération x, y, z.



Dans cet exemple, la caméra enregistrait pendant que l'opérateur l'installait. Les pics indiquent un mouvement soudain lorsque la caméra était soulevée, tournée et placée sur son épaule. L'idéal serait d'utiliser la partie finale de la vidéo, celle qui présente des relevés plus stables, une fois la caméra stabilisée.

3 Dans le coin supérieur gauche du graphique, cliquez sur la flèche de déplacement.



Vous pouvez activer ou désactiver les relevés en fonction du contenu de la prise et de la nature du mouvement de la caméra.

Titres

Dans la page Montage, vous pouvez ajouter du texte à une scène immersive en utilisant n'importe lequel des outils de titre du panneau Effets — Text, Text+, MultiText, Scroll, et toutes les variétés de synthés.

Lorsqu'il est placé au-dessus d'un plan immersif, le générateur de titre utilisera les données de profil d'objectif immersif (ILPD) du plan pour déformer avec précision le texte afin qu'il corresponde à la distorsion de l'objectif de la scène.

Si aucune ILPD n'est disponible (par exemple, lors de l'ajout de texte sur un fond noir ou une scène CGI immersive), une distorsion d'objectif par défaut sera appliquée pour garantir que le texte apparaisse toujours correctement déformé lorsqu'il est visualisé dans l'Apple Vision Pro.

1 Dans le panneau Effets, naviguez vers Boîte à outils > Titres.

2 Faites glisser le générateur Texte dans la timeline. Placez-le sur une piste vidéo au-dessus d'un plan immersif pour l'intégrer à la scène.



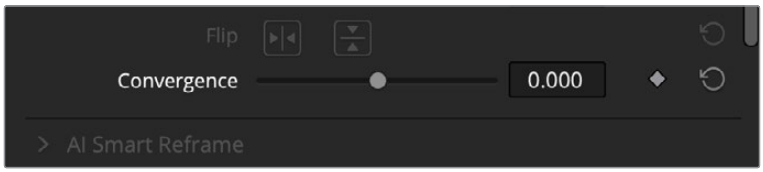
Comme pour les arrière-plans, les générateurs de texte ne peuvent pas franchir les limites des clips. Étant donné que chaque clip est rendu avec un espace d'objectif individuel, si des titres se chevauchent, ils risquent d'être déformés entre les plans.

3 Rognez le titre au(x) point(s) de coupe vidéo.

4 Utilisez l'Inspecteur pour écrire et formater le texte du titre.

5 Allez à l'onglet Paramètres et utilisez le curseur Convergence pour positionner le texte dans l'espace 3D.

Déplacer le curseur vers la gauche recule le titre dans l'espace, et le déplacer vers la droite le rapproche.



La convergence s'observe mieux sur l'Apple Vision Pro. Si vous utilisez d'autres moyens de monitoring, vous pouvez modifier la façon dont la convergence du texte apparaîtra dans le viewer en changeant le mode Vision de la couche vidéo sous-jacente dans la page Étalonnage.

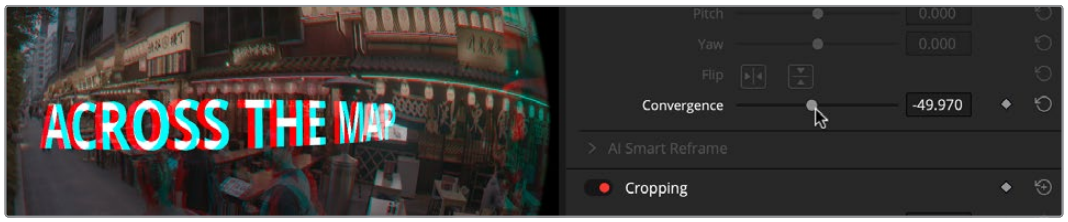
6 Allez sur la page Étalonnage.

7 Ouvrez la palette 3D.

La colonne Vision au centre de la palette vous permet de modifier la norme stéréoscopique utilisée pour projeter le média immersif.

8 Sélectionnez l'onglet Stéréo.

9 Changez le mode Sortie en Anaglyphe (Couleur).



La modification du paramètre Convergence du générateur de texte dans le panneau Inspecteur de la page Montage sera désormais plus intuitive si elle est visualisée à l'aide de lunettes anaglyphes (ou anachromes).

Considérations créatives

En cette phase de lancement de la caméra Blackmagic URSA Cine Immersive et de l'Apple Vision Pro, les réalisateurs sont invités à expérimenter avec ce format pour identifier les meilleures pratiques, établir de nouvelles règles et découvrir des workflows que le cinéma traditionnel n'avait pas envisagés.

Dans cette perspective, certains aspects de la production immersive ont déjà été identifiés comme nécessitant une attention particulière, afin d'assurer une immersion maximale du public et un confort visuel optimal.

1 Le montage doit commencer dès la pré-production. En raison des spécificités liées à la durée des plans et au confort de mouvement pour le public, les réalisateurs doivent filmer les scènes en ayant à l'esprit le montage final.

2 Évitez ou limitez l'utilisation de clips avec des mouvements de caméra rapides, car ils peuvent désorienter le spectateur. Privilégiez plutôt les plans fixes, les panoramiques lents et les travellings.

3 Les angles de caméra déséquilibrés ou extrêmes peuvent également provoquer un sentiment de désorientation. Optez pour des prises de vue à niveau avec le sujet et à une distance confortable (pas trop rapprochée).

4 Les nouveaux environnements prennent plus de temps à être assimilés dans les médias immersifs que lors du visionnage d'une scène en 2D. Les coupes doivent être plus lentes pour laisser au spectateur le temps de s'adapter à son nouvel environnement.

5 Lors des coupes entre les clips, tenez compte de l'endroit où le spectateur regardait dans les plans précédents. Une action importante ne doit pas se dérouler sur le côté gauche du cadre si l'attention du public était auparavant concentrée sur la droite.

6 L'audio spatial est un important moteur narratif. Considérez-le comme un personnage à part entière dans les productions immersives.

Compositing VFX

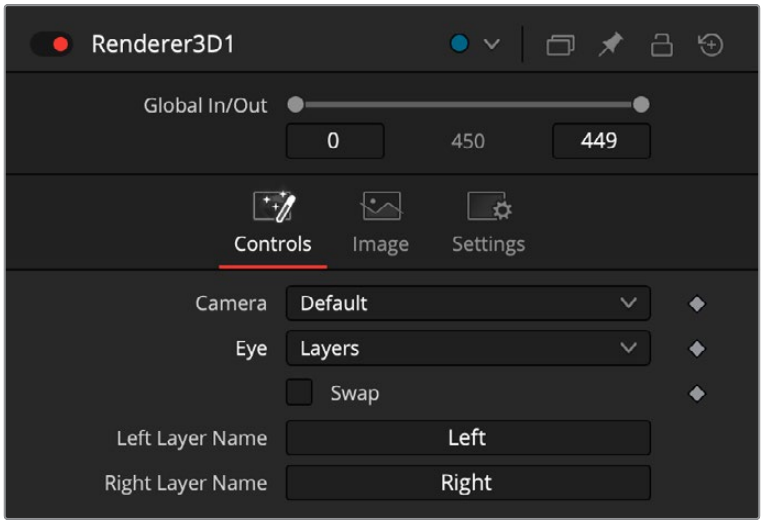
Lorsque **Apple Immersive workflow** est activé dans les paramètres du projet, des réglages dédiés dans Fusion permettent des workflows stéréoscopiques plus aisés et un compositing conforme aux exigences immersives. Les nœuds MediaOut offrent des options de sortie pour l'œil gauche et l'œil droit, et l'outil Immersive Patcher permet une projection et une composition équirectangulaires.

Nœuds MediaOut

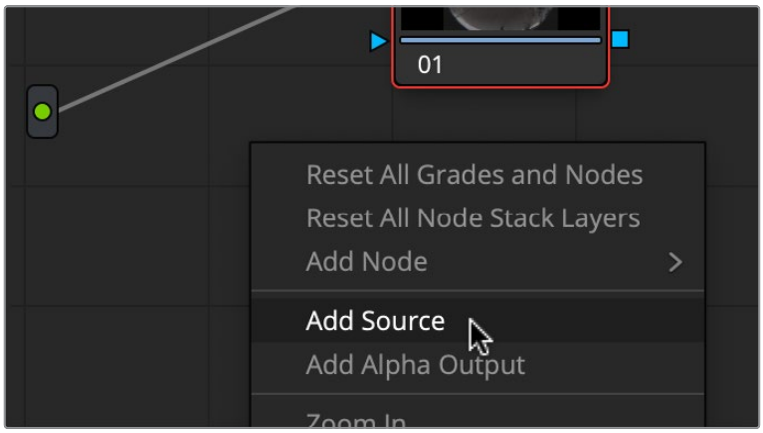
Lorsque vous entrez pour la première fois dans la page Fusion en travaillant sur un clip immersif, le graphique des nœuds contient deux nœuds MediaIn connectés à deux nœuds MediaOut correspondants.



Dans l'Inspecteur MediaOut, le paramètre Œil stéréo peut être utilisé pour acheminer le signal de sortie de l'œil.



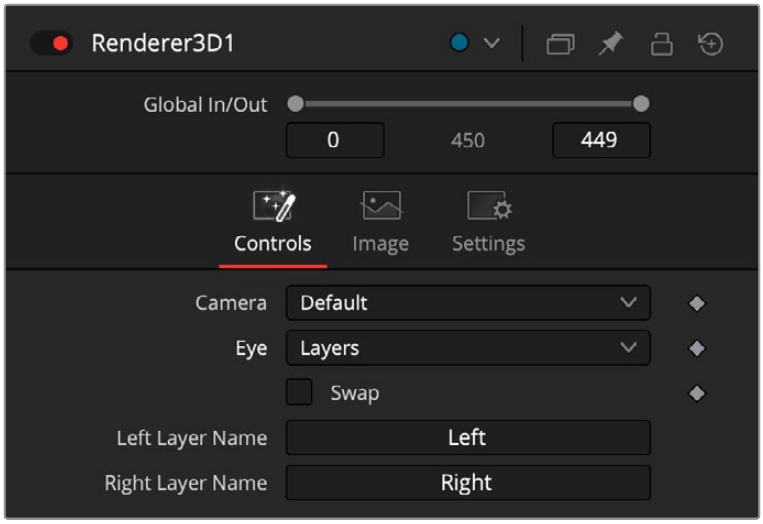
Par défaut, MediaOut1 est configuré pour l'œil gauche et MediaOut2 pour l'œil droit. Cette configuration vous permet de commencer immédiatement à assembler des compositions et des incrustations graphiques compatibles avec la stéréo. Les deux nœuds MediaOut permettent également d'envoyer deux variantes de signal à l'éditeur de nœuds de la page Étalonnage pour un affinement ultérieur de l'image. Vous pouvez accéder au signal MediaOut2 dans la page Étalonnage en faisant un clic droit dans l'éditeur de nœuds et en sélectionnant **Ajouter une source**.



Lorsque le paramètre **Œil stéréo** est réglé sur **Auto**, le nœud MediaOut combine les signaux et les attribue aux yeux appropriés sur la timeline. Cela vous permet d'utiliser le workflow multicouche de Fusion pour envoyer les signaux gauche et droit à un seul nœud MediaOut.

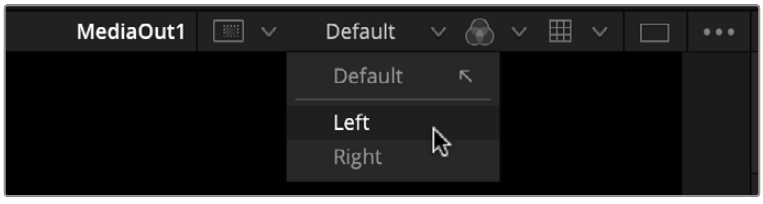
Renderer3D

Lorsque vous travaillez avec des graphiques 3D ou des CGI sur une timeline immersive, le nœud Renderer3D propose plusieurs configurations pour les yeux et la capacité de traiter les couches gauche et droite.



Cela permet de créer des titres 3D complexes ou des scènes CGI complètes et de les envoyer directement à une timeline stéréoscopique dans les projets immersifs.

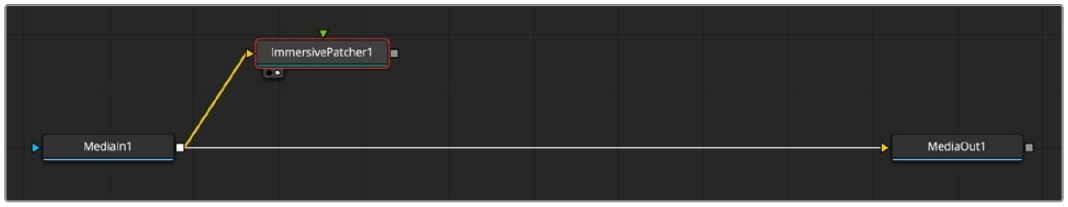
Avec cette configuration, vous pouvez vous servir du menu déroulant dans le coin supérieur droit des viewers Fusion pour alterner entre la visualisation de l'œil gauche et de l'œil droit de votre composition.



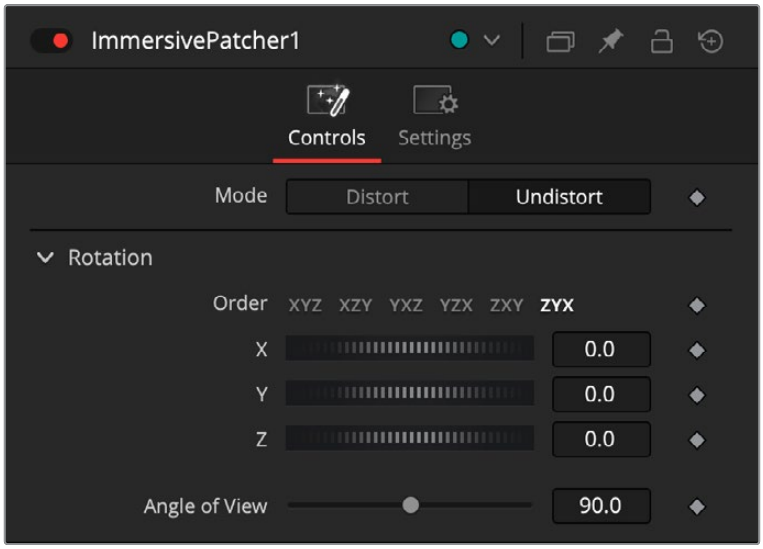
Compositing avec l'Immersive Patcher

L'outil Immersive Patcher vous permet de corriger temporairement la distorsion d'une section d'un clip immersif pour faciliter le tracking, le masquage et le compositing sur un plan plat. Lorsque vous avez terminé, vous pouvez utiliser un second Immersive Patcher pour « re-déformer » le signal immersif et intégrer les nouveaux éléments composités dans la scène.

1 Ajoutez **Immersive Patcher** à la structure nodale.



2 Dans l'Inspecteur, cliquez sur **Undistort**.



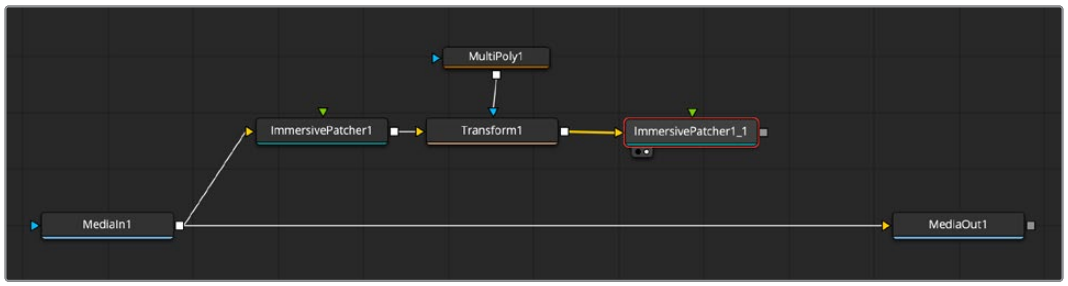
3 Utilisez les paramètres de rotation X, Y et Z pour choisir une section de l'image immersive à aplatir.

4 Procédez au compositing comme vous le feriez avec un signal vidéo 2D standard. Cela comprend (sans s'y limiter) l'utilisation de méthodes de masquage standard pour supprimer les objets indésirables, l'échantillonnage de sections du cadre avec des transformations et des polygones pour recréer un décor, ou l'application de caches, de texte et de calques d'images/vidéos externes pour remplacer des panneaux ou des écrans.

   Une fois que vous avez terminé, vous devez « re-déformer » l'image plate pour qu'elle corresponde à la distorsion immersive d'origine.

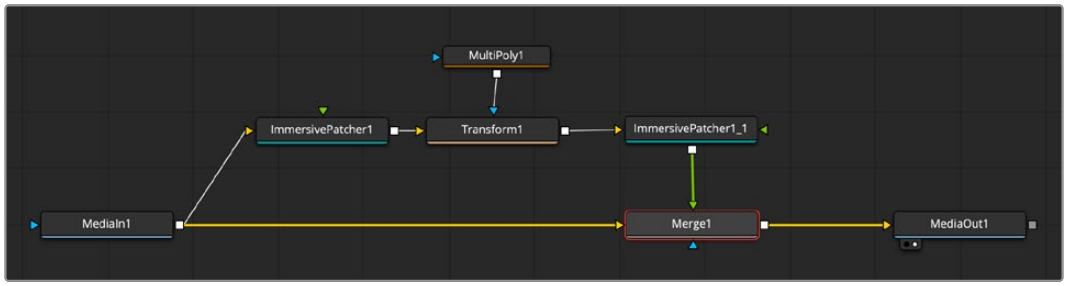
5 Copiez l'Immersive Patcher pour conserver les informations de rotation et d'angle de vue d'origine.

6 Collez l'Immersive Patcher après les nœuds de compositing et connectez-le à la fin du pipeline de composition.



7 Dans l'Inspecteur, cliquez sur **Distort**.

8 Fusionnez le patch composité avec le clip original pour intégrer l'effet à la scène immersive.



Si vous travaillez en collaboration, vous pouvez quitter la page Fusion et continuer à monter, mixer et étalonner la timeline immersive dans DaVinci Resolve.

Pour effectuer des exportations immersives, exportez les yeux individuellement en tant qu'EXR en ACES AP0 afin de conserver les propriétés maximales pour une diffusion immersive.

Étalonnage

Cette section couvre les outils optiques, d'étalonnage et 3D que vous pouvez utiliser en travaillant avec des médias immersifs sur la page Étalonnage de DaVinci Resolve.

En raison de la nature unique du contenu immersif, tous les outils ne pourront pas être utilisés en post-production. Actuellement indisponibles, d'autres outils sont en cours d'adaptation pour prendre en charge le contenu immersif.

Réduction de bruit

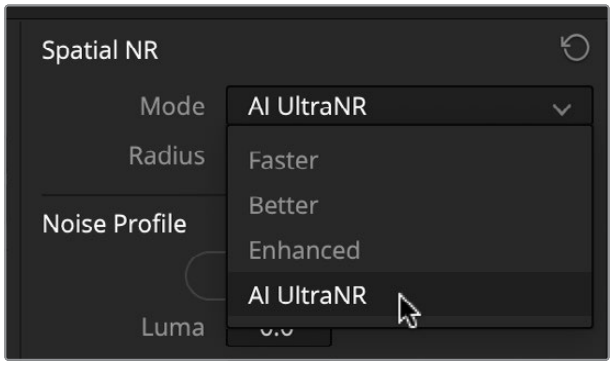
Lors du travail avec la vidéo immersive, il est possible de réduire le bruit. Dans la palette **Effets de mouvement**, la réduction de bruit spatiale analyse le motif de bruit du média source afin de le réduire au maximum tout en conservant les détails de l'image d'origine.

1 Dans la barre d'outils de la page Étalonnage, ouvrez la palette **Effets de mouvement**.

Dans la colonne **Réduction de bruit spatiale**, vous pouvez sélectionner le mode de réduction du bruit que vous souhaitez utiliser.

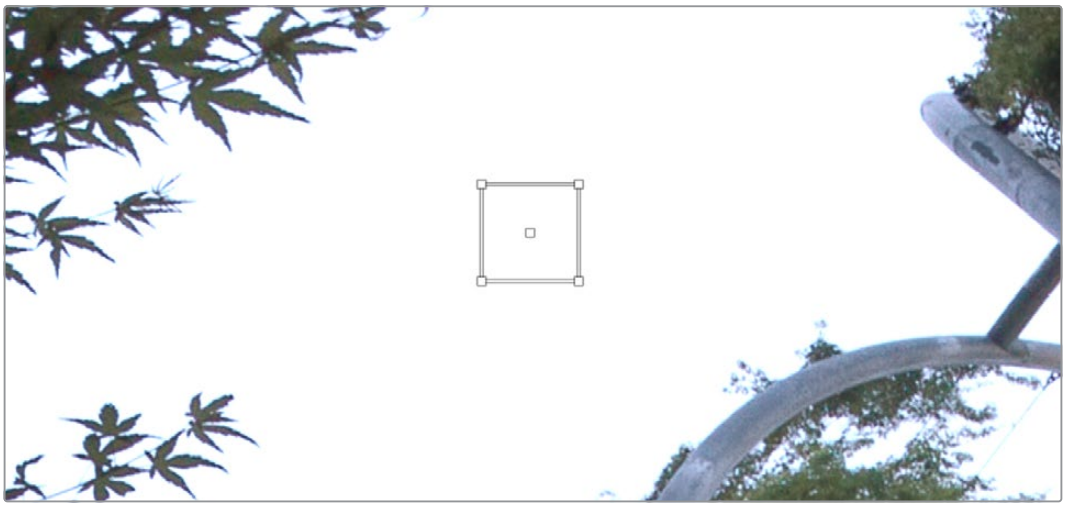
- Les modes Faster, Better, et Enhanced effectuent tous leur analyse de motif dans une taille de rayon donnée et configurent leur sortie en fonction du seuil défini. Ils sont répertoriés par ordre de complexité de traitement.
- *AI UltraNR* est un identifiant de motif de bruit piloté par l'IA qui extrait le motif de bruit du signal en localisant et en analysant la section la plus « plate » du cadre pour le bruit pur.

2 Réglez le mode de réduction de bruit spatial sur **AI UltraNR**.



3 Cliquez sur **Analyser**.

Un cadre de délimitation apparaît dans le viewer, identifiant la zone de données de pixels la plus « vide » dans le cadre donné. Sans contenu visuel tangible, les données à l'intérieur du carré sont idéales pour extraire le bruit numérique produit par le capteur de la caméra.



Vous pouvez repositionner et redimensionner le cadre de délimitation dans le viewer si vous le souhaitez, mais ce n'est généralement pas conseillé. AI UltraNR est meilleur pour détecter les régions d'analyse optimales que nos yeux.

Vous pouvez ajuster manuellement les paramètres de seuil de bruit Luma et Chroma, mais à nouveau, l'outil AI UltraNR a tendance à atteindre automatiquement le mélange idéal de seuils pour une réduction du bruit propre.

Étalonnage primaire

L'étalonnage primaire fonctionne de la même manière qu'avec des séquences vidéo normales. Utilisez les mêmes considérations pour la normalisation, l'équilibrage et la correspondance lors de l'utilisation des roues chromatiques primaires et des roues Log, de la palette HDR et des Courbes HSL. Les palettes ColorSlice et Déformation Couleur fonctionnent également comme prévu.

Corrections secondaires

Les méthodes de sélection basées sur les pixels, comme le Sélecteur, fonctionnent comme prévu avec les contenus immersifs.

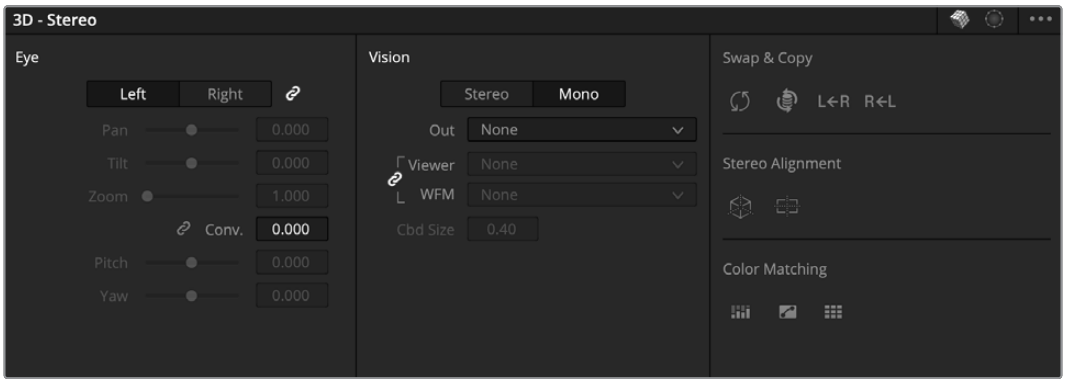
Les sélections basées sur des objets ou des fenêtres ne peuvent fonctionner avec précision que dans un environnement plat équirectangulaire (ou en mode viewport). La plupart des méthodes secondaires nécessitant une sélection précise de forme sont donc actuellement indisponibles. Cela inclut les Power Windows (et, par extension, la palette Tracker) ainsi que l'outil Magic Mask.

Échelle

Aucune commande de redimensionnement (par exemple, zoom ou panoramique) n'est disponible pour les contenus immersifs.

Palette 3D

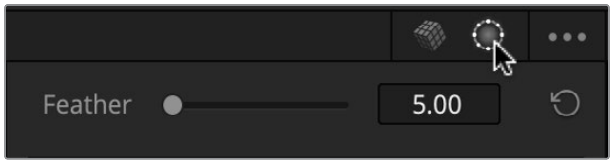
La palette 3D de la page Étalonnage propose des commandes permettant de gérer divers formats de caméras et de graphismes 3D. Contrairement à d'autres formats stéréoscopiques, il n'est pas nécessaire d'ajuster l'alignement des yeux, car les objectifs, la distance focale et l'iris de la Blackmagic URSA Cine Immersive sont fixes.



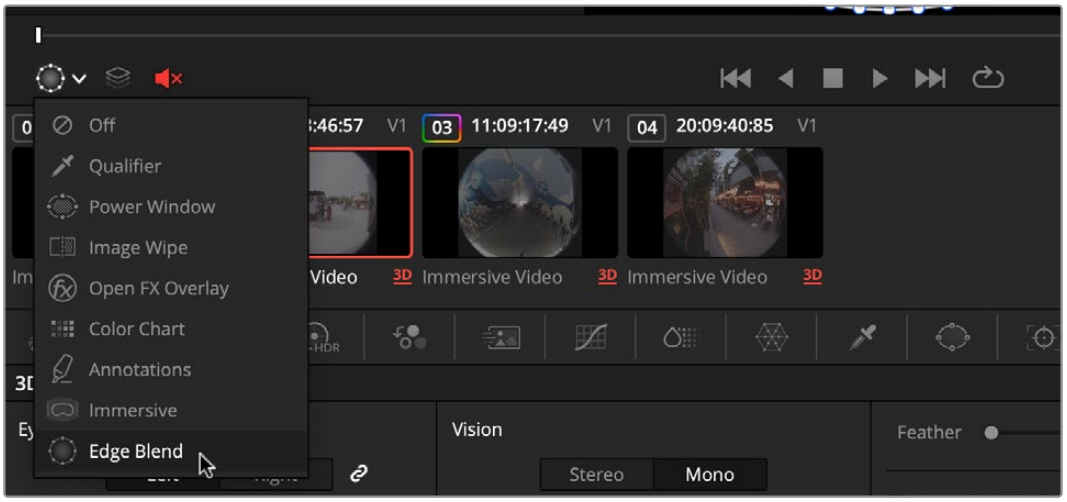
Le mode Edge Blend est exclusif aux contenus immersifs et permet d'ajuster et d'adoucir le périmètre du cadre vidéo immersif. Cela peut servir à dissimuler du matériel de tournage visible et à limiter le champ de vision de l'observateur. Au-delà de ce bord, l'observateur verra l'arrière-plan.

1 Ouvrez la palette 3D.

2 Dans le coin supérieur droit, changez le mode en **Edge Blend**.



3 Assurez-vous que les commandes à l'écran du viewer sont définies sur Edge Blend.



Vous devriez voir un cercle de points de contrôle autour du périmètre du cadre vidéo, avec une poignée centrale.

4 Faites glisser un point de contrôle pour changer sa position,



ou faites un clic droit pour le supprimer.

5 Dans la colonne de droite de la palette 3D, utilisez le paramètre Feather pour adoucir le bord du cadre.

6 Sélectionnez **visionOS Effect** pour que les nouvelles données de bord soient correctement traitées lors de la lecture sur l'Apple Vision Pro. Si cette option n'est pas activée, le bord sera intégré de façon permanente dans le fichier vidéo rendu.

7 Dans la colonne de gauche, utilisez les onglets Droite et Gauche pour commuter entre les vues.

Veuillez noter que de nombreuses commandes de la palette 3D apparaîtront grisées. Dans DaVinci Resolve, un paramètre grisé indique qu'il n'est pas applicable au média actuellement utilisé, comme ici, avec le contenu immersif.

8 Dans la colonne du milieu, sélectionnez l'onglet **Stéréo** ou **Mono** afin d'afficher le clip immersif en mode binoculaire ou monoculaire.

Gestion des couleurs CST

L'utilisation de l'effet Color Space Transform (CST) sur les nœuds dans l'éditeur de nœuds de la page Étalonnage permet un mappage précis des couleurs et un contrôle personnalisé de la luminance maximale en nits de la timeline.

Le workflow recommandé comporte quatre nœuds CST :

1 INPUT CST – de la caméra à l'espace colorimétrique de travail.

La correction colorimétrique se fait entre ces deux nœuds.

2 OUTPUT CST – de l'espace colorimétrique de travail à l'affichage Mac HDR.

3 MONITOR CST – de 1000 nits à 250 nits pour le monitoring HDR.

4 AVP CST – de 250 nits à 108 nits pour le monitoring et l'exportation sur l'Apple Vision Pro.



Mappage de la timeline CST

La première étape consiste à mapper l'espace colorimétrique source de la caméra vers l'espace colorimétrique « de travail » de la timeline. La caméra Blackmagic URSA Cine Immersive utilise le gamut Blackmagic Design Wide Gamut Gen 4/5 avec la courbe log DaVinci Intermediate.

1 Ouvrez le panneau Effets.

2 Glissez Color Space Transform sur un nœud vide (01) dans l'éditeur de nœuds.

Lors du mappage de l'espace colorimétrique d'entrée vers l'espace colorimétrique de la timeline, ce nœud doit être placé parmi les premiers de la structure nodale, avant toute opération d'étalonnage.

Si vous mappez directement de l'espace colorimétrique d'entrée vers l'espace colorimétrique de monitoring ou d'exportation, ce nœud doit être parmi les derniers, après l'étalonnage.

3 Nommez le nœud 01 **INPUT CST**.

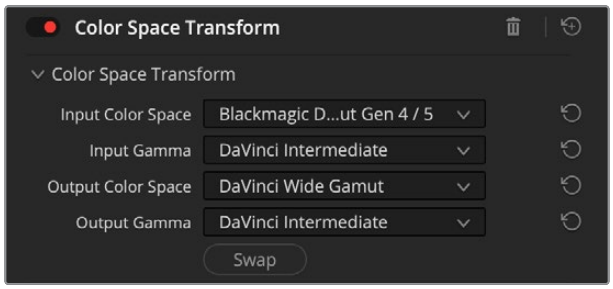
REMARQUE : Les noms de nœuds utilisés dans ces instructions sont donnés pour faciliter la lecture de la documentation et des captures d'écran. Vous pouvez utiliser la convention de dénomination qui vous convient, ou ne pas attribuer de nom du tout.

4 Réglez l'option Espace colorimétrique d'entrée (Input Color Space) sur Blackmagic Design Wide Gamut Gen 4/5.

5 Réglez le gamma d'entrée (Input Gamma) sur DaVinci Intermediate.

6 Réglez l'Output Color Space sur DaVinci Wide Gamut.

7 Réglez le gamma de destination (Output Gamma) sur DaVinci Intermediate.



8 Créez un nouveau nœud en série (02).

9 Copiez-collez le premier nœud (01) dans ce nouveau nœud (02).

10 Nommez le nœud 02 **OUTPUT CST**.

Il faudra que l'espace colorimétrique et le gamma de sortie correspondent à l'entrée de ce nouveau nœud en série, afin de garantir la cohérence de l'espace colorimétrique de la timeline dans toute la structure nodale.

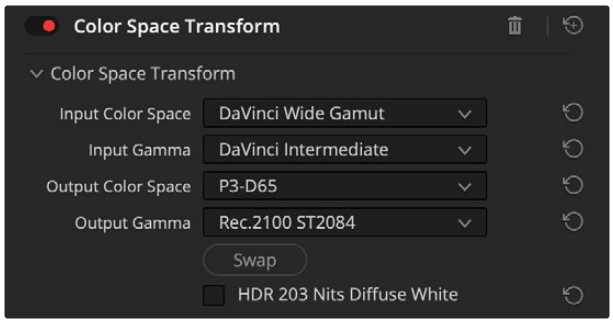
11 Appuyez sur **Échanger** pour inverser les paramètres d'entrée et de sortie.

L'espace colorimétrique et le gamma d'entrée sont désormais appropriés pour ce nœud CST.

12 Réglez l'espace colorimétrique de destination (Output Color Space) sur P3-D65. C'est le gamut par défaut actuel des écrans Mac.

13 Réglez le gamma de destination (Output Gamma) sur Rec.2100 ST2084, la norme HDR pour un signal à 1000 nits.

Toutes les opérations d'étalonnage auront lieu entre ces deux nœuds. Toutefois, comme vous travaillez avec une gestion des couleurs basée sur les nœuds, vous avez la possibilité unique d'effectuer des opérations en dehors des limites de la gestion colorimétrique de la timeline. De nombreux étalonneurs utilisent cette méthode pour appliquer des corrections spécifiques, des effets ou des éléments tiers.



Profils ICC

Pour vous assurer que le signal de la Blackmagic URSA Cine Immersive est correctement mappé à votre écran Mac, allez dans Réglages Système > Moniteurs et vérifiez que le profil colorimétrique prédéfini est réglé sur Apple XDR Display (P3-1600 nits).

Mappage CST pour le moniteur d'étalonnage

Vous pouvez utiliser des CST supplémentaires pour limiter temporairement la luminance maximale en nits pendant l'étalonnage.

1 Créez un nouveau nœud en série (03) après le nœud OUTPUT CST (02).

2 Nommez le nœud 03 **MONITOR**.



3 Ajoutez l'effet Color Space Transform au nœud MONITOR (03).

Les valeurs d'entrée doivent correspondre au signal sortant du CST précédent.

4 Réglez l'espace colorimétrique source (Input Color Space) sur P3-D65.

5 Réglez le gamma d'entrée (Input Gamma) sur Rec.2100 ST2084.

L'objectif de ce nœud CST est uniquement de mapper la luminance de l'image vers une valeur maximale plus faible en nits. Les paramètres d'entrée et de sortie peuvent donc rester identiques.

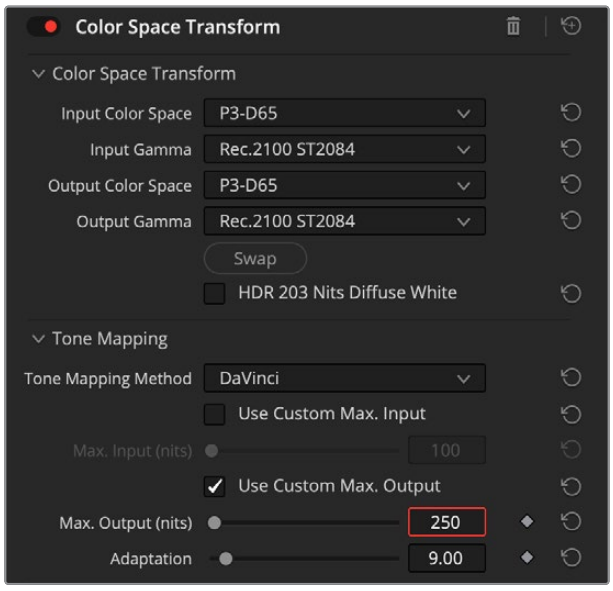
6 Réglez l'espace colorimétrique de destination (Output Color Space) sur P3-D65.

7 Réglez le gamma de destination (Output Gamma) sur Rec.2100 ST2084.

REMARQUE : Dans certains cas, il est inutile de définir l'espace colorimétrique ou le gamma d'un CST si les valeurs d'entrée et de sortie sont identiques (aucun mappage). Ici, cela reste nécessaire car l'application d'un mappage personnalisé en nits dépend de la valeur et de la forme de la courbe log du signal.

8 Sélectionnez **Use Custom Max. Output**.

9 Réglez **Max. Output (nits)** sur 250.



Le signal de 1000 nits sera désormais mappé sur 250 nits grâce à la méthode de « mappage tonal DaVinci ».

On obtiendra ainsi une image correspondant fidèlement au signal rendu de l'Apple Vision Pro sur un moniteur d'étalonnage HDR (par exemple, le Flanders XMP310, le Sony BVM-HX310, etc.).

Cependant, pour visualiser votre projet sur un Apple Vision Pro, vous devrez effectuer une opération de mappage tonal supplémentaire.

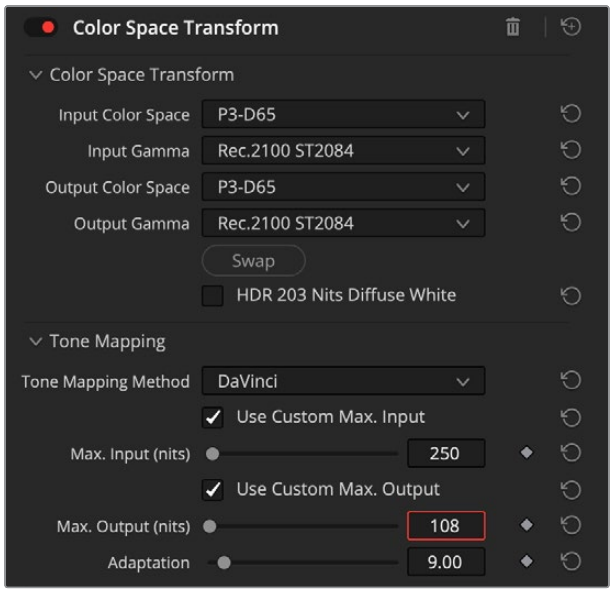
Mappage CST pour l'Apple Vision Pro

Un dernier nœud CST peut être utilisé pour permettre le monitoring en direct, l'aperçu et l'exportation vers les Apple Vision Pro.

1. Créez un nouveau nœud en série (04) après le nœud MONITOR (03).
2. Copiez-collez le nœud MONITOR (03) dans le nœud 04. Vous utiliserez les mêmes paramètres d'entrée et de sortie, en ne changeant que la valeur en nits.
3. Renommez le nœud 04 **AVP**.



4. Sélectionnez **Use Custom Max. Input**.
5. Réglez **Max. Output (nits)** sur 250 pour interpréter correctement le signal mappé depuis le nœud MONITOR.
6. Réglez **Max. Output (nits)** sur 108, ce qui correspond à la valeur maximale en nits pour le monitoring et l'exportation pour l'Apple Vision Pro.



Vous pouvez maintenant procéder à l'étalonnage, en désactivant les nœuds CST si (et quand) cela est nécessaire en fonction de votre workflow. Par exemple, lors de votre premier étalonnage, au cours duquel vous utiliserez votre moniteur d'étalonnage externe, vous devriez désactiver le nœud AVP (04) pour vous assurer de visualiser correctement le signal de 250 nits sur votre moniteur d'étalonnage HDR. Lorsque vous passez à la prévisualisation dans l'Apple Vision Pro ou que vous êtes prêts pour l'exportation, réactivez le nœud AVP (04).

Les CST peuvent-ils être placés dans l'éditeur de nœuds de la timeline ?

Non. Pour le moment, le mappage tonal du clip repose sur l'application précise du paramètre d'adaptation de la transformation de l'espace colorimétrique, qui doit s'effectuer clip par clip.

Audio

L'Apple Vision Pro et DaVinci Resolve prennent en charge l'audio spatial et ambisonique, les formats AVP (Apple Vision Pro) et AIA (Apple Immersive Audio).

Le travail en audio spatial ne modifie en rien de nombreux aspects de la post-production audio. L'une des différences majeures est qu'au lieu d'effectuer le monitoring avec des haut-parleurs haut de gamme, les ingénieurs du son devraient plutôt utiliser un casque ou des AirPods pour une véritable immersion spatiale et pour une meilleure reproduction de l'environnement d'écoute.

Alors que dans les mixages 2D, la majorité des dialogues est mixée sur les canaux centraux, l'audio spatial doit être mixé dans la zone d'où provient le son. Le ducking, l'égalisation et la balance doivent être utilisés pour mixer les stems audio de manière à ne pas submerger ou dérouter l'auditeur.

Les éléments spatiaux immersifs peuvent être combinés avec un stem stéréo standard, permettant à certains sons de rester constants quels que soient les mouvements de la tête. Ceci est utile pour conserver la clarté et la facilité d'interprétation du son non diégétique, comme la musique originale et les voix off.

Les sons d'ambiance ont tendance à être perçus plus faiblement en audio spatial, et devront donc souvent être mixés de manière plus agressive pour être audibles. Cependant, cela ne doit pas se faire au risque de désorienter l'auditeur.

La réverbération contribue grandement à l'immersion, car c'est un phénomène que les auditeurs expérimentent naturellement au quotidien. Chaque son produit dans un environnement donné rebondit sur les surfaces qui entourent l'auditeur, ce qui crée une réverbération ample et enveloppante. La réverbération est donc encouragée dans la conception audio spatiale pour assurer une immersion totale.

Le format Apple Immersive Audio (AIA) est un conteneur pour plusieurs formats :

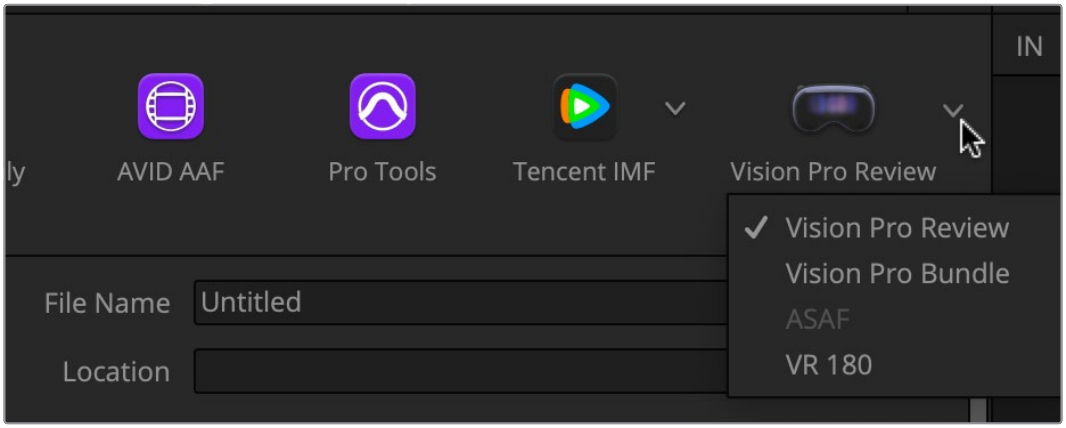
- **L'Apple Spatial Audio Format (ASAF)** est le format audio de travail brut.
- **L'Apple Positional Audio Codec (APAC)** est le format .mp4 encodé, lisible sur les appareils compatibles APAC, comme les AirPods, ou à partir d'un fichier AIVU sur l'Apple Vision Pro.

Apple Spatial Audio Format (ASAF)

L'Apple Spatial Audio Format (ASAF) est un format audio immersif développé principalement pour les applications de VR et d'AR dans l'Apple Vision Pro. Exclusif à macOS, l'ASAF est pris en charge par ProTools (via des plug-ins) et DaVinci Resolve (en natif). L'ASAF dispose de capacités techniques avancées, incluant la prise en charge native des sources ambisoniques (appelées Scenes), et l'acheminement via les assignations de bus. Le mixage est virtuel, sans bus physique, et les utilisateurs peuvent choisir parmi divers formats de monitoring, y compris le monitoring binaural avec suivi des mouvements de la tête via une intégration directe des AirPods d'Apple. La gestion du panoramique s'effectue via les outils existants de DaVinci Resolve comme la boîte de dialogue de panoramique, l'automation, SpaceView, et les outils de panoramique dans le viewer, avec des contrôles secondaires dans l'Inspecteur. Ceux-ci permettent le placement spatial précis des sources, y compris le suivi automatique par rapport à une vidéo lat-long. Les mixages ASAF peuvent être exportés aux formats WAV ou MP4 (accompagnés des métadonnées correspondantes) via l'option Audio Immersif > Exporter fichier master, bien qu'ils ne puissent pas actuellement être importés ou lus dans DaVinci Resolve.

Remise des projets

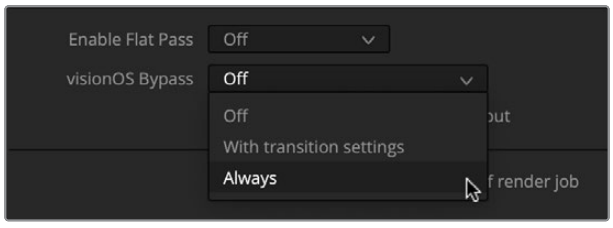
Dans la page Exportation, deux paramètres de rendu vous permettent d'effectuer le rendu de fichiers pour l'Apple Vision Pro, à des fins de révision et d'exportation finale.



- **Vision Pro Review** – crée un petit package MV-HEVC à des fins de prévisualisation. Vous pouvez « partager » le fichier obtenu sur l'Apple Vision Pro via le Wi-Fi et le visionner dans l'application Apple Immersive Video Utility (AIVU), disponible sur l'App Store.
- **Vision Pro Bundle** – crée un volumineux paquet de fichiers ProRes qui peut être archivé ou traité dans Apple Compressor pour le préparer à la lecture sur l'Apple TV.
- **ASAF** – effectue le rendu d'un mixage audio ASAF aux formats WAV ou MP4.
- **VR180** – génère un fichier vidéo stéréoscopique conforme à la norme VR180.

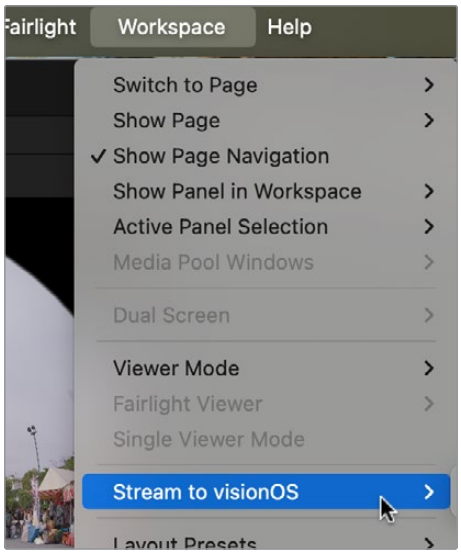
Ignorer les transitions

Si une transition est ajoutée à une timeline immersive, elle sera pré-rendue et transformée en une seule piste de lecture. Le fichier XML généré pour le contenu immersif décrit le processus de commutation entre ces flux rendus (les clips, puis la transition pré-rendue, puis de nouveau les clips). Dans les paramètres avancés du panneau Paramètres d'export, vous avez la possibilité d'ignorer les transitions de visionOS (Always), de les appliquer en fonction de leur statut dans la timeline (With transition settings), ou de les inclure telles qu'elles apparaissent dans la timeline (Off).



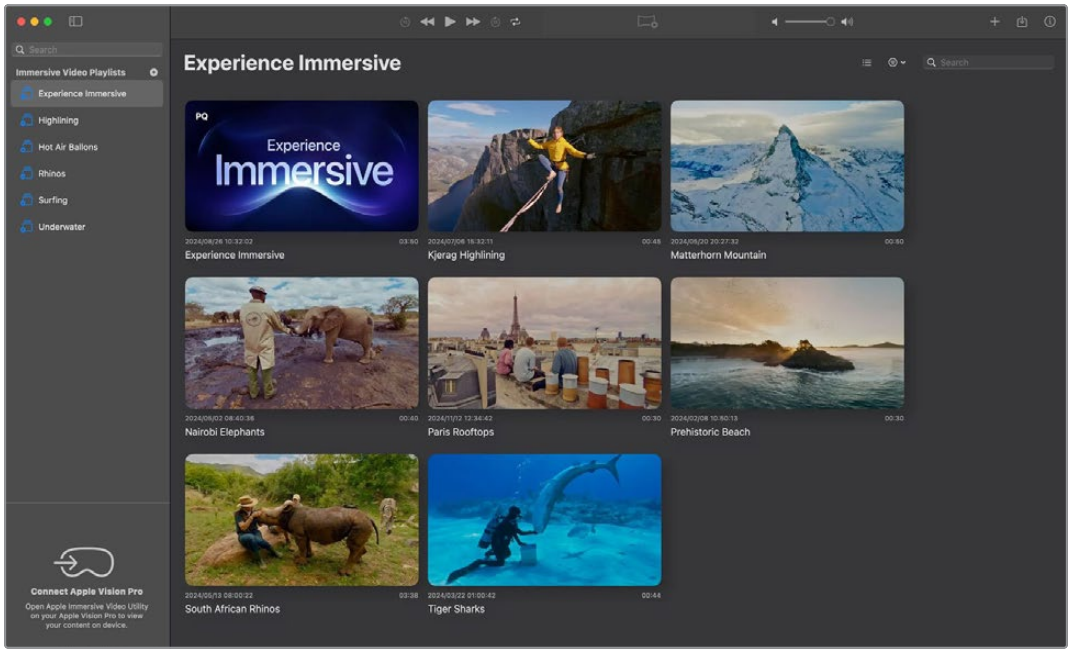
Streaming vers l’Apple Vision Pro

Il est possible d’effectuer un streaming vers l’Apple Vision Pro depuis les pages Montage, Étalonnage et Exportation de DaVinci Resolve en allant dans le menu Espace de travail, puis en choisissant Stream to visionOS et en sélectionnant votre appareil AVP. Un réseau sans fil rapide (Wi-Fi 5 minimum ; Wi-Fi 6 de préférence) est recommandé pour un streaming fluide.



À propos de l’utilitaire Apple Immersive Video

L’utilitaire Apple Immersive Video est une application spécialisée pour macOS et visionOS, conçue pour simplifier la création, la gestion et la lecture de contenu Apple Immersive Video. Destiné aux cinéastes professionnels et aux créateurs de médias immersifs, l’utilitaire permet aux utilisateurs d’importer, d’organiser, d’assembler et de visionner des fichiers vidéo stéréoscopiques 8K avec un champ de vision de 180 degrés et l’audio spatial. Il prend en charge la création de fichiers AIVU (Apple Immersive Video Universal), permettant l’inspection et la modification des métadonnées pour les workflows de post. Les utilisateurs peuvent créer des playlists, trier et effectuer des recherches dans leurs bibliothèques, et même diffuser du contenu en streaming directement vers un ou plusieurs appareils Apple Vision Pro.



Developer Information

Camera Control REST API

If you are a software or hardware developer you can build custom applications or leverage ready to use tools such as REST client or Postman to seamlessly control and interact with your compatible Blackmagic camera using Camera Control REST API. This API enables you to perform a wide range of operations, such as starting or stopping recordings, accessing disk information and much more. Whether you're developing a custom application tailored to your specific needs or utilizing existing tools, this API empowers you to unlock the full potential of your Blackmagic camera with ease. We look forward to seeing what you come up with!

NOTE It's important to mention that controlling Blackmagic cameras via REST API relies on the web manager being enabled on each compatible Blackmagic camera. Enable the web media manager in the Blackmagic Camera Setup 'network access' settings for each camera you are controlling.

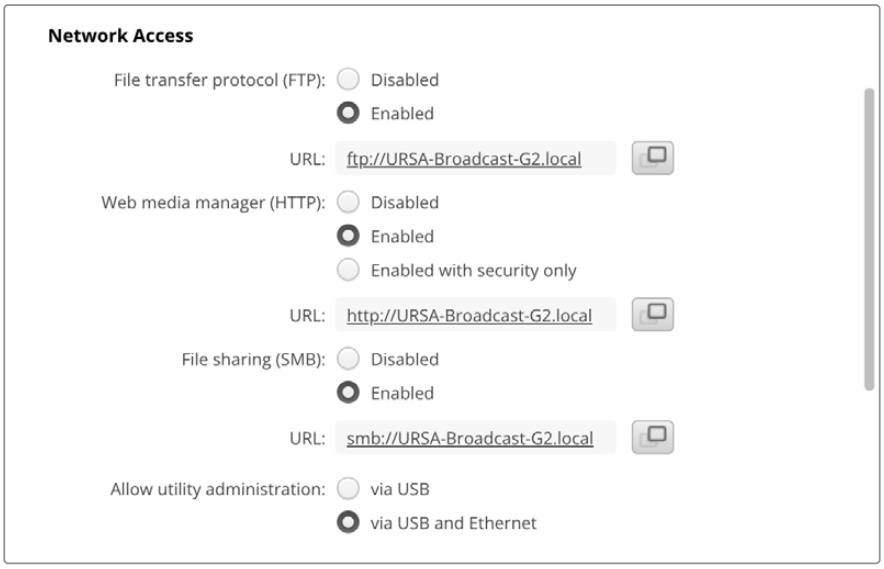
The following Blackmagic cameras are compatible with Camera Control REST API:

- Blackmagic URSA Cine 12K LF
- Blackmagic URSA Cine 17K 65
- Blackmagic URSA Cine Immersive
- Blackmagic PYXIS 6K
- Blackmagic Cinema Camera 6K
- Blackmagic URSA Broadcast G2
- Blackmagic Micro Studio Camera 4K G2
- Blackmagic Studio Camera 4K Plus
- Blackmagic Studio Camera 4K Pro
- Blackmagic Studio Camera 6K Pro
- Blackmagic Studio Camera 4K Plus G2
- Blackmagic Studio Camera 4K Pro G2

Sending API Commands

To send an API command to your camera from a third party application such as Postman, add /control/api/v1/ to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/api/v1/

You can find the Web media manager URL and IP address information in Blackmagic Camera Setup.



The Web media manager URL in Blackmagic Camera Setup

Downloading API's from your Camera

You can download REST API YAML documentation from your camera by adding /control/documentation.html to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/documentation.html

NOTE It's worth noting that changing the camera name in Blackmagic Camera Setup will also change the camera's Web media manager URL.

Livestream Control API

API for controlling Livestreams on Blackmagic Design products.

GET /livestreams/0

Get the livestream's current status.

Response

200 - Livestream's current status.

The response is JSON.

Name	Type	Description
status (required)	string	Possible values are: Idle, Connecting, Streaming, Flushing, Interrupted.
bitrate (required)	integer	Current bitrate (bps).
effectiveVideoFormat (required)	string	Effective video format for the livestream, serialised as a string.
duration	integer	Current stream duration in seconds. Absent if livestream is idle.
cache	integer	Current stream cache usage percentage.

GET /livestreams/0/start

Determine if the livestream is active.

Response

200 - Livestream active status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is active.

PUT /livestreams/0/start

Start the livestream.

Response

204 - Livestream started.

GET /livestreams/0/stop

Determine if the livestream is inactive.

Response

200 - Livestream inactive status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is inactive.

PUT /livestreams/0/stop

Stop the livestream.

Response

204 - Livestream stopped.

GET /livestreams/0/activePlatform

Get the currently selected platform configuration for the livestream.

Response

200 - Livestream active platform configuration.

The response is JSON.

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

PUT /livestreams/0/activePlatform

Set the currently selected platform configuration for the livestream.

Parameters

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

Response

204 - Livestream active platform configuration updated.

400 - Bad Request

GET /livestreams/platforms

Get the list of available platforms.

Response

200 - List of available platforms.

The response is JSON.

Name	Type	Description
	array	List of available platforms names.
[i]	string	Platform name.

GET /livestreams/platforms/platformName

Get the service configuration for a platform.

Parameters

Name	Type	Description
platformName (required)	string	Name of the platform.

Response

200 - Service configuration for specified platform.

The response is JSON.

Livestream platform service configuration.

Name	Type	Description
platform (required)	string	Corresponding platform name.
key	string	Default stream key.
servers (required)	array	List of server configurations.
servers[i]	object	Server configuration.
servers[i].server (required)	string	Server name.
servers[i].url (required)	string	Livestream destination.
servers[i].srtExtensions	array	Miscellaneous tags used for SRT livestreams.
servers[i].srtExtensions[i]	object	Dictionary object mapping SRT tag strings to values.
servers[i].srtExtensions[i][**key**]	string	SRT tag value.
servers[i].group	string	Logical grouping of the server.
profiles (required)	array	List of profile configurations.
profiles[i]	object	Quality configuration.
profiles[i].profile (required)	string	Quality level name.
profiles[i].configs (required)	array	List of video format configurations.
profiles[i].configs[i]	object	Video format configuration for profiles.
profiles[i].configs[i].resolution (required)	string	Video format serialised as a string.
profiles[i].configs[i].fps (required)	string	Frames per second.
profiles[i].configs[i].bitrate (required)	integer	Pixel bitrate (bps).
profiles[i].configs[i].audioBitrate	integer	Audio bitrate (bps).
profiles[i].configs[i].keyFrameInterval	integer	How often a key frame is sent, in seconds.
profiles[i].configs[i].videoCodecs	array	Supported video encoding algorithm/s.

Name	Type	Description
profiles[i].configs[i].videoCodecs[i]	string	Video encoding algorithm. Possible values are: H264, H265.
profiles[i].lowLatency (required)	boolean	If true, fewer frames will be buffered in the livestream.
defaultProfile	string	Quality level name.
credentials	object	Credientials used for RTMP streams.
credentials.username (required)	string	The username part of the creditials. Only used for RTMP streams.
credentials.password (required)	string	Used for RTMP streams, also used as Passphrase for SRT streams.
customizableUrlEnabled	boolean	True when the server URL is customizable.

400 - Bad Request

GET /livestreams/customPlatforms

Get a list of custom platform files.

Response

200 - List of custom platform files.

The response is JSON.

Name	Type	Description
	array	List of custom platform file names.
[i]	string	Custom platform file name.

DELETE /livestreams/customPlatforms

Remove all custom configuration files.

Response

204 - All custom configuration files removed.

GET /livestreams/customPlatforms/filename

Get a custom platform file.

Parameters

Name	Type	Description
filename (required)	string	Name of the file to get.

Response

200 - Custom platform file.

The response is XML.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

404 - Not Found

PUT /livestreams/customPlatforms/filename

Update a custom platform file if it exists, if not, create a new file with the given file name.

Parameters

Name	Type	Description
filename (required)	string	Name of the file to update/create.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

Response

204 - Custom platform file created or updated.

400 - Bad Request

DELETE /livestreams/customPlatforms/filename

Remove the given custom platform file.

Parameters

Name	Type	Description
filename (required)	string	Name of the file to be removed.

Response

204 - Custom platform file removed.

404 - Not Found

Clips Control API

API for listing clips on disk.

GET /clips

Get the list of clips on the active disk.

Response

200 - List of clips on the active disk.

The response is JSON.

List of media clips.

Name	Type	Description
clips (required)	array	
clips[i]	object	Media clip.
clips[i].clipUniqueId (required)	integer	Unique ID used to identify this clip.
clips[i].filePath	string	Path to the file relative to the root of a mount.
clips[i].fileSize	integer	Size of file on disk in bytes.
clips[i].codecFormat	object	
clips[i].codecFormat.codec	string	Currently selected codec.

Name	Type	Description
clips[i].codecFormat.container	string	Multimedia container format.
clips[i].videoFormat	object	Video format configuration.
clips[i].videoFormat.name (required)	string	Video format serialised as a string.
clips[i].videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
clips[i].videoFormat.height	number	Height dimension of video format.
clips[i].videoFormat.width	number	Width dimension of video format.
clips[i].videoFormat.interlaced	boolean	Is the display format interlaced?
clips[i].startTimecode	string	Start timecode of the clip serialised as string.
clips[i].durationTimecode	string	Duration of the clip in timecode format serialised as string.
clips[i].frameCount	integer	Number of frames in clip; duration of the clip in frames.

404 - There is no active disk.

Media Pool Control API

API to manage media pool and handle uploads and project data.

GET /cloud/projects

List all projects within the media pool.

Response

200 - Successfully retrieved the list of all projects.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].libraryID	string	
[i].name	string	
[i].private	boolean	
[i].shared	boolean	
[i].clips	array	List of clips associated with the project.
[i].clips[i]	string	
[i].status	object	
[i].status.numClipsRequested	integer	
[i].status.numClipsComplete	integer	
[i].status.uploadPercent	integer	Percentage of upload completion.
[i].status.numClipsPaused	integer	
[i].status.outOfSpace	boolean	
[i].status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
[i].status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/active

Retrieve data of the actively uploading project.

Response

200 - Successfully retrieved the active project's data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/projectID

Retrieve specific project data by project ID.

Parameters

Name	Type	Description
projectID (required)	integer	Unique identifier of the project.

Response

200 - Successfully retrieved the project data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	

Name	Type	Description
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

404 - Project not found.

GET /cloud/clips

List all clips within the media pool.

Response

200 - Successfully retrieved the list of all clips.

The response is JSON.

Name	Type	Description
	array	
[i]	string	

GET /cloud/clips/activeUploading

Retrieve data of actively uploading clips.

Response

200 - Successfully retrieved the list of actively uploading clips.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].path	string	
[i].projectID	integer	
[i].status	object	
[i].status.projectID	integer	
[i].status.outOfSpace	boolean	
[i].status.proxyExtension	string	
[i].status.growingFile	boolean	
[i].status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.originalClipTotalSize	integer	
[i].status.proxyClipTotalSize	integer	
[i].status.originalClipCompletedSize	integer	
[i].status.proxyClipCompletedSize	integer	
[i].status.secsRemaining	integer	

GET /cloud/clips/deviceName/path

Retrieve specific clip data by device and path.

Parameters

Name	Type	Description
deviceName (required)	string	Name of the device where the clip is stored.
path (required)	string	Path to the clip.

Response

200 - Successfully retrieved the clip data.

The response is JSON.

Name	Type	Description
path	string	
projectID	integer	
status	object	
status.projectID	integer	
status.outOfSpace	boolean	
status.proxyExtension	string	
status.growingFile	boolean	
status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.originalClipTotalSize	integer	
status.proxyClipTotalSize	integer	
status.originalClipCompletedSize	integer	
status.proxyClipCompletedSize	integer	
status.secsRemaining	integer	

404 - Clip not found.

Monitoring Control API

API for monitoring and controlling display settings in video equipment.

GET /monitoring/display

Retrieve a list of all display names.

Response

200 - Returns a list of display names.

The response is JSON.

Name	Type	Description
displays	array	List of display names available.
displays[i]	string	

GET /monitoring/displayName/cleanFeed

Get the clean feed enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/displayName/cleanFeed

Set the clean feed enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Clean feed enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/displayName/displayLUT

Get the display LUT enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

400 - Invalid display name.

404 - Display name not found.

PUT /monitoring/displayName/displayLUT

Set the display LUT enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Display LUT enabled/disabled successfully.

400 - Invalid input.

422 - Unprocessable Entity - Unable to process the contained instructions.

GET /monitoring/displayName/zebra

Get the zebra enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/displayName/zebra

Set the zebra enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Zebra enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/displayName/focusAssist

Get the focus assist enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/displayName/focusAssist

Set the focus assist enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/focusAssist

Get the focus assist settings.

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

404 - Display name not found.

PUT /monitoring/focusAssist

Set the focus assist settings.

Parameters

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/displayName/frameGuide

Get the frame guide enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame guide enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/displayName/frameGuide

Set the frame guide enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame guide state updated successfully.

422 - Unable to update the frame guide state.

GET /monitoring/frameGuideRatio

Get the current frame guide ratio.

Response

200 - Returns the current frame guide ratio.

The response is JSON.

Name	Type	Description
ratio	string	The frame guide ratio.

PUT /monitoring/frameGuideRatio

Set the frame guide ratio.

Parameters

Name	Type	Description
ratio	string	The frame guide ratio.

Response

204 - Frame guide ratio updated successfully.

422 - Unable to update the frame guide ratio.

GET /monitoring/frameGuideRatio/presets

Get the presets for frame guide ratios.

Response

200 - Returns a list of preset frame guide ratios.

The response is JSON.

Name	Type	Description
presets	array	
presets[i]	string	A frame guide ratio.

GET /monitoring/displayName/frameGrids

Get the frame grids enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame grids enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/displayName/frameGrids

Set the frame grids enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame grids state updated successfully.

422 - Unable to update the frame grids state.

GET /monitoring/frameGrids

Get the global frame grids settings.

Response

200 - Returns the current frame grids settings.

The response is JSON.

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

PUT /monitoring/frameGrids

Set the global frame grids settings.

Parameters

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

Response

204 - Frame grids settings updated successfully.

400 - Invalid input, check the number of frame grids or values.

422 - Unable to update the frame grids settings.

GET /monitoring/displayName/safeArea

Get the safe area enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the safe area enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/displayName/safeArea

Set the safe area enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Safe area state updated successfully.

422 - Unable to update the safe area state.

GET /monitoring/safeAreaPercent

Get the current safe area percentage.

Response

200 - Returns the current safe area percentage.

The response is JSON.

Name	Type	Description
percent	integer	Safe area coverage percentage.

PUT /monitoring/safeAreaPercent

Set the safe area percentage.

Parameters

Name	Type	Description
percent	integer	Safe area coverage percentage to set.

Response

204 - Safe area percentage updated successfully.

400 - Invalid percentage value.

422 - Unable to update the safe area percentage.

GET /monitoring/displayName/falseColor

Get the false color enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the false color enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/displayName/falseColor

Set the false color enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - False color state updated successfully.

422 - Unable to update the false color state.

Event Control API

API For working with built-in websocket.

GET /event/list

Get the list of events that can be subscribed to using the websocket API.

Response

200 - Websocket API events list.

The response is JSON.

Name	Type	Description
events	array	List of events that can be subscribed to using the websocket API.
events[i]	string	

System Control API

API for controlling the System Modes on Blackmagic Design products.

GET /system

Get device system information.

Response

200 - System summary.

The response is JSON.

The properties will be populated only with the values that are supported/implemented by the device in use.

Name	Type	Description
codecFormat	object	Codec format configuration.
codecFormat.codec	string	Codec serialised as string.
codecFormat.container	string	Multimedia container format.
videoFormat	object	Video format configuration.
videoFormat.name (required)	string	Video format serialised as a string.
videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
videoFormat.height	number	Height dimension of video format.
videoFormat.width	number	Width dimension of video format.
videoFormat.interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/product

Get device product information.

Response

200 - Device product information.

The response is JSON.

Product information.

Name	Type	Description
deviceName	string	Name of device as displayed in Setup.
productName	string	Device's product name.
softwareVersion	string	Software version running on device.

501 - This functionality is not implemented for the device in use.

GET /system/supportedCodecFormats

Get the list of supported codecs.

Response

200 - List of supported codec formats.

The response is JSON.

Name	Type	Description
codecs	array	
codecs[i]	object	Codec format configuration.
codecs[i].codec	string	Codec serialised as string.
codecs[i].container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

GET /system/codecFormat

Get the currently selected codec.

Response

200 - Current codec format.

The response is JSON.

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

PUT /system/codecFormat

Update the system codec.

Parameters

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

Response

204 - The codec updated successfully.

400 - The specified codec format is unsupported.

501 - This functionality is not implemented for the device in use.

GET /system/videoFormat

Get the currently selected video format.

Response

200 - Current system video format.

The response is JSON.

Video format configuration.

Name	Type	Description
name (required)	string	Video format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height	number	Height dimension of video format.
width	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

PUT /system/videoFormat

Set the system video format.

Parameters

This parameter can be one of the following types:

Name	Type	Description
name (required)	string	Video format serialised as a string.

Name	Type	Description
frameRate (required)	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height (required)	number	Height dimension of video format.
width (required)	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

Response

204 - The video format updated successfully.

400 - Invalid request.

409 - Operation unsupported in the current state.

501 - This functionality is not implemented for the device in use.

GET /system/supportedVideoFormats

Get the list of supported video formats for the current system state.

Response

200 - List of supported video formats.

The response is JSON.

List of supported video formats.

Name	Type	Description
formats	array	List of video formats.
formats[i]	object	Video format configuration.
formats[i].name (required)	string	Video format serialised as a string.
formats[i].frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
formats[i].height	number	Height dimension of video format.
formats[i].width	number	Width dimension of video format.
formats[i].interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/supportedFormats

Get supported formats.

Response

200 - List of supported formats.

The response is JSON.

Name	Type	Description
supportedFormats	array	
supportedFormats[i]	object	
supportedFormats[i].codecs	array	
supportedFormats[i].codecs[i]	string	
supportedFormats[i].frameRates	array	
supportedFormats[i].frameRates[i]	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
supportedFormats[i]. maxOffSpeedFrameRate	number	
supportedFormats[i]. minOffSpeedFrameRate	number	
supportedFormats[i]. recordResolution	object	
supportedFormats[i]. recordResolution.height	number	Height of the resolution.
supportedFormats[i]. recordResolution.width	number	Width of the resolution.
supportedFormats[i]. sensorResolution	object	
supportedFormats[i]. sensorResolution.height	number	Height of the resolution.
supportedFormats[i]. sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

GET /system/format

Get current format.

Response

200 - Current format.

The response is JSON.

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

PUT /system/format

Set the format.

Parameters

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

Response

204 - System format updated.

501 - This functionality is not implemented for the device in use.

Transport Control API

API for controlling Transport on Blackmagic Design products.

GET /transports/0

Get device's basic transport status.

Response

200 - Transport status.

The response is JSON.

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

PUT /transports/0

Set device's basic transport status.

Parameters

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, Output.

Response

204 - Transport mode was set.

400 - Failed to set transport mode.

GET /transports/0/stop

Determine if transport is stopped.

Response

200 - Transport stop response.

The response is JSON.

Name	Type	Description
	boolean	True when transport mode is InputPreview or when in Output mode and speed is 0.

PUT /transports/0/stop

Stop transport. Deprecated, use POST /transports/0/stop instead.

Response

204 - Transport stopped.

POST /transports/0/stop

Stop transport.

Response

204 - Transport stopped.

GET /transports/0/play

Determine if transport is playing.

Response

200 - Transport play response.

The response is JSON.

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

PUT /transports/0/play

Start playing on transport. Deprecated, use POST /transports/0/play instead.

Response

204 - Transport playing.

400 - Failed to set transport to play.

POST /transports/0/play

Start playing on transport.

Response

204 - Transport playing.

400 - Failed to set transport to play.

GET /transports/0/playback

Get playback state.

Response

200 - Transport playback state.

The response is JSON.

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

PUT /transports/0/playback

Set playback state.

Parameters

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

Response

204 - Updated transport playback state.

400 - Failed to set transport playback state.

GET /transports/0/record

Get record state.

Response

200 - Recording state.

The response is JSON.

Name	Type	Description
recording	boolean	If true, transport is in InputRecord mode.

PUT /transports/0/record

Set record state. Deprecated, use POST /transports/0/record instead.

Parameters

Name	Type	Description
recording (required)	boolean	If true, starts a recording, otherwise stops.
clipName	string	Optional, sets the requested clip name to record to, when “recording” attribute is set to true.

Response

204 - Recording state updated.

400 - Failed to update recording state.

POST /transports/0/record

Start recording.

Parameters

Name	Type	Description
clipName	string	Optional, provides a specific name of clip to record to.

Response

204 - Recording started.

400 - Failed to start recording.

GET /transports/0/clipIndex

Get the clip index of the currently playing clip on the timeline.

Response

200 - Clip index response.

The response is JSON.

Name	Type	Description
clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

GET /transports/0/timecode

Get device timecode.

Response

200 - Timecode response.

The response is JSON.

Name	Type	Description
display	string	The display timecode serialised as a string.
timeline	string	The timeline timecode serialised as a string.

GET /transports/0/timecode/source

Get timecode source selected on device.

Response

200 - Timecode source response.

The response is JSON.

Name	Type	Description
timecode	string	Possible values are: Timeline, Clip.

Timeline Control API

API for controlling playback timeline.

GET /timelines/0

Get the playback timeline.

Response

200 - Playback timeline.

The response is JSON.

Name	Type	Description
clips	array	
clips[i]	object	Timeline clip.
clips[i].clipUniqueId (required)	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).
clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

404 - No timeline / disk available.

DELETE /timelines/0

Clear the current playback timeline. Deprecated, prefer to use POST /timelines/0/clear

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

POST /timelines/0

Add a clip to the timeline.

Parameters

This parameter can be one of the following types:

Add multiple media clips to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips[i].clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips[i].clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips[i].frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add multiple media clips to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add a single clip to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips.clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips.clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips.frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add a single clip to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the timeline as specified.

501 - The operation is not supported on this device.

POST /timelines/0/add

Add a clip to the end of the timeline. Deprecated, use POST /timelines/0 to add clips within the timeline.

Parameters

This parameter can be one of the following types:

Add one clip to the end of the timeline.

Name	Type	Description
clips	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add many clips to the end of the timeline.

Name	Type	Description
clips	array	List of clipUniqueIds of clips to add to end of timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the end of the timeline.

501 - The operation is not supported on this device.

POST /timelines/0/clear

Clear the playback timeline.

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

DELETE /timelines/0/clips/timelineClipIndex

Remove the specified clip from the timeline.

Parameters

Name	Type	Description
timelineClipIndex (required)	integer	The (0-based) timeline clip index of the clip to remove from the timeline.

Response

204 - The specified clip was removed from the timeline.

501 - The operation is not supported on this device.

Media Control API

API for controlling media devices in Blackmagic Design products.

GET /media/workingset

Get the list of media devices currently in the working set.

Response

200 - The list of media devices in the working set.

The response is JSON.

Name	Type	Description
size	integer	The fixed size of this device's working set.
workingset	array	The device's working set.
workingset[i]	object | null	Device within the working set. null if no device is present within the given working set slot.
workingset[i].volume	string	Volume name.
workingset[i].deviceName	string	Internal device name of this media device.
workingset[i].remainingRecordTime	integer	Remaining record time using current codec and video format in seconds.
workingset[i].totalSpace	integer	Total space on media device in bytes.
workingset[i].remainingSpace	integer	Remaining space on media device in bytes.
workingset[i].clipCount	integer	Number of clips currently on the device.

GET /media/active

Get the currently active media device.

Response

200 - The current active media device.

The response is JSON.

The active media device, or null if there is no active media.

Name	Type	Description
workingsetIndex	integer	Working set index of the active media device.
deviceName	string	Device name of media device.

204 - No media is currently active.

PUT /media/active

Set the currently active media device.

Parameters

Name	Type	Description
workingsetIndex	integer	Working set index of the media to make active.

Response

204 - The active media device was set successfully.

400 - Setting the currently active media device is not possible in the current state.

GET /media/devices/doformatSupportedFilesystems

Get the list of filesystems available to format a media device.

Response

200 - The list of filesystems permitted for formatting.

The response is JSON.

Name	Type	Description
	array	List of filesystems permitted for formatting media.
[i]	string	Filesystem serialised as string.

GET /media/devices/deviceName

Get information about a requested device.

Parameters

Name	Type	Description
deviceName (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Information about the requested device.

The response is JSON.

Media device state.

Name	Type	Description
state	string	The current state of the media device. Possible values are: None, Scanning, Mounted, Uninitialised, Formatting, RaidComponent.

400 - Invalid device name.

404 - Device not found.

GET /media/devices/deviceName/doformat

Get a format key, used to format the device with a PUT request.

Parameters

Name	Type	Description
deviceName (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Format prepared.

The response is JSON.

Name	Type	Description
deviceName	string	Device name of media device to format.
key	string	The key required to format this device, provide to PUT /media/devices/**deviceName**/doformat to perform format of media device.

400 - Cannot format the device.

404 - Device not found.

PUT /media/devices/deviceName/doformat

Perform a format of the specified media device.

Parameters

Name	Type	Description
deviceName (required)	string	Device name of the media device. Retrieved by "deviceName" member of GET /media/workingset or GET /media/active.

Name	Type	Description
key	string	The key used to format this device, retrieved from prepare format media request GET /media/devices/**deviceName**/doformat. Format key provided cannot be reused after successful format.
filesystem	string	Filesystem to format to. Supported filesystems can be retrieved with GET /media/devices/doFormatSupportedFilesystems.
volume	string	Volume name to set for the disk after format.

Response

204 - Format successful.

400 - Cannot format the device, invalid filesystem or key.

404 - Device not found.

Slate Control API

API to manage digital slate data.

GET /slates/nextClip

Retrieve the digital slate for the next clip.

Response

200 - Returns the slate data for the next clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	

Name	Type	Description
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Slate data is not available.

PUT /slates/nextClip

Update the slate data for the next clip.

Parameters

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

POST /slates/nextClip/resetProjectData

Reset the project data for the next clip's slate.

Response

200 - Project data reset successfully.

POST /slates/clips/deviceName/path/resetProjectData

Reset the project data for the next clip's slate.

Parameters

Name	Type	Description
deviceName (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
path (required)	string	Path to the clip.

Response

200 - Project data reset successfully.

POST /slates/nextClip/resetLensData

Reset the lens data for the next clip's slate.

Response

200 - Lens data reset successfully.

POST /slates/clips/deviceName/path/resetLensData

Reset the lens data for the next clip's slate.

Parameters

Name	Type	Description
deviceName (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
path (required)	string	Path to the clip.

Response

200 - Lens data reset successfully.

GET /slates/clips/deviceName/path

Retrieve slate data for a specific clip.

Parameters

Name	Type	Description
deviceName (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
path (required)	string	Path to the clip.

Response

200 - Returns the slate data for the specified clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

404 - Clip not found.

PUT /slates/clips/deviceName/path

Update the slate data for a specific clip.

Parameters

Name	Type	Description
deviceName (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
path (required)	string	Path to the clip.

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.

Name	Type	Description
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

Preset Control API

API For controlling the presets on Blackmagic Design products

GET /presets

Get the list of the presets on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presets	array	List of the presets on the camera (.cset files)
presets[i]	string	

POST /presets

Send a preset file to the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presetAdded	string	Name of the preset uploaded (without .cset extension)

400 - Bad request - missing Content-Disposition header or filename

GET /presets/active

Get the currently active preset on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

PUT /presets/active

Set the active preset on the camera

Parameters

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

Response

204 - No Content

404 - Preset file not found

PARAMETERS /presets/presetName

GET /presets/presetName

Download the preset file

Response

200 - OK

The response is a binary file.

404 - File does not exist

PUT /presets/presetName

Save current camera state as a preset

Response

204 - No Content

DELETE /presets/presetName

Delete a preset from the camera

Response

204 - No Content

404 - Preset file not found

Audio Control API

API For controlling audio on Blackmagic Design Cameras

GET /audio/channels

Get the total number of audio channels available

Response

200 - Returns the total number of channels

The response is JSON.

Name	Type	Description
channels	integer	Total number of audio channels available

GET /audio/supportedInputs

Get the list of supported audio inputs

Response

200 - List of all supported audio inputs

The response is JSON.

Name	Type	Description
	array	
[i]	string	A supported audio input

GET /audio/channel/channelIndex/input

Get the audio input (source and type) for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently selected input

The response is JSON.

Name	Type	Description
input	string	Audio input source and type

404 - Channel does not exist

PUT /audio/channel/channelIndex/input

Set the audio input for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
input	string	Audio input source and type

Response

204 - No Content

400 - Invalid audio input

404 - Channel does not exist

GET /audio/channel/channelIndex/input/description

Get the description of the current input of the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Description of the current input of the selected channel

The response is JSON.

Name	Type	Description
description	object	
description.gainRange	object	
description.gainRange.Min	number	The minimum gain value in dB
description.gainRange.Max	number	The maximum gain value in dB
description.capabilities	object	
description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter
description.capabilities.Padding	object	
description.capabilities.Padding. available	boolean	Input supports setting of padding
description.capabilities.Padding. forced	boolean	Padding is forced to be set for the input
description.capabilities.Padding. value	number	Value of the padding in dB

404 - Channel does not exist

GET /audio/channel/channelIndex/supportedInputs

Get the list of supported inputs and their availability to switch to for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its supported inputs are being queried. (Channels index from 0)

Response

200 - The list of supported inputs

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

404 - Channel does not exist

GET /audio/channel/channelIndex/level

Get the audio input level for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set level for the selected channel

The response is JSON.

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

404 - Channel does not exist

PUT /audio/channel/channelIndex/level

Set the audio input level for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

Response

204 - No Content

400 - Invalid input or value out of range

404 - Channel does not exist

GET /audio/channel/channelIndex/phantomPower

Get the audio input phantom power status for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set phantom power for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Phantom power enabled state

404 - Channel does not exist

PUT /audio/channel/channelIndex/phantomPower

Set the audio phantom power for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Phantom power enabled state

Response

204 - No Content

400 - Phantom power is not supported for this input

404 - Channel does not exist

GET /audio/channel/channelIndex/padding

Get the audio input padding status for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set padding for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Padding enabled state

404 - Channel does not exist

PUT /audio/channel/channelIndex/padding

Set the audio input padding for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Padding enabled state

Response

204 - No Content

400 - Padding is not supported or is forced for this input

404 - Channel does not exist

GET /audio/channel/channelIndex/lowCutFilter

Get the audio input low cut filter status for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set low cut filter for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Low cut filter enabled state

404 - Channel does not exist

PUT /audio/channel/channelIndex/lowCutFilter

Set the audio input low cut filter for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Low cut filter enabled state

Response

204 - No Content

400 - Low cut filter is not supported for this input

404 - Channel does not exist

GET /audio/channel/channelIndex/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set availability for the selected channel

The response is JSON.

Name	Type	Description
available	boolean	Whether the input is currently available

404 - Channel does not exist

Lens Control API

API For controlling the lens on Blackmagic Design products

GET /lens/iris

Get lens' aperture

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number

PUT /lens/iris

Set lens' aperture

Parameters

Name	Type	Description
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number
adjustmentStep	integer	Signed value for relative aperture adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens iris is not controllable or is controlled by auto exposure

GET /lens/zoom

Get lens' zoom

Response

200 - OK

The response is JSON.

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value

PUT /lens/zoom

Set lens' zoom

Parameters

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value
adjustmentFocalLength	integer	Signed value for relative focal length adjustment
adjustmentNormalised	number	Signed normalized value for relative zoom adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens zoom is not controllable

GET /lens/focus

Get lens' focus

Response

200 - OK

The response is JSON.

Name	Type	Description
normalised	number	Normalised value

PUT /lens/focus

Set lens' focus

Parameters

Name	Type	Description
normalised	number	Normalised value
focusDistance	integer	Focus distance value

Response

204 - No Content

400 - Bad Request if out of range value is provided

PUT /lens/focus/doAutoFocus

Perform auto focus

Parameters

Name	Type	Description
position	object	
position.x	number	Normalized x coordinate for autofocus ROI
position.y	number	Normalized y coordinate for autofocus ROI

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens focus is not controllable

GET /lens/opticalImageStabilization

Get optical image stabilization status

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Whether optical image stabilization is enabled

501 - Not Implemented if optical image stabilization is not supported on this product

PUT /lens/opticalImageStabilization

Enable or disable optical image stabilization

Parameters

Name	Type	Description
enabled	boolean	Enable or disable optical image stabilization

Response

204 - No Content

501 - Not Implemented if optical image stabilization is not supported on this product

GET /lens/iris/description

Get detailed description of lens' iris capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the iris can be controlled
apertureStop	object	
apertureStop.min	number	Minimum aperture stop
apertureStop.max	number	Maximum aperture stop

GET /lens/zoom/description

Get detailed description of lens' zoom capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the zoom can be controlled
focalLength	object	
focalLength.adjustable	boolean	If focal length is adjustable
focalLength.min	integer	Minimum focal length
focalLength.max	integer	Maximum focal length

GET /lens/focus/description

Get detailed description of lens' focus capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the focus can be controlled
focusDistance	object	
focusDistance.adjustable	boolean	If focus distance is adjustable
focusDistance.min	number	Minimum focus distance
focusDistance.max	number	Maximum focus distance

Video Control API

API For controlling the video on Blackmagic Design products

GET /video/iso

Get current ISO

Response

200 - OK

The response is JSON.

Name	Type	Description
iso	integer	Current ISO value

PUT /video/iso

Set current ISO

Parameters

Name	Type	Description
iso	integer	ISO value to set

Response

204 - No Content

403 - ISO cannot be changed in the current state

GET /video/supportedISOs

Get the list of supported ISO settings

Response

200 - List of supported ISO values

The response is JSON.

Name	Type	Description
supportedISOs	array	Array of supported ISO values
supportedISOs[i]	integer	

GET /video/gain

Get current gain value in decibels

Response

200 - OK

The response is JSON.

Name	Type	Description
gain	integer	Current gain value in decibels

PUT /video/gain

Set current gain value

Parameters

Name	Type	Description
gain	integer	Gain value in decibels to set

Response

204 - No Content

403 - Gain cannot be changed in the current state

GET /video/supportedGains

Get the list of supported gain settings in decibels

Response

200 - List of supported gain values in decibels

The response is JSON.

Name	Type	Description
supportedGains	array	Array of supported gain values in decibels
supportedGains[i]	integer	

GET /video/whiteBalance

Get current white balance

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	integer	Current white balance

PUT /video/whiteBalance

Set current white balance

Parameters

Name	Type	Description
whiteBalance	integer	White balance to set

Response

204 - No Content

400 - Invalid white balance temperature

GET /video/whiteBalance/description

Get white balance range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	object	
whiteBalance.min	integer	Minimum color temperature
whiteBalance.max	integer	Maximum color temperature

PUT /video/whiteBalance/doAuto

Set current white balance automatically

Response

204 - No Content

GET /video/whiteBalanceTint

Get white balance tint

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	integer	Current white balance tint

PUT /video/whiteBalanceTint

Set white balance tint

Parameters

Name	Type	Description
whiteBalanceTint	integer	White balance tint to set

Response

204 - No Content

400 - Invalid white balance tint

GET /video/whiteBalanceTint/description

Get white balance tint range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	object	
whiteBalanceTint.min	integer	Minimum white balance tint
whiteBalanceTint.max	integer	Maximum white balance tint

GET /video/ndFilter

Get ND filter stop

Response

200 - OK

The response is JSON.

Name	Type	Description
stop	number	Current filter power (fStop)

501 - Not implemented for this device

PUT /video/ndFilter

Set ND filter stop

Parameters

Name	Type	Description
stop	number	Filter power (fStop) to set

Response

204 - No Content

400 - Invalid ND filter stop

501 - Not implemented for this device

GET /video/supportedNDFilters

Get the list of available ND filter stops

Response

200 - List of available ND filter stops

The response is JSON.

Name	Type	Description
supportedStops	array	Array of available ND filter stops
supportedStops[i]	number	

501 - Not implemented for this device

GET /video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

Response

200 - List of supported display modes

The response is JSON.

Name	Type	Description
supportedDisplayModes	array	Array of supported display modes
supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

GET /video/ndFilter/displayMode

Get ND filter display mode on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

PUT /video/ndFilter/displayMode

Set ND filter display mode on the camera

Parameters

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

Response

204 - No Content

400 - Invalid display mode for ND filter

501 - Not implemented for this device

GET /video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

Response

200 - Indicates if ND filter is selectable

The response is JSON.

Name	Type	Description
selectable	boolean	True if ND filter adjustments are selectable via a slider

501 - Not implemented for this device

GET /video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

PUT /video/shutter

Set current shutter

Parameters

Name	Type	Description
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

Response

204 - No Content

GET /video/shutter/measurement

Get the current shutter measurement mode

Response

200 - OK

The response is JSON.

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

PUT /video/shutter/measurement

Set the shutter measurement mode

Parameters

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

Response

204 - No Content

400 - Invalid shutter measurement

GET /video/supportedShutters

Get supported shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of supported shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of supported shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of flicker-free shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of flicker-free shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/autoExposure

Get current auto exposure mode

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

PUT /video/autoExposure

Set auto exposure

Parameters

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

Response

204 - No Content

400 - Failed to set auto exposure mode

GET /video/detailSharpening

Get the current state of detail sharpening

Response

200 - Current detail sharpening state

The response is JSON.

Name	Type	Description
enabled	boolean	Whether detail sharpening is enabled

501 - Not implemented for this device

PUT /video/detailSharpening

Enable or disable detail sharpening

Parameters

Name	Type	Description
enabled	boolean	Enable or disable detail sharpening

Response

204 - Detail sharpening state updated

501 - Not implemented for this device

GET /video/detailSharpeningLevel

Get the current detail sharpening level

Response

200 - Current detail sharpening level

The response is JSON.

Name	Type	Description
level	string	Current detail sharpening level Possible values are: Low, Medium, High.

501 - Not implemented for this device

PUT /video/detailSharpeningLevel

Set the detail sharpening level

Parameters

Name	Type	Description
level	string	Desired level of detail sharpening Possible values are: Low, Medium, High.

Response

204 - Detail sharpening level updated

400 - Invalid detail sharpening level

501 - Not implemented for this device

Camera Control API

API For controlling the Camera specific features on Blackmagic Design products

GET /camera/colorBars

Get the status of color bars display

Response

200 - Returns the current status of color bars

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the color bars are currently enabled

PUT /camera/colorBars

Set the status of color bars display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the color bars

Response

204 - Color bars status updated successfully

GET /camera/programFeedDisplay

Get the status of program feed display

Response

200 - Returns the current status of program feed display

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the program feed display is currently enabled

PUT /camera/programFeedDisplay

Set the status of program feed display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the program feed display

Response

204 - Program feed display status updated successfully

GET /camera/tallyStatus

Get the tally status of the camera

Response

200 - Returns the current tally status of the camera

The response is JSON.

Name	Type	Description
status	string	Current tally status of the camera Possible values are: None, Preview, Program.

GET /camera/power

Get the power status of the camera

Response

200 - Returns the current power status

The response is JSON.

Name	Type	Description
source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
batteries	array	List of batteries currently connected to the camera
batteries[i]	object	
batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
batteries[i].chargeRemainingPercent	integer	Remaining battery charge percentage
batteries[i].statusFlags	array	List of battery status flags
batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

GET /camera/power/displayMode

Get the power display mode of the camera

Response

200 - Returns the current power display mode

The response is JSON.

Name	Type	Description
mode	string	Current power display mode Possible values are: Percentage, Voltage.

PUT /camera/power/displayMode

Set the power display mode of the camera

Parameters

Name	Type	Description
mode	string	Power display mode to set Possible values are: Percentage, Voltage.

Response

204 - Power display mode updated successfully

400 - Invalid power display mode

GET /camera/timingReferenceLock

Get the timing reference lock status

Response

200 - Returns the timing reference lock status

The response is JSON.

Name	Type	Description
locked	boolean	Indicates if timing reference is locked

Immersive Control API

API for controlling immersive camera settings on Blackmagic Design cameras

GET /immersive/display/displayName/eye

Get the current eye view for a specific display

Parameters

Name	Type	Description
displayName (required)	string	The display name to query (from /monitoring/display endpoint)

Response

200 - OK

The response is JSON.

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

400 - Invalid display name format

404 - Display not found

422 - Failed to get eye view

PUT /immersive/display/displayName/eye

Set the eye view for a specific display

Parameters

Name	Type	Description
displayName (required)	string	The display name to control (from /monitoring/display endpoint)

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

Response

204 - No Content

400 - Invalid input or display name format

404 - Display not found

422 - Failed to set eye view

Color Correction Control API

API For controlling the color correction on Blackmagic Design products based on DaVinci Resolve Color Corrector

GET /colorCorrection/lift

Get color correction lift

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

PUT /colorCorrection/lift

Set color correction lift

Parameters

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gamma

Get color correction gamma

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

PUT /colorCorrection/gamma

Set color correction gamma

Parameters

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gain

Get color correction gain

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

PUT /colorCorrection/gain

Set color correction gain

Parameters

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/offset

Get color correction offset

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

PUT /colorCorrection/offset

Set color correction offset

Parameters

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/contrast

Get color correction contrast

Response

200 - OK

The response is JSON.

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/contrast

Set color correction contrast

Parameters

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/color

Get color correction color properties

Response

200 - OK

The response is JSON.

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/color

Set color correction color properties

Parameters

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/lumaContribution

Get color correction luma contribution

Response

200 - OK

The response is JSON.

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

PUT /colorCorrection/lumaContribution

Set color correction luma contribution

Parameters

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

Response

204 - No Content

Notification websocket - 1.0.0

Service that notifies subscribers of device state changes.

messages

Subscribe (The messages from the server/device)

Websocket Opened Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: websocketOpened.
.type	string	Possible values are: event.

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/**displayName**/cleanFeed, /monitoring/**displayName**/displayLUT, /monitoring/**displayName**/zebra, /monitoring/**displayName**/focusAssist, /monitoring/**displayName**/frameGuide, /monitoring/**displayName**/frameGrids, /monitoring/**displayName**/safeArea, /monitoring/**displayName**/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/**channelIndex**/input, /audio/channel/**channelIndex**/supportedInputs, /audio/channel/**channelIndex**/level, /audio/channel/**channelIndex**/phantomPower, /audio/channel/**channelIndex**/padding, /audio/channel/**channelIndex**/lowCutFilter, /audio/channel/**channelIndex**/available, /audio/channel/**channelIndex**/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*

Name	Type	Description
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/**displayName**/cleanFeed, /monitoring/**displayName**/displayLUT, /monitoring/**displayName**/zebra, /monitoring/**displayName**/focusAssist, /monitoring/**displayName**/frameGuide, /monitoring/**displayName**/frameGrids, /monitoring/**displayName**/safeArea, /monitoring/**displayName**/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/**channelIndex**/input, /audio/channel/**channelIndex**/supportedInputs, /audio/channel/**channelIndex**/level, /audio/channel/**channelIndex**/phantomPower, /audio/channel/**channelIndex**/padding, /audio/channel/**channelIndex**/lowCutFilter, /audio/channel/**channelIndex**/available, /audio/channel/**channelIndex**/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Event Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: propertyValueChanged.

Name	Type	Description
.data.property	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/**displayName**/cleanFeed, /monitoring/**displayName**/displayLUT, /monitoring/**displayName**/zebra, /monitoring/**displayName**/focusAssist, /monitoring/**displayName**/frameGuide, /monitoring/**displayName**/frameGrids, /monitoring/**displayName**/safeArea, /monitoring/**displayName**/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/**channelIndex**/input, /audio/channel/**channelIndex**/supportedInputs, /audio/channel/**channelIndex**/level, /audio/channel/**channelIndex**/phantomPower, /audio/channel/**channelIndex**/padding, /audio/channel/**channelIndex**/lowCutFilter, /audio/channel/**channelIndex**/available, /audio/channel/**channelIndex**/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.value	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.type	string	Possible values are: event.

Publish (The messages that user can send to the server/device)

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	

Name	Type	Description
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/**displayName**/cleanFeed, /monitoring/**displayName**/displayLUT, /monitoring/**displayName**/zebra, /monitoring/**displayName**/focusAssist, /monitoring/**displayName**/frameGuide, /monitoring/**displayName**/frameGrids, /monitoring/**displayName**/safeArea, /monitoring/**displayName**/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/**channelIndex**/input, /audio/channel/**channelIndex**/supportedInputs, /audio/channel/**channelIndex**/level, /audio/channel/**channelIndex**/phantomPower, /audio/channel/**channelIndex**/padding, /audio/channel/**channelIndex**/lowCutFilter, /audio/channel/**channelIndex**/available, /audio/channel/**channelIndex**/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	

Name	Type	Description
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/**displayName**/cleanFeed, /monitoring/**displayName**/displayLUT, /monitoring/**displayName**/zebra, /monitoring/**displayName**/focusAssist, /monitoring/**displayName**/frameGuide, /monitoring/**displayName**/frameGrids, /monitoring/**displayName**/safeArea, /monitoring/**displayName**/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/**channelIndex**/input, /audio/channel/**channelIndex**/supportedInputs, /audio/channel/**channelIndex**/level, /audio/channel/**channelIndex**/phantomPower, /audio/channel/**channelIndex**/padding, /audio/channel/**channelIndex**/lowCutFilter, /audio/channel/**channelIndex**/available, /audio/channel/**channelIndex**/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Device Properties

/media/workingset

The value JSON returned via the eventResponse when the `/media/workingset` property changes on the device:

Name	Type	Description
.size	integer	The fixed size of this device's working set.
.workingset	array	Array of devices within the working set. null if no device is present within the given working set slot.
.workingset[i]		

/media/active

The value JSON returned via the eventResponse when the `/media/active` property changes on the device:

Name	Type	Description
.workingsetIndex	integer	Working set index of the active media device.
.deviceName	string	Internal device name of this media device.

/system

The value JSON returned via the eventResponse when the `/system` property changes on the device:

Name	Type	Description
.codecFormat	object	Codec format configuration.
.codecFormat.codec	string	Codec format serialised as a string.
.codecFormat.container	string	Multimedia container format.
.videoFormat	object	Currently selected video format.
.videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.videoFormat.height	number	Height dimension of video format.
.videoFormat.width	number	Width dimension of video format.
.videoFormat.interlaced	boolean	Is the display format interlaced?.
.videoFormat.name	string	Video format serialised as a string.

/system/codecFormat

Codec format configuration.

The value JSON returned via the eventResponse when the `/system/codecFormat` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.container	string	Multimedia container format.

/system/videoFormat

Currently selected video format.

The value JSON returned via the eventResponse when the `/system/videoFormat` property changes on the device:

Name	Type	Description
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.height	number	Height dimension of video format.
.width	number	Width dimension of video format.
.interlaced	boolean	Is the display format interlaced?.
.name	string	Video format serialised as a string.

/system/format

The value JSON returned via the eventResponse when the `/system/format` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.maxOffSpeedFrameRate	number	
.minOffSpeedFrameRate	number	
.offSpeedEnabled	boolean	
.offSpeedFrameRate	number	
.recordResolution	object	
.recordResolution.height	number	Height of the resolution.
.recordResolution.width	number	Width of the resolution.
.sensorResolution	object	
.sensorResolution.height	number	Height of the resolution.
.sensorResolution.width	number	Width of the resolution.

/system/supportedFormats

The value JSON returned via the eventResponse when the `/system/supportedFormats` property changes on the device:

Name	Type	Description
.supportedFormats	array	
.supportedFormats[i]	object	
.supportedFormats[i].codecs	array	
.supportedFormats[i].codecs[i]	string	
.supportedFormats[i].frameRates	array	
.supportedFormats[i].frameRates[i]	string	Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.supportedFormats[i].maxOffSpeedFrameRate	number	
.supportedFormats[i].minOffSpeedFrameRate	number	
.supportedFormats[i].recordResolution	object	
.supportedFormats[i].recordResolution.height	number	Height of the resolution.
.supportedFormats[i].recordResolution.width	number	Width of the resolution.
.supportedFormats[i].sensorResolution	object	
.supportedFormats[i].sensorResolution.height	number	Height of the resolution.
.supportedFormats[i].sensorResolution.width	number	Width of the resolution.

/timelines/0

The value JSON returned via the eventResponse when the `/timelines/0` property changes on the device:

Name	Type	Description
.clips	array	
.clips[i]	object	Timeline clip.
.clips[i].clipUniqueId	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
.clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
.clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
.clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
.clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).

Name	Type	Description
.clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
.clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

/transports/0

The value JSON returned via the eventResponse when the `/transports/0` property changes on the device:

Name	Type	Description
.mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

/transports/0/stop

true when transport mode is InputPreview or when in Output mode and speed is 0.

The value JSON returned via the eventResponse when the `/transports/0/stop` property changes on the device:

Name	Type	Description
	boolean	true when transport mode is InputPreview or when in Output mode and speed is 0.

/transports/0/play

True when transport is in Output mode and speed is non-zero.

The value JSON returned via the eventResponse when the /transports/0/play property changes on the device:

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

/transports/0/playback

The value JSON returned via the eventResponse when the `/transports/0/playback` property changes on the device:

Name	Type	Description
.type	string	Possible values are: Play, Jog, Shuttle, Var.
.loop	boolean	When true playback loops from the end of the timeline to the beginning of the timeline.
.singleClip	boolean	When true playback loops from the end of the current clip to the beginning of the current clip.
.speed	number	Playback speed, 1.0 for normal forward playback
.position	integer	Playback position on the timeline in units of video frames.

/transports/0/record

The value JSON returned via the eventResponse when the `/transports/0/record` property changes on the device:

Name	Type	Description
.recording	boolean	Is transport in Input Record mode.

/transports/0/timecode

The value JSON returned via the eventResponse when the `/transports/0/timecode` property changes on the device:

Name	Type	Description
.display	string	The display timecode serialised as a string.
.timeline	string	The timeline timecode serialised as a string.

/transports/0/timecode/source

The value JSON returned via the eventResponse when the `/transports/0/timecode/source` property changes on the device:

Name	Type	Description
.timecode	string	Possible values are: Timeline, Clip.

/transports/0/clipIndex

The value JSON returned via the eventResponse when the `/transports/0/clipIndex` property changes on the device:

Name	Type	Description
.clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

/slates/nextClip

The value JSON returned via the eventResponse when the `/slates/nextClip` property changes on the device:

Name	Type	Description
.clip	object	
.clip.clipName	string	
.clip.reel	integer	
.clip.scene	string	
.clip.sceneLocation	string	Possible values are: Interior, Exterior.
.clip.sceneTime	string	Possible values are: Day, Night.
.clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
.clip.take	integer	
.clip.takeType	string	Possible values are: None, PU, VFX, SER.
.clip.goodTake	boolean	
.lens	object	
.lens.lensType	string	
.lens.iris	string	

Name	Type	Description
.lens.focalLength	string	
.lens.distance	string	
.lens.filter	string	
.project	object	
.project.projectName	string	
.project.director	string	
.project.camera	string	
.project.cameraOperator	string	

/monitoring/displayName/cleanFeed

The value JSON returned via the eventResponse when the `/monitoring/`**displayName**`/cleanFeed` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/displayName/displayLUT

The value JSON returned via the eventResponse when the `/monitoring/`**displayName**`/displayLUT` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/displayName/zebra

The value JSON returned via the eventResponse when the `/monitoring/`**displayName**`/zebra` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/displayName/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/`**displayName**`/focusAssist` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/displayName/frameGuide

The value JSON returned via the eventResponse when the `/monitoring/`**displayName**`/frameGuide` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/displayName/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/`**displayName**`/frameGrids` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/displayName/safeArea

The value JSON returned via the eventResponse when the `/monitoring/`**displayName**`/safeArea` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/displayName/falseColor

The value JSON returned via the eventResponse when the `/monitoring/`**displayName**`/falseColor` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/focusAssist` property changes on the device:

Name	Type	Description
.mode	string	Possible values are: Peak, ColoredLines.
.color	string	Possible values are: Red, Green, Blue, White, Black.
.intensity	integer	

/monitoring/frameGuideRatio

The value JSON returned via the eventResponse when the `/monitoring/frameGuideRatio` property changes on the device:

Name	Type	Description
.ratio	string	

/monitoring/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/frameGrids` property changes on the device:

Name	Type	Description
.frameGrids	array	
.frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

/monitoring/safeAreaPercent

The value JSON returned via the eventResponse when the `/monitoring/safeAreaPercent` property changes on the device:

Name	Type	Description
.percent	integer	Safe area coverage percentage.

/audio/channel/channelIndex/input

Get the audio input (source and type) for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/`**channelIndex**`/input` property changes on the device:

Name	Type	Description
.input	string	Audio input source and type

/audio/channel/channelIndex/supportedInputs

The value JSON returned via the eventResponse when the `/audio/channel/`**channelIndex**`/supportedInputs` property changes on the device:

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

/audio/channel/channelIndex/level

Get the audio input level for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/`**channelIndex**`/level` property changes on the device:

Name	Type	Description
.gain	number	Gain value in dB
.normalised	number	Normalised level value between 0.0 and 1.0

/audio/channel/channelIndex/phantomPower

Get the audio input phantom power status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/`**channelIndex**`/phantomPower` property changes on the device:

Name	Type	Description
.enabled	boolean	Phantom power enabled state

/audio/channel/channelIndex/padding

Get the audio input padding status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/`**channelIndex**`/padding` property changes on the device:

Name	Type	Description
.enabled	boolean	Padding enabled state

/audio/channel/channelIndex/lowCutFilter

Get the audio input low cut filter status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/`**channelIndex**`/lowCutFilter` property changes on the device:

Name	Type	Description
.enabled	boolean	Low cut filter enabled state

/audio/channel/channelIndex/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

The value JSON returned via the eventResponse when the `/audio/channel/`**channelIndex**`/available` property changes on the device:

Name	Type	Description
.available	boolean	Whether the input is currently available

/audio/channel/channelIndex/input/description

Description of the current input of the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/`**channelIndex**`/input/description` property changes on the device:

Name	Type	Description
.description	object	
.description.gainRange	object	
.description.gainRange.Min	number	The minimum gain value in dB
.description.gainRange.Max	number	The maximum gain value in dB
.description.capabilities	object	
.description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
.description.capabilities. LowCutFilter	boolean	Input supports setting of low cut filter

Name	Type	Description
.description.capabilities.Padding	object	
.description.capabilities.Padding.available	boolean	Input supports setting of padding
.description.capabilities.Padding.forced	boolean	Padding is forced to be set for the input
.description.capabilities.Padding.value	number	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.

/colorCorrection/lift

Get color correction lift

The value JSON returned via the eventResponse when the `/colorCorrection/lift` property changes on the device:

Name	Type	Description
.red	number	Red lift component
.green	number	Green lift component
.blue	number	Blue lift component
.luma	number	Luma lift component

/colorCorrection/gamma

Get color correction gamma

The value JSON returned via the eventResponse when the `/colorCorrection/gamma` property changes on the device:

Name	Type	Description
.red	number	Red gamma component
.green	number	Green gamma component
.blue	number	Blue gamma component
.luma	number	Luma gamma component

/colorCorrection/gain

Get color correction gain

The value JSON returned via the eventResponse when the `/colorCorrection/gain` property changes on the device:

Name	Type	Description
.red	number	Red gain component
.green	number	Green gain component
.blue	number	Blue gain component
.luma	number	Luma gain component

/colorCorrection/offset

Get color correction offset

The value JSON returned via the eventResponse when the `/colorCorrection/offset` property changes on the device:

Name	Type	Description
.red	number	Red offset component
.green	number	Green offset component
.blue	number	Blue offset component
.luma	number	Luma offset component

/colorCorrection/contrast

Get color correction contrast

The value JSON returned via the eventResponse when the `/colorCorrection/contrast` property changes on the device:

Name	Type	Description
.pivot	number	Contrast pivot point
.adjust	number	Contrast adjustment

/colorCorrection/color

Get color correction color properties

The value JSON returned via the eventResponse when the `/colorCorrection/color` property changes on the device:

Name	Type	Description
.hue	number	Color hue adjustment
.saturation	number	Color saturation adjustment

/colorCorrection/lumaContribution

Get color correction luma contribution

The value JSON returned via the eventResponse when the `/colorCorrection/lumaContribution` property changes on the device:

Name	Type	Description
.lumaContribution	number	Luma contribution value

/lens/iris

Get lens' aperture

The value JSON returned via the eventResponse when the `/lens/iris` property changes on the device:

Name	Type	Description
.continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
.apertureStop	number	Aperture stop value
.normalised	number	Normalised value
.apertureNumber	integer	Aperture number

/lens/iris/description

Get detailed description of lens' iris capabilities

The value JSON returned via the eventResponse when the `/lens/iris/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the iris can be controlled
.apertureStop	object	
.apertureStop.min	number	Minimum aperture stop
.apertureStop.max	number	Maximum aperture stop

/lens/focus

Get lens' focus

The value JSON returned via the eventResponse when the `/lens/focus` property changes on the device:

Name	Type	Description
.normalised	number	Normalised value

/lens/focus/description

Get detailed description of lens' focus capabilities

The value JSON returned via the eventResponse when the `/lens/focus/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the focus can be controlled
.focusDistance	object	
.focusDistance.adjustable	boolean	If focus distance is adjustable
.focusDistance.min	number	Minimum focus distance
.focusDistance.max	number	Maximum focus distance

/lens/zoom

Get lens' zoom

The value JSON returned via the eventResponse when the `/lens/zoom` property changes on the device:

Name	Type	Description
.focalLength	integer	Focal length in mm
.normalised	number	Normalised value

/lens/zoom/description

Get detailed description of lens' zoom capabilities

The value JSON returned via the eventResponse when the `/lens/zoom/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the zoom can be controlled
.focalLength	object	
.focalLength.adjustable	boolean	If focal length is adjustable
.focalLength.min	integer	Minimum focal length
.focalLength.max	integer	Maximum focal length

/presets

Get the list of the presets on the camera

The value JSON returned via the eventResponse when the `/presets` property changes on the device:

Name	Type	Description
.presets	array	List of the presets on the camera (.cset files)
.presets[i]	string	

/presets/active

Get the currently active preset on the camera

The value JSON returned via the eventResponse when the `/presets/active` property changes on the device:

Name	Type	Description
.preset	string	Name of the active preset (with .cset extension, or 'default')

/camera/colorBars

Get the status of color bars display

The value JSON returned via the eventResponse when the `/camera/colorBars` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the color bars are currently enabled

/camera/programFeedDisplay

Get the status of program feed display

The value JSON returned via the eventResponse when the `/camera/programFeedDisplay` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the program feed display is currently enabled

/camera/tallyStatus

Get the tally status of the camera

The value JSON returned via the eventResponse when the `/camera/tallyStatus` property changes on the device:

Name	Type	Description
.status	string	Current tally status of the camera Possible values are: None, Preview, Program.

/camera/power

Get the power status of the camera

The value JSON returned via the eventResponse when the `/camera/power` property changes on the device:

Name	Type	Description
.source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
.milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
.batteries	array	
.batteries[i]	object	
.batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
.batteries[i]. chargeRemainingPercent	integer	Remaining battery charge percentage
.batteries[i].statusFlags	array	List of battery status flags
.batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

/camera/power/displayMode

Get the power display mode of the camera

The value JSON returned via the eventResponse when the `/camera/power/displayMode` property changes on the device:

Name	Type	Description
.mode	string	Current power display mode Possible values are: Percentage, Voltage.

/camera/timingReferenceLock

Get the timing reference lock status

The value JSON returned via the eventResponse when the `/camera/timingReferenceLock` property changes on the device:

Name	Type	Description
.locked	boolean	Indicates if timing reference is locked

/video/iso

Get current ISO

The value JSON returned via the eventResponse when the `/video/iso` property changes on the device:

Name	Type	Description
.iso	integer	Current ISO value

/video/supportedISOs

Get the list of supported ISO settings

The value JSON returned via the eventResponse when the `/video/supportedISOs` property changes on the device:

Name	Type	Description
.supportedISOs	array	Array of supported ISO values
.supportedISOs[i]	integer	

/video/gain

Get current gain value in decibels

The value JSON returned via the eventResponse when the `/video/gain` property changes on the device:

Name	Type	Description
.gain	integer	Current gain value in decibels

/video/supportedGains

Get the list of supported gain settings in decibels

The value JSON returned via the eventResponse when the `/video/supportedGains` property changes on the device:

Name	Type	Description
.supportedGains	array	Array of supported gain values in decibels
.supportedGains[i]	integer	

/video/whiteBalance

Get current white balance

The value JSON returned via the eventResponse when the `/video/whiteBalance` property changes on the device:

Name	Type	Description
.whiteBalance	integer	Current white balance

/video/whiteBalance/description

Get white balance range

The value JSON returned via the eventResponse when the `/video/whiteBalance/description` property changes on the device:

Name	Type	Description
.whiteBalance	object	
.whiteBalance.min	integer	Minimum color temperature
.whiteBalance.max	integer	Maximum color temperature

/video/whiteBalanceTint

Get white balance tint

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint` property changes on the device:

Name	Type	Description
.whiteBalanceTint	integer	Current white balance tint

/video/whiteBalanceTint/description

Get white balance tint range

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint/description` property changes on the device:

Name	Type	Description
.whiteBalanceTint	object	
.whiteBalanceTint.min	integer	Minimum white balance tint
.whiteBalanceTint.max	integer	Maximum white balance tint

/video/ndFilter

Get ND filter stop

The value JSON returned via the eventResponse when the `/video/ndFilter` property changes on the device:

Name	Type	Description
.stop	number	Current filter power (fStop)

/video/supportedNDFilters

Get the list of available ND filter stops

The value JSON returned via the eventResponse when the `/video/supportedNDFilters` property changes on the device:

Name	Type	Description
.supportedStops	array	Array of available ND filter stops
.supportedStops[i]	number	

/video/ndFilter/displayMode

Get ND filter display mode on the camera

The value JSON returned via the eventResponse when the `/video/ndFilter/displayMode` property changes on the device:

Name	Type	Description
.displayMode	string	ND filter display mode Possible values are: Stop, Number, Fraction.

/video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

The value JSON returned via the eventResponse when the `/video/supportedNDFilterDisplayModes` property changes on the device:

Name	Type	Description
.supportedDisplayModes	array	Array of supported display modes
.supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

/video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

The value JSON returned via the eventResponse when the `/video/ndFilterSelectable` property changes on the device:

Name	Type	Description
.selectable	boolean	True if ND filter adjustments are selectable via a slider

/video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

The value JSON returned via the eventResponse when the `/video/shutter` property changes on the device:

Name	Type	Description
.continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
.shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
.shutterAngle	number	Shutter angle

/video/shutter/measurement

Get the current shutter measurement mode

The value JSON returned via the eventResponse when the `/video/shutter/measurement` property changes on the device:

Name	Type	Description
.measurement	string	Current shutter measurement mode Possible values are: ShutterAngle, ShutterSpeed.

/video/supportedShutters

Get supported shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/supportedShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of supported shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array of flicker-free shutter speeds
.shutterSpeeds[i]	integer	

/video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/flickerFreeShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of flicker-free shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array
.shutterSpeeds[i]	integer	

/video/autoExposure

Get current auto exposure mode

The value JSON returned via the eventResponse when the `/video/autoExposure` property changes on the device:

Name	Type	Description
.mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
.type	string	Comma-separated list of device types in the auto exposure stack

/video/detailSharpening

Get the current state of detail sharpening

The value JSON returned via the eventResponse when the `/video/detailSharpening` property changes on the device:

Name	Type	Description
.enabled	boolean	Whether detail sharpening is enabled

/video/detailSharpeningLevel

Get the current detail sharpening level

The value JSON returned via the eventResponse when the `/video/detailSharpeningLevel` property changes on the device:

Name	Type	Description
.level	string	Current detail sharpening level of supported shutter speeds Possible values are: Low, Medium, High.

Assistance

Le moyen le plus rapide d'obtenir de l'aide est d'accéder aux pages d'assistance en ligne de Blackmagic Design et de consulter les dernières informations concernant votre caméra.

Pages d'assistance en ligne de Blackmagic Design

Les dernières versions du manuel, du logiciel et des notes d'assistance peuvent être consultées sur la page d'assistance technique de Blackmagic Design : www.blackmagicdesign.com/fr/support.

Contacter le service d'assistance de Blackmagic Design

Si vous ne parvenez pas à trouver l'aide dont vous avez besoin dans notre matériel de support, veuillez utiliser l'option « Envoyer un email » disponible sur la page d'assistance pour envoyer une demande d'assistance par email. Vous pouvez également cliquer sur le bouton « Trouver un support technique » situé sur la page d'assistance et contacter ainsi le centre de support technique Blackmagic Design le plus proche de chez vous.

Vérifier le logiciel actuel

Pour vérifier quelle version du logiciel Blackmagic Camera Setup utility est installée sur votre ordinateur, ouvrez la fenêtre intitulée About Blackmagic Camera Setup utility.

- Sur Mac, ouvrez le Blackmagic Camera Setup Utility à partir du dossier Applications. Sélectionnez About Blackmagic Camera Setup utility dans le menu d'application pour connaître le numéro de version.
- Sur Windows, ouvrez le logiciel Blackmagic Camera Setup utility dans votre menu de Démarrage ou sur l'écran de Démarrage. Cliquez sur le menu Aide et sélectionnez About Blackmagic Camera Setup utility pour connaître le numéro de version.

Comment obtenir les dernières mises à jour logicielles

Après avoir vérifié la version du logiciel Blackmagic Camera Utility installée sur votre ordinateur, veuillez vous rendre sur la page d'assistance Blackmagic Design à l'adresse suivante www.blackmagicdesign.com/fr/support pour vérifier les dernières mises à jour. Même s'il est généralement conseillé d'exécuter les dernières mises à jour, évitez d'effectuer une mise à jour logicielle au milieu d'un projet important.

Avis règlementaires



Élimination des déchets d'équipements électriques et électroniques au sein de l'Union européenne.

Le symbole imprimé sur ce produit indique qu'il ne doit pas être jeté avec les autres déchets. Cet appareil doit être déposé dans un point de collecte agréé pour être recyclé. Le tri, l'élimination et le recyclage séparés de vos équipements usagés permettent de préserver les ressources naturelles et d'assurer le recyclage de ces équipements dans le respect de l'homme et de l'environnement. Pour plus d'informations quant aux lieux d'éliminations des déchets d'équipements électriques, vous pouvez contacter votre mairie ou le service local de traitement des ordures ménagères.



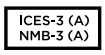
Cet équipement a été testé et déclaré conforme aux limites imposées aux appareils numériques de classe A, en vertu du chapitre 15 des règles de la FCC. Ces limites ont pour objectif d'assurer une protection suffisante contre les interférences nuisibles lorsque l'équipement est utilisé dans un environnement commercial. Cet appareil génère, utilise et émet de l'énergie de fréquences radio et peut, en cas d'installation ou d'utilisation non conforme aux instructions, engendrer des interférences nuisibles au niveau des communications radio. L'utilisation de cet équipement en zone résidentielle est susceptible de provoquer des interférences nuisibles, auquel cas il sera demandé à l'utilisateur de corriger ces interférences à ses frais.

L'utilisation de cet appareil est soumise aux deux conditions suivantes :

1. Cet appareil ne doit pas causer d'interférences nuisibles.
2. Cet appareil doit accepter toute interférence reçue, y compris les interférences pouvant entraîner un dysfonctionnement.



R-R-BMD-20250213001



Déclaration de ISDE Canada

Cet appareil est conforme aux normes canadiennes relatives aux appareils numériques de Classe A.

Toute modification ou utilisation de ce produit en dehors de son utilisation prévue peut annuler la conformité avec ces normes.

Les connexions aux interfaces HDMI doivent être effectuées avec des câbles HDMI blindés d'excellente qualité.

Cet équipement a été testé pour être en conformité avec une utilisation prévue dans un environnement commercial. Si cet équipement est utilisé dans un environnement domestique, il peut provoquer des interférences radio.

Informations de sécurité

L'adaptateur AC vers DC 24V, fourni avec la Blackmagic URSA Cine, doit être connecté à une prise secteur équipée d'un conducteur de protection.

Les pièces de la Blackmagic URSA Cine ne sont pas réparables par l'opérateur. Toute opération d'entretien doit être effectuée par un centre de service Blackmagic Design.

Lorsque vous connectez le connecteur d'alimentation de l'entrée à une source de batterie externe DC, le câblage d'alimentation doit inclure une limitation de courant ou un fusible.

Le câblage utilisé doit être marqué de VW-1 ou être conforme aux pièces concernées du IEC 60332 ou IEC 60695".

Nous recommandons que la source DC de cette caméra soit de 24 V. Si une source DC 12 V est utilisée, le câblage utilisé doit être suffisant pour un courant de 14 A.

Un support pour batterie B-Mount est fourni avec ce produit. Nous recommandons l'utilisation d'une batterie 24 V (28,8 V) ou d'une batterie bitension.

Pour une batterie 12 V (14,4 V), reportez-vous au manuel de votre batterie ou aux marquages pour déterminer le courant de décharge continu maximum. Nous recommandons une caractéristique nominale minimale de 12 A.

Déclaration de l'État de Californie

Ce produit est susceptible de vous exposer à des produits chimiques, dont des traces de polybromobiphényle dans les parties en plastique, reconnu par l'État de Californie comme étant responsable de cancers, d'anomalies congénitales ou d'autres effets nocifs sur la reproduction.

Pour de plus amples informations, veuillez vous rendre sur www.P65Warnings.ca.gov.

Bureau européen

Blackmagic Design Europe B.V.
Rijnlanderweg 766, Unit D2132
NM Hoofddorp
NL

Blackmagic URSA Cine

En cas d'ensoleillement, prenez soin de garder votre caméra à l'ombre. Évitez toute exposition de la caméra ou de la batterie en lithium à un ensoleillement prolongé. Gardez les batteries en lithium à l'abri des sources de chaleur.

Le connecteur de sortie 24V « RS » est adapté pour fournir une alimentation aux accessoires tiers. Veuillez vous assurer que la consommation d'énergie est inférieure à 48 W.

Garantie

Garantie limitée

Par la présente, Blackmagic Design garantit que ce produit sera exempt de défauts matériels et de fabrication pendant une durée d'un an à compter de la date d'achat. Si un produit s'avère défectueux pendant la période de garantie, Blackmagic Design peut, à sa seule discrétion, réparer le produit défectueux sans frais pour les pièces et la main-d'œuvre, ou le remplacer.

Pour se prévaloir du service offert en vertu de la présente garantie, il vous incombe d'informer Blackmagic Design de l'existence du défaut avant expiration de la période de garantie, et de prendre les mesures nécessaires pour l'exécution des dispositions de ce service. Le consommateur a la responsabilité de s'occuper de l'emballage et de l'expédition du produit défectueux au centre de service nommément désigné par Blackmagic Design, en frais de port prépayé. Il incombe au Consommateur de payer tous les frais de transport, d'assurance, droits de douane et taxes et toutes autres charges relatives aux produits qui nous auront été retournés et ce, quelle que soit la raison.

La présente garantie ne saurait en aucun cas s'appliquer à des défauts, pannes ou dommages causés par une utilisation inappropriée ou un entretien inadéquat ou incorrect. Blackmagic Design n'est en aucun cas obligé en vertu de la présente garantie : a) de réparer les dommages résultant de tentatives de réparations, d'installations ou tous services effectués par du personnel non qualifié par Blackmagic Design, b) de réparer tout dommage résultant d'une utilisation inadéquate ou d'une connexion à du matériel incompatible, c) de réparer tout dommage ou dysfonctionnement causé par l'utilisation de pièces ou de fournitures n'appartenant pas à la marque de Blackmagic Design, d) d'examiner un produit qui a été modifié ou intégré à d'autres produits quand l'impact d'une telle modification ou intégration augmente les délais ou la difficulté d'examiner ce produit.

Exposer le viseur URSA Viewfinder directement à la lumière du soleil peut causer des dommages à l'écran du viseur, car les lentilles agissent comme une loupe. La rémanence ou les brûlures peuvent se produire sur les écrans OLED lorsque des images statiques ou des images fortement contrastées, tels que les repères de cadrage, sont affichées à l'écran pendant une période prolongée. Afin d'éviter tout désagrément, veillez à ce que le capteur infrarouge ne soit pas délibérément couvert et pensez à déconnecter le viseur lorsque celui-ci n'est pas utilisé pendant une période prolongée. La rémanence n'est pas couverte par la garantie du produit.

CETTE GARANTIE REMPLACE TOUTE GARANTIE EXPLICITE OU TACITE. BLACKMAGIC DESIGN ET SES REVENDEURS DÉCLINENT EXPRESSÉMENT TOUTE GARANTIE IMPLICITE DE COMMERCIALISATION OU D'ADAPTATION, QUEL QU'EN SOIT LE BUT. LA RESPONSABILITÉ DE BLACKMAGIC DESIGN DE RÉPARER OU REMPLACER UN PRODUIT S'AVÉRANT DÉFECTUEUX PENDANT LA PÉRIODE DE LA GARANTIE CONSTITUE LA TOTALITÉ ET LE SEUL RECOURS EXCLUSIF PRÉVU ET FOURNI AU CONSOMMATEUR. BLACKMAGIC DESIGN N'EST PAS RESPONSABLE DES DOMMAGES INDIRECTS, SPÉCIFIQUES, ACCIDENTELS OU CONSÉCUTIFS, INDÉPENDAMMENT DU FAIT QUE BLACKMAGIC DESIGN OU LE REVENDEUR AIENT ÉTÉ AVISÉS AU PRÉALABLE DE L'ÉVENTUALITÉ DE CES DOMMAGES. BLACKMAGIC DESIGN NE PEUT ÊTRE TENU POUR RESPONSABLE DE TOUTE UTILISATION ILLICITE DU MATÉRIEL PAR LE CONSOMMATEUR. BLACKMAGIC DESIGN N'EST PAS RESPONSABLE DES DOMMAGES RÉSULTANT DE L'UTILISATION DE CE PRODUIT. LE CONSOMMATEUR MANIPULE CE PRODUIT À SES SEULS RISQUES.

© Copyright 2025 Blackmagic Design. Tous droits réservés. 'Blackmagic Design', 'URSA', 'DeckLink', 'HDLink', 'Workgroup Videohub', 'Multibridge Pro', 'Multibridge Extreme', 'Intensity' et 'Leading the creative video revolution' sont des marques déposées aux États-Unis et dans les autres pays. Tous les autres noms de société et de produits peuvent être des marques déposées des sociétés respectives auxquelles ils sont associés.

Le nom de la marque Bluetooth® ainsi que ses logos sont des marques déposées et sont la propriété de Bluetooth SIG, Inc. L'utilisation de ces marques par Blackmagic Design est sous licence. Les autres marques déposées et noms de marques appartiennent à leurs propriétaires respectifs.

Oktober 2025

Leitfaden zur Kamera

Blackmagicdesign

Blackmagic URSA Cine Immersive



Blackmagic URSA Cine Immersive

Inhalt

Blackmagic URSA Cine Immersive 461
Was bedeutet „Immersive“? 461
Apple Immersive Video 461
URSA Cine Immersive 462
Erste Schritte 463
Stromversorgung der Kamera 463
Vorgeben der Aufnahme-Einstellungen 464
Monitoring Ihrer Bilder 466
Monitoring-Optionen und Overlays 466
Kadrieren Ihrer ersten Einstellung 467
Höhe der Kamera 467
Vertikalschwenks der Kamera 468
Positionierung 469
Handlungsspielraum des Publikums 470
Kamerabewegung 470
Datenraten-Tabellen 471
Audio und Timecode 472
Apple Spatial Audio Format (ASAF) 472
Ambisonic 472
Lavalier- und Minimikrofone 473
Handmikrofone 473
Anschließen einer Timecode-Quelle 473
Filmen 475
Voraussetzungen für die Belichtung 475
Verschlusszeit 475
ISO und Dynamikumfang 476
ND 477
Offspeed-Aufzeichnung 478
Histogramm 479
LCD-Monitoring-Optionen 479
Aufnehmen 480
Wiedergeben 481
Transferieren von Clips 482
Zusammenfassung 483
DaVinci Resolve Studio 484
Workstation-Empfehlungen 484
Projekteinrichtung 485
Haupteinstellungen 485
Video-Monitoring 486
Bildskalierung 486
Farbmanagement 487
Infos DaVinci Wide Gamut 488
Verarbeiten von Apple Immersive 489
Ingesting 489
Schnitt 491
Zusammenschnitt 491
Backdrops 491
Verwenden des Viewers 493
Übergänge 495
Apple Immersive Bewegungsdaten 496
Titel 497
Gestalterische Erwägungen 498
VFX-Compositing 499
MediaOut-Nodes 499
Renderer3D 500
Compositing mit dem Immersive Patcher 501
Colorgrading 502
Rauschreduzierung 502
Primäre Farbkorrektur 503
Sekundäre Farbkorrektur 503
Skalierung 504
3D-Palette 504
CST-Farbmanagement 506
CST-Timeline-Mapping 506
CST-Grading-Monitor-Mapping 507
CST-Monitoring mit der Apple Vision Pro 509
Audio 510
Apple Spatial Audio Format (ASAF) 510
Auslieferung 511
Umgehen von Übergängen 511
Streamen auf die Apple Vision Pro 512
Infos zu Apple Immersive Video Utility 512
Informationen für Entwickler (englisch) 513
Camera Control REST API 513
Livestream Control API 515

Clips Control API 519
Media Pool Control API 520
Monitoring Control API 523
Event Control API 532
System Control API 532
Transport Control API 538
Timeline Control API 542
Media Control API 545
Slate Control API 548
Preset Control API 554
Audio Control API 556
Lens Control API 561
Video Control API 565
Camera Control API 573
Immersive Control API 576
Color Correction Control API 577
Notification websocket - 1.0.0 582
messages 582
Hilfe 606
Gesetzliche Vorschriften 607
Sicherheitshinweise 608
Garantie 609

Blackmagic URSA Cine Immersive

Wir gratulieren Ihnen, dass Sie sich eine Blackmagic URSA Cine Immersive angeschafft oder gemietet haben.

Entwickelt wurde die Blackmagic URSA Cine Immersive für die Erfassung von 180-Grad-Stereo-VR-Filmmaterial, gemeinhin als immersives Video bezeichnet. Insbesondere wurde diese Kamera für die Erfassung von Apple Immersive Video für die Apple Vision Pro optimiert, die das hochwertigste, derzeit verfügbare Immersive-Erlebnis liefert.

Dieser Leitfaden stellt die Blackmagic URSA Cine Immersive Kamera vor und führt in ihre Verwendung ein. Er befasst sich vorrangig mit der grundsätzlichen Bedienung der Kamera und dem einzigartigen Format, das für die Bereitstellung von Apple Immersive Video entwickelt wurde. Ein Verständnis des Formats wird Ihnen helfen, auch die Kamera selbst besser zu verstehen.

Die folgenden Seiten geben Ihnen technische und kreative Tipps, die Ihnen mit dem Gebrauch der Kamera helfen und Sie in die Vorteile des immersiven Formats einweihen.

Was bedeutet „Immersive“?

„Immersive“ ist ein dreidimensionales Videoformat, das auch als spatiales oder räumliches Video bezeichnet wird. Es ist Ihnen wahrscheinlich bekannt, dass es bereits seit geraumer Zeit andere, ähnliche Technologien gibt. Für ein besseres Verständnis der Unterschiede zu „Immersive“, folgt hier ein grober Überblick über andere Technologien:

- **Traditionelles 3D-Kino** ermöglicht es dem Publikum, die Tiefe im Einzelbild zu erleben, indem jede Szene mit zwei Kameras aufgenommen wird, deren Abstand zueinander dem Abstand zwischen den menschlichen Augen entspricht. Stereoskopische Bilder ermöglichen dem Publikum zwar das Erleben von Tiefe, seine perspektivische Wahrnehmung wird aber durch die Komposition des Kinematografen bzw. der Kinematografin gesteuert.
- **Sphärisches Video** kombiniert die Prinzipien von 3D- und 360-Grad-Kameras, sodass das Publikum in jede Richtung blicken und gleichzeitig die Tiefe erleben kann. 360-Grad-Bilder erfordern sehr hohe Auflösungen, damit der kleine Teil des Bilds, den das Publikum betrachtet, eine ausreichende Wiedergabetreue aufweist. Da die menschliche Physiologie es uns nur erlaubt, den Kopf von einer Schulter zur anderen zu drehen, um hinter uns schauen zu können, stehen oder sitzen Zuschauende in der Regel auf einem Drehstuhl.
- **Volumetrisches Video** ermöglicht es den Zuschauenden nicht nur, Tiefe zu erleben und sich umzusehen, sondern auch, sich zu bewegen und ihre Perspektive zu ändern. Kamera-Arrays können zwar volumetrische Videos aufnehmen, die meisten Virtual-Reality- oder Augmented-Reality-Anwendungen nutzen aber dynamisch gerenderte 3D-Assets und erfordern eine sichere Umgebung für Teilnehmende.

Immersive vereint das Beste aus traditionellem Kino und interaktiven 3D-Formaten. In seiner einfachsten Form ist es ein hochaufgelöstes stereoskopisches 180-Grad-Format mit hoher Framerate.

Apple Immersive Video

Die Blackmagic URSA Cine Immersive ist ausdrücklich für die Erfassung von Apple Immersive Video ausgelegt. Dieses Format wird mit der Apple Vision Pro verwendet und unterstützt hochauflösende, stereoskopische 3D-Videos mit einem Sichtfeld von 180 Grad.

Apple Immersive Video hat folgende Merkmale:

- Wiedergabe mit 90 Frames pro Sekunde.
- Kann in Auflösungen mit 4320 x 4320 Pixeln pro Auge verteilt werden.
- Ein einzigartiges Metadatensystem dient zur Zuordnung des ursprünglichen Kamerabilds in die 180-Grad-Umgebung.
- Das Apple Spatial Audio Format stellt den 3D-Sound bereit.

Apple Immersive Video erfordert für die Wiedergabe von hochaufgelöstem, stereoskopischem 180-Grad-Video geeignete Hardware. Aktuell unterstützt allein die Apple Vision Pro dieses Format.



Apple Vision Pro

Das Headset verfügt über zwei Micro-OLED-Displays mit einer Auflösung von 3660 x 3200 Pixeln und einem Pixelabstand von 7,5 Mikron. Es verfügt über 13 Tracking-Kameras, einen LiDAR-Scanner, Inertialsensoren und Eye-Tracking-Sensoren, um Bildinhalte auf Kopf- und Augenbewegungen auszurichten.

Ein M2-Prozessor mit einer 8-Kern-CPU und einer 10-Kern-GPU ermöglicht die Wiedergabe mit bis zu 100 Frames pro Sekunde.

URSA Cine Immersive

Die Blackmagic URSA Cine Immersive ist eine zur Erfassung von Apple Immersive Video für die Apple Vision Pro konzipierte Kamera.



Blackmagic URSA Cine Immersive

Die URSA Cine Immersive verwendet zwei aufeinander abgestimmte 58-Megapixel-RGBW-Sensoren. Die Sensoren sind nahezu quadratisch, um das volle 180-Grad-Sichtfeld sowohl in der Horizontalen als auch in der Vertikalen zu erfassen. Jeder Sensor ist mit einem 210-Grad-Fischauge gepaart. Die Objektivmitten sind 64 mm voneinander entfernt platziert.

Integrierte Neutraldichtefilter ermöglichen eine Lichtregulierung von bis zu acht Blendenstufen. Die Blendenöffnung ist auf eine Blendenstufe von circa f/4,5 fixiert.

Die Kamera zeichnet zwei 8K-Bilddatenströme mit 90 Frames pro Sekunde auf. Dabei entstehen deutlich höhere Datendurchsatzraten als beim Erfassen von regulärem 4K-Material bei 24p.

Erste Schritte

Die ersten Schritte mit Ihrer Blackmagic URSA Cine Immersive sind einfach: Versorgen Sie Ihre Kamera mit Strom und schalten Sie sie ein. Das in den Modulschacht der Kamera geladene Blackmagic Media Module 8TB ist bereits formatiert und bereit für die Aufzeichnung Ihrer Clips.

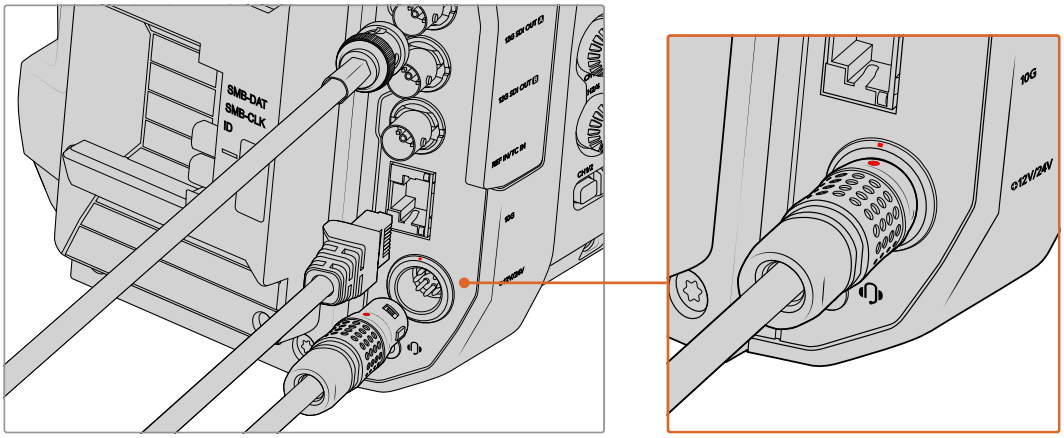
Stromversorgung der Kamera

Am schnellsten versorgen Sie Ihre Kamera mit Strom, indem Sie sie über das mitgelieferte Netzteil an eine externe Stromquelle anschließen.

So schließen Sie die Kamera an eine externe Stromquelle an:

1. Entfernen Sie zunächst die Gummistaubschutzkappe und verwahren Sie sie an einem sicheren Ort. Schließen Sie das 24V-Netzteil mit AC/DC-Wandler über ein standardmäßiges IEC-C13-Netzstromkabel an eine Steckdose an.
2. Stecken Sie den 8-Stift-Stecker des Netzteils in die 24V-DC-Strombuchse am Rücken Ihrer Kamera.

Wenn sowohl ein externes Netzteil als auch ein Akku angeschlossen sind, bezieht die Kamera ihre Energie aus der externen Stromquelle. Sobald die externe Stromquelle entfernt wird, aber ein geladener Akku angeschlossen ist, schaltet Ihre Kamera unterbrechungsfrei auf Akkustrom um.



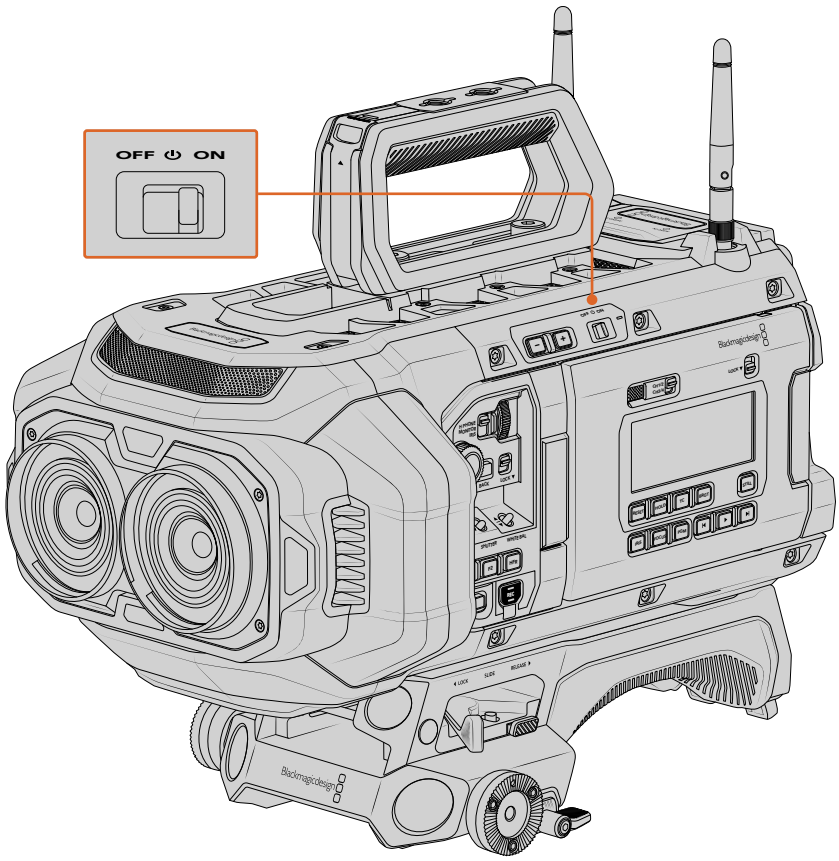
Zum Verbinden der Kamera mit einer Stromquelle, richten Sie den roten Punkt des Steckers auf den roten Punkt an der Kamerabuchse aus. Führen Sie dann den Stecker behutsam ein, bis er einrastet. Um den Stecker zu entfernen, halten Sie seine Metallummantelung fest und ziehen Sie ihn behutsam von der Kamera ab. Dies schiebt die Ummantelung zurück, was den Stecker aus der Buchse löst

Die mit Ihrer Kamera gelieferte Akkuplatte ist eine für B-Mount-Akkus mit 24 Volt ausgelegte B-Mount-Trägerplatte. Der ideale Spannungsbereich für den Betrieb der URSA Cine Immersive liegt zwischen 24 und 34 Volt. Höhere Spannungen reduzieren die Hitzeentwicklung, was die Aufzeichnung mit hohen Bildwechselraten ermöglicht, und erlauben die Stromversorgung mehrerer Zubehörteile Ihrer Kamera.

Des Weiteren sind optionale VLock- (für den 24V-Betrieb) und Gold-Mount-Platten (für den 26V-Betrieb) von Fremdherstellern erhältlich.

HINWEIS Bei Betrieb der URSA Cine Immersive mit 12–18V-Netzteilen oder -Akkus ist die Stromversorgung von Zubehör über den 3-poligen RS-Anschluss oder den 2-poligen Lemo-Anschluss nicht möglich.

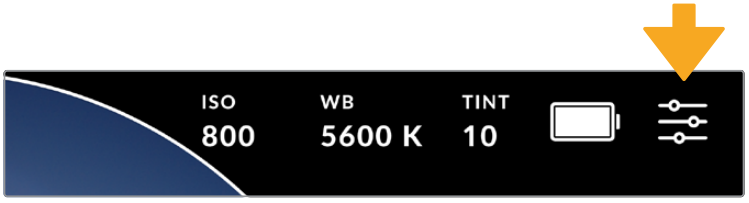
Nach Aufstecken eines Akkus oder Anschließen einer externen Stromquelle, schalten Sie Ihre URSA Cine Immersive ein. Stellen Sie den Ein-/Ausschalter oben an der Kamera auf ON.



Stellen Sie den Schalter auf ON, um Ihre Kamera einzuschalten

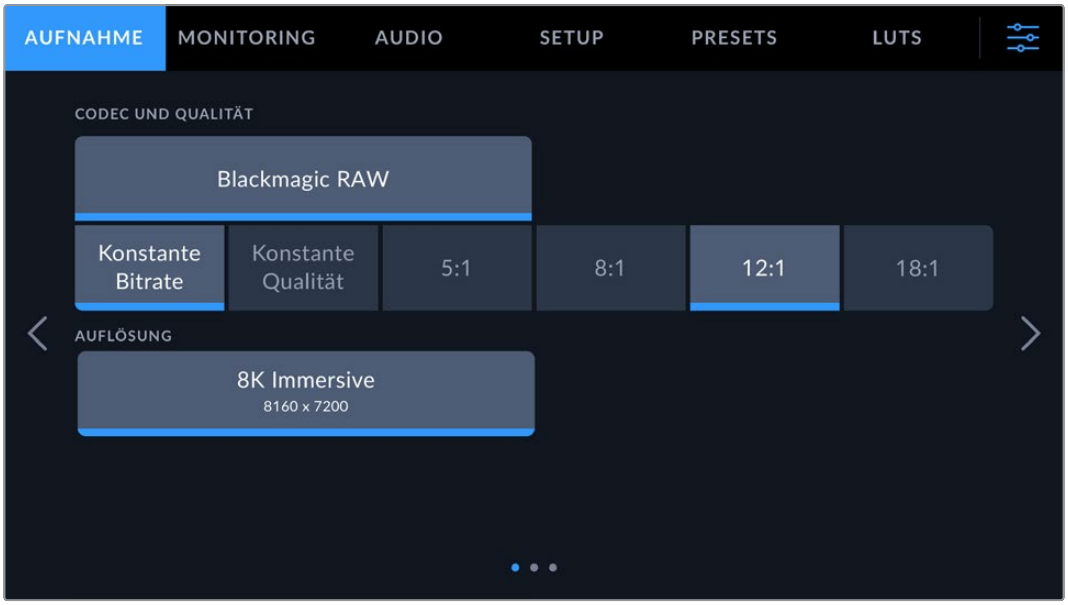
Vorgeben der Aufnahme-Einstellungen

Dieser Abschnitt informiert über spezifische Kameraeinstellungen, die zum Erfassen von immersiven Inhalten dienen oder wichtig sind. Öffnen Sie das Menü mit den AUFNAHME-Einstellungen Ihrer Kamera durch Antippen des Menü-Icons oben rechts auf dem Ausklapp- oder Assist-Touchscreen.



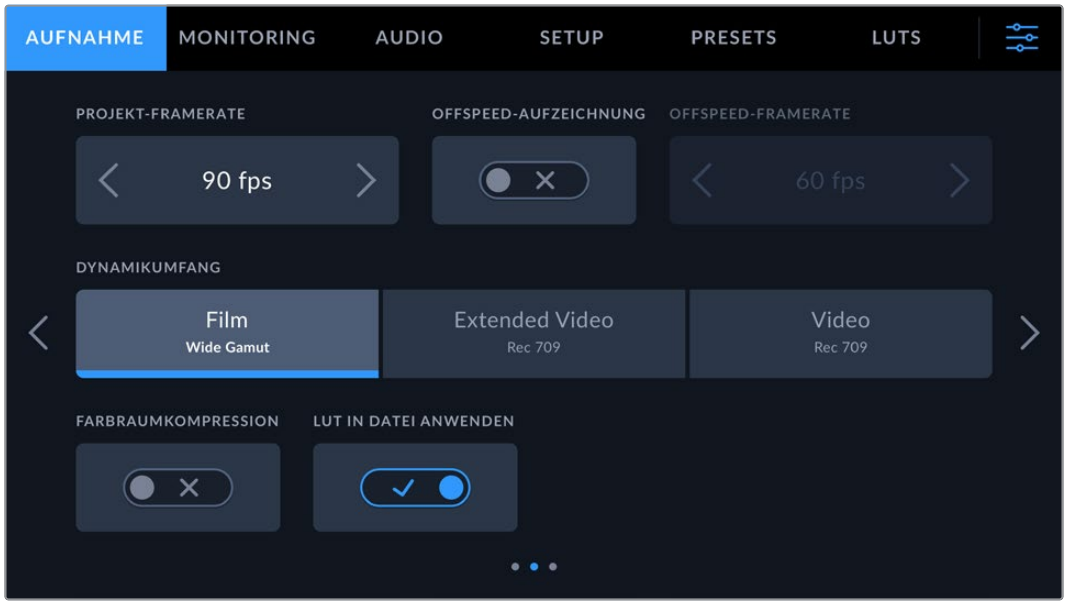
AUFNAHME-Einstellungen 1

Über die Optionen CODEC UND QUALITÄT auf dieser Seite geben Sie die für Ihr aktuelles Projekt am besten geeigneten Codec-Einstellungen für Blackmagic RAW vor. Die Optionen sind je nach Kompression in vier Einstellungen für „Konstante Bitrate“ und zwei für „Konstante Qualität“ aufgeteilt. Zum Aufnehmen von Inhalten für die Apple Vision Pro mit 90 fps raten wir dringend zur Verwendung der 12:1- oder Q3-Kompression. Diese Einstellungen bieten den optimalen Kompromiss zwischen Datenraten und Dateigrößen, die mit aktueller Computerhardware handhabbar sind, und einer weiterhin extrem hohen Bildqualität.



AUFNAHME-Einstellungen 2

Die zweite Seite der AUFNAHME-Einstellungen Ihrer Kamera enthält die Frameraten-Einstellungen.



Setzen Sie die PROJEKT-FRAMERATE Ihrer Kamera auf 90 fps, was der zur Wiedergabe auf der Apple Vision Pro erforderlichen Framerate entspricht.

Wenn Sie für andere Systeme filmen, mag eine geringere Framerate wie 60 fps oder 50 fps ausreichen. Bei herkömmlichen Filmstilen verwendete niedrigere Frameraten, wie 24 fps oder 25 fps, eignen sich ggf. zum Filmen von Visual-Effects-Plates für Kinofilm-Inhalte mit 24p.

Das war's schon, um mit Ihrer URSA Cine Immersive loszulegen! Da das Media Modul in der Kamera bereits formatiert ist, können Sie sofort anfangen zu filmen.

Monitoring Ihrer Bilder

Zur Prüfung Ihrer Bilder beim Filmen bietet Ihre URSA Cine Immersive mehrere Möglichkeiten.

Der Klapptouchscreen ist zur Verwendung durch den Kameraoperator bzw. die -operatorin gedacht. Um das physische Profil der Kamera zu reduzieren, lässt sich der Touchscreen umkehren, wobei er weiterhin sichtbar bleibt. Alternativ positioniert man ihn nach vorne, wenn man das Monitoring während des Filmens selbst machen will.

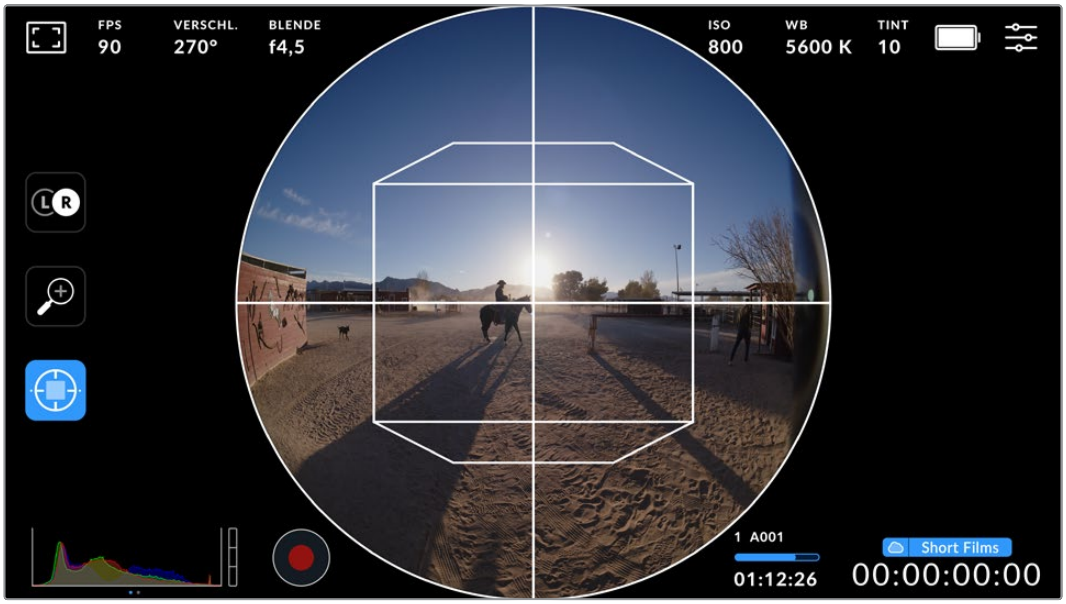
Der Assistenz-Touchscreen ermöglicht es einem anderen Crew-Mitglied, die Bildkontrolle vorzunehmen, ohne das Sichtfeld des Operators zu behindern. Aufgrund des extrem breiten Sichtfelds der Kamera ist darum bei Verwendung des Assistenz-Touchscreens Achtsamkeit geboten.





Monitoring-Optionen und Overlays

Die Kamera zeigt den als Lens Space bezeichneten kamerabezogenen Bild- bzw. Projektionsraum an. Die Ausklapp- und Assistenz-Touchscreens sind für die Anzeige eines Sichtfelds von 180 Grad ausgelegt. Die SDI-Ausgabe hingegen zeigt die vollen 210 Grad des von jedem der beiden Sensoren erfassten Bilds an.



Alle Monitoring-Ausgaben zeigen das Material für dasselbe Auge an. Um zwischen dem linken und dem rechten Auge zu wechseln, tippen Sie auf dem jeweiligen Bildschirm auf den L/R-Umschalter.

Es gibt ein optionales Overlay für die Bildschirme, das man durch Antippen des Overlay-Buttons aktiviert. Das Overlay besteht aus folgenden Elementen:

- Eine äußere 180-Grad-Begrenzung.
- Ein mittiges Fadenkreuz, das den Bildmittelpunkt markiert.
- Innenlinien, die beim Geradeaus-Gucken ungefähr dem Sichtfeld der Apple Vision Pro entsprechen.

Diese Werkzeuge wurden entwickelt, um eine genaue Kadrierung und Positionierung bei der Erfassung von immersivem Material zu unterstützen.

Kadrieren Ihrer ersten Einstellung

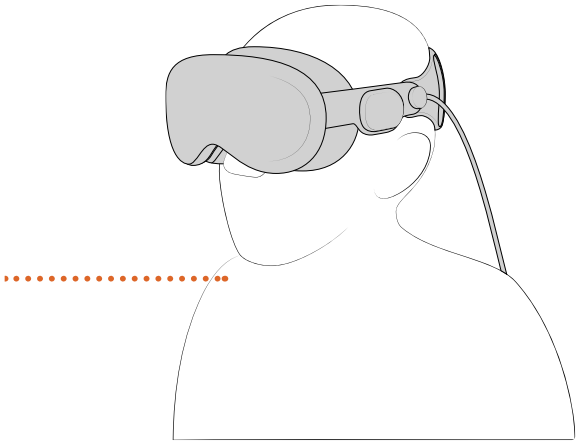
Immersive und traditionelle 2D-Inhalte haben viele Gemeinsamkeiten, aber auch deutliche Unterschiede.

Beispiel: Sind Sie im Auto schon einmal reisekrank geworden, weil Sie zu viel auf Ihrem Handy gescrollt haben? Die Reisekrankheit wird durch eine sensorische Diskrepanz verursacht, wenn Ihr Gehirn widersprüchliche Signale empfängt. Ihr Innenohr signalisiert Ihrem Körper, dass Sie sich bewegen, aber Ihre Augen signalisieren ihm, dass Sie stillstehen.

Mit Immersive-Inhalten lässt sich ganz einfach ein ähnliches Erlebnis erzeugen. Beispiel: Ein Kameraschwenk sagt den Augen des Zuschauers, dass sie sich bewegen, aber sein Nacken und sein Innenohr sagen ihm, dass sich überhaupt nichts bewegt. Daher lernt man am besten, zunächst stationäre Einstellungen zu meistern, bevor man mit Kamerabewegungen übt, die für das Publikum angenehme Aufnahmen erzielen.

Höhe der Kamera

Im traditionellen 2D-Kino positioniert man die Kamera hoch oder tief, um das Machtverhältnis in einer Szene zu verdeutlichen. Erwägen Sie beim Filmen von Immersive-Inhalten, welche Auswirkungen eine auf dem Boden platzierte Kamera hat. Wie mögen solche Bilder auf das Publikum wirken? Sie könnten dem Betrachtenden das Gefühl vermitteln, bis zum Nacken eingegraben zu sein.



Die wahrgenommene Höhe des Bodens kann das Seherlebnis Ihres Publikums beeinflussen. Eine niedrige Kameraposition vermittelt dem Publikum eine Bodenhöhe, die vielleicht keinen erzählerischen Sinn ergibt



Beim Drehen von Immersive können niedrige Blickwinkel dem Publikum Unwohlsein einflößen

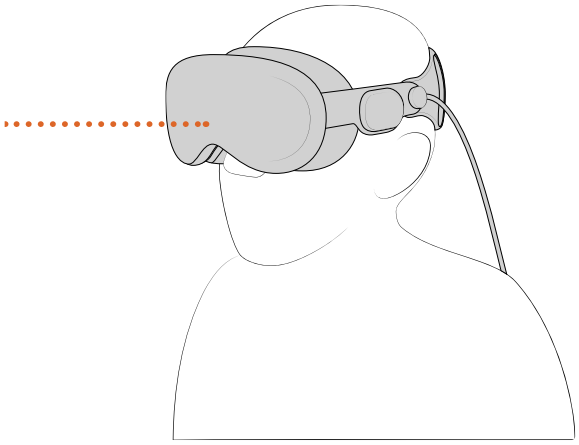
Um Zuschauenden das Gefühl zu vermitteln, physisch anwesend zu sein, empfiehlt es sich, eine Kamerahöhe und -position zu wählen, die ihnen suggeriert, in der Szene zu stehen oder zu sitzen. Ihnen bleibt dennoch genug kreativer Spielraum. Wenn sich Personen in der Szene befinden, kann eine etwas höher positionierte Kamera dem Betrachter ein Gefühl von Macht vermitteln, da er auf andere herabblicken kann. Umgekehrt zwingt eine etwas tiefer angebrachte Kamera den Betrachter dazu, zu anderen Personen aufzublicken.

Dennoch lässt sich der Komfort des Publikums auch mit gewagteren Kamerapositionen aufrechterhalten. Fliegende Kameras, Aufnahmen von oben, niedrige Blickwinkel und viele andere Techniken werden erfolgreich in immersiven Produktionen eingesetzt. Zusammen mit der passenden Montage, Bewegung und dem richtigen Kontext können sie zur Unterhaltung Ihres Publikums zu dienen.

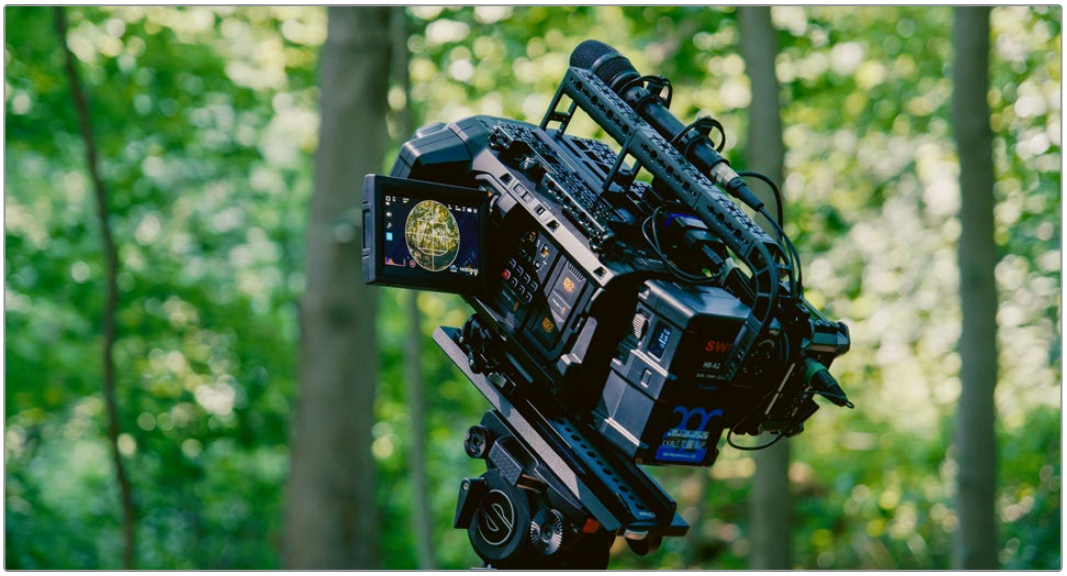
Vertikalschwenks der Kamera

Beim Filmen traditioneller 2D-Inhalte und der Kameraplatzierung in großer oder geringer Höhe gebrauchen Kameraleute häufig Auf- und Abwärtsschwenks für ihre Kompositionen. Erwägen Sie, wie sich das auf ein immersives Bild auswirkt.

Wenn Sie die Kamera nach unten neigen, damit Ihre Zuschauenden den Horizont sehen können, müssten sie ihre Köpfe anheben. Dies würde sie wahrscheinlich nicht reisekrank machen, könnte aber ihr Gefühl der Präsenz beeinträchtigen, da der Horizont im realen Leben immer direkt vor einem liegt.



Immersive Aufnahmen sind am glaubwürdigsten, wenn der Horizont bei null Grad liegt



Ein Abwärtsschwenk mit der URSA Cine Immersive Kamera lässt für die Zuschauenden den Eindruck entstehen, dass der Horizont an einer ungewohnten Stelle erscheint. Sie müssen dann an einen anderen, unerwarteten Punkt blicken, um den Horizont zu sehen

Daher werden die meisten immersiven Einstellungen mit einer Kameraneigung von null oder nahezu null Grad aufgenommen. Da die Kamera sowohl horizontal als auch vertikal ein Sichtfeld von über 180 Grad erfasst, kann das Publikum weiterhin nach oben und unten schauen. Dabei zwingt der Filmemachende das Publikum aber nicht, in eine bestimmte Richtung zu schauen, sondern überlässt dem Publikum die Kontrolle.

Filmemachende können den Blick des Publikums auch mit anderen Techniken auf verschiedene Bildbereiche lenken. Zum Beispiel mit Beleuchtung, Rahmung durch bildinterne Elemente oder mit blickführenden Linien innerhalb des Bilds.

Positionierung

Immersive Inhalte werden normalerweise im Sitzen angesehen. Da es sich nicht um ein Augmented-Reality- oder Virtual-Reality-Format handelt, stehen die Zuschauenden nicht und bewegen sich auch nicht. Sitzen Sie aktuell? Dann machen Sie dieses Experiment: Drehen Sie Ihren Kopf von einer zur anderen Seite und nach unten und oben. Wie weit können Sie den Kopf bequem drehen, ohne Ihren Körper zu bewegen? Das Sichtfeld des immersiven Formats übersteigt den für die meisten Menschen angenehmen Drehbereich. Das ist gut, weil dieses Format auch bei Drehungen nach links oder rechts weiterhin peripheres Sehen ermöglicht. Dieses Experiment hilft zu verstehen, wo die Hauptaktion innerhalb des Sichtfelds zu positionieren ist.

Normaler Bewegungsbereich des Kopfes:

Rotation: 120–160 Grad
Beugung und Streckung: 110–130 Grad

Kombiniert man die Kopfdrehung mit einer Augenbewegung, kann man über die eigene Schulter schauen. Für die meisten Menschen ist es aber unangenehm, lange in dieser Position zu verweilen.

Aktivieren Sie als Kadragehilfe das Overlay Ihrer Kamera. Durch Platzieren der Handlung innerhalb der inneren Linien, die das typische Sichtfeld anzeigen, verbessert dies die sichtfeldgerechte Wahrnehmbarkeit Ihrer Inhalte.

Auch die Bewegung von visuellen Schwerpunkten innerhalb des Bilds sollte bedacht werden. Um eine Person nachzuverfolgen, die sich von ganz links nach ganz rechts im Bild bewegt, sind extreme Kopfbewegungen erforderlich. Wenn von einer Einstellung, bei der sich der visuelle Schwerpunkt am Rand des Bilds befindet, zu einer Einstellung, bei der sich dieser Schwerpunkt am gegenüberliegenden Rand befindet, geschnitten wird, fordert das Zuschauenden eine extreme Kopfbewegung ab. Eine sorgfältig durchdachte Komposition kann die visuelle Komfortzone des Publikums verbessern und gleichzeitig eine kreative Kommunikation ermöglichen.

Handlungsspielraum des Publikums

Beim Filmen von herkömmlichen 2D-Inhalten können Sie die Aufmerksamkeit der Zuschauenden durch die Wahl von Bildausschnitt, Objektiv und Brennweite lenken. Beim Filmen von immersiven Inhalten hingegen ist es aufgrund des 180-Grad-Sichtfelds schwerer, Informationen auszublenden und den Blick des Publikums zu führen. Mit Immersive hat das Publikum erheblich mehr Entscheidungsfreiheit, Ihre Bilder eigenständig zu erkunden. Das ist einer der Faktoren, der immersive Inhalte derart fesselnd macht. Dennoch gibt es Möglichkeiten, die Aufmerksamkeit des Publikums durch die achtsame Verwendung von Sound, Beleuchtung, Choreografie und Komposition zu lenken.

Kamerabewegung

Eine immersive Kamera muss nicht statisch bleiben, sollte aber achtsam bewegt werden, da manche Kamerabewegungen unnatürlich oder unangenehm wirken können. Das Auto ist eine gute Analogie, um zu verstehen, wie man immersive Kamerabewegungen angenehm gestaltet. Ein großes Fahrzeug hat Masse und Trägheit, die plötzlichen Änderungen der Geschwindigkeit oder Richtung entgegenwirken. Als solches beschleunigt oder bremst man es allmählich, um es auf vorhersehbare und kontrollierte Weise zu bewegen. Wenn Sie die Aufnahme statischer Einstellungen meistern, üben Sie sich an bewegten, immersiven Einstellungen. Beachten Sie dabei folgende Faktoren, um Bewegungen ansprechend aufzunehmen:

- **Dauer:** Geben Sie Ihrem Publikum Zeit, sich an die Kamerabewegung zu gewöhnen und sie zu genießen. Kurz andauernde Bewegungen führen eher zu Orientierungslosigkeit.
- **Intensität:** Weniger intensive Bewegungen lassen sich leichter in einen immersiven Schnitt integrieren. Hochintensive Bewegungen sind möglich, allerdings lässt sich mit Aufwärmen und Abkühlen ggf. der Komfort des Publikums optimieren.
- **Visuelle Ankerpunkte:** Dabei handelt es sich um Bereiche der Szene, die sich nicht bewegen. Eine Einstellung ohne visuelle Ankerpunkte kann orientierungslos wirken. Vordergrundelemente wie das Innere eines Autos oder entfernte Objekte wie der Horizont helfen dabei, Ihre bewegten Aufnahmen zu erden.

Datenraten-Tabellen

Die nachstehenden Tabellen enthalten die Datendurchsatzraten für immersiven Content in 8K×7K-Auflösung mit 90 und 24 Frames pro Sekunde.

Maximale Datenraten bei 90 fps mit Blackmagic RAW bei konstanter Bitrate

8K×7K-Immersive			
Codec	**Anzahl der Pixel**	**Framerate**	**Datenrate**
Blackmagic RAW 5:1	8160 × 7200 Stereo	90	3203 MB/s
Blackmagic RAW 8:1	8160 × 7200 Stereo	90	2002 MB/s
Blackmagic RAW 12:1	8160 × 7200 Stereo	90	1335 MB/s
Blackmagic RAW 18:1	8160 × 7200 Stereo	90	889 MB/s

Maximale Datenraten bei 24 fps mit Blackmagic RAW bei konstanter Bitrate

8K×7K-Immersive			
Codec	**Anzahl der Pixel**	**Framerate**	**Datenrate**
Blackmagic RAW 5:1	8160 × 7200 Stereo	24	854 MB/s
Blackmagic RAW 8:1	8160 × 7200 Stereo	24	534 MB/s
Blackmagic RAW 12:1	8160 × 7200 Stereo	24	356 MB/s
Blackmagic RAW 18:1	8160 × 7200 Stereo	24	237 MB/s

Audio und Timecode

Der Ton ist ein wesentlicher Teil der immersiven Erfahrung. Darum ist es wichtig, Ihrer URSA Cine Immersive geeignetes Equipment für Tonaufnahmen an die Seite zu stellen.

Apple Spatial Audio Format (ASAF)

Das Apple Spatial Audio Format (ASAF) dient beim Ansehen von Apple Immersive Video zur Bereitstellung von 3D-Audio. ASAF ist Teil des umfassenderen Apple Positional Audio Framework und kann drei verschiedene Arten von Audioaufnahmen speichern.

- **Kanalbasiert:** Sie können einen herkömmlichen Surround-Mix in Ihre ASAF-Datei einbetten. Das ist ideal für Dinge wie am Kopf verankerte Erzählstimmen und Musik.
- **Objektbasiert:** Sounds können unabhängig von einzelnen Lautsprecherkanälen als einzelne Objekte mit Metadaten behandelt werden, die ihre Position im 3D-Raum beschreiben. Das sind normalerweise Monoaufnahmen. Ihre 3D-Koordinaten können animiert werden, um Objekten zu folgen, die sich innerhalb der Szene bewegen. Dies ist ideal für diegetisches Audio, das sich auf in der Szene sichtbare Objekte bezieht.
- **Ambisonic:** Ein Surround-Sound, der den gesamten 3D-Raum um die Hörposition unter Verwendung von Spherical Harmonics abdeckt. Er ist ideal für Atmos.

Der Rest dieses Abschnitts bietet einen Überblick über das erforderliche Equipment zum Erstellen einer effektiven ASAF-Datei.

Ambisonic

Ein Ambisonic-Mikrofon enthält vier oder mehr in einem präzisen geometrischen Muster angeordnete Kapseln. Jede Kapsel erfasst dieselben Geräusche mit unterschiedlichen Eintrittszeiten und unterschiedlicher Intensität, je nach der Richtung, aus der ein Geräusch kommt. Die Sounds dieser Kapseln werden zur späteren Verarbeitung mit einem Computer auf separaten Kanälen aufgezeichnet. Durch den Vergleich der Unterschiede kann die Software die Richtung und Art der aufgezeichneten Geräusche abschätzen.



Ein Ambisonic-Mikrofon erster Ordnung

Ambisonic-Mikrofone

Ambisonic-Mikrofone werden nach der Anzahl der darin enthaltenen Kapseln klassifiziert. Mehr Kapseln bedeuten, dass das Mikrofon räumliche Details genauer wiedergeben kann.

Erste Ordnung:	4 Kanäle
Zweite Ordnung:	9 Kanäle
Dritte Ordnung:	16 Kanäle
Vierte Ordnung:	25 Kanäle und mehr

Unabhängig davon, ob sich das Mikrofon an oder neben der Kamera befindet, müssen Ambisonic-Mikrofone korrekt auf die Aufnahmerichtung der Kamera ausgerichtet sein. Ambisonic-Mikrofone können in verschiedenen Konfigurationen eingesetzt werden, bspw. in Längsrichtung (endfire), seitlich (broadside), nach oben oder nach unten gerichtet. Darum ist es wichtig, dass Ihre Tontechnikleute die Art der verwendeten Ausrichtung kennen. Diese Informationen lassen sich über Metadaten auf dem Audiorekorder oder anhand von Notizen übermitteln.

Ambisonic-Aufnahmen können als Soundbed für die sphärische Klangumgebung verwendet werden. Man kann andere Kanäle und objektbasierte Soundeffekte darüber legen. Mit der korrekten Ausrichtung auf die Kamera erfasste Ambisonic-Aufnahmen können den Tonleuten auch als Referenz beim Hinzufügen weiterer Sounddesign-Elemente dienen.

Lavalier- und Minimikrofone

Wenn Sie mit einer 180-Grad-Kamera filmen, verbietet sich die Verwendung einer Tonangel, da sie in Ihrer Szene sichtbar wäre. Demnach werden Sie versteckte Mikrofone und Lavaliermikrofone verwenden müssen, um diegetischen Ton und Dialoge aufzunehmen.

Hier sind ein paar Anwendungsszenarien:

- Verbinden Sie diese Audioquellen über drahtlose Sender und Empfänger direkt mit den beiden XLR-Eingängen der URSA Cine Immersive.
- Schließen Sie sie an externe Audiorekorder an. Bspw. ein großes Mehrkanalgerät oder ein kleines, am Körper getragenes Gerät.

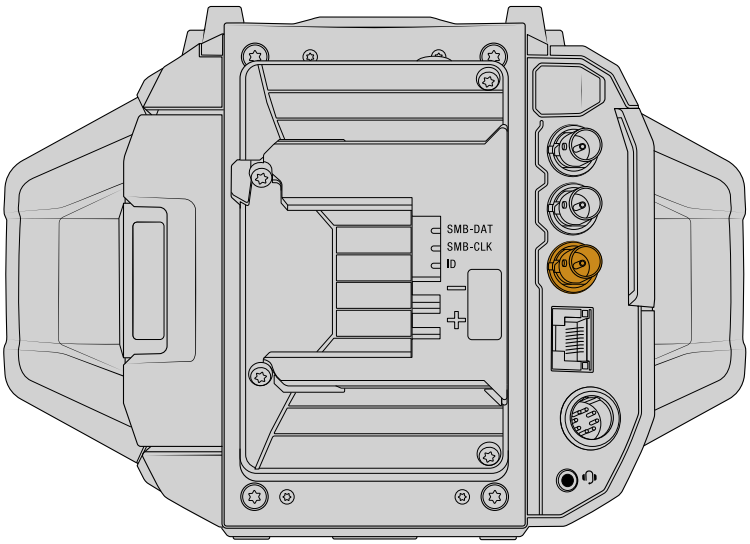
Die Tonaufnahmen können dann synchronisiert und entweder als kanal- oder objektbasierte Sounds gemischt werden.

Handmikrofone

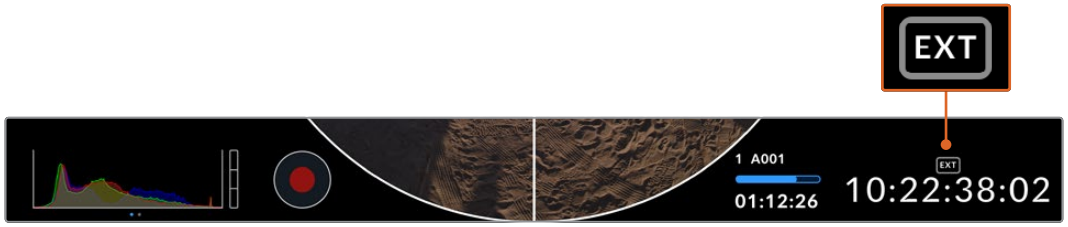
Indem man zusätzliche Soundeffekte vor Ort aufnimmt, nachdem die Kamera nicht mehr läuft, spart man potenziell Zeit in der Postproduktion und fängt möglicherweise noch dazu einen authentischeren Klang ein. Es lohnt sich, ein Handmikrofon mit Hypernierencharakteristik oder ähnlicher Richtwirkung mitzuführen, um damit Aktionen mit möglichst nah am Geschehen positioniertem Mikrofon zu wiederholen. Diese Aufnahmen können dann entweder als kanal- oder objektbasierte Sounds in den endgültigen Mix geschnitten werden.

Anschließen einer Timecode-Quelle

Um externen Timecode mit Ihrer URSA Cine Immersive zu verwenden, schließen Sie einen Timecode-Generator mit einem BNC-Kabel an den Timecode-Eingang an der Rückseite Ihrer Kamera an. Synchronisierter Timecode gewährleistet beim Filmen mit einem dualen System, dass Ton und Bild in der Postproduktion problemlos abgeglichen werden können.



Ist eine externe Timecode-Quelle mit passender Framerate an diesen Eingang angeschlossen, greift die Kamera den eingehenden Timecode automatisch auf. Auf dem LCD der URSA Cine Immersive erscheint beim Synchronisieren mit externem Timecode das Kürzel EXT. Wenn Sie das Kabel herausziehen, wird der Timecode einmalig übernommen und bleibt synchron mit der Kamera. Das nun erscheinende INT-Kürzel bedeutet, dass der Timecode nun vom internen Timecode-Generator Ihrer Kamera läuft.



Über dem Kamera-Timecode erscheint EXT, solange ein externer Timecode-Generator angeschlossen ist

HINWEIS Es gibt SMPTE-Timecode-Standards für gängige Frameraten wie 23,976, 24, 30 fps usw. Es gibt jedoch keinen SMPTE-Timecode-Standard für 90 fps. Darum verwendet die URSA Cine Immersive tatsächlich einen Timecode von 30 fps, obwohl ihre Projekt-Framerate auf 90 fps eingestellt ist. Die SDI-Ausgänge geben ebenfalls mit 30p aus. Setzen Sie verwendete Timecode-Geräte oder externe Tonrekorder ebenfalls auf 30 fps.

Filmen

Da Sie jetzt über grundlegende Kenntnisse zur Bedienung Ihrer Blackmagic URSA Cine Immersive Kamera verfügen und das Immersive-Format optimal zu nutzen wissen, können Sie sich in diesem Kapitel über weitere Funktionen der Kamera informieren.

Voraussetzungen für die Belichtung

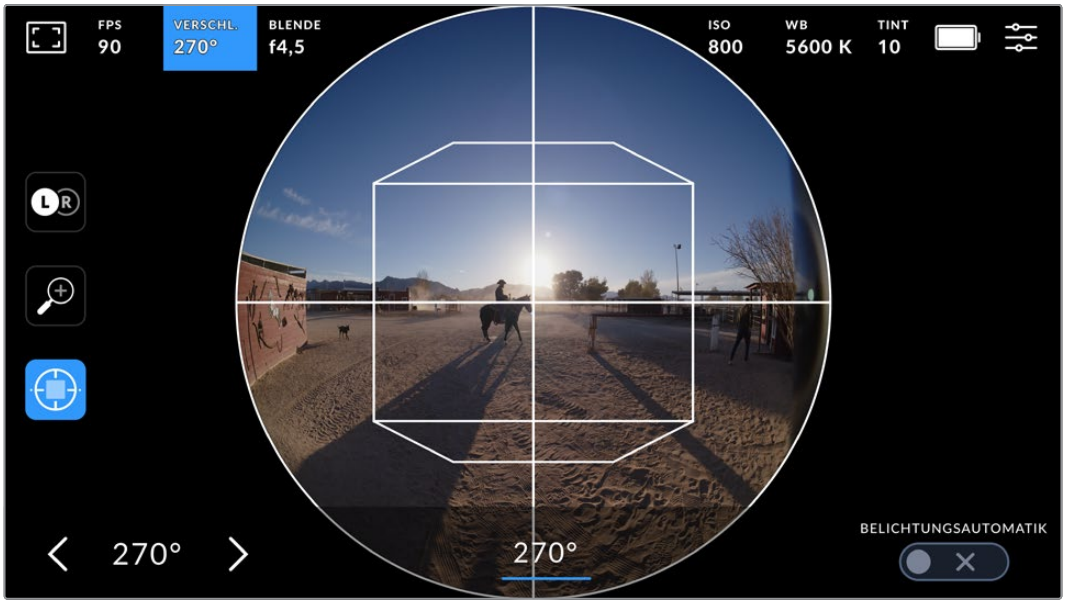
Apple Immersive Video erfordert mehr Licht als herkömmliches Filmen mit 24 fps, daher ist vielleicht ein Umdenken bei Ihrer Herangehensweise an Dreh und Beleuchtung angebracht. Wenn Sie einen 180-Grad-Verschluss verwenden, benötigt eine Aufnahme mit 90 fps fast viermal mehr Licht, um die gleiche Belichtung zu erzielen. Darüber hinaus ist die feste Blende der Objektive mit etwa f/4,5 bis zu drei Blendenstufen dunkler als bei typischen Super-Speed-Objektiven. Das heißt, Sie brauchen ggf. sieben Blendenstufen mehr Licht für eine angemessene Belichtung. Wenn Sie jedoch mit 90 fps filmen, kann ein großer Verschlusswinkel von 270°–360° helfen, den Lichtverlust bei hohen Frameraten auszugleichen und dennoch eine ansprechende Bewegungsunschärfe zu erzielen.

Verschlusszeit

Der Verschluss-Indikator zeigt den Verschlusswinkel bzw. die Verschlusszeit an. Die Einheit geben Sie im SETUP-Menü unter „Verschlusswert“ vor, wo Sie die Anzeige als Verschlusswinkel oder Verschlusszeit auswählen können.

So ändern Sie die Verschlusszeit:

1. Tippen Sie in der oberen Toolbar auf das Verschluss-Icon.
2. Durchlaufen Sie dann die gängigen Verschlusszeit-Presets anhand der Links- und Rechtspfeile. Um einen bestimmten Verschlusszeitwert auszuwählen, tippen Sie unten links auf dem Bildschirm zweimal auf die Anzeige des aktuellen Werts. Dann erscheint eine kleine Tastatur, über die Sie einen beliebigen Verschlusswert mit bis zu zwei Dezimalstellen von 11,2°–360° eintippen können.



Benutzerdefinierte Frameraten können ebenfalls über die Onscreen-Tastatur eingegeben werden

Die Verschlusszeit hat weniger Einfluss auf das Erscheinungsbild von immersiven Inhalten mit hoher Bildfrequenz. Obwohl viele Filmemachende immer noch einen 180°-Verschluss für Aufnahmen mit 24 oder 30p bevorzugen, ist es beim Filmen von immersivem Material üblicher, die Verschlusszeit als Hilfe für die Belichtung zu verwenden.

Beim Drehen bei wenig Licht kann man den Verschluss auf 270° oder in Ausnahmefällen sogar auf 360° erhöhen.

Beim Drehen unter hellen Bedingungen kann die Verschlusszeit auf bis zu 11,2° gesenkt werden. Es ist bei hellen Lichtverhältnissen jedoch üblicher, die Verschlusszeit auf der für die Bewegung bevorzugten Einstellung zu belassen und die Belichtung mit ND-Filtern zu steuern.

Es ist zu beachten, dass die hohe Bildwechselfrequenz von Apple Immersive Video mit 90 fps bei Aufnahmen mit flackernden Lichtern Probleme verursachen kann. Beim Filmen in 60-Hz-Umgebungen wird ein 270°-Verschluss empfohlen, in 50-Hz-Umgebungen ein 324°-Verschluss.

ISO und Dynamikumfang

Der ISO-Indikator zeigt die aktuelle ISO-Einstellung bzw. die Lichtempfindlichkeit Ihrer Kamera an. Durch Antippen dieser Indikatoren können Sie Ihren ISO-Wert an wechselnde Lichtverhältnisse anpassen. Die URSA Cine Immersive verfügt über ISO-Werte von 200–3200. Der optimale ISO-Wert liegt bei 800.



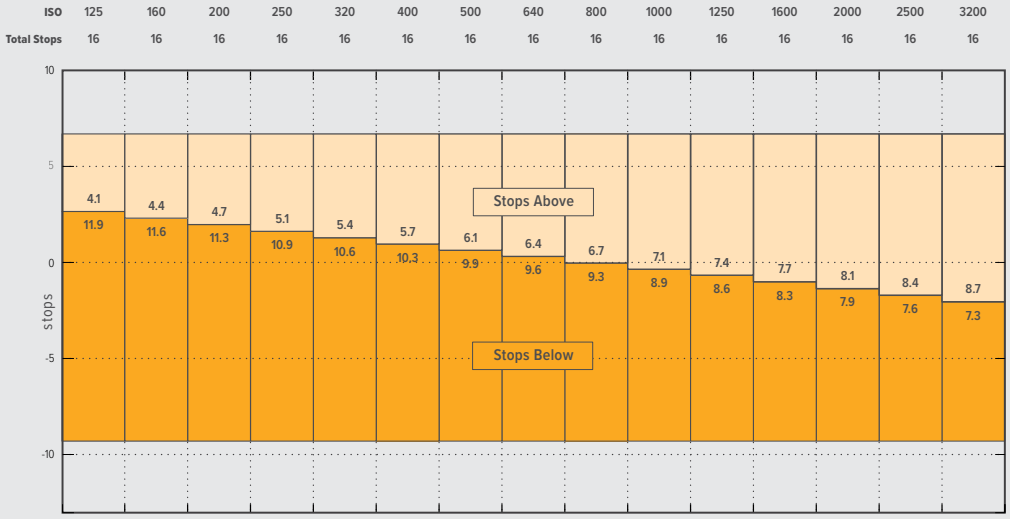
Der ISO-Wert lässt sich auch durch Antippen des ISO-Indikators konfigurieren

Numerische Presets ermöglichen die Wahl eines ISO-Werts in vollen Blendenstufen, indem man auf die entsprechende Zahl tippt. Mit dem Schieberegler unterhalb der Presets können Sie Ihre ISO-Einstellung in Werten zwischen diesen Presets in 1/3-Blendenstufen feinabstimmen.

Je nach Situation mag eine niedrigere oder höhere ISO-Einstellung geeigneter sein. Beispielsweise mag ein ISO-Wert von 1600 in Lowlight-Situationen geeignet sein, kann aber ein sichtbares Bildrauschen verursachen. Bei hellen Lichtverhältnissen bewirkt ein ISO-Wert von 200 ggf. stärker gesättigte Farben.

Blackmagic Kameras können breite Dynamikumfänge erfassen. Das gibt Ihnen viel Flexibilität beim Aufzeichnen von Bildern und beim Colorgrading in DaVinci Resolve. Die nachstehende Tabelle zeigt die mit Ihrer URSA Cine Immersive verfügbaren Dynamikumfänge an.

Dynamikumfänge der URSA Cine Immersive



ND

Zumeist sind die Neutraldichtefilter (ND) der Kamera die gängigsten Hilfen zur Regulierung der Belichtung. ND-Filter reduzieren die auf den Kamerasensor treffende Lichtmenge um eine vordefinierte Anzahl von Blendenstufen.



Die ND-Filter der URSA Cine Immersive werden mit den Tasten links und rechts oberhalb des Klapptouchscreens eingestellt. Die verfügbaren Einstellungen neben einem klaren Filter („Clear“) sind 2, 4, 6 und 8 Blendenstufen.

Die aktuelle ND-Einstellung zeigt der Indikator oben links auf dem Touchscreen an. Wenn ein ND-Filter verwendet wird, bleibt der Indikator ständig aktiviert. Bei Verwendung des klaren ND-Filters verschwindet der Indikator nach 4 Sekunden.



HINWEIS Die vom ND-Filterindikator angezeigte Einheit können Sie nach Belieben ändern. Verwenden Sie das SETUP-Menü, um eine der Optionen ND-Zahl, Blendenstufe oder Bruchzahl vorzugeben.

Offspeed-Aufzeichnung

Die Projekt-Framerate im SETUP-Menü ist die Framerate des Aufzeichnungsformats der URSA Cine Immersive. Sie bestimmt die Geschwindigkeit, mit der Ihr Bildmaterial wiedergegeben wird. Die Framerate wird normalerweise so eingestellt, dass sie der in Ihrem Postproduktions-Workflow verwendeten Wiedergabegeschwindigkeit und der Audiosynchronisation sowie den Liefervorgaben entspricht.



Tippen Sie auf den „FPS“-Indikator auf dem Bildschirm Ihrer URSA Cine Immersive, um die Framerate-Einstellungen zu öffnen. Ihre Blackmagic Cine Immersive verfügt über neun Projekt-Frameraten: 23,98; 24; 25; 29,97; 30; 50; 59,94; 60 und 90 Bilder pro Sekunde. Beim Filmen im Immersive-Format, sollten Sie die Projekt-Framerate auf 90 fps einstellen.

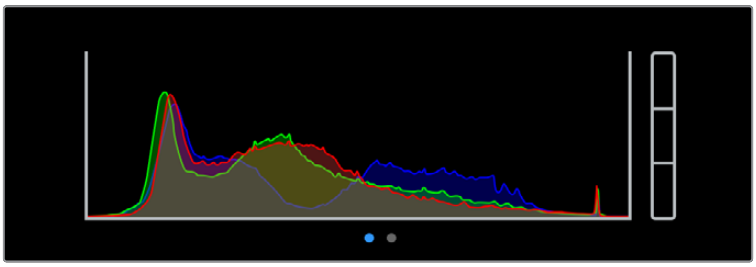
Anhand der Sensor-Framerate bestimmen Sie, mit welcher Geschwindigkeit Ihr Sensor Bilder aufnimmt. Standardmäßig sind die Projekt- und Sensor-Frameraten Ihrer URSA Cine Immersive auf eine natürliche Wiedergabegeschwindigkeit eingestellt. Wenn Sie jedoch rechts unten im FPS-Menü Ihrer Kamera auf den Schalter OFFSPEED-FRAMERATE tippen, können Sie die Sensor-Framerate frei wählen.

Ist der Schalter OFFSPEED-FRAMERATE deaktiviert, können Sie die Sensor-Framerate manuell einstellen. Es ist erwähnenswert, dass 90 fps die höchste Bildrate der Kamera ist und der Projekt-Framerate von 90 fps zum Filmen von Apple Vision Pro Material entspricht. Wenn Sie jedoch eine niedrigere Projekt-Framerate wie 30 oder 60 fps wählen und mit einer Geschwindigkeit von 90 fps drehen, können Sie wunderschöne Zeitlupenaufnahmen erstellen.

Ist der Schalter OFFSPEED-FRAMERATE aktiviert, zeigt die Bildratenanzeige oben links auf dem Bildschirm die Sensor-Bildrate gefolgt von der Projekt-Bildrate an.

Histogramm

Unten links auf dem Touchscreen Ihrer URSA Cine Immersive wird ein Histogramm angezeigt. Ein Histogramm ist ein Graph, der die Helligkeitsverteilung in einem Bild aufzeigt. Links werden die Schatten, in der Mitte die Mitteltöne und ganz rechts die Lichter dargestellt. Die Höhe des Graphen gibt an, wie viel Prozent des Bilds diese Helligkeit aufweisen.



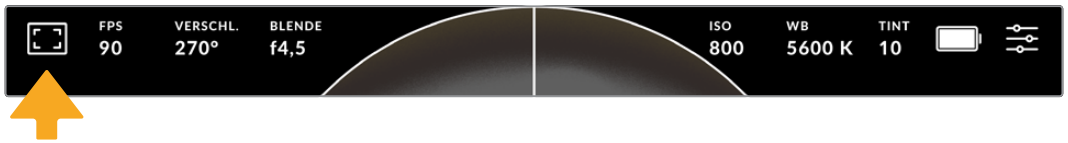
Das Histogramm gibt Ihnen auf einen Blick einen Eindruck von der Tonwertverteilung in Ihrem Clip. Ein Bild, dessen Graph sich auf der linken Seite befindet, ist wahrscheinlich unterbelichtet. Entsprechend ist ein Bild mit dem Graphen auf der rechten Seite wahrscheinlich überbelichtet. Das Histogramm sollte immer im Zusammenhang mit dem Bild interpretiert werden. Im Allgemeinen weist ein gut belichtetes Bild jedoch einen ausgewogenen, mittig liegenden Graphen auf. Erhöht man die Belichtung, verschiebt sich das Histogramm nach rechts, verringert man sie, verschiebt es sich nach links.

Der Indikator rechts neben dem Histogramm zeigt an, wenn die Tonwerte in einzelnen Farbkanälen abgeschnitten werden. In der Regel sollten Sie ein solches Clipping in allen Kanälen vermeiden. Es gibt jedoch Situationen, in denen Clipping im Bild akzeptabel ist, wenn sich so die Mitteltöne und Schatten bewahren lassen.

„Expose to the Right“ ist eine Technik, bei der Sie ein Bild absichtlich überbelichten, indem Sie das Histogramm so weit wie möglich nach rechts verschieben, ohne die Farbkanäle zu beschneiden. Dies maximiert das Signal und reduziert Bildrauschen. Solange die Belichtung in der Postproduktion reduziert wird, ist dies eine hervorragende Methode zur Erstellung klarer Bilder.

LCD-Monitoring-Optionen

Durch Antippen des Monitor-Icons oben links auf dem Touchscreen Ihrer URSA Cine Immersive gelangen Sie zu den LCD-Monitor-Einstellungen. Dieses Menü enthält nützliche Optionen wie die Bildschirmhelligkeit. Auch bietet es leistungsstarke Belichtungswerkzeuge.



Zebra

Mit der „Zebra“-Einstellung verändern Sie das Erscheinungsbild der Zebra-Funktion auf dem Touchscreen und stellen gleichzeitig das Zebra-Level ein. Im Zebra-Modus werden auf Bildbereichen, die den voreingestellten Belichtungspegel übersteigen, diagonale Linien eingeblendet. Die Prozenteinstellung wird verwendet, um überbelichtete Bereiche in Ihrem Bild anzuzeigen. Üblicherweise setzt man diesen Wert auf 95 % statt auf 100 %, da Sie dies die Bildbereiche identifizieren lässt, deren Belichtung korrigiert werden kann. Mit der zweiten Zebra-Einstellung können Sie die mittleren grauen Zebra-Bereiche ein- oder ausschalten oder auf mittleres Grau plus 1 Blendenstufe einstellen.

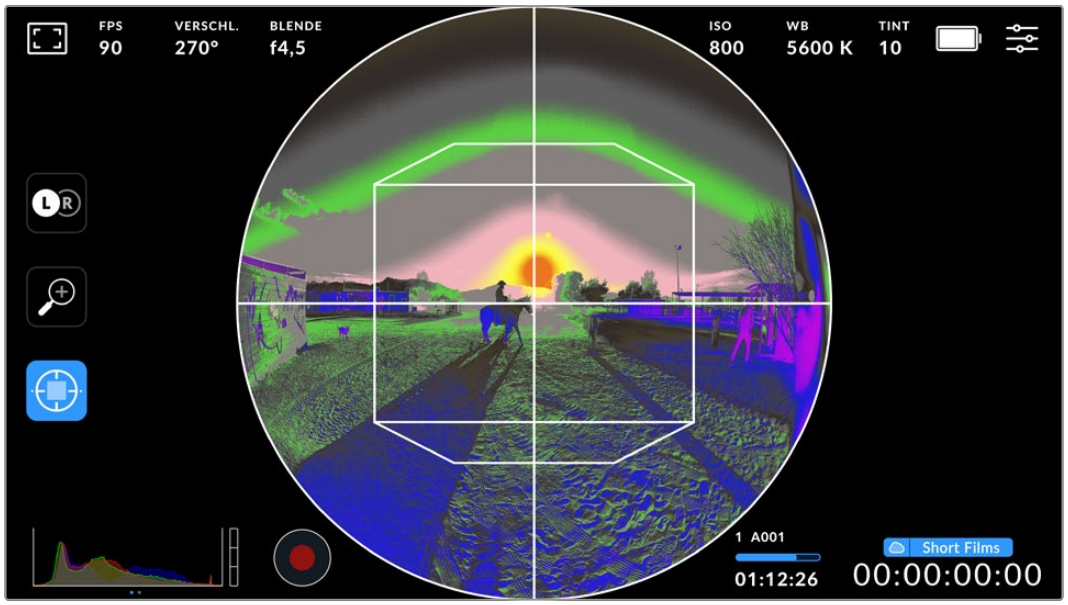
Raster

Über die „Raster“-Einstellung wählen Sie eines der Kompositionsraster zur Anzeige auf Ihrem LCD-Touchscreen: „Drittel“, „Horizont“, „Fadenkreuz“ oder „Punkt“. Zudem legen Sie hier fest, welches Overlay auf allen Ausgaben der URSA Cine Immersive sichtbar sein soll. Beim Filmen von immersiven Inhalten ist die Wasserwaage sehr nützlich. Sie zeigt ein doppeltes Fadenkreuz in der Bildmitte an, das bei vollständig waagerecht stehender Kamera ausgerichtet ist. Sobald die Kamera nicht mehr waagerecht steht, laufen die Linien des Fadenkreuzes auseinander.

Falschfarben

Falschfarben-Overlays stellen unterschiedliche Helligkeitswerte von Elementen in Ihrem Bild in verschiedenen Farben dar. Pink repräsentiert bspw. eine optimale Belichtung für hellere Hauttöne, während Grün für dunklere Hauttöne besser geeignet ist. Wenn Sie beim Filmen von Menschen anhand der Falschfarben-Funktion Pink oder Grün kontrollieren, lässt sich eine gleichmäßige Belichtung von Hauttönen erzielen.

Gleichermaßen zeigt Ihnen eine vom Gelben ins Rote wechselnde Falschfarben-Darstellung an, dass diese Bildbereiche überbelichtet sind. Ähnlich wie das Histogramm vermittelt Ihnen dieses Werkzeug einen sehr schnellen und intuitiven Überblick über die Tonwertverteilung im Bild.



Aufnehmen

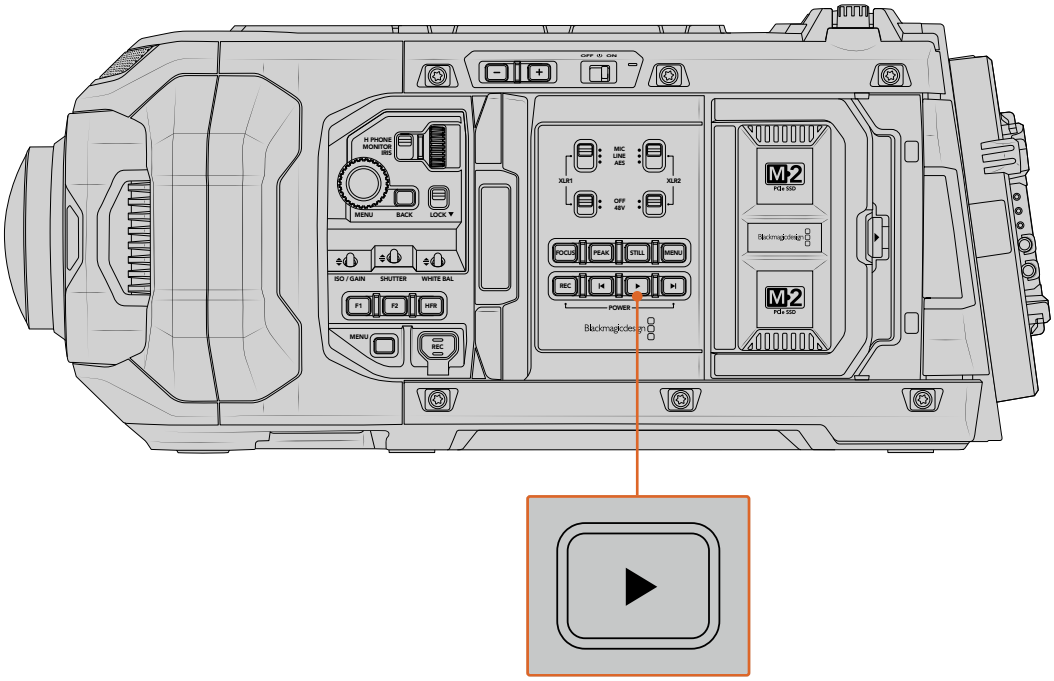
Starten Sie die Aufnahme durch Drücken einer der roten Aufnahmetasten (REC) der Kamera. Es befindet sich je eine REC-Taste am Frontbedienfeld, am internen Bedienfeld hinter dem Klapptouchscreen, am Bedienfeld der Assistenzstation und an beiden Touchscreens. Drücken Sie die REC-Taste erneut, um die Aufnahme zu stoppen.

Die Speicher-Indikatoren zeigen die verbleibende Aufnahmezeit an. Die Zeit wird in Stunden:Minuten:Sekunden entsprechend der von Ihnen gewählten Framerate angezeigt. Je nach aktuellem Status werden die Datenträgerbalken über der Aufnahmezeit in blau, weiß oder rot angezeigt. Sie stellen den Füllstand des Datenträgers dar.



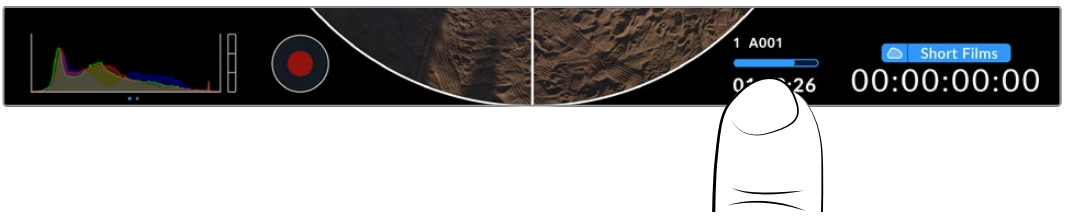
Wiedergeben

Nachdem Sie Ihre Clips aufgezeichnet haben, können Sie sie mithilfe der Transporttasten wiedergeben. Um den zuletzt aufgenommenen Clip wiederzugeben, drücken Sie auf der linken Kameraseite die Wiedergabetaste hinter dem Ausklapptouchscreen. Mit den Vor- und Rücklauftasten können Sie zu anderen Clips navigieren. Um den Wiedergabemodus zu beenden und die Kamera für die Fortsetzung der Aufnahme vorzubereiten, drücken Sie einmal auf die Aufnahmetaste.

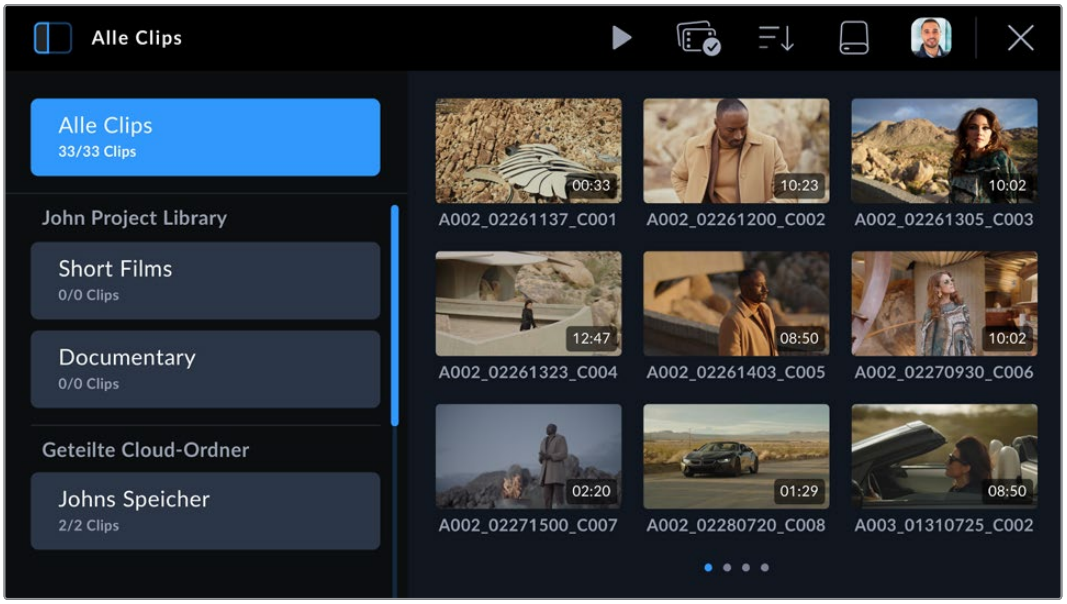


Drücken Sie die Vor- oder Rücklauftaste, um zum Anfang oder Ende eines Clips zu springen. Einmaliges Drücken der Rücklauftaste bringt Sie an den Anfang des aktuellen Clips zurück, zweimaliges Drücken bringt Sie zum Anfang des vorherigen Clips zurück. Halten Sie die Vor- oder Rücklauftaste gedrückt, um Ihren Clip in doppelter Geschwindigkeit abzuspielen oder zurückzuspulen. Drücken Sie beim Spulen zweimal die Vor- bzw. Rücklauftaste, um die Geschwindigkeit zu vervierfachen. Dreimaliges Drücken beschleunigt sie um das Achtfache und viermaliges um das Sechzehnfache.

Für eine umfassendere Überprüfung Ihres aufgenommenen Filmmaterials werden Sie wahrscheinlich die Mediathek bevorzugen. Tippen Sie zum Öffnen der Mediathek unten auf dem Touchscreen Ihrer Kamera auf einen der Speicherindikatoren.



Auf der Browser-Startseite werden Vorschaubilder von allen Clips auf den an Ihre Kamera angeschlossenen Datenträgern angezeigt. Die Vorschaubilder werden in drei Reihen angezeigt. Um in weiteren Reihen gespeicherte Clips zu sichten, wischen Sie auf dem Touchscreen zur nächsten Vorschaubilder-Seite. Die Anzahl der Seiten mit verfügbaren Clips erkennen Sie an der Anzahl der unten auf dem Display angezeigten Punkte.



Verbinden Sie die URSA Cine Immersive mit einem Blackmagic Cloud-Konto und senden Sie das Filmmaterial direkt an ein DaVinci Resolve-Projekt.

Sie können die Mediathek jederzeit verlassen, indem Sie oben rechts auf das X tippen oder einmal auf eine der Aufnahmetasten der Kamera drücken.

Transferieren von Clips

Es gibt mehrere Methoden, um Clips aus Ihrem Kameraspeicher in andere Speicher zu übertragen. Bspw. können Sie Ihre URSA Cine Immersive über ein 10G-Ethernet-Kabel mit Ihrem Computer oder Netzwerk verbinden und Clips aus dem Speicher Ihrer Kamera über die Ethernet-Verbindung kopieren. Alternativ können Sie sogar das Media Modul in ein Blackmagic Media Dock laden, das mit Ihrem Netzwerk oder Computer verbunden ist, und gleichzeitig drei Media Module verwenden.

Wenn Sie Ihre Kamera per Ethernet anschließen, erkennt der Computer das Blackmagic Media Module auf dieselbe Weise wie beim Anschließen eines Blackmagic Cloud Stores.

Da Ihre Kamera standardmäßig auf DHCP eingestellt ist, weist Ihr Computer oder Netzwerk der Kamera automatisch eine IP-Adresse zu.

Aktivieren von Dateifreigaben

Wenn Sie beim erstmaligen Einrichten Ihre Kamera via Ethernet an Ihren Computer oder Ihr Netzwerk anschließen, ist die SMB-Dateifreigabe standardmäßig deaktiviert. Das ist so, weil Nutzende anders als mit einem Blackmagic Cloud Store die Dateien ggf. nicht sofort für andere Netzwerkteilnehmer freigeben oder ihnen nach einem wichtigen Drehtag nicht dieselben Zugriffsrechte für die Kameramasterdateien erteilen wollen. Darum haben wir dafür gesorgt, dass Sie die SMB-Dateifreigabe selber aktivieren können.

1. Um das Teilen von Dateien zu aktivieren, müssen Sie Blackmagic Camera Setup herunterladen und auf Ihrem Computer installieren. Das Dienstprogramm ist eine Verwaltungssoftware zum Aktualisieren der Produktsoftware Ihrer Kamera und Ändern der Verwaltungseinstellungen. Blackmagic Camera Setup steht zum Download auf unserer Website unter https://www.blackmagicdesign.com/de/support/family/professional-cameras bereit.
2. Nachdem Sie das Setup-Dienstprogramm heruntergeladen und installiert haben, schließen Sie Ihren Computer via USB an. Führen Sie dann Blackmagic Camera Setup aus. Verbinden Sie Ihren Computer über den USB-Port im hinteren Bereich der Kameraoberseite.

3 Um die Einstellungen zu öffnen, klicken Sie auf das Bild Ihrer URSA Cine Immersive oder auf das Einstellungssymbol.

4 Scrollen Sie abwärts zu den Netzzugriffseinstellungen „Network Access“ und aktivieren Sie die Dateifreigabe „File sharing (SMB)“. Tippen Sie anschließend unten rechts auf „Save“, damit diese Einstellungen wirksam werden.

5 Gehen Sie dann zu den URL-Einstellungen weiter unten und klicken Sie auf das URL-Icon, um die Adresse automatisch zu laden. Die URL-Adresse zeigt den Namen Ihrer Kamera als lokalen Netzwerkstandort an.

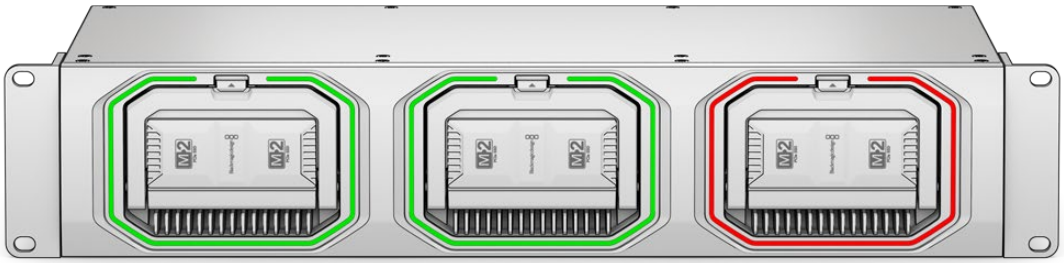
6 Bestätigen Sie mit „Save“.

Gehen Sie direkt zu Ihren Clips und klicken Sie auf den URL-Namen Ihrer Kamera. Das nun erscheinende Fenster fordert Sie auf, ein Passwort einzugeben oder sich als Gast anzumelden. Klicken Sie auf das „Guest“-Optionsfeld und klicken Sie auf „Connect“.

7 Ihr Computer navigiert nun über den Finder (Mac) oder Windows Explorer (Windows) automatisch zu Ihrem Media Modul. Das Dienstprogramm speichert Ihr Ethernet-Setup und Sie brauchen es nur beim erstmaligen Setup einzurichten. Beim nächsten Anschließen der Kamera via Ethernet haben Sie sofortigen Zugriff auf das Media Modul, ohne diese Einstellungen erneut vorzunehmen.

Verwenden eines Blackmagic Media Dock

Blackmagic Media Module sind zum schnellen Laden und Entnehmen konzipiert, damit Sie Clips schnell an externe Speicher übermitteln oder ein volles Modul mit einem leeren austauschen und weiter aufzeichnen können. Über das Blackmagic Media Dock kann ohne Kamera auf das Filmmaterial eines Media Moduls zugegriffen werden.



Zusammenfassung

Apple Immersive Video führt einen neuen Ansatz für visuelles Storytelling ein und bietet eine Alternative zum herkömmlichem 2D-Kino. Es soll kein Ersatz sein, sondern ist als zusätzliches Format gedacht, für das man sich je nach den Erfordernissen eines Projekts entscheiden kann. Wie bei jedem Format hängen kreative Entscheidungen von der erzählten Geschichte ab.

Häufig gehen neue Formate mit komplexen Workflows einher. Die URSA Cine Immersive wurde jedoch ausdrücklich dafür entwickelt, das Filmen von Apple Immersive Video zu vereinfachen. Sie ermöglicht einen geradlinigen Produktionsweg für immersive Inhalte und kann auch für visuelle Effekte und andere Anwendungen verwendet werden.

Dieser Leitfaden beschreibt die wichtigsten Funktionen und Workflows der URSA Cine Immersive. Er ist als Starthilfe gedacht. Wie bei jedem Kamerasystem empfehlen sich weitere Versuche und Tests, um die besten Ergebnisse für Ihre spezifischen Produktionsanforderungen zu erzielen.

DaVinci Resolve Studio



DaVinci Resolve Studio ist eine umfassende Postproduktionssoftware, die die Tools für Schnitt, Farbkorrektur, visuelle Effekte, Motion Graphics und die Tonnachbearbeitung nahtlos vereint. Sie unterstützt Workflows, die auf die einzigartigen Eigenschaften des Blackmagic RAW Immersive Formats zugeschnitten sind – darunter stereoskopische Videos mit ultrahoher Auflösung, Bildraten von 90 fps und das Mischen von Spatial Audio. Die komplett für die Produktion, Prüfung und Bereitstellung von Apple Immersive Video ausgerüstete Softwarevariante DaVinci Resolve Studio ist ein leistungsstarkes Werkzeug für Entwicklerinnen und Entwickler immersiver Inhalte.

Workstation-Empfehlungen

Hardware-Empfehlungen

- Mac Studio M3 Ultra Desktop
- Monitor mit hoher Bildwiederholfrequenz (mindestens 60 Hz)

Mindestanforderungen für Betriebssysteme

- macOS 15 oder höher
- 64-GB-Arbeitsspeicher
- Fürs Monitoring die Blackmagic Design Software Desktop Video 12.9 oder höher
- Auf Apple Silicon basierter Computer oder eine GPU, die Metal unterstützt

Anforderungen für immersives Monitoring

- Apple Vision Pro
- Hochgeschwindigkeits-WLAN-Netzwerk (mindestens Wi-Fi 5; Wi-Fi 6 bevorzugt)

Projekteinrichtung

Bevor Sie die Rohdaten der Blackmagic URSA Cine Immersive importieren, sollten Sie die Projekteinstellungen von DaVinci Resolve überprüfen, um sicherzustellen, dass die Daten für eine optimale Bildausgabe und einfache Bearbeitung korrekt debayered und verarbeitet wurden.

Haupteinstellungen

Geben Sie bei der Einrichtung eines Projekts unbedingt an, dass Sie Inhalte für die Apple Vision Pro erstellen werden. Dies gewährleistet, dass Ihnen alle richtigen Einstellungen und Workflow-Optionen angezeigt werden, während Sie Ihr Projekt vorbereiten, zusammenstellen und liefern.

1. Öffnen Sie DaVinci Resolve Studio.
2. Klicken Sie unten rechts im „Projektmanager“-Fenster auf „Neues Projekt“.
3. Geben Sie einen Projektnamen ein, bestimmen Sie den Speicherort der Medien (sofern erforderlich) und klicken Sie auf „Erstellen“.
4. Öffnen Sie die „Projekteinstellungen“, indem Sie auf der DaVinci Resolve Studio Bedienoberfläche unten rechts auf das Zahnradsymbol klicken.

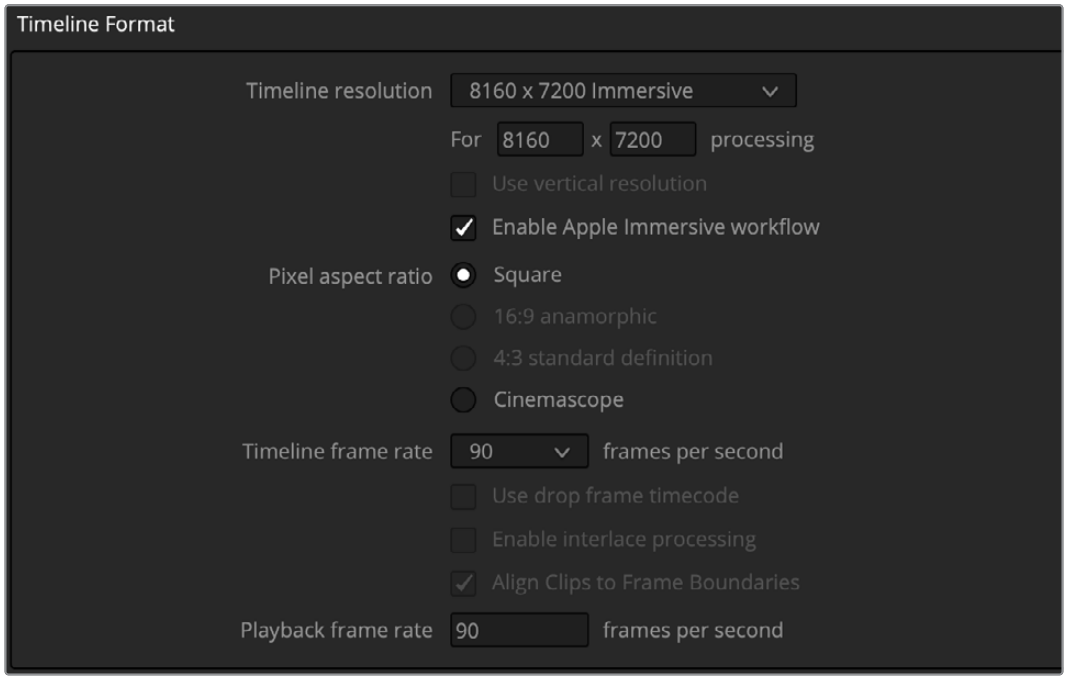
   Geben Sie als Erstes die Auflösung und die Framerate für das Projekt vor.
5. Markieren Sie in der Seitenleiste die Option „Haupteinstellungen“.
6. Wählen Sie im Bereich „Timeline-Format“ die Option „Apple Immersive Workflow aktivieren“.



   Dies bewirkt die Konfiguration von mehreren DaVinci Resolve Einstellungen auf die am besten für immersive Workflows geeigneten. Wählen Sie, bevor Sie ein immersives Projekt erstellen, immer zuerst diese Einstellung.
7. Setzen Sie die „Timeline-Auflösung“ auf „8160 x 7200 Immersive“, damit die Clips in ihrer nativen Sensorauflösung verarbeitet werden.

   Wählen Sie beim Arbeiten mit beschränkter Rechenleistung „4080 x 3600 Immersive“, um die Auflösung für eine schnellere Wiedergabe zu halbieren.
8. Setzen Sie die „Timeline-Framerate“ auf 90 Frames pro Sekunde. Dieser Parameter wird automatisch konfiguriert, wenn Sie einen Apple Immersive Workflow aktivieren.
9. Setzen Sie die „Wiedergabe-Framerate“ für die akkurateste Darstellung Ihres immersiven Videocontents auf 90 Frames pro Sekunde.

   Wer mit beschränkter Rechenleistung (oder einem Bildschirm mit langsamerer Bildwiederholrate) arbeitet, sollte die Framerate für eine stabile Wiedergabe weiter reduzieren. Empfohlen wird eine Framerate, die einen Bruchteil der nativen Framerate beträgt, bspw. 30 fps oder 60 fps.



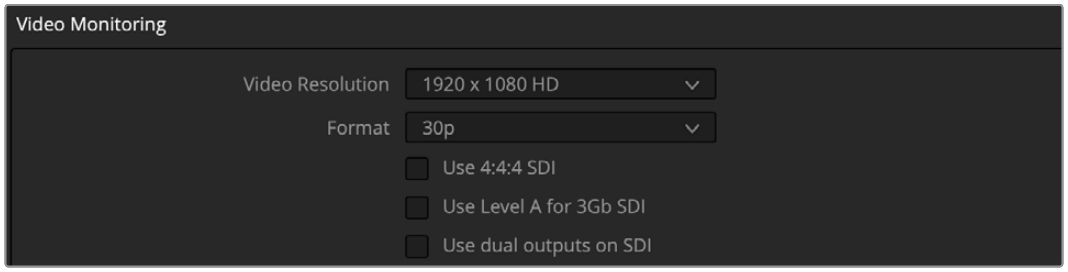
Video-Monitoring

Wenn Sie einen externen Bildschirm verwenden (über ein Capture-/Playback-Gerät wie eine DeckLink oder UltraStudio), konfigurieren Sie die Einstellungen fürs Videomonitoring gemäß der Spezifikationen Ihres Bildschirms.

1. Geben Sie im Bereich „Video-Monitoring“ eine „Video-Auflösung“ vor, die der Auflösung Ihres Bildschirms entspricht. Bspw. 1920 x 1080 HD oder 3840 x 2160 Ultra HD.

   Obwohl die native Bildwechselrate von immersivem Video 90 fps beträgt, unterstützen die wenigsten Broadcast-Monitore diese Bildwechselrate.

2. Stellen Sie das Format passend zur Framerate Ihres Monitors ein. Bspw. 30p.



Bildskalierung

Mit den Einstellungen für die Bildskalierung wird gewährleistet, dass die Auflösung und das Seitenverhältnis der immersiven Medien im Viewer korrekt angezeigt werden. Diese Parameter wurden automatisch konfiguriert, nachdem Sie die Option „Apple Immersive Workflow“ in den Haupteinstellungen aktiviert haben. Sie sollten jedoch trotzdem überprüfen, ob sie korrekt sind.

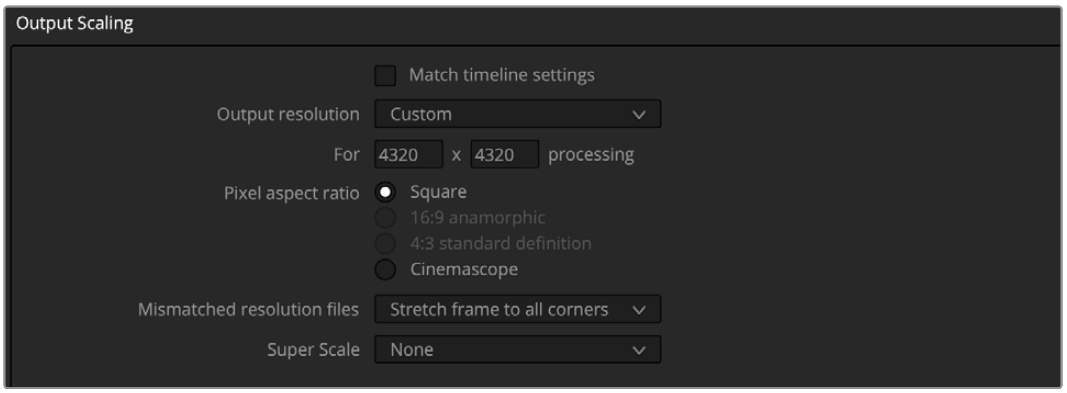
1. Wählen Sie in der „Projekteinstellungen“-Seitenleiste die Option „Bildskalierung“.
2. Vergewissern Sie sich, dass im Bereich „Ausgabeskalierung“ die Option „Timeline-Einstellungen anpassen“ deaktiviert ist.

3 Vergewissern Sie sich, dass die „Ausgabeauflösung“ auf „Individualisiert“ mit „**4320 x 4320**“ gesetzt ist.

Oder für kleinere Bildschirme und/oder schnellere Wiedergabe auf die halbe Auflösung (**2160 x 2160**).

Diese Auflösungen mit quadratischem Seitenverhältnis sind erforderlich, um Apple Immersive Videoinhalte in DaVinci Resolve Studio wiederzugeben und an die Apple Vision Pro zu streamen (über WLAN oder Developer Strap).

4 Vergewissern Sie sich, dass „Bei nicht übereinstimmender Auflösung“ die Option „Frame an alle Ecken anpassen“ gewählt ist.



Farbmanagement

Aufgrund der Notwendigkeit einer benutzerdefinierten Steuerung der maximalen Nit-Luminanz wird die Verwendung von Farbmanagement mithilfe von CST-Nodes (Color Space Transform) im Farbe-Modul empfohlen. Das Ziel der Farbmanagement-Einstellungen in den Projekteinstellungen besteht darin, den Timeline- bzw. „Arbeits“-Farbraum im Farbe-Modul festzulegen und die Farbraum-Metadaten zu definieren, die mit der gerenderten Datei verknüpft werden sollen.

1 Wählen Sie in der „Projekteinstellungen“-Seitenleiste die Option „Farbmanagement“.

2 Lassen Sie für „Farbmetrik“ die Option „DaVinci YRGB“ aktiviert.

3 Geben Sie für „Timeline-Farbraum“ die Option „DaVinciWG/Intermediate“ vor.

Dies ist ein zukunftssicherer Farbraum mit großem Farbumfang, der HDR-Workflows unterstützt und sich somit ideal für Medien aus der Blackmagic URSA Cine Immersive und Deliverables in Apple Vision Pro eignet.

4 Geben Sie für den „Ausgabe-Farbraum“ die Optionen „P3-D65“, „Rec. ST2084“ sowie 1000 Nit“ vor.

Dieser Farbraum verfügt über den geeigneten Farbumfang fürs Monitoring der DaVinci Resolve Viewer auf einem Mac Studio Desktop, der in der Regel den Farbraum Display P3 verwendet. Wenn Sie einen Grading-Monitor oder ein anderes Apple ICC-Profil verwenden, sollte der Ausgabefarbraum an den Farbstandard Ihres Bildschirms angepasst werden.

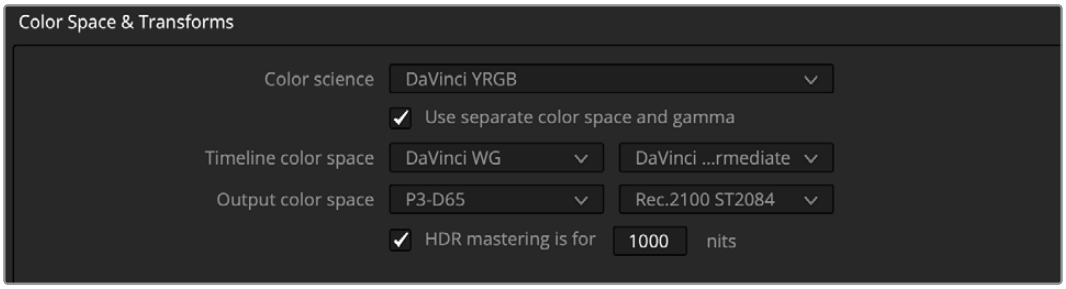
Sie müssen jedoch die Gamma-Werte (Übertragungsfunktion) ändern, um eine korrekte Zuordnung zum ICC-Profil zu gewährleisten.

5 Klicken Sie oberhalb der Farbraumeinstellungen auf „Separaten Farbraum und Gamma verwenden“.

Das trennt die Drop-down-Felder für Farbraum und Gamma.

6 Stellen Sie den Gammawert für die Ausgabe (rechte Spalte) auf „Rec. 2100 ST2084“ ein.

7 Geben Sie neben „HDR-Mastering ist für“ den Wert 1000 Nit vor.



Um weniger Zeit mit der Einrichtung Ihres nächsten Apple Immersive Projekts aufzuwenden, können Sie die aktuellen Einstellungen als Voreinstellung speichern.

8 Gehen Sie in der oberen rechten Ecke des „Projekteinstellungen“-Fensters zum Drei-Punkte-Optionsmenü und wählen Sie „Aktuelle Einstellungen als Voreinstellung speichern“.

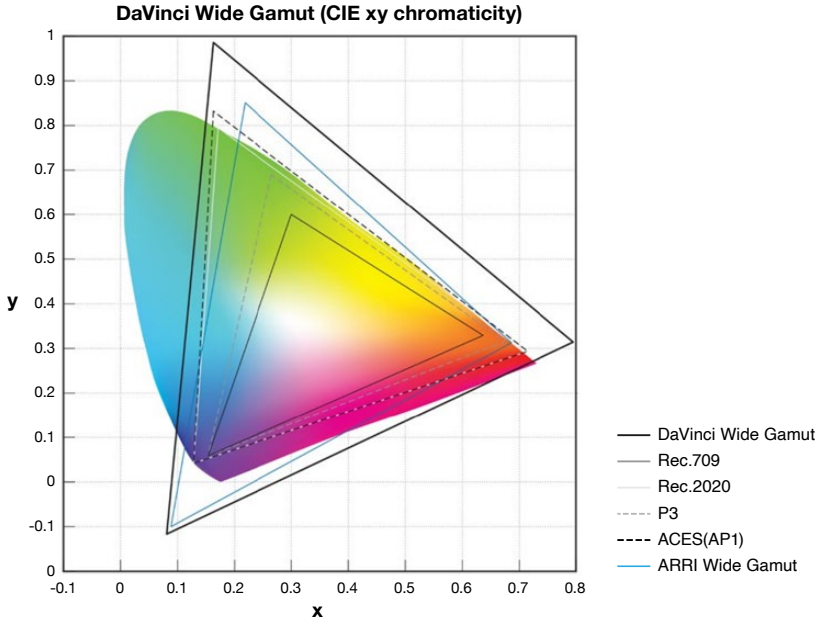
9 Benennen Sie die Voreinstellungen und klicken Sie auf OK.

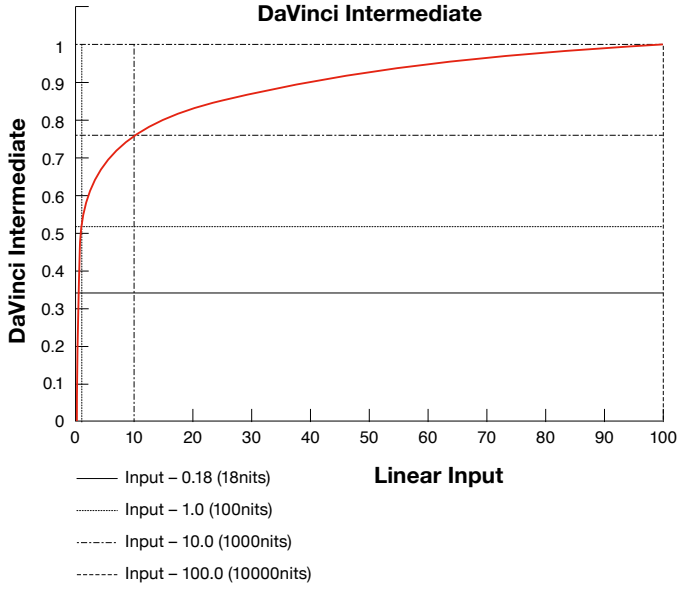
Zur schnellen Konfiguration der Projekteinstellungen, wenn Sie das nächste Mal mit immersiven Medien arbeiten, öffnen Sie das Fenster „Projekteinstellungen“, klicken Sie auf das Optionsmenü und wählen Sie die aufgeführte Voreinstellung aus.

10 Schließen Sie die Projekteinstellungen, indem Sie auf „Speichern unter“ klicken.

Infos DaVinci Wide Gamut

Der DaVinci Wide Gamut ist der fortschrittliche interne Farbraum von Blackmagic Design. Er wurde für DaVinci Resolve entwickelt und bietet eine außergewöhnlich breite und präzise Umgebung für die Farbkorrektur und Bildbearbeitung. In Kombination mit DaVinci Intermediate Gamma bildet er einen einheitlichen Arbeitsraum, der das Farbmanagement bei gemischten Kameraformaten vereinfacht und eine konsistente Tonwert- und Sättigungszuordnung von der Eingabe bis zur Ausgabe gewährleistet. Dieser breite Gamut ist ideal für die Arbeit mit HDR- und immersiven Medien. Er ermöglicht es Coloristinnen und Coloristen, nuancierte Anpassungen vorzunehmen, ohne dabei Streifenbildung oder Tonwert-Clipping beim Mastern für High-Fidelity-Ausgabeformate zu riskieren.





Verarbeiten von Apple Immersive

In der Seitenleiste „Projekteinstellungen“ können Sie im Abschnitt „Apple Immersive“ benutzerdefinierte ILPD-Daten (Immersive Lens Profile Data) festlegen, um die Kalibrierung und Projektion der immersiven Inhalte anzupassen.

1. Wählen Sie in der Seitenleiste „Projekteinstellungen“ die Option „Apple Immersive“.
2. Klicken Sie auf „Kalibrierungsordner öffnen“.
3. Fügen Sie eine benutzerdefinierte ILPD-Datei hinzu oder aktualisieren Sie sie.
4. Klicken Sie auf „Listen aktualisieren“, um über die Mediathek verfügbare ILPD-Inhalte neu zu laden.
5. Schließen Sie die Projekteinstellungen, indem Sie auf „Speichern unter“ klicken.
6. Rechtsklicken Sie in der Mediathek auf einen immersiven Clip.
7. Navigieren Sie im Kontextmenü zu „Apple Immersive Kalibrierung“ und wählen Sie entweder die Quell-ILPD („Keine Apple Immersive Kalibrierungsdatei ausgewählt“), die benutzerdefinierte ILPD oder die Gap-ILPD.

Standardmäßig befindet sich der benutzerdefinierte ILPD-Kalibrierungsordner im ProIM-Ordner, neben dem Standard-LUT-Ordner.

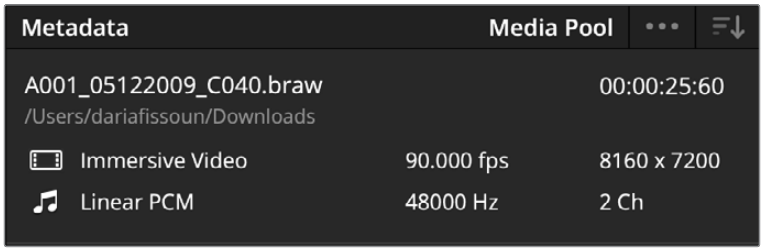
Ingesting

Beim Einspielen von immersiven Inhalten im Rohformat (RAW) wird das Material für beide Augen und alle zugehörigen immersiven Metadaten mit dem importierten BRAW-Clip übertragen.

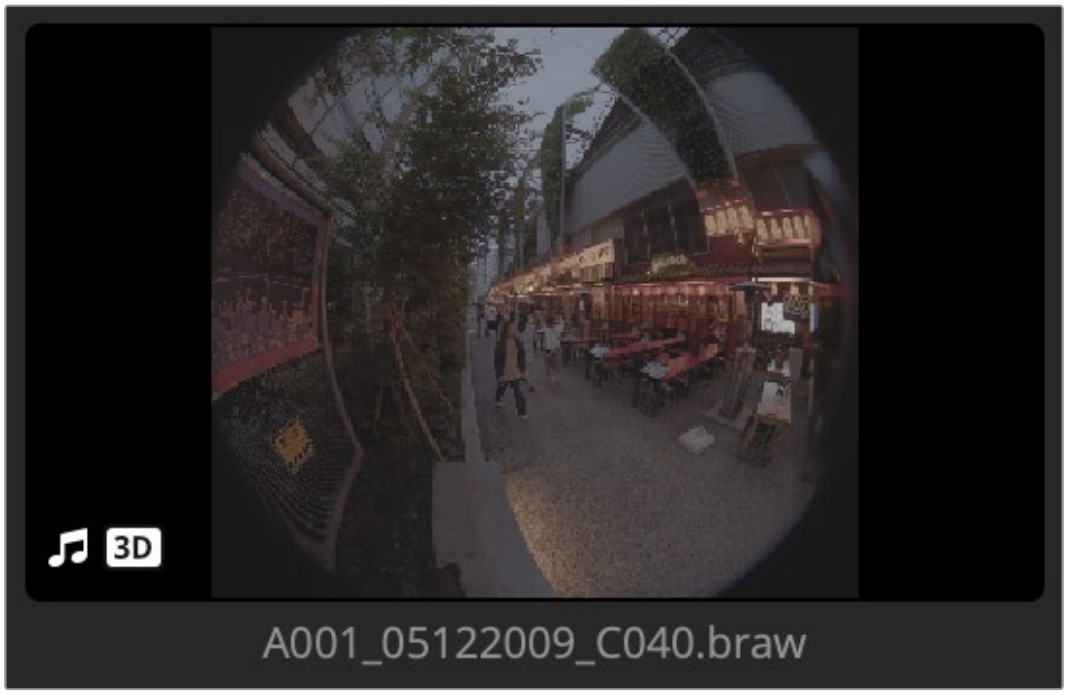
1. Gehen Sie zum Medien-Modul.
2. Navigieren Sie im „Medienspeicher“-Feld zum Speicherort des immersiven Videocontents.
3. Ziehen Sie die Medien per Drag-and-drop in die Mediathek darunter.

TIPP Um die ursprüngliche Ordnerstruktur zu bewahren, ziehen Sie Ordner vom „Medienspeicher“-Feld in die Seitenleiste der Mediathek. Beim Verschieben in den Master-Ordner (oder einen anderen von Ihnen erstellten Ordner) bleiben Ordner und Unterordner erhalten.

Wird ein Apple Immersive-Videoclip gewählt, zeigt der Codec-Parameter im Kopfbereich des „Metadaten“-Fensters an, dass es sich um ein Immersive Video handelt, zusammen mit der nativen Bildrate und Auflösung.



Zusätzlich bestätigt ein 3D-Symbol in der unteren linken Ecke der Videominiaturansicht, dass es sich um einen 3D-Clip handelt.



HINWEIS Wenn Sie zum ersten Mal immersive Medien in die Mediathek ziehen und dort ablegen, wird ggf. ein Dialogfenster angezeigt, in dem Sie zur Auswahl aufgefordert werden, die Einstellungen beizubehalten oder sie zu ändern. Bei Auswahl von „Einstellungen ändern“ werden als „Projekteinstellungen“ automatisch die Framerate und Auflösung der Quellmedien übernommen.

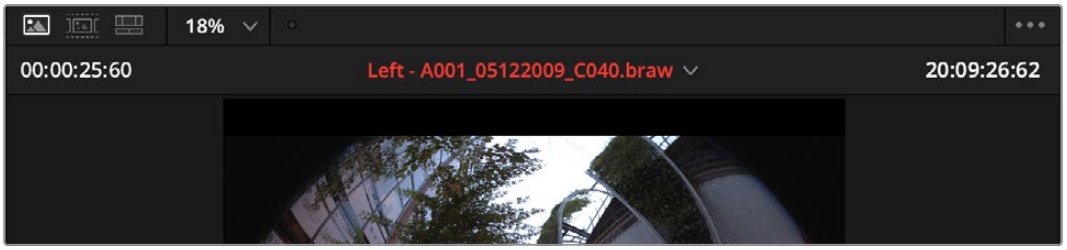
Schnitt

Bei der Arbeit mit immersiven Inhalten werden weitgehend dieselben Schnittwerkzeuge und -aktionen wie beim Arbeiten mit herkömmlichen 2D-Videoformaten verwendet. Die folgenden Anweisungen beschreiben Schnitt-Funktionen oder Aktionen, die speziell für immersive Video-Workflows gelten.

Zusammenschnitt

Bei der Vorschau von Clips während der Timeline-Zusammenstellung können Sie festlegen, welches Auge (links oder rechts) Ihnen im Quell-Viewer angezeigt wird. Ansonsten bleibt der Prozess des Zusammenschneidens einer Montage in der Timeline unverändert gegenüber den Standard-NLE-Workflows.

1. Öffnen Sie das Montage-Modul.
2. Doppelklicken Sie auf einen Clip, um ihn im linksseitigen Quell-Viewer zu öffnen. Das Label oben im Viewer zeigt an, das Material welchen Auges Sie gerade betrachten.



3. Um das im Viewer angezeigte Material für das jeweils andere Auge zu sichten, klicken Sie mit der rechten Maustaste auf den Quellclip in der Mediathek und wählen Sie „Stereoskopisches 3D“ > „Stereoskopischer 3D-Modus“ > „Linkes Auge“/„Rechtes Auge“/„Beide Augen“.

TIPP Wenn Sie die Live-Medienvorschau in den Optionen des Quell-Viewers aktiviert haben, können Sie auch mit der Maus über die Miniaturansichten in der Mediathek fahren, um Clips im Quell-Viewer zu scrubben.

4. Verwenden Sie die Tastenkombinationen I und O, um einen Bereich im Quellcode-Viewer zu definieren.
5. Klicken Sie in der Toolbar auf „Clip einfügen“ (oder verwenden Sie die Tastenkombination F9), um Clips in die Timeline einzufügen.

TIPP Sie können Clips auch per Drag-and-drop aus der Mediathek in das Timeline-Fenster ziehen.

6. Verwenden Sie Ihre Maus (im Auswahl- oder Trimm-Schnittmodus), um Clips in der Timeline zu trimmen.

Backdrops

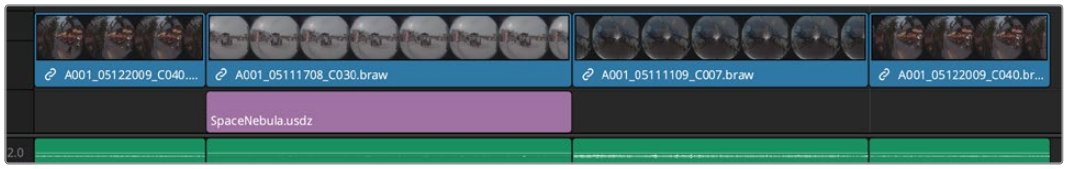
Wenn Sie in den Projekteinstellungen „Apple Immersive Workflow aktivieren“ auswählen, erscheint automatisch eine Backdrop-Spur (BD) unter der Spur „Video 1“.

Backdrops beziehen sich auf den Bereich, den der Betrachter sieht, wenn er über den Rand des immersiven Videoinhalts hinausblickt, der in der Regel auf ein Sichtfeld von 180–210 Grad beschränkt ist.

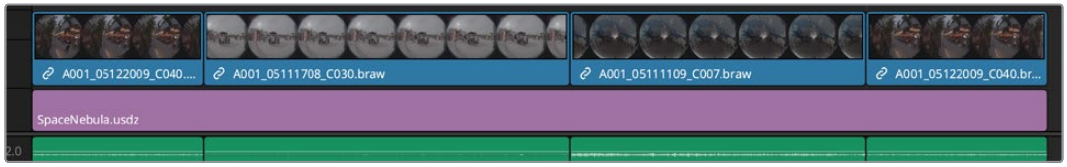
Standardmäßig ist dieser Bereich schwarz, aber Sie können über die Backdrop-Spur individualisierte Hintergrundbilder (im USDZ-Format) einfügen.

1 Ziehen Sie eine USDZ-Backdrop-Datei in die Backdrop-Spur, um sie der Timeline hinzuzufügen.

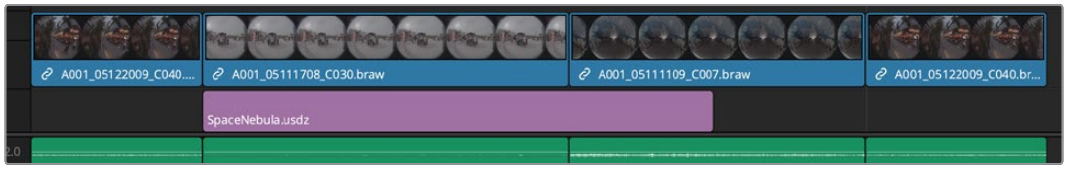
Die Dauer der Backdrop-Bildclips muss mit den Clip-Grenzen in den Videospuren übereinstimmen. Eine einzelne Backdrop-Datei kann sich über einen Clip, mehrere Clips oder die gesamte Timeline erstrecken, aber sie kann nicht in die Mitte eines immersiven BRAW-Clips (sprich „L-Schnitt“) geschnitten werden.



Ein Clip – richtig ✓

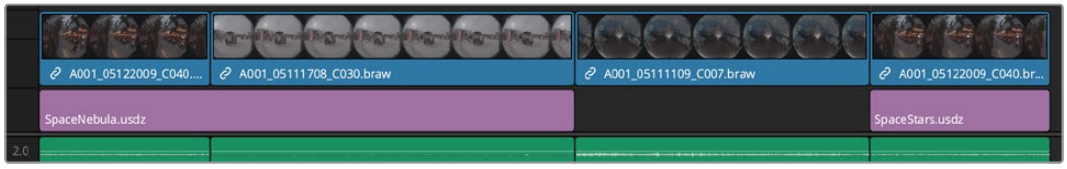


Mehrere Clips – richtig ✓



L-Schnitt – falsch ✗

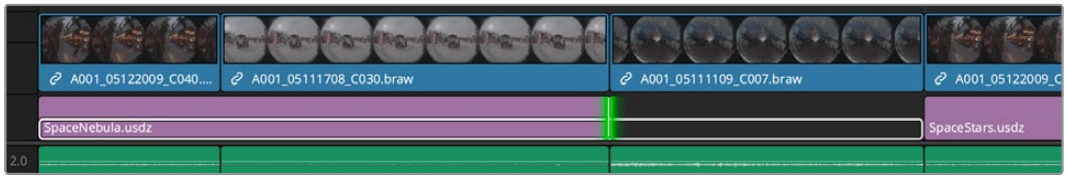
Um den standardmäßigen schwarzen Hintergrund beizubehalten, können Sie mehrere Backdrop-Dateien innerhalb einer einzigen Timeline verwenden oder gar kein Bild hinzufügen.



Verschiedene Bilder und Lücken – richtig ✓

2 Verwenden Sie die Standard-Trimmsteuerung, um die Länge einer Backdrop-Datei anzupassen.

Ziehen Sie die Kante der USDZ-Datei (im Auswahl- oder Trimmschnitt-Modus), um die Länge anzupassen.



Sie können auch die Befehle , (Komma) und . (Punkt) verwenden, um die Schnittkante jeweils um einen Frame zu verschieben.

TIPP Um sicherzustellen, dass Backdrops immer an den Schnitten für Immersive Video ausgerichtet sind, aktivieren Sie den Koppelmodus in der Toolbar des Montage-Moduls.

Verwenden des Viewers

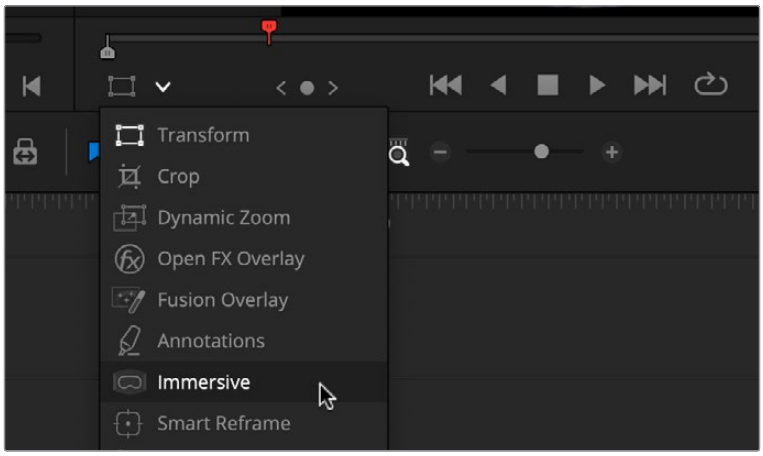
Standardmäßig werden immersive Clips in den Viewern der DaVinci Resolve Module im kreisförmigen „Lens Space“ angezeigt. Die zur Projektion des Bilds verwendete visuelle Objektivkalibrierung stammt von der Kamera und ist für jeden Clip eindeutig.



Sie haben auch die Möglichkeit, den Ansichtsmodus auf „LatLong“ oder „Viewport“ umzustellen.

1 Klicken Sie in unten links im Timeline-Viewer auf den Dropdown-Pfeil neben Button für die Bildschirmsteuerung.

2 Wählen Sie „Immersive“ aus.



Über dem Viewer erscheint eine neue Toolbar mit zusätzlichen Anzeigeoptionen.

3 Klicken Sie auf „LatLong“, um eine gleichwinklige Projektion des Bilds angezeigt zu bekommen. Dieser Modus verzerrt das Bild, sodass es den Proportionen traditioneller 2D-Medien ähnlicher ist, was bestimmte Postproduktionsprozesse wie Tracking und Compositing erleichtert.



4 Klicken Sie auf „Viewport“, um einen mittigen Ausschnitt der immersiven Szene in einem quadratischen Rahmen zu betrachten. Dieser Modus ist hilfreich, um die Details einer Szene auf einem Standardmonitor in den richtigen Proportionen klar zu sehen. Achten Sie aber darauf, dass Ihnen keine Aktionen außerhalb des Bildausschnitts entgehen.

Sie können die Transformationssteuerung in der immersiven Toolbar oben verwenden, um das stereoskopische Bild im Rahmen zu neigen und zu schwenken sowie das horizontale und vertikale Sichtfeld anzupassen.



Übergänge

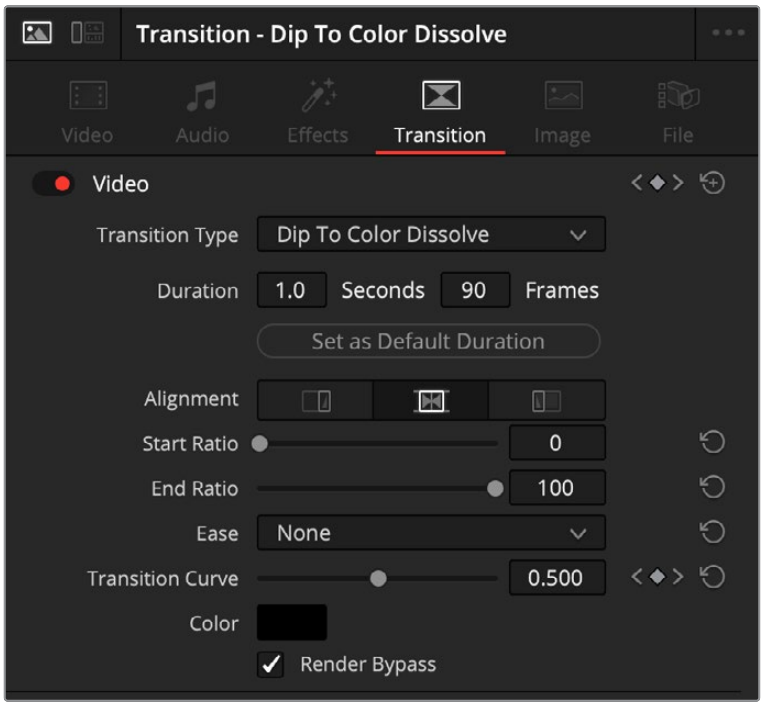
Für einen Übergang zwischen zwei immersiven Videoclips können Sie entweder einen direkten Schnitt durchführen, den Überhang an beiden Enden des Clips zum Ein- oder Ausblenden verwenden oder einen Blende-über-Farbe-Effekt anwenden. Übergänge sollten nicht in das Filmmaterial eingebrannt werden, denn sie werden während der Wiedergabe von der Apple Vision Pro selbst gerendert.

1. Öffnen Sie das „Effekte“-Feld.
2. Navigieren Sie in der Seitenleiste zu „Toolbox“ > „Videoübergänge“.
3. Ziehen Sie den Übergang „Blende über Farbe“ auf einen Schnittpunkt zwischen zwei Clips.



HINWEIS Sie müssen über die Schnittpunkte beider Clips hinaus genügende Trimmdaten haben, um einen Übergang anwenden zu können. Wenn nicht genügend Trimmdaten (Überhang) vorhanden sind, wird der Übergang nicht angewendet.

4. Navigieren Sie im Inspektor-Panel zum „Übergänge“-Tab und wählen Sie „Render umgehen“. So wird gewährleistet, dass die Überblendung von der Apple Vision Pro korrekt gerendert wird.
5. Anschließend können Sie die verbleibenden Parameter im Inspektor für „Blende über Farbe“ verwenden, um die Dauer, Ausrichtung, Animationsgeschwindigkeit und Farbe des Übergangs anzupassen.



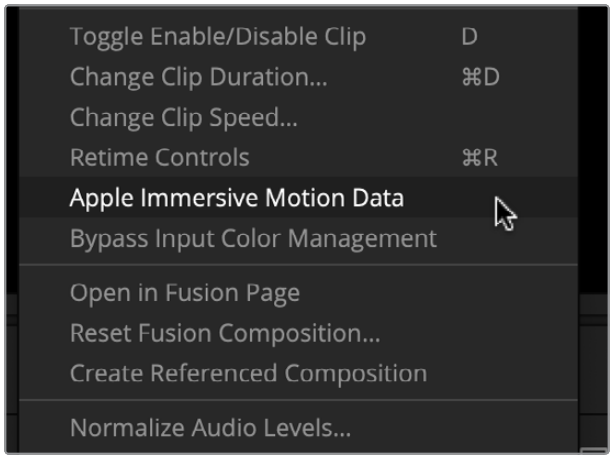
Apple Immersive Bewegungsdaten

Bei der Arbeit mit immersiven Inhalten können schnelle oder plötzliche Kamerabewegungen die Betrachtenden verwirren. Vermeiden Sie zur Gewährleistung des Zuschauerkomforts Aufnahmen mit schnellen Schwenks/Neigungen, Drehbewegungen oder ungleichmäßigen Horizonten.

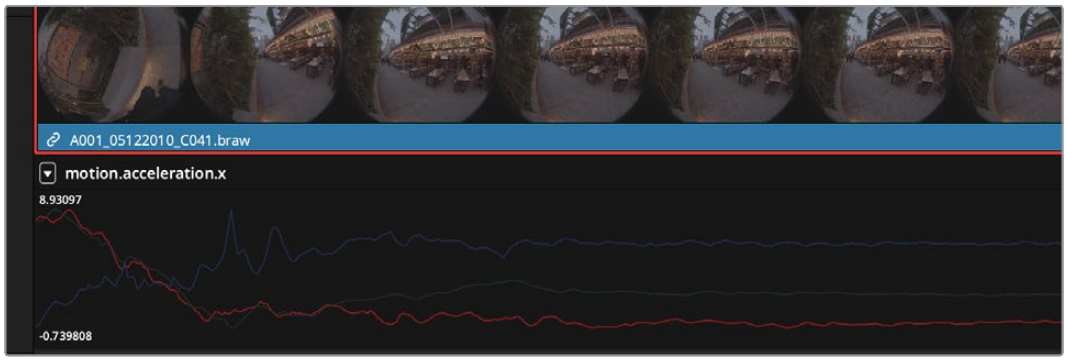
Die Funktion „Apple Immersive Bewegungsdaten“ stellt die x-, y- und z-Beschleunigungs- sowie die Gyroskop-Positionswerte grafisch dar, um objektiv aufzuzeigen, wo Kamerabewegungen und andere Bewegungen im Bild möglicherweise zu stark sind. Editoren können anhand dieser Informationen davon absehen, verwirrende Abschnitte des Filmmaterials zu verwenden.

1 Rechtsklicken Sie auf einen immersiven Clip in der Timeline.

2 Wählen Sie „Apple Immersive Bewegungsdaten“.

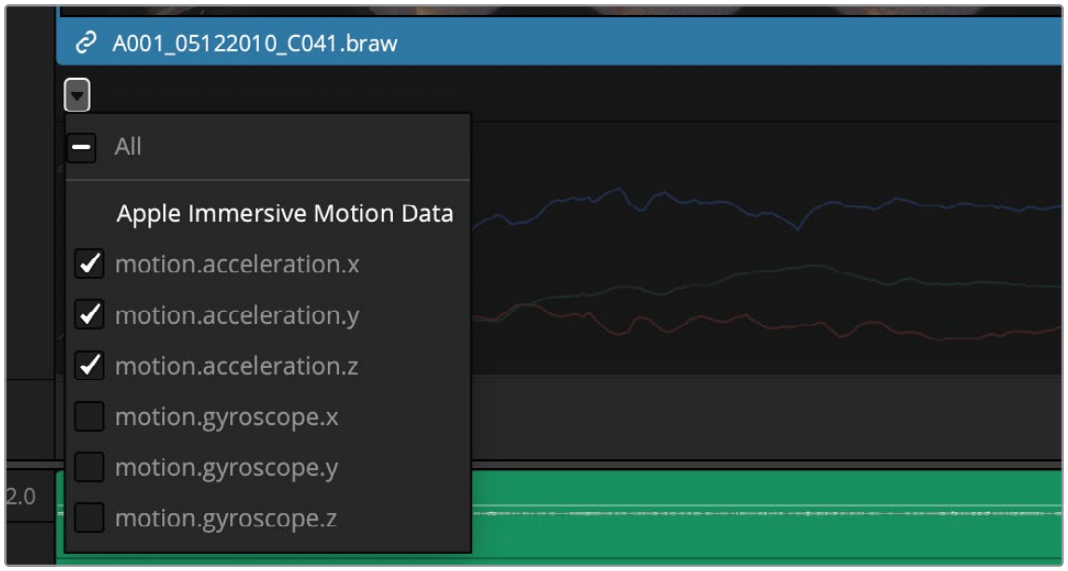


Das nun in der Videospur angezeigte Diagramm stellt die x-, y- und z-Beschleunigungsraten dar.



In diesem Beispiel hat die Kamera den Kameraoperator während ihrer Einrichtung aufgenommen. Die Spitzen im Diagramm zeigen plötzliche Bewegungen an, als die Kamera angehoben, gedreht und auf die Schultern des Operators gesetzt wurde. Idealerweise verwenden Sie den letzten Teil des Videos mit den niedrigeren Messwerten, nachdem die Kamera stabilisiert wurde.

3 Klicken Sie in der oberen linken Ecke des Diagramms auf den Verschiebungspfeil.



Sie können die Messung basierend auf dem Inhalt der Einstellung und der Kamerabewegungsart aktivieren oder deaktivieren.

Titel

Im Montage-Modul können Sie einer immersiven Szene Text hinzufügen, indem Sie eines der Titelwerkzeuge im „Effekte“-Feld verwenden. Bspw. „Text“, „Text+“, „MultiText“, „Scrollen“ und alle „Unteres Drittel“-Varianten.

Wird der Titelgenerator über einen immersiven Clip platziert, verwendet er die ILPD-Daten (Immersive Lens Profile Data) des Clips, um den Text genau der Linsenverzerrung der Szene entsprechend zu verzerren.

Sind keine ILPD verfügbar (z. B. beim Hinzufügen von Text auf einen schwarzen Backdrop oder in eine immersive CGI-Szene), wird die Standard-Linsenverzerrung angewendet, damit der Text bei der Anzeige in der Apple Vision Pro weiterhin korrekt verzerrt erscheint.

1 Navigieren Sie im „Effekte“-Feld zu „Toolbox“ > „Titel“.

2 Ziehen Sie den Text-Generator in die Timeline. Platzieren Sie ihn auf eine Videospur über einem immersiven Clip, um ihn in die Szene einzubinden.



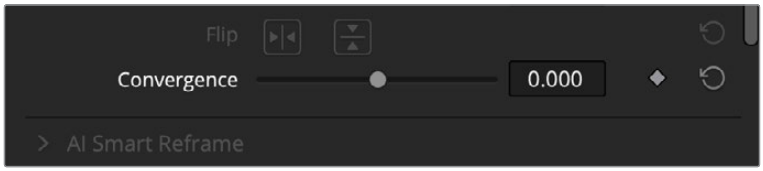
Genau wie mit Backdrops, dürfen Text-Generatoren die Clipgrenzen nicht überschreiten. Da jeder Clip mit einem individuellen Lens Space gerendert wird, besteht bei überlappenden Titeln das Risiko, dass sie zwischen Einstellungen verzerrt werden.

3 Trimmen Sie den Titel bis an den bzw. die Videoschnittpunkte.

4 Verwenden Sie zum Schreiben und Formatieren von Titeltext den Inspektor.

5 Gehen Sie zum „Einstellungen“-Tab und positionieren Sie den Text mit dem „Konvergenz“-Schieberegler im 3D-Raum.

Durch Bewegen des Schiebereglers nach links wird der Titel im Raum nach hinten verschoben, durch Bewegen nach rechts wird er nach vorne verschoben.



Am besten lässt sich die Konvergenz beim Monitoring auf der Apple Vision Pro beobachten. Wenn Sie andere Monitoring-Tools verwenden, können Sie die Darstellung der Textkonvergenz im Viewer ändern, indem Sie den Visionsmodus der zugrunde liegenden Videoebene im Farbe-Modul ändern.

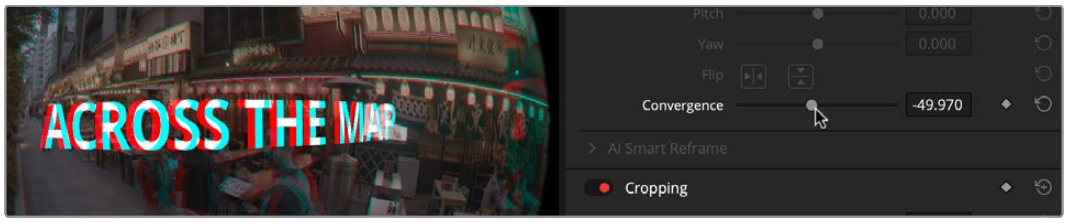
6 Gehen Sie zum Farbe-Modul.

7 Öffnen Sie die 3D-Palette.

Über die Spalte „Visionsmodus“ in der Mitte der Palette können Sie den stereoskopischen Standard ändern, der für die Projektion der immersiven Medien verwendet wird.

8 Wählen Sie den „Stereo“-Tab aus.

9 Stellen Sie für den „Out“-Modus die Option „Anaglyph (Farbe)“ ein.



Das Ändern des Konvergenzparameters des Textgenerators im Inspektor-Panel des Montage-Moduls erscheint nun intuitiver, wenn es durch eine Anaglyphenbrille (oder Anachrom-Brille) betrachtet wird.

Gestalterische Erwägungen

In dieser Anfangsphase der Produktion mit der Blackmagic URSA Cine Immersive Kamera und Apple Vision Pro werden Filmemachende ermutigt, mit dem Format zu experimentieren, um bewährte Verfahren zu identifizieren, neue Regeln aufzustellen und im traditionellen Filmemachen bisher undenkbare Arbeitsabläufe zu entwickeln.

Vor diesem Hintergrund wurden bereits einige Aspekte der immersiven Produktion zur besonderen Berücksichtigung vorgemerkt, um eine maximale Einbindung des Publikums und einen optimalen Sehkomfort zu gewährleisten.

1 Die Montage sollte bereits in der Vorproduktion beginnen. Aufgrund der besonderen Anforderungen an die Länge der Einstellungen und den bewegungsbezogenen Komfort der Zuschauenden sollten Filmemachende Szenen mit Blick auf den endgültigen Schnitt drehen.

2 Vermeiden oder begrenzen Sie deshalb die Verwendung von Clips mit schnellen Kamerabewegungen, da diese möglicherweise verwirrend auf die Zuschauenden wirken. Verwenden Sie stattdessen Einstellungen mit statischer Kamera oder langsame Schwenks und Kamerafahrten.

3 Unausgewogene oder extreme Kameraperspektiven können ebenfalls verwirrend sein. Filmen Sie Einstellungen lieber auf Augenhöhe mit dem Motiv und aus angenehmer Entfernung, also nicht zu nah.

4 Mit immersiven Medien dauert es länger, sich in neue Umgebungen einzufinden, als beim Betrachten einer 2D-Szene. Verwenden Sie deshalb langsamere Schnittfolgen, damit Zuschauende Zeit haben, sich an die neue Umgebung zu gewöhnen.

5 Achten Sie beim Schneiden zwischen Clips darauf, wohin die Zuschauenden in den vorherigen Einstellungen geblickt haben. Wichtige Handlungen sollten nicht auf der linken Seite des Bilds stattfinden, wenn die Aufmerksamkeit des Publikums zuvor auf die rechte Seite gerichtet war.

6 Da Spatial Audio maßgeblich zum Storytelling beiträgt, behandeln Sie es in immersiven Produktionen als eine weitere Figur.

VFX-Compositing

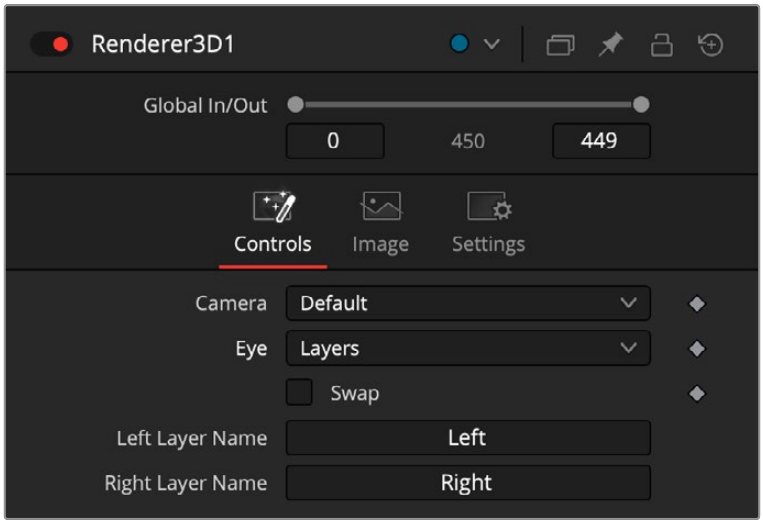
Wenn „Apple Immersive Workflow“ in den Projekteinstellungen aktiviert ist, ermöglichen spezielle Fusion-Parameter nahtlose stereoskopische Workflows und immersionskonformes Compositing. MediaOut-Nodes verfügen über Optionen für die Ausgabe für das linke und rechte Auge, und das Immersive Patcher-Tool ermöglicht eine gleichwinklige Projektion und Komposition.

MediaOut-Nodes

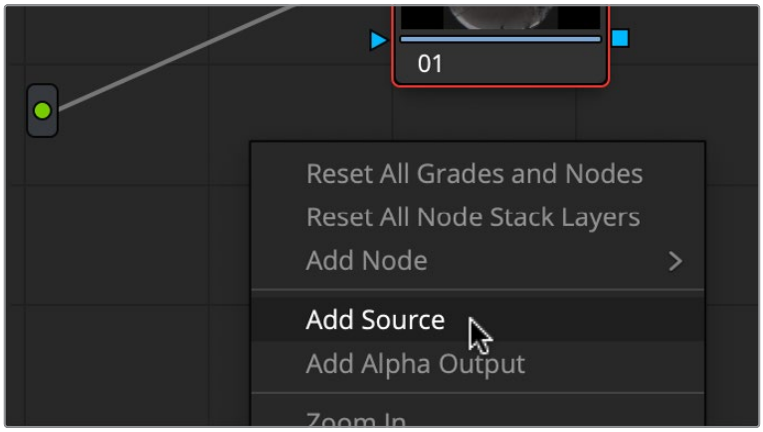
Wenn Sie beim Arbeiten an einem immersiven Clip erstmals das Fusion-Modul aufrufen, sehen Sie einen Node-Graphen mit zwei MediaIn-Nodes, die mit zwei dazugehörigen MediaOut-Nodes verbunden sind.



Im MediaOut-Inspektor dient der Parameter „Stereo-Auge“ zur Zuweisung des für dieses Auge ausgegebenen Signals.



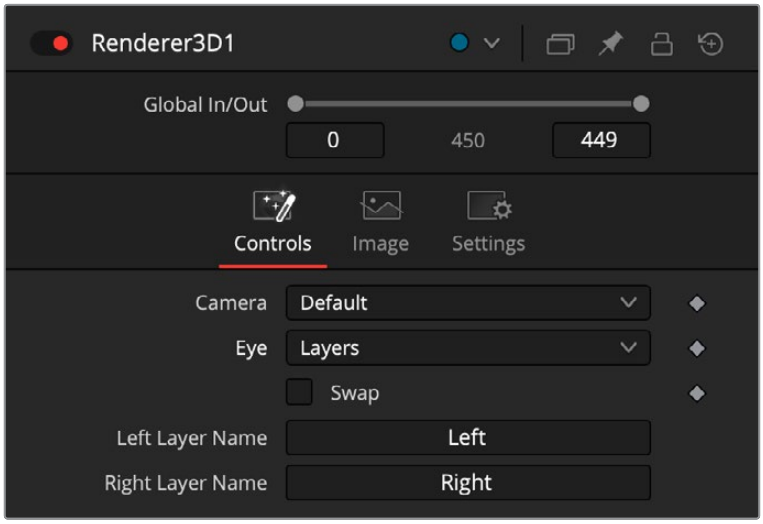
Standardmäßig ist MediaOut1 auf das linke Auge und MediaOut2 auf das rechte Auge eingestellt. Mit dieser Konfiguration können Sie sofort mit der Montage von stereokonformen Kompositionen und grafischen Overlays beginnen. Die dualen MediaOut-Nodes bewirken auch, dass für die weitere Bildverfeinerung zwei Signalvarianten an den Node-Editor im Farbe-Modul gesendet werden. Sie können aus dem Farbe-Modul auf das MediaOut2-Signal zugreifen, indem Sie per Rechtsklick im Node-Editor „Quelle hinzufügen” auswählen.



Ist der Parameter „Stereo-Auge“ auf „Auto“ eingestellt, kombiniert der MediaOut-Node die Signale und weist sie den entsprechenden Augen in der Timeline zu. Das lässt Sie den Mehrebenen-Workflow von Fusion nutzen, um die Signale für das linke und rechte Auge an einen einzelnen MediaOut-Node zu senden.

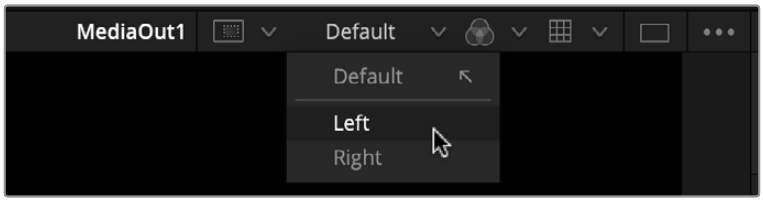
Renderer3D

Bei der Arbeit mit 3D-Grafiken oder CGI in einer immersiven Timeline bietet der Renderer3D-Node mehrere Augenkonfigurationen und die Möglichkeit, die linke und die rechte Ebene zu verarbeiten.



Dies ermöglicht es, komplexe 3D-Titel oder ganze CGI-Szenen zu erstellen und diese in immersiven Projekten direkt an eine stereoskopische Timeline zu senden.

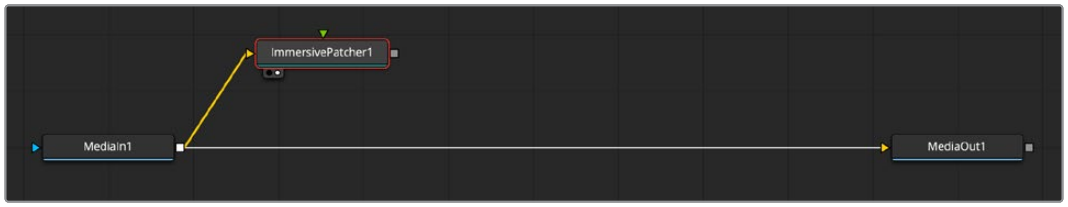
Verwenden Sie bei der Arbeit mit dieser Konfiguration das Dropdown-Menü oben rechts im Fusion-Viewer, um zwischen der Ansicht für das linke und das rechte Auge Ihrer Komposition zu wechseln.



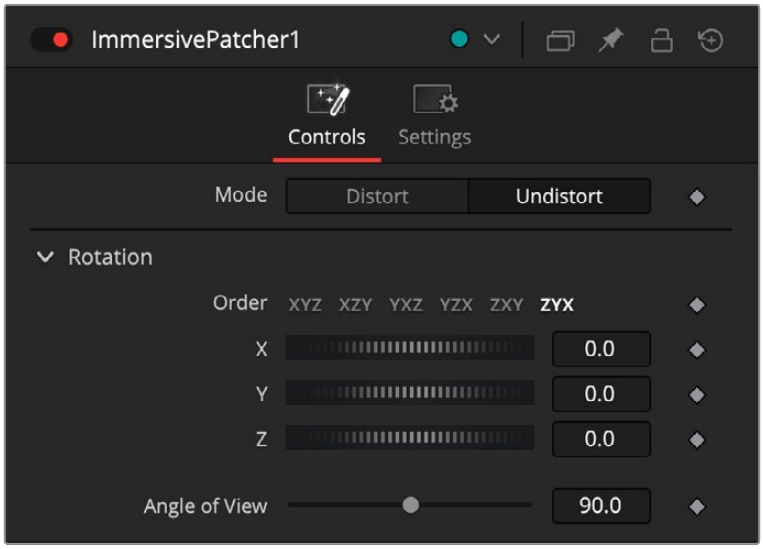
Compositing mit dem Immersive Patcher

Mit dem Immersive Patcher-Tool lässt sich ein Abschnitt eines immersiven Clips vorübergehend „entzerren“, um das Tracking, Painting und Compositing auf einer planen Ebene zu vereinfachen. Sobald Sie fertig sind, können Sie mit einem zweiten Immersive Patcher das immersive Signal erneut verzerren und die neu zusammengesetzten Elemente in die Szene integrieren.

1 Fügen Sie den Immersive Patcher zur Node-Pipeline hinzu.



2 Klicken Sie im Inspektor auf „Entzerren“.



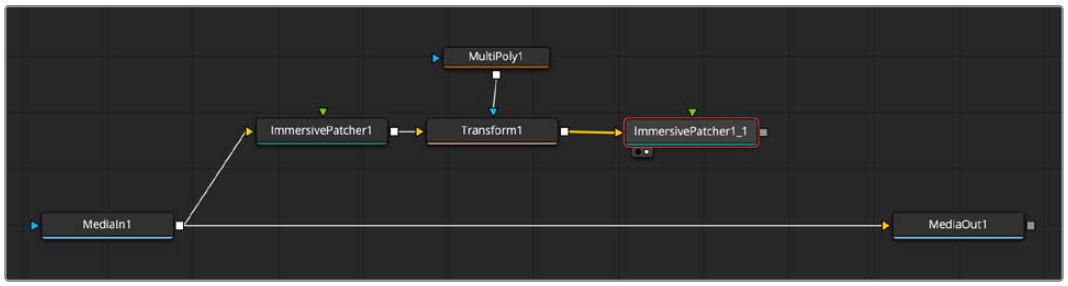
3 Verwenden Sie die Rotationsparameter X, Y und Z, um einen Abschnitt des immersiven Bilds auszuwählen, der abgeflacht werden soll.

4 Fahren Sie nun mit dem Compositing fort, wie Sie es mit einem normalen 2D-Videosignal tun würden. Dazu gehören (unter anderem) die Verwendung von Standard-Painting-Techniken zum Entfernen unerwünschter Objekte, das Sampling von Bildausschnitten mit Transformationen und Polygonen zur Set-Erweiterung oder das Anwenden von Matten, Text und externen Bild-/Videoebenen zum Ersetzen von Schildern oder Bildschirmen.

   Sobald Sie fertig sind, verzerren Sie das flache Bild erneut, damit es wieder der ursprünglichen immersiven Verzerrung entspricht.

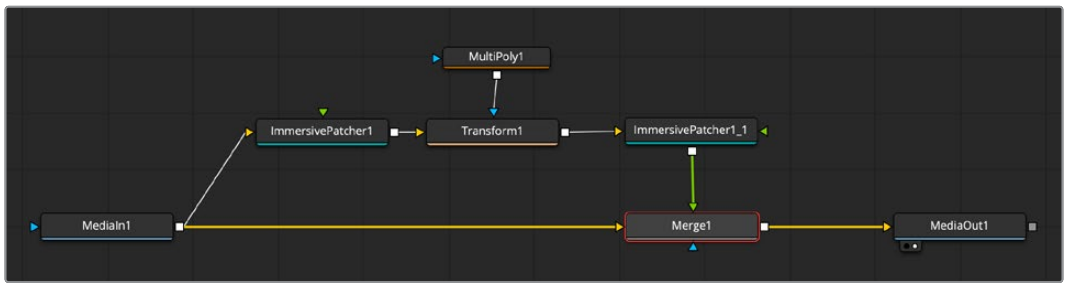
5 Kopieren Sie den Immersive Patcher, um die ursprünglichen Informationen zu Rotation und Ansichtswinkel beizubehalten.

6 Fügen Sie den Immersive Patcher hinter den Composite-Node ein und verbinden Sie ihn mit dem Ende der Compositing-Pipeline.



7 Klicken Sie im Inspektor auf „Verzerren“.

8 Führen Sie den zusammengesetzten Patch mit dem ursprünglichen Clip zusammen, um den Effekt in die immersive Szene zu integrieren.



Wenn Sie mit anderen zusammenarbeiten, können Sie das Fusion-Modul verlassen und die immersive Timeline in DaVinci Resolve weiter bearbeiten, mischen und farbkorrigieren.

Für immersive Turnovers, also die Übergabe bearbeiteter Dateien, sollten die einzelnen Augen als EXRs in ACES AP0 gerendert werden, um die maximalen Eigenschaften für die immersive Wiedergabe zu erhalten.

Colorgrading

Dieser Abschnitt behandelt die optischen, Farb- und 3D-Werkzeuge, die für Ihre Arbeit mit immersiven Medien im Farbe-Modul von DaVinci Resolve bereitstehen.

Aufgrund der besonderen Beschaffenheit immersiver Inhalte können nicht alle Tools in der Postproduktion verwendet werden. Alternative Tools sind derzeit nicht verfügbar, aber ihre Adaption für immersive Inhalte ist aktuell in Arbeit.

Rauschreduzierung

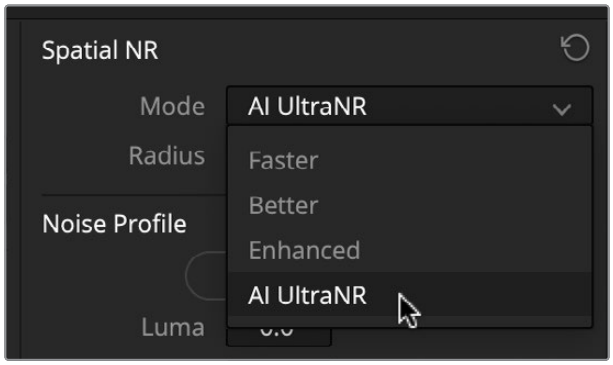
Die Rauschreduzierung kann bei der Arbeit mit immersiven Videos verwendet werden. Auf der Palette „Bewegungseffekte“ analysiert „Spatial NR“ das Rauschmuster der Quellmedien, um es maximal zu reduzieren und gleichzeitig die ursprünglichen Bilddetails beizubehalten.

1 Öffnen Sie in der Symbolleiste im Farbe-Modul die Palette „Bewegungseffekte“.

In der Spalte „Spatial NR“ können Sie den für die Rauschunterdrückung zu verwendenden Modus vorgeben.

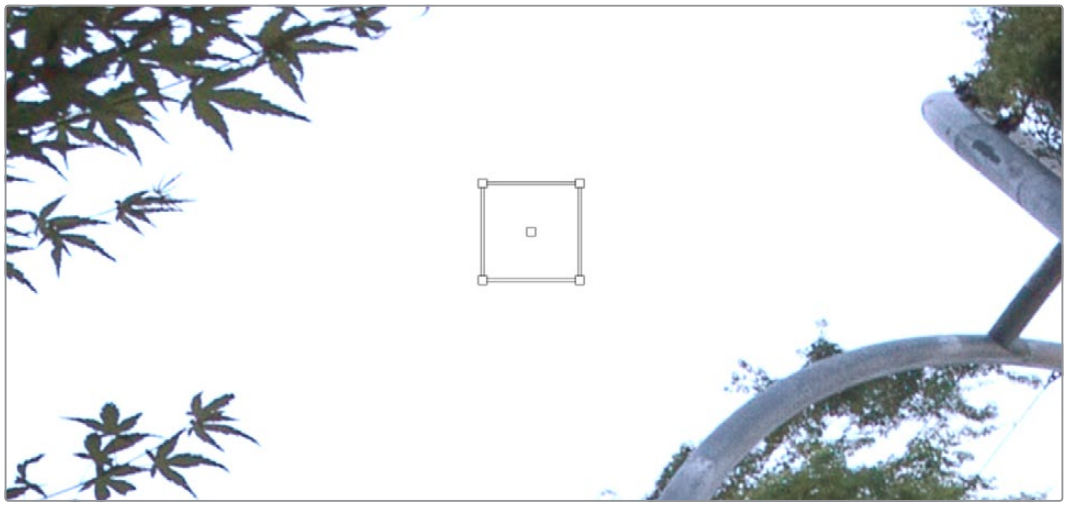
- Die Modi „*Schneller*“, „*Besser*“ und „*Verbessert*“ führen ihre Musteranalyse alle innerhalb eines bestimmten Radius durch und konfigurieren ihre Ausgabe auf der Grundlage des festgelegten Schwellenwerts. Sie sind in der Reihenfolge der Komplexität ihrer Verarbeitung aufgelistet.
- *AI UltraNR* ist ein KI-gestützter Identifikator für Rauschmuster. Er extrahiert das Rauschmuster aus dem Signal, indem er den „planarsten“ Abschnitt des Frames für reines Rauschen auffindet und analysiert.

2 Geben Sie als Modus zur räumlichen Rauschreduzierung „AI UltraNR“ vor.



3 Klicken Sie auf „Analysieren“.

Der nun im Viewer angezeigte Begrenzungsrahmen kennzeichnet den „leersten“ Bereich, sprich die geringste Pixeldatendichte im jeweiligen Frame. In Ermangelung erkennbarer visueller Inhalte sind die Daten innerhalb des Quadrats ideal zum Extrahieren des vom Kamerasensor erzeugten digitalen Rauschens.



Der Begrenzungsrahmen im Viewer lässt sich beliebig neu positionieren und in der Größe anpassen. In der Regel wird jedoch davon abgeraten, da der AI UltraNR optimale Analysebereiche besser erkennt als unsere Augen.

Alternativ können Sie die Einstellungen für das Rauschprofil „Luma“ und den „Chroma-Schwellenwert“ manuell anpassen. Allerdings gilt auch hier: Der AI UltraNR findet die ideale Kombination von Schwellenwerten für eine saubere Rauschreduzierung in der Regel automatisch.

Primäre Farbkorrektur

Die primäre Farbkorrektur funktioniert genauso wie bei normalem Videomaterial. Berücksichtigen Sie demnach dieselben Faktoren für die Normalisierung, den Aus- und Abgleich, wenn Sie die Primärfarben- und Log-Räder, die HDR-Palette und die HSL-Kurven verwenden. Die Paletten „ColorSlice“ und „Color Warper“ funktionieren ebenfalls wie erwartet.

Sekundäre Farbkorrektur

Auch pixelbasierte Auswahlmethoden, wie der Qualifizierer, arbeiten mit immersiven Medien ebenfalls wie erwartet.

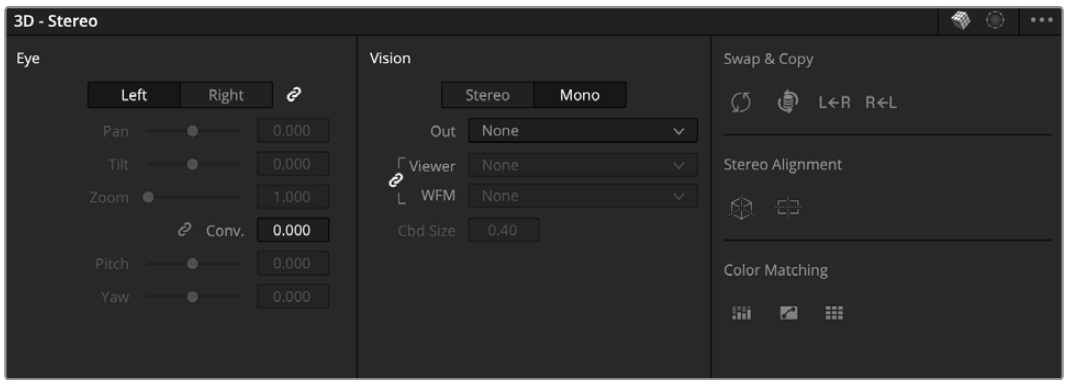
Die objekt- oder fensterbasierte Auswahl kann nur in einer planen, gleichwinkligen (oder Viewport-) Umgebung präzise funktionieren. Demnach sind die meisten sekundären Methoden, die eine präzise Form zur Qualifizierung erfordern, derzeit nicht verfügbar. Dazu gehören Power Windows (und damit auch die Tracker-Palette) und das „Magic Mask“-Werkzeug.

Skalierung

Für immersive Inhalte sind derzeit keine Bedienelemente zum Skalieren (bspw. Zoomen und Schwenken) verfügbar.

3D-Palette

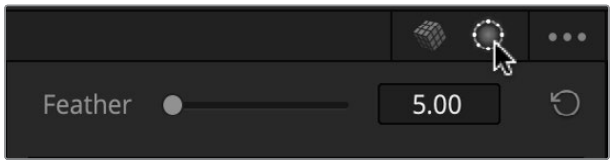
Die 3D-Palette im Farbe-Modul enthält Bedienelemente zum Verwalten einer Vielzahl von 3D-Kamera- und Grafikformaten. Im Gegensatz zu anderen stereoskopischen Formaten brauchen Sie die Augenausrichtung nicht anzupassen, da die Objektive, Brennweiten und Blendeneinstellungen der Blackmagic URSA Cine Immersive Kamera fixiert sind.



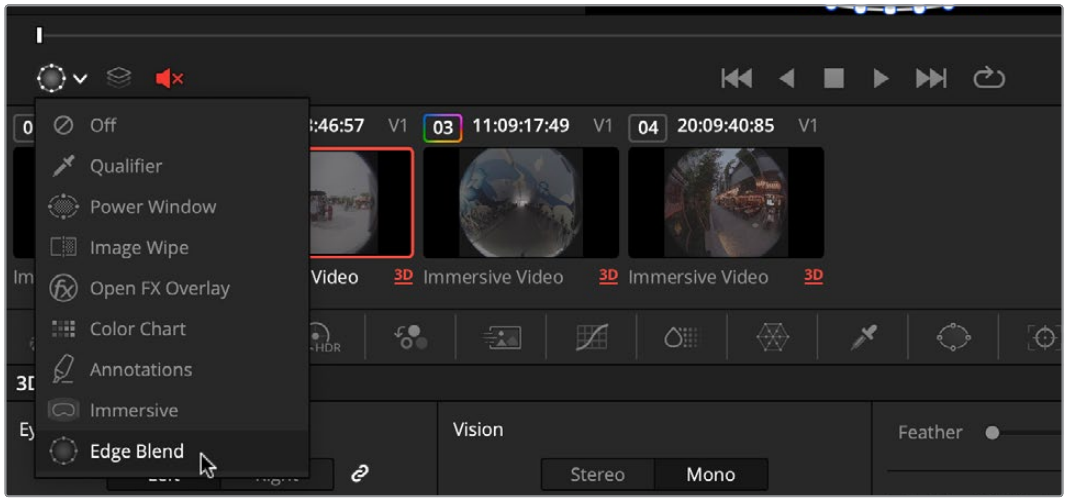
Der Modus „Kantenüberblendung“ ist ausschließlich für immersive Inhalte verfügbar und ermöglicht es, den Rand des immersiven Videobilds anzupassen und zu glätten. In diesem Modus lässt sich im Bild sichtbare Filmausrüstung verbergen und das Sichtfeld des Beobachtenden einschränken. Über diese Kante hinaus sehen Betrachtende den Backdrop.

1 Öffnen Sie die 3D-Palette.

2 Setzen Sie den Modus in der oberen rechten Ecke auf „Kantenüberblendung”.



3 Vergewissern Sie sich, dass die Onscreen-Bedienelemente des Betrachtenden auf „Kantenüberblendung“ eingestellt ist.



Nun sollte ein Kreis aus Steuerpunkten mit einem Fixiergriff in seiner Mitte, der das Videobild umgibt, zu sehen sein.

4 Ziehen Sie an einem Steuerpunkt, um ihn anders zu positionieren.



Alternativ löschen Sie einen Steuerpunkt durch Ausführen eines Rechtsklicks.

5 Verwenden Sie in der rechten Spalte der 3D-Palette den Parameter „Gradueller Verlauf", um die Kanten des Bilds fließender zu machen.

6 Wählen Sie den „visionOS-Effekt", damit die neuen Kantendaten bei der Wiedergabe auf der Apple Vision Pro korrekt verarbeitet werden. Bei Nichtauswahl dieser Option wird die Kante in die gerenderte Videodatei eingebrannt.

7 In der linken Spalte können Sie über die Tabs „Links" und „Rechts" im Viewer zwischen den Ansichten für die Augen wechseln.

Beachten Sie, dass viele Steuerelemente in der 3D-Palette ausgegraut sind. Parameter in DaVinci Resolve werden grau unterlegt, wenn sie für die aktuell bearbeiteten Medien nicht anwendbar sind. Das trifft hier auf die immersiven Inhalte zu.

8 In der mittleren Spalte können Sie über die Tabs „Stereo" und „Mono" zwischen Einzel- und Doppelobjektiv-Viewern für den immersiven Videoclip wechseln.

CST-Farbmanagement

Der im Farbe-Modul auf Nodes im Node-Editor angewendete Effekt „Color Space Transform“ (CST) ermöglicht eine genaue Farbzuweisung und eine benutzerdefinierte Steuerung der maximalen Ausgabe-Nit-Leuchtdichte in der Timeline.

Der empfohlene Arbeitsablauf umfasst vier CST-Nodes:

1. EINGABE-CST – Kamera zum Arbeitsfarbraum.

   Die Farbkorrektur erfolgt zwischen diesen beiden Nodes.

2. AUSGABE-CST – Arbeitsfarbraum zum HDR-Mac-Bildschirm.
3. MONITOR-CST – 1000 Nit zu 25 Nit fürs HDR-Monitoring.
4. AVP-CST – 250 Nit zu 108 Nit fürs Monitoring mit der und die Auslieferung an die Apple Vision Pro.



CST-Timeline-Mapping

Ordnen Sie den Farbraum der Kameraquelle dem in der Timeline verwendeten „Arbeitsfarbraum“ zu. Die Blackmagic URSA Cine Immersive Kamera arbeitet im „Blackmagic Design Wide Gamut Gen 4/5“-Farbraum mit der „DaVinci Intermediate“-Log-Kurve.

1. Öffnen Sie das „Effekte“-Feld.
2. Ziehen Sie „Color Space Transform“ auf einen leeren Node (01) im Node-Editor.

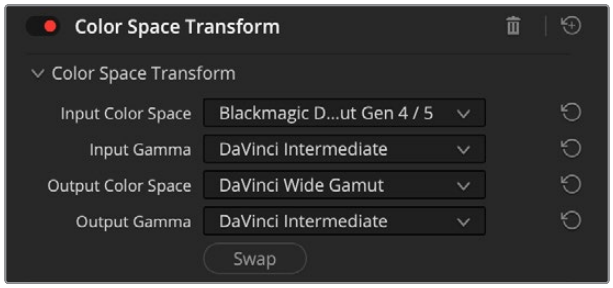
   Bei der Zuordnung vom Eingabefarbraum zum Timeline-Farbraum sollte dieser Node einer der ersten im Node-Baum sein, noch vor allen Grading-Vorgängen.

   Wenn Sie den Eingabefarbraum direkt dem Monitoring- oder Exportfarbraum zuordnen, sollte dieser Node zu den letzten im Node-Baum gehören, d. h. nach den Grading-Vorgängen.

3. Versehen Sie Node 01 mit der Kennung **EINGABE-CST**.

HINWEIS Die in dieser Anleitung verwendeten Node-Kennungen dienen der leichteren Bezugnahme auf die Dokumentation und Screenshots. Es steht Ihnen frei, Ihre bevorzugte Namenskonvention zu verwenden oder ganz auf die Beschriftung verzichten.

4. Stellen Sie als Eingabefarbraum „Blackmagic Design Wide Gamut Gen 4/5“ ein.
5. Stellen Sie den Eingabe-Gammawert auf „DaVinci Intermediate“ ein.
6. Stellen Sie den Ausgabefarbraum auf „DaVinci Wide Gamut“ ein.
7. Stellen Sie den Ausgabe-Gammawert auf „DaVinci Intermediate“ ein.



8 Erstellen Sie einen neuen seriellen Node (02).

9 Kopieren Sie den ersten Node (01) und fügen Sie ihn in den neuen Node (02) ein.

10 Versehen Sie Node 02 mit der Kennung **AUSGABE-CST**.

Hier sollten Farbraum und Gammawert der Eingabe für den neuen seriellen Node entsprechen, damit der Farbraum der Timeline des Node-Baums einheitlich bleibt.

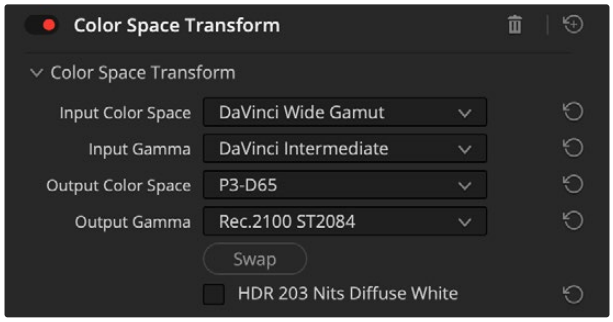
11 Klicken Sie auf den „Tauschen"-Button, um die Ein- und Ausgangsparameter zu wechseln.

Nun passen Eingabefarbraum und Gammawert zu diesem CST-Node.

12 Stellen Sie den Ausgabefarbraum auf P3-D65 ein. Dies ist der aktuelle Standardfarbraum von Mac-Bildschirmen.

13 Stellen Sie den Ausgabe-Gammawert auf Rec. 2100 ST2084 ein. Das ist der HDR-Standard für ein 1000-Nit-Signal.

Alle Grading-Vorgänge finden zwischen diesen beiden Nodes statt. Da Sie jedoch mit einem Node-basierten Farbmanagement arbeiten, haben Sie auch die einzigartige Möglichkeit, Vorgänge außerhalb des Timeline-Farbmanagements durchzuführen. Das nutzen viele Coloristinnen und Coloristen, um bestimmte Grading-Vorgänge, Effekte oder Elemente von Fremdanbietern anzuwenden.



ICC-Profile

Als weitere Versicherung, dass das Signal der Blackmagic URSA Cine Immersive auf Ihrem Mac-Bildschirm korrekt angezeigt wird, gehen Sie zu „Systemeinstellungen" > „Bildschirme". Überprüfen Sie dort, ob das Farbprofil „Voreinstellung" auf „Apple XDR Display (P3-1600 Nit)" eingestellt ist.

CST-Grading-Monitor-Mapping

Mit zusätzlichen CSTs können Sie die maximalen Nit-Ausgabewerte für das Monitoring während der Farbkorrektur vorübergehend begrenzen.

1 Erstellen Sie einen neuen seriellen Node (03) nach dem Farbmanagement-Node OUTPUT CST (02).

2 Kennzeichnen Sie Node 03 mit **MONITOR**.



3 Fügen Sie den Effekt „Color Space Transform“ zum MONITOR Node (03) hinzu.

Die Eingabewerte müssen mit dem Ausgabesignal des vorherigen CST übereinstimmen.

4 Stellen Sie den Eingabefarbraum auf P3-D65 ein.

5 Stellen Sie den Eingabe-Gammawert auf Rec. 2100 ST2084 ein.

Dieser CST-Node dient lediglich dazu, die Leuchtdichte des Bilds mit einem kleineren Nit-Wert abzubilden. Demnach können die Ein- und Ausgabeparameter unverändert bleiben.

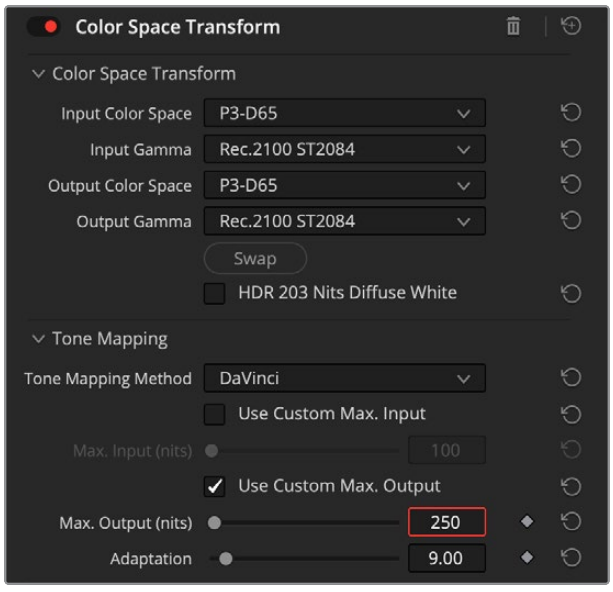
6 Stellen Sie den Ausgabefarbraum auf P3-D65 ein.

7 Stellen Sie den Ausgabe-Gammawert auf Rec. 2100 ST2084 ein.

HINWEIS Unter bestimmten Umständen ist es nicht erforderlich, den Farbraum oder den Gammawert eines CST festzulegen, wenn Eingabe und Ausgabe identisch sind (da keine Zuordnung erfolgt). In diesem Fall ist dies notwendig, da die Anwendung der benutzerdefinierten Nit-Zuordnung durch den Wert und die Form der Log-Kurve des Signals beeinflusst wird.

8 Wählen Sie „Eigenen Ausgabehöchstwert verwenden“.

9 Stellen Sie den Ausgabehöchstwert (Nit) auf **250** ein.



Das 1000-Nit-Signal wird nun (unter Verwendung der DaVinci-Tone-Mapping-Methode) mit 250 Nit abgebildet.

Dies ergibt ein Bild, das dem gerenderten Apple Vision Pro Signal auf einem HDR-Grading-Monitor (bspw. einem Flanders XMP310, Sony BVM-HX310 usw.) sehr nahe kommt.

Um Ihr Projekt jedoch auf einer Apple Vision Pro zu prüfen, müssen Sie einen weiteren Tone-Mapping-Vorgang durchführen.

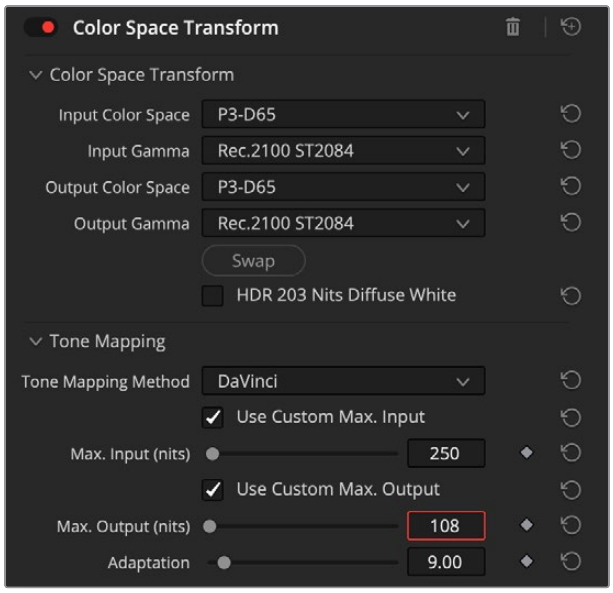
CST-Monitoring mit der Apple Vision Pro

Ein finaler CST-Node kann verwendet werden, um Live-Monitoring, Vorschau und die endgültige Auslieferung an Apple Vision Pro Geräte zu ermöglichen.

1. Erstellen Sie einen neuen seriellen Node (04) nach dem MONITOR-Node (03).
2. Kopieren Sie den MONITOR-Node (03) und setzen Sie ihn auf Node 04. Verwenden Sie dabei dieselben Ein- und Ausgabeparameter und ändern Sie lediglich die Anzahl der Nits.
3. Kennzeichnen Sie Node 04 mit **AVP**.



4. Wählen Sie „Eigenen Ausgabehöchstwert Eingabehöchstwert verwenden“.
5. Stellen Sie den Eingabehöchstwert (Nit) auf **250** ein, um das von dem MONITOR-Node zugewiesene Signal korrekt auszulesen.
6. Stellen Sie den Ausgabehöchstwert (Nit) auf **108** ein. Das ist der maximale Nit-Wert für das Monitoring und die Auslieferung an die Apple Vision Pro.



Nun können Sie mit dem Grading fortfahren und die CST-Nodes bei Bedarf (und wenn erforderlich) für Ihren Arbeitsablauf deaktivieren. Beispiel: Während Ihres ersten Grading-Durchgangs, bei dem Sie Ihren externen Grading-Monitor verwenden, sollten Sie den AVP-Node (04) deaktivieren, damit Ihnen das 250-Nit-Signal auf Ihrem HDR-Grading-Monitor korrekt angezeigt wird. Wenn Sie zur Vorschau oder zur endgültigen Auslieferung an die Apple Vision Pro wechseln, aktivieren Sie den AVP-Node (04) erneut.

Können die CSTs im Timeline-Node-Editor platziert werden?

Nein. Derzeit basiert die Tonwertzuordnung von Clips auf der genauen Anwendung des Parameters „Color Space Transform anpassen“, der für jeden Clip einzeln vorgenommen werden muss.

Audio

Die Apple Vision Pro und DaVinci Resolve unterstützen räumliches und Ambisonic Audio in den Formaten AVP-Audio (Apple Vision Pro) und AIA (Apple Immersive Audio).

Viele Aspekte der Audio-Postproduktion bleiben bei der Arbeit mit räumlichem Audio unverändert. Einer der wesentlichen Unterschiede besteht darin, dass Toningenieurinnen und Toningineure anstelle von Highend-Lautsprechern lieber Kopfhörer oder Airpods verwenden sollten, um ein echtes räumliches Eintauchen zu erzielen und die Hörumgebung des Publikums besser wiederzugeben.

Während bei 2D-Mischungen das Gros der Dialoge auf die mittleren Kanäle gemischt wird, muss räumliches Audio nach dem Ursprungsort des Tons gemischt werden. Audio-Stems müssen anhand von Ducking, Entzerrung und Balancing so gemischt werden, dass sie die Hörerschaft nicht überfordern oder verwirren.

Immersive räumliche Elemente lassen sich mit einem regulären Stereo-Stem kombinieren, sodass bestimmte Sounds unabhängig von der Kopfbewegung konstant bleiben. Dies ist hilfreich, um nicht-diegetische Sounds wie Musik und Voice-overs klar und leicht verständlich zu halten.

Umgebungsgeräusche werden in räumlichem Audio tendenziell leiser wahrgenommen, weshalb sie oft aggressiver gemischt werden, um bemerkt zu werden. Hier sollte man aber nicht übertreiben, da die Hörerschaft sonst womöglich verwirrt wird.

Der Hall trägt wesentlich zur Immersion bei, da er etwas ist, das Zuhörende jeden Tag ganz natürlich erleben. Jeder in einer bestimmten Umgebung erzeugte Sound wird von den Oberflächen, die den Zuhörenden umgeben, reflektiert, was zu einer „blühenden“ Hallqualität führt. Darum empfiehlt sich der Einsatz von Halleffekten in spatialem Audio zur Erzielung einer kompletten Immersion.

Das Format Apple Immersive Audio (AIA) ist ein Container für mehrere Formate:

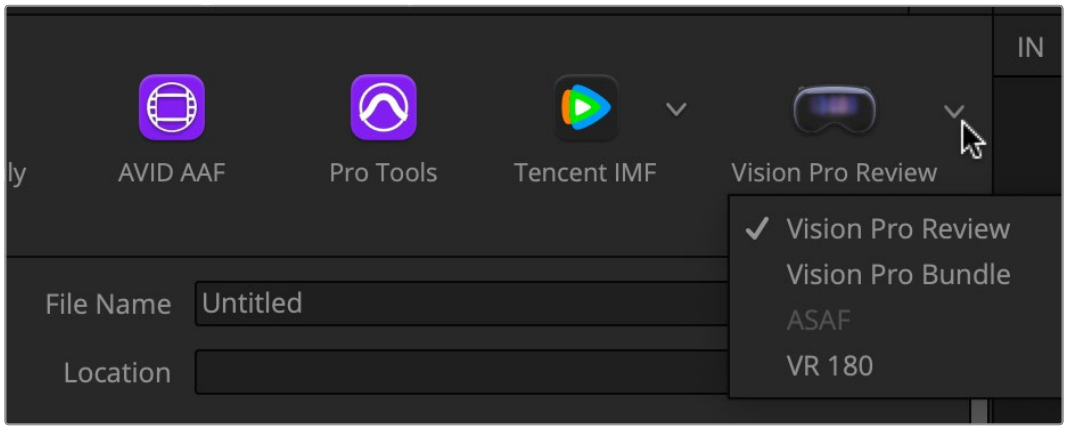
- **Apple Spatial Audio Format (ASAF)** ist das rohe Arbeitsformat für Audio.
- **Apple Positional Audio Codec (APAC)** ist das codierte MP4-Format, das auf APAC-unterstützten Geräten wie Airpods oder aus einer AIVU-Datei auf der Apple Vision Pro abgespielt werden kann.

Apple Spatial Audio Format (ASAF)

Das Apple Spatial Audio Format (ASAF) ist ein immersives Audioformat, das vorrangig für VR- und AR-Anwendungen in der Apple Vision Pro entwickelt wurde. ASAF ist exklusiv für macOS verfügbar und wird über ProTools (via Plugins) und DaVinci Resolve (integriert) unterstützt. ASAF verfügt über fortschrittliche technische Funktionen wie die integrierte Unterstützung für Ambisonics-Quellen (sogenannte „Szenen“) und Routing über Buszuweisungen. Das Mischen erfolgt virtuell ohne physischen Bus. Nutzende haben verschiedene Monitoring-Formate zur Wahl, u. a. binaural mit Head-Tracking über die direkte Integration von Apple AirPods. Die Panoramaverteilung erfolgt mit vorhandenen Tools von DaVinci Resolve wie dem „Pan“-Dialog, Automatisierung, SpaceView und den Panorama-Tools im Viewer. Der Inspektor hält sekundäre Steuerelemente bereit. Diese ermöglichen die präzise räumliche Platzierung von Quellen, einschließlich der automatischen Nachverfolgung anhand von Lat-Long-Video. ASAF-Mischungen können über die Option „Immersive Audio“ > „Masterdatei exportieren“ im WAV- oder MP4-Format (mit zugehörigen Metadaten) exportiert werden. Derzeit ist es jedoch nicht möglich, sie in DaVinci Resolve zu importieren oder wiederzugeben.

Auslieferung

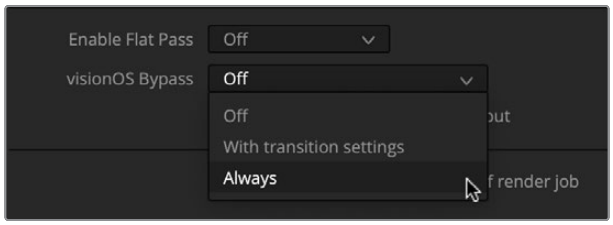
Im Export-Modul können Sie mit zwei Render-Einstellungen Apple Vision Pro Dateien zur Überprüfung und endgültigen Auslieferung rendern.



- **Vision Pro Überprüfung** – erstellt ein kleines MV-HEVC-Paket für Vorschauzwecke. Apple „teilt“ die resultierende Datei über WLAN mit der Apple Vision Pro und zeigt sie in der im App Store erhältlichen App „Apple Immersive Video Utility“ (AIVU) an.
- **Vision Pro Bundle** – erstellt ein großes ProRes-Dateibündel, das archiviert oder in Apple Compressor verarbeitet werden kann, um es für die Wiedergabe auf Apple TV vorzubereiten.
- **ASAF** – rendert einen ASAF-Audiomix im WAV- oder MP4-Format.
- **VR180** – generiert eine mit VR180 konforme stereoskopische Videodatei.

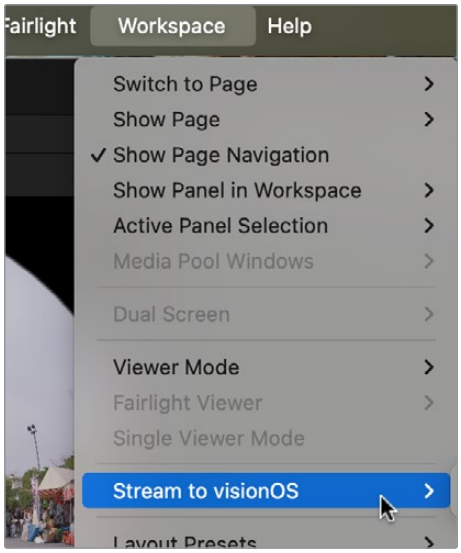
Umgehen von Übergängen

Wenn ein Übergang zu einer immersiven Timeline hinzugefügt wird, wird er vorgerendert und in eine einzelne Wiedergabespur umgewandelt. Die für immersive Inhalte generierte XML-Datei beschreibt den Prozess des Wechsels zwischen diesen gerenderten Streams (Filmmaterial, dann vorgerenderter Übergang, dann zurück zum Filmmaterial). In den erweiterten Einstellungen des Fensters mit den Render-Einstellungen verfügen Sie über die Optionen „visionOS-Übergänge umgehen“ („Immer“), sie entsprechend ihrem Status in der Timeline („Mit Übergangseinstellungen“) anzuwenden oder sie so einzufügen, wie sie in der Timeline erscheinen („Aus“).



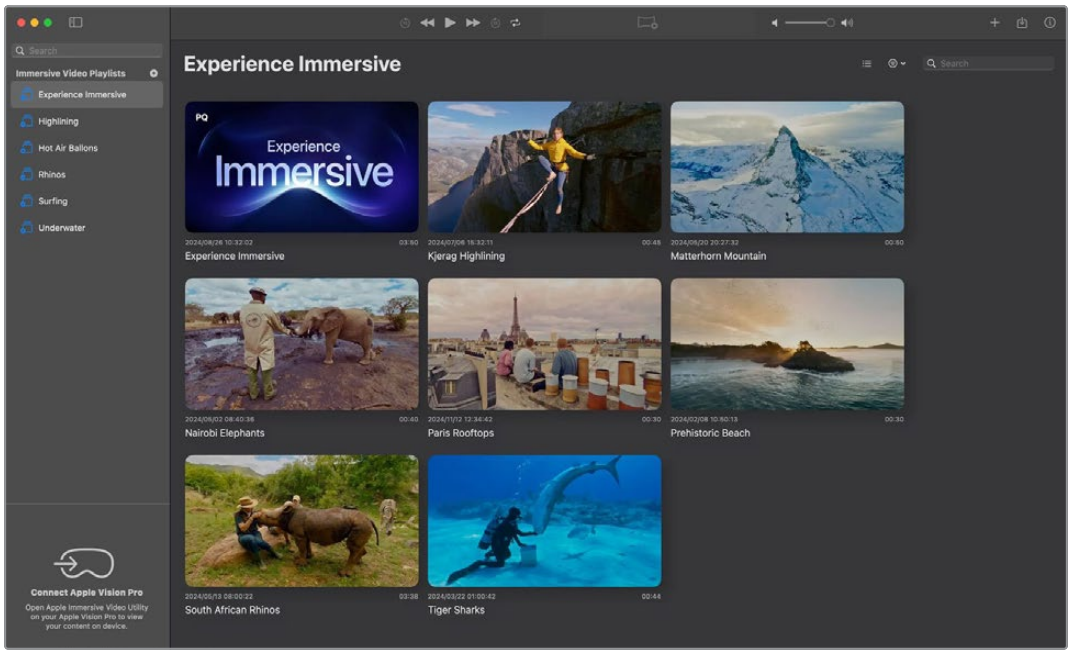
Streamen auf die Apple Vision Pro

Es ist möglich, aus den DaVinci Resolve Modulen „Montage“, „Farbe“ und „Export“ an die Apple Vision Pro zu streamen, indem Sie im Menü „Arbeitsbereich“ die Option „Auf visionOS streamen“ aktivieren und Ihr AVP-Gerät vorgeben. Zum flüssigen Streamen wird ein schnelles WLAN-Netzwerk (mindestens Wi-Fi 5, vorzugsweise Wi-Fi 6) empfohlen.



Infos zu Apple Immersive Video Utility

Apple Immersive Video Utility ist eine spezielle macOS- und visionOS-App, die entwickelt wurde, um die Erstellung, Verwaltung und Wiedergabe von Apple Immersive Video Inhalten zu optimieren. Das speziell für professionelle Filmemachende und Entwickler immersiver Medien entwickelte Dienstprogramm ermöglicht es Benutzern, stereoskopische 8K-Videodateien mit einem Sichtfeld von 180 Grad und räumlichem Audio zu importieren, zu organisieren, zu verpacken und zu überprüfen. Es unterstützt die Erstellung von AIVU-Dateien (Apple Immersive Video Universal) und ermöglicht die Überprüfung und Änderung von Metadaten für Postproduktions-Workflows. Damit können Nutzende Wiedergabelisten erstellen, ihre Bibliotheken sortieren und durchsuchen sowie sogar Inhalte direkt auf ein oder mehrere Apple Vision Pro Geräte streamen.



Informationen für Entwickler (englisch)

Camera Control REST API

If you are a software or hardware developer you can build custom applications or leverage ready to use tools such as REST client or Postman to seamlessly control and interact with your compatible Blackmagic camera using Camera Control REST API. This API enables you to perform a wide range of operations, such as starting or stopping recordings, accessing disk information and much more. Whether you're developing a custom application tailored to your specific needs or utilizing existing tools, this API empowers you to unlock the full potential of your Blackmagic camera with ease. We look forward to seeing what you come up with!

NOTE It's important to mention that controlling Blackmagic cameras via REST API relies on the web manager being enabled on each compatible Blackmagic camera. Enable the web media manager in the Blackmagic Camera Setup 'network access' settings for each camera you are controlling.

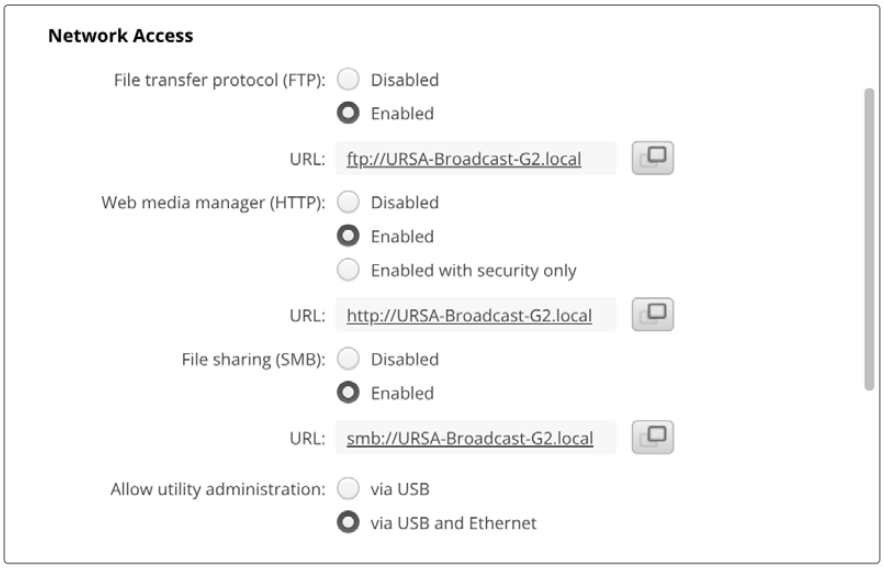
The following Blackmagic cameras are compatible with Camera Control REST API:

- Blackmagic URSA Cine 12K LF
- Blackmagic URSA Cine 17K 65
- Blackmagic URSA Cine Immersive
- Blackmagic PYXIS 6K
- Blackmagic Cinema Camera 6K
- Blackmagic URSA Broadcast G2
- Blackmagic Micro Studio Camera 4K G2
- Blackmagic Studio Camera 4K Plus
- Blackmagic Studio Camera 4K Pro
- Blackmagic Studio Camera 6K Pro
- Blackmagic Studio Camera 4K Plus G2
- Blackmagic Studio Camera 4K Pro G2

Sending API Commands

To send an API command to your camera from a third party application such as Postman, add /control/api/v1/ to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/api/v1/

You can find the Web media manager URL and IP address information in Blackmagic Camera Setup.



The Web media manager URL in Blackmagic Camera Setup

Downloading API's from your Camera

You can download REST API YAML documentation from your camera by adding /control/documentation.html to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/documentation.html

NOTE It's worth noting that changing the camera name in Blackmagic Camera Setup will also change the camera's Web media manager URL.

Livestream Control API

API for controlling Livestreams on Blackmagic Design products.

GET /livestreams/0

Get the livestream's current status.

Response

200 - Livestream's current status.

The response is JSON.

Name	Type	Description
status (required)	string	Possible values are: Idle, Connecting, Streaming, Flushing, Interrupted.
bitrate (required)	integer	Current bitrate (bps).
effectiveVideoFormat (required)	string	Effective video format for the livestream, serialised as a string.
duration	integer	Current stream duration in seconds. Absent if livestream is idle.
cache	integer	Current stream cache usage percentage.

GET /livestreams/0/start

Determine if the livestream is active.

Response

200 - Livestream active status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is active.

PUT /livestreams/0/start

Start the livestream.

Response

204 - Livestream started.

GET /livestreams/0/stop

Determine if the livestream is inactive.

Response

200 - Livestream inactive status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is inactive.

PUT /livestreams/0/stop

Stop the livestream.

Response

204 - Livestream stopped.

GET /livestreams/0/activePlatform

Get the currently selected platform configuration for the livestream.

Response

200 - Livestream active platform configuration.

The response is JSON.

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

PUT /livestreams/0/activePlatform

Set the currently selected platform configuration for the livestream.

Parameters

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

Response

204 - Livestream active platform configuration updated.

400 - Bad Request

GET /livestreams/platforms

Get the list of available platforms.

Response

200 - List of available platforms.

The response is JSON.

Name	Type	Description
	array	List of available platforms names.
[i]	string	Platform name.

GET /livestreams/platforms/{platformName}

Get the service configuration for a platform.

Parameters

Name	Type	Description
{platformName} (required)	string	Name of the platform.

Response

200 - Service configuration for specified platform.

The response is JSON.

Livestream platform service configuration.

Name	Type	Description
platform (required)	string	Corresponding platform name.
key	string	Default stream key.
servers (required)	array	List of server configurations.
servers[i]	object	Server configuration.
servers[i].server (required)	string	Server name.
servers[i].url (required)	string	Livestream destination.
servers[i].srtExtensions	array	Miscellaneous tags used for SRT livestreams.
servers[i].srtExtensions[i]	object	Dictionary object mapping SRT tag strings to values.
servers[i].srtExtensions[i][{key}]	string	SRT tag value.
servers[i].group	string	Logical grouping of the server.
profiles (required)	array	List of profile configurations.
profiles[i]	object	Quality configuration.
profiles[i].profile (required)	string	Quality level name.
profiles[i].configs (required)	array	List of video format configurations.
profiles[i].configs[i]	object	Video format configuration for profiles.
profiles[i].configs[i].resolution (required)	string	Video format serialised as a string.
profiles[i].configs[i].fps (required)	string	Frames per second.
profiles[i].configs[i].bitrate (required)	integer	Pixel bitrate (bps).
profiles[i].configs[i].audioBitrate	integer	Audio bitrate (bps).
profiles[i].configs[i].keyFrameInterval	integer	How often a key frame is sent, in seconds.
profiles[i].configs[i].videoCodecs	array	Supported video encoding algorithm/s.

Name	Type	Description
profiles[i].configs[i].videoCodecs[i]	string	Video encoding algorithm. Possible values are: H264, H265.
profiles[i].lowLatency (required)	boolean	If true, fewer frames will be buffered in the livestream.
defaultProfile	string	Quality level name.
credentials	object	Credientials used for RTMP streams.
credentials.username (required)	string	The username part of the creditials. Only used for RTMP streams.
credentials.password (required)	string	Used for RTMP streams, also used as Passphrase for SRT streams.
customizableUrlEnabled	boolean	True when the server URL is customizable.

400 - Bad Request

GET /livestreams/customPlatforms

Get a list of custom platform files.

Response

200 - List of custom platform files.

The response is JSON.

Name	Type	Description
	array	List of custom platform file names.
[i]	string	Custom platform file name.

DELETE /livestreams/customPlatforms

Remove all custom configuration files.

Response

204 - All custom configuration files removed.

GET /livestreams/customPlatforms/{filename}

Get a custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to get.

Response

200 - Custom platform file.

The response is XML.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

404 - Not Found

PUT /livestreams/customPlatforms/{filename}

Update a custom platform file if it exists, if not, create a new file with the given file name.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to update/create.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

Response

204 - Custom platform file created or updated.

400 - Bad Request

DELETE /livestreams/customPlatforms/{filename}

Remove the given custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to be removed.

Response

204 - Custom platform file removed.

404 - Not Found

Clips Control API

API for listing clips on disk.

GET /clips

Get the list of clips on the active disk.

Response

200 - List of clips on the active disk.

The response is JSON.

List of media clips.

Name	Type	Description
clips (required)	array	
clips[i]	object	Media clip.
clips[i].clipUniqueId (required)	integer	Unique ID used to identify this clip.
clips[i].filePath	string	Path to the file relative to the root of a mount.
clips[i].fileSize	integer	Size of file on disk in bytes.
clips[i].codecFormat	object	
clips[i].codecFormat.codec	string	Currently selected codec.

Name	Type	Description
clips[i].codecFormat.container	string	Multimedia container format.
clips[i].videoFormat	object	Video format configuration.
clips[i].videoFormat.name (required)	string	Video format serialised as a string.
clips[i].videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
clips[i].videoFormat.height	number	Height dimension of video format.
clips[i].videoFormat.width	number	Width dimension of video format.
clips[i].videoFormat.interlaced	boolean	Is the display format interlaced?
clips[i].startTimecode	string	Start timecode of the clip serialised as string.
clips[i].durationTimecode	string	Duration of the clip in timecode format serialised as string.
clips[i].frameCount	integer	Number of frames in clip; duration of the clip in frames.

404 - There is no active disk.

Media Pool Control API

API to manage media pool and handle uploads and project data.

GET /cloud/projects

List all projects within the media pool.

Response

200 - Successfully retrieved the list of all projects.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].libraryID	string	
[i].name	string	
[i].private	boolean	
[i].shared	boolean	
[i].clips	array	List of clips associated with the project.
[i].clips[i]	string	
[i].status	object	
[i].status.numClipsRequested	integer	
[i].status.numClipsComplete	integer	
[i].status.uploadPercent	integer	Percentage of upload completion.
[i].status.numClipsPaused	integer	
[i].status.outOfSpace	boolean	
[i].status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
[i].status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/active

Retrieve data of the actively uploading project.

Response

200 - Successfully retrieved the active project's data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/{projectID}

Retrieve specific project data by project ID.

Parameters

Name	Type	Description
{projectID} (required)	integer	Unique identifier of the project.

Response

200 - Successfully retrieved the project data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	

Name	Type	Description
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

404 - Project not found.

GET /cloud/clips

List all clips within the media pool.

Response

200 - Successfully retrieved the list of all clips.

The response is JSON.

Name	Type	Description
	array	
[i]	string	

GET /cloud/clips/activeUploading

Retrieve data of actively uploading clips.

Response

200 - Successfully retrieved the list of actively uploading clips.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].path	string	
[i].projectID	integer	
[i].status	object	
[i].status.projectID	integer	
[i].status.outOfSpace	boolean	
[i].status.proxyExtension	string	
[i].status.growingFile	boolean	
[i].status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.originalClipTotalSize	integer	
[i].status.proxyClipTotalSize	integer	
[i].status.originalClipCompletedSize	integer	
[i].status.proxyClipCompletedSize	integer	
[i].status.secsRemaining	integer	

GET /cloud/clips/{deviceName}/{path}

Retrieve specific clip data by device and path.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored.
{path} (required)	string	Path to the clip.

Response

200 - Successfully retrieved the clip data.

The response is JSON.

Name	Type	Description
path	string	
projectID	integer	
status	object	
status.projectID	integer	
status.outOfSpace	boolean	
status.proxyExtension	string	
status.growingFile	boolean	
status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.originalClipTotalSize	integer	
status.proxyClipTotalSize	integer	
status.originalClipCompletedSize	integer	
status.proxyClipCompletedSize	integer	
status.secsRemaining	integer	

404 - Clip not found.

Monitoring Control API

API for monitoring and controlling display settings in video equipment.

GET /monitoring/display

Retrieve a list of all display names.

Response

200 - Returns a list of display names.

The response is JSON.

Name	Type	Description
displays	array	List of display names available.
displays[i]	string	

GET /monitoring/{displayName}/cleanFeed

Get the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/cleanFeed

Set the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Clean feed enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/displayLUT

Get the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

400 - Invalid display name.

404 - Display name not found.

PUT /monitoring/{displayName}/displayLUT

Set the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Display LUT enabled/disabled successfully.

400 - Invalid input.

422 - Unprocessable Entity - Unable to process the contained instructions.

GET /monitoring/{displayName}/zebra

Get the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/zebra

Set the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Zebra enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/focusAssist

Get the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/focusAssist

Set the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/focusAssist

Get the focus assist settings.

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

404 - Display name not found.

PUT /monitoring/focusAssist

Set the focus assist settings.

Parameters

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/frameGuide

Get the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame guide enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGuide

Set the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame guide state updated successfully.

422 - Unable to update the frame guide state.

GET /monitoring/frameGuideRatio

Get the current frame guide ratio.

Response

200 - Returns the current frame guide ratio.

The response is JSON.

Name	Type	Description
ratio	string	The frame guide ratio.

PUT /monitoring/frameGuideRatio

Set the frame guide ratio.

Parameters

Name	Type	Description
ratio	string	The frame guide ratio.

Response

204 - Frame guide ratio updated successfully.

422 - Unable to update the frame guide ratio.

GET /monitoring/frameGuideRatio/presets

Get the presets for frame guide ratios.

Response

200 - Returns a list of preset frame guide ratios.

The response is JSON.

Name	Type	Description
presets	array	
presets[i]	string	A frame guide ratio.

GET /monitoring/{displayName}/frameGrids

Get the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame grids enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGrids

Set the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame grids state updated successfully.

422 - Unable to update the frame grids state.

GET /monitoring/frameGrids

Get the global frame grids settings.

Response

200 - Returns the current frame grids settings.

The response is JSON.

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

PUT /monitoring/frameGrids

Set the global frame grids settings.

Parameters

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

Response

204 - Frame grids settings updated successfully.

400 - Invalid input, check the number of frame grids or values.

422 - Unable to update the frame grids settings.

GET /monitoring/{displayName}/safeArea

Get the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the safe area enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/safeArea

Set the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Safe area state updated successfully.

422 - Unable to update the safe area state.

GET /monitoring/safeAreaPercent

Get the current safe area percentage.

Response

200 - Returns the current safe area percentage.

The response is JSON.

Name	Type	Description
percent	integer	Safe area coverage percentage.

PUT /monitoring/safeAreaPercent

Set the safe area percentage.

Parameters

Name	Type	Description
percent	integer	Safe area coverage percentage to set.

Response

204 - Safe area percentage updated successfully.

400 - Invalid percentage value.

422 - Unable to update the safe area percentage.

GET /monitoring/{displayName}/falseColor

Get the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the false color enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/falseColor

Set the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - False color state updated successfully.

422 - Unable to update the false color state.

Event Control API

API For working with built-in websocket.

GET /event/list

Get the list of events that can be subscribed to using the websocket API.

Response

200 - Websocket API events list.

The response is JSON.

Name	Type	Description
events	array	List of events that can be subscribed to using the websocket API.
events[i]	string	

System Control API

API for controlling the System Modes on Blackmagic Design products.

GET /system

Get device system information.

Response

200 - System summary.

The response is JSON.

The properties will be populated only with the values that are supported/implemented by the device in use.

Name	Type	Description
codecFormat	object	Codec format configuration.
codecFormat.codec	string	Codec serialised as string.
codecFormat.container	string	Multimedia container format.
videoFormat	object	Video format configuration.
videoFormat.name (required)	string	Video format serialised as a string.
videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
videoFormat.height	number	Height dimension of video format.
videoFormat.width	number	Width dimension of video format.
videoFormat.interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/product

Get device product information.

Response

200 - Device product information.

The response is JSON.

Product information.

Name	Type	Description
deviceName	string	Name of device as displayed in Setup.
productName	string	Device's product name.
softwareVersion	string	Software version running on device.

501 - This functionality is not implemented for the device in use.

GET /system/supportedCodecFormats

Get the list of supported codecs.

Response

200 - List of supported codec formats.

The response is JSON.

Name	Type	Description
codecs	array	
codecs[i]	object	Codec format configuration.
codecs[i].codec	string	Codec serialised as string.
codecs[i].container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

GET /system/codecFormat

Get the currently selected codec.

Response

200 - Current codec format.

The response is JSON.

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

PUT /system/codecFormat

Update the system codec.

Parameters

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

Response

204 - The codec updated successfully.

400 - The specified codec format is unsupported.

501 - This functionality is not implemented for the device in use.

GET /system/videoFormat

Get the currently selected video format.

Response

200 - Current system video format.

The response is JSON.

Video format configuration.

Name	Type	Description
name (required)	string	Video format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height	number	Height dimension of video format.
width	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

PUT /system/videoFormat

Set the system video format.

Parameters

This parameter can be one of the following types:

Name	Type	Description
name (required)	string	Video format serialised as a string.

Name	Type	Description
frameRate (required)	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height (required)	number	Height dimension of video format.
width (required)	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

Response

204 - The video format updated successfully.

400 - Invalid request.

409 - Operation unsupported in the current state.

501 - This functionality is not implemented for the device in use.

GET /system/supportedVideoFormats

Get the list of supported video formats for the current system state.

Response

200 - List of supported video formats.

The response is JSON.

List of supported video formats.

Name	Type	Description
formats	array	List of video formats.
formats[i]	object	Video format configuration.
formats[i].name (required)	string	Video format serialised as a string.
formats[i].frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
formats[i].height	number	Height dimension of video format.
formats[i].width	number	Width dimension of video format.
formats[i].interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/supportedFormats

Get supported formats.

Response

200 - List of supported formats.

The response is JSON.

Name	Type	Description
supportedFormats	array	
supportedFormats[i]	object	
supportedFormats[i].codecs	array	
supportedFormats[i].codecs[i]	string	
supportedFormats[i].frameRates	array	
supportedFormats[i].frameRates[i]	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
supportedFormats[i]. maxOffSpeedFrameRate	number	
supportedFormats[i]. minOffSpeedFrameRate	number	
supportedFormats[i]. recordResolution	object	
supportedFormats[i]. recordResolution.height	number	Height of the resolution.
supportedFormats[i]. recordResolution.width	number	Width of the resolution.
supportedFormats[i]. sensorResolution	object	
supportedFormats[i]. sensorResolution.height	number	Height of the resolution.
supportedFormats[i]. sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

GET /system/format

Get current format.

Response

200 - Current format.

The response is JSON.

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

PUT /system/format

Set the format.

Parameters

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

Response

204 - System format updated.

501 - This functionality is not implemented for the device in use.

Transport Control API

API for controlling Transport on Blackmagic Design products.

GET /transports/0

Get device's basic transport status.

Response

200 - Transport status.

The response is JSON.

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

PUT /transports/0

Set device's basic transport status.

Parameters

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, Output.

Response

204 - Transport mode was set.

400 - Failed to set transport mode.

GET /transports/0/stop

Determine if transport is stopped.

Response

200 - Transport stop response.

The response is JSON.

Name	Type	Description
	boolean	True when transport mode is InputPreview or when in Output mode and speed is 0.

PUT /transports/0/stop

Stop transport. Deprecated, use POST /transports/0/stop instead.

Response

204 - Transport stopped.

POST /transports/0/stop

Stop transport.

Response

204 - Transport stopped.

GET /transports/0/play

Determine if transport is playing.

Response

200 - Transport play response.

The response is JSON.

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

PUT /transports/0/play

Start playing on transport. Deprecated, use POST /transports/0/play instead.

Response

204 - Transport playing.

400 - Failed to set transport to play.

POST /transports/0/play

Start playing on transport.

Response

204 - Transport playing.

400 - Failed to set transport to play.

GET /transports/0/playback

Get playback state.

Response

200 - Transport playback state.

The response is JSON.

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

PUT /transports/0/playback

Set playback state.

Parameters

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

Response

204 - Updated transport playback state.

400 - Failed to set transport playback state.

GET /transports/0/record

Get record state.

Response

200 - Recording state.

The response is JSON.

Name	Type	Description
recording	boolean	If true, transport is in InputRecord mode.

PUT /transports/0/record

Set record state. Deprecated, use POST /transports/0/record instead.

Parameters

Name	Type	Description
recording (required)	boolean	If true, starts a recording, otherwise stops.
clipName	string	Optional, sets the requested clip name to record to, when “recording” attribute is set to true.

Response

204 - Recording state updated.

400 - Failed to update recording state.

POST /transports/0/record

Start recording.

Parameters

Name	Type	Description
clipName	string	Optional, provides a specific name of clip to record to.

Response

204 - Recording started.

400 - Failed to start recording.

GET /transports/0/clipIndex

Get the clip index of the currently playing clip on the timeline.

Response

200 - Clip index response.

The response is JSON.

Name	Type	Description
clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

GET /transports/0/timecode

Get device timecode.

Response

200 - Timecode response.

The response is JSON.

Name	Type	Description
display	string	The display timecode serialised as a string.
timeline	string	The timeline timecode serialised as a string.

GET /transports/0/timecode/source

Get timecode source selected on device.

Response

200 - Timecode source response.

The response is JSON.

Name	Type	Description
timecode	string	Possible values are: Timeline, Clip.

Timeline Control API

API for controlling playback timeline.

GET /timelines/0

Get the playback timeline.

Response

200 - Playback timeline.

The response is JSON.

Name	Type	Description
clips	array	
clips[i]	object	Timeline clip.
clips[i].clipUniqueId (required)	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).
clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

404 - No timeline / disk available.

DELETE /timelines/0

Clear the current playback timeline. Deprecated, prefer to use POST /timelines/0/clear

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

POST /timelines/0

Add a clip to the timeline.

Parameters

This parameter can be one of the following types:

Add multiple media clips to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips[i].clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips[i].clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips[i].frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add multiple media clips to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add a single clip to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips.clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips.clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips.frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add a single clip to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the timeline as specified.

501 - The operation is not supported on this device.

POST /timelines/0/add

Add a clip to the end of the timeline. Deprecated, use POST /timelines/0 to add clips within the timeline.

Parameters

This parameter can be one of the following types:

Add one clip to the end of the timeline.

Name	Type	Description
clips	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add many clips to the end of the timeline.

Name	Type	Description
clips	array	List of clipUniqueIds of clips to add to end of timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the end of the timeline.

501 - The operation is not supported on this device.

POST /timelines/0/clear

Clear the playback timeline.

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

DELETE /timelines/0/clips/{timelineClipIndex}

Remove the specified clip from the timeline.

Parameters

Name	Type	Description
{timelineClipIndex} (required)	integer	The (0-based) timeline clip index of the clip to remove from the timeline.

Response

204 - The specified clip was removed from the timeline.

501 - The operation is not supported on this device.

Media Control API

API for controlling media devices in Blackmagic Design products.

GET /media/workingset

Get the list of media devices currently in the working set.

Response

200 - The list of media devices in the working set.

The response is JSON.

Name	Type	Description
size	integer	The fixed size of this device's working set.
workingset	array	The device's working set.
workingset[i]	object | null	Device within the working set. null if no device is present within the given working set slot.
workingset[i].volume	string	Volume name.
workingset[i].deviceName	string	Internal device name of this media device.
workingset[i].remainingRecordTime	integer	Remaining record time using current codec and video format in seconds.
workingset[i].totalSpace	integer	Total space on media device in bytes.
workingset[i].remainingSpace	integer	Remaining space on media device in bytes.
workingset[i].clipCount	integer	Number of clips currently on the device.

GET /media/active

Get the currently active media device.

Response

200 - The current active media device.

The response is JSON.

The active media device, or null if there is no active media.

Name	Type	Description
workingsetIndex	integer	Working set index of the active media device.
deviceName	string	Device name of media device.

204 - No media is currently active.

PUT /media/active

Set the currently active media device.

Parameters

Name	Type	Description
workingsetIndex	integer	Working set index of the media to make active.

Response

204 - The active media device was set successfully.

400 - Setting the currently active media device is not possible in the current state.

GET /media/devices/doformatSupportedFilesystems

Get the list of filesystems available to format a media device.

Response

200 - The list of filesystems permitted for formatting.

The response is JSON.

Name	Type	Description
	array	List of filesystems permitted for formatting media.
[i]	string	Filesystem serialised as string.

GET /media/devices/{deviceName}

Get information about a requested device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Information about the requested device.

The response is JSON.

Media device state.

Name	Type	Description
state	string	The current state of the media device. Possible values are: None, Scanning, Mounted, Uninitialised, Formatting, RaidComponent.

400 - Invalid device name.

404 - Device not found.

GET /media/devices/{deviceName}/doformat

Get a format key, used to format the device with a PUT request.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Format prepared.

The response is JSON.

Name	Type	Description
deviceName	string	Device name of media device to format.
key	string	The key required to format this device, provide to PUT /media/devices/{deviceName}/doformat to perform format of media device.

400 - Cannot format the device.

404 - Device not found.

PUT /media/devices/{deviceName}/doformat

Perform a format of the specified media device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Name	Type	Description
key	string	The key used to format this device, retrieved from prepare format media request GET /media/devices/{deviceName}/doformat. Format key provided cannot be reused after successful format.
filesystem	string	Filesystem to format to. Supported filesystems can be retrieved with GET /media/devices/doFormatSupportedFilesystems.
volume	string	Volume name to set for the disk after format.

Response

204 - Format successful.

400 - Cannot format the device, invalid filesystem or key.

404 - Device not found.

Slate Control API

API to manage digital slate data.

GET /slates/nextClip

Retrieve the digital slate for the next clip.

Response

200 - Returns the slate data for the next clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	

Name	Type	Description
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Slate data is not available.

PUT /slates/nextClip

Update the slate data for the next clip.

Parameters

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

POST /slates/nextClip/resetProjectData

Reset the project data for the next clip's slate.

Response

200 - Project data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetProjectData

Reset the project data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Project data reset successfully.

POST /slates/nextClip/resetLensData

Reset the lens data for the next clip's slate.

Response

200 - Lens data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetLensData

Reset the lens data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Lens data reset successfully.

GET /slates/clips/{deviceName}/{path}

Retrieve slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Returns the slate data for the specified clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

404 - Clip not found.

PUT /slates/clips/{deviceName}/{path}

Update the slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.

Name	Type	Description
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

Preset Control API

API For controlling the presets on Blackmagic Design products

GET /presets

Get the list of the presets on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presets	array	List of the presets on the camera (.cset files)
presets[i]	string	

POST /presets

Send a preset file to the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presetAdded	string	Name of the preset uploaded (without .cset extension)

400 - Bad request - missing Content-Disposition header or filename

GET /presets/active

Get the currently active preset on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

PUT /presets/active

Set the active preset on the camera

Parameters

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

Response

204 - No Content

404 - Preset file not found

PARAMETERS /presets/{presetName}

GET /presets/{presetName}

Download the preset file

Response

200 - OK

The response is a binary file.

404 - File does not exist

PUT /presets/{presetName}

Save current camera state as a preset

Response

204 - No Content

DELETE /presets/{presetName}

Delete a preset from the camera

Response

204 - No Content

404 - Preset file not found

Audio Control API

API For controlling audio on Blackmagic Design Cameras

GET /audio/channels

Get the total number of audio channels available

Response

200 - Returns the total number of channels

The response is JSON.

Name	Type	Description
channels	integer	Total number of audio channels available

GET /audio/supportedInputs

Get the list of supported audio inputs

Response

200 - List of all supported audio inputs

The response is JSON.

Name	Type	Description
	array	
[i]	string	A supported audio input

GET /audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently selected input

The response is JSON.

Name	Type	Description
input	string	Audio input source and type

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/input

Set the audio input for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
input	string	Audio input source and type

Response

204 - No Content

400 - Invalid audio input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/input/description

Get the description of the current input of the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Description of the current input of the selected channel

The response is JSON.

Name	Type	Description
description	object	
description.gainRange	object	
description.gainRange.Min	number	The minimum gain value in dB
description.gainRange.Max	number	The maximum gain value in dB
description.capabilities	object	
description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter
description.capabilities.Padding	object	
description.capabilities.Padding. available	boolean	Input supports setting of padding
description.capabilities.Padding. forced	boolean	Padding is forced to be set for the input
description.capabilities.Padding. value	number	Value of the padding in dB

404 - Channel does not exist

GET /audio/channel/{channelIndex}/supportedInputs

Get the list of supported inputs and their availability to switch to for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its supported inputs are being queried. (Channels index from 0)

Response

200 - The list of supported inputs

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

404 - Channel does not exist

GET /audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set level for the selected channel

The response is JSON.

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/level

Set the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

Response

204 - No Content

400 - Invalid input or value out of range

404 - Channel does not exist

GET /audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set phantom power for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Phantom power enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/phantomPower

Set the audio phantom power for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Phantom power enabled state

Response

204 - No Content

400 - Phantom power is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set padding for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Padding enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/padding

Set the audio input padding for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Padding enabled state

Response

204 - No Content

400 - Padding is not supported or is forced for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set low cut filter for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Low cut filter enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/lowCutFilter

Set the audio input low cut filter for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Low cut filter enabled state

Response

204 - No Content

400 - Low cut filter is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set availability for the selected channel

The response is JSON.

Name	Type	Description
available	boolean	Whether the input is currently available

404 - Channel does not exist

Lens Control API

API For controlling the lens on Blackmagic Design products

GET /lens/iris

Get lens' aperture

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number

PUT /lens/iris

Set lens' aperture

Parameters

Name	Type	Description
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number
adjustmentStep	integer	Signed value for relative aperture adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens iris is not controllable or is controlled by auto exposure

GET /lens/zoom

Get lens' zoom

Response

200 - OK

The response is JSON.

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value

PUT /lens/zoom

Set lens' zoom

Parameters

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value
adjustmentFocalLength	integer	Signed value for relative focal length adjustment
adjustmentNormalised	number	Signed normalized value for relative zoom adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens zoom is not controllable

GET /lens/focus

Get lens' focus

Response

200 - OK

The response is JSON.

Name	Type	Description
normalised	number	Normalised value

PUT /lens/focus

Set lens' focus

Parameters

Name	Type	Description
normalised	number	Normalised value
focusDistance	integer	Focus distance value

Response

204 - No Content

400 - Bad Request if out of range value is provided

PUT /lens/focus/doAutoFocus

Perform auto focus

Parameters

Name	Type	Description
position	object	
position.x	number	Normalized x coordinate for autofocus ROI
position.y	number	Normalized y coordinate for autofocus ROI

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens focus is not controllable

GET /lens/opticalImageStabilization

Get optical image stabilization status

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Whether optical image stabilization is enabled

501 - Not Implemented if optical image stabilization is not supported on this product

PUT /lens/opticalImageStabilization

Enable or disable optical image stabilization

Parameters

Name	Type	Description
enabled	boolean	Enable or disable optical image stabilization

Response

204 - No Content

501 - Not Implemented if optical image stabilization is not supported on this product

GET /lens/iris/description

Get detailed description of lens' iris capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the iris can be controlled
apertureStop	object	
apertureStop.min	number	Minimum aperture stop
apertureStop.max	number	Maximum aperture stop

GET /lens/zoom/description

Get detailed description of lens' zoom capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the zoom can be controlled
focalLength	object	
focalLength.adjustable	boolean	If focal length is adjustable
focalLength.min	integer	Minimum focal length
focalLength.max	integer	Maximum focal length

GET /lens/focus/description

Get detailed description of lens' focus capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the focus can be controlled
focusDistance	object	
focusDistance.adjustable	boolean	If focus distance is adjustable
focusDistance.min	number	Minimum focus distance
focusDistance.max	number	Maximum focus distance

Video Control API

API For controlling the video on Blackmagic Design products

GET /video/iso

Get current ISO

Response

200 - OK

The response is JSON.

Name	Type	Description
iso	integer	Current ISO value

PUT /video/iso

Set current ISO

Parameters

Name	Type	Description
iso	integer	ISO value to set

Response

204 - No Content

403 - ISO cannot be changed in the current state

GET /video/supportedISOs

Get the list of supported ISO settings

Response

200 - List of supported ISO values

The response is JSON.

Name	Type	Description
supportedISOs	array	Array of supported ISO values
supportedISOs[i]	integer	

GET /video/gain

Get current gain value in decibels

Response

200 - OK

The response is JSON.

Name	Type	Description
gain	integer	Current gain value in decibels

PUT /video/gain

Set current gain value

Parameters

Name	Type	Description
gain	integer	Gain value in decibels to set

Response

204 - No Content

403 - Gain cannot be changed in the current state

GET /video/supportedGains

Get the list of supported gain settings in decibels

Response

200 - List of supported gain values in decibels

The response is JSON.

Name	Type	Description
supportedGains	array	Array of supported gain values in decibels
supportedGains[i]	integer	

GET /video/whiteBalance

Get current white balance

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	integer	Current white balance

PUT /video/whiteBalance

Set current white balance

Parameters

Name	Type	Description
whiteBalance	integer	White balance to set

Response

204 - No Content

400 - Invalid white balance temperature

GET /video/whiteBalance/description

Get white balance range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	object	
whiteBalance.min	integer	Minimum color temperature
whiteBalance.max	integer	Maximum color temperature

PUT /video/whiteBalance/doAuto

Set current white balance automatically

Response

204 - No Content

GET /video/whiteBalanceTint

Get white balance tint

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	integer	Current white balance tint

PUT /video/whiteBalanceTint

Set white balance tint

Parameters

Name	Type	Description
whiteBalanceTint	integer	White balance tint to set

Response

204 - No Content

400 - Invalid white balance tint

GET /video/whiteBalanceTint/description

Get white balance tint range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	object	
whiteBalanceTint.min	integer	Minimum white balance tint
whiteBalanceTint.max	integer	Maximum white balance tint

GET /video/ndFilter

Get ND filter stop

Response

200 - OK

The response is JSON.

Name	Type	Description
stop	number	Current filter power (fStop)

501 - Not implemented for this device

PUT /video/ndFilter

Set ND filter stop

Parameters

Name	Type	Description
stop	number	Filter power (fStop) to set

Response

204 - No Content

400 - Invalid ND filter stop

501 - Not implemented for this device

GET /video/supportedNDFilters

Get the list of available ND filter stops

Response

200 - List of available ND filter stops

The response is JSON.

Name	Type	Description
supportedStops	array	Array of available ND filter stops
supportedStops[i]	number	

501 - Not implemented for this device

GET /video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

Response

200 - List of supported display modes

The response is JSON.

Name	Type	Description
supportedDisplayModes	array	Array of supported display modes
supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

GET /video/ndFilter/displayMode

Get ND filter display mode on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

PUT /video/ndFilter/displayMode

Set ND filter display mode on the camera

Parameters

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

Response

204 - No Content

400 - Invalid display mode for ND filter

501 - Not implemented for this device

GET /video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

Response

200 - Indicates if ND filter is selectable

The response is JSON.

Name	Type	Description
selectable	boolean	True if ND filter adjustments are selectable via a slider

501 - Not implemented for this device

GET /video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

PUT /video/shutter

Set current shutter

Parameters

Name	Type	Description
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

Response

204 - No Content

GET /video/shutter/measurement

Get the current shutter measurement mode

Response

200 - OK

The response is JSON.

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

PUT /video/shutter/measurement

Set the shutter measurement mode

Parameters

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

Response

204 - No Content

400 - Invalid shutter measurement

GET /video/supportedShutters

Get supported shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of supported shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of supported shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of flicker-free shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of flicker-free shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/autoExposure

Get current auto exposure mode

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

PUT /video/autoExposure

Set auto exposure

Parameters

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

Response

204 - No Content

400 - Failed to set auto exposure mode

GET /video/detailSharpening

Get the current state of detail sharpening

Response

200 - Current detail sharpening state

The response is JSON.

Name	Type	Description
enabled	boolean	Whether detail sharpening is enabled

501 - Not implemented for this device

PUT /video/detailSharpening

Enable or disable detail sharpening

Parameters

Name	Type	Description
enabled	boolean	Enable or disable detail sharpening

Response

204 - Detail sharpening state updated

501 - Not implemented for this device

GET /video/detailSharpeningLevel

Get the current detail sharpening level

Response

200 - Current detail sharpening level

The response is JSON.

Name	Type	Description
level	string	Current detail sharpening level Possible values are: Low, Medium, High.

501 - Not implemented for this device

PUT /video/detailSharpeningLevel

Set the detail sharpening level

Parameters

Name	Type	Description
level	string	Desired level of detail sharpening Possible values are: Low, Medium, High.

Response

204 - Detail sharpening level updated

400 - Invalid detail sharpening level

501 - Not implemented for this device

Camera Control API

API For controlling the Camera specific features on Blackmagic Design products

GET /camera/colorBars

Get the status of color bars display

Response

200 - Returns the current status of color bars

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the color bars are currently enabled

PUT /camera/colorBars

Set the status of color bars display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the color bars

Response

204 - Color bars status updated successfully

GET /camera/programFeedDisplay

Get the status of program feed display

Response

200 - Returns the current status of program feed display

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the program feed display is currently enabled

PUT /camera/programFeedDisplay

Set the status of program feed display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the program feed display

Response

204 - Program feed display status updated successfully

GET /camera/tallyStatus

Get the tally status of the camera

Response

200 - Returns the current tally status of the camera

The response is JSON.

Name	Type	Description
status	string	Current tally status of the camera Possible values are: None, Preview, Program.

GET /camera/power

Get the power status of the camera

Response

200 - Returns the current power status

The response is JSON.

Name	Type	Description
source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
batteries	array	List of batteries currently connected to the camera
batteries[i]	object	
batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
batteries[i].chargeRemainingPercent	integer	Remaining battery charge percentage
batteries[i].statusFlags	array	List of battery status flags
batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

GET /camera/power/displayMode

Get the power display mode of the camera

Response

200 - Returns the current power display mode

The response is JSON.

Name	Type	Description
mode	string	Current power display mode Possible values are: Percentage, Voltage.

PUT /camera/power/displayMode

Set the power display mode of the camera

Parameters

Name	Type	Description
mode	string	Power display mode to set Possible values are: Percentage, Voltage.

Response

204 - Power display mode updated successfully

400 - Invalid power display mode

GET /camera/timingReferenceLock

Get the timing reference lock status

Response

200 - Returns the timing reference lock status

The response is JSON.

Name	Type	Description
locked	boolean	Indicates if timing reference is locked

Immersive Control API

API for controlling immersive camera settings on Blackmagic Design cameras

GET /immersive/display/{displayName}/eye

Get the current eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to query (from /monitoring/display endpoint)

Response

200 - OK

The response is JSON.

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

400 - Invalid display name format

404 - Display not found

422 - Failed to get eye view

PUT /immersive/display/{displayName}/eye

Set the eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to control (from /monitoring/display endpoint)

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

Response

204 - No Content

400 - Invalid input or display name format

404 - Display not found

422 - Failed to set eye view

Color Correction Control API

API For controlling the color correction on Blackmagic Design products based on DaVinci Resolve Color Corrector

GET /colorCorrection/lift

Get color correction lift

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

PUT /colorCorrection/lift

Set color correction lift

Parameters

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gamma

Get color correction gamma

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

PUT /colorCorrection/gamma

Set color correction gamma

Parameters

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gain

Get color correction gain

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

PUT /colorCorrection/gain

Set color correction gain

Parameters

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/offset

Get color correction offset

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

PUT /colorCorrection/offset

Set color correction offset

Parameters

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/contrast

Get color correction contrast

Response

200 - OK

The response is JSON.

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/contrast

Set color correction contrast

Parameters

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/color

Get color correction color properties

Response

200 - OK

The response is JSON.

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/color

Set color correction color properties

Parameters

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/lumaContribution

Get color correction luma contribution

Response

200 - OK

The response is JSON.

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

PUT /colorCorrection/lumaContribution

Set color correction luma contribution

Parameters

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

Response

204 - No Content

Notification websocket - 1.0.0

Service that notifies subscribers of device state changes.

messages

Subscribe (The messages from the server/device)

Websocket Opened Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: websocketOpened.
.type	string	Possible values are: event.

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*

Name	Type	Description
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Event Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: propertyValueChanged.

Name	Type	Description
.data.property	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.value	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.type	string	Possible values are: event.

Publish (The messages that user can send to the server/device)

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	

Name	Type	Description
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	

Name	Type	Description
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Device Properties

/media/workingset

The value JSON returned via the eventResponse when the `/media/workingset` property changes on the device:

Name	Type	Description
.size	integer	The fixed size of this device's working set.
.workingset	array	Array of devices within the working set. null if no device is present within the given working set slot.
.workingset[i]		

/media/active

The value JSON returned via the eventResponse when the `/media/active` property changes on the device:

Name	Type	Description
.workingsetIndex	integer	Working set index of the active media device.
.deviceName	string	Internal device name of this media device.

/system

The value JSON returned via the eventResponse when the `/system` property changes on the device:

Name	Type	Description
.codecFormat	object	Codec format configuration.
.codecFormat.codec	string	Codec format serialised as a string.
.codecFormat.container	string	Multimedia container format.
.videoFormat	object	Currently selected video format.
.videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.videoFormat.height	number	Height dimension of video format.
.videoFormat.width	number	Width dimension of video format.
.videoFormat.interlaced	boolean	Is the display format interlaced?.
.videoFormat.name	string	Video format serialised as a string.

/system/codecFormat

Codec format configuration.

The value JSON returned via the eventResponse when the `/system/codecFormat` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.container	string	Multimedia container format.

/system/videoFormat

Currently selected video format.

The value JSON returned via the eventResponse when the `/system/videoFormat` property changes on the device:

Name	Type	Description
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.height	number	Height dimension of video format.
.width	number	Width dimension of video format.
.interlaced	boolean	Is the display format interlaced?.
.name	string	Video format serialised as a string.

/system/format

The value JSON returned via the eventResponse when the `/system/format` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.maxOffSpeedFrameRate	number	
.minOffSpeedFrameRate	number	
.offSpeedEnabled	boolean	
.offSpeedFrameRate	number	
.recordResolution	object	
.recordResolution.height	number	Height of the resolution.
.recordResolution.width	number	Width of the resolution.
.sensorResolution	object	
.sensorResolution.height	number	Height of the resolution.
.sensorResolution.width	number	Width of the resolution.

/system/supportedFormats

The value JSON returned via the eventResponse when the `/system/supportedFormats` property changes on the device:

Name	Type	Description
.supportedFormats	array	
.supportedFormats[i]	object	
.supportedFormats[i].codecs	array	
.supportedFormats[i].codecs[i]	string	
.supportedFormats[i].frameRates	array	
.supportedFormats[i].frameRates[i]	string	Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.supportedFormats[i].maxOffSpeedFrameRate	number	
.supportedFormats[i].minOffSpeedFrameRate	number	
.supportedFormats[i].recordResolution	object	
.supportedFormats[i].recordResolution.height	number	Height of the resolution.
.supportedFormats[i].recordResolution.width	number	Width of the resolution.
.supportedFormats[i].sensorResolution	object	
.supportedFormats[i].sensorResolution.height	number	Height of the resolution.
.supportedFormats[i].sensorResolution.width	number	Width of the resolution.

/timelines/0

The value JSON returned via the eventResponse when the `/timelines/0` property changes on the device:

Name	Type	Description
.clips	array	
.clips[i]	object	Timeline clip.
.clips[i].clipUniqueId	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
.clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
.clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
.clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
.clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).

Name	Type	Description
.clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
.clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

/transports/0

The value JSON returned via the eventResponse when the `/transports/0` property changes on the device:

Name	Type	Description
.mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

/transports/0/stop

true when transport mode is InputPreview or when in Output mode and speed is 0.

The value JSON returned via the eventResponse when the `/transports/0/stop` property changes on the device:

Name	Type	Description
	boolean	true when transport mode is InputPreview or when in Output mode and speed is 0.

/transports/0/play

True when transport is in Output mode and speed is non-zero.

The value JSON returned via the eventResponse when the /transports/0/play property changes on the device:

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

/transports/0/playback

The value JSON returned via the eventResponse when the `/transports/0/playback` property changes on the device:

Name	Type	Description
.type	string	Possible values are: Play, Jog, Shuttle, Var.
.loop	boolean	When true playback loops from the end of the timeline to the beginning of the timeline.
.singleClip	boolean	When true playback loops from the end of the current clip to the beginning of the current clip.
.speed	number	Playback speed, 1.0 for normal forward playback
.position	integer	Playback position on the timeline in units of video frames.

/transports/0/record

The value JSON returned via the eventResponse when the `/transports/0/record` property changes on the device:

Name	Type	Description
.recording	boolean	Is transport in Input Record mode.

/transports/0/timecode

The value JSON returned via the eventResponse when the `/transports/0/timecode` property changes on the device:

Name	Type	Description
.display	string	The display timecode serialised as a string.
.timeline	string	The timeline timecode serialised as a string.

/transports/0/timecode/source

The value JSON returned via the eventResponse when the `/transports/0/timecode/source` property changes on the device:

Name	Type	Description
.timecode	string	Possible values are: Timeline, Clip.

/transports/0/clipIndex

The value JSON returned via the eventResponse when the `/transports/0/clipIndex` property changes on the device:

Name	Type	Description
.clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

/slates/nextClip

The value JSON returned via the eventResponse when the `/slates/nextClip` property changes on the device:

Name	Type	Description
.clip	object	
.clip.clipName	string	
.clip.reel	integer	
.clip.scene	string	
.clip.sceneLocation	string	Possible values are: Interior, Exterior.
.clip.sceneTime	string	Possible values are: Day, Night.
.clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
.clip.take	integer	
.clip.takeType	string	Possible values are: None, PU, VFX, SER.
.clip.goodTake	boolean	
.lens	object	
.lens.lensType	string	
.lens.iris	string	

Name	Type	Description
.lens.focalLength	string	
.lens.distance	string	
.lens.filter	string	
.project	object	
.project.projectName	string	
.project.director	string	
.project.camera	string	
.project.cameraOperator	string	

/monitoring/{displayName}/cleanFeed

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/cleanFeed` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/displayLUT

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/displayLUT` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/zebra

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/zebra` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/focusAssist` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGuide

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGuide` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGrids` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/safeArea

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/safeArea` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/falseColor

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/falseColor` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/focusAssist` property changes on the device:

Name	Type	Description
.mode	string	Possible values are: Peak, ColoredLines.
.color	string	Possible values are: Red, Green, Blue, White, Black.
.intensity	integer	

/monitoring/frameGuideRatio

The value JSON returned via the eventResponse when the `/monitoring/frameGuideRatio` property changes on the device:

Name	Type	Description
.ratio	string	

/monitoring/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/frameGrids` property changes on the device:

Name	Type	Description
.frameGrids	array	
.frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

/monitoring/safeAreaPercent

The value JSON returned via the eventResponse when the `/monitoring/safeAreaPercent` property changes on the device:

Name	Type	Description
.percent	integer	Safe area coverage percentage.

/audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input` property changes on the device:

Name	Type	Description
.input	string	Audio input source and type

/audio/channel/{channelIndex}/supportedInputs

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/supportedInputs` property changes on the device:

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

/audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/level` property changes on the device:

Name	Type	Description
.gain	number	Gain value in dB
.normalised	number	Normalised level value between 0.0 and 1.0

/audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/phantomPower` property changes on the device:

Name	Type	Description
.enabled	boolean	Phantom power enabled state

/audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/padding` property changes on the device:

Name	Type	Description
.enabled	boolean	Padding enabled state

/audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/lowCutFilter` property changes on the device:

Name	Type	Description
.enabled	boolean	Low cut filter enabled state

/audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/available` property changes on the device:

Name	Type	Description
.available	boolean	Whether the input is currently available

/audio/channel/{channelIndex}/input/description

Description of the current input of the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input/description` property changes on the device:

Name	Type	Description
.description	object	
.description.gainRange	object	
.description.gainRange.Min	number	The minimum gain value in dB
.description.gainRange.Max	number	The maximum gain value in dB
.description.capabilities	object	
.description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
.description.capabilities. LowCutFilter	boolean	Input supports setting of low cut filter

Name	Type	Description
.description.capabilities.Padding	object	
.description.capabilities.Padding.available	boolean	Input supports setting of padding
.description.capabilities.Padding.forced	boolean	Padding is forced to be set for the input
.description.capabilities.Padding.value	number	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.

/colorCorrection/lift

Get color correction lift

The value JSON returned via the eventResponse when the `/colorCorrection/lift` property changes on the device:

Name	Type	Description
.red	number	Red lift component
.green	number	Green lift component
.blue	number	Blue lift component
.luma	number	Luma lift component

/colorCorrection/gamma

Get color correction gamma

The value JSON returned via the eventResponse when the `/colorCorrection/gamma` property changes on the device:

Name	Type	Description
.red	number	Red gamma component
.green	number	Green gamma component
.blue	number	Blue gamma component
.luma	number	Luma gamma component

/colorCorrection/gain

Get color correction gain

The value JSON returned via the eventResponse when the `/colorCorrection/gain` property changes on the device:

Name	Type	Description
.red	number	Red gain component
.green	number	Green gain component
.blue	number	Blue gain component
.luma	number	Luma gain component

/colorCorrection/offset

Get color correction offset

The value JSON returned via the eventResponse when the `/colorCorrection/offset` property changes on the device:

Name	Type	Description
.red	number	Red offset component
.green	number	Green offset component
.blue	number	Blue offset component
.luma	number	Luma offset component

/colorCorrection/contrast

Get color correction contrast

The value JSON returned via the eventResponse when the `/colorCorrection/contrast` property changes on the device:

Name	Type	Description
.pivot	number	Contrast pivot point
.adjust	number	Contrast adjustment

/colorCorrection/color

Get color correction color properties

The value JSON returned via the eventResponse when the `/colorCorrection/color` property changes on the device:

Name	Type	Description
.hue	number	Color hue adjustment
.saturation	number	Color saturation adjustment

/colorCorrection/lumaContribution

Get color correction luma contribution

The value JSON returned via the eventResponse when the `/colorCorrection/lumaContribution` property changes on the device:

Name	Type	Description
.lumaContribution	number	Luma contribution value

/lens/iris

Get lens' aperture

The value JSON returned via the eventResponse when the `/lens/iris` property changes on the device:

Name	Type	Description
.continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
.apertureStop	number	Aperture stop value
.normalised	number	Normalised value
.apertureNumber	integer	Aperture number

/lens/iris/description

Get detailed description of lens' iris capabilities

The value JSON returned via the eventResponse when the `/lens/iris/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the iris can be controlled
.apertureStop	object	
.apertureStop.min	number	Minimum aperture stop
.apertureStop.max	number	Maximum aperture stop

/lens/focus

Get lens' focus

The value JSON returned via the eventResponse when the `/lens/focus` property changes on the device:

Name	Type	Description
.normalised	number	Normalised value

/lens/focus/description

Get detailed description of lens' focus capabilities

The value JSON returned via the eventResponse when the `/lens/focus/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the focus can be controlled
.focusDistance	object	
.focusDistance.adjustable	boolean	If focus distance is adjustable
.focusDistance.min	number	Minimum focus distance
.focusDistance.max	number	Maximum focus distance

/lens/zoom

Get lens' zoom

The value JSON returned via the eventResponse when the `/lens/zoom` property changes on the device:

Name	Type	Description
.focalLength	integer	Focal length in mm
.normalised	number	Normalised value

/lens/zoom/description

Get detailed description of lens' zoom capabilities

The value JSON returned via the eventResponse when the `/lens/zoom/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the zoom can be controlled
.focalLength	object	
.focalLength.adjustable	boolean	If focal length is adjustable
.focalLength.min	integer	Minimum focal length
.focalLength.max	integer	Maximum focal length

/presets

Get the list of the presets on the camera

The value JSON returned via the eventResponse when the `/presets` property changes on the device:

Name	Type	Description
.presets	array	List of the presets on the camera (.cset files)
.presets[i]	string	

/presets/active

Get the currently active preset on the camera

The value JSON returned via the eventResponse when the `/presets/active` property changes on the device:

Name	Type	Description
.preset	string	Name of the active preset (with .cset extension, or 'default')

/camera/colorBars

Get the status of color bars display

The value JSON returned via the eventResponse when the `/camera/colorBars` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the color bars are currently enabled

/camera/programFeedDisplay

Get the status of program feed display

The value JSON returned via the eventResponse when the `/camera/programFeedDisplay` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the program feed display is currently enabled

/camera/tallyStatus

Get the tally status of the camera

The value JSON returned via the eventResponse when the `/camera/tallyStatus` property changes on the device:

Name	Type	Description
.status	string	Current tally status of the camera Possible values are: None, Preview, Program.

/camera/power

Get the power status of the camera

The value JSON returned via the eventResponse when the `/camera/power` property changes on the device:

Name	Type	Description
.source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
.milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
.batteries	array	
.batteries[i]	object	
.batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
.batteries[i]. chargeRemainingPercent	integer	Remaining battery charge percentage
.batteries[i].statusFlags	array	List of battery status flags
.batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

/camera/power/displayMode

Get the power display mode of the camera

The value JSON returned via the eventResponse when the `/camera/power/displayMode` property changes on the device:

Name	Type	Description
.mode	string	Current power display mode Possible values are: Percentage, Voltage.

/camera/timingReferenceLock

Get the timing reference lock status

The value JSON returned via the eventResponse when the `/camera/timingReferenceLock` property changes on the device:

Name	Type	Description
.locked	boolean	Indicates if timing reference is locked

/video/iso

Get current ISO

The value JSON returned via the eventResponse when the `/video/iso` property changes on the device:

Name	Type	Description
.iso	integer	Current ISO value

/video/supportedISOs

Get the list of supported ISO settings

The value JSON returned via the eventResponse when the `/video/supportedISOs` property changes on the device:

Name	Type	Description
.supportedISOs	array	Array of supported ISO values
.supportedISOs[i]	integer	

/video/gain

Get current gain value in decibels

The value JSON returned via the eventResponse when the `/video/gain` property changes on the device:

Name	Type	Description
.gain	integer	Current gain value in decibels

/video/supportedGains

Get the list of supported gain settings in decibels

The value JSON returned via the eventResponse when the `/video/supportedGains` property changes on the device:

Name	Type	Description
.supportedGains	array	Array of supported gain values in decibels
.supportedGains[i]	integer	

/video/whiteBalance

Get current white balance

The value JSON returned via the eventResponse when the `/video/whiteBalance` property changes on the device:

Name	Type	Description
.whiteBalance	integer	Current white balance

/video/whiteBalance/description

Get white balance range

The value JSON returned via the eventResponse when the `/video/whiteBalance/description` property changes on the device:

Name	Type	Description
.whiteBalance	object	
.whiteBalance.min	integer	Minimum color temperature
.whiteBalance.max	integer	Maximum color temperature

/video/whiteBalanceTint

Get white balance tint

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint` property changes on the device:

Name	Type	Description
.whiteBalanceTint	integer	Current white balance tint

/video/whiteBalanceTint/description

Get white balance tint range

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint/description` property changes on the device:

Name	Type	Description
.whiteBalanceTint	object	
.whiteBalanceTint.min	integer	Minimum white balance tint
.whiteBalanceTint.max	integer	Maximum white balance tint

/video/ndFilter

Get ND filter stop

The value JSON returned via the eventResponse when the `/video/ndFilter` property changes on the device:

Name	Type	Description
.stop	number	Current filter power (fStop)

/video/supportedNDFilters

Get the list of available ND filter stops

The value JSON returned via the eventResponse when the `/video/supportedNDFilters` property changes on the device:

Name	Type	Description
.supportedStops	array	Array of available ND filter stops
.supportedStops[i]	number	

/video/ndFilter/displayMode

Get ND filter display mode on the camera

The value JSON returned via the eventResponse when the `/video/ndFilter/displayMode` property changes on the device:

Name	Type	Description
.displayMode	string	ND filter display mode Possible values are: Stop, Number, Fraction.

/video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

The value JSON returned via the eventResponse when the `/video/supportedNDFilterDisplayModes` property changes on the device:

Name	Type	Description
.supportedDisplayModes	array	Array of supported display modes
.supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

/video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

The value JSON returned via the eventResponse when the `/video/ndFilterSelectable` property changes on the device:

Name	Type	Description
.selectable	boolean	True if ND filter adjustments are selectable via a slider

/video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

The value JSON returned via the eventResponse when the `/video/shutter` property changes on the device:

Name	Type	Description
.continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
.shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
.shutterAngle	number	Shutter angle

/video/shutter/measurement

Get the current shutter measurement mode

The value JSON returned via the eventResponse when the `/video/shutter/measurement` property changes on the device:

Name	Type	Description
.measurement	string	Current shutter measurement mode Possible values are: ShutterAngle, ShutterSpeed.

/video/supportedShutters

Get supported shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/supportedShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of supported shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array of flicker-free shutter speeds
.shutterSpeeds[i]	integer	

/video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/flickerFreeShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of flicker-free shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array
.shutterSpeeds[i]	integer	

/video/autoExposure

Get current auto exposure mode

The value JSON returned via the eventResponse when the `/video/autoExposure` property changes on the device:

Name	Type	Description
.mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
.type	string	Comma-separated list of device types in the auto exposure stack

/video/detailSharpening

Get the current state of detail sharpening

The value JSON returned via the eventResponse when the `/video/detailSharpening` property changes on the device:

Name	Type	Description
.enabled	boolean	Whether detail sharpening is enabled

/video/detailSharpeningLevel

Get the current detail sharpening level

The value JSON returned via the eventResponse when the `/video/detailSharpeningLevel` property changes on the device:

Name	Type	Description
.level	string	Current detail sharpening level of supported shutter speeds Possible values are: Low, Medium, High.

Hilfe

Am schnellsten erhalten Sie Hilfe über die Online-Support-Seiten auf der Blackmagic Design Website. Sehen Sie dort nach der aktuellsten Support-Dokumentation für Ihre Kamera.

Blackmagic Design Online Support Seiten

Die aktuellste Software kann im Blackmagic Design Support-Center unter www.blackmagicdesign.com/de/support heruntergeladen werden.

Kontaktaufnahme mit Blackmagic Design Support

Wenn unser Support-Material Ihnen nicht wie gewünscht hilft, gehen Sie auf unsere Support-Seite. Klicken Sie dort auf „Senden Sie uns eine E-Mail“ und schicken Sie uns Ihre Support-Anfrage. Oder klicken Sie auf „Finden Sie Ihr lokales Support-Team“ und rufen Sie Ihre nächstgelegene Blackmagic Design Support-Stelle an.

Überprüfen der aktuell installierten Softwareversion

Um zu überprüfen, welche Version von Blackmagic Camera Setup auf Ihrem Computer installiert ist, öffnen Sie das Fenster „About Blackmagic Camera Setup“.

- Unter Mac: Öffnen Sie das Blackmagic Camera Setup Dienstprogramm über den Ordner „Programme“. Wählen Sie im Anwendungsmenü „About Blackmagic Camera Setup“, um die Versionsnummer nachzusehen.
- Unter Windows: Öffnen Sie Blackmagic Camera Setup über das Startmenü oder den Startbildschirm. Um die Versionsnummer anzuzeigen, klicken Sie auf das Menü „Hilfe“ und wählen Sie „About Blackmagic Camera Setup“.

So erhalten Sie die aktuellsten Software-Updates

Prüfen Sie zunächst, welche Version des Dienstprogramms Blackmagic Camera Utility auf Ihrem Computer installiert ist. Sehen Sie dann im Blackmagic Support Center www.blackmagicdesign.com/de/support nach den neuesten Aktualisierungen. In der Regel empfiehlt es sich, die neuesten Updates zu laden. Vermeiden Sie Software-Updates jedoch mitten in einem wichtigen Projekt.

Gesetzliche Vorschriften



Entsorgung von elektrischen und elektronischen Geräten innerhalb der Europäischen Union

Das auf dem Produkt abgebildete Symbol weist darauf hin, dass dieses Gerät nicht zusammen mit anderen Abfallstoffen entsorgt werden darf. Altgeräte müssen daher zur Wiederverwertung an eine dafür vorgesehene Sammelstelle übergeben werden. Mülltrennung und Wiederverwertung von Altgeräten tragen zum nachhaltigen Umgang mit natürlichen Ressourcen bei. Gleichzeitig wird sichergestellt, dass die Wiederverwertung nicht zulasten der menschlichen Gesundheit und der Umwelt geht. Weitere Informationen zur Entsorgung von Altgeräten sowie zu den Standorten der zuständigen Sammelstellen erhalten Sie von Ihren örtlichen Müllentsorgungsbetrieben sowie vom Händler, bei dem Sie dieses Produkt erworben haben.



Dieses Gerät wurde geprüft und entspricht den Grenzwerten für Digitalgeräte der Klasse A gemäß Abschnitt 15 der FCC-Bestimmungen für Funkentstörung. Diese Grenzwerte dienen dem angemessenen Schutz gegen schädliche Störungen bei Betrieb des Geräts in einer gewerblichen Umgebung. Geräte dieser Art erzeugen und verwenden Hochfrequenzen und können diese auch ausstrahlen. Bei Nichteinhaltung der Installations- und Gebrauchsvorschriften können sie zu Störungen beim Rundfunkempfang führen. Der Betrieb solcher Geräte in Wohngebieten führt mit großer Wahrscheinlichkeit zu Funkstörungen. In diesem Fall kann vom Betreiber verlangt werden, selbst für die Beseitigung solcher Störungen aufzukommen.

Der Betrieb unterliegt den folgenden zwei Bedingungen:

1. Dieses Gerät darf keine schädigenden Störungen hervorrufen.
2. Dieses Gerät muss allen Störungen standhalten, einschließlich Störungen, die ggf. einen unerwünschten Betrieb zur Folge haben.



R-R-BMD-20250213001



ISED-Zertifizierung für den kanadischen Markt

Dieses Gerät erfüllt die kanadischen Vorschriften für digitale Geräte der Klasse A.

Jedwede an diesem Produkt vorgenommene Änderung oder unsachgemäße Verwendung kann die Konformitätserklärung zum Erlöschen bringen.

Verbindungen zu HDMI-Schnittstellen müssen über abgeschirmte HDMI-Kabel hergestellt werden.

Dieses Gerät wurde geprüft und erfüllt die Bestimmungen zum Einsatz in einem gewerblichen Umfeld. Bei Einsatz des Geräts in einer häuslichen Umgebung verursacht es möglicherweise Funkstörungen.

Sicherheitshinweise

Das für die Blackmagic URSA Cine Immersive mitgelieferte 24V-Netzteil mit AC/DC-Wandler muss an ein vorschriftmäßig geerdetes Stromnetz angeschlossen werden.

Das Gehäuseinnere der Blackmagic Blackmagic URSA Cine Immersive enthält keine durch den Anwender zu wartenden Teile. Wenden Sie sich für die Wartung an ein Blackmagic Design Service-Center in Ihrer Nähe.

Wenn der Eingangsstromanschluss an einen externen DC-Akku angeschlossen wird, muss die Versorgungsverkabelung eine Strombegrenzung oder Sicherung enthalten.

Die verwendeten Kabel sollten mit VW-1 gekennzeichnet sein oder die entsprechenden Vorgaben der Normen IEC 60332 oder IEC 60695 erfüllen.

Wir empfehlen für diese Kamera eine Gleichstromquelle mit 24 Volt. Bei Verwendung einer 12V-Gleichstromquelle sollte die Verkabelung für eine Stromstärke von 14 A ausgelegt sein.

Im Lieferumfang dieses Produkts ist eine B-Mount-Akkuplatte enthalten. Wir empfehlen die Verwendung eines 24V-Akkus (28,8 Volt) oder eines Doppelspannungsakkus.

Bei Verwendung eines 12V-Akkus (14,4 Volt) entnehmen Sie den maximalen Dauerentladestrom bitte dem Handbuch Ihres Akkus oder den Angaben auf dem Akku. Wir empfehlen eine Mindeststromstärke von 12 A.

California Proposition 65

Plastikteile dieses Produkts können Spuren von polybromierten Biphenylen enthalten. Im US-amerikanischen Bundesstaat Kalifornien werden diese Chemikalien mit Krebs, Geburtsfehlern und anderen Schäden der Fortpflanzungsfähigkeit in Verbindung gebracht.

Weitere Informationen finden Sie unter www.P65Warnings.ca.gov.

Niederlassung in Europa:

Blackmagic Design Europe B.V.
Rijnlanderweg 766, Unit D
2132 NM Hoofddorp
NL

Blackmagic URSA Cine

Sorgen Sie bei sonnigen Verhältnissen für Schatten, um zu vermeiden, dass Ihre Kamera und der Lithium-Akku für längere Zeit direkter Sonneneinstrahlung ausgesetzt sind. Halten Sie alle Lithium-Akkus von Hitzequellen fern.

Der 24V-Ausgangsverbinder „RS“ eignet sich zur Stromversorgung von Fremdanbieter-Zubehörteilen. Es muss gewährleistet sein, dass die Stromabnahme weniger als 48 W beträgt.

Garantie

Eingeschränkte Garantie

Für dieses Produkt gewährt die Firma Blackmagic Design eine Garantie auf Material- und Verarbeitungsfehler von 12 Monaten ab Kaufdatum. Sollte sich ein Produkt innerhalb dieser Garantiezeit als fehlerhaft erweisen, wird die Firma Blackmagic Design nach ihrem Ermessen das defekte Produkt entweder ohne Kostenerhebung für Teile und Arbeitszeit reparieren oder Ihnen das defekte Produkt ersetzen.

Zur Inanspruchnahme der Garantieleistungen müssen Sie als Kunde Blackmagic Design über den Defekt innerhalb der Garantiezeit in Kenntnis setzen und die entsprechenden Vorkehrungen für die Leistungserbringung treffen. Es obliegt dem Kunden, für die Verpackung und den bezahlten Versand des defekten Produkts an ein spezielles von Blackmagic Design benanntes Service Center zu sorgen und hierfür aufzukommen. Sämtliche Versandkosten, Versicherungen, Zölle, Steuern und sonstige Abgaben im Zusammenhang mit der Rücksendung von Waren an uns, ungeachtet des Grundes, sind vom Kunden zu tragen.

Diese Garantie gilt nicht für Mängel, Fehler oder Schäden, die durch unsachgemäße Handhabung oder unsachgemäße oder unzureichende Wartung und Pflege verursacht wurden. Blackmagic Design ist im Rahmen dieser Garantie nicht verpflichtet, die folgenden Serviceleistungen zu erbringen: a) Behebung von Schäden infolge von Versuchen Dritter, die Installation, Reparatur oder Wartung des Produkts vorzunehmen, b) Behebung von Schäden aufgrund von unsachgemäßer Handhabung oder Anschluss an nicht kompatible Geräte, c) Behebung von Schäden oder Störungen, die durch die Verwendung von nicht Blackmagic Design Ersatzteilen oder -Verbrauchsmaterialien entstanden sind, d) Service für ein Produkt, das verändert oder in andere Produkte integriert wurde, sofern eine solche Änderung oder Integration zu einer Erhöhung des Zeitaufwands oder zu Schwierigkeiten bei der Wartung des Produkts führt.

Schützen Sie Ihren URSA Viewfinder vor direktem Sonnenlicht, da seine Optik wie eine Lupe wirkt und das Sucher-Display andernfalls beschädigt werden könnte. Wenn auf dem OLED über einen längeren Zeitraum statische oder hochkontrastige Bilder wie Bildrandmarkierungen angezeigt werden, können dort Geisterbilder und Burn-ins auftreten. Verhindern Sie dies, indem Sie ein versehentliches Verdecken des IR-Sensors vermeiden und den Sucher ausschalten, wenn er für einen längeren Zeitraum nicht in Gebrauch ist. Das Auftreten von Geisterbildern fällt nicht unter die Produktgarantie.

ÜBER DIE IN DIESER GARANTIEERKLÄRUNG AUSDRÜCKLICH AUFGEFÜHRTEN ANSPRÜCHE HINAUS ÜBERNIMMT BLACKMAGIC DESIGN KEINE WEITEREN GARANTIEN, WEDER AUSDRÜCKLICH NOCH STILLSCHWEIGEND. DIE FIRMA BLACKMAGIC DESIGN UND IHRE HÄNDLER LEHNEN JEGLICHE STILLSCHWEIGENDEN GARANTIEN IN BEZUG AUF AUSSAGEN ZUR MARKTGÄNGIGKEIT UND GEBRAUCHSTAUGLICHKEIT FÜR EINEN BESTIMMTEN ZWECK AB. DIE VERANTWORTUNG VON BLACKMAGIC DESIGN, FEHLERHAFTE PRODUKTE WÄHREND DER GARANTIEZEIT ZU REPARIEREN ODER ZU ERSETZEN, IST DIE EINZIGE UND AUSSCHLIESSLICHE ABHILFE, DIE GEGENÜBER DEM KUNDEN ZUR VERFÜGUNG GESTELLT WIRD. BLACKMAGIC DESIGN HAFTET NICHT FÜR INDIREKTE, SPEZIELLE, NEBEN- ODER FOLGESCHÄDEN, UNABHÄNGIG DAVON, OB BLACKMAGIC DESIGN ODER DER HÄNDLER VON DER MÖGLICHKEIT SOLCHER SCHÄDEN ZUVOR IN KENNTNIS GESETZT WURDE. BLACKMAGIC DESIGN IST NICHT HAFTBAR FÜR JEGLICHE WIDERRECHTLICHE VERWENDUNG DER GERÄTE DURCH DEN KUNDEN. BLACKMAGIC HAFTET NICHT FÜR SCHÄDEN, DIE SICH AUS DER VERWENDUNG DES PRODUKTS ERGEBEN. NUTZUNG DES PRODUKTS AUF EIGENE GEFAHR.

© Copyright 2025 Blackmagic Design. Alle Rechte vorbehalten. „Blackmagic Design“, „URSA“, „DeckLink“, „HDLink“, „Workgroup Videohub“, „Multibridge Pro“, „Multibridge Extreme“, „Intensity“ und „Leading the creative video revolution“ sind eingetragene Warenzeichen in den USA und in anderen Ländern. Alle anderen Unternehmens- und Produktnamen sind möglicherweise Warenzeichen der jeweiligen Firmen, mit denen sie verbunden sind.

Die Bluetooth®-Wortmarke und -Logos sind als Eigentum der Bluetooth SIG, Inc. registriert. Die Nutzung dieser Marken durch Blackmagic erfolgt unter Lizenz. Andere ausgewiesene Markennamen und Handelsmarken gehören ihren jeweiligen Eigentümern.

Octubre de 2025

Guía de usuario

Blackmagicdesign

Blackmagic URSA Cine Immersive



Índice

Blackmagic URSA Cine Immersive 613
¿Qué es el formato inmersivo? 613
Apple Immersive Video 613
Blackmagic URSA Cine Immersive 614
Primeros pasos 615
Encendido de la cámara 615
Ajustes de grabación 616
Supervisión de imágenes 618
Opciones de supervisión e información en pantalla 618
Cómo encuadrar la primera toma 619
Altura de la cámara 619
Desplazamiento vertical 620
Posicionamiento 621
Independencia del público 622
Movimiento de cámara 622
Velocidad de transferencia de datos 623
Audio y código de tiempo 624
Formato Apple Spatial Audio 624
Sonido ambisónico 624
Micrófonos de solapa y miniatura 625
Micrófonos de mano 625
Código de tiempo 625
Grabación 627
Requerimientos de exposición 627
Velocidad de obturación 627
Sensibilidad ISO y rango dinámico 628
Filtros de densidad neutra 629
Frecuencia independiente 630
Histograma 631
Opciones de supervisión 631
Grabación 632
Reproducción 633
Transferencia de clips 634
Resumen 635
DaVinci Resolve Studio 636
Recomendaciones para la terminal de trabajo 636
Configuración del proyecto 637
Ajustes principales 637
Monitorización de imágenes 638
Redimensionamiento 638
Gestión cromática 639
Acerca de DaVinci Wide Gamut 640
Procesamiento en formato Apple Immersive 641
Ingesta 641
Edición 643
Montaje 643
Fondos 643
Uso del visor 645
Transiciones 647
Datos de movimiento inmersivos de Apple 648
Títulos 649
Consideraciones creativas 650
Composiciones con efectos visuales (VFX) 651
Nodos MediaOut 651
Nodo Renderer3D 652
Composiciones con herramientas inmersivas de parche 653
Ajustes cromáticos 654
Reducción de ruido 654
Correcciones cromáticas primarias 655
Correcciones cromáticas secundarias 655
Ajuste de tamaño 656
Panel de imágenes estereoscópicas (3D) 656
Gestión cromática con cambios del espacio cromático 658
Asignación de cambio del espacio cromático en la línea de tiempo 658
Cambio del espacio cromático en monitores de etalonaje 659
Cambio del espacio cromático para supervisión en dispositivos Apple Vision Pro 661
Audio 662
Formato Apple Spatial Audio Format (ASAF) 662
Entrega 663
Omitir transiciones 663
Transmisión a dispositivos Apple Vision Pro 664
Acerca de Apple Immersive Video Utility 664
Developer Information 665
Camera Control REST API 665
Livestream Control API 667
Clips Control API 671

Media Pool Control API 672
Monitoring Control API 675
Event Control API 684
System Control API 684
Transport Control API 690
Timeline Control API 694
Media Control API 697
Slate Control API 700
Preset Control API 706
Audio Control API 708
Lens Control API 713
Video Control API 717
Camera Control API 725
Immersive Control API 728
Color Correction Control API 729
Notification websocket - 1.0.0 734
messages 734
Ayuda 758
Normativas 759
Seguridad 760
Garantía 761

Blackmagic URSA Cine Immersive

Felicidades por haber adquirido o alquilado el modelo Blackmagic URSA Cine Immersive.

Esta cámara ha sido diseñada para filmar contenidos estereoscópicos de realidad virtual en 180°, los cuales suelen denominarse inmersivos. En particular, se ha optimizado para grabar en formato Apple Immersive Video para dispositivos Apple Vision Pro y brindar una experiencia inmersiva con la mejor calidad disponible en la actualidad.

Esta guía ha sido elaborada con el fin de brindar una introducción sobre esta unidad y cómo grabar con ella, al tiempo que pone énfasis en su funcionamiento básico y en los archivos Apple Immersive Video, para los cuales fue desarrollada. Comprender cómo se comporta este formato será de utilidad para entender mejor este modelo.

En las páginas a continuación, se presenta una combinación de asesoramiento técnico y creativo que será beneficioso tanto para utilizar la cámara como para conocer las ventajas del formato inmersivo.

¿Qué es el formato inmersivo?

Se trata de un tipo de formato de imágenes espaciales tridimensionales. Es posible que esté al tanto de otras tecnologías similares que existen desde hace tiempo. A continuación se brinda una visión general de ellas a fin de ayudar a ilustrar en qué se diferencia el formato inmersivo.

- **Cine 3D tradicional:** Permite al público experimentar la profundidad dentro de un fotograma al captar cada escena con dos cámaras que emplean una distancia interaxial que imita la separación que hay entre los ojos en los humanos. Si bien las imágenes estereoscópicas brindan al espectador la posibilidad de experimentar la profundidad, en este caso la perspectiva que percibe está determinada por la composición del director de fotografía.
- **Contenidos esféricos:** Combinan los principios de las cámaras 3D y de las de 360° con el fin de permitir que el espectador pueda experimentar la profundidad al mirar en cualquier dirección. Las imágenes en 360° requieren de resoluciones muy elevadas para garantizar que la pequeña sección de la imagen que mira el espectador sea lo suficientemente fiel. Dado que la fisiología humana solo nos permite girar la cabeza de lado a lado, para poder mirar hacia atrás la persona suele tener que estar de pie o sentarse en una silla giratoria.
- **Imágenes volumétricas:** Permiten al público no solo experimentar la profundidad y mirar alrededor, sino también moverse y cambiar de perspectiva. Si bien es posible combinar varias cámaras configuradas de una manera particular con el fin de captar contenidos audiovisuales volumétricos, la mayoría de las imágenes de realidad virtual o aumentada recurren a archivos 3D renderizados dinámicamente y requieren de un ambiente seguro para los participantes.

El formato inmersivo combina lo mejor del cine tradicional con archivos 3D interactivos. En términos simples, son imágenes estereoscópicas en 180° con una alta resolución y una frecuencia de imagen elevada.

Apple Immersive Video

El modelo Blackmagic URSA Cine Immersive ha sido diseñado específicamente para grabar en formato Apple Immersive Video, formato empleado por los dispositivos Apple Vision Pro, que es compatible con contenidos 3D estereoscópicos en alta resolución con un campo de visión de 180°.

Características del formato Apple Immersive Video:

- Reproducción a 90 fotogramas por segundo.
- Distribución a una resolución máxima de 4320 × 4320 por ojo.
- Sistema de metadatos único para representar la imagen original de la cámara en un entorno de 180°.
- Formato Apple Spatial Audio a fin de incluir sonido 3D.

El formato Apple Immersive Video requiere de equipos capaces de reproducir imágenes estereoscópicas de alta resolución en 180°. Actualmente, Apple Vision Pro es el único dispositivo compatible.



Apple Vision Pro

Este visor dispone de dos pantallas micro OLED de 3660 × 3200 pixeles con una densidad de pixeles de 7.5 micrones. Además, incluye 13 cámaras de seguimiento, un escáner LiDAR, unidades de medida internas, sensores de seguimiento a fin de alinear las imágenes con el movimiento de la cabeza y los ojos.

Asimismo, dispone de un procesador M2 con una CPU de 8 núcleos y una GPU de 10 núcleos que permiten reproducir contenidos a una frecuencia de imagen máxima de 100 fotogramas por segundo.

Blackmagic URSA Cine Immersive

El modelo Blackmagic URSA Cine Immersive ha sido concebido para grabar contenidos en formato Apple Immersive Video para dispositivos Apple Vision Pro.



Blackmagic URSA Cine Immersive

Esta cámara se vale de dos sensores RGBW de 58 megapixeles que presentan un diseño casi cuadrado y ofrecen la oportunidad de filmar un campo de visión de 180°, tanto en sentido horizontal como vertical. Además, los sensores incluyen objetivos ojo de pez de 210°, entre las cuales hay una separación de 64 mm.

Los filtros de densidad neutra integrados proporcionan 8 pasos de control de luz. La apertura de los objetivos es fija y tiene un valor aproximado de f4.5.

La cámara graba dos imágenes por separado en 8K a 90 fotogramas por segundo, lo cual da como resultado velocidades de transferencia más altas que al grabar en 4K 24p.

Primeros pasos

Para comenzar a utilizar la cámara, basta con conectarla a una fuente de alimentación y encenderla. Este modelo incluye una unidad de almacenamiento Blackmagic Media Module 8TB en el compartimiento respectivo. Este soporte de grabación ya se encuentra formateado y permite comenzar a grabar de inmediato.

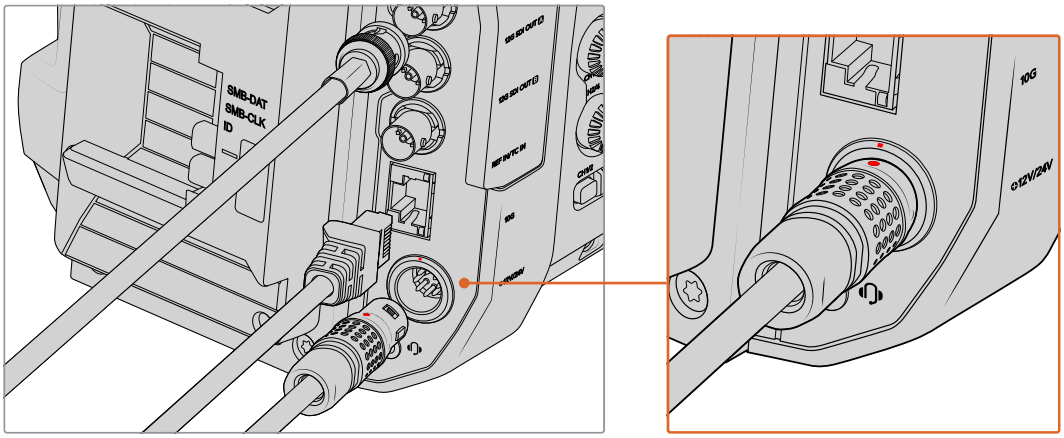
Encendido de la cámara

La forma más rápida de encender la cámara es conectando el adaptador incluido.

Para conectar la cámara a una fuente de alimentación externa:

1. En primer lugar, quite la tapa protectora de goma y guárdela en un sitio seguro. Conecte el transformador de 24 V para CC incluido a la red de suministro eléctrico mediante un cable IEC C13.
2. Enchufe el conector de 8 pines del transformador a la entrada de alimentación (24 V CC) situada en el panel trasero de la cámara.

Si hay una batería y una fuente de alimentación externa conectadas simultáneamente, la cámara solo utilizará la corriente suministrada por esta última. Al desconectarla, seguirá funcionando con la batería.



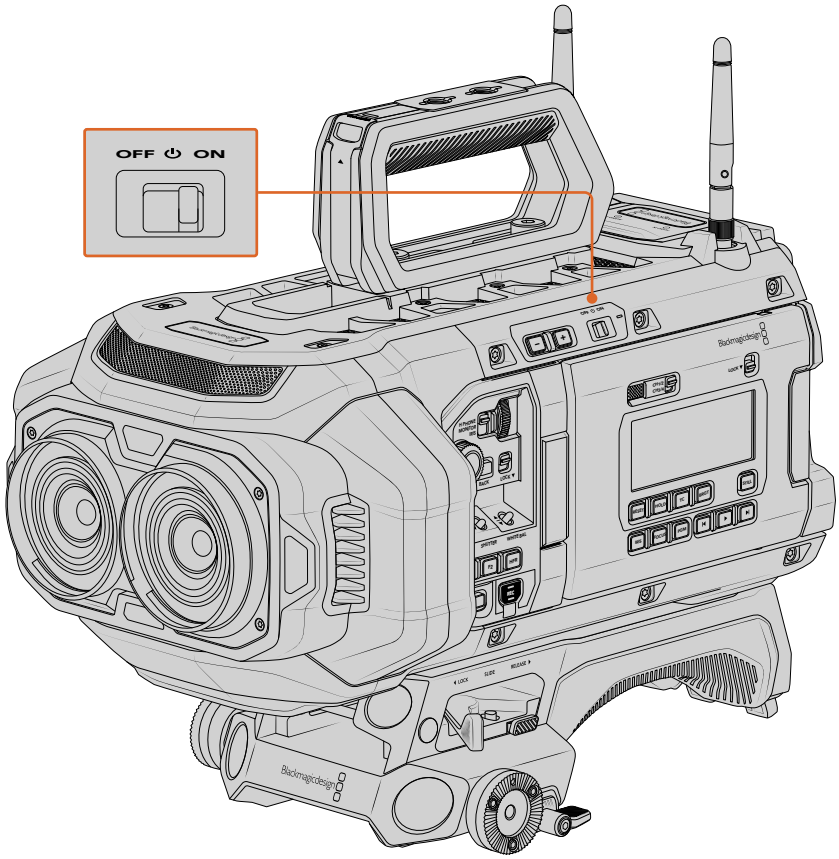
A los efectos de conectar la fuente de alimentación, haga coincidir los puntos rojos y empuje suavemente el conector hasta que calce en su lugar. Para desconectarla, sostenga el capuchón metálico del conector y tire del mismo delicadamente.

La cámara incluye un soporte para baterías de 24 V con montura en B. Un voltaje de 24 a 34 V es ideal para el funcionamiento de este modelo, ya que reduce las pérdidas de temperatura y permite grabar a frecuencias de imagen más altas, así como alimentar diversos accesorios.

También existen soportes opcionales para instalar baterías VLock y Gold Mount compatibles con 24 V y monturas para baterías Gold Mount Plus de 26 V fabricadas por terceros.

NOTA: Cuando la cámara funciona con fuentes de alimentación cuyo voltaje se sitúa entre los 12 V y los 18 V, no es posible alimentar accesorios mediante el conector RS de 3 pines o el LEMO de 2 pines.

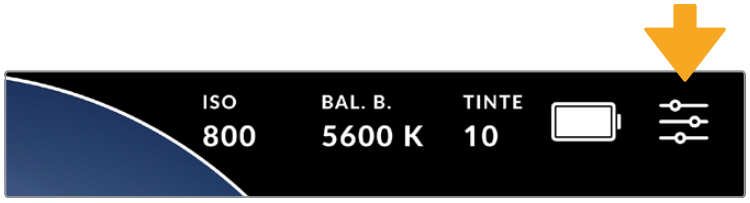
Después de colocar la batería o conectar una fuente de alimentación externa, encienda la cámara. Mueva el interruptor en la parte superior de la unidad a la posición **ON**.



Mueva el interruptor de encendido a la posición **ON** para prender la cámara.

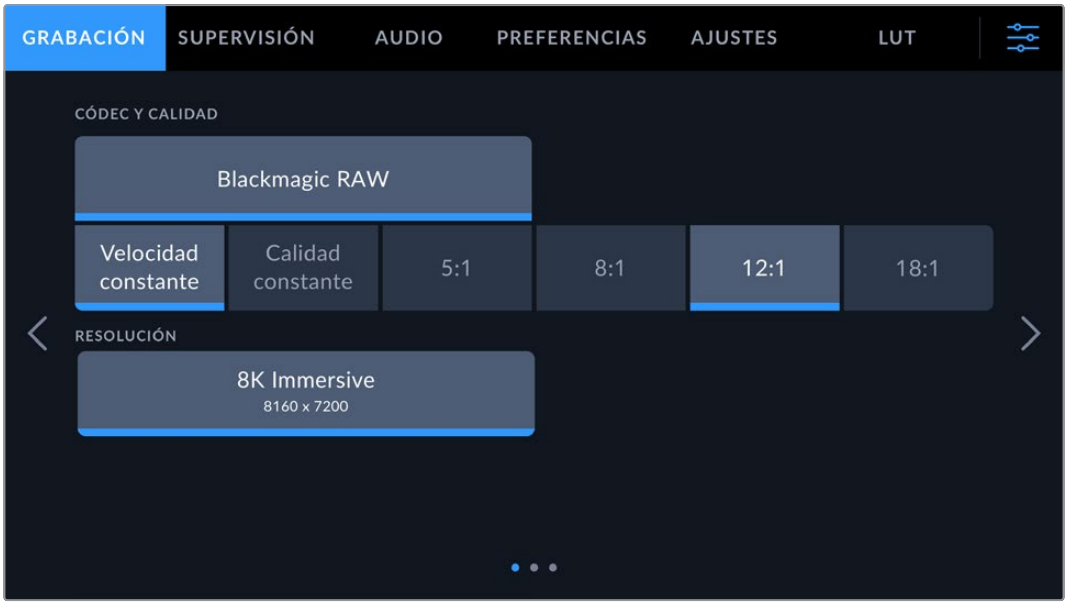
Ajustes de grabación

Este apartado proporciona información sobre los ajustes de cámara específicos o importantes al filmar contenidos inmersivos. Para abrir el menú de ajustes de cámara, pulse sobre el ícono correspondiente en la esquina superior derecha de la pantalla desplegable o secundaria.



Grabación 1

Las opciones de **CÓDEC Y CALIDAD** en este grupo de ajustes permiten seleccionar el formato Blackmagic RAW que se adecue a su proyecto. Estas se dividen en cuatro ajustes de velocidad de transferencia y calidad constantes. Es muy recomendable utilizar una compresión de 12:1 o Q3 al grabar contenidos a 90 f/s para dispositivos Apple Vision Pro. Estos ajustes son ideales para mantener una velocidad de transferencia manejable y archivos de un tamaño factible para los equipos informáticos actuales y, a la vez, retener una calidad de imagen sumamente alta.



Grabación 2

El segundo grupo de ajustes de grabación incluye opciones para configurar la frecuencia de imagen.



Esta debería establecerse en 90 fotogramas por segundo, de modo que sea acorde a la de reproducción en dispositivos Apple Vision Pro.

En grabaciones destinadas a otros sistemas, podría ser suficiente emplear frecuencias de imagen menores, por ejemplo, 60 o 50 f/s. Las frecuencias más bajas, como 24 o 25 f/s, propias del estilo cinematográfico tradicional, pueden resultar adecuadas para grabar secuencias para efectos visuales que se utilizarán en contenidos cinematográficos en 24p.

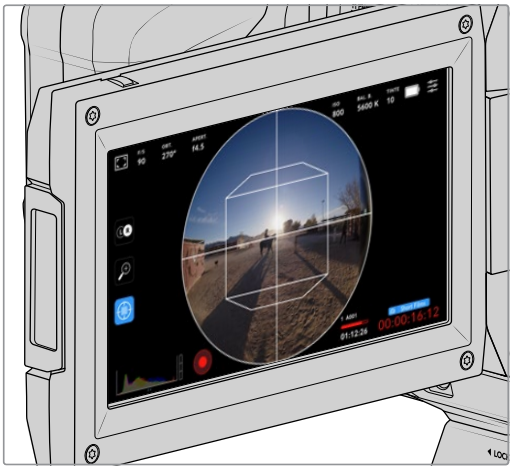
Esto es todo lo que es preciso saber para comenzar a utilizar la cámara. Dado que el módulo de grabación ya se encuentra formateado, es posible comenzar a filmar inmediatamente.

Supervisión de imágenes

Esta cámara ofrece diversas opciones para supervisar las imágenes durante la grabación.

La pantalla táctil desplegable se concibió para ser usada por el operador de cámara y puede plegarse para reducir el perfil físico de la cámara sin ocultar la pantalla, o bien posicionarse hacia delante para supervisar las imágenes al grabarse a uno mismo.

La pantalla secundaria permite que otros miembros del equipo técnico supervisen la señal sin obstruir la vista del operador. Dado que la cámara tiene un campo de visión extremadamente amplio, se debe tener cautela al utilizar esta pantalla.





Opciones de supervisión e información en pantalla

La cámara muestra el espacio lenticular de a un ojo por vez. Las pantallas táctiles plegables secundarias están configuradas para mostrar una vista de 180°. La señal transmitida a través de la salida SDI muestra la imagen de 210° captada por cada sensor.



Todos los monitores conectados a las salidas de supervisión muestran la señal correspondiente al mismo ojo. Para alternar entre el izquierdo y el derecho, pulse **I D** en cualquiera de las pantallas táctiles.

Asimismo, es posible ver información adicional presionando el botón correspondiente. Estos datos incluyen:

- Un límite externo de 180°.
- Una cruz filar que indica el centro del fotograma.
- Una guía interna que representa el campo de visión aproximado que se percibe al usar dispositivos Apple Vision Pro y mirar hacia delante.

Estas herramientas están diseñadas para ayudar a lograr un encuadre y un posicionamiento precisos durante la captura de imágenes inmersivas.

Cómo encuadrar la primera toma

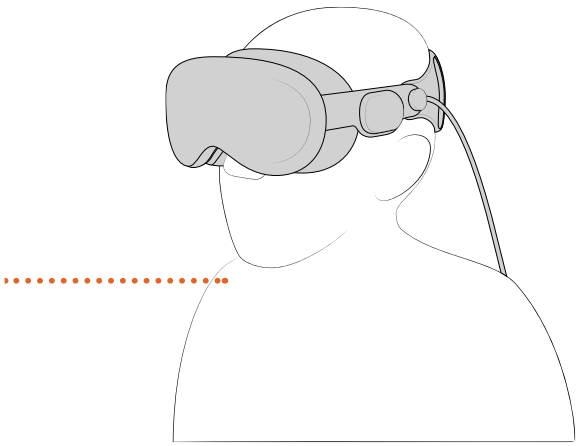
Los contenidos y las imágenes 2D tradicionales comparten muchas características en común, pero también presentan diferencias marcadas.

Por ejemplo, es posible que alguna vez estando en el auto haya sufrido mareo cinético por estar demasiado absorto mirando el teléfono. Esto se produce por una discordancia sensorial, ya que el cerebro recibe señales contradictorias: el oído interno indica al cuerpo que se está moviendo, pero los ojos indican lo contrario.

Esto puede ocurrir fácilmente con los contenidos inmersivos. Por ejemplo, al hacer un paneo de cámara, los ojos del espectador le indicarán que se está moviendo, pero el cuello y el oído interno le indican que esto no es así. En consecuencia, se recomienda dominar las tomas fijas antes de aprender a hacer movimientos de cámara que resulten placenteros para el público.

Altura de la cámara

En el cine 2D tradicional, una técnica es ubicar la cámara en una posición de mayor o menor elevación para indicar el equilibrio de poder en una escena. Al rodar en formato inmersivo, es preciso considerar el impacto de colocar la cámara en el suelo. ¿Cómo podría afectar al espectador? Podría darle la sensación de que está enterrado hasta el cuello.



La altura del suelo percibida puede afectar la experiencia visual. Si la cámara se sitúa a una altura baja, la posición del suelo percibida podría parecer poco lógica.



Al filmar contenidos inmersivos, los ángulos contrapicados podrían resultar incómodos para el público.

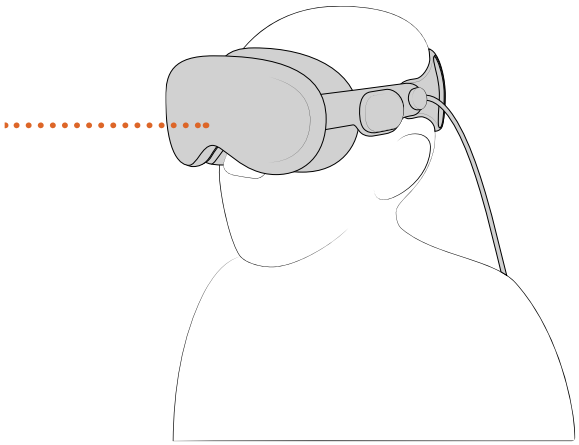
Para ayudar al espectador a sentirse presente en el lugar, suele ser mejor optar por una altura y una posición de cámara que le haga sentir que está parado o sentado en una escena. A pesar de esto, aún hay margen para expresar la intención creativa. Si hay personas presentes en la escena, elevar ligeramente la cámara puede infundir poder en el espectador, al permitirle mirar a otros desde arriba. Por el contrario, al situarla en una posición levemente inferior, la persona se verá forzada a mirar a otros desde abajo.

No obstante, esto no significa que no sea posible mantener la comodidad del público ubicando la cámara de una manera más osada. Las cámaras voladoras, los planos cenitales, los ángulos contrapicados y muchas otras técnicas se están utilizando con éxito en las producciones inmersivas. Al combinarlas con la visión, el movimiento y el contexto adecuados, pueden resultar entretenidas para el público.

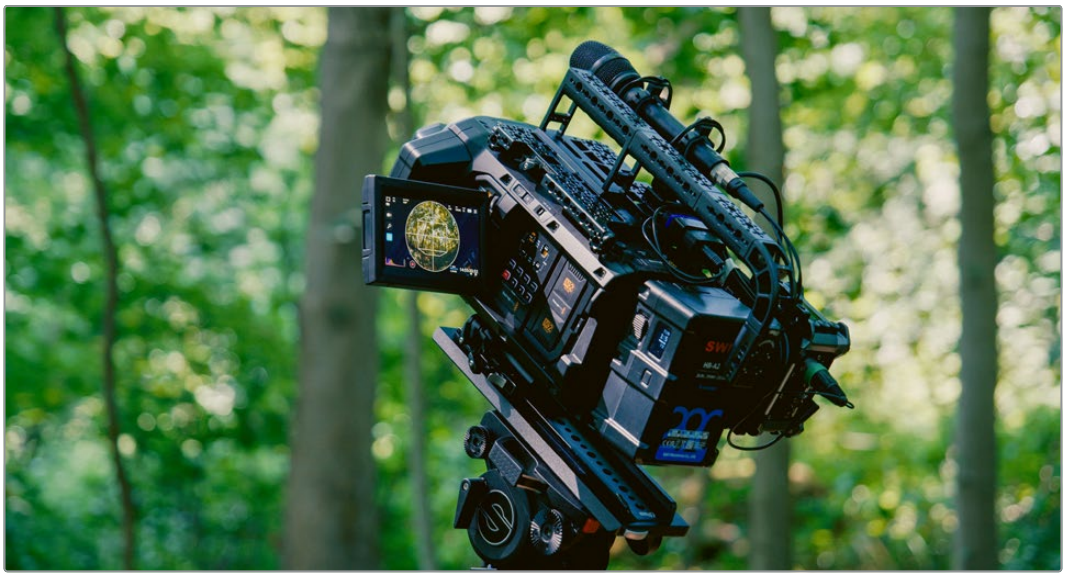
Desplazamiento vertical

Al ubicar la cámara por lo alto o lo bajo en rodajes 2D tradicionales, el director de fotografía suele girar la cámara hacia arriba o abajo para mantener la composición. Por lo tanto, se debe considerar cómo esto podría afectar en las imágenes inmersivas.

Si se apuntase la cámara hacia abajo, para poder ver el horizonte el público tendría que mover la cabeza hacia arriba. Sería poco probable que esto generara mareo cinético, pero podría provocar un quiebre en la escena en la que se encuentra porque, en la vida real, el horizonte siempre está hacia delante.



Las tomas inmersivas serán más lógicas cuando el horizonte se ubique a 0°.



Si se girara la cámara verticalmente, el público sentiría una discrepancia entre hacia dónde tiene que mirar y adónde miraría normalmente para ver el horizonte.

Por consiguiente, la mayoría de las escenas inmersivas se captan con una inclinación de 0° o un valor próximo a 0. A medida que la cámara registra un campo de visión de más de 180°, tanto en sentido vertical como horizontal, el público todavía puede mirar hacia arriba y hacia abajo. Pero en lugar de que el cineasta fuerce al espectador a mirar en una dirección en particular, la persona tiene el control.

Para alentar al espectador a mirar hacia distintas regiones de la imagen, se puede recurrir a las líneas guía y a técnicas de iluminación y de encuadre modificando la apertura del diafragma.

Posicionamiento

Las imágenes inmersivas suelen mirarse estando sentado. Dado que no se trata de un formato de realidad aumentada o virtual, los espectadores no están de pie o moviéndose por el lugar. ¿Se encuentra sentado en este momento? A modo de experimento, gire la cabeza de lado a lado y de arriba abajo. ¿Hasta qué punto puede girar la cabeza con comodidad sin mover el cuerpo? El campo de visión del formato inmersivo supera el rango de rotación de la mayoría de las personas. Esto es bueno, dado que si se gira hacia la derecha o la izquierda, las imágenes aún le proporcionarán visión periférica. Este experimento lo ayudará a comprender dónde ubicar la acción principal dentro del campo de visión.

Rango de giro típico de la cabeza:

Rotación de 120° a 160°
Flexión y extensión de 110° a 130°.

Al combinar la rotación de la cabeza con el movimiento de los ojos, es posible mirar sobre el hombro; no obstante, mantener esta posición suele ser incómodo.

Para facilitar la composición, active la información en pantalla. Mantener la acción dentro de las marcas internas que muestran el campo de visión típico ayudará a que el contenido se pueda mirar cómodamente.

También debe tenerse en cuenta el movimiento de los puntos de interés dentro de un fotograma. Seguir a una persona mientras camina del extremo izquierdo al derecho requiere girar la cabeza considerablemente. Del mismo modo, pasar de una toma en la que el punto de interés está en el límite de la imagen a otra en la que este se encuentra en el extremo opuesto requiere mover la cabeza excesivamente. Considerar la composición cuidadosamente puede ser de utilidad para mantener la comodidad del público durante la comunicación creativa.

Independencia del público

Al grabar contenidos 2D tradicionales, es posible usar el encuadre, modificar los ajustes de la distancia focal y el objetivo a fin de dirigir la atención del espectador. Sin embargo, al trabajar con contenidos inmersivos, el campo de visión de 180° hace que dirigir la atención del público y excluir cierta información sea más complejo. Este formato brinda a la audiencia más independencia para explorar las imágenes, que es uno de los factores que hace que este tipo de contenidos sea más entretenido. Aún así, es posible guiar la atención del público mediante el uso meticuloso del sonido direccional, la iluminación, la composición y la puesta en escena.

Movimiento de cámara

Las cámaras inmersivas no tienen que mantenerse fijas, pero sí es preciso ser cauteloso, ya que algunos movimientos podrían resultar poco naturales o incomodar al espectador. La analogía del auto es útil para ayudar a comprender cómo hacer que el movimiento de la cámara sea más placentero. Un vehículo grande tiene masa e inercia, lo que hace que resista cambios repentinos de velocidad o dirección. Por lo tanto, se acelerará o desacelerará a lo largo del tiempo y se desplazará de una manera predecible y controlada. Tras haber dominado las tomas fijas y pasar a filmar escenas en movimiento, tener en cuenta los factores a continuación ayudará a crear un movimiento agradable.

- **Duración:** Dé tiempo al público para que se acostumbre al movimiento de la cámara y lo disfrute. Los movimientos breves son más proclives a desorientar al espectador.
- **Intensidad:** Los movimientos de baja intensidad son más fáciles de integrar en la edición de imágenes inmersivas. Es posible incluir algunos más enérgicos, sin embargo, se recomienda que su intensidad aumente o disminuya paulatinamente.
- **Anclaje visual:** Son las partes de la escena que se mantienen fijas. Una toma sin anclajes visuales puede resultar desorientadora. Los elementos en el primer plano, por ejemplo, el interior de un automóvil, o los objetos distantes, como el horizonte, ayudan a estabilizar las imágenes en movimiento.

Velocidad de transferencia de datos

Las tablas a continuación muestran las velocidades de transferencia de contenidos inmersivos en resolución 8K × 7K a 90 y 24 fotogramas por segundo.

Frecuencias de imagen máximas a 90 f/s con una velocidad de transferencia constante en Blackmagic RAW

8K × 7K inmersivo			
Códec	**Dimensión en pixeles**	**Frecuencia de imagen**	**Velocidad de datos**
Blackmagic RAW 5:1	8160 × 7200 estéreo	90	3203 MB/s
Blackmagic RAW 8:1	8160 × 7200 estéreo	90	2002 MB/s
Blackmagic RAW 12:1	8160 × 7200 estéreo	90	1335 MB/s
Blackmagic RAW 18:1	8160 × 7200 estéreo	90	889 MB/s

Frecuencias de imagen máximas a 24 f/s con una velocidad de transferencia constante en Blackmagic RAW

8K × 7K inmersivo			
Códec	**Dimensión en pixeles**	**Frecuencia de imagen**	**Velocidad de datos**
Blackmagic RAW 5:1	8160 × 7200 estéreo	24	854 MB/s
Blackmagic RAW 8:1	8160 × 7200 estéreo	24	534 MB/s
Blackmagic RAW 12:1	8160 × 7200 estéreo	24	356 MB/s
Blackmagic RAW 18:1	8160 × 7200 estéreo	24	237 MB/s

Audio y código de tiempo

El audio es una parte central de la experiencia inmersiva. Es importante combinar la cámara con equipamiento exclusivamente dedicado a la grabación de audio.

Formato Apple Spatial Audio

El formato Apple Spatial Audio, o ASAF, permite proporcionar sonido 3D al mirar contenidos en dispositivos Apple Immersive Video y forma parte del marco más amplio de Audio Posicional de Apple, por lo que permite almacenar tres tipos de grabaciones de sonido diferentes.

- **Grabación basada en el canal:** Es posible integrar mezclas de audio envolvente tradicionales en archivos ASAF. Esto resulta ideal para música y narraciones fijas en relación con el movimiento de la cabeza.
- **Grabación basada en el objeto:** Los sonidos pueden considerarse como objetos por separado con metadatos que describen su posición en el espacio 3D, independientemente de cada canal del altavoz, y suelen ser grabaciones monoaurales. Las coordenadas 3D se pueden animar de manera que sigan los objetos que se mueven dentro de una escena. Esto es ideal para el audio diegético, referido a los objetos que se ven en la escena.
- **Grabación ambisónica:** Un formato envolvente en 360° capaz de renderizar el audio en todas las direcciones mediante armónicos esféricos. Esto resulta ideal para el sonido ambiental.

A continuación, este apartado presenta una visión general del equipamiento que se necesitará para crear un archivo ASAF efectivo.

Sonido ambisónico

Un micrófono ambisónico contiene cuatro o más cápsulas dispuestas en un patrón geométrico preciso. Cada una de ellas capta los mismos sonidos con diferentes tiempos de llegada e intensidad, en función de la dirección de donde provenga. Estas cápsulas se graban en canales discretos para su posterior procesamiento en un equipo informático. Al comparar las diferencias, el programa puede estimar la dirección y la naturaleza de los sonidos grabados.



Micrófono ambisónico de primer orden.

Micrófonos ambisónicos

Los micrófonos ambisónicos se clasifican según el número de cápsulas que contienen. Cuanto mayor sea, mayor será la precisión con la que el micrófono puede reproducir los detalles espaciales.

Primer orden:	4 canales
Segundo orden:	9 canales
Tercer orden:	16 canales
Cuarto orden:	25 canales o más

Sin importar si el micrófono ambisónico se encuentra sobre la cámara o en un lateral, es fundamental alinearlo correctamente con la dirección en la que graba la cámara. Asimismo, dado que este tipo de micrófono se puede configurar de diferentes maneras —por ejemplo, en un extremo en la dirección del sonido, perpendicular a este, hacia arriba o hacia abajo—, es importante asegurarse de que el técnico de sonido sepa qué configuración se ha utilizado. La información se puede transmitir a través de los metadatos del grabador de sonido o mediante notas.

Las grabaciones ambisónicas se pueden utilizar como sonido ambiental. Esto permite superponer otras capas de efectos de sonido basados en el canal o en objetos. Asimismo, las grabaciones ambisónicas que están correctamente alineadas con la cámara también pueden servir de referencia para el ingeniero de sonido en el proceso de diseño de sonido.

Micrófonos de solapa y miniatura

Al rodar con una cámara de 180°, es imposible usar una jirafa ya que se vería en la escena. En consecuencia, es preciso recurrir a micrófonos escondidos y de solapa a fin de grabar el audio y el diálogo diegético.

Algunas opciones son:

- Conectar esas fuentes de audio directamente a las dos entradas XLR de la cámara mediante transmisores y recibidores inalámbricos.
- Conectarlas a un grabador de audio externo. Esto podría ser un dispositivo multicanal de gran tamaño o uno miniatura que se engancha a la ropa.

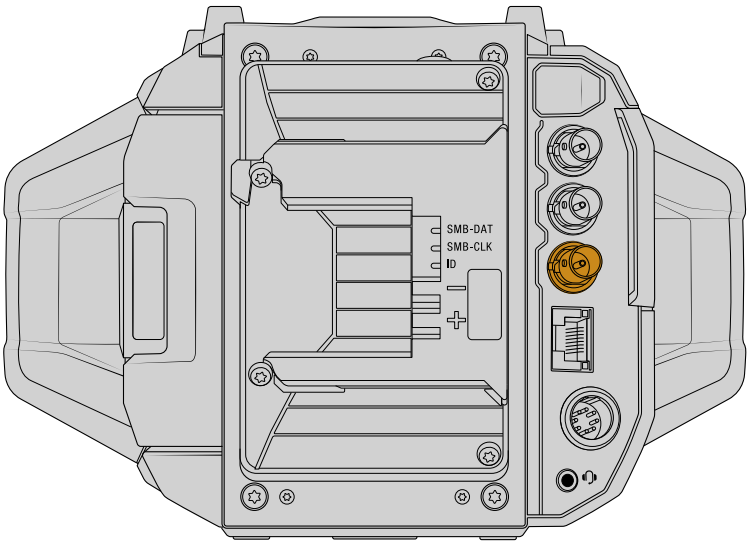
Posteriormente, estas grabaciones se pueden sincronizar y mezclar como sonidos basados en el canal o en objetos.

Micrófonos de mano

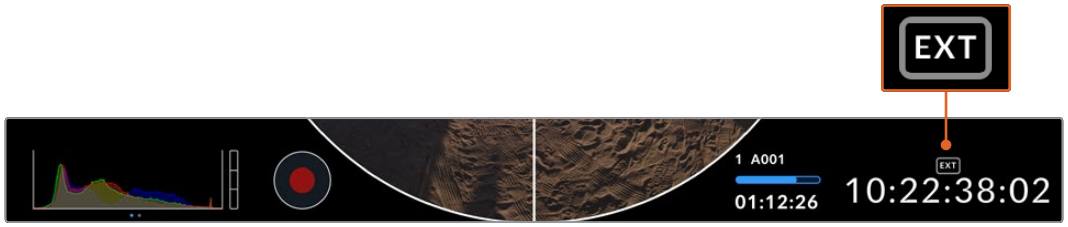
Grabar efectos de sonido adicionales en exteriores una vez finalizado el rodaje puede no solo ahorrar tiempo de posproducción, sino también dar como resultado un sonido más auténtico. Contar con un micrófono de mano hipercardioide o uno similar durante la producción es muy ventajoso, ya que brinda la posibilidad de volver a realizar las acciones acercando el micrófono. Luego, estas grabaciones se pueden editar e incorporar en la mezcla final como sonidos basados en el canal o en objetos.

Código de tiempo

Para usar un generador de código de tiempo externo con el modelo URSA Cine Immersive, conecte el dispositivo a la entrada correspondiente en el panel trasero de la cámara mediante un cable BNC. Sincronizar el código de tiempo al grabar con un sistema de dos cámaras garantiza que el audio y la imagen se puedan sincronizar con facilidad en la posproducción.



Al recibir una señal externa de un código de tiempo con la frecuencia de imagen asociada a través de este conector, la cámara se sincronizará automáticamente, y el indicador **EXT** aparecerá junto al reloj en la pantalla. Al desconectar el cable, dicho indicador cambiará a **INT**, a fin de informar al usuario que se utilizará el código de tiempo interno de la unidad.



Una vez conectada al generador de código de tiempo, el indicador **EXT** aparecerá junto a este en la cámara.

NOTA: Existen normas SMPTE de código de tiempo para frecuencias de imagen típicas, tales como 23.976, 24, 30, entre otras, pero no para aquellas de 90 fotogramas por segundo. Por lo tanto, al emplear una frecuencia de imagen de 90 fotogramas por segundo, la cámara utilizará el código de tiempo correspondiente a 30 f/s. La salida SDI también funciona a 30p. Al utilizar cualquier accesorio para el código de tiempo o grabadores de audio externos, seleccione una frecuencia de imagen de 30 fotogramas por segundo.

Grabación

Ahora ya tiene conocimientos básicos sobre cómo operar el modelo Blackmagic URSA Cine Immersive y sacar el máximo provecho del formato inmersivo. A continuación, este apartado explora su funcionamiento en mayor profundidad.

Requerimientos de exposición

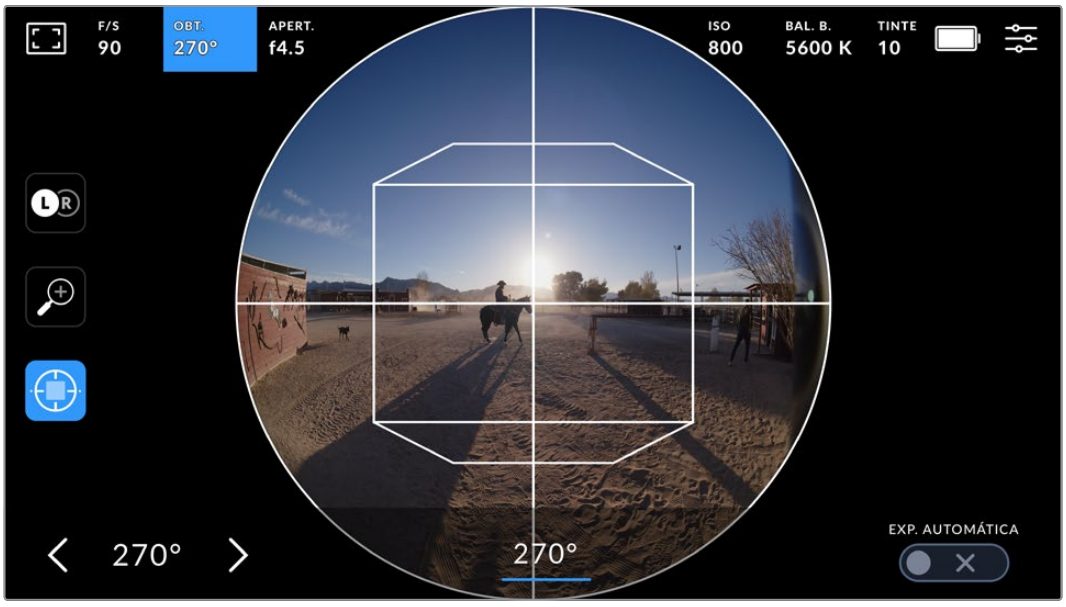
El formato Apple Immersive Video requiere de un valor de exposición más elevado que las imágenes tradicionales captadas a 24 f/s, por lo que puede que sea preciso reconsiderar el enfoque adoptado para grabación y la iluminación. Si el valor de obturación se mantiene en 180°, al grabar a 90 f/s, se necesitará de una cantidad de luz casi cuatro veces superior para exponer la misma cantidad de la imagen. Asimismo, los objetivos presentan un diafragma fijo de aproximadamente f4.5, de manera que es hasta tres pasos más oscuro que los objetivos Super Speed típicos. Esto significa que se podría necesitar un máximo de siete pasos más de luz para conseguir una exposición adecuada. No obstante, al rodar a 90 f/s, optar por un ángulo de obturación amplio de entre 270° y 360° puede ayudar a compensar la pérdida de luz resultante de las frecuencias de imagen altas y, a la vez, permitir conseguir un desenfoque por el movimiento placentero.

Velocidad de obturación

El indicador **OBT.** muestra el ángulo o la velocidad de obturación. Por su parte, la opción **Valor de obturación** en el menú de ajustes permite escoger cuál de estos parámetros muestra el indicador.

Para cambiar la velocidad de obturación:

1. Pulse **OBT.** en la barra de herramientas superior
2. Use las flechas izquierda o derecha para ver las opciones más comunes. Si desea seleccionar un valor de obturación específico, pulse dos veces el indicador situado en la esquina inferior izquierda de la pantalla. Se abrirá un teclado numérico que permite escribir cualquier valor con dos números decimales entre 11.2 y 360°.



Es posible escribir la frecuencia de imagen deseada mediante el teclado numérico digital.

La velocidad de obturación tiene un efecto más leve en la apariencia de los contenidos inmersivos captados a frecuencias de imagen altas. Si bien es preferible emplear un ángulo de obturación de 180° al rodar a 24 o 30p, al filmar en formato inmersivo, es más habitual ver que la velocidad de obturación se emplee para asistir con la exposición.

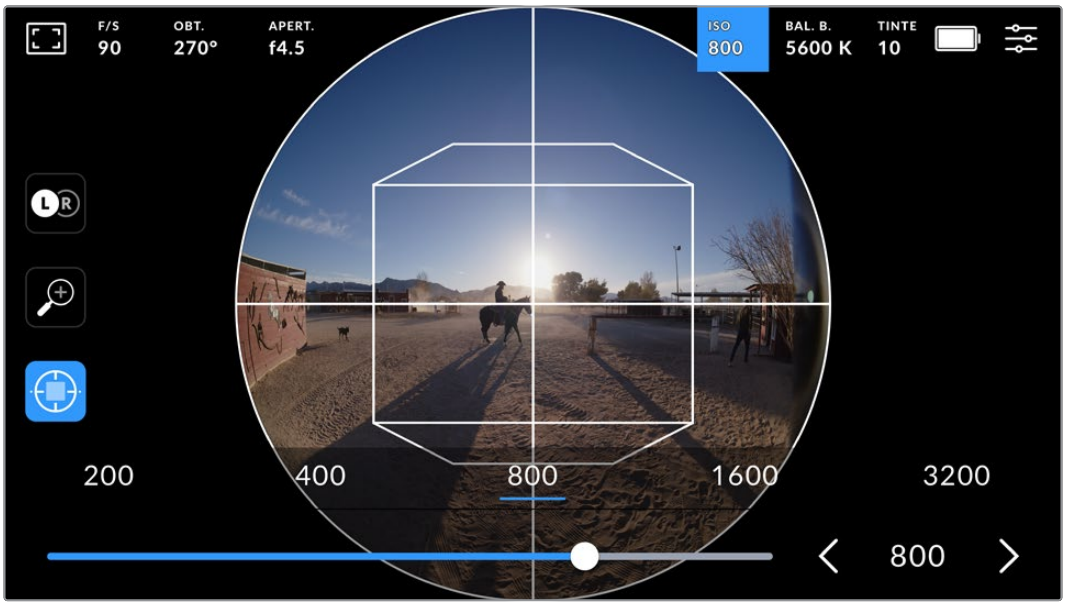
Al grabar con poca luminosidad, el valor de obturación se puede incrementar a 270° o, en circunstancias excepcionales, a 360°.

Por el contrario, en condiciones de luz brillante, este se puede reducir a un mínimo de 11.2°. Sin embargo, en este caso es más común configurar el obturador de la manera que se prefiera para tomas en movimiento y recurrir a filtros de densidad neutra a la hora de controlar la exposición.

Cabe señalar que la alta frecuencia de imagen de 90 f/s exigida por el formato Apple Immersive Video puede provocar problemas al grabar con luces parpadeantes. Se recomienda utilizar un valor de obturación de 270° al trabajar en entornos de 60 Hz y uno de 324°en ambientes de 50 Hz.

Sensibilidad ISO y rango dinámico

El indicador **ISO** muestra la sensibilidad fotométrica de la cámara. Al pulsarlo, es posible modificar dicho valor según las condiciones de iluminación. El modelo URSA Cine Immersive presenta un rango de sensibilidad ISO de 200 a 3200. El valor óptimo es de 800.



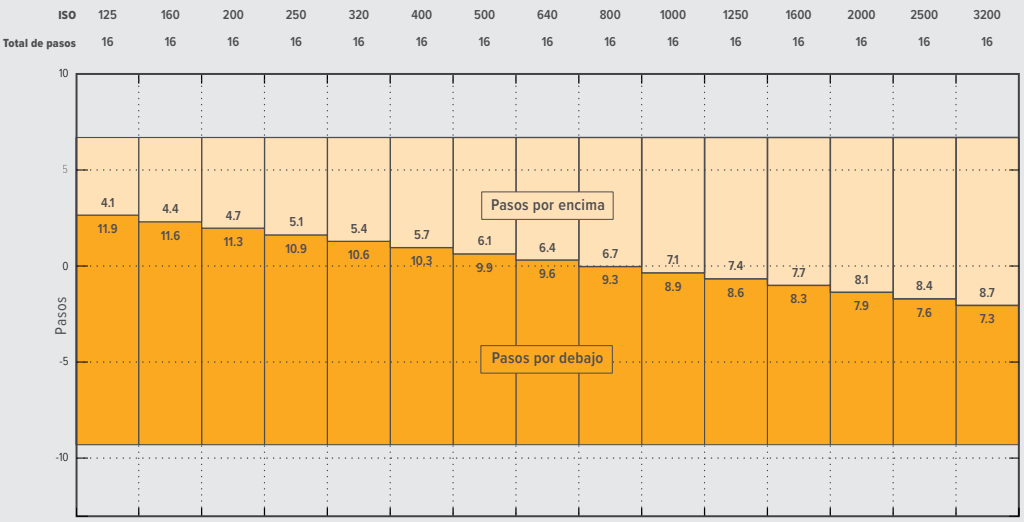
Este parámetro también se puede configurar pulsando el indicador **ISO**.

Los valores numéricos predeterminados brindan la opción de seleccionar un valor ISO en incrementos de pasos completos al pulsar el número correspondiente. El control deslizante que figura debajo de los ajustes predeterminados permite modificar el valor ISO en incrementos de 1/3.

En función de las circunstancias, es posible seleccionar distintos valores. Por ejemplo, un índice de 1600 podría resultar apropiado en condiciones de luz escasa, pero a su vez es posible que genere algo de ruido en la imagen. Al filmar con buena iluminación, se recomienda elegir un valor ISO de 200 para obtener colores más intensos.

Las cámaras Blackmagic son capaces de captar imágenes con un rango dinámico amplio, lo cual brinda una gran flexibilidad tanto al grabar como al etalonar el material en DaVinci Resolve. Este apartado incluye dos gráficas con las opciones de rango dinámico disponibles en las cámaras URSA Cine Immersive.

Rango dinámico en el modelo URSA Cine Immersive



Filtros de densidad neutra

Los filtros de densidad neutra (ND) se emplean comúnmente con el fin de controlar la exposición y permiten reducir la cantidad de luz captada por el sensor de la cámara, según el número de pasos predefinido.



En el modelo URSA Cine Immersive, es posible ajustar los filtros mediante los botones situados a la izquierda, sobre la pantalla táctil plegable. Las opciones disponibles son **CLEAR**, **2**, **4**, **6** y **8** pasos.

Los ajustes seleccionados actualmente pueden verse en el indicador situado en la esquina superior izquierda de la pantalla táctil. Al aplicar un filtro ND, este indicador permanece encendido. Por el contrario, al escoger la opción **CLEAR**, desaparecerá después de cuatro segundos.



NOTA: Es posible cambiar la terminología que utiliza el indicador de filtros de densidad neutra según las preferencias del usuario. Es posible alternar entre el número de filtro ND, los pasos y las fracciones desde el menú de preferencias.

Frecuencia independiente

El menú **PREFERENCIAS** muestra la frecuencia de imagen del proyecto, es decir la asignada al formato de grabación, y define la velocidad de reproducción de las imágenes. Normalmente, este valor se ajusta de manera que coincida con la velocidad de reproducción y la sincronización del audio empleadas durante la posproducción o según los requisitos de entrega del proyecto.



Pulse el indicador **F/S** para acceder a los ajustes de la frecuencia de imagen. Las ocho opciones disponibles son: 23.98, 24, 25, 29.97, 30, 50, 59.94, 60 y 90 f/s. Al rodar en formato inmersivo, este valor se debe establecer en 90 f/s.

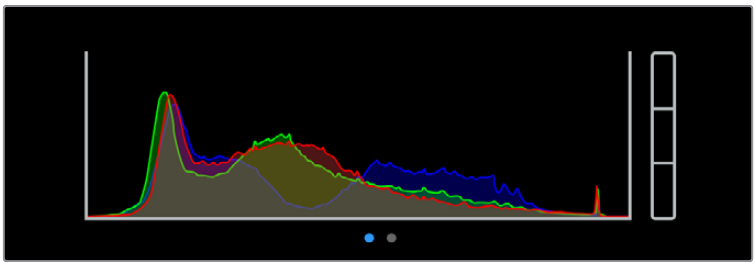
La frecuencia de imagen del sensor permite definir la velocidad a la cual se capta la imagen. Por defecto, la frecuencia de imagen del sensor y del proyecto coinciden, de modo que el material grabado se reproduzca a una velocidad normal. Sin embargo, al activar la opción **FREC. INDEPENDIENTE** en la región inferior derecha del menú **F/S**, es posible determinar valores diferentes para el sensor de manera independiente.

Cabe mencionar que 90 f/s es la frecuencia de imagen más alta que admite la cámara, y este valor coincidirá con el del proyecto al filmar imágenes para dispositivos Apple Vision Pro. No obstante, escoger uno menor, como 30 o 60 f/s, y luego grabar a una frecuencia independiente de 90 f/s, permitirá crear imágenes en cámara lenta atractivas.

Cuando la opción **FREC. INDEPENDIENTE** se encuentra activada, el indicador de frecuencia de imagen en la esquina superior izquierda de la pantalla mostrará la frecuencia de imagen correspondiente al sensor primero, y después la del proyecto.

Histograma

El histograma de la señal puede apreciarse en la parte inferior izquierda de la pantalla. Se trata de una gráfica que muestra la distribución del brillo en una imagen. Las zonas más oscuras de la imagen se representan a la izquierda, los tonos medios, en el centro, y las partes más brillantes, a la derecha. La altura de la gráfica indica qué porción de la imagen presenta ese nivel de brillo.



El histograma proporciona una indicación visual de la distribución tonal del clip. Una imagen que presenta un histograma sesgado hacia la izquierda posiblemente esté subexpuesta. De igual modo, una cuyo histograma muestra un sesgo hacia la derecha posiblemente esté sobreexpuesta. La gráfica siempre debe interpretarse dentro del contexto de la imagen, pero, en términos generales, si la exposición es correcta, el gráfico debería estar equilibrado y exhibir una curva de distribución más pronunciada en el centro. Al aumentar la exposición, la curva debería desplazarse hacia la derecha, y al reducirla, hacia la izquierda.

A la derecha del histograma hay un indicador que muestra cuándo se omite información en cada canal de color. Por lo regular, se debería tratar de evitar omitir información en cualquiera de los canales. Sin embargo, hay situaciones en las que es aceptable hacerlo para preservar los tonos medios y las sombras en la imagen.

La exposición a la derecha es una técnica en que la imagen se sobreexpone intencionalmente desplazando el histograma lo más posible a la derecha sin omitir datos en los canales de los colores. Esto maximiza la señal y minimiza el ruido. Siempre y cuando la exposición se reduzca en la posproducción, esta es una manera excelente de conseguir imágenes limpias.

Opciones de supervisión

Pulse el ícono del monitor, situado en la esquina superior izquierda de la pantalla, para acceder a las opciones de supervisión. Este menú contiene opciones útiles como el brillo del monitor, además de herramientas de exposición de gran utilidad.



Cebra

Esta opción permite activar o desactivar la función **Cebra** y ajustar su intensidad. Esta función muestra líneas diagonales sobre las zonas de la imagen que superan el límite de exposición predeterminado. El porcentaje indica las partes de la imagen que están sobreexpuestas. Es común establecer este valor en 95 %, en lugar de 100 %, dado que esto ayuda a identificar partes de la imagen en que la sobreexposición se puede corregir. El segundo ajuste de esta función brinda la posibilidad de ver las zonas grises intermedias por sí solas o con un paso adicional.

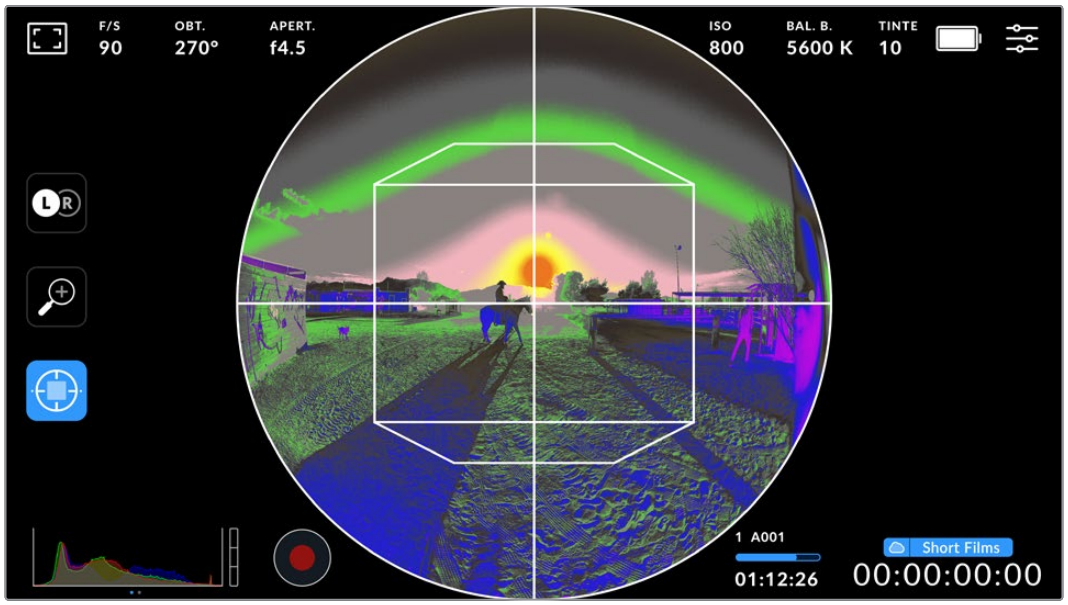
Cuadrículas

Esta opción permite ver una cuadrícula de 3×3 (regla de los tercios), una cruz filar, un horizonte virtual o un punto central sobre la imagen visualizada en la pantalla táctil o en los monitores conectados a la cámara a través de la salida SDI. Al grabar en formato inmersivo, el horizonte virtual resulta de gran utilidad, ya que muestra una cruz filar doble en el centro de la imagen, que se verá alineada si la cámara está completamente nivelada, o de lo contrario, se verán separadas.

Falso color

Esta herramienta superpone diferentes colores en la imagen para representar los valores de exposición correspondientes a diversos elementos. Por ejemplo, el rosa representa un nivel de exposición ideal para tonos de piel más claros, mientras que el verde es adecuado para matices más oscuros. Al observar dichos colores cuando se filma el rostro de una persona, es posible mantener una exposición constante en este elemento específico.

De manera similar, resulta sencillo identificar las áreas sobreexpuestas de la imagen cuando el color cambia de amarillo a rojo. Al igual que el histograma, esta herramienta brinda un entendimiento intuitivo y rápido de la distribución tonal de la imagen.



Grabación

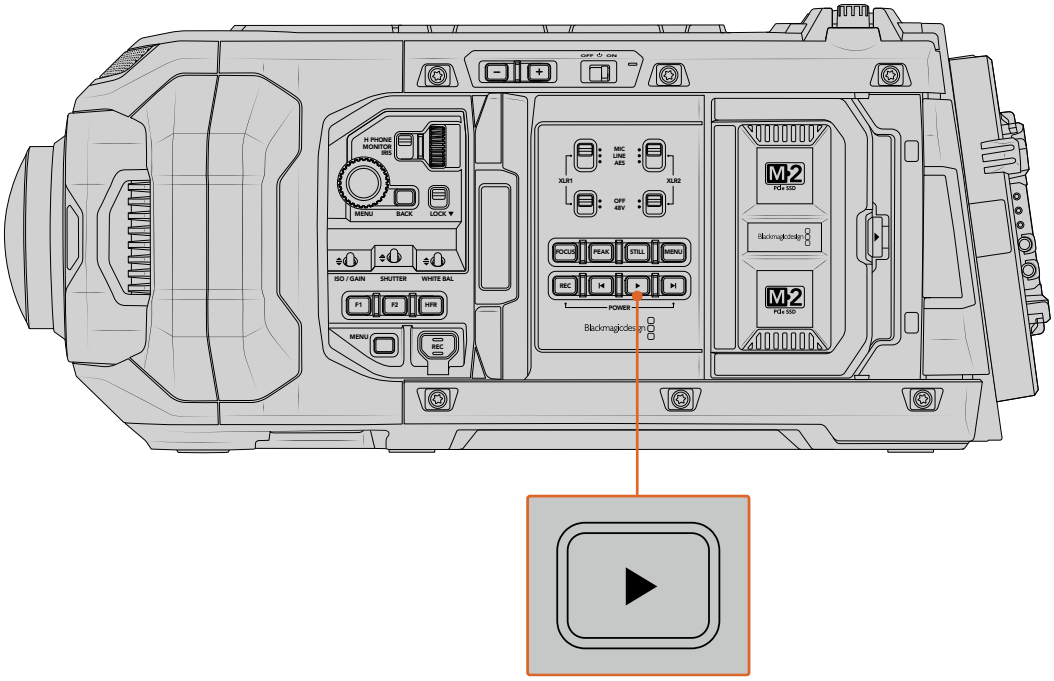
Para comenzar a grabar, simplemente presione uno de los botones de grabación situados en el panel de control delantero, la parte trasera de la pantalla táctil plegable o el módulo de asistencia, o bien pulse el ícono correspondiente en cualquiera de las pantallas LCD. Presione uno de estos botones nuevamente para finalizar.

Los indicadores de capacidad muestran el tiempo de grabación restante. El tiempo se muestra en horas:minutos:segundos, en función de la frecuencia de imagen seleccionada. Esta barra arriba del tiempo de grabación se destaca en azul o rojo según el estado e indica el espacio utilizado en la unidad de almacenamiento.



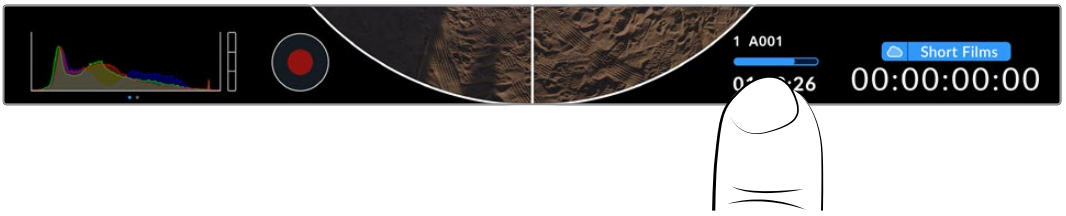
Reproducción

Después de realizar la grabación, es posible utilizar los controles de reproducción para ver las imágenes en la pantalla de la cámara. Para ver el último clip, oprima el botón de reproducción físico que se ubica en el lado izquierdo de la cámara, detrás de la pantalla plegable. Es posible utilizar los botones de avance y retroceso para ir a otros clips. Si desea salir del modo de reproducción y continuar grabando, presione el botón de grabación una vez.

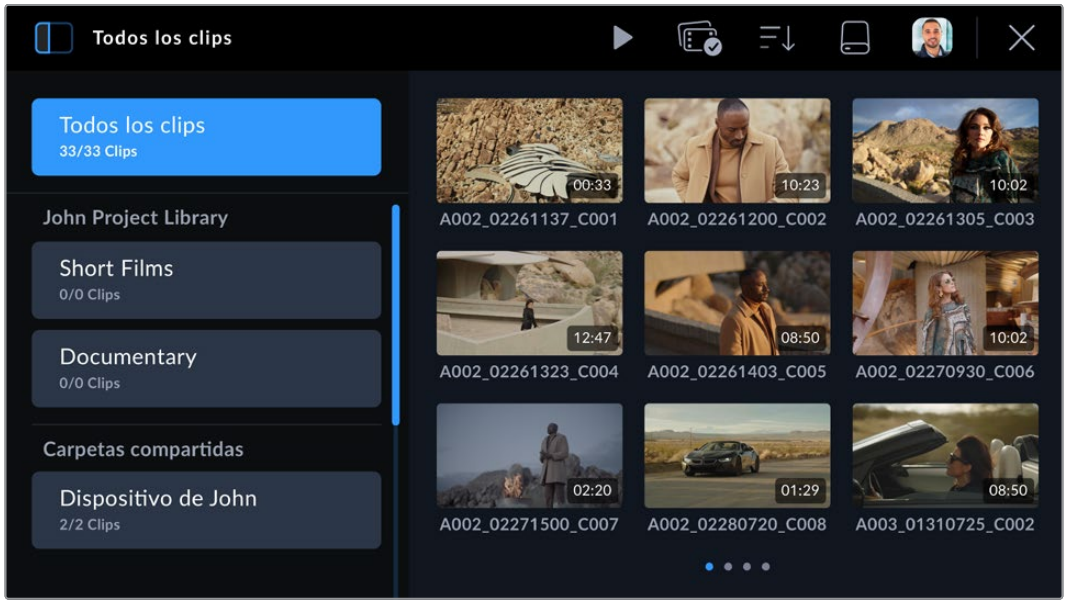


Presione los botones de avance o retroceso para ir al comienzo o al final de una secuencia. Oprima el botón de retroceso una vez para reproducir el clip actual desde el inicio, o dos veces para retroceder hasta el comienzo del clip anterior. Mantenga presionado el botón de avance o el de retroceso para adelantar o retroceder las imágenes al doble de velocidad. Presione estos botones 2, 3 o 4 veces para aumentar la velocidad 4, 8 o 16 veces, respectivamente.

Si se requiere una revisión más exhaustiva del material grabado, posiblemente sea preferible utilizar el panel multimedia. Para acceder a él, pulse uno de los indicadores correspondientes a los soportes de grabación en la parte inferior de la pantalla táctil.



La pantalla principal muestra miniaturas de todos los clips grabados en los soportes de almacenamiento conectados a la cámara. Es posible ver tres filas de miniaturas, pero, a medida que se agregan más clips, basta con desplazar el dedo sobre la pantalla táctil para acceder a los demás archivos. En la parte inferior de la pantalla se indican cuántos grupos de clips hay.



Conecte la cámara a una cuenta de Blackmagic Cloud y envíe las imágenes directamente a un proyecto de DaVinci Resolve.

Si desea salir del panel multimedia, pulse la **X** situada en la esquina superior derecha o presione una vez cualquiera de los botones de grabación de la cámara.

Transferencia de clips

Existen varias maneras de transferir los clips de la cámara a otras ubicaciones. Por ejemplo, es posible conectar la unidad a un equipo informático o a una red mediante un cable Ethernet 10G para copiar los archivos. De manera alternativa, inserte el módulo de almacenamiento en un dispositivo Blackmagic Media Dock conectado a la red o al equipo informático. Este método permite utilizar hasta tres módulos a la vez.

Al utilizar una red Ethernet, este último detectará el módulo al igual que una unidad Blackmagic Cloud Store.

El modelo URSA Cine utiliza el protocolo DHCP de manera predeterminada, y en consecuencia la red o el equipo informático asignarán automáticamente una dirección IP a la unidad.

Permitir el uso compartido de archivos

Al configurar la cámara por primera vez y conectarla a un equipo informático o una red Ethernet, la función para compartir archivos mediante el protocolo SMB estará desactivada. Esto se debe a que, a diferencia de lo que ocurre con los dispositivos Blackmagic Cloud Store, cuyo contenido probablemente se comparta con otros usuarios de inmediato, es posible que no se desee otorgar acceso enseguida a los archivos originales de la cámara después de un día de rodaje importante. Por tal motivo, dicha función debe ser activada por el usuario.

1. Para compartir archivos mediante el protocolo SMB, es necesario descargar e instalar el programa Blackmagic Camera Setup en el equipo informático. Este no solo permite actualizar la cámara, sino además modificar ciertos ajustes, y se encuentra disponible en nuestra página de soporte técnico.
2. Después de descargar e instalar el programa utilitario, conecte la cámara al equipo informático mediante el puerto USB situado en el panel superior de la unidad, cerca de la parte trasera. A continuación, ejecute el programa.

3 Para acceder a los ajustes, haga clic en la imagen de la cámara o en el ícono de configuración.

4 En la sección **Network Access**, marque la opción **File sharing (SMB)**. Haga clic en **Save** en la esquina inferior derecha para aplicar los ajustes.

5 Haga clic en el campo del enlace URL más abajo para que la dirección se complete automáticamente. Esta mostrará el nombre de la cámara como una ubicación de red local.

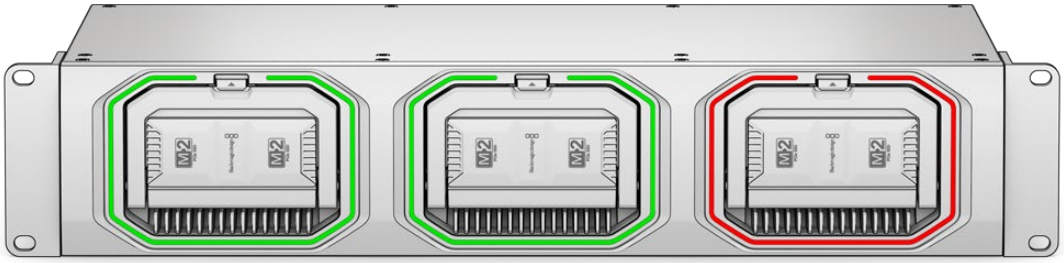
6 Haga clic en el botón **Save**.

Para acceder a los clips, haga clic en el enlace URL de la cámara. Se abrirá una ventana donde se le solicitará que introduzca una clave o se conecte como invitado. Marque la opción **Guest** y haga clic en el botón **Connect**.

7 El equipo informático mostrará el contenido del módulo de almacenamiento en la aplicación Finder (Mac) o el explorador de Windows. El programa utilitario recordará los ajustes de la red Ethernet, por lo cual solo es necesario llevar a cabo este procedimiento al configurar la cámara por primera vez. Luego, será posible acceder al contenido del módulo de inmediato sin necesidad de modificarlos.

Blackmagic Media Dock

El módulo ha sido diseñado para colocarse y quitarse rápidamente cuando es necesario transferir los clips a otra ubicación o cambiarlo por uno vacío para continuar la grabación. Además, puede utilizarse para acceder a las imágenes en una unidad Media Module sin requerir de la cámara.



Resumen

El formato Apple Immersive Video adopta un nuevo enfoque hacia las narraciones visuales y ofrece una alternativa al cine 2D tradicional. No se ha concebido como un reemplazo, sino como un formato adicional que se puede seleccionar en función de los requerimientos de cada proyecto. Al igual que en otros casos, las opciones creativas dependen de la historia que se quiera contar.

Si bien los archivos inmersivos implican dinámicas de trabajo complejas, el modelo URSA Cine Immersive ha sido diseñado para simplificar el proceso de grabación de imágenes en formato Apple Immersive Video. Esta cámara proporciona un sistema de trabajo sencillo para trabajar con contenidos inmersivos y también permite crear efectos visuales, entre otras cosas.

Esta guía delinea las características principales de la cámara URSA Cine Immersive y las dinámicas de trabajo compatibles. Esta unidad ha sido desarrollada para aportar un punto de partida. Al igual que con cualquier otro sistema de cámaras, alentamos a los usuarios a continuar explorando y llevado a cabo pruebas con ella con el fin de alcanzar los mejores resultados para cada proyecto.

DaVinci Resolve Studio



DaVinci Resolve Studio es un programa de posproducción integral que combina herramientas profesionales de edición, etalonaje, efectos visuales, animaciones gráficas y procesamiento de audio con gran eficiencia. Facilita dinámicas de trabajo adaptadas a las prestaciones únicas del formato Blackmagic RAW Immersive, tales como compatibilidad con imágenes estereoscópicas UHD, frecuencias de imagen de hasta 90 f/s y mezcla de audio espacial. DaVinci Resolve Studio está totalmente equipado para gestionar la supervisión, la producción y la entrega de material en formato Apple Immersive Video, lo que lo convierte en una aplicación sumamente atractiva para creadores de contenidos especializados en este tipo de archivos.

Recomendaciones para la terminal de trabajo

Soporte físico

- Equipo de escritorio Mac Studio M3 Ultra
- Monitor con tasa de refresco alta (mínimo 60 Hz)

Requisitos mínimos del sistema

- Sistema operativo macOS 15 o versiones posteriores
- 64 GB de memoria
- Blackmagic Design Desktop Video 12.9 o versiones posteriores (para supervisión)
- Equipo informático con chip Apple Silicon o GPU compatible con Metal

Requisitos de supervisión de video inmersivo

- Apple Vision Pro
- Conexión inalámbrica de alta velocidad (como mínimo Wi-Fi 5; se recomienda Wi-Fi 6)

Configuración del proyecto

Antes de importar los archivos RAW del modelo Blackmagic URSA Cine Immersive, es recomendable revisar la configuración del proyecto en DaVinci Resolve para asegurarse de procesar y aplicar la interpolación cromática al material correctamente, a fin de obtener un resultado visual óptimo y, al mismo tiempo, facilitar su edición.

Ajustes principales

Es fundamental indicar que se creará contenido para Apple Vision Pro al configurar un proyecto. Esto garantizará que todos los ajustes y las opciones correctas de la dinámica de trabajo estén disponibles mientras se prepara, arma y entrega un proyecto.

1. Abra DaVinci Resolve Studio.
2. En la esquina inferior derecha de la ventana **Organizador de proyectos**, haga clic en **Crear proyecto**.
3. Indique un nombre para el proyecto y la ubicación de los archivos multimedia (si es necesario), y haga clic en **Crear**.
4. Haga clic en el ícono del engranaje situado en la esquina inferior izquierda para abrir la ventana **Configuración del proyecto**.

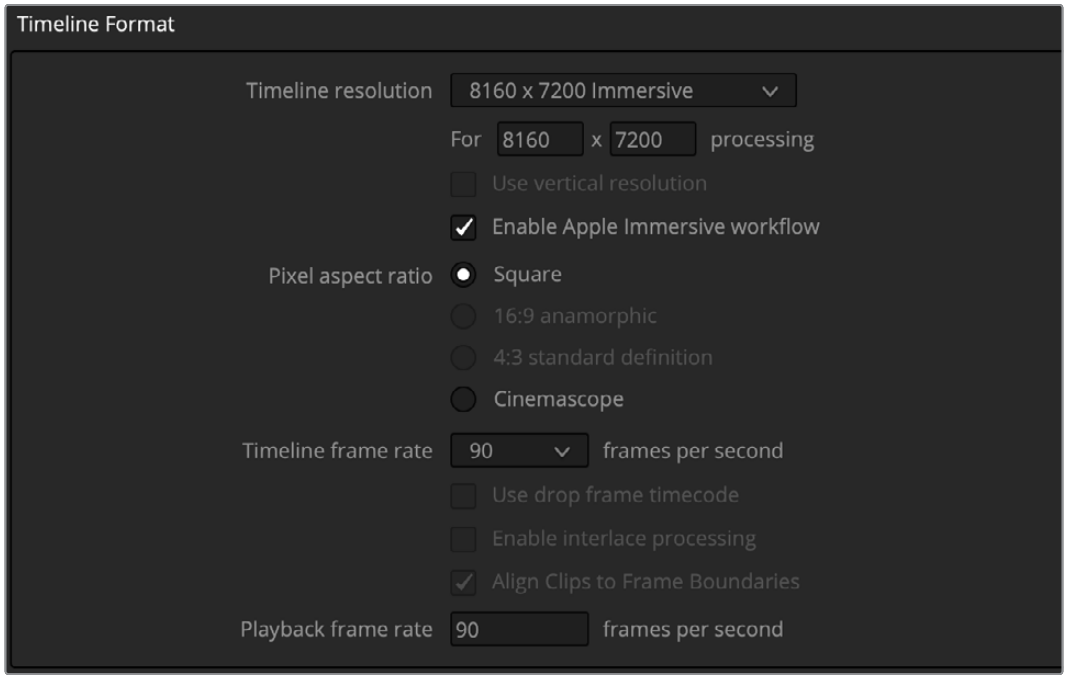
   Primero, es necesario establecer la resolución y la frecuencia de imagen del proyecto.
5. Seleccione el panel **Ajustes principales** en el lateral.
6. En la sección **Formato de la línea de tiempo**, marque la casilla **Activar dinámica Apple Immersive**.

   Esto configurará múltiples ajustes de DaVinci Resolve para adaptarse a las dinámicas de trabajo inmersivas de la mejor manera posible. Al crear un proyecto inmersivo, asegúrese de siempre marcar esta opción antes de empezar.
7. Cambie la opción **Resolución** a **8160 x 7200 Immersive** para procesar los clips en la resolución nativa del sensor.

   Si trabaja con recursos de procesamiento limitados, seleccione **4080 x 3600 Immersive** a fin de reducir la resolución a la mitad y agilizar la reproducción.
8. La opción **Frecuencia de imagen** debe configurarse en **90 fotogramas por segundo**, la cual se determina automáticamente al activar las dinámicas de trabajo Apple Immersive.
9. Ajuste la opción **Frecuencia de reproducción** a **90 fotogramas por segundo** para lograr la reproducción más precisa de los contenidos inmersivos.

   Si trabaja con recursos de procesamiento limitados (o un monitor con una frecuencia de refresco menor), reduzca la frecuencia de imagen para lograr una reproducción más estable. Se recomienda seleccionar una frecuencia que sea una fracción de la nativa, por ejemplo, 30 f/s o 60 f/s.



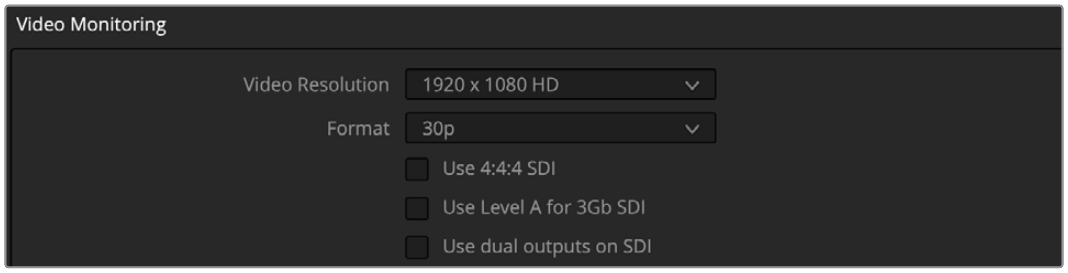
Monitorización de imágenes

Al utilizar un monitor externo (mediante un dispositivo de captura y reproducción como DeckLink o UltraStudio), configure los ajustes de supervisión de imágenes según las especificaciones del monitor.

1 En la sección **Monitorización de imágenes**, seleccione la opción **Resolución de video** que coincida con la de su monitor. Por ejemplo, **1920 x 1080 HD** o **3840 x 2160 Ultra HD**.

Aunque la frecuencia de imagen nativa del video inmersivo es de 90 f/s, la mayoría de los monitores no son compatibles con dicha frecuencia.

2 Configure la opción **Formato** para que coincida con la frecuencia de imagen de su monitor, por ejemplo, **30p**.



Redimensionamiento

Los ajustes del panel **Redimensionamiento** garantizan que la resolución y la relación de aspecto de los archivos inmersivos se muestren correctamente en el visor. Estos parámetros se configuran automáticamente al activar la opción correspondiente a dinámicas Apple Immersive en **Ajustes principales**, pero igualmente debería verificar que sean correctos.

1 Seleccione el panel **Redimensionamiento** en el lateral de la ventana **Configuración del proyecto**.

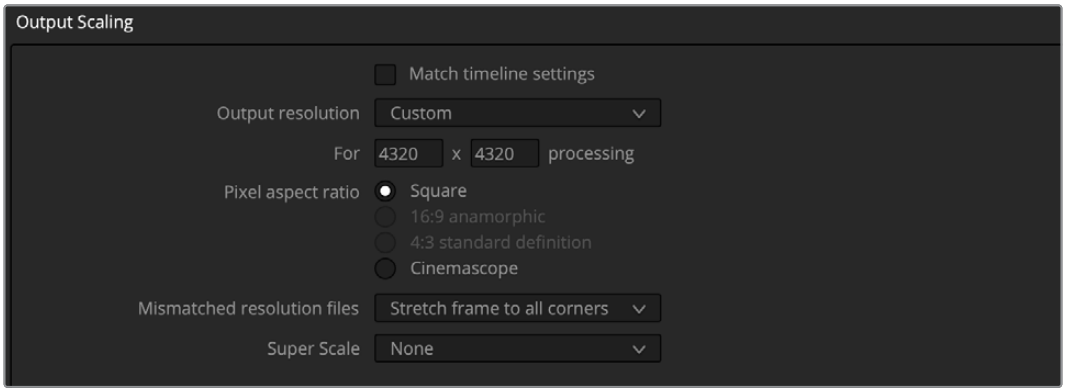
2 En la sección **Salida**, asegúrese de que la casilla **Ajustar según configuración de la línea de tiempo** no esté marcada.

3 Verifique que en **Resolución final** esté seleccionada la opción **Personalizada**, y luego **4320 x 4320**.

Reduzca la resolución a la mitad (**2160 x 2160**) para pantallas más pequeñas o una reproducción más rápida.

Estas resoluciones con relación de aspecto cuadrada son necesarias para reproducir contenidos Apple Immersive en DaVinci Resolve Studio y para transmitirlos al dispositivo Apple Vision Pro (mediante Wi-Fi o la correa para desarrolladores).

4 Asegúrese de elegir **Estirar toda la imagen** en la opción **Archivos con resolución diferente**.



Gestión cromática

Dado que es necesario contar con un control personalizado sobre la luminancia máxima de salida en nits, se recomienda utilizar la gestión cromática mediante nodos **Cambio del espacio cromático (CST)** en el módulo Color. El objetivo de los ajustes de gestión cromática en la ventana **Configuración del proyecto** es especificar el espacio cromático de la línea de tiempo (o *en uso*) en el módulo Color y establecer los metadatos del espacio cromático que se asociarán al archivo renderizado.

1 Seleccione el panel **Gestión cromática** en el lateral de la ventana **Configuración del proyecto**.

2 Asegúrese de elegir la opción **Gestión cromática YRGB (DaVinci)** en el menú **Colorimetría**.

3 Seleccione la opción **DaVinciWG/Intermediate** en **Espacio cromático de la línea de tiempo**.

Este es un amplio espectro cromático que garantiza su uso a futuro y que admite dinámicas de trabajo en HDR, lo que lo convierte en la opción ideal para el material inmersivo captado con el modelo Blackmagic URSA Cine y su posterior entrega para visores Apple Vision Pro.

4 Seleccione la opción **P3-D65 ST2084 1000 nit.** en **Espacio cromático final**.

Este espacio cromático incluye el espectro cromático adecuado para supervisar las imágenes en los visores de DaVinci Resolve en un equipo Mac Studio, que suele ser Display P3. Si utiliza un monitor de etalonaje o un perfil ICC de Apple diferente, deberá cambiar el espacio cromático final para que coincida con el de su pantalla.

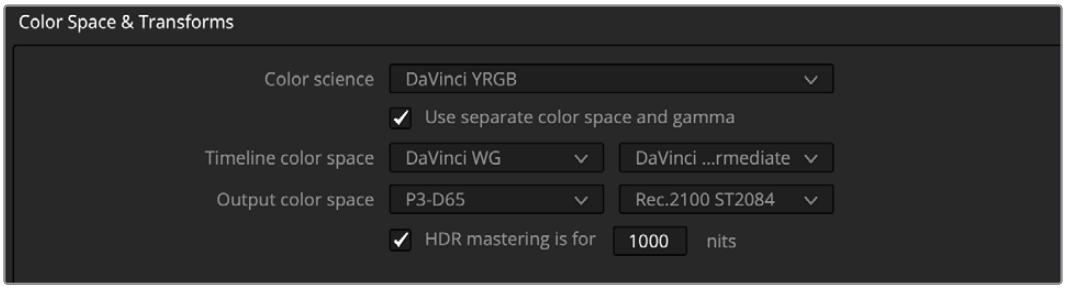
Sin embargo, deberá modificar la curva de respuesta tonal (función de transferencia) a fin de asignarla correctamente al perfil ICC.

5 Haga clic en la casilla **Selección independiente**, la cual se encuentra arriba de los ajustes de espacio cromático.

Esto permite dividir los campos desplegables del espacio cromático y la curva de respuesta tonal.

6 Seleccione la opción **Rec.2100 ST2084** para la curva de respuesta tonal final (en la columna derecha).

7 Marque la casilla **La masterización en HDR es para 1000 nits**.



Para destinar menos tiempo a la configuración del próximo proyecto Apple Immersive, es posible establecer los ajustes actuales como predeterminados.

8 En el menú de opciones (tres puntos) situado en la esquina superior derecha de la ventana **Configuración del proyecto**, seleccione **Establecer ajustes actuales como predeterminados**.

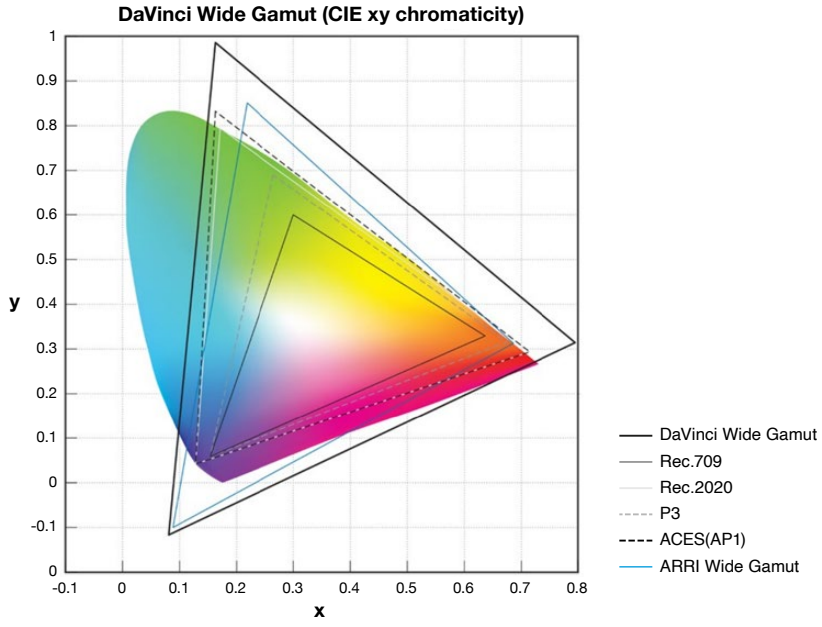
9 Asigne un nombre a la configuración y haga clic en **Aceptar**.

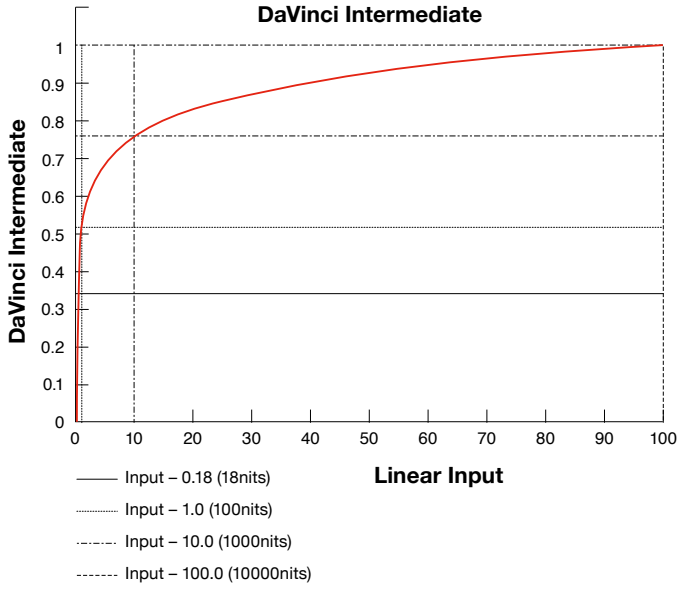
Para configurar rápidamente el proyecto la próxima vez que trabaje con contenidos en formato inmersivo, abra la ventana **Configuración del proyecto**, haga clic en el menú de opciones y seleccione la configuración predeterminada guardada.

10 Haga clic en **Guardar** para cerrar la ventana **Configuración del proyecto**.

Acerca de DaVinci Wide Gamut

DaVinci Wide Gamut es el espacio cromático interno avanzado de Blackmagic Design desarrollado para DaVinci Resolve, que ofrece un entorno excepcionalmente amplio y preciso para el etalonaje y el procesamiento de imágenes. Esto, sumado a la curva de respuesta tonal **DaVinci Intermediate**, crea un espacio de trabajo unificado que simplifica la gestión cromática en proyectos con formatos de cámara diversos, lo que garantiza una asignación consistente de tonos y saturación de principio a fin. Ideal para trabajar con material HDR e inmersivo, este espectro cromático amplio permite a los coloristas realizar ajustes detallados sin riesgo de degradado ni recortes tonales al momento de masterizar para formatos de entrega de alta fidelidad.





Procesamiento en formato Apple Immersive

En el lateral de la ventana **Configuración del proyecto**, la sección **Apple Immersive** permite personalizar la información del perfil de objetivos inmersivos (ILPD) para modificar cómo se calibra y proyecta el contenido inmersivo.

1. Seleccione el panel **Apple Immersive** en el lateral de la ventana **Configuración del proyecto**.
2. Haga clic en **Abrir carpeta de calibración**.
3. Agregue o actualice un archivo ILPD personalizado.
4. Haga clic en el botón **Actualizar listas** para actualizar los contenidos ILPD a los que se puede acceder desde el panel multimedia.
5. Haga clic en **Guardar** para cerrar la ventana **Configuración del proyecto**.
6. En el panel multimedia, haga clic con el botón derecho en un clip inmersivo.
7. En el menú contextual, seleccione **Calibración Apple Immersive** y elija entre el archivo ILPD de origen (**No se ha seleccionado ningún archivo de calibración Apple Immersive**), el ILPD personalizado o el ILPD genérico.

Por defecto, la carpeta **Calibración ILPD** personalizada se encuentra en la carpeta **ProIM**, junto a la de LUT predeterminadas.

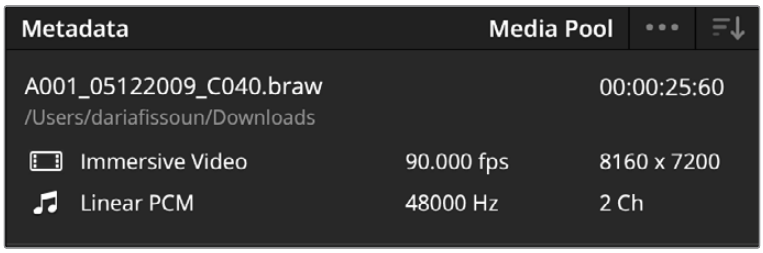
Ingesta

Al realizar una ingesta de contenidos inmersivos en formato RAW, ambos ojos y todos los metadatos asociados se incluyen en el clip .braw importado.

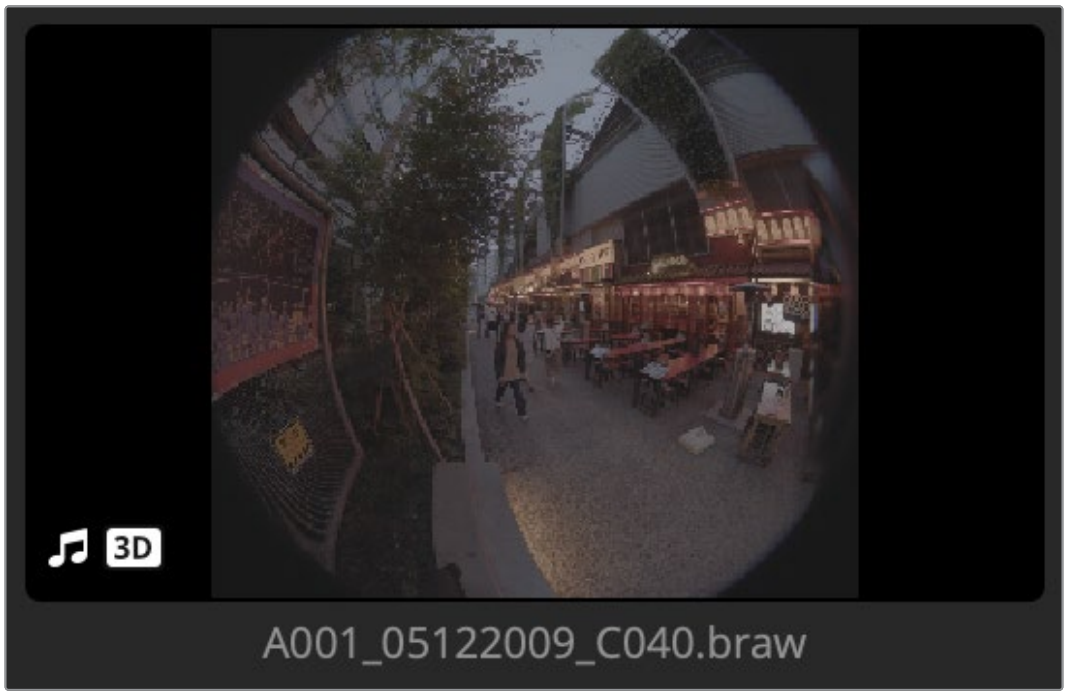
1. Abra el módulo Medios.
2. En el panel **Explorador multimedia**, abra la carpeta donde se encuentra el contenido inmersivo.
3. Arrastre los archivos y suéltelos en el panel multimedia.

SUGERENCIA: Para conservar la estructura de carpetas original, arrástrelas desde el explorador multimedia hasta el panel multimedia en el lateral. Al moverlas a la bandeja principal (o cualquiera que haya creado), se preservan las carpetas y subcarpetas.

Cuando se selecciona un clip de video Apple Immersive, el parámetro **Códec** en el encabezado del panel **Metadatos** indicará que se trata de un video inmersivo, junto con la resolución y la frecuencia de imagen nativas.



Además, se incluye un ícono **3D** en la esquina inferior izquierda de la miniatura, lo que confirma que es un clip 3D.



NOTA: Al arrastrar y soltar archivos inmersivos en el panel multimedia por primera vez, es posible que aparezca una ventana de diálogo que pregunte si desea conservar los ajustes o cambiarlos. Si opta por cambiarlos, la configuración del proyecto se ajustará automáticamente para adoptar la frecuencia de imagen y la resolución del material original.

Edición

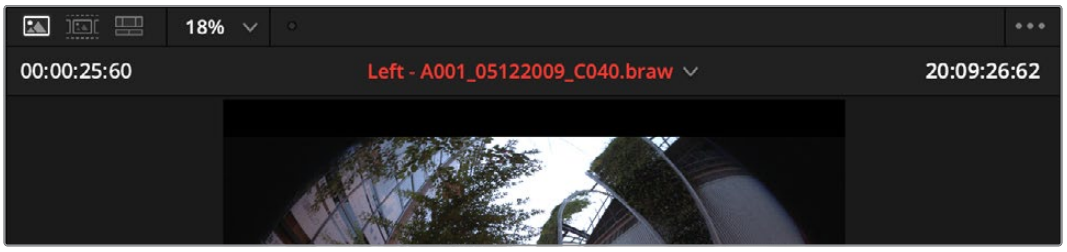
Al trabajar con contenidos inmersivos, las herramientas y acciones de edición son prácticamente las mismas que al trabajar con formatos 2D convencionales. Las siguientes instrucciones describen las funciones o acciones de edición exclusivas de las dinámicas de trabajo de material inmersivo.

Montaje

Al previsualizar clips durante el montaje en la línea de tiempo, es posible cambiar qué ojo (izquierdo o derecho) se muestra en el visor de medios originales. A excepción de esto, el proceso de montaje en la línea de tiempo no difiere de la dinámica de trabajo habitual en un programa de edición no lineal estándar.

1 Abra el módulo Edición.

2 Haga doble clic en un clip para abrirlo en el visor de medios originales a la izquierda. En la parte superior, se indica qué ojo se está visualizando.



3 Para cambiar el ojo en el visor, haga clic con el botón derecho en el clip original dentro del panel multimedia y seleccione **Estereoscópico 3D** > **Modo estereoscópico 3D** > **Izquierdo**, **Derecho** o **Ambos ojos**.

SUGERENCIA: También es posible desplazar el cursor sobre las miniaturas en el panel multimedia para ver los clips en el visor de medios originales, siempre que la opción **Previsualizar contenidos** esté activada en las opciones correspondientes.

4 Utilice las teclas **I** y **O** para definir un intervalo en el visor de medios originales.

5 Presione **Insertar clip** en la barra de herramientas (o la tecla **F9**) para agregar clips a la línea de tiempo.

SUGERENCIA: También es posible arrastrar clips desde el panel multimedia hasta la línea de tiempo.

6 Utilice el mouse (en modo de selección o recorte) para recortar clips en la línea de tiempo.

Fondos

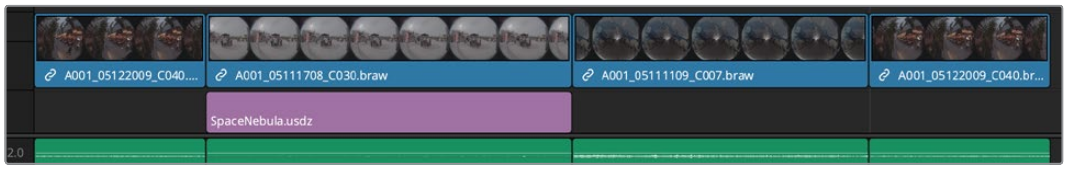
Al marcar la casilla **Activar dinámica Apple Immersive** en la ventana **Configuración del proyecto**, aparece automáticamente una pista de fondos (**BD**) debajo de la pista **Video 1**.

Los fondos corresponden al área que el espectador ve cuando mira más allá del perímetro del contenido inmersivo, el cual generalmente está limitado a un campo de visión de 180–210 grados.

Por defecto, esta área es negra, pero es posible insertar imágenes de fondo personalizadas (en formato .usdz) desde la pista de fondos.

1 Arrastre un archivo .usdz a la pista para agregarlo a la línea de tiempo.

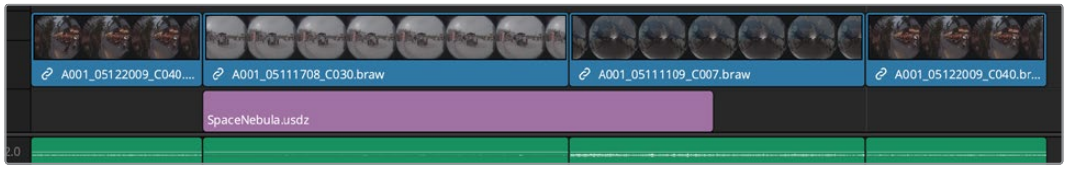
La duración de las imágenes de fondo debe coincidir con los límites de los clips en las pistas de video. Un único archivo de fondo puede abarcar un clip, varios o toda la línea de tiempo, pero no puede interrumpirse en medio de un clip inmersivo .braw (también conocido como *corte L*).



Un clip – Correcto ✓

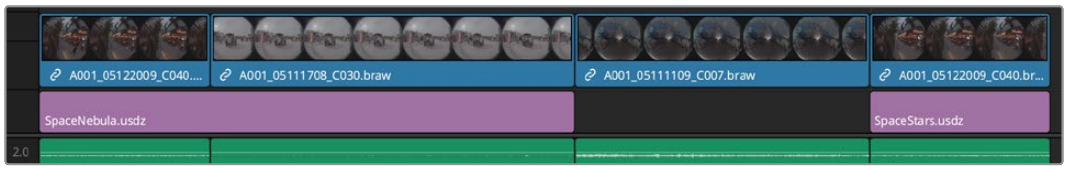


Múltiples clips – Correcto ✓



Corte L – Incorrecto ✗

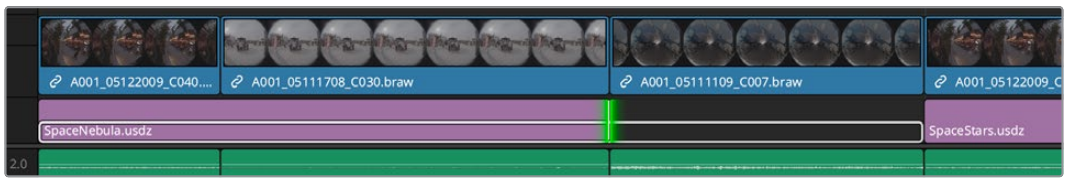
Es posible optar por incluir diversos archivos de fondo dentro de una misma línea de tiempo o no agregar ninguna imagen para mantener el fondo negro predeterminado.



Diferentes imágenes y espacios vacíos – Correcto ✓

2 Utilice los controles de recorte estándar para ajustar la duración del archivo de fondo.

Arrastre el margen del archivo .usdz (en modo de selección o de recorte) para modificar su longitud.



También es posible usar la coma (,) y el punto (.) para desplazar el margen un fotograma a la vez.

SUGERENCIA: Para asegurarse de que los fondos siempre estén alineados con las ediciones de archivos Immersive Video, active el modo **Atraer cabezal** en la barra de herramientas del módulo Edición.

Uso del visor

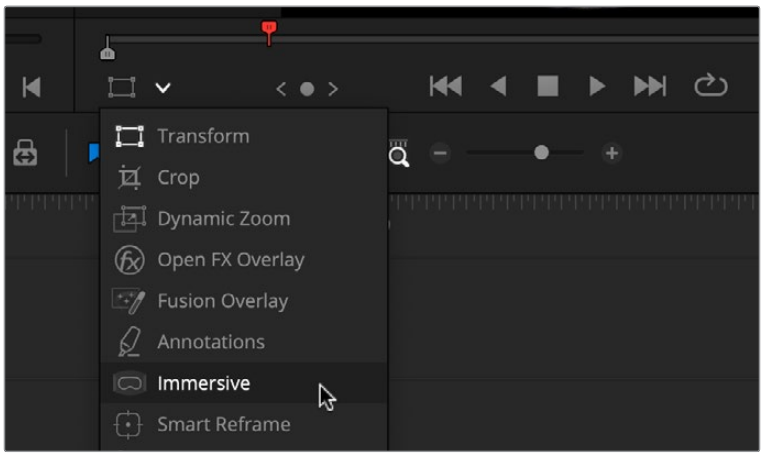
Por defecto, los clips inmersivos aparecen en los visores de los módulos de DaVinci Resolve dentro del *espacio circular del objetivo*. La calibración visual del objetivo utilizada para proyectar la imagen proviene de la cámara y es única para cada clip.



También existe la opción de cambiar el modo de visualización a **LatLong** o **Viewport**.

1 En la esquina inferior izquierda del visor de la línea de tiempo, haga clic en la flecha desplegable junto al botón de controles en pantalla.

2 Seleccione la opción **Inmersivo**.



Aparece una nueva barra de herramientas encima del visor con opciones adicionales.

3 Haga clic en **LatLong** para ver una proyección equirrectangular de la imagen. Este modo distorsiona la imagen para que se asemeje más a las proporciones del material 2D tradicional, lo que facilita ciertos procesos de posproducción, como el seguimiento y la composición.



4 Haga clic en **Viewport** para ver un recorte central de la escena inmersiva en un encuadre cuadrado. Este modo permite observar con claridad los detalles de la escena con las proporciones correctas en un monitor común, aunque es necesario asegurarse de no pasar por alto lo que ocurra fuera del encuadre.

Puede utilizar los controles de transformación en la barra de herramientas correspondiente para inclinar y desplazar la imagen estereoscópica dentro del encuadre, así como para ajustar los campos de visión horizontal y vertical.



Transiciones

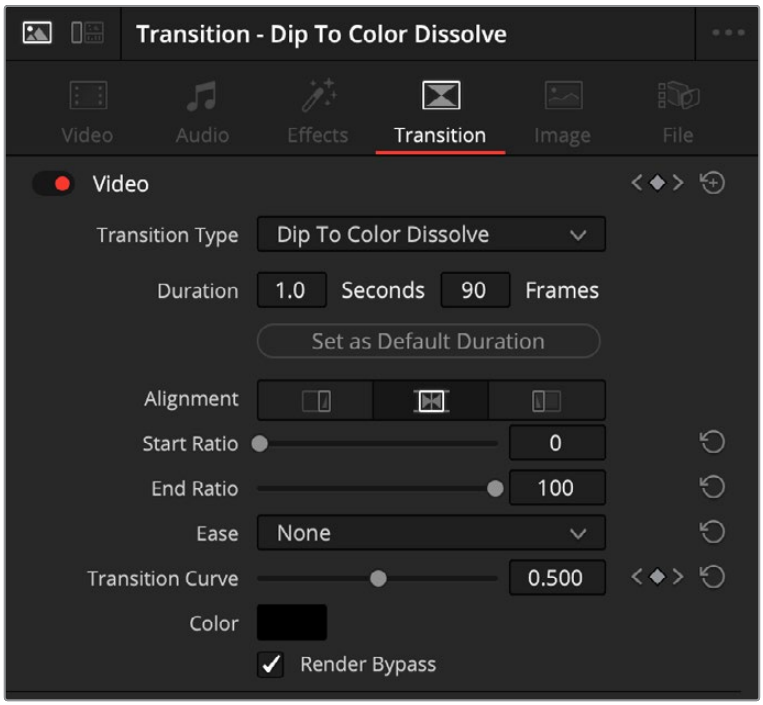
Para realizar una transición entre dos clips de video inmersivo, puede realizar un corte directo, usar los márgenes de edición del clip para aplicar un fundido inicial o uno final, o emplear el efecto **Fundido a un color**. Las transiciones no deben integrarse en el material, sino que deberían ser procesadas por el dispositivo Apple Vision Pro durante su reproducción.

1. Abra el panel **Efectos**.
2. En el lateral, abra la sección **Herramientas** y seleccione **Transiciones de video**.
3. Arrastre el efecto **Fundido a un color** hacia un punto de corte entre dos clips.



NOTA: Es necesario contar con suficiente material adicional más allá de los puntos de corte de ambos clips para aplicar una transición. Si no hay suficiente material (*márgenes de edición*), la transición no se aplicará.

4. En el inspector, abra la pestaña **Transición** y marque la casilla **Omitir en la renderización**. Esto garantiza que la disolvencia sea procesada correctamente por el dispositivo Apple Vision Pro.
5. Luego, es posible utilizar los otros parámetros en el inspector del efecto **Fundido a un color** para modificar la duración, la alineación, la velocidad de la animación y el color de la transición.

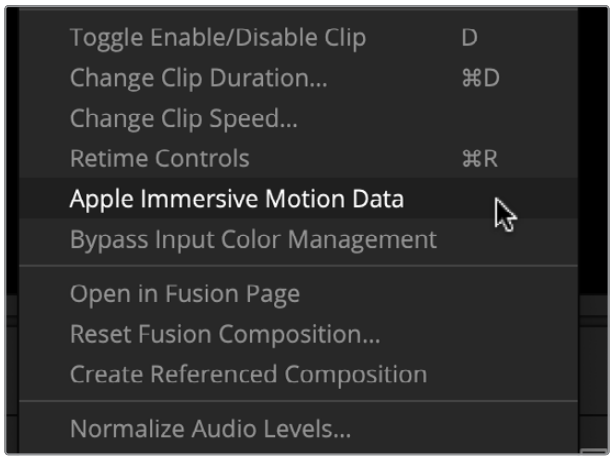


Datos de movimiento inmersivos de Apple

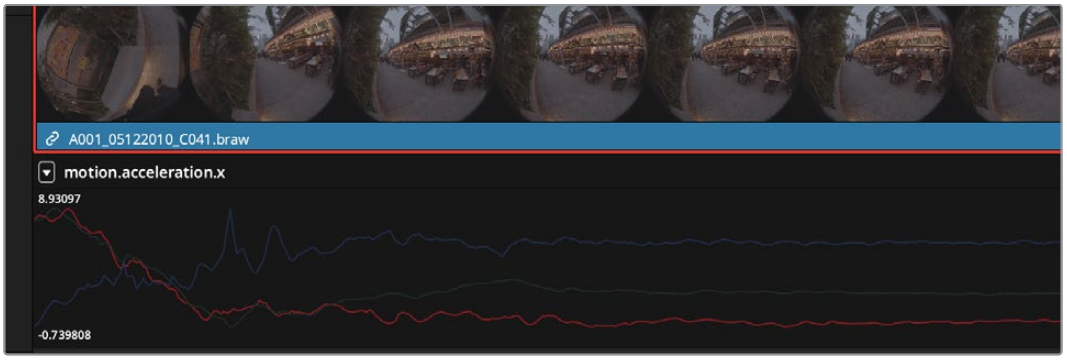
Al trabajar con contenidos inmersivos, los movimientos rápidos o repentinos de la cámara pueden desorientar al espectador. Con miras a garantizar una experiencia agradable, evite utilizar tomas con desplazamientos horizontales, verticales o rotaciones, e incluso aquellas de horizontes desnivelados.

La función Apple Immersive Motion Data muestra una gráfica de los valores de la aceleración (X, Y, Z) y de la posición del giroscopio a fin de indicar objetivamente los lugares donde el desplazamiento y el movimiento de la cámara podrían ser demasiado pronunciados. Esta información resulta útil para los editores cuando se desea evitar el uso de partes de secuencias que puedan desorientar al espectador.

1. Haga clic con el botón derecho del mouse en un clip inmersivo en la línea de tiempo.
2. Seleccione **Apple Immersive Motion Data**.

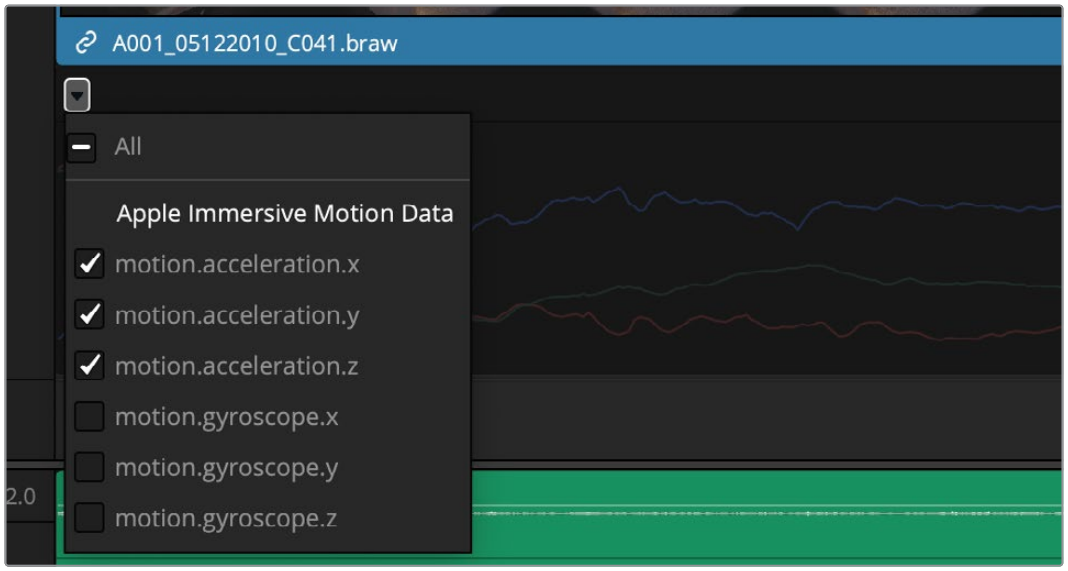


Aparecerá una gráfica en la pista de video con las tasas de aceleración X, Y y Z.



En este ejemplo, la cámara estaba grabando mientras el operador la estaba montando. Las puntas en la gráfica indican un movimiento brusco producido al alzar y rotar la cámara o al colocársela sobre el hombro. Idealmente, es preferible usar la última parte de la secuencia que presente valores similares sin extremos y se haya captado después de haber estabilizado la cámara.

3 En la esquina superior izquierda de la gráfica, haga clic en la flecha de desplazamiento.



Es posible activar o desactivar la lectura de datos en función del contenido de la toma y de la naturaleza del movimiento de la cámara.

Títulos

En el módulo Edición, es posible añadir texto a una escena inmersiva mediante cualquiera de las herramientas de títulos que se encuentran en el panel **Efectos**, por ejemplo, **Texto**, **Texto+**, **MultiTexto**, **Desplazar**, o cualquiera de las opciones para el tercio inferior.

Al ubicar el generador de títulos sobre un clip inmersivo, este utilizará los datos del perfil inmersivo del objetivo (ILPD) asociados al clip con el fin de distorsionar el texto con precisión, de manera que coincida con la distorsión del objetivo presente en la escena.

En caso de no disponer de datos ILPD, por ejemplo al añadir texto sobre un fondo negro o una escena inmersiva con imágenes computarizadas, se aplicará la distorsión predeterminada del objetivo para garantizar que el texto aparezca con la distorsión correcta al usar dispositivos Apple Vision Pro.

1 En el panel **Efectos**, haga clic en **Herramientas** > **Títulos**.

2 Arrastre el generador de texto a la línea de tiempo. Colóquelo en una pista de video sobre un clip inmersivo para incorporarlo en la escena.



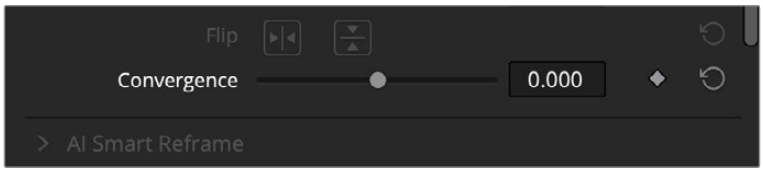
Al igual que al usar fondos, los generadores de texto no pueden superar los límites del clip. Dado que cada clip se renderiza con una perspectiva determinada por el objetivo, todo título superpuesto corre el riesgo de sufrir una distorsión entre tomas.

3 Recorte el título al punto de edición en el video.

4 En el inspector, escriba el título y aplíquele el formato de texto que desee.

5 El control deslizante **Convergencia** en la pestaña **Ajustes** permite posicionar el texto en el espacio 3D.

Muévalo hacia la izquierda para alejar el título o hacia la derecha para acercarlo.



Este efecto se visualiza mejor al supervisar las imágenes en dispositivos Apple Vision Pro. Si se emplean otros medios de supervisión, es posible modificar cómo se muestra este efecto en el visor, cambiando el modo de visualización de la capa de video de fondo en el módulo Color.

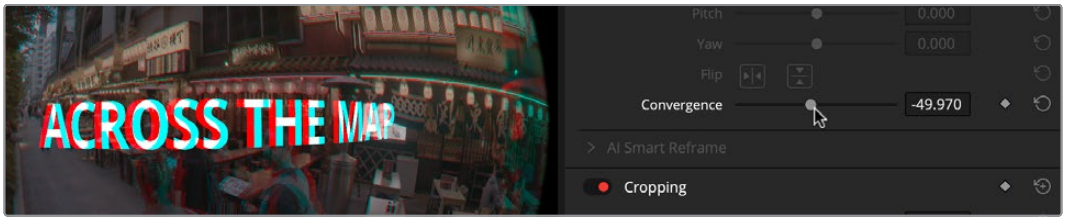
6 Abra el módulo Color.

7 Abra el panel **Imágenes estereoscópicas**.

La columna **Visión** en el centro del panel permite cambiar el formato estereoscópico en que se proyectan los archivos inmersivos.

8 Seleccione la pestaña **Estéreo**.

9 Cambie el modo de proyección a **Anaglifo (Color)**.



Ahora las modificaciones al parámetro de convergencia del generador de texto en el inspector del módulo Edición son más intuitivas al usar gafas anaglifas o anacromáticas.

Consideraciones creativas

Durante estas etapas iniciales de producción de contenidos con el modelo Blackmagic URSA Cine Immersive y dispositivos Apple Vision Pro, se alienta a los cineastas a experimentar con el formato con el objetivo de identificar las mejores prácticas, establecer nuevas guías y descubrir dinámicas de trabajo anteriormente impensadas para el cine tradicional.

A tal fin, se han señalado algunos aspectos de la producción inmersiva para su especial consideración, a fin de garantizar la comodidad visual y que el espectador quede completamente sumergido en la experiencia.

1 La edición debería comenzar en la etapa de preproducción. Dado que es preciso prestar particular atención a la longitud de las tomas y la comodidad del espectador ante el movimiento, los cineastas deben tener en consideración la edición final durante el rodaje.

2 Evite o límite el uso de clips con movimientos de cámara rápidos, ya que pueden desorientar al espectador. En cambio, opte por planos fijos o paneos de cámara lentos, o bien utilice un travelín.

3 Los ángulos de cámara desequilibrados o extremos también pueden ser desorientadores. Se recomienda elegir tomas que estén al mismo nivel del sujeto y a una distancia cómoda, es decir, no demasiado cerca.

4 Al mirar imágenes inmersivas, el espectador tarda más en asimilar un nuevo entorno en comparación con las escenas 2D. Los cambios de plano deben ser más lentos y brindar más tiempo para adaptarse al nuevo entorno.

5 Al hacer cortes entre clips, tenga en cuenta dónde habría estado mirando el espectador en las tomas anteriores. Si la atención estaba dirigida a la derecha, no deberían ocurrir hechos importantes a la izquierda del fotograma.

6 El audio espacial tiene gran relevancia en la narrativa, considérelo como otro personaje en las producciones inmersivas.

Composiciones con efectos visuales (VFX)

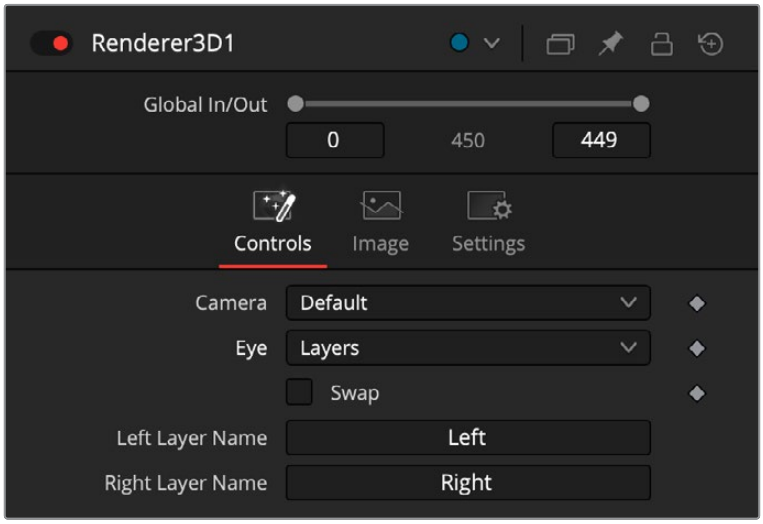
El módulo Fusion dispone de parámetros exclusivos para trabajar fluidamente con archivos estereoscópicos y realizar composiciones inmersivas. Para acceder a ellos, marque la casilla **Activar dinámica Apple Immersive** en la ventana **Configuración del proyecto**. Los nodos **MediaOut** ofrecen opciones de visualización para el ojo derecho y el izquierdo, mientras que la herramienta **Parche Inmersivo** brinda la posibilidad de efectuar proyecciones y composiciones equirrectangulares.

Nodos MediaOut

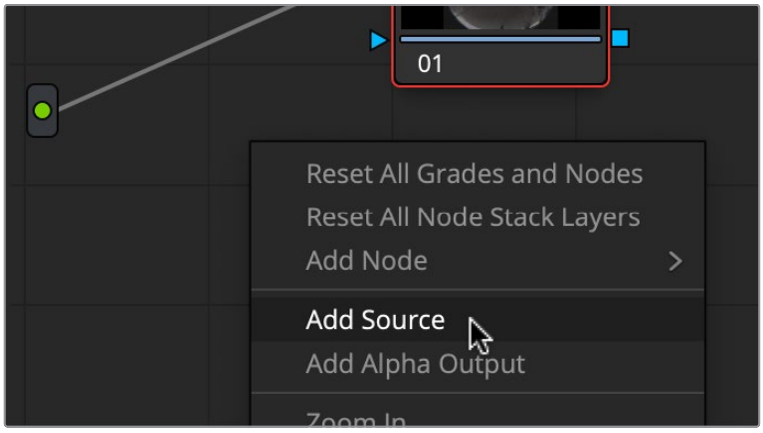
Al abrir el módulo Fusion por primera vez para trabajar con un clip inmersivo, la estructura nodal mostrará dos nodos **MediaIn** conectados a los dos nodos **MediaOut** correspondientes.



En el inspector del nodo **MediaOut**, el parámetro **Stereo Eye** permite asignar la señal para cada ojo.



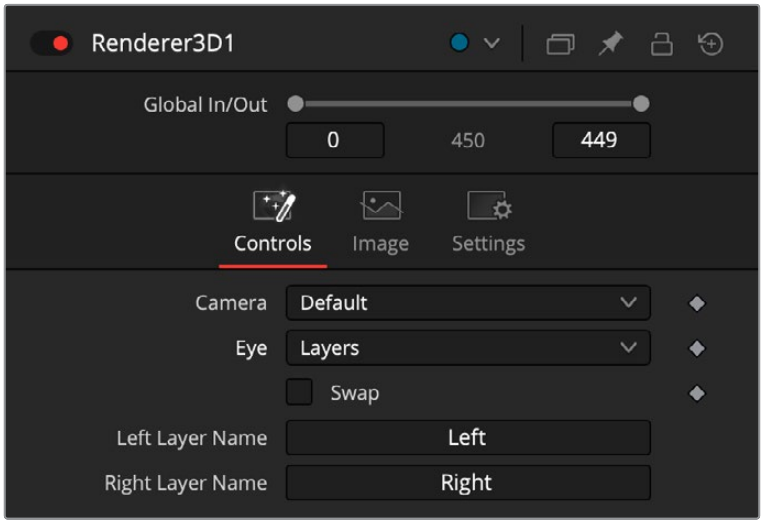
Por defecto, el nodo **MediaOut1** está configurado para el ojo izquierdo, y el **MediaOut2** para el derecho. Esto permite comenzar a componer imágenes estereoscópicas y superponer gráficos de inmediato. Al usar estos dos nodos, también se envían dos variantes de las señales al editor de nodos del módulo Color para continuar perfeccionando la imagen. Para acceder a la señal del nodo **MediaOut2** desde el módulo Color, haga clic con el botón derecho en el editor de nodos y elija **Agregar inicio de estructura**.



Al seleccionar la opción **Auto** para el parámetro **Stereo Eye**, el nodo **MediaOut** combina las señales y las asigna al ojo correspondiente en la línea de tiempo. Esto permite implementar una dinámica de trabajo multicapa en Fusion con el fin de transmitir las señales para el ojo derecho y el izquierdo a un solo nodo **MediaOut**.

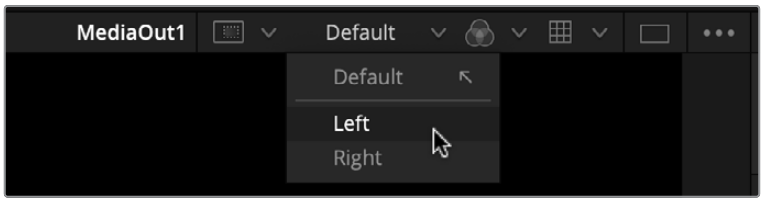
Nodo Renderer3D

Al emplear gráficos 3D o elementos computarizados en una línea de tiempo inmersiva, el nodo **Renderer3D** ofrece varias configuraciones para cada ojo y la capacidad de procesar las capas izquierda y derecha.



Esto brinda la posibilidad de crear títulos 3D complejos o escenas completas con elementos computarizados y enviar todo directamente a la una línea de tiempo estereoscópica en proyectos inmersivos.

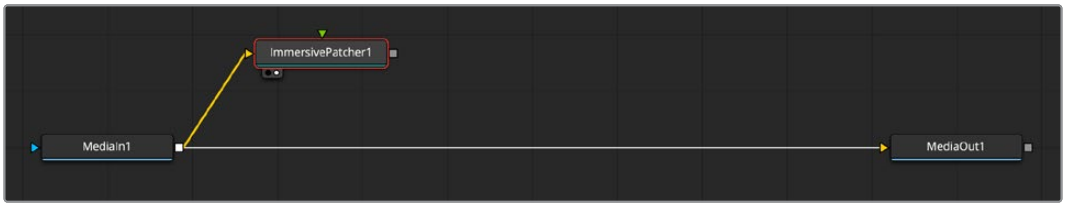
Al trabajar de esta manera, es posible alternar entre la composición para el ojo izquierdo y el derecho desde el menú desplegable en la esquina superior derecha de los visores en el módulo Fusion.



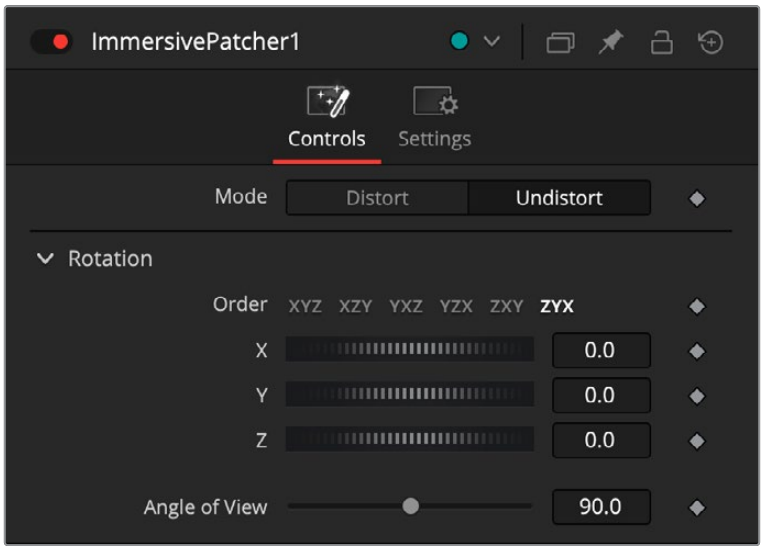
Composiciones con herramientas inmersivas de parche

La herramienta **Parche inmersivo** permiten eliminar temporalmente la distorsión de un clip inmersivo a fin de facilitar el seguimiento, los ajustes cromáticos y la composición en un plano 2D. Al finalizar, es posible utilizar un segundo parche inmersivo para volver a distorsionar la señal e integrar los nuevos elementos compuestos en la escena.

1 Añada un nodo **ImmersivePatcher** a la estructura nodal.



2 En el inspector, haga clic en **Deshacer distorsión**.



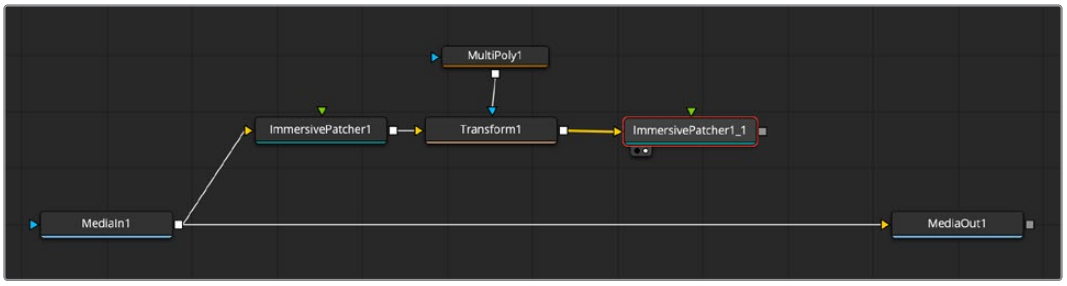
3 Utilice los parámetros de rotación **X**, **Y** y **Z** para seleccionar la sección de la imagen inmersiva a aplanar.

4 Continúe trabajando con la imagen compuesta al igual que con una señal 2D tradicional. Esto incluye, entre otras cosas, métodos estándar para eliminar objetos no deseados; muestrear secciones de un fotograma mediante transformaciones y polígonos para ampliar el set; o aplicar mates, texto y capas externas de imágenes o video para reemplazar cartelería o pantallas.

Al finalizar, deberá volver a distorsionar la imagen plana a modo de qué coincida con la distorsión de la imagen inmersiva original.

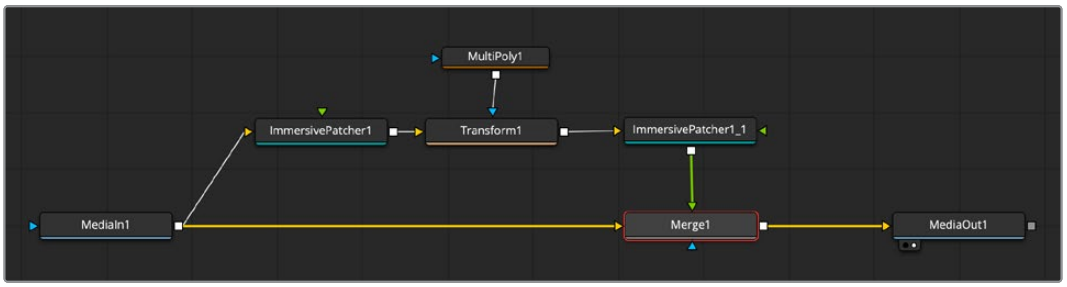
5 Copie el nodo **ImmersivePatcher** para conservar la información de la rotación y el ángulo de visión originales.

6 Posteriormente, péguelo después con los nodos de composición y conéctelo al final de la estructura correspondiente.



7 En el inspector, haga clic en **Distorsionar**.

8 Una el parche compuesto con el clip original a fin de integrar el efecto en una escena inmersiva.



Al colaborar con otras personas, es posible salir del módulo Fusión y continuar editando, mezclando y etalonando los clips en la línea de tiempo inmersiva en DaVinci Resolve.

Si es necesario entregar contenidos inmersivos, se recomienda renderizar las imágenes correspondientes a cada ojo por separado en archivos EXR, usando el espacio cromático ACES AP0, ya que esto permitirá conservar las propiedades al máximo.

Ajustes cromáticos

Este apartado brinda información sobre las herramientas ópticas, de etalonaje y 3D que se pueden utilizar al trabajar con archivos inmersivos en el módulo Color de DaVinci Resolve.

Dada la naturaleza singular de este tipo de contenidos, no todas las prestaciones estarán disponibles durante la posproducción. Sin embargo, algunas de ellas que aún no son compatibles están siendo adaptadas para que funcionen con clips inmersivos.

Reducción de ruido

Al trabajar con contenidos inmersivos es posible reducir el ruido. En el panel **Efectos de movimiento**, la opción **Reducción de ruido espacial** analiza el patrón de ruido del archivo original con el fin de reducirlo al máximo, conservando a la vez los detalles de la imagen.

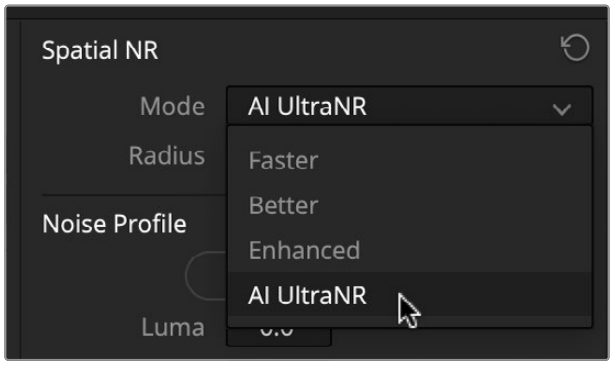
1 En la barra de herramientas del módulo Color, abra el panel **Efectos de movimiento**.

En la columna **Reducción de ruido espacial**, es posible seleccionar el modo de reducción de ruido que se desea aplicar.

- *Los modos* **Mayor velocidad**, **Mejor calidad** y **Mejorado** llevarán a cabo su análisis de patrones dentro de un radio determinado y configurarán la imagen resultante en función del umbral establecido. Estos modos se enumeran según el orden de complejidad de procesamiento.

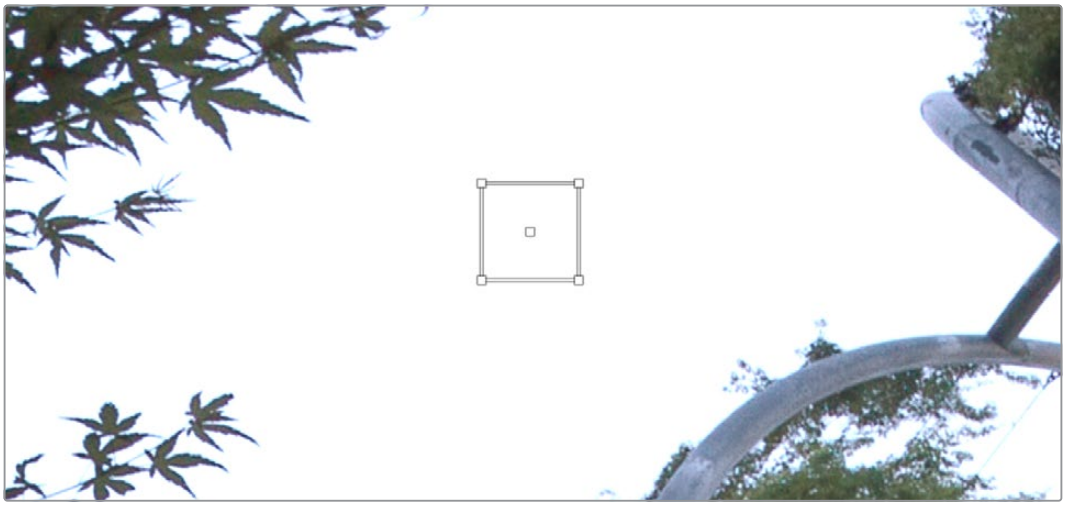
- La opción **UltraNR (IA)** es un identificador de patrones basado en inteligencia artificial que reconoce el patrón de ruido en la señal ubicando y analizando la sección con menos detalles del fotograma en busca del ruido puro.

2 En el modo de reducción de ruido espacial, seleccione la opción **UltraNR (IA)**.



3 Haga clic en **Analizar**.

Aparecerá un recuadro en el visor, que identifica el área con menos datos de pixeles en un fotograma determinado AI no presentar contenidos visuales tangibles, los datos dentro de ese recuadro son ideales para extraer el ruido digital provocado por el sensor de la cámara.



Si bien es posible reposicionar y redimensionar el recuadro en el visor si así se desea, esto no es recomendable, ya que la herramienta **UltraNR (IA)** muestra un mejor desempeño que el ojo humano al detectar las regiones de análisis óptimas.

Asimismo, es posible modificar los ajustes del umbral de luminancia y crominancia del perfil de ruido, aunque en este caso la herramienta **UltraNR (IA)** también tiende a alcanzar automáticamente una combinación ideal de umbrales para una reducción del ruido adecuada.

Correcciones cromáticas primarias

Las correcciones cromáticas primarias funcionan de la misma manera que al trabajar con material grabado 2D. Aplique las mismas consideraciones para la normalización, el balance y la correspondencia entre imágenes al utilizar los círculos cromáticos primarios y logarítmicos, el panel **Etalonaje HDR** y las curvas HSL. Los paneles **Color Slice** y **Distorsión cromática** también funcionan normalmente.

Correcciones cromáticas secundarias

Al trabajar con archivos inmersivos, los métodos de selección de pixeles, por ejemplo, el de colores específicos, se comportan normalmente.

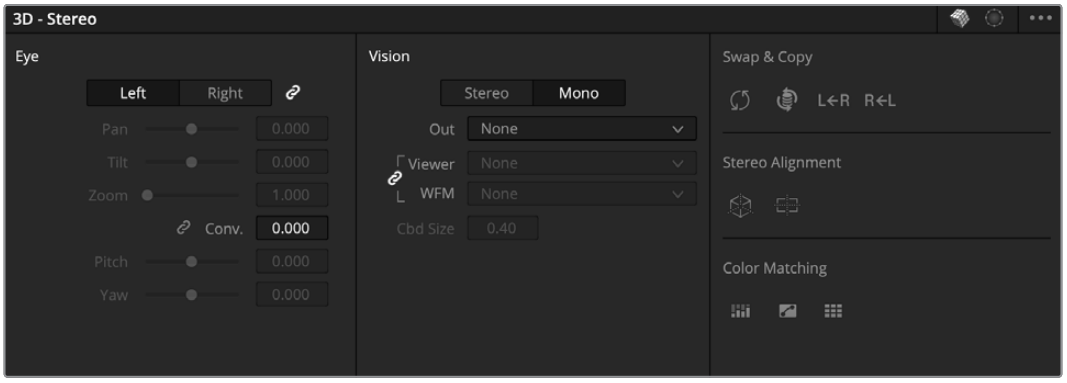
La selección basada en objetos o ventanas solo funciona correctamente en un entorno plano equirrectangular (o área de visualización), es decir que la mayoría de los métodos secundarios que requieren de precisión al detectar el color específico de una figura actualmente no están disponibles. Esto incluye las Power Window, y por ende en panel **Seguimiento**, y la herramienta Magic Mask.

Ajuste de tamaño

Los controles de ajuste de tamaño (esto es, distancia focal y distribución) no están disponibles para contenidos inmersivos.

Panel de imágenes estereoscópicas (3D)

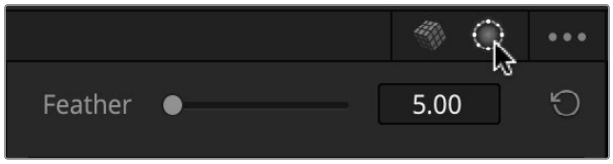
En el módulo Color, el panel **Imágenes estereoscópicas** ofrece controles para organizar distintos formatos de cámara y gráficos 3D. A diferencia de otros formatos estereoscópicos, no es preciso ajustar la alineación de los ojos, dado que los objetivos, las distancias focales y los diafragmas de la cámara Blackmagic URSA Cine Immersive son fijos.



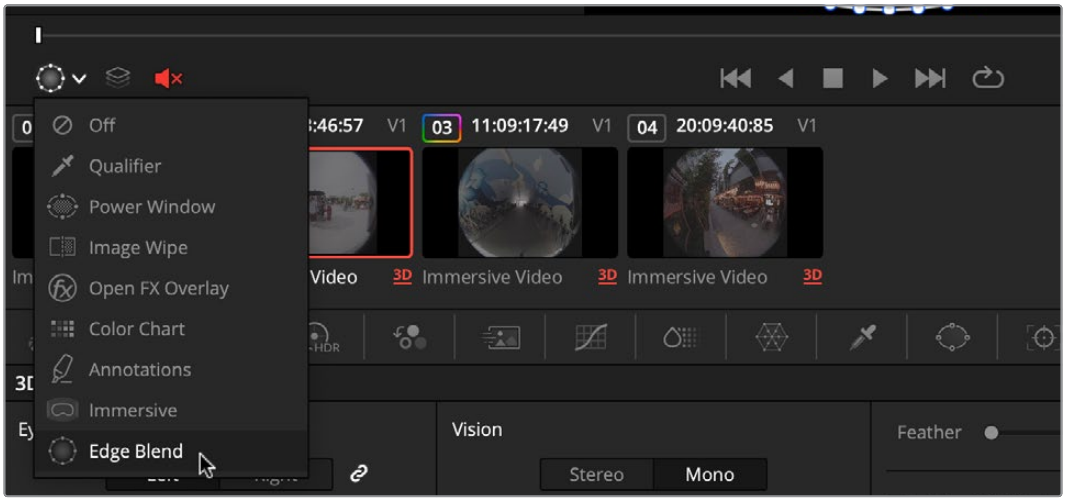
El modo **Atenuación de bordes** funciona exclusivamente con contenidos inmersivos y permite modificar y suavizar el perímetro de los fotogramas de las imágenes inmersivas. Esto resulta de utilidad para esconder el equipamiento de filmación y limitar el campo de visión del observador. Más allá de este borde, el observador verá el fondo.

1 Abra el panel **Imágenes estereoscópicas**.

2 En la esquina superior derecha, cambie el modo a **Atenuación de bordes**.



3 Asegúrese de que los controles del visor que se muestran en la pantalla se encuentren en el mismo modo.



Debería ver un círculo de puntos de control alrededor del perímetro del fotograma con un margen de fijación en el centro.

4 Arrastre un punto de control para cambiar su posición.



También puede hacer clic con el botón derecho del mouse para eliminarlo.

5 En la columna derecha del panel **Imágenes estereoscópicas**, utilice el parámetro **Suavizar** para atenuar el borde del fotograma.

6 Elija **Efecto VisionOS** a fin de garantizar que los nuevos datos del borde se procesen correctamente durante la reproducción en dispositivos Apple Vision Pro. Si no se selecciona esta opción, el borde se verá permanentemente en el archivo renderizado.

7 En la columna izquierda, utilice las pestañas **Izquierdo** y **Derecho** para alternar entre cada ojo en el visor.

Cabe señalar que muchos de los controles en el panel **Imágenes estereoscópicas** no están disponibles y aparecerán en gris. Esto también ocurrirá con diversos parámetros en el programa cuando no sean compatibles con los archivos con los que se está trabajando, por ejemplo, contenidos inmersivos.

8 En la columna del medio, use las pestañas **Estéreo** y **Mono** para alternar entre el visor de un objetivo y el de los dos en el clip inmersivo.

Gestión cromática con cambios del espacio cromático

El efecto **Cambio del espacio cromático** (CST) puede aplicarse a los nodos presentes en el editor de nodos del módulo Color con el fin de conseguir una asignación cromática precisa y controlar como se desee el nivel máximo de nits de la luminancia en la línea de tiempo.

La dinámica de trabajo recomendada incluye cuatro nodos CST:

1. **CST INICIAL** – Cámara a espacio cromático en uso.

   La corrección cromática ocurre entre estos dos nodos.

2. **CST FINAL** – Espacio cromático en uso a pantalla Mac HDR.
3. **CST DE MONITORIZACIÓN** – 1000 nits a 250 nits para supervisión en HDR.
4. **CST AVP** – 250 nits a 108 nits para supervisión y entrega de contenidos para dispositivos Apple Vision Pro.



Asignación de cambio del espacio cromático en la línea de tiempo

En primer lugar, se realiza la asignación desde el espacio cromático inicial de la fuente proveniente de la cámara al espacio cromático de la línea de tiempo en que se trabaja. El modelo Blackmagic URSA Cine Immersive funciona con el espectro cromático Blackmagic Design Wide Gamut Gen. 4/5 y la curva logarítmica **DaVinci Intermediate**.

1. Abra el panel **Efectos**.
2. Arrastre el efecto **Cambio del espacio cromático** a un nodo vacío (01) en el editor de nodos.

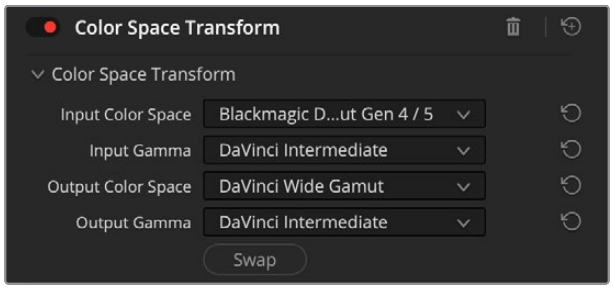
   Al hacer una asignación desde el espacio cromático inicial directamente al espacio cromático de la línea de tiempo, este nodo debería ubicarse entre los primeros en la estructura nodal, antes de comenzar a etalonar.

   En los casos en que la asignación se realiza desde el espacio cromático inicial directamente al de supervisión o entrega, este nodo debería estar entre los últimos en la estructura nodal una vez finalizados los ajustes cromáticos.

3. Cambie el nombre del nodo 01 a **CST INICIAL**.

NOTA: Los nombres empleados para los nodos en estas instrucciones facilitan las referencias a la documentación y las imágenes en el manual. No obstante, es posible usar la nomenclatura que resulte más adecuada u omitir este paso.

4. Seleccione **Blackmagic Design Wide Gamut Gen. 4/5** como espacio cromático inicial.
5. En **Gamma inicial**, elija la opción **DaVinci Intermediate**.
6. En **Espacio cromático final**, seleccione **DaVinci Wide Gamut**.
7. En **Gamma final** seleccione la opción **DaVinci Intermediate**.



8 Añada un nodo en serie (02).

9 Copie y pegue el primer nodo (01) en el nuevo nodo (02).

10 Cambie el nombre del nodo 02 a **CST FINAL**.

En este nuevo nodo serial, se deben seleccionar el espacio cromático y la curva de respuesta tonal como datos de entrada para garantizar la congruencia con el espacio cromático de la línea de tiempo de la estructura nodal.

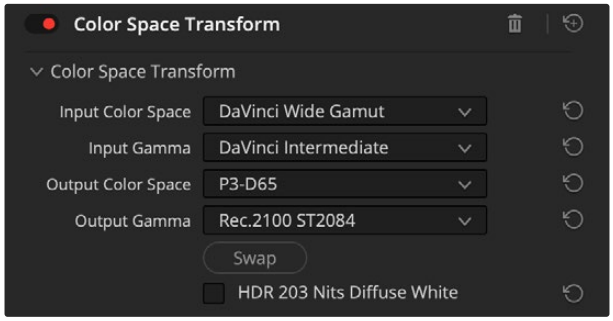
11 Haga clic en el botón **Intercambiar** a fin de cambiar los parámetros de inicio y fin.

El espacio cromático inicial y el *Gamma inicial* ahora son adecuados para este nodo CTS.

12 Seleccione **P3-D65** como espacio cromático final. Este es el espectro cromático predeterminado en las pantallas Mac.

13 En el menú **Gamma final** elija la opción **Rec.2100 ST2084**, que es el formato HDR para una señal de 1000 nits.

Todas las operaciones de etalonaje ocurrirán entre estos dos nodos. Sin embargo, dado que se emplea una gestión cromática que depende del nodo, se cuenta con la oportunidad única de llevar a cabo operaciones fuera de los límites de la gestión cromática de la línea de tiempo; muchos coloristas sacan provecho de esta posibilidad aplicando correcciones cromáticas, efectos o elementos de terceros específicos.



Perfiles ICC

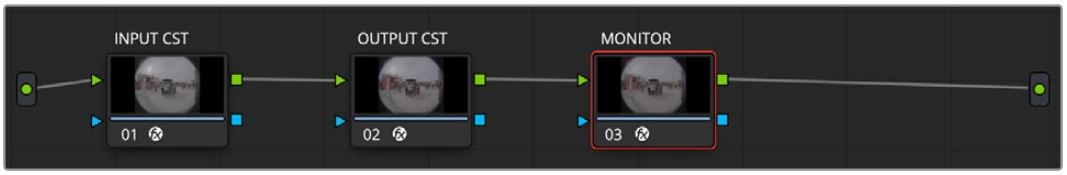
Para más seguridad de que la señal de la cámara Blackmagic URSA Cine Immersive se muestre correctamente en su equipo Mac, vaya a **Ajustes del sistema** (o **Configuración del Sistema**) > **Pantallas** y verifique que los ajustes predeterminados del perfil cromático estén configurados como **Apple XDR Display (P3-1600 nits)**.

Cambio del espacio cromático en monitores de etalonaje

Es posible aplicar cambios del espacio cromático adicionales con el fin de limitar temporalmente los valores máximos de nits que se visualizan al supervisar contenidos durante el etalonaje.

1 Añada un nuevo nodo serial (03) después del espacio cromático **CST FINAL** (02).

2 Cambie el nombre del nodo 03 a **MONITOR**.



3 Añada el efecto de cambio del espacio cromático al nodo (03) **MONITOR**.

Los valores iniciales deben coincidir con los de la señal saliente del nodo CST anterior.

4 Seleccione **P3-D65** como espacio cromático inicial.

5 En **Gamma inicial**, seleccione la opción **Rec.2100 ST2084**.

El propósito de este nodo **CST** es simplemente representar la luminancia de la imagen con un valor menor de nits, de manera que los parámetros iniciales y finales permanezcan iguales.

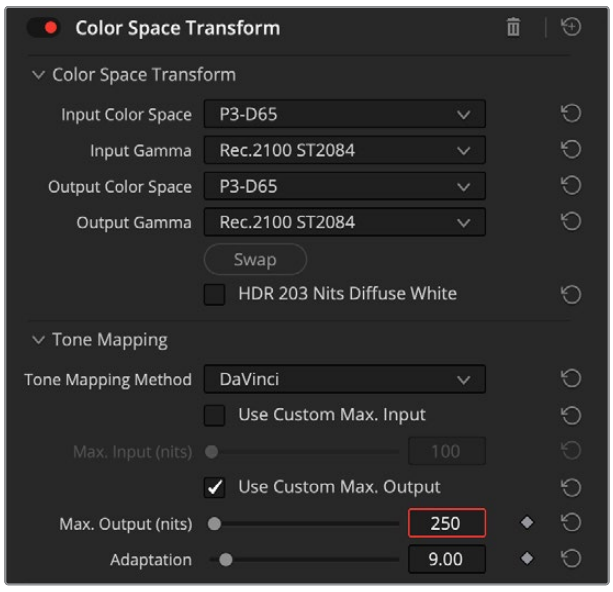
6 Seleccione **P3-D65** como espacio cromático final.

7 En **Gamma final**, seleccione la opción **Rec.2100 ST2084**.

NOTA: En ciertos casos en que los valores iniciales y finales son idénticos (no hay una transformación), es preciso establecer el espacio cromático o la curva de respuesta tonal de un CST. En este caso, resulta necesario porque la aplicación de la asignación de nits personalizada se ve afectada por el valor y la forma de la curva logarítmica de la señal.

8 Seleccione **Valor máximo final personalizado**.

9 Mueva el control deslizante para la opción **Valor final máx. (nits)** hasta **250**.



Ahora la señal de 1000 nits se representará en 250 nits mediante el método de asignación de tonos de DaVinci.

Esto dará como resultado una imagen que coincidirá con la señal renderizada para Apple Vision Pro en un monitor de etalonaje (por ejemplo, Flanders XMP310, Sony BVM-HX310, etc.).

No obstante, si se desea supervisar el proyecto en un dispositivo Apple Vision Pro, será preciso llevar a cabo una asignación de tonos adicional.

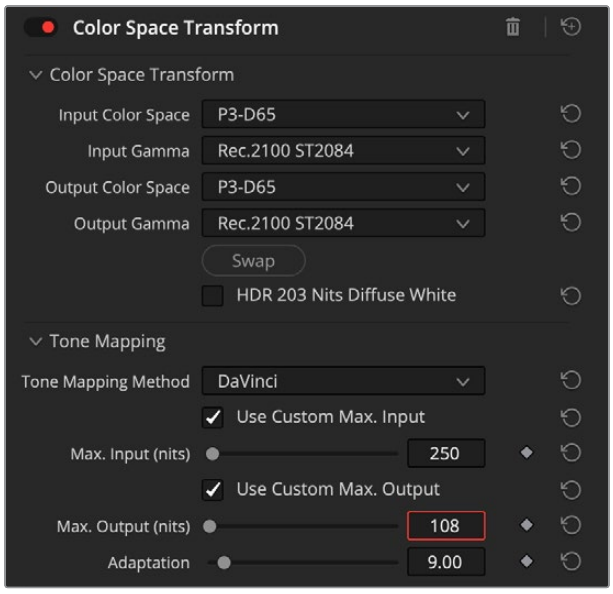
Cambio del espacio cromático para supervisión en dispositivos Apple Vision Pro

Es posible emplear un último nodo CST con el fin de posibilitar la supervisión de contenidos en directo, las vistas previas y la entrega final para dispositivos Apple Vision Pro.

1. Añada un nuevo nodo serial (04) después del nodo (03) **MONITOR**
2. Copie y pegue el nodo (03) **MONITOR** al nodo (04). Se aplicarán los mismos parámetros iniciales y finales, solamente cambiará el valor de nits.
3. Modifique el nombre del nodo 04 a **AVP**.



4. Marque la casilla **Valor máximo inicial personalizado**.
5. Mueva el control deslizante correspondiente a la opción **Valor inicial máx. (nits)** hasta **250** a fin de interpretar correctamente la señal asignada desde el nodo **MONITOR.**
6. Mueva el control deslizante correspondiente a la opción **Valor final máx. (nits)** hasta **180**; este es el valor máximo de nits para la supervisión y entrega de contenidos destinados a dispositivos Apple Vision Pro.



Ahora es posible proceder con el etalonaje, desactivando los nodos **CST**, siempre que esto sea necesario en función de la dinámica de trabajo. Por ejemplo, durante las correcciones cromáticas iniciales en que se utiliza el monitor de etalonaje externo, debería desactivar el nodo (04) **AVP** a fin de garantizar que la señal de 250 nits se visualice correctamente en el monitor HDR. Al pasar a la vista previa antes de la entrega final o en un dispositivo Apple Vision Pro, vuelva a activar el nodo (04) **AVP**.

¿Los cambios del espacio cromático se pueden ubicar en el editor de nodos de la línea de tiempo?

No, por el momento la asignación de tonos a clips depende de la aplicación precisa del parámetro **Adaptación del cambio del espacio cromático**, que se realiza en cada clip por separado.

Audio

El dispositivo Apple Vision Pro y DaVinci Resolve admiten audio de tipo espacial, ambisónico y AVP (Apple Vision Pro), además de formatos AIA (Apple Immersive Audio).

Muchos aspectos de la posproducción de audio permanecen iguales al trabajar con archivos espaciales. Una de las principales diferencias es que, en lugar de supervisar el sonido con altavoces profesionales, los ingenieros de sonido deben utilizar auriculares o AirPod con el fin de conseguir una verdadera experiencia espacial inmersiva y reproducir con más precisión el entorno auditivo que percibe el público.

Si bien en los archivos 2D la mayoría del diálogo se mezcla en los canales centrales, al trabajar con audio espacial es preciso hacerlo en el área donde se origina el sonido. Para mezclar los componentes individuales de audio de una manera que no abrume o confunda al oyente, se deben utilizar los controles ecualización y balance, y el efecto **Ducking**.

Los elementos espaciales inmersivos se pueden combinar con componentes comunes de audio en estéreo, lo cual permite que ciertos sonidos permanezcan constantes sin importar en qué dirección se mueva la cabeza. Esto resulta de utilidad para mantener la claridad y la facilidad de interpretación de los sonidos no diegéticos, por ejemplo, la música cinematográfica y las narraciones.

Los sonidos ambientales tienden a interpretarse más sutilmente en el audio espacial, de modo que a menudo deben aumentarse en la mezcla para que puedan ser reconocidos. Aun así, esto no debe hacerse a costa de desorientar al oyente.

Las reverberaciones contribuyen considerablemente al efecto inmersivo, algo que experimentamos naturalmente todos los días. Cada sonido que se produce en un ambiente determinado rebota contra las superficies que nos rodean, dando lugar a una reverberación del tipo *floreciente*. Por lo tanto, se promueve el uso de reverberaciones al diseñar audio espacial, con el fin de garantizar una inmersión completa.

El formato Apple Immersive Audio (AIA) comprende diversas variantes:

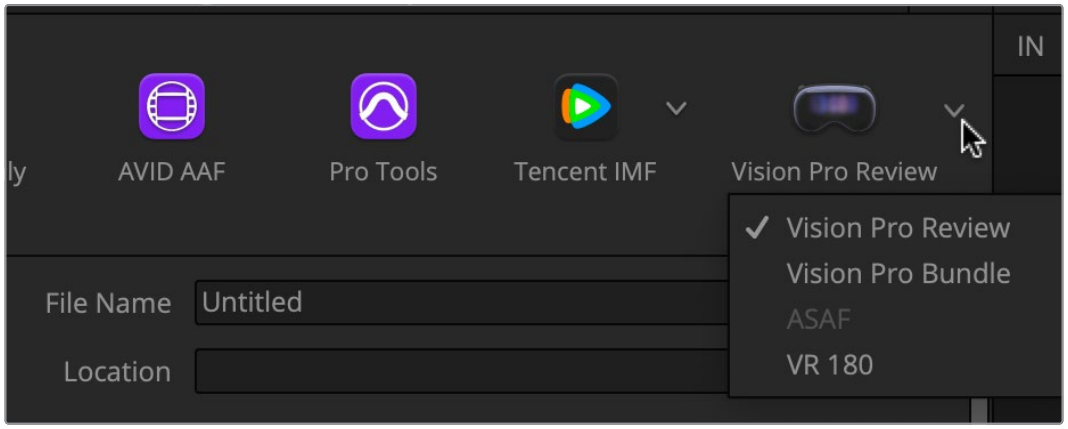
- **Apple Spatial Audio Format (ASAF)** es el formato de trabajo de audio original.
- **Apple Positional Audio Codec (APAC)** es el formato MP4 codificado que se puede reproducir en dispositivos compatibles con APAC, tales como Airpod, o desde un archivo AIVU en dispositivos Apple Vision Pro.

Formato Apple Spatial Audio Format (ASAF)

Apple Spatial Audio Format (ASAF) es un formato de audio inmersivo desarrollado principalmente para reproducir contenidos de realidad virtual y realidad aumentada en dispositivos Apple Vision Pro. Es exclusivo de macOS y compatible con herramientas como ProTools (mediante complementos) y DaVinci Resolve (integrado). ASAF dispone de capacidades técnicas avanzadas, entre ellas compatibilidad integrada con fuentes ambisónicas, llamadas escenas, y asignación mediante buses. La mezcla se lleva a cabo de manera virtual, sin un bus físico, y los usuarios pueden elegir entre diversos formatos de supervisión, por ejemplo, biaural con seguimiento de la cabeza mediante integración directa con Apple AirPod. La distribución se realiza manualmente mediante las herramientas existentes de DaVinci Resolve, como las de distribución de diálogos, automatización, SpaceView, y las de distribución en el visor, además de controles secundarios en el inspector. Esto brinda la posibilidad de ubicar las fuentes en el espacio con precisión, así como de hacer un seguimiento automático en función de las coordenadas de latitud y longitud de las imágenes. Las mezclas ASAF se pueden exportar en formato WAV o MP4 (con los datos acompañantes) desde **Sonido inmersivo** > **Exportar archivo maestro**, a pesar de que actualmente no se pueden importar a DaVinci Resolve o reproducir desde allí.

Entrega

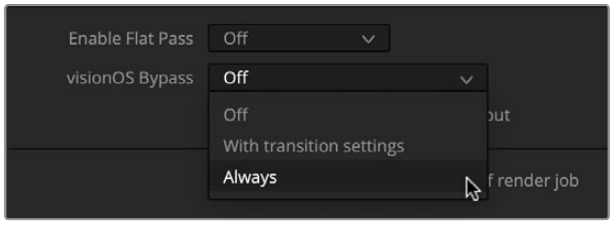
En el módulo Entrega, hay dos ajustes de renderización que permiten renderizar archivos para dispositivos Apple Vision Pro a fin de revisarlos y entregarlos.



- **Vision Pro Review** – Crea un paquete MV-HEVC pequeño a los efectos de acceder a la vista previa. Es posible compartir el archivo resultante con el dispositivo Apple Vision Pro a través de una conexión Wi-Fi y mirarlo en la aplicación Apple Immersive Video Utility (AIVU), disponible en la tienda App Store.
- **Vision Pro Bundle** – Crea un paquete de archivos ProRes que se puede archivar o procesar en la aplicación Compressor a fin de prepararlo para reproducirlo mediante Apple TV.
- **ASAF** – Renderiza una mezcla de audio ASAF en formato WAV o MP4.
- **VR180** – Genera un archivo estereoscópico compatible con el formato VR180.

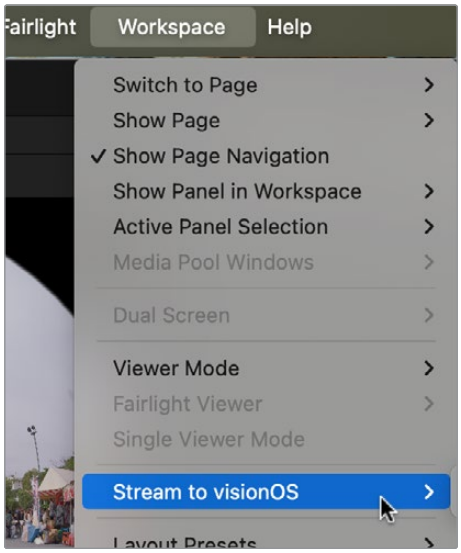
Omitir transiciones

Si se añade una transición en una línea de tiempo inmersiva, se hará una prerrenderización y se convertirá a una única lista de reproducción. El archivo XML generado para los contenidos inmersivos describe el proceso que ocurre al pasar de un componente renderizado a otro (grabación del contenido audiovisual, posterior renderización de transición y reconversión a material grabado). En la sección **Ajustes avanzados** del panel **Ajustes de renderización**, es posible omitir las transiciones visionOS (**Siempre**), aplicarlas según su estado en la línea de tiempo (**Con ajustes de transición**), o incluirlas tal como aparecen en la línea de tiempo (**Desactivada**).



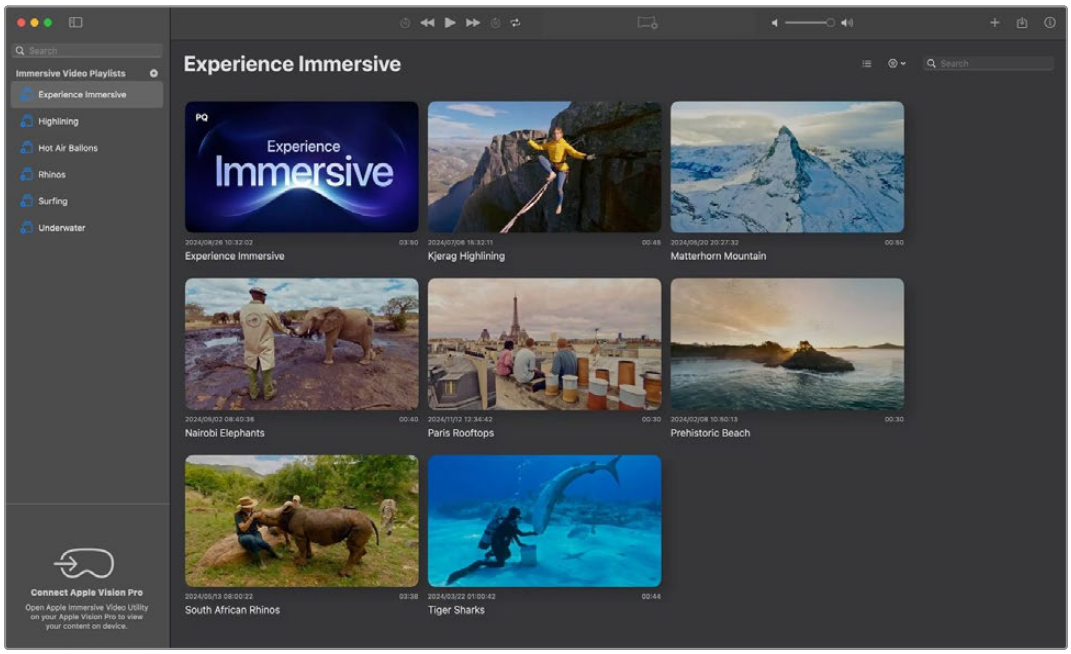
Transmisión a dispositivos Apple Vision Pro

Es posible transmitir contenidos a Apple Vision Pro desde los módulos Edición, Color y Entrega en DaVinci Resolve desde el menú **Área de trabajo**, seleccionando **Transmitir a visionOS** y, luego, el dispositivo AVP. Al realizar transmisiones por Internet, se recomienda utilizar una red inalámbrica de alta velocidad (Wi-Fi 5 como mínimo, y preferentemente Wi-Fi 6).



Acerca de Apple Immersive Video Utility

Apple Immersive Video Utility es una aplicación especializada para macOS y visionOS, diseñada con el fin de simplificar la creación, la gestión y la reproducción de contenidos Apple Immersive Video. El programa ha sido concebido específicamente para profesionales del cine y creadores de contenidos inmersivos, y permite a los usuarios importar, organizar, crear paquetes de archivos y revisar contenidos estereoscópicos en resolución 8K con un campo de visión de 180° y audio espacial. A su vez, permite crear archivos AIVU (Apple Immersive Video Universal), además de inspeccionar y modificar los metadatos durante la posproducción. Asimismo, los usuarios pueden crear listas de reproducción, hacer búsquedas en sus bibliotecas e incluso transmitir contenidos directamente a uno o más dispositivos Apple Vision Pro.



Developer Information

Camera Control REST API

If you are a software or hardware developer you can build custom applications or leverage ready to use tools such as REST client or Postman to seamlessly control and interact with your compatible Blackmagic camera using Camera Control REST API. This API enables you to perform a wide range of operations, such as starting or stopping recordings, accessing disk information and much more. Whether you're developing a custom application tailored to your specific needs or utilizing existing tools, this API empowers you to unlock the full potential of your Blackmagic camera with ease. We look forward to seeing what you come up with!

NOTE It's important to mention that controlling Blackmagic cameras via REST API relies on the web manager being enabled on each compatible Blackmagic camera. Enable the web media manager in the Blackmagic Camera Setup 'network access' settings for each camera you are controlling.

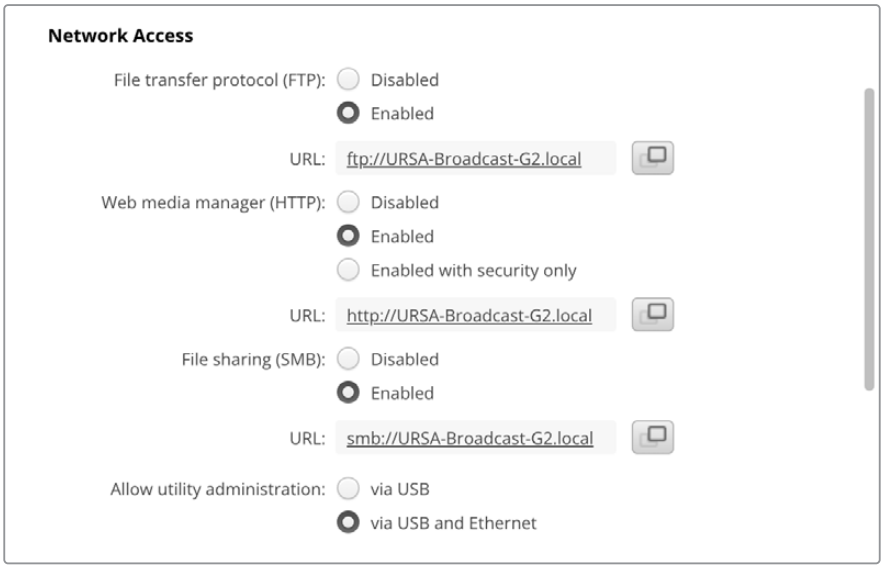
The following Blackmagic cameras are compatible with Camera Control REST API:

- Blackmagic URSA Cine 12K LF
- Blackmagic URSA Cine 17K 65
- Blackmagic URSA Cine Immersive
- Blackmagic PYXIS 6K
- Blackmagic Cinema Camera 6K
- Blackmagic URSA Broadcast G2
- Blackmagic Micro Studio Camera 4K G2
- Blackmagic Studio Camera 4K Plus
- Blackmagic Studio Camera 4K Pro
- Blackmagic Studio Camera 6K Pro
- Blackmagic Studio Camera 4K Plus G2
- Blackmagic Studio Camera 4K Pro G2

Sending API Commands

To send an API command to your camera from a third party application such as Postman, add /control/api/v1/ to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/api/v1/

You can find the Web media manager URL and IP address information in Blackmagic Camera Setup.



The Web media manager URL in Blackmagic Camera Setup

Downloading API's from your Camera

You can download REST API YAML documentation from your camera by adding /control/documentation.html to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/documentation.html

NOTE It's worth noting that changing the camera name in Blackmagic Camera Setup will also change the camera's Web media manager URL.

Livestream Control API

API for controlling Livestreams on Blackmagic Design products.

GET /livestreams/0

Get the livestream's current status.

Response

200 - Livestream's current status.

The response is JSON.

Name	Type	Description
status (required)	string	Possible values are: Idle, Connecting, Streaming, Flushing, Interrupted.
bitrate (required)	integer	Current bitrate (bps).
effectiveVideoFormat (required)	string	Effective video format for the livestream, serialised as a string.
duration	integer	Current stream duration in seconds. Absent if livestream is idle.
cache	integer	Current stream cache usage percentage.

GET /livestreams/0/start

Determine if the livestream is active.

Response

200 - Livestream active status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is active.

PUT /livestreams/0/start

Start the livestream.

Response

204 - Livestream started.

GET /livestreams/0/stop

Determine if the livestream is inactive.

Response

200 - Livestream inactive status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is inactive.

PUT /livestreams/0/stop

Stop the livestream.

Response

204 - Livestream stopped.

GET /livestreams/0/activePlatform

Get the currently selected platform configuration for the livestream.

Response

200 - Livestream active platform configuration.

The response is JSON.

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

PUT /livestreams/0/activePlatform

Set the currently selected platform configuration for the livestream.

Parameters

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

Response

204 - Livestream active platform configuration updated.

400 - Bad Request

GET /livestreams/platforms

Get the list of available platforms.

Response

200 - List of available platforms.

The response is JSON.

Name	Type	Description
	array	List of available platforms names.
[i]	string	Platform name.

GET /livestreams/platforms/{platformName}

Get the service configuration for a platform.

Parameters

Name	Type	Description
{platformName} (required)	string	Name of the platform.

Response

200 - Service configuration for specified platform.

The response is JSON.

Livestream platform service configuration.

Name	Type	Description
platform (required)	string	Corresponding platform name.
key	string	Default stream key.
servers (required)	array	List of server configurations.
servers[i]	object	Server configuration.
servers[i].server (required)	string	Server name.
servers[i].url (required)	string	Livestream destination.
servers[i].srtExtensions	array	Miscellaneous tags used for SRT livestreams.
servers[i].srtExtensions[i]	object	Dictionary object mapping SRT tag strings to values.
servers[i].srtExtensions[i][{key}]	string	SRT tag value.
servers[i].group	string	Logical grouping of the server.
profiles (required)	array	List of profile configurations.
profiles[i]	object	Quality configuration.
profiles[i].profile (required)	string	Quality level name.
profiles[i].configs (required)	array	List of video format configurations.
profiles[i].configs[i]	object	Video format configuration for profiles.
profiles[i].configs[i].resolution (required)	string	Video format serialised as a string.
profiles[i].configs[i].fps (required)	string	Frames per second.
profiles[i].configs[i].bitrate (required)	integer	Pixel bitrate (bps).
profiles[i].configs[i].audioBitrate	integer	Audio bitrate (bps).
profiles[i].configs[i].keyFrameInterval	integer	How often a key frame is sent, in seconds.
profiles[i].configs[i].videoCodecs	array	Supported video encoding algorithm/s.

Name	Type	Description
profiles[i].configs[i].videoCodecs[i]	string	Video encoding algorithm. Possible values are: H264, H265.
profiles[i].lowLatency (required)	boolean	If true, fewer frames will be buffered in the livestream.
defaultProfile	string	Quality level name.
credentials	object	Credientials used for RTMP streams.
credentials.username (required)	string	The username part of the creditials. Only used for RTMP streams.
credentials.password (required)	string	Used for RTMP streams, also used as Passphrase for SRT streams.
customizableUrlEnabled	boolean	True when the server URL is customizable.

400 - Bad Request

GET /livestreams/customPlatforms

Get a list of custom platform files.

Response

200 - List of custom platform files.

The response is JSON.

Name	Type	Description
	array	List of custom platform file names.
[i]	string	Custom platform file name.

DELETE /livestreams/customPlatforms

Remove all custom configuration files.

Response

204 - All custom configuration files removed.

GET /livestreams/customPlatforms/{filename}

Get a custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to get.

Response

200 - Custom platform file.

The response is XML.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

404 - Not Found

PUT /livestreams/customPlatforms/{filename}

Update a custom platform file if it exists, if not, create a new file with the given file name.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to update/create.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

Response

204 - Custom platform file created or updated.

400 - Bad Request

DELETE /livestreams/customPlatforms/{filename}

Remove the given custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to be removed.

Response

204 - Custom platform file removed.

404 - Not Found

Clips Control API

API for listing clips on disk.

GET /clips

Get the list of clips on the active disk.

Response

200 - List of clips on the active disk.

The response is JSON.

List of media clips.

Name	Type	Description
clips (required)	array	
clips[i]	object	Media clip.
clips[i].clipUniqueId (required)	integer	Unique ID used to identify this clip.
clips[i].filePath	string	Path to the file relative to the root of a mount.
clips[i].fileSize	integer	Size of file on disk in bytes.
clips[i].codecFormat	object	
clips[i].codecFormat.codec	string	Currently selected codec.

Name	Type	Description
clips[i].codecFormat.container	string	Multimedia container format.
clips[i].videoFormat	object	Video format configuration.
clips[i].videoFormat.name (required)	string	Video format serialised as a string.
clips[i].videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
clips[i].videoFormat.height	number	Height dimension of video format.
clips[i].videoFormat.width	number	Width dimension of video format.
clips[i].videoFormat.interlaced	boolean	Is the display format interlaced?
clips[i].startTimecode	string	Start timecode of the clip serialised as string.
clips[i].durationTimecode	string	Duration of the clip in timecode format serialised as string.
clips[i].frameCount	integer	Number of frames in clip; duration of the clip in frames.

404 - There is no active disk.

Media Pool Control API

API to manage media pool and handle uploads and project data.

GET /cloud/projects

List all projects within the media pool.

Response

200 - Successfully retrieved the list of all projects.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].libraryID	string	
[i].name	string	
[i].private	boolean	
[i].shared	boolean	
[i].clips	array	List of clips associated with the project.
[i].clips[i]	string	
[i].status	object	
[i].status.numClipsRequested	integer	
[i].status.numClipsComplete	integer	
[i].status.uploadPercent	integer	Percentage of upload completion.
[i].status.numClipsPaused	integer	
[i].status.outOfSpace	boolean	
[i].status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
[i].status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/active

Retrieve data of the actively uploading project.

Response

200 - Successfully retrieved the active project's data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/{projectID}

Retrieve specific project data by project ID.

Parameters

Name	Type	Description
{projectID} (required)	integer	Unique identifier of the project.

Response

200 - Successfully retrieved the project data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	

Name	Type	Description
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

404 - Project not found.

GET /cloud/clips

List all clips within the media pool.

Response

200 - Successfully retrieved the list of all clips.

The response is JSON.

Name	Type	Description
	array	
[i]	string	

GET /cloud/clips/activeUploading

Retrieve data of actively uploading clips.

Response

200 - Successfully retrieved the list of actively uploading clips.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].path	string	
[i].projectID	integer	
[i].status	object	
[i].status.projectID	integer	
[i].status.outOfSpace	boolean	
[i].status.proxyExtension	string	
[i].status.growingFile	boolean	
[i].status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.originalClipTotalSize	integer	
[i].status.proxyClipTotalSize	integer	
[i].status.originalClipCompletedSize	integer	
[i].status.proxyClipCompletedSize	integer	
[i].status.secsRemaining	integer	

GET /cloud/clips/{deviceName}/{path}

Retrieve specific clip data by device and path.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored.
{path} (required)	string	Path to the clip.

Response

200 - Successfully retrieved the clip data.

The response is JSON.

Name	Type	Description
path	string	
projectID	integer	
status	object	
status.projectID	integer	
status.outOfSpace	boolean	
status.proxyExtension	string	
status.growingFile	boolean	
status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.originalClipTotalSize	integer	
status.proxyClipTotalSize	integer	
status.originalClipCompletedSize	integer	
status.proxyClipCompletedSize	integer	
status.secsRemaining	integer	

404 - Clip not found.

Monitoring Control API

API for monitoring and controlling display settings in video equipment.

GET /monitoring/display

Retrieve a list of all display names.

Response

200 - Returns a list of display names.

The response is JSON.

Name	Type	Description
displays	array	List of display names available.
displays[i]	string	

GET /monitoring/{displayName}/cleanFeed

Get the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/cleanFeed

Set the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Clean feed enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/displayLUT

Get the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

400 - Invalid display name.

404 - Display name not found.

PUT /monitoring/{displayName}/displayLUT

Set the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Display LUT enabled/disabled successfully.

400 - Invalid input.

422 - Unprocessable Entity - Unable to process the contained instructions.

GET /monitoring/{displayName}/zebra

Get the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/zebra

Set the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Zebra enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/focusAssist

Get the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/focusAssist

Set the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/focusAssist

Get the focus assist settings.

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

404 - Display name not found.

PUT /monitoring/focusAssist

Set the focus assist settings.

Parameters

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/frameGuide

Get the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame guide enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGuide

Set the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame guide state updated successfully.

422 - Unable to update the frame guide state.

GET /monitoring/frameGuideRatio

Get the current frame guide ratio.

Response

200 - Returns the current frame guide ratio.

The response is JSON.

Name	Type	Description
ratio	string	The frame guide ratio.

PUT /monitoring/frameGuideRatio

Set the frame guide ratio.

Parameters

Name	Type	Description
ratio	string	The frame guide ratio.

Response

204 - Frame guide ratio updated successfully.

422 - Unable to update the frame guide ratio.

GET /monitoring/frameGuideRatio/presets

Get the presets for frame guide ratios.

Response

200 - Returns a list of preset frame guide ratios.

The response is JSON.

Name	Type	Description
presets	array	
presets[i]	string	A frame guide ratio.

GET /monitoring/{displayName}/frameGrids

Get the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame grids enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGrids

Set the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame grids state updated successfully.

422 - Unable to update the frame grids state.

GET /monitoring/frameGrids

Get the global frame grids settings.

Response

200 - Returns the current frame grids settings.

The response is JSON.

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

PUT /monitoring/frameGrids

Set the global frame grids settings.

Parameters

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

Response

204 - Frame grids settings updated successfully.

400 - Invalid input, check the number of frame grids or values.

422 - Unable to update the frame grids settings.

GET /monitoring/{displayName}/safeArea

Get the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the safe area enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/safeArea

Set the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Safe area state updated successfully.

422 - Unable to update the safe area state.

GET /monitoring/safeAreaPercent

Get the current safe area percentage.

Response

200 - Returns the current safe area percentage.

The response is JSON.

Name	Type	Description
percent	integer	Safe area coverage percentage.

PUT /monitoring/safeAreaPercent

Set the safe area percentage.

Parameters

Name	Type	Description
percent	integer	Safe area coverage percentage to set.

Response

204 - Safe area percentage updated successfully.

400 - Invalid percentage value.

422 - Unable to update the safe area percentage.

GET /monitoring/{displayName}/falseColor

Get the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the false color enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/falseColor

Set the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - False color state updated successfully.

422 - Unable to update the false color state.

Event Control API

API For working with built-in websocket.

GET /event/list

Get the list of events that can be subscribed to using the websocket API.

Response

200 - Websocket API events list.

The response is JSON.

Name	Type	Description
events	array	List of events that can be subscribed to using the websocket API.
events[i]	string	

System Control API

API for controlling the System Modes on Blackmagic Design products.

GET /system

Get device system information.

Response

200 - System summary.

The response is JSON.

The properties will be populated only with the values that are supported/implemented by the device in use.

Name	Type	Description
codecFormat	object	Codec format configuration.
codecFormat.codec	string	Codec serialised as string.
codecFormat.container	string	Multimedia container format.
videoFormat	object	Video format configuration.
videoFormat.name (required)	string	Video format serialised as a string.
videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
videoFormat.height	number	Height dimension of video format.
videoFormat.width	number	Width dimension of video format.
videoFormat.interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/product

Get device product information.

Response

200 - Device product information.

The response is JSON.

Product information.

Name	Type	Description
deviceName	string	Name of device as displayed in Setup.
productName	string	Device's product name.
softwareVersion	string	Software version running on device.

501 - This functionality is not implemented for the device in use.

GET /system/supportedCodecFormats

Get the list of supported codecs.

Response

200 - List of supported codec formats.

The response is JSON.

Name	Type	Description
codecs	array	
codecs[i]	object	Codec format configuration.
codecs[i].codec	string	Codec serialised as string.
codecs[i].container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

GET /system/codecFormat

Get the currently selected codec.

Response

200 - Current codec format.

The response is JSON.

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

PUT /system/codecFormat

Update the system codec.

Parameters

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

Response

204 - The codec updated successfully.

400 - The specified codec format is unsupported.

501 - This functionality is not implemented for the device in use.

GET /system/videoFormat

Get the currently selected video format.

Response

200 - Current system video format.

The response is JSON.

Video format configuration.

Name	Type	Description
name (required)	string	Video format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height	number	Height dimension of video format.
width	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

PUT /system/videoFormat

Set the system video format.

Parameters

This parameter can be one of the following types:

Name	Type	Description
name (required)	string	Video format serialised as a string.

Name	Type	Description
frameRate (required)	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height (required)	number	Height dimension of video format.
width (required)	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

Response

204 - The video format updated successfully.

400 - Invalid request.

409 - Operation unsupported in the current state.

501 - This functionality is not implemented for the device in use.

GET /system/supportedVideoFormats

Get the list of supported video formats for the current system state.

Response

200 - List of supported video formats.

The response is JSON.

List of supported video formats.

Name	Type	Description
formats	array	List of video formats.
formats[i]	object	Video format configuration.
formats[i].name (required)	string	Video format serialised as a string.
formats[i].frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
formats[i].height	number	Height dimension of video format.
formats[i].width	number	Width dimension of video format.
formats[i].interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/supportedFormats

Get supported formats.

Response

200 - List of supported formats.

The response is JSON.

Name	Type	Description
supportedFormats	array	
supportedFormats[i]	object	
supportedFormats[i].codecs	array	
supportedFormats[i].codecs[i]	string	
supportedFormats[i].frameRates	array	
supportedFormats[i].frameRates[i]	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
supportedFormats[i]. maxOffSpeedFrameRate	number	
supportedFormats[i]. minOffSpeedFrameRate	number	
supportedFormats[i]. recordResolution	object	
supportedFormats[i]. recordResolution.height	number	Height of the resolution.
supportedFormats[i]. recordResolution.width	number	Width of the resolution.
supportedFormats[i]. sensorResolution	object	
supportedFormats[i]. sensorResolution.height	number	Height of the resolution.
supportedFormats[i]. sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

GET /system/format

Get current format.

Response

200 - Current format.

The response is JSON.

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

PUT /system/format

Set the format.

Parameters

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

Response

204 - System format updated.

501 - This functionality is not implemented for the device in use.

Transport Control API

API for controlling Transport on Blackmagic Design products.

GET /transports/0

Get device's basic transport status.

Response

200 - Transport status.

The response is JSON.

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

PUT /transports/0

Set device's basic transport status.

Parameters

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, Output.

Response

204 - Transport mode was set.

400 - Failed to set transport mode.

GET /transports/0/stop

Determine if transport is stopped.

Response

200 - Transport stop response.

The response is JSON.

Name	Type	Description
	boolean	True when transport mode is InputPreview or when in Output mode and speed is 0.

PUT /transports/0/stop

Stop transport. Deprecated, use POST /transports/0/stop instead.

Response

204 - Transport stopped.

POST /transports/0/stop

Stop transport.

Response

204 - Transport stopped.

GET /transports/0/play

Determine if transport is playing.

Response

200 - Transport play response.

The response is JSON.

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

PUT /transports/0/play

Start playing on transport. Deprecated, use POST /transports/0/play instead.

Response

204 - Transport playing.

400 - Failed to set transport to play.

POST /transports/0/play

Start playing on transport.

Response

204 - Transport playing.

400 - Failed to set transport to play.

GET /transports/0/playback

Get playback state.

Response

200 - Transport playback state.

The response is JSON.

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

PUT /transports/0/playback

Set playback state.

Parameters

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

Response

204 - Updated transport playback state.

400 - Failed to set transport playback state.

GET /transports/0/record

Get record state.

Response

200 - Recording state.

The response is JSON.

Name	Type	Description
recording	boolean	If true, transport is in InputRecord mode.

PUT /transports/0/record

Set record state. Deprecated, use POST /transports/0/record instead.

Parameters

Name	Type	Description
recording (required)	boolean	If true, starts a recording, otherwise stops.
clipName	string	Optional, sets the requested clip name to record to, when “recording” attribute is set to true.

Response

204 - Recording state updated.

400 - Failed to update recording state.

POST /transports/0/record

Start recording.

Parameters

Name	Type	Description
clipName	string	Optional, provides a specific name of clip to record to.

Response

204 - Recording started.

400 - Failed to start recording.

GET /transports/0/clipIndex

Get the clip index of the currently playing clip on the timeline.

Response

200 - Clip index response.

The response is JSON.

Name	Type	Description
clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

GET /transports/0/timecode

Get device timecode.

Response

200 - Timecode response.

The response is JSON.

Name	Type	Description
display	string	The display timecode serialised as a string.
timeline	string	The timeline timecode serialised as a string.

GET /transports/0/timecode/source

Get timecode source selected on device.

Response

200 - Timecode source response.

The response is JSON.

Name	Type	Description
timecode	string	Possible values are: Timeline, Clip.

Timeline Control API

API for controlling playback timeline.

GET /timelines/0

Get the playback timeline.

Response

200 - Playback timeline.

The response is JSON.

Name	Type	Description
clips	array	
clips[i]	object	Timeline clip.
clips[i].clipUniqueId (required)	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).
clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

404 - No timeline / disk available.

DELETE /timelines/0

Clear the current playback timeline. Deprecated, prefer to use POST /timelines/0/clear

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

POST /timelines/0

Add a clip to the timeline.

Parameters

This parameter can be one of the following types:

Add multiple media clips to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips[i].clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips[i].clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips[i].frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add multiple media clips to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add a single clip to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips.clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips.clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips.frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add a single clip to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the timeline as specified.

501 - The operation is not supported on this device.

POST /timelines/0/add

Add a clip to the end of the timeline. Deprecated, use POST /timelines/0 to add clips within the timeline.

Parameters

This parameter can be one of the following types:

Add one clip to the end of the timeline.

Name	Type	Description
clips	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add many clips to the end of the timeline.

Name	Type	Description
clips	array	List of clipUniqueIds of clips to add to end of timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the end of the timeline.

501 - The operation is not supported on this device.

POST /timelines/0/clear

Clear the playback timeline.

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

DELETE /timelines/0/clips/{timelineClipIndex}

Remove the specified clip from the timeline.

Parameters

Name	Type	Description
{timelineClipIndex} (required)	integer	The (0-based) timeline clip index of the clip to remove from the timeline.

Response

204 - The specified clip was removed from the timeline.

501 - The operation is not supported on this device.

Media Control API

API for controlling media devices in Blackmagic Design products.

GET /media/workingset

Get the list of media devices currently in the working set.

Response

200 - The list of media devices in the working set.

The response is JSON.

Name	Type	Description
size	integer	The fixed size of this device's working set.
workingset	array	The device's working set.
workingset[i]	object | null	Device within the working set. null if no device is present within the given working set slot.
workingset[i].volume	string	Volume name.
workingset[i].deviceName	string	Internal device name of this media device.
workingset[i].remainingRecordTime	integer	Remaining record time using current codec and video format in seconds.
workingset[i].totalSpace	integer	Total space on media device in bytes.
workingset[i].remainingSpace	integer	Remaining space on media device in bytes.
workingset[i].clipCount	integer	Number of clips currently on the device.

GET /media/active

Get the currently active media device.

Response

200 - The current active media device.

The response is JSON.

The active media device, or null if there is no active media.

Name	Type	Description
workingsetIndex	integer	Working set index of the active media device.
deviceName	string	Device name of media device.

204 - No media is currently active.

PUT /media/active

Set the currently active media device.

Parameters

Name	Type	Description
workingsetIndex	integer	Working set index of the media to make active.

Response

204 - The active media device was set successfully.

400 - Setting the currently active media device is not possible in the current state.

GET /media/devices/doformatSupportedFilesystems

Get the list of filesystems available to format a media device.

Response

200 - The list of filesystems permitted for formatting.

The response is JSON.

Name	Type	Description
	array	List of filesystems permitted for formatting media.
[i]	string	Filesystem serialised as string.

GET /media/devices/{deviceName}

Get information about a requested device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Information about the requested device.

The response is JSON.

Media device state.

Name	Type	Description
state	string	The current state of the media device. Possible values are: None, Scanning, Mounted, Uninitialised, Formatting, RaidComponent.

400 - Invalid device name.

404 - Device not found.

GET /media/devices/{deviceName}/doformat

Get a format key, used to format the device with a PUT request.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Format prepared.

The response is JSON.

Name	Type	Description
deviceName	string	Device name of media device to format.
key	string	The key required to format this device, provide to PUT /media/devices/{deviceName}/doformat to perform format of media device.

400 - Cannot format the device.

404 - Device not found.

PUT /media/devices/{deviceName}/doformat

Perform a format of the specified media device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Name	Type	Description
key	string	The key used to format this device, retrieved from prepare format media request GET /media/devices/{deviceName}/doformat. Format key provided cannot be reused after successful format.
filesystem	string	Filesystem to format to. Supported filesystems can be retrieved with GET /media/devices/doFormatSupportedFilesystems.
volume	string	Volume name to set for the disk after format.

Response

204 - Format successful.

400 - Cannot format the device, invalid filesystem or key.

404 - Device not found.

Slate Control API

API to manage digital slate data.

GET /slates/nextClip

Retrieve the digital slate for the next clip.

Response

200 - Returns the slate data for the next clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	

Name	Type	Description
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Slate data is not available.

PUT /slates/nextClip

Update the slate data for the next clip.

Parameters

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

POST /slates/nextClip/resetProjectData

Reset the project data for the next clip's slate.

Response

200 - Project data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetProjectData

Reset the project data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Project data reset successfully.

POST /slates/nextClip/resetLensData

Reset the lens data for the next clip's slate.

Response

200 - Lens data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetLensData

Reset the lens data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Lens data reset successfully.

GET /slates/clips/{deviceName}/{path}

Retrieve slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Returns the slate data for the specified clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

404 - Clip not found.

PUT /slates/clips/{deviceName}/{path}

Update the slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.

Name	Type	Description
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

Preset Control API

API For controlling the presets on Blackmagic Design products

GET /presets

Get the list of the presets on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presets	array	List of the presets on the camera (.cset files)
presets[i]	string	

POST /presets

Send a preset file to the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presetAdded	string	Name of the preset uploaded (without .cset extension)

400 - Bad request - missing Content-Disposition header or filename

GET /presets/active

Get the currently active preset on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

PUT /presets/active

Set the active preset on the camera

Parameters

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

Response

204 - No Content

404 - Preset file not found

PARAMETERS /presets/{presetName}

GET /presets/{presetName}

Download the preset file

Response

200 - OK

The response is a binary file.

404 - File does not exist

PUT /presets/{presetName}

Save current camera state as a preset

Response

204 - No Content

DELETE /presets/{presetName}

Delete a preset from the camera

Response

204 - No Content

404 - Preset file not found

Audio Control API

API For controlling audio on Blackmagic Design Cameras

GET /audio/channels

Get the total number of audio channels available

Response

200 - Returns the total number of channels

The response is JSON.

Name	Type	Description
channels	integer	Total number of audio channels available

GET /audio/supportedInputs

Get the list of supported audio inputs

Response

200 - List of all supported audio inputs

The response is JSON.

Name	Type	Description
	array	
[i]	string	A supported audio input

GET /audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently selected input

The response is JSON.

Name	Type	Description
input	string	Audio input source and type

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/input

Set the audio input for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
input	string	Audio input source and type

Response

204 - No Content

400 - Invalid audio input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/input/description

Get the description of the current input of the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Description of the current input of the selected channel

The response is JSON.

Name	Type	Description
description	object	
description.gainRange	object	
description.gainRange.Min	number	The minimum gain value in dB
description.gainRange.Max	number	The maximum gain value in dB
description.capabilities	object	
description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter
description.capabilities.Padding	object	
description.capabilities.Padding. available	boolean	Input supports setting of padding
description.capabilities.Padding. forced	boolean	Padding is forced to be set for the input
description.capabilities.Padding. value	number	Value of the padding in dB

404 - Channel does not exist

GET /audio/channel/{channelIndex}/supportedInputs

Get the list of supported inputs and their availability to switch to for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its supported inputs are being queried. (Channels index from 0)

Response

200 - The list of supported inputs

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

404 - Channel does not exist

GET /audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set level for the selected channel

The response is JSON.

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/level

Set the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

Response

204 - No Content

400 - Invalid input or value out of range

404 - Channel does not exist

GET /audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set phantom power for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Phantom power enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/phantomPower

Set the audio phantom power for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Phantom power enabled state

Response

204 - No Content

400 - Phantom power is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set padding for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Padding enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/padding

Set the audio input padding for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Padding enabled state

Response

204 - No Content

400 - Padding is not supported or is forced for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set low cut filter for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Low cut filter enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/lowCutFilter

Set the audio input low cut filter for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Low cut filter enabled state

Response

204 - No Content

400 - Low cut filter is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set availability for the selected channel

The response is JSON.

Name	Type	Description
available	boolean	Whether the input is currently available

404 - Channel does not exist

Lens Control API

API For controlling the lens on Blackmagic Design products

GET /lens/iris

Get lens' aperture

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number

PUT /lens/iris

Set lens' aperture

Parameters

Name	Type	Description
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number
adjustmentStep	integer	Signed value for relative aperture adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens iris is not controllable or is controlled by auto exposure

GET /lens/zoom

Get lens' zoom

Response

200 - OK

The response is JSON.

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value

PUT /lens/zoom

Set lens' zoom

Parameters

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value
adjustmentFocalLength	integer	Signed value for relative focal length adjustment
adjustmentNormalised	number	Signed normalized value for relative zoom adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens zoom is not controllable

GET /lens/focus

Get lens' focus

Response

200 - OK

The response is JSON.

Name	Type	Description
normalised	number	Normalised value

PUT /lens/focus

Set lens' focus

Parameters

Name	Type	Description
normalised	number	Normalised value
focusDistance	integer	Focus distance value

Response

204 - No Content

400 - Bad Request if out of range value is provided

PUT /lens/focus/doAutoFocus

Perform auto focus

Parameters

Name	Type	Description
position	object	
position.x	number	Normalized x coordinate for autofocus ROI
position.y	number	Normalized y coordinate for autofocus ROI

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens focus is not controllable

GET /lens/opticalImageStabilization

Get optical image stabilization status

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Whether optical image stabilization is enabled

501 - Not Implemented if optical image stabilization is not supported on this product

PUT /lens/opticalImageStabilization

Enable or disable optical image stabilization

Parameters

Name	Type	Description
enabled	boolean	Enable or disable optical image stabilization

Response

204 - No Content

501 - Not Implemented if optical image stabilization is not supported on this product

GET /lens/iris/description

Get detailed description of lens' iris capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the iris can be controlled
apertureStop	object	
apertureStop.min	number	Minimum aperture stop
apertureStop.max	number	Maximum aperture stop

GET /lens/zoom/description

Get detailed description of lens' zoom capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the zoom can be controlled
focalLength	object	
focalLength.adjustable	boolean	If focal length is adjustable
focalLength.min	integer	Minimum focal length
focalLength.max	integer	Maximum focal length

GET /lens/focus/description

Get detailed description of lens' focus capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the focus can be controlled
focusDistance	object	
focusDistance.adjustable	boolean	If focus distance is adjustable
focusDistance.min	number	Minimum focus distance
focusDistance.max	number	Maximum focus distance

Video Control API

API For controlling the video on Blackmagic Design products

GET /video/iso

Get current ISO

Response

200 - OK

The response is JSON.

Name	Type	Description
iso	integer	Current ISO value

PUT /video/iso

Set current ISO

Parameters

Name	Type	Description
iso	integer	ISO value to set

Response

204 - No Content

403 - ISO cannot be changed in the current state

GET /video/supportedISOs

Get the list of supported ISO settings

Response

200 - List of supported ISO values

The response is JSON.

Name	Type	Description
supportedISOs	array	Array of supported ISO values
supportedISOs[i]	integer	

GET /video/gain

Get current gain value in decibels

Response

200 - OK

The response is JSON.

Name	Type	Description
gain	integer	Current gain value in decibels

PUT /video/gain

Set current gain value

Parameters

Name	Type	Description
gain	integer	Gain value in decibels to set

Response

204 - No Content

403 - Gain cannot be changed in the current state

GET /video/supportedGains

Get the list of supported gain settings in decibels

Response

200 - List of supported gain values in decibels

The response is JSON.

Name	Type	Description
supportedGains	array	Array of supported gain values in decibels
supportedGains[i]	integer	

GET /video/whiteBalance

Get current white balance

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	integer	Current white balance

PUT /video/whiteBalance

Set current white balance

Parameters

Name	Type	Description
whiteBalance	integer	White balance to set

Response

204 - No Content

400 - Invalid white balance temperature

GET /video/whiteBalance/description

Get white balance range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	object	
whiteBalance.min	integer	Minimum color temperature
whiteBalance.max	integer	Maximum color temperature

PUT /video/whiteBalance/doAuto

Set current white balance automatically

Response

204 - No Content

GET /video/whiteBalanceTint

Get white balance tint

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	integer	Current white balance tint

PUT /video/whiteBalanceTint

Set white balance tint

Parameters

Name	Type	Description
whiteBalanceTint	integer	White balance tint to set

Response

204 - No Content

400 - Invalid white balance tint

GET /video/whiteBalanceTint/description

Get white balance tint range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	object	
whiteBalanceTint.min	integer	Minimum white balance tint
whiteBalanceTint.max	integer	Maximum white balance tint

GET /video/ndFilter

Get ND filter stop

Response

200 - OK

The response is JSON.

Name	Type	Description
stop	number	Current filter power (fStop)

501 - Not implemented for this device

PUT /video/ndFilter

Set ND filter stop

Parameters

Name	Type	Description
stop	number	Filter power (fStop) to set

Response

204 - No Content

400 - Invalid ND filter stop

501 - Not implemented for this device

GET /video/supportedNDFilters

Get the list of available ND filter stops

Response

200 - List of available ND filter stops

The response is JSON.

Name	Type	Description
supportedStops	array	Array of available ND filter stops
supportedStops[i]	number	

501 - Not implemented for this device

GET /video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

Response

200 - List of supported display modes

The response is JSON.

Name	Type	Description
supportedDisplayModes	array	Array of supported display modes
supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

GET /video/ndFilter/displayMode

Get ND filter display mode on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

PUT /video/ndFilter/displayMode

Set ND filter display mode on the camera

Parameters

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

Response

204 - No Content

400 - Invalid display mode for ND filter

501 - Not implemented for this device

GET /video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

Response

200 - Indicates if ND filter is selectable

The response is JSON.

Name	Type	Description
selectable	boolean	True if ND filter adjustments are selectable via a slider

501 - Not implemented for this device

GET /video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

PUT /video/shutter

Set current shutter

Parameters

Name	Type	Description
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

Response

204 - No Content

GET /video/shutter/measurement

Get the current shutter measurement mode

Response

200 - OK

The response is JSON.

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

PUT /video/shutter/measurement

Set the shutter measurement mode

Parameters

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

Response

204 - No Content

400 - Invalid shutter measurement

GET /video/supportedShutters

Get supported shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of supported shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of supported shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of flicker-free shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of flicker-free shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/autoExposure

Get current auto exposure mode

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

PUT /video/autoExposure

Set auto exposure

Parameters

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

Response

204 - No Content

400 - Failed to set auto exposure mode

GET /video/detailSharpening

Get the current state of detail sharpening

Response

200 - Current detail sharpening state

The response is JSON.

Name	Type	Description
enabled	boolean	Whether detail sharpening is enabled

501 - Not implemented for this device

PUT /video/detailSharpening

Enable or disable detail sharpening

Parameters

Name	Type	Description
enabled	boolean	Enable or disable detail sharpening

Response

204 - Detail sharpening state updated

501 - Not implemented for this device

GET /video/detailSharpeningLevel

Get the current detail sharpening level

Response

200 - Current detail sharpening level

The response is JSON.

Name	Type	Description
level	string	Current detail sharpening level Possible values are: Low, Medium, High.

501 - Not implemented for this device

PUT /video/detailSharpeningLevel

Set the detail sharpening level

Parameters

Name	Type	Description
level	string	Desired level of detail sharpening Possible values are: Low, Medium, High.

Response

204 - Detail sharpening level updated

400 - Invalid detail sharpening level

501 - Not implemented for this device

Camera Control API

API For controlling the Camera specific features on Blackmagic Design products

GET /camera/colorBars

Get the status of color bars display

Response

200 - Returns the current status of color bars

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the color bars are currently enabled

PUT /camera/colorBars

Set the status of color bars display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the color bars

Response

204 - Color bars status updated successfully

GET /camera/programFeedDisplay

Get the status of program feed display

Response

200 - Returns the current status of program feed display

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the program feed display is currently enabled

PUT /camera/programFeedDisplay

Set the status of program feed display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the program feed display

Response

204 - Program feed display status updated successfully

GET /camera/tallyStatus

Get the tally status of the camera

Response

200 - Returns the current tally status of the camera

The response is JSON.

Name	Type	Description
status	string	Current tally status of the camera Possible values are: None, Preview, Program.

GET /camera/power

Get the power status of the camera

Response

200 - Returns the current power status

The response is JSON.

Name	Type	Description
source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
batteries	array	List of batteries currently connected to the camera
batteries[i]	object	
batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
batteries[i].chargeRemainingPercent	integer	Remaining battery charge percentage
batteries[i].statusFlags	array	List of battery status flags
batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

GET /camera/power/displayMode

Get the power display mode of the camera

Response

200 - Returns the current power display mode

The response is JSON.

Name	Type	Description
mode	string	Current power display mode Possible values are: Percentage, Voltage.

PUT /camera/power/displayMode

Set the power display mode of the camera

Parameters

Name	Type	Description
mode	string	Power display mode to set Possible values are: Percentage, Voltage.

Response

204 - Power display mode updated successfully

400 - Invalid power display mode

GET /camera/timingReferenceLock

Get the timing reference lock status

Response

200 - Returns the timing reference lock status

The response is JSON.

Name	Type	Description
locked	boolean	Indicates if timing reference is locked

Immersive Control API

API for controlling immersive camera settings on Blackmagic Design cameras

GET /immersive/display/{displayName}/eye

Get the current eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to query (from /monitoring/display endpoint)

Response

200 - OK

The response is JSON.

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

400 - Invalid display name format

404 - Display not found

422 - Failed to get eye view

PUT /immersive/display/{displayName}/eye

Set the eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to control (from /monitoring/display endpoint)

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

Response

204 - No Content

400 - Invalid input or display name format

404 - Display not found

422 - Failed to set eye view

Color Correction Control API

API For controlling the color correction on Blackmagic Design products based on DaVinci Resolve Color Corrector

GET /colorCorrection/lift

Get color correction lift

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

PUT /colorCorrection/lift

Set color correction lift

Parameters

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gamma

Get color correction gamma

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

PUT /colorCorrection/gamma

Set color correction gamma

Parameters

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gain

Get color correction gain

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

PUT /colorCorrection/gain

Set color correction gain

Parameters

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/offset

Get color correction offset

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

PUT /colorCorrection/offset

Set color correction offset

Parameters

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/contrast

Get color correction contrast

Response

200 - OK

The response is JSON.

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/contrast

Set color correction contrast

Parameters

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/color

Get color correction color properties

Response

200 - OK

The response is JSON.

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/color

Set color correction color properties

Parameters

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/lumaContribution

Get color correction luma contribution

Response

200 - OK

The response is JSON.

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

PUT /colorCorrection/lumaContribution

Set color correction luma contribution

Parameters

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

Response

204 - No Content

Notification websocket - 1.0.0

Service that notifies subscribers of device state changes.

messages

Subscribe (The messages from the server/device)

Websocket Opened Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: websocketOpened.
.type	string	Possible values are: event.

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*

Name	Type	Description
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Event Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: propertyValueChanged.

Name	Type	Description
.data.property	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.value	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.type	string	Possible values are: event.

Publish (The messages that user can send to the server/device)

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	

Name	Type	Description
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	

Name	Type	Description
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Device Properties

/media/workingset

The value JSON returned via the eventResponse when the `/media/workingset` property changes on the device:

Name	Type	Description
.size	integer	The fixed size of this device's working set.
.workingset	array	Array of devices within the working set. null if no device is present within the given working set slot.
.workingset[i]		

/media/active

The value JSON returned via the eventResponse when the `/media/active` property changes on the device:

Name	Type	Description
.workingsetIndex	integer	Working set index of the active media device.
.deviceName	string	Internal device name of this media device.

/system

The value JSON returned via the eventResponse when the `/system` property changes on the device:

Name	Type	Description
.codecFormat	object	Codec format configuration.
.codecFormat.codec	string	Codec format serialised as a string.
.codecFormat.container	string	Multimedia container format.
.videoFormat	object	Currently selected video format.
.videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.videoFormat.height	number	Height dimension of video format.
.videoFormat.width	number	Width dimension of video format.
.videoFormat.interlaced	boolean	Is the display format interlaced?.
.videoFormat.name	string	Video format serialised as a string.

/system/codecFormat

Codec format configuration.

The value JSON returned via the eventResponse when the `/system/codecFormat` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.container	string	Multimedia container format.

/system/videoFormat

Currently selected video format.

The value JSON returned via the eventResponse when the `/system/videoFormat` property changes on the device:

Name	Type	Description
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.height	number	Height dimension of video format.
.width	number	Width dimension of video format.
.interlaced	boolean	Is the display format interlaced?.
.name	string	Video format serialised as a string.

/system/format

The value JSON returned via the eventResponse when the `/system/format` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.maxOffSpeedFrameRate	number	
.minOffSpeedFrameRate	number	
.offSpeedEnabled	boolean	
.offSpeedFrameRate	number	
.recordResolution	object	
.recordResolution.height	number	Height of the resolution.
.recordResolution.width	number	Width of the resolution.
.sensorResolution	object	
.sensorResolution.height	number	Height of the resolution.
.sensorResolution.width	number	Width of the resolution.

/system/supportedFormats

The value JSON returned via the eventResponse when the `/system/supportedFormats` property changes on the device:

Name	Type	Description
.supportedFormats	array	
.supportedFormats[i]	object	
.supportedFormats[i].codecs	array	
.supportedFormats[i].codecs[i]	string	
.supportedFormats[i].frameRates	array	
.supportedFormats[i].frameRates[i]	string	Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.supportedFormats[i].maxOffSpeedFrameRate	number	
.supportedFormats[i].minOffSpeedFrameRate	number	
.supportedFormats[i].recordResolution	object	
.supportedFormats[i].recordResolution.height	number	Height of the resolution.
.supportedFormats[i].recordResolution.width	number	Width of the resolution.
.supportedFormats[i].sensorResolution	object	
.supportedFormats[i].sensorResolution.height	number	Height of the resolution.
.supportedFormats[i].sensorResolution.width	number	Width of the resolution.

/timelines/0

The value JSON returned via the eventResponse when the `/timelines/0` property changes on the device:

Name	Type	Description
.clips	array	
.clips[i]	object	Timeline clip.
.clips[i].clipUniqueId	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
.clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
.clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
.clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
.clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).

Name	Type	Description
.clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
.clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

/transports/0

The value JSON returned via the eventResponse when the `/transports/0` property changes on the device:

Name	Type	Description
.mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

/transports/0/stop

true when transport mode is InputPreview or when in Output mode and speed is 0.

The value JSON returned via the eventResponse when the `/transports/0/stop` property changes on the device:

Name	Type	Description
	boolean	true when transport mode is InputPreview or when in Output mode and speed is 0.

/transports/0/play

True when transport is in Output mode and speed is non-zero.

The value JSON returned via the eventResponse when the /transports/0/play property changes on the device:

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

/transports/0/playback

The value JSON returned via the eventResponse when the `/transports/0/playback` property changes on the device:

Name	Type	Description
.type	string	Possible values are: Play, Jog, Shuttle, Var.
.loop	boolean	When true playback loops from the end of the timeline to the beginning of the timeline.
.singleClip	boolean	When true playback loops from the end of the current clip to the beginning of the current clip.
.speed	number	Playback speed, 1.0 for normal forward playback
.position	integer	Playback position on the timeline in units of video frames.

/transports/0/record

The value JSON returned via the eventResponse when the `/transports/0/record` property changes on the device:

Name	Type	Description
.recording	boolean	Is transport in Input Record mode.

/transports/0/timecode

The value JSON returned via the eventResponse when the `/transports/0/timecode` property changes on the device:

Name	Type	Description
.display	string	The display timecode serialised as a string.
.timeline	string	The timeline timecode serialised as a string.

/transports/0/timecode/source

The value JSON returned via the eventResponse when the `/transports/0/timecode/source` property changes on the device:

Name	Type	Description
.timecode	string	Possible values are: Timeline, Clip.

/transports/0/clipIndex

The value JSON returned via the eventResponse when the `/transports/0/clipIndex` property changes on the device:

Name	Type	Description
.clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

/slates/nextClip

The value JSON returned via the eventResponse when the `/slates/nextClip` property changes on the device:

Name	Type	Description
.clip	object	
.clip.clipName	string	
.clip.reel	integer	
.clip.scene	string	
.clip.sceneLocation	string	Possible values are: Interior, Exterior.
.clip.sceneTime	string	Possible values are: Day, Night.
.clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
.clip.take	integer	
.clip.takeType	string	Possible values are: None, PU, VFX, SER.
.clip.goodTake	boolean	
.lens	object	
.lens.lensType	string	
.lens.iris	string	

Name	Type	Description
.lens.focalLength	string	
.lens.distance	string	
.lens.filter	string	
.project	object	
.project.projectName	string	
.project.director	string	
.project.camera	string	
.project.cameraOperator	string	

/monitoring/{displayName}/cleanFeed

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/cleanFeed` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/displayLUT

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/displayLUT` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/zebra

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/zebra` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/focusAssist` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGuide

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGuide` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGrids` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/safeArea

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/safeArea` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/falseColor

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/falseColor` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/focusAssist` property changes on the device:

Name	Type	Description
.mode	string	Possible values are: Peak, ColoredLines.
.color	string	Possible values are: Red, Green, Blue, White, Black.
.intensity	integer	

/monitoring/frameGuideRatio

The value JSON returned via the eventResponse when the `/monitoring/frameGuideRatio` property changes on the device:

Name	Type	Description
.ratio	string	

/monitoring/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/frameGrids` property changes on the device:

Name	Type	Description
.frameGrids	array	
.frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

/monitoring/safeAreaPercent

The value JSON returned via the eventResponse when the `/monitoring/safeAreaPercent` property changes on the device:

Name	Type	Description
.percent	integer	Safe area coverage percentage.

/audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input` property changes on the device:

Name	Type	Description
.input	string	Audio input source and type

/audio/channel/{channelIndex}/supportedInputs

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/supportedInputs` property changes on the device:

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

/audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/level` property changes on the device:

Name	Type	Description
.gain	number	Gain value in dB
.normalised	number	Normalised level value between 0.0 and 1.0

/audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/phantomPower` property changes on the device:

Name	Type	Description
.enabled	boolean	Phantom power enabled state

/audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/padding` property changes on the device:

Name	Type	Description
.enabled	boolean	Padding enabled state

/audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/lowCutFilter` property changes on the device:

Name	Type	Description
.enabled	boolean	Low cut filter enabled state

/audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/available` property changes on the device:

Name	Type	Description
.available	boolean	Whether the input is currently available

/audio/channel/{channelIndex}/input/description

Description of the current input of the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input/description` property changes on the device:

Name	Type	Description
.description	object	
.description.gainRange	object	
.description.gainRange.Min	number	The minimum gain value in dB
.description.gainRange.Max	number	The maximum gain value in dB
.description.capabilities	object	
.description.capabilities.PhantomPower	boolean	Input supports setting of phantom power
.description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter

Name	Type	Description
.description.capabilities.Padding	object	
.description.capabilities.Padding.available	boolean	Input supports setting of padding
.description.capabilities.Padding.forced	boolean	Padding is forced to be set for the input
.description.capabilities.Padding.value	number	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.

/colorCorrection/lift

Get color correction lift

The value JSON returned via the eventResponse when the `/colorCorrection/lift` property changes on the device:

Name	Type	Description
.red	number	Red lift component
.green	number	Green lift component
.blue	number	Blue lift component
.luma	number	Luma lift component

/colorCorrection/gamma

Get color correction gamma

The value JSON returned via the eventResponse when the `/colorCorrection/gamma` property changes on the device:

Name	Type	Description
.red	number	Red gamma component
.green	number	Green gamma component
.blue	number	Blue gamma component
.luma	number	Luma gamma component

/colorCorrection/gain

Get color correction gain

The value JSON returned via the eventResponse when the `/colorCorrection/gain` property changes on the device:

Name	Type	Description
.red	number	Red gain component
.green	number	Green gain component
.blue	number	Blue gain component
.luma	number	Luma gain component

/colorCorrection/offset

Get color correction offset

The value JSON returned via the eventResponse when the `/colorCorrection/offset` property changes on the device:

Name	Type	Description
.red	number	Red offset component
.green	number	Green offset component
.blue	number	Blue offset component
.luma	number	Luma offset component

/colorCorrection/contrast

Get color correction contrast

The value JSON returned via the eventResponse when the `/colorCorrection/contrast` property changes on the device:

Name	Type	Description
.pivot	number	Contrast pivot point
.adjust	number	Contrast adjustment

/colorCorrection/color

Get color correction color properties

The value JSON returned via the eventResponse when the `/colorCorrection/color` property changes on the device:

Name	Type	Description
.hue	number	Color hue adjustment
.saturation	number	Color saturation adjustment

/colorCorrection/lumaContribution

Get color correction luma contribution

The value JSON returned via the eventResponse when the `/colorCorrection/lumaContribution` property changes on the device:

Name	Type	Description
.lumaContribution	number	Luma contribution value

/lens/iris

Get lens' aperture

The value JSON returned via the eventResponse when the `/lens/iris` property changes on the device:

Name	Type	Description
.continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
.apertureStop	number	Aperture stop value
.normalised	number	Normalised value
.apertureNumber	integer	Aperture number

/lens/iris/description

Get detailed description of lens' iris capabilities

The value JSON returned via the eventResponse when the `/lens/iris/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the iris can be controlled
.apertureStop	object	
.apertureStop.min	number	Minimum aperture stop
.apertureStop.max	number	Maximum aperture stop

/lens/focus

Get lens' focus

The value JSON returned via the eventResponse when the `/lens/focus` property changes on the device:

Name	Type	Description
.normalised	number	Normalised value

/lens/focus/description

Get detailed description of lens' focus capabilities

The value JSON returned via the eventResponse when the `/lens/focus/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the focus can be controlled
.focusDistance	object	
.focusDistance.adjustable	boolean	If focus distance is adjustable
.focusDistance.min	number	Minimum focus distance
.focusDistance.max	number	Maximum focus distance

/lens/zoom

Get lens' zoom

The value JSON returned via the eventResponse when the `/lens/zoom` property changes on the device:

Name	Type	Description
.focalLength	integer	Focal length in mm
.normalised	number	Normalised value

/lens/zoom/description

Get detailed description of lens' zoom capabilities

The value JSON returned via the eventResponse when the `/lens/zoom/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the zoom can be controlled
.focalLength	object	
.focalLength.adjustable	boolean	If focal length is adjustable
.focalLength.min	integer	Minimum focal length
.focalLength.max	integer	Maximum focal length

/presets

Get the list of the presets on the camera

The value JSON returned via the eventResponse when the `/presets` property changes on the device:

Name	Type	Description
.presets	array	List of the presets on the camera (.cset files)
.presets[i]	string	

/presets/active

Get the currently active preset on the camera

The value JSON returned via the eventResponse when the `/presets/active` property changes on the device:

Name	Type	Description
.preset	string	Name of the active preset (with .cset extension, or 'default')

/camera/colorBars

Get the status of color bars display

The value JSON returned via the eventResponse when the `/camera/colorBars` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the color bars are currently enabled

/camera/programFeedDisplay

Get the status of program feed display

The value JSON returned via the eventResponse when the `/camera/programFeedDisplay` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the program feed display is currently enabled

/camera/tallyStatus

Get the tally status of the camera

The value JSON returned via the eventResponse when the `/camera/tallyStatus` property changes on the device:

Name	Type	Description
.status	string	Current tally status of the camera Possible values are: None, Preview, Program.

/camera/power

Get the power status of the camera

The value JSON returned via the eventResponse when the `/camera/power` property changes on the device:

Name	Type	Description
.source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
.milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
.batteries	array	
.batteries[i]	object	
.batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
.batteries[i]. chargeRemainingPercent	integer	Remaining battery charge percentage
.batteries[i].statusFlags	array	List of battery status flags
.batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

/camera/power/displayMode

Get the power display mode of the camera

The value JSON returned via the eventResponse when the `/camera/power/displayMode` property changes on the device:

Name	Type	Description
.mode	string	Current power display mode Possible values are: Percentage, Voltage.

/camera/timingReferenceLock

Get the timing reference lock status

The value JSON returned via the eventResponse when the `/camera/timingReferenceLock` property changes on the device:

Name	Type	Description
.locked	boolean	Indicates if timing reference is locked

/video/iso

Get current ISO

The value JSON returned via the eventResponse when the `/video/iso` property changes on the device:

Name	Type	Description
.iso	integer	Current ISO value

/video/supportedISOs

Get the list of supported ISO settings

The value JSON returned via the eventResponse when the `/video/supportedISOs` property changes on the device:

Name	Type	Description
.supportedISOs	array	Array of supported ISO values
.supportedISOs[i]	integer	

/video/gain

Get current gain value in decibels

The value JSON returned via the eventResponse when the `/video/gain` property changes on the device:

Name	Type	Description
.gain	integer	Current gain value in decibels

/video/supportedGains

Get the list of supported gain settings in decibels

The value JSON returned via the eventResponse when the `/video/supportedGains` property changes on the device:

Name	Type	Description
.supportedGains	array	Array of supported gain values in decibels
.supportedGains[i]	integer	

/video/whiteBalance

Get current white balance

The value JSON returned via the eventResponse when the `/video/whiteBalance` property changes on the device:

Name	Type	Description
.whiteBalance	integer	Current white balance

/video/whiteBalance/description

Get white balance range

The value JSON returned via the eventResponse when the `/video/whiteBalance/description` property changes on the device:

Name	Type	Description
.whiteBalance	object	
.whiteBalance.min	integer	Minimum color temperature
.whiteBalance.max	integer	Maximum color temperature

/video/whiteBalanceTint

Get white balance tint

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint` property changes on the device:

Name	Type	Description
.whiteBalanceTint	integer	Current white balance tint

/video/whiteBalanceTint/description

Get white balance tint range

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint/description` property changes on the device:

Name	Type	Description
.whiteBalanceTint	object	
.whiteBalanceTint.min	integer	Minimum white balance tint
.whiteBalanceTint.max	integer	Maximum white balance tint

/video/ndFilter

Get ND filter stop

The value JSON returned via the eventResponse when the `/video/ndFilter` property changes on the device:

Name	Type	Description
.stop	number	Current filter power (fStop)

/video/supportedNDFilters

Get the list of available ND filter stops

The value JSON returned via the eventResponse when the `/video/supportedNDFilters` property changes on the device:

Name	Type	Description
.supportedStops	array	Array of available ND filter stops
.supportedStops[i]	number	

/video/ndFilter/displayMode

Get ND filter display mode on the camera

The value JSON returned via the eventResponse when the `/video/ndFilter/displayMode` property changes on the device:

Name	Type	Description
.displayMode	string	ND filter display mode Possible values are: Stop, Number, Fraction.

/video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

The value JSON returned via the eventResponse when the `/video/supportedNDFilterDisplayModes` property changes on the device:

Name	Type	Description
.supportedDisplayModes	array	Array of supported display modes
.supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

/video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

The value JSON returned via the eventResponse when the `/video/ndFilterSelectable` property changes on the device:

Name	Type	Description
.selectable	boolean	True if ND filter adjustments are selectable via a slider

/video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

The value JSON returned via the eventResponse when the `/video/shutter` property changes on the device:

Name	Type	Description
.continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
.shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
.shutterAngle	number	Shutter angle

/video/shutter/measurement

Get the current shutter measurement mode

The value JSON returned via the eventResponse when the `/video/shutter/measurement` property changes on the device:

Name	Type	Description
.measurement	string	Current shutter measurement mode Possible values are: ShutterAngle, ShutterSpeed.

/video/supportedShutters

Get supported shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/supportedShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of supported shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array of flicker-free shutter speeds
.shutterSpeeds[i]	integer	

/video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/flickerFreeShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of flicker-free shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array
.shutterSpeeds[i]	integer	

/video/autoExposure

Get current auto exposure mode

The value JSON returned via the eventResponse when the `/video/autoExposure` property changes on the device:

Name	Type	Description
.mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
.type	string	Comma-separated list of device types in the auto exposure stack

/video/detailSharpening

Get the current state of detail sharpening

The value JSON returned via the eventResponse when the `/video/detailSharpening` property changes on the device:

Name	Type	Description
.enabled	boolean	Whether detail sharpening is enabled

/video/detailSharpeningLevel

Get the current detail sharpening level

The value JSON returned via the eventResponse when the `/video/detailSharpeningLevel` property changes on the device:

Name	Type	Description
.level	string	Current detail sharpening level of supported shutter speeds Possible values are: Low, Medium, High.

Ayuda

Visite nuestra página de soporte técnico a fin de obtener ayuda rápidamente y acceder al material de apoyo más reciente para los productos descritos en este manual.

Página de soporte técnico

Las versiones más recientes de este manual, los distintos programas mencionados y el material de apoyo se encuentran disponibles en nuestra página de soporte técnico.

Cómo ponerse en contacto con Blackmagic Design

Si no encuentra la ayuda que necesita en el material de apoyo, solicite asistencia mediante el botón **Enviar correo electrónico**, situado en la parte inferior de nuestra página de soporte técnico. De manera alternativa, haga clic en el botón **Soporte técnico local** para acceder al número telefónico del centro de atención más cercano.

Cómo comprobar la versión del programa utilitario instalado

Para comprobar la versión del programa utilitario instalada en el equipo informático, seleccione la opción **About Blackmagic Camera Setup**.

- En macOS, ejecute el programa desde la carpeta de aplicaciones. Seleccione el menú **About Blackmagic Camera Setup** en la barra superior de la ventana para ver el número de la versión.
- En Windows, ejecute el programa Blackmagic Camera Setup haciendo clic en el ícono situado en el menú **Inicio**. En el menú **Help**, seleccione la opción **About Blackmagic Camera Setup** para ver el número de la versión.

Cómo obtener las actualizaciones más recientes

Después de verificar la versión del programa instalado, visite el centro de soporte técnico de Blackmagic Design para comprobar si hay actualizaciones disponibles. Aunque generalmente es recomendable instalar las versiones más recientes, evite realizar modificaciones al sistema operativo de la unidad si se encuentra en medio de un proyecto importante.

Normativas

Tratamiento de residuos de equipos eléctricos y electrónicos en la Unión Europea.

Este símbolo indica que el dispositivo no debe desecharse junto con otros residuos domésticos. A tales efectos, es preciso llevarlo a un centro de recolección para su posterior reciclaje. Esto ayuda a preservar los recursos naturales y garantiza que dicho procedimiento se realice protegiendo la salud y el medioambiente. Para obtener más información al respecto, comuníquese con el distribuidor o el centro de reciclaje más cercano.

Según las pruebas realizadas, este equipo cumple con los límites indicados para dispositivos digitales Clase A, en conformidad con la sección 15 de las normas establecidas por la Comisión Federal de Comunicaciones. Esto permite proporcionar una protección razonable contra interferencias nocivas al operar el dispositivo en un entorno comercial. Este equipo usa, genera y puede irradiar energía de radiofrecuencia, y si no se instala o utiliza de acuerdo con el manual de instrucciones, podría ocasionar interferencias nocivas para las comunicaciones radiales. El funcionamiento de este equipo en una zona residencial podría ocasionar interferencias nocivas, en cuyo caso el usuario deberá solucionar dicho inconveniente por cuenta propia.

El funcionamiento de este equipo está sujeto a las siguientes condiciones:

1. El dispositivo no debe ocasionar interferencias nocivas.
2. El dispositivo debe admitir cualquier interferencia recibida, incluidas aquellas que pudieran provocar un funcionamiento incorrecto del mismo.

R-R-BMD-20250213001

ICES-3 (A)
NMB-3 (A)

Declaración ISED (Canadá)

Este dispositivo cumple con las normas del gobierno de Canadá relativas a equipos digitales Clase A.

Cualquier modificación o uso indebido del mismo podría acarrear un incumplimiento de dichas normas.

Las conexiones a interfaces HDMI deberán realizarse mediante cables blindados.

Este equipo cumple con las normas descritas anteriormente al emplearse en entornos comerciales. Nótese que podría ocasionar interferencia radial al utilizarlo en ambientes domésticos.

Seguridad

Es necesario conectar la fuente de alimentación suministrada con la cámara a una toma de corriente que disponga de una conexión a tierra.

La reparación de los componentes internos de la cámara no debe ser llevada a cabo por el usuario. Comuníquese con nuestro centro de atención más cercano para obtener información al respecto.

Al conectar la entrada de alimentación a una batería externa para corriente continua, el cable de suministro debe incluir un limitador de corriente.

Los cables empleados deberán estar marcados como VW-1 o cumplir con las partes relevantes de los estándares IEC 60332 o IEC 60695.

A los efectos de alimentar la cámara, se recomienda que la corriente continua suministrada sea de 24 V. Si se utiliza una fuente de 12 V, el cable deberá ser capaz de transmitir una corriente de 14 A.

Este producto incluye un soporte para baterías con montura en B. Es aconsejable usar baterías de 24 V (28.8 V) o de doble voltaje.

Al emplear baterías de 12 V (14.4 V), compruebe la unidad o consulte su manual a fin de determinar la corriente máxima de descarga continua. Es recomendable que estas puedan admitir una intensidad mínima de 12 A.

Declaración del Estado de California

Las partes plásticas de este producto pueden contener trazas de compuestos químicos, tales como polibromobifenilos (PBB), que el Estado de California reconoce como causantes de cáncer, anomalías congénitas o daños reproductivos.

Consulte el sitio www.P65Warnings.ca.gov para obtener más información al respecto.

Oficina europea

Blackmagic Design Europe B.V.
Rijnlanderweg 766 D,
Hoofddorp, 2132NM,
Países Bajos

Blackmagic URSA Cine

Aconsejamos resguardar el equipo a la sombra en exteriores para prevenir la exposición del equipo o la batería de litio a la luz solar durante períodos prolongados. Mantenga las baterías de litio alejadas de cualquier fuente de calor.

La capacidad del conector **RS** de 24 V es adecuada para alimentar accesorios de otros fabricantes. Compruebe que el consumo energético total sea menor de 48 W.

Garantía

Garantía limitada

Blackmagic Design garantiza que el producto adquirido no presentará defectos en los materiales o en su fabricación por un período de 12 meses a partir de la fecha de compra. Si un producto resulta defectuoso durante el período de validez de la garantía, Blackmagic Design podrá optar por reemplazarlo o repararlo sin cargo alguno por concepto de piezas y/o mano de obra.

Para acceder al servicio proporcionado de acuerdo con los términos de esta garantía, el Cliente deberá dar aviso del defecto a Blackmagic Design antes del vencimiento del período de garantía y encargarse de los arreglos necesarios para su prestación. El Cliente será responsable del empaque y el envío del producto defectuoso al centro de servicio técnico designado por Blackmagic Design y deberá abonar las tarifas postales por adelantado. El Cliente será responsable de todos los gastos de envío, seguros, aranceles, impuestos y cualquier otro importe que surja con relación a la devolución de productos por cualquier motivo.

Esta garantía carecerá de validez ante defectos o daños causados por un uso indebido del producto o por falta de cuidado y mantenimiento. Según los términos estipulados en esta garantía, Blackmagic Design no tendrá obligación alguna de (a) reparar daños provocados por intentos de personal ajeno a Blackmagic Design de instalar, reparar o realizar un mantenimiento del producto; (b) reparar daños resultantes del uso de equipos incompatibles o conexiones a los mismos; (c) reparar cualquier daño o mal funcionamiento provocado por el uso de piezas o repuestos no suministrados por Blackmagic Design; o (d) brindar servicio técnico a un producto que haya sido modificado o integrado con otros productos cuando dicha modificación o integración tenga como resultado un aumento de la dificultad o el tiempo necesario para reparar el producto.

La pantalla del dispositivo URSA Viewfinder podría sufrir daños en caso de exposición a la luz solar directa, dado que el sistema óptico actúa como una lupa. La pantalla OLED podría quemarse o presentar problemas de retención de imagen al mostrar un fotograma determinado o contenidos de alto contraste, tales como las guías de encuadre, durante períodos prolongados. Para prevenir este inconveniente, evite cubrir el sensor infrarrojo deliberadamente y desconecte el visor si no va utilizarlo durante un largo tiempo. La garantía del producto no cubre daños por problemas de retención de imagen.

ESTA GARANTÍA OFRECIDA POR BLACKMAGIC DESIGN SUSTITUYE CUALQUIER OTRA GARANTÍA, EXPRESA O IMPLÍCITA. POR MEDIO DE LA PRESENTE, BLACKMAGIC DESIGN Y SUS DISTRIBUIDORES RECHAZAN CUALQUIER GARANTÍA IMPLÍCITA DE COMERCIALIZACIÓN O IDONEIDAD PARA UN PROPÓSITO PARTICULAR. LA RESPONSABILIDAD DE BLACKMAGIC DESIGN EN CUANTO A LA REPARACIÓN O SUSTITUCIÓN DE PRODUCTOS DEFECTUOSOS DURANTE EL PERÍODO DE LA GARANTÍA CONSTITUYE UNA COMPENSACIÓN COMPLETA Y EXCLUSIVA PROPORCIONADA AL CLIENTE. BLACKMAGIC DESIGN NO ASUME RESPONSABILIDAD ALGUNA POR CUALQUIER DAÑO INDIRECTO, ESPECIAL, FORTUITO O EMERGENTE, AL MARGEN DE QUE BLACKMAGIC DESIGN O SUS DISTRIBUIDORES HAYAN SIDO ADVERTIDOS CON ANTERIORIDAD SOBRE LA POSIBILIDAD DE TALES DAÑOS. BLACKMAGIC DESIGN NO SE HACE RESPONSABLE POR EL USO ILEGAL DE EQUIPOS POR PARTE DEL CLIENTE. BLACKMAGIC DESIGN NO SE HACE RESPONSABLE POR DAÑOS CAUSADOS POR EL USO DE ESTE PRODUCTO. EL USUARIO UTILIZA EL PRODUCTO BAJO SU PROPIA RESPONSABILIDAD.

© Copyright 2025 Blackmagic Design. Todos los derechos reservados. Blackmagic Design, URSA, DeckLink, HDLink, Videohub Workgroup, Multibridge Pro, Multibridge Extreme, Intensity y *Leading the creative video revolution* son marcas registradas en Estados Unidos y otros países. Todos los demás nombres de compañías y productos pueden ser marcas comerciales de las respectivas empresas a las que estén asociados.

El término Bluetooth® y el logotipo correspondiente son propiedad de Bluetooth SIG, Inc. Blackmagic Design cuenta con autorización para hacer uso de los mismos. Otros nombres comerciales y marcas registradas pertenecen a sus respectivos propietarios.

2025年10月
摄影机指南

Blackmagicdesign

Blackmagic
URSA Cine Immersive



目录

Blackmagic URSA Cine Immersive 765
什么是沉浸式? 765
Apple沉浸视频 765
URSA Cine Immersive 766
开始使用 767
为摄影机连接电源 767
录制设置 768
监看画面 770
监看和叠加选项 770
为第一个镜头构图 771
摄影机高度 771
摄影机倾斜 772
调整位置 773
观众自主权 774
摄影机移动 774
数据速率表 775
音频和时间码接口 776
Apple空间音频格式 776
Ambisonic 776
领夹式和微型麦克风 777
手持式麦克风 777
连接时间码 777
拍摄 779
曝光要求 779
快门速度 779
ISO和动态范围 780
ND 781
变速录制 782
直方图 783
LCD监看选项 783
录制 784
播放 785
传输片段 786
总结 787
DaVinci Resolve Studio 788
工作站配置建议 788
项目设置 789
主设置 789
视频监看 790
图像缩放调整 790
色彩管理 791
关于DaVinci Wide Gamut 792
Apple Immersive处理 793
摄取 793
剪辑 795
组合 795
背景 795
使用检视器 797
转场 799
Apple Immersive Motion Data 800
标题 801
创意考量 802
视觉特效合成 803
媒体输出节点 803
Renderer3D (渲染器3D) 804
使用Immersive Patcher合成 805
调色 806
降噪 806
一级调色 807
二级调色 807
调整大小 808
3D面板 808
CST色彩管理 810
CST时间线映射 810
CST调色监视器映射 811
CST Apple Vision Pro监看 813
音频 814
Apple空间音频格式 (ASAF) 814
交付 815
绕过转场 815
流式传输到Apple Vision Pro 816
关于Apple Immersive Video实用程序 816
Developer Information 817
Camera Control REST API 817

Livestream Control API 819
Clips Control API 823
Media Pool Control API 824
Monitoring Control API 827
Event Control API 836
System Control API 836
Transport Control API 842
Timeline Control API 846
Media Control API 849
Slate Control API 852
Preset Control API 858
Audio Control API 860
Lens Control API 865
Video Control API 869
Camera Control API 877
Immersive Control API 880
Color Correction Control API 881
Notification websocket – 1.0.0 886
messages 886
帮助 910
监管声明 911
安全信息 912
保修 913

Blackmagic URSA Cine Immersive

祝贺您购买或租赁Blackmagic URSA Cine Immersive。

Blackmagic URSA Cine Immersive专为拍摄180度立体VR影像量身打造，它们通常被称为“沉浸式视频”。这款摄影机经过优化，专为拍摄用于Apple Vision Pro的Apple沉浸视频所设计，能带来高品质沉浸式体验。

本指南旨在介绍Blackmagic URSA Cine Immersive以及如何使用它进行拍摄，重点讲解了这款摄影机的基本操作和专为交付Apple沉浸视频的独特格式。了解这种格式有助于您更好地了解这款摄影机。

在接下来的内容中，本指南会提供一系列技术和创意方面的建议，这些建议不仅可以帮助您更好地使用这款摄影机，还能让您了解沉浸式格式的优势。

什么是沉浸式?

沉浸式是一种三维空间视频格式。您可能了解，其他类似技术早已面世，并已存在多时。为了理解沉浸式技术的不同之处，下面是对其他技术的概述：

- **传统3D电影**使用两台摄像机拍摄每个场景，并且机位间隔距离模拟人的双眼距离，从而让观众体验到画面中的深度。虽然立体影像可以让观众体验深度，但观众的视角是受摄影师构图所控制的。
- **球形全景视频**结合3D和360度摄像机原理，让观众在体验深度的同时可以从任何方向观看。360度图像需要非常高的分辨率，以确保观众看到的那一小部分图像具有足够的保真度。由于人体生理机能只允许我们将头从一侧肩膀转向另一侧肩膀，如果要向后看，观众通常需要站着或坐在转椅上。
- **体积视频**不仅能让观众体验深度和环顾四周，还能移动和改变他们的视角。虽然摄像机阵列可以拍摄体积视频，但大多数虚拟现实或增强现实使用动态渲染3D素材，并且需要为参与者提供安全的环境。

沉浸式结合了传统电影和3D互动格式的优点。最简单的形式是高帧率、高分辨率的180度立体格式。

Apple沉浸视频

Blackmagic URSA Cine Immersive专为拍摄Apple沉浸视频所设计。此格式可用于Apple Vision Pro，支持高分辨率、180度视野立体3D视频。

Apple沉浸视频具有以下特点：

- 采用90fps播放。
- 每只眼的分辨率可上至4320 x 4320。
- 独特的元数据系统，可用于将原始摄像机影像映射到180度环境中。
- Apple空间音频格式，可提供3D声音。

Apple沉浸视频需要可以处理高分辨率立体180度播放的硬件。Apple Vision Pro是目前为止支持该格式的设备。



Apple Vision Pro

这款头戴式显示器搭载双3660 x 3200像素micro–OLED显示屏及7.5微米像素间距。它搭载13个追踪摄像头、一个激光雷达扫描仪、多个惯性测量单元 (IMU) 和眼动追踪感应器，可将图像与头部和眼部移动保持一致。

其M2处理器搭载8核CPU和10核GPU，可实现高达100fps的播放帧率。

URSA Cine Immersive

Blackmagic URSA Cine Immersive专为采集用于Apple Vision Pro的Apple沉浸视频所设计。



Blackmagic URSA Cine Immersive

URSA Cine Immersive采用双5800万像素RGBW传感器。这些传感器几乎是方形的，可以捕捉180度水平和垂直视野。每个传感器配有210度鱼眼镜头。镜头中心相距64mm。

内置ND滤镜，提供上至8档光源控制。镜头光圈固定在f4.5左右。

这款摄影机可以90fps录制两路8K图像流，因此其数据速率要比标准4K 24p录制高得多。

开始使用

为Blackmagic URSA Cine Immersive安装镜头并开启摄影机即可开始使用。同时，摄影机的模块插槽内已经装载一块经过格式化的Blackmagic Media Module 8TB，可直接开始录制。

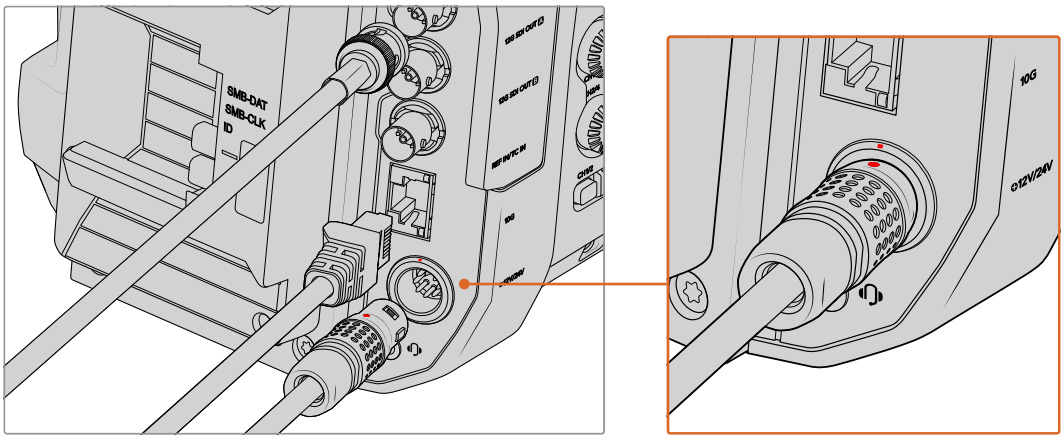
为摄影机连接电源

摄影机最快捷的电源方案是使用内附的电源适配器连接外部电源。

连接外部电源步骤如下：

1. 取下橡胶防尘塞，将其放置在某处妥善保管。使用标准IEC C13主电源线缆，将AC到24V DC适配器插头连接到您的主电源插座中。
2. 将适配器的8针接口连接到摄影机后面板的24V DC电源接口。

同时连接外部电源和电池电源时，摄影机将使用外部电源。如果您在连接了已充电电池的情况下断开外部电源，摄影机会直接切换到电池电源，不会中断运行。



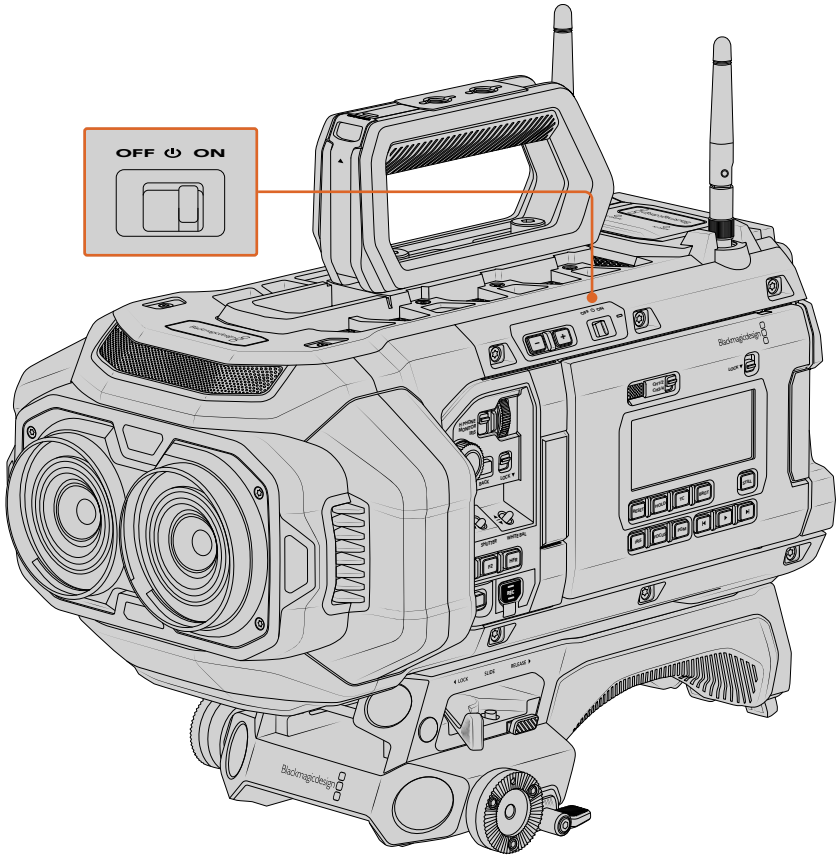
连接电源时，将插头和插孔上的红色圆点对齐，然后轻轻将插头插入摄影机，直至锁定到位。断开电源时，捏住插头外圈的金属套，然后轻轻将其从摄影机上拉离。该操作可将金属套往回拉，从而将插头从插孔松开。

摄影机标配B型电池扣板，该电池扣板专为24V B型电池设计。在24V到34V的电压范围内操作，非常适合与URSA Cine Immersive搭配使用，因为较高的电压可减少热耗，从而实现高帧率录制，并为摄影机的多个配件供电。

您可以选配兼容24V操作环境的VLock和Gold Mount扣板，也可使用适用于26V第三方电池的Gold Mount Plus扣板。

备注 使用12V到18V电源或电池来运行URSA Cine Immersive时，无法通过3针RS接口或2针Lemo接口为配件充电。

安装好电池或连接外部电源后，开启URSA Cine Immersive。将摄影机顶部的开关拨至“ON”的位置可开启摄影机。



将开关拨至“ON”的位置开启您的摄影机

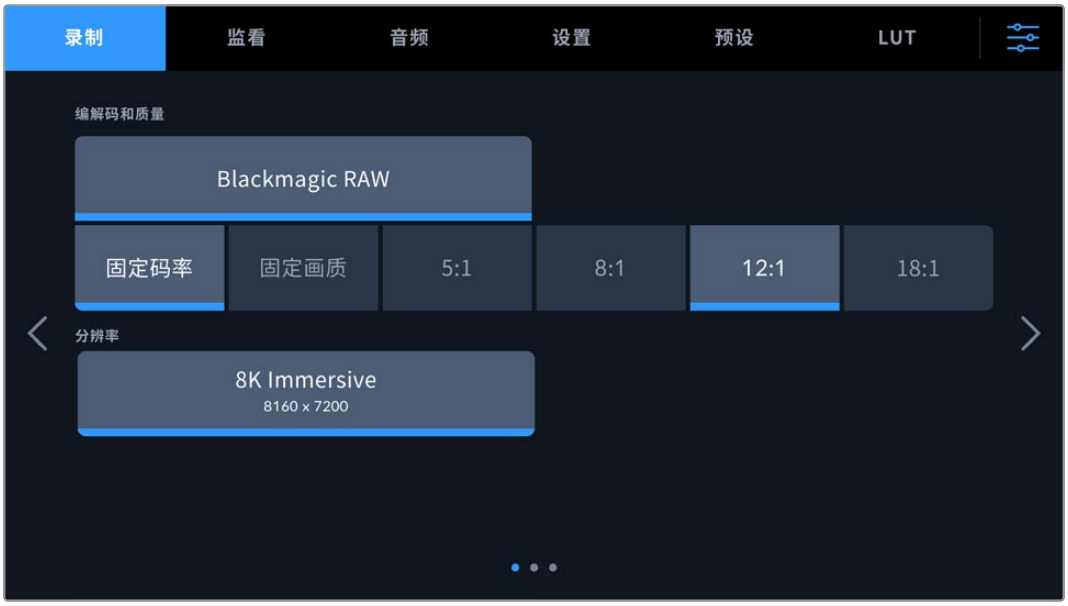
录制设置

本节将介绍拍摄沉浸式内容所独有或重要的具体摄影机设置信息。点按翻折式触摸屏或辅助区触摸屏右上角的菜单设置图标，打开摄影机设置菜单。



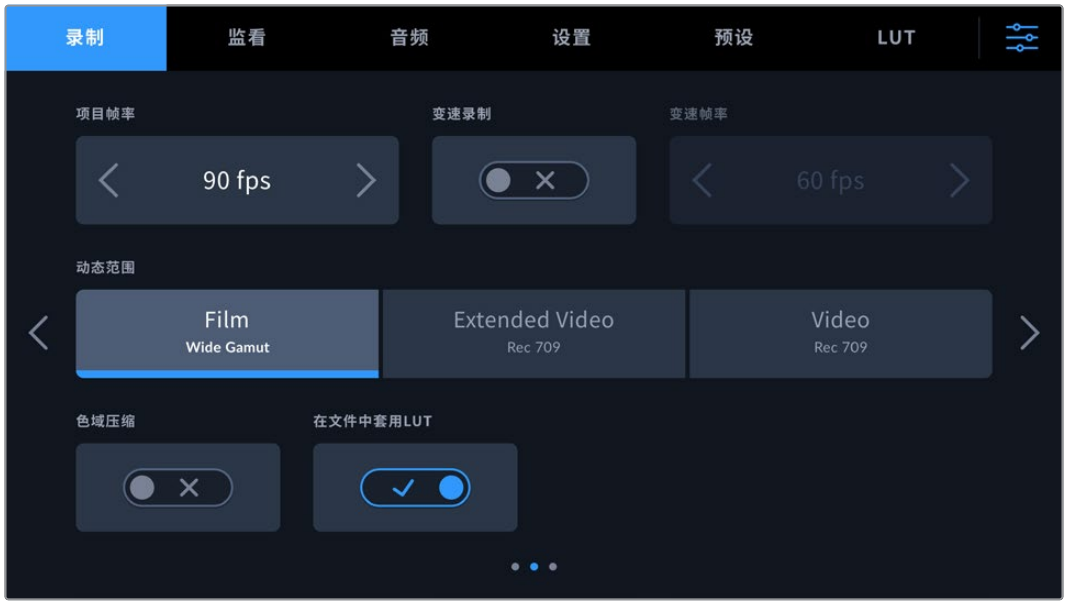
录制设置 1

您可以使用本页的“编解码和质量”菜单，为您的项目选择合适的Blackmagic RAW编解码设置。具体选项分别为四个“固定码率”和四个“固定画质”设置。如果为Apple Vision Pro录制90fps内容，强烈建议您使用12:1或Q3压缩设置。这些设置提供了在当前计算硬件下、保持可管理的数据速率和文件大小的最佳设置，同时仍然保持了极高的图像质量。



录制设置 2

摄影机“录制”设置的第二页包括了帧率设置。



摄影机项目帧率应设为90fps，以符合Apple Vision Pro的显示帧率。

如果您是为其他系统拍摄，60fps或50fps等较低的帧率或已足够。如果是为24p电影内容拍摄视觉特效画面，24fps或25fps等更低的传统电影风格采集帧率可能较为适合。

现在，URSA Cine Immersive已准备就绪，可以开始使用。摄影机内部的Media Module已经完成格式化，可以直接开始录制。

监看画面

URSA Cine Immersive提供众多监看选项，可让您在拍摄的同时监看画面。

翻折式触摸屏专供摄影师使用。它可以翻折并收起，让摄影机更紧凑小巧的同时依然能保持屏幕内容可见；也可以向前翻转从而和镜头同向，方便自拍时进行画面监看。

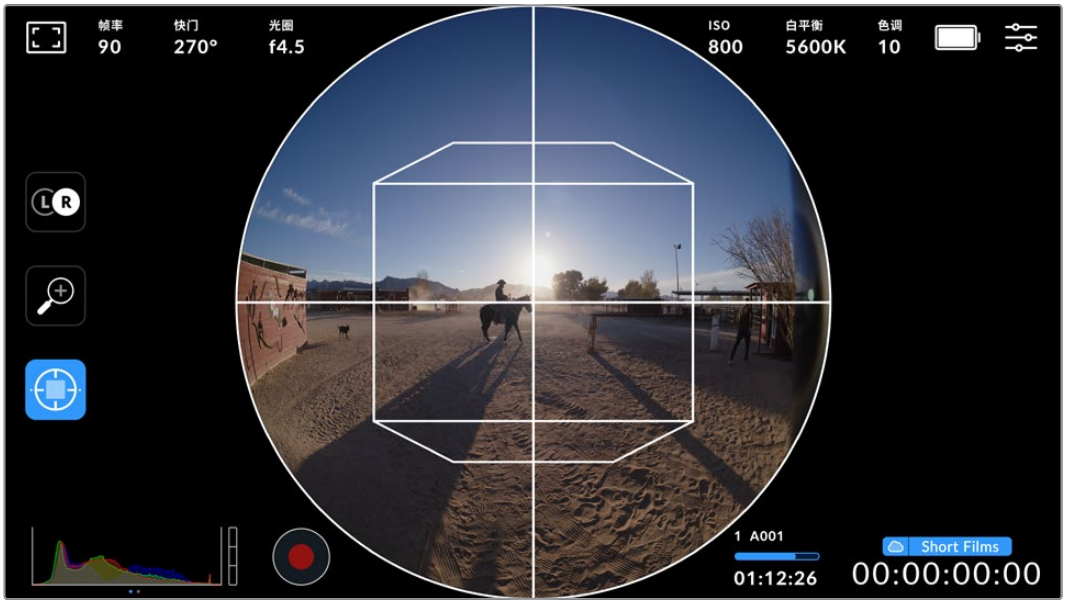
辅助区触摸屏可供另一名摄制组成员使用，以便在不阻碍摄影师视线的情况下监看画面。由于摄影机具有非常宽的视野，使用辅助区触摸屏时需要多加注意。





监看和叠加选项

摄影机以Lens Space镜头空间显示，一次只能查看单个眼的画面。翻折式触摸屏和辅助区触摸屏会加以遮挡处理，以显示180度视野。其SDI输出则显示每个传感器采集的210度全画面。



所有监看输出均显示同一只眼。您可以在任何一个触摸屏上点按“L R”左右眼切换按钮，在左右眼间切换。

按下叠加按钮可在触摸屏上开启可选的叠加选项。叠加选项如下:

- 180度外边界线。
- 中心十字准线，用于标记画面中心。
- 内部引导线，用于表示在Apple Vision Pro上直视前方时所看到的大致视野。

这些工具都是为了在沉浸式采集时辅助精确构图和定位所设计的。

为第一个镜头构图

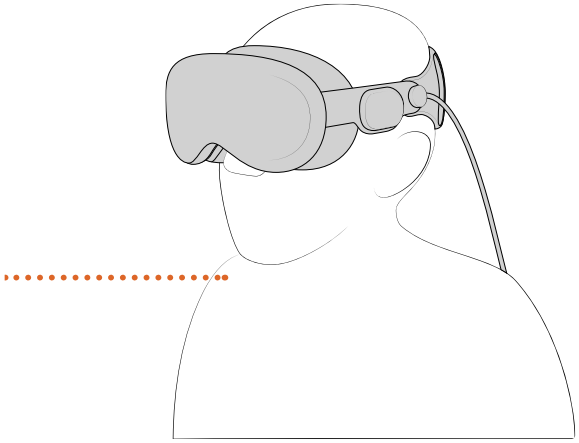
沉浸式和传统2D内容有很多共同之处，但也有非常明显的区别。

例如，您是否曾经因为忙着刷手机而感到晕车？这是因为大脑接收到了不一致的信号引起感官冲突所造成的。在车上刷手机时，内耳向大脑发出移动的信号，但眼睛却向大脑发出静止的信号。

沉浸式体验很容易产生类似的感觉。例如，如果您平移摄影机，观众的眼睛就会告诉他们此刻正在移动，而他们的脖子和内耳会告诉他们此刻是静止的。因此，在开始学习如何通过移动摄影机拍摄为观众带来悦目观感之前，还需先掌握如何拍好固定机位的画面。

摄影机高度

在传统2D电影中，您可能会通过将摄影机部署在高处或低处来表示场景中的力量平衡。而拍摄沉浸式内容时，如果将摄影机部署在地面上，会对观众产生怎样的影响？这可能会给观众一种脖子以下被埋起来的感觉。



对地面的高度感知会影响到他们的观看体验。较低的摄影机部署位置会让观众感觉地面高度不合理。



拍摄沉浸式内容时，低角度可能会让观众感觉不适。

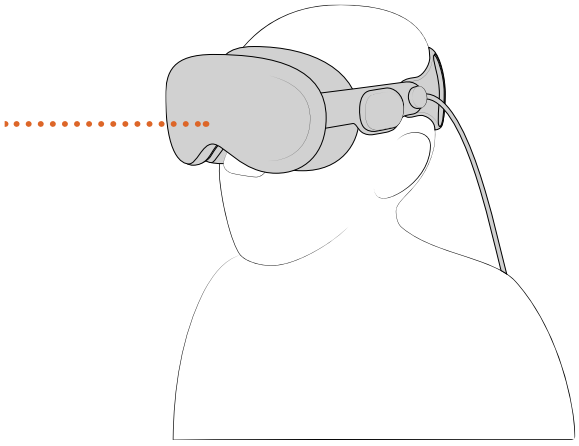
为了实现身临其境的感觉，一般来说最好是选择一个合适的摄影机高度和位置，让观众感觉自己是站在或坐在场景中。这样一来，您仍然有足够的创意表达空间。如果场景中出现人物，将摄影机稍微调高一点可以给观众一种居高临下俯视他人的力量感；反之，将摄影机稍微调低一点则会给观众一种仰视他人的感觉。

当然，这并不表示大胆新颖的机位角度就一定无法保持观影舒适度。飞行画面、俯拍视角、低角度等许多其他技术也都被成功地应用到沉浸式制作当中。只要加以恰当的剪辑、移动和背景内容，这些方式同样可以取悦观众。

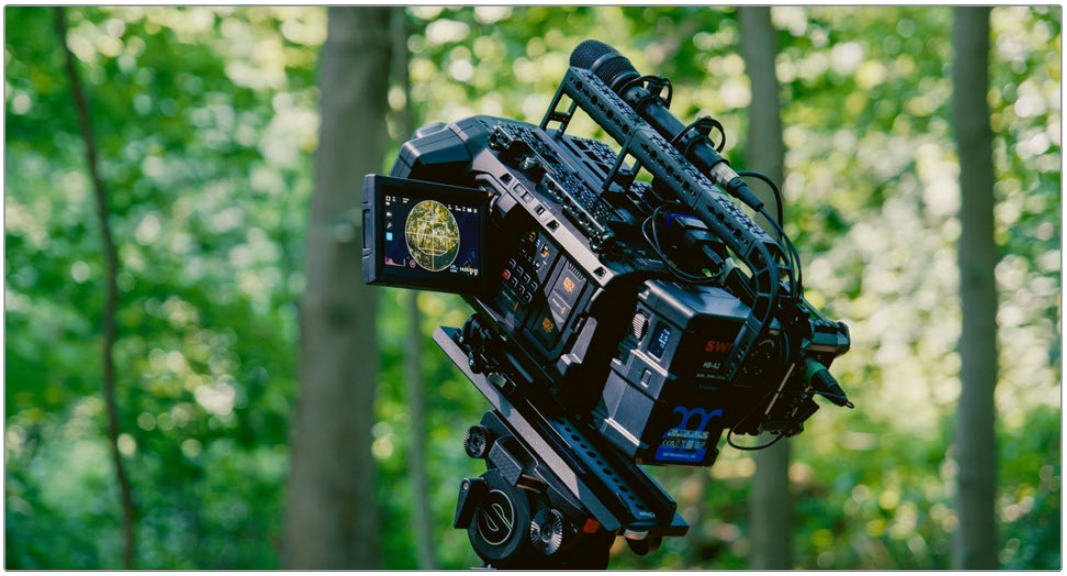
摄影机倾斜

在拍摄传统2D内容时，如果将摄影机部署在高处或低处，电影摄影师经常会运用摄影机倾斜来保持构图。不妨思考一下，这种方式会如何影响沉浸式影像。

如果摄影机的拍摄角度是向下的，那么观众就需要抬头才能看到地平线。这样做虽然不太会引起眩晕，但可能会破坏观众的临场感，因为在现实生活中，地平线应该在正前方。



对于沉浸式镜头来说，地平线处于零度才最符合常理。



如果您倾斜URSA Cine Immersive，那么观众在影像中看到的地平线位置就会和他们平日里看到的地平线位置不一致。

因此，大部分沉浸式镜头会采用零度或者接近零度的倾斜角度拍摄。由于摄影机采集的是水平和垂直180+度视野，观众仍然可以向上和向下看。但是，观众具体看向某个方向并不由电影制作人强制，而是由观众自行控制。

要鼓励观众观看您图像的各个部分，可以使用灯光、间隙构图和引导线等其他技术来实现。

调整位置

沉浸式一般是坐下观看。由于它不是增强现实或虚拟现实格式，观众不必站立或四处走动。您坐好了吗？现在做个试验，请先左右转动头部，再上下移动头部。在不移动身体且确保舒适的情况下，您的头部移动幅度有多大？沉浸式格式的视野超过了大多数人转动头部的舒适范围。这是一件好事，因为它意味着如果您向左或向右转头，影像仍然能够提供周边视觉。这个试验可以帮助您理解主体画面应位于视野中的哪个位置。

典型的头部移动范围：

旋转：120度到160度
屈伸：110度到130度

当头部移动和眼部移动结合在一起时，人的视野范围可以超过自己的肩膀，但对于大多数人来说，维持这种状态会感到不太舒服。

想要辅助构图，可以开启摄影机叠加选项。内部标记可显示典型视野范围，将画面保持其中能有助于内容的可见性。

除此之外，您还需要考虑到兴趣点在画面中的移动情况。如果要目视跟踪一个人从最左侧走到最右侧，观察者的头部移动幅度会达到极限。类似地，从兴趣点在图像一侧边缘的镜头切换到兴趣点在图像另一侧边缘的镜头，观察者的头部移动幅度也会达到极限。周全的构图设计有助于提升观众的舒适度，同时还能保持创意的沟通。

观众自主权

当拍摄传统2D内容时，您可以使用构图、镜头选择和对焦设置来引导观众注意力。当拍摄沉浸式内容时，180度的视野使得排除信息和引导观众注意力变得更加困难。沉浸式带给观众更大的自主权去探索影像。这是沉浸式内容如此吸引人的因素之一。慎重使用有指导性的声音、灯光、阻挡和构图，仍然有可能引导观众的注意力。

摄影机移动

沉浸式摄影机不需要保持静止，但一定要慎重操作，因为有些摄影机移动会造成不自然或不适感。拿汽车来举例，这样会有助于理解如何让沉浸式摄影机运动感到舒适。大型车辆具有质量和惯性，可以抵抗速度或方向的突然变化。因此，它会随着时间推移加速和减速，并以可预测和可控的方式移动。掌握了静态镜头和移动的沉浸式镜头后，考虑这些因素将有助于创造令人愉悦的运动：

- **时长：**让观众有时间去适应摄影机运动并享受它。短暂的移动更容易引起方向混乱。
- **强度**：低强度动作更容易整合到沉浸式剪辑里。高强度移动不是不可行，但需要热身和冷却处理让观众感到舒适。
- **视觉锚点：**场景里的有些部分是不动的。没有视觉锚点的镜头会让人迷失方向。汽车内部等前景元素或地平线等远处物体，将有助于移动拍摄。

数据速率表

下表包含了90fps和24fps 8K x 7K沉浸式内容的数据速率。

Blackmagic RAW固定码率选项在90fps下的最高数据速率

8K x 7K沉浸式			
编解码	**像素尺寸**	**帧率**	**数据速率**
Blackmagic RAW 5:1	8160 x 7200立体	90	3203MB/s
Blackmagic RAW 8:1	8160 x 7200立体	90	2002MB/s
Blackmagic RAW 12:1	8160 x 7200立体	90	1335MB/s
Blackmagic RAW 18:1	8160 x 7200立体	90	889MB/s

Blackmagic RAW固定码率选项在24fps下的最高数据速率

8K x 7K沉浸式			
编解码	**像素尺寸**	**帧率**	**数据速率**
Blackmagic RAW 5:1	8160 x 7200立体	24	854MB/s
Blackmagic RAW 8:1	8160 x 7200立体	24	534MB/s
Blackmagic RAW 12:1	8160 x 7200立体	24	356MB/s
Blackmagic RAW 18:1	8160 x 7200立体	24	237MB/s

音频和时间码接口

音频是沉浸式体验的关键部分。因此，为URSA Cine Immersive配备专门的音频采集设备十分重要。

Apple空间音频格式

Apple空间音频格式，即ASAF，可用于在观看Apple沉浸视频时提供3D音频。ASAF是更宽泛的Apple Positional Audio Framework音频框架的一部分，可以存储三种不同类型的录音。

- **基于声道：** 您可以将传统环绕声混合嵌入ASAF文件。这对于头部锁定的旁白和音乐等场景来说非常理想。
- **基于对象：** 声音可以被当成是单个对象，它们的元数据可描述其在3D空间中的位置，独立于单个扬声器通道。这些通常是单声道录制。它们的3D坐标可以动画处理，从而跟随场景中移动的对象。这非常适合剧情声，即场景中可见的对象。
- **Ambisonic**：全方位球形环绕声格式，可使用球面谐波渲染所有方向的音频。这非常适合环境声。

本节的其余部分介绍了构建有效ASAF文件所需的设备。

Ambisonic

Ambisonic麦克风搭载四个或更多胶囊，这些胶囊按照精确的几何结构排列。每个胶囊均能根据声源的方向采集具有不同到达时间和强度的同一个声源。这些胶囊录制到各自的通道后，稍后由计算机进行处理。软件会通过对比差异来估计录制声音的方向和性质。



一阶Ambisonic麦克风

Ambisonic麦克风

Ambisonic麦克风根据其所含胶囊的数量进行分类。胶囊越多意味着麦克风可以越准确地再现空间细节。

一阶: 4声道

二阶: 9声道

三阶: 16声道

四阶: 25声道及更多

不论麦克风是在摄影机上还是在旁边，非常关键的是要把Ambisonic麦克风正确地对齐摄影机拍摄方向。此外，因为Ambisonic麦克风可用于不同配置，比如端射、垂射、向上或向下等，请确保您的音频技术员知道要使用何种配置。这一信息可通过录音装置上的元数据或分享备注来传达。

Ambisonic录音可作为环境声床使用。其他基于声道和基于对象的音效可以叠加其上。如果正确地与摄影机对齐，Ambisonic录音还可以作为参考供音频工程师将来添加音响设计所用。

领夹式和微型麦克风

当您使用180度摄影机拍摄时，是不可能使用吊杆麦克风的，因为它会出现在场景中！这意味着，您需要使用隐藏麦克风和领夹式麦克风来录制剧情声和对白。

例如，您可以:

- 通过无线传输器和接收器，将音频源直接连至URSA Cine Immersive的两个XLR输入。
- 将其连接至外部录音设备。可以是大型多声道设备或微型佩带式装置。

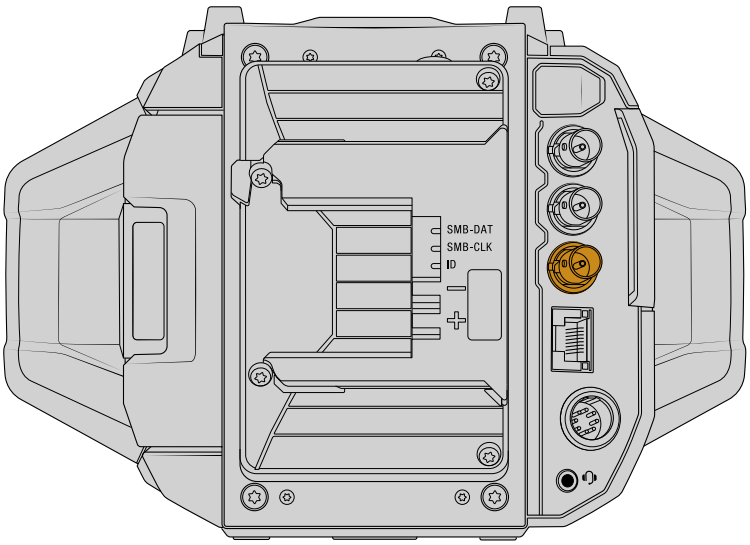
然后这些录音可同步或混音为基于声道或基于对象的声音。

手持式麦克风

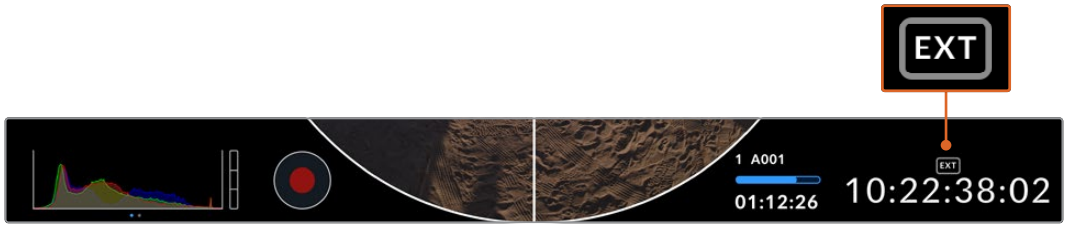
摄影机停止拍摄后，在现场录制额外音效不仅可以节省后期制作时间，还可以制作出更真实的声音。在制作过程中携带手持超心型指向或类似麦克风是非常有益的，这样可以在麦克风尽可能靠近的情况下重新录制。然后这些录音可剪辑成终混作为基于声道或基于对象的声音。

连接时间码

要将外部时间码连接至URSA Cine Immersive，可使用BNC线缆将时间码发生器连接至摄影机后面板上的时间码输入上。从事双系统拍摄时连接同步时间码可确保音频和图像在后期制作过程中轻松同步。



当这个接口连接了帧率相匹配的外部时间码时，摄影机将自动锁定输入时间码。锁定到外部时间码时，URSA Cine Immersive摄影机LCD屏幕上的时间码计数器一侧将出现“EXT”徽标。如果您拔出线缆，时间码将保持锁定状态，“EXT”徽标会变成“INT”，提示您现在正在运行摄影机的内部时间码时钟。



当连接至时间码发生器时，摄影机时间码会显示“EXT”

备注 SMPTE时间码标准适用于23.976、24、30等典型的帧率。90fps没有SMPTE时间码标准。因此，当URSA Cine Immersive的项目帧率设为90fps时，摄影机实际上使用的是30fps时间码。其SDI输出运行的也是30p。任何时间码配件或外部录音设备应该配置为30fps。

拍摄

现在基本了解了如何操作Blackmagic URSA Cine Immersive摄影机以及充分利用沉浸式格式后，本章节将进一步探讨摄影机的操作。

曝光要求

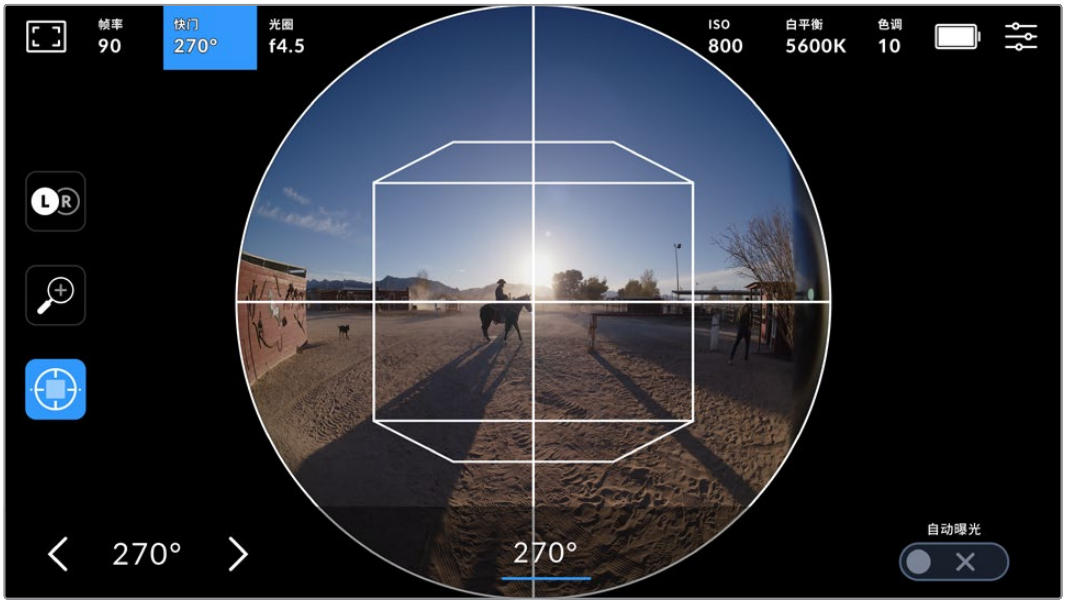
Apple沉浸视频比传统24fps拍摄具有更高的曝光要求，可能需要您重新考虑拍摄和打光方式。如果保持180度快门，90fps的拍摄就会需要近乎四倍的照明来实现同样的曝光效果。此外，镜头的固定光圈大约是f4.5，这比典型的超高速镜头要暗上3档。因此，您可能需要多出上至7档的照明来实现足够的曝光。但如果您以90fps拍摄，使用270°到360°之间的宽快门角度，可以有助于补偿高帧率的光损失，同时仍然达到令人愉悦的运动模糊量。

快门速度

“快门”提示可显示您的快门速度或快门角度。“设置”菜单里的快门类型设置可通过显示快门信息来选择“快门角度”或“快门速度”。

要更改快门速度：

1. 在顶部工具栏点按“快门”图标。
2. 使用左右箭头可浏览常用的快门速度预设。如果您希望选择一个特定的快门值，可以连按两次屏幕左下方的当前快门提示，然后，在打开的小型键盘中输入快门数值，最多可在11.2和360角度之间精确到小数点后两位。



通过屏幕键盘可键入自定义帧率。

快门速度对高帧率沉浸式内容外观的影响较小。虽然以24p或30p拍摄时，180度快门仍然受到许多人的青睐，但在拍摄沉浸式内容时，更常见的是使用快门速度来辅助曝光。

当在低光环境下拍摄时，快门可增至270度，或在某些特殊情况下，可以是360度。

当在明亮环境下拍摄时，快门速度可低至11.2度。但在明亮条件下，更常见的是将快门速度保持在您所喜欢的运动设置上，使用ND滤镜来控制曝光。

需要注意的是，Apple沉浸视频的90fps高帧率可能会在拍摄闪烁的灯光的场景时出现问题。在60hz环境下工作时建议使用270度快门，在50hz环境下工作时建议324度快门。

ISO和动态范围

“ISO”提示可显示当前的感光度设置。点按此提示可调整ISO值来用于不同的照明条件。URSA Cine Immersive的ISO范围从ISO 200到3200。最佳ISO为800。



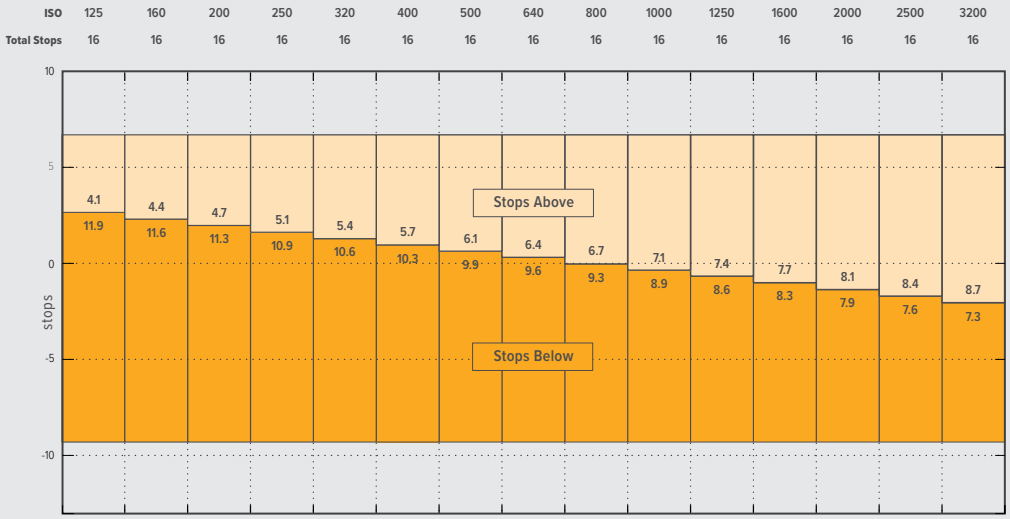
点按ISO提示可配置ISO。

点按数字可获得不同数值的预设，以一档为增量选择ISO设置。预设下方提供了一个滑块，能以1/3档为增量微调ISO设置。

这一ISO设置可能会偏高或偏低，因此请根据拍摄现场的不同情况妥善设置。例如，在低光照条件下拍摄时可使用1600的感光度设置，但可能导致一些可见噪点。而在明亮环境下拍摄时，则可使用200的ISO设置，以便获得更浓郁的画面色彩。

Blackmagic摄影机可捕捉宽动态范围，不论是影像拍摄还是使用DaVinci Resolve进行调色，都能为您提供非常高的灵活性。这里的图表显示了URSA Cine Immersive中可用的动态范围。

URSA Cine Immersive动态范围



ND

摄影机ND滤镜通常是控制曝光的最常用方法，可让您通过“档”来预设不同曝光程度，从而减少到达摄影机传感器上的光线。



URSA Cine Immersive的ND可通过翻折式触摸屏上方左侧的按钮来调整。可用选项为“2”、“4”、“6”和“8”档，以及“透明”滤镜。

触摸屏左上方的提示会显示当前ND设置。使用ND时，该提示会一直保留。当ND设为透明滤镜时，该提示将在4秒后消失。



备注 您可以根据自己的习惯，调整ND滤镜提示所使用的术语。使用设置菜单可在ND编号、档数、以及分数之间选择。

变速录制

设置菜单中的项目帧率是URSA Cine Immersive的录制格式帧率并定义了素材播放的速度。这通常会设定为与您后期制作工作流程和交付所使用的播放速度和音频同步所匹配的帧率。



点按URSA Cine Immersive的帧率提示可调出帧率设置。URSA Cine Immersive共有9种项目帧率设置，分别为每秒23.98、24、25、29.97、30、50、59.94、60以及90帧。拍摄沉浸式内容时，项目帧率应设为90fps。

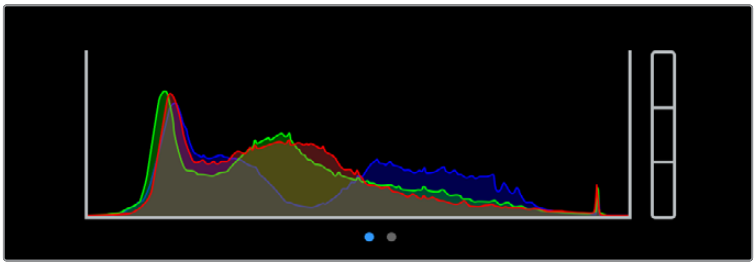
传感器帧率可让您定义传感器采集图像的速度。默认情况下，URSA Cine Immersive会采用相同的项目帧率和传感器帧率进行拍摄，以匹配正常播放速度。但是，您可以点按摄影机“帧率”菜单右下方的“变速帧率”切换图标来单独设置传感器帧率。

当启用变速帧率时，您可以手动设置传感器帧率。需要注意的是，90fps是这款摄影机的最高帧率，该设置匹配为Apple Vision Pro拍摄的90fps项目帧率。但是先选择30或60帧的较低项目帧率，然后再以90fps拍摄变速可让您构建出美丽的慢动作镜头。

当使用变速帧率后，屏幕左上方的帧率提示将显示传感器帧率，然后是项目帧率。

直方图

直方图位于URSA Cine Immersive触摸屏的左下方，它可以显示图像的亮度分布情况。左侧表示阴影，中间表示中间调，右侧表示高光。图表中所分布的高低起伏表示图像中相应亮度级别所对应的像素数量多少。



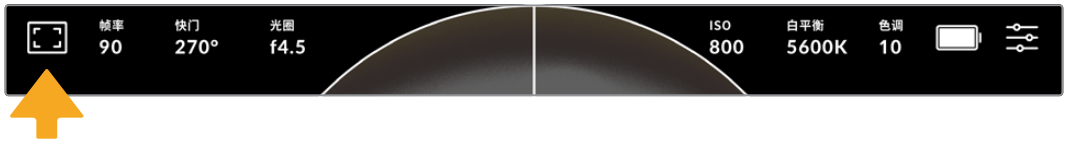
直方图可以直观呈现片段的色调分布。如果图表分布靠左，表示图像可能欠曝。类似地，如果图表分布靠右，则表示图像可能过曝。直方图应该结合图像的具体情况来解读。但一般来说，一个曝光良好的图像，其直方图会平衡地堆积在中部。提高曝光度时，直方图会向右移动；降低曝光度时，直方图会向左移动。

直方图右侧的图标可在各个色彩通道发生裁切时显示提示。通常情况下，应避免任何通道出现裁切。但是在某些情况下，如果图像裁切有助于保留图像的中间调和阴影，则视为可以接受。

“向右曝光”是一种刻意为之的画面过曝处理技巧，指在不裁切色彩通道的情况下，将直方图尽可能推向右侧。这样可以最大化信号并最小化噪点。只要在后期降低曝光，这是一种生成干净图像的好方法！

LCD监看选项

点按URSA Cine Immersive LCD触摸屏左上角的监视器图标可进入LCD监看设置。该菜单包括了监视器亮度等实用选项，还包括了强大的曝光工具。



斑马纹

斑马纹设置可开启或关闭触摸屏上的斑马纹显示，还可以设置斑马纹级别。斑马纹会将画面中超过设定曝光的部分以斜线标出。百分比设置用于显示图像中过曝的部分。通常是将该数值设为95%之类而不是100%，因为这样可让您识别图像中可矫正的过曝部分。第二个斑马纹设置可用来开启或关闭中灰斑马纹区域，或者切换到中灰 + 1档。

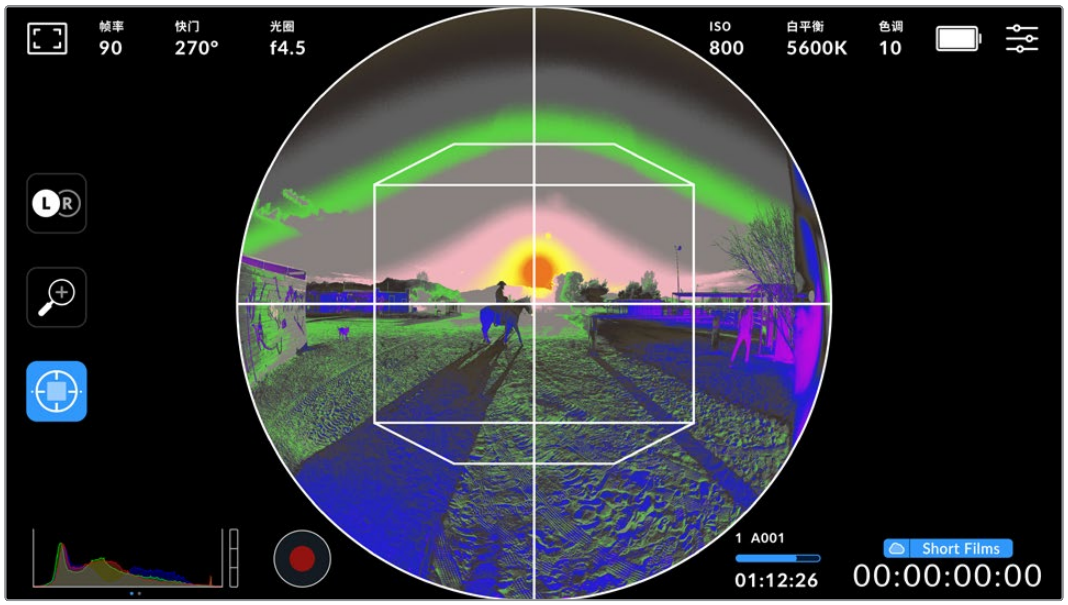
网格

网格设置可在LCD触摸屏上开启“三分法”、“水平线”、“十字准线”或“中心点”显示，还可以设置在所有URSA Cine Immersive输出上显示哪些叠加信息。拍摄沉浸式内容时，水平线指示会非常有帮助。该功能可在图像中心显示一个双十字准线，摄影机完全水平时，双准线会对齐，摄影机不水平时，双准线会偏离。

伪色

伪色功能可在您的图像上叠加不同色彩，以代表图像中不同元素的曝光值。例如，粉色代表了适合较浅肤色的理想曝光，而绿色则是较深肤色调的对应色彩。在录制人像画面时，您可以通过查看粉色或绿色伪色来保持肤色曝光的一致性。

以此类推，当您图像中的元素从黄色变到红色，就意味着现在处于过曝状态。与直方图类似，这个工具可让您快速直观地了解图像的色调分布。



录制

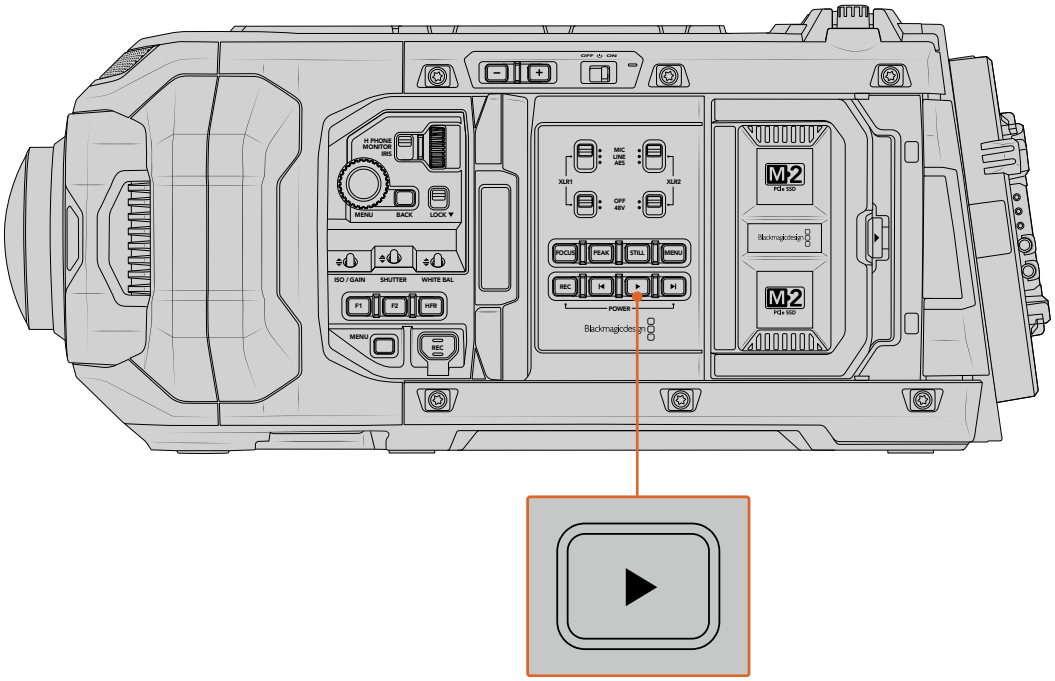
按下摄影机上红色的录制按钮即可开始录制。前方控制面板、翻折式LCD触摸屏后侧的内部控制面板、辅助区控制面板，以及两块LCD触摸屏上都设有录制按钮。再次按下录制按钮可停止录制。

存储状态提示可显示剩余录制时间。该时间根据您所选的帧率，将以时：分：秒的格式显示。录制时间上方的存储介质提示条图标会根据当前状态分别显示为蓝色、白色或红色，并且会显示存储介质上已使用的空间。



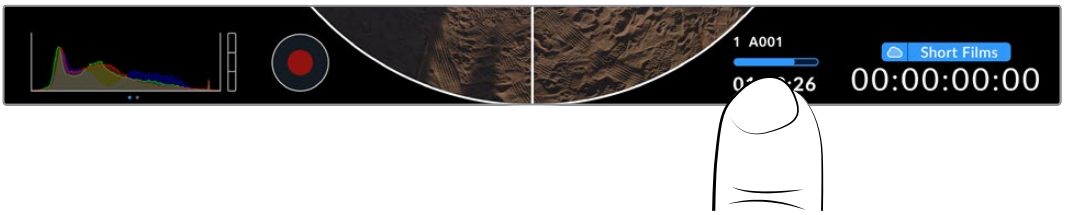
播放

片段录制完毕后，您可以使用播放控制按钮来播放片段。要播放您所录制的最后一个片段，可按下位于摄影机左侧翻折式触摸屏背面的物理播放按钮。使用前进或倒退按钮导航到其他片段。要离开播放模式，并准备让摄影机继续录制，请再次按录制按钮。

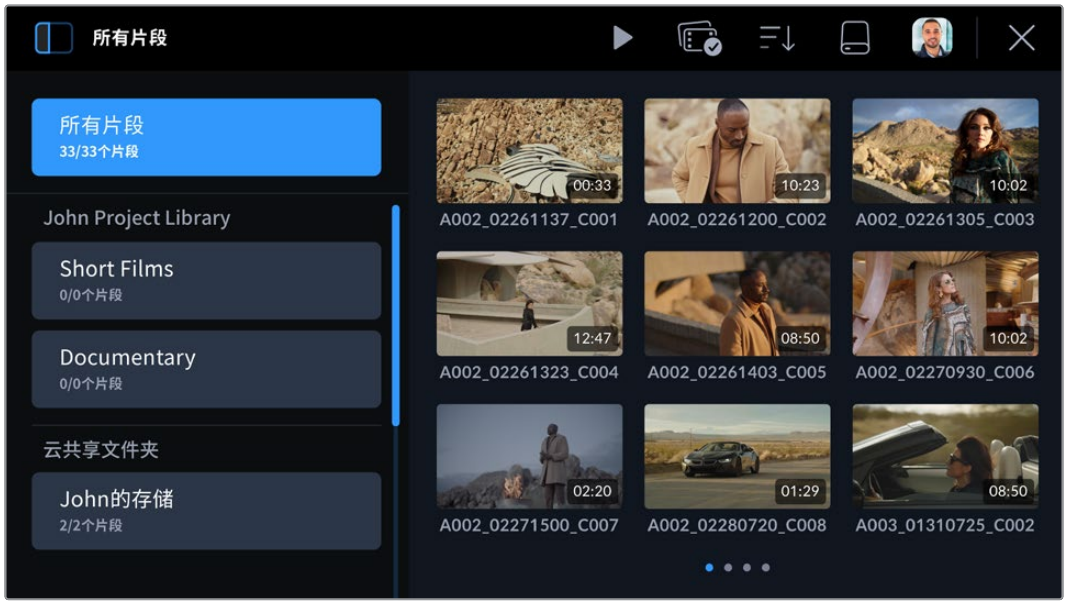


按前进或倒退按钮可跳到片段的开始或结束处。按后退按钮一次会跳转到当前素材片段的起始位置，按两次则跳转到上一个片段的起始位置。长按前进或后退按钮可以两倍的速度快进或快退。快进或快退时，按两次前进或后退按钮可获得4倍速度，按三次可获得8倍速度，按四次可获得16倍速度。

如果需要对拍摄的素材更密集地查看，您可能会更想要使用媒体池。要打开媒体池，可点按摄影机触摸显示屏底部的其中一个存储提示。



主页浏览器可显示摄影机所连接的所有存储介质上所有录制片段的缩略图。缩略图分三行显示，随着片段的逐渐增多，您可以在触摸屏上滑动以显示下一页缩略图。显示屏底部的页面提示可显示共有几页片段。



将URSA Cine Immersive连接至Blackmagic Cloud账户，将素材直接发送至DaVinci Resolve项目。

如果您要在任何时候退出媒体池，点按右上角的X图标，或再次按下摄影机的任何一个录制按钮即可。

传输片段

您可以通过多种方法从摄影机存储传输片段。例如，您可以通过10G以太网线缆将URSA Cine Immersive连接到计算机或网络，并通过以太网从摄影机存储中复制片段；甚至可以将Media Module装载到连接网络或计算机的Blackmagic Media Dock上，这样就可以同时使用最多3个媒体模块。

通过以太网连接计算机时，您的计算机会像连接Blackmagic Cloud Store一样识别到Blackmagic Media Module。

摄影机默认设置为使用DHCP，这表示您的计算机或网络可以检测到摄影机并自动为其指派一个IP地址。

启用文件共享

首次设置摄影机时，如果您将摄影机通过以太网连接计算机或网络，SMB文件共享默认为禁用。这是因为它不同于Blackmagic Cloud Store。使用Blackmagic Cloud Store时，您可能希望立即与网络上的其他用户共享文件，但在一天的拍摄结束后，您可能不希望立即允许对主摄影机文件进行相同级别的访问。出于这一原因，我们决定把是否启用SMB文件共享的选择权交给您。

1. 要启用文件共享，您需要在计算机中下载并安装Blackmagic Camera Setup。这款设置实用程序是管理软件程序，可以让您更新摄影机并更改管理员设置。Blackmagic Camera Setup下载网址：https://www.blackmagicdesign.com/cn/support/family/professional-cameras。
2. 下载并安装设置实用程序后，请将摄影机通过USB连接到计算机，并运行Blackmagic Camera Setup。通过位于摄影机顶板靠近后面板的USB端口进行连接。
3. 要打开设置，可点击URSA Cine Immersive的图像或者对应的设置图标。
4. 向下滚动到“Network Access”（网络访问）设置，并启用“File Sharing (SMB)”功能。然后在右下角点击“Save”将其生效。

5 找到下方的URL设置，点击URL图标，即可自动填写地址。URL地址会将您的摄影机名称显示为本地网络地址。

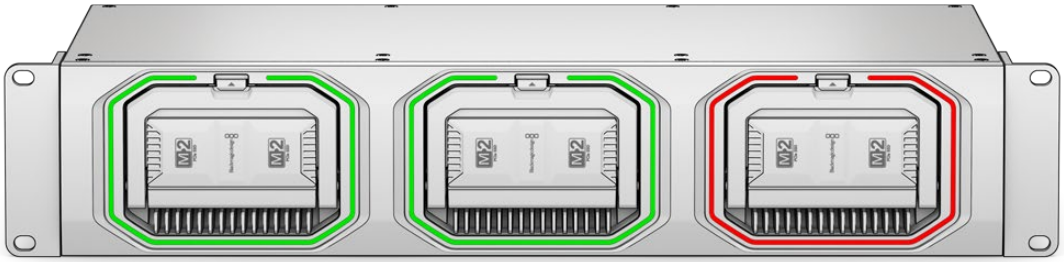
6 点击“Save”（保存）。

要直接进入片段，请点击摄影机的URL名称。界面会打开一个窗口，向您询问密码或者作为访客连接。点击“Guest”（访客）单选按钮，然后点击“Connect”（连接）。

7 您的计算机会自动导航到Media Module在Mac计算机访达中的位置，或者Windows计算机资源管理器中的位置。实用程序将会记住您的以太网设置，因此只需执行一次操作，待您下次再通过以太网连接摄影机时，就可以直接访问Media Module，无需重复上述操作了。

使用Blackmagic Media Dock

Blackmagic Media Module采用快速装载和卸载设计，方便您将片段传输到外部存储，或者快速更换空白媒体模块继续录制。Blackmagic Media Dock可在不需要摄影机的情况下访问Media Module上的素材。



总结

Apple沉浸视频引入了一种新的视觉叙事方法，为传统2D电影提供了另一种选择。它的出现并不是为了取代传统2D电影，而是作为一种额外的形式，供电影制作人根据项目需求选择。与任何形式一样，创意选择可以根据故事不同而有所不同。

虽然新格式通常涉及复杂的工作流程，但URSA Cine Immersive旨在简化拍摄Apple沉浸视频的过程。它为沉浸式内容提供了直接制作路径，也可用于视觉特效和其他应用程序。

本指南概述了URSA Cine Immersive的核心功能和工作流程。这只是一个入门介绍。与任何摄影机系统一样，您需要根据具体制作需求进行进一步探索和测试，从而获得理想结果。

DaVinci Resolve Studio



DaVinci Resolve Studio是一套全面综合的后期制作软件，它将剪辑、调色、视觉特效、动态图形以及音频后期制作无缝整合到一起。它支持针对Blackmagic RAW Immersive格式的特性所定制的工作流程，包括超高分辨率立体视频、90fps帧率和空间音频混合。DaVinci Resolve Studio包含丰富全面的功能，可用于Apple沉浸视频的制作、监看以及交付，是沉浸式内容创作者的强大工具。

工作站配置建议

硬件配置建议

- Mac Studio M3 Ultra台式机
- 高刷新率监视器（最低60Hz）

最低系统配置要求

- macOS 15或更高版本
- 64GB系统内存
- 监看：Blackmagic Design Desktop Video 12.9或更高版本
- 基于Apple Silicon的计算机或支持Metal的GPU

沉浸式监看配置要求

- Apple Vision Pro
- 高速无线网络（最低Wi-Fi 5，首选Wi-Fi 6）

项目设置

开始摄取Blackmagic URSA Cine Immersive RAW媒体文件之前，请先检查DaVinci Resolve的项目设置，以确保媒体文件进行正确的解拜耳处理，以便获得优化的视觉输出和便捷的剪辑工作。

主设置

在设置项目时，需要明确您是为Apple Vision Pro创建内容，这一点至关重要。这能确保您在项目的准备、组合、交付等阶段都能获取正确的设置和工作流程选项。

1. 运行DaVinci Resolve Studio。
2. 点击“项目管理器”窗口右下角的“新建项目”。
3. 输入项目名称，设置媒体位置（如有需要），然后点击“创建”。
4. 点击DaVinci Resolve Studio右下角的齿轮图标，打开“项目设置”。

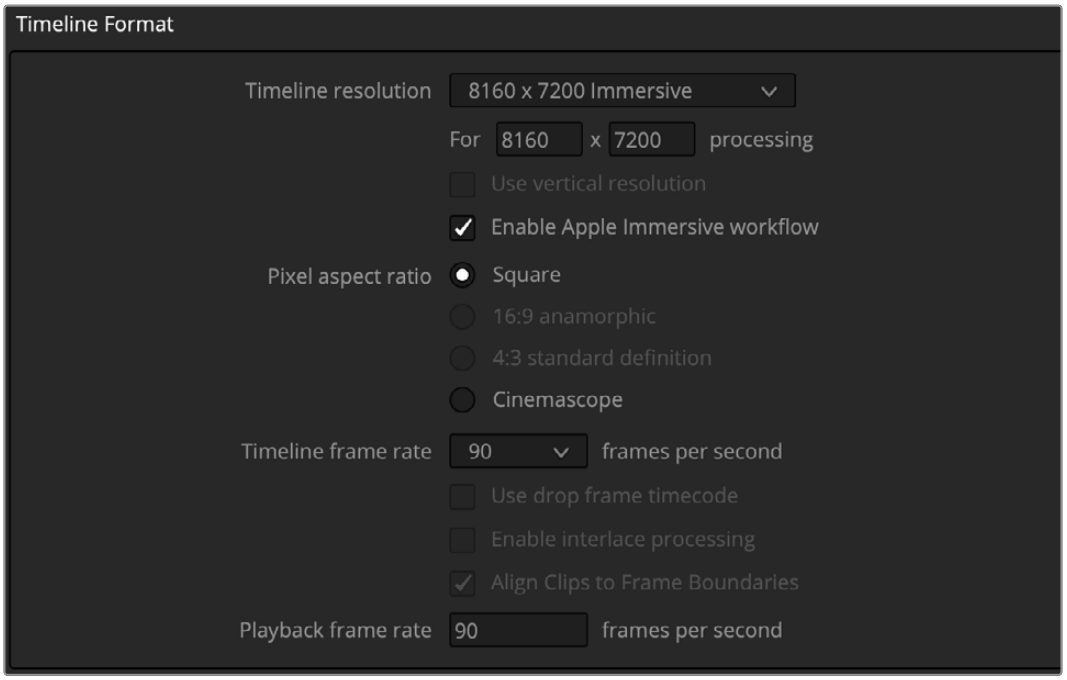
   首先，您需要设置项目的分辨率和帧率。
5. 在侧边栏中选择“主设置”。
6. 在“时间线格式”类别下，选择“Enable Apple Immersive workflow”。



   启用后，软件会对DaVinci Resolve的多项设置进行配置，从而更好地适配沉浸式工作流程。创建沉浸式项目时，请始终确保先选择这一设置。
7. 将“时间线分辨率”更改为“8160 x 7200 Immersive”，从而以原生传感器分辨率处理片段。

   如果处理性能有限，可选择“4080 x 3600 Immersive”将分辨率减半，以获得更快的播放性能。
8. “时间线帧率”应设置为每秒90帧。这项参数会在您启用Apple Immersive工作流程时自动配置。
9. 将“播放帧率”设置为每秒90帧，从而最准确地还原沉浸式视频内容。

   如果处理性能有限，或者监视器的刷新率较低，可选择较低的帧率，以获得更稳定的播放性能。建议您选择30fps或60fps等原生帧率的分数帧率。



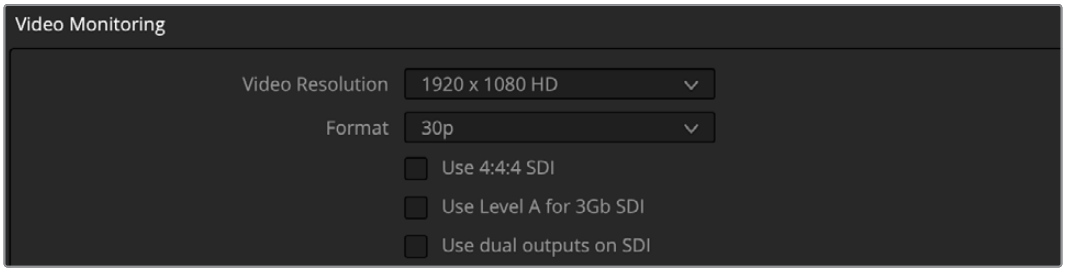
视频监看

如果您使用外部监视器（通过DeckLink或UltraStudio等采集/播放设备），请根据该监视器的具体规格来配置视频监看设置。

1 在“视频监看”类别下，将“视频分辨率”设置为和您的监视器分辨率匹配的选项。例如：1920 x 1080 HD或3840 x 2160 Ultra HD。

沉浸式视频的原生帧率为90fps，但这并不是大部分广播级监视器所支持的帧率。

2 将“格式”设置为和您的监视器帧率匹配的选项。例如：30p。



图像缩放调整

“图像缩放调整”设置可用来确保沉浸式媒体的分辨率和宽高比能够在检视器中准确显示。这些参数会当您在“主设置”中启用Apple Immersive工作流程选项后自动配置，但您依然需要检查这些参数是否正确，以防万一。

1 在“项目设置”的侧边栏中选择“图像缩放调整”。

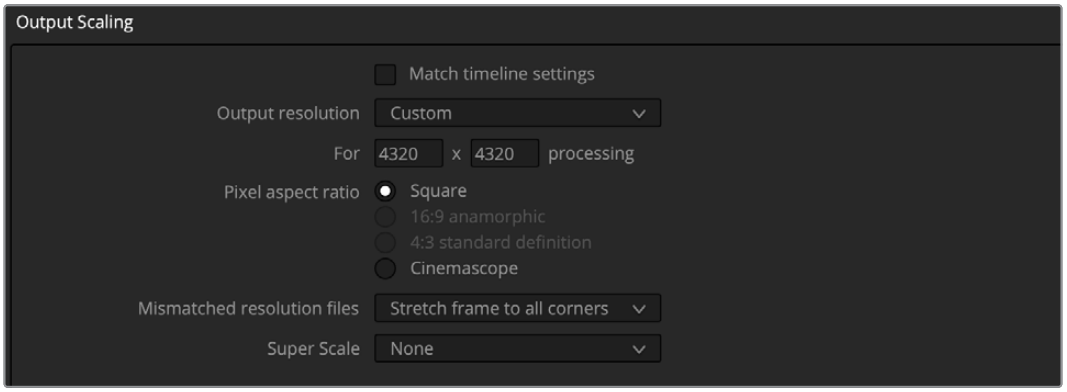
2 在“输出缩放调整”类别下，确保取消勾选“匹配时间线设置”选项。

3 确保“输出分辨率”设置为“Custom”和“4320 x 4320”。

如果使用较小的显示器，或者想要获得更快的播放性能，可将分辨率减半 (2160 x 2160) 。

这些正方形宽高比分辨率是在DaVinci Resolve Studio中播放Apple沉浸视频内容以及流式传输到Apple Vision Pro (通过wifi或Developer Strap) 所必需的。

4 确保将“分辨率不匹配的文件”设置为“将原图拉伸至布满全帧四角”。



色彩管理

由于需要对最大输出尼特亮度进行自定义控制，因此建议您使用调色页面中的“色彩空间转换” (CST) 节点进行色彩管理。“项目设置”中的各项色彩管理设置是为了指定调色页面的时间线色彩空间 (或“工作”色彩空间) ，以及设置与渲染文件相关联的色彩空间元数据。

1 在“项目设置”的侧边栏中选择“色彩管理”。

2 将“色彩科学”保留为“DaVinci YRGB”。

3 将“时间线色彩空间”设置为“DaVinci WG/Intermediate”。

这是一类具有前瞻性的宽色域色彩空间，它支持HDR工作流程，是用于Blackmagic URSA Cine Immersive媒体以及Apple Vision Pro交付的理想选择。

4 将“输出色彩空间”设置为“P3–D65 ST2084 1000 nits”。

这类色彩空间所具备的色域适合在Mac Studio台式机 (通常为Display P3) 上监看DaVinci Resolve的检视器内容。如果您使用调色监视器或者不同的Apple ICC配置文件，就应该将输出色彩空间更改为和您的显示器色彩标准匹配的选项。

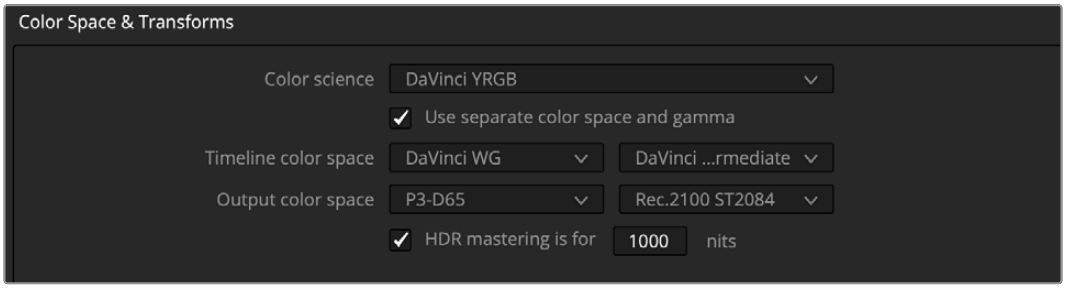
但是，您需要更改伽马 (传递函数) ，使其准确映射到ICC配置文件。

5 点击勾选色彩空间设置上方的“使用独立的色彩空间和伽马”。

勾选后，色彩空间和伽马下拉菜单栏会分成两组。

6 将输出伽马 (位于右侧栏) 设置为“Rec.2100 ST2084”。

7 选择“HDR母版制作亮度为1000尼特”。



为了在今后的Apple Immersive项目设置中节省时间，您可以将当前设置保存为预设。

8 点击位于“项目设置”窗口右上角的三点选项菜单，然后选择“将当前设置保存为预设”。

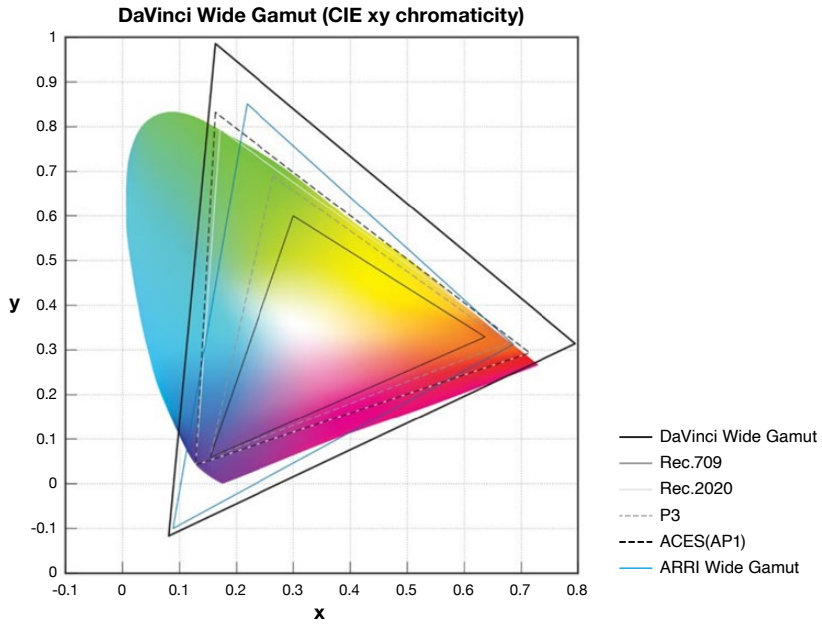
9 为预设命名，然后点击“OK”。

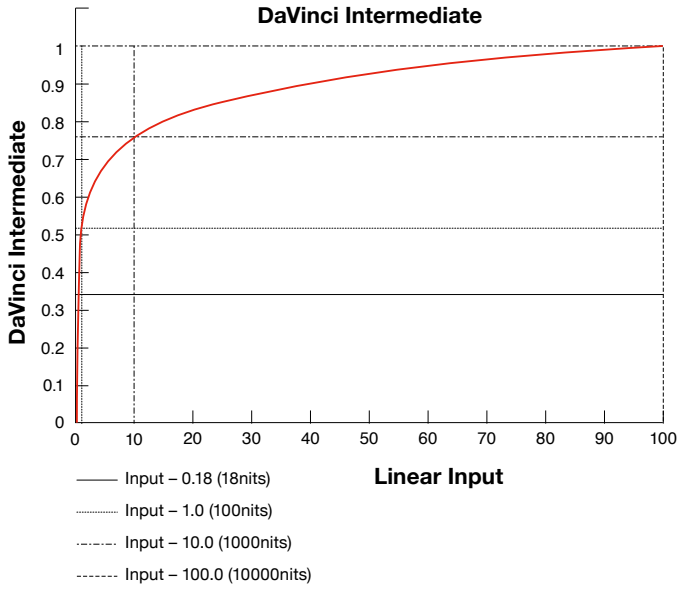
当您下次进行沉浸式媒体制作时，只要打开“项目设置”窗口，点击选项菜单，在列出的预设中选择，就能快速完成项目设置配置。

10 点击“保存”关闭“项目设置”。

关于DaVinci Wide Gamut

DaVinci Wide Gamut宽色域是Blackmagic Design为DaVinci Resolve开发的先进内部色彩空间，它能为调色和图像处理提供相当广泛且精确的环境。当它和DaVinci Intermediate中间片伽马配合使用时，能形成一致的工作空间，从而简化多种摄影机格式之间的色彩管理，确保从输入到输出都具备一致的色调和饱和度映射。这一宽色域非常适合用于HDR和沉浸式媒体制作，它能帮助调色师在制作高保真母版交付格式时进行细微的调整，并且不会出现色带或色调裁切的问题。





Apple Immersive处理

您可以使用“项目设置”侧边栏的“Apple Immersive”类别来设置自定义ILPD (沉浸式镜头配置文件数据)，从而修改沉浸式内容的校准和投影方式。

1 在“项目设置”的侧边栏中选择“Apple Immersive”。

2 点击“Open Calibration Folder”。

3 添加或更新自定义ILPD文件。

4 点击“更新列表”刷新可在媒体池中获取的ILPD内容。

5 点击“保存”关闭“项目设置”。

6 到媒体池中，右键点击一个沉浸式片段。

7 在出现的上下文菜单中，导航到“Apple Immersive Calibration”，然后选择源ILPD (“No Apple Immersive Calibration File Selected”)、自定义ILPD或gap ILPD。

默认情况下，自定义ILPD Calibration文件夹位于“ProIM”文件夹中，和默认的LUT文件夹一起。

摄取

摄取RAW沉浸式内容时，双眼内容和所有关联的沉浸式元数据都会跟随导入的.braw片段。

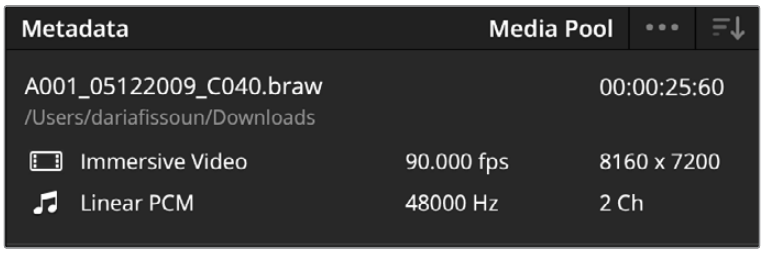
1 前往媒体页面。

2 在“媒体存储”面板中，导航到沉浸式视频内容所在位置。

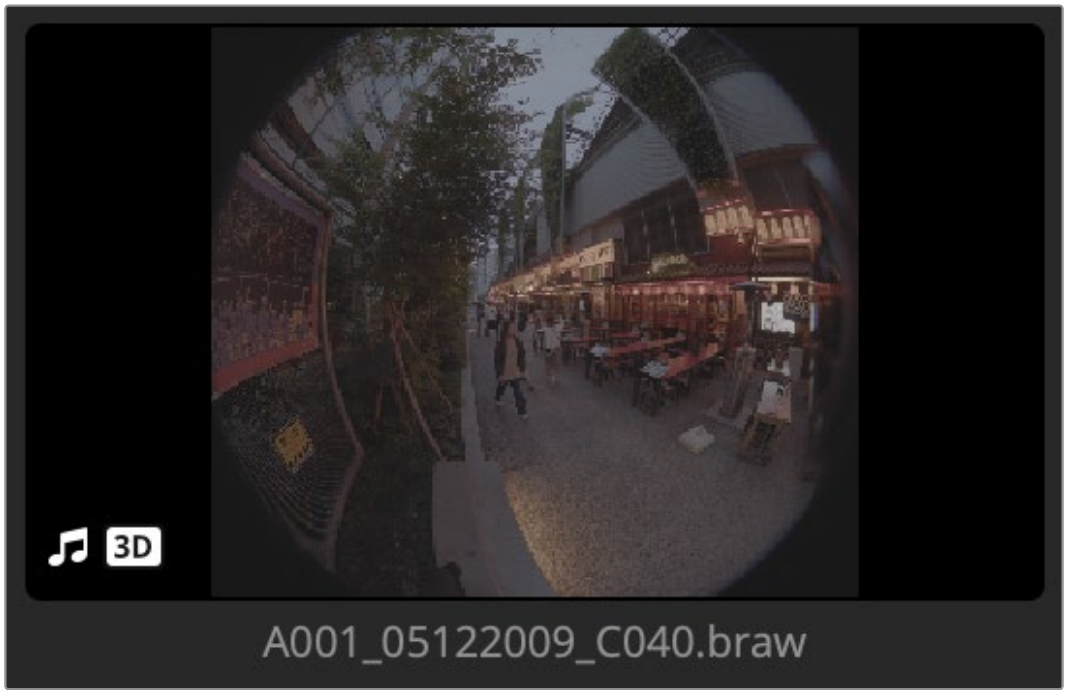
3 将媒体拖放到下方的媒体池中。

提示：如果要保留原始文件夹结构，可将文件夹从“媒体存储”面板拖放到媒体池侧边栏中。拖放到“Master”媒体夹 (或者您创建的任何媒体夹) 后，文件夹和子文件夹都会保留。

选中某个Apple沉浸视频片段后，元数据面板中的编解码参数会表明这是一个沉浸式视频，并且会显示它的原生帧率和分辨率。



此外，视频缩略图左下角的“3D”标志也进一步确认该视频状态为3D片段。



备注：首次将沉浸式媒体拖放到媒体池中时，软件会弹出一个对话框窗口，让您选择保存设置或更改设置。如果选择更改设置，那么项目设置就会自动应用源媒体的帧率和分辨率。

剪辑

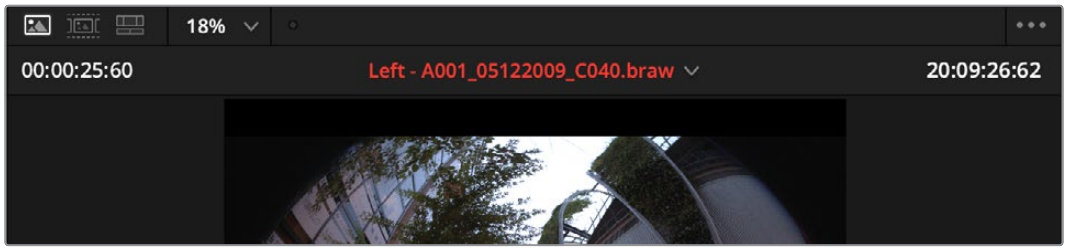
制作沉浸式内容时所使用的剪辑工具及操作和常规2D视频格式环境中的基本相同。下面的内容将介绍沉浸式视频工作流程所特有的剪辑功能和操作。

组合

在组合时间线阶段预览片段时，您可以更改源片段检视器中所显示的眼通道（左眼或右眼）。除此之外，在时间线上组合画面剪接的过程和标准NLE非编工作流程基本一致。

1 前往剪辑页面。

2 双击某个片段，将它加载到左侧的源片段检视器中。检视器上方的标签可提示您当前正在查看哪个眼通道。



3 如果要更改检视器中所显示的眼通道，可在媒体池中右键点击源片段，然后依次选择“立体3D”>“立体3D模式”>“左眼”/“右眼”/“双眼”。

> **提示**：如果在源片段检视器选项中启用了“实时媒体预览”功能，您还可以在媒体池的缩略图上方移动鼠标，从而搓擦浏览源片段检视器中的片段。

4 使用I和O快捷键在源片段检视器中定义一段范围。

5 点击工具栏中的“插入片段”，或使用F9快捷键，将片段插入到时间线上。

> **提示**：您也可将片段从媒体池拖放到时间线面板中。

6 使用鼠标（在“选择模式”或“修剪编辑模式”下）修剪时间线上的片段。

背景

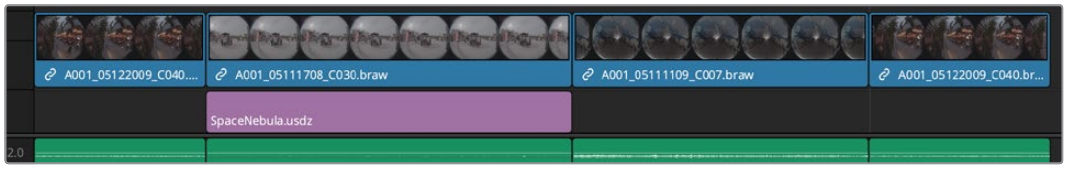
当您在项目设置中选择“Enable Apple Immersive workflow”时，背景轨道（“BD”）会自动出现在“视频1”轨道下方。

背景是指观察者在沉浸式视频内容范围之外所看到的区域，通常限制在180度到210度的视野范围内。

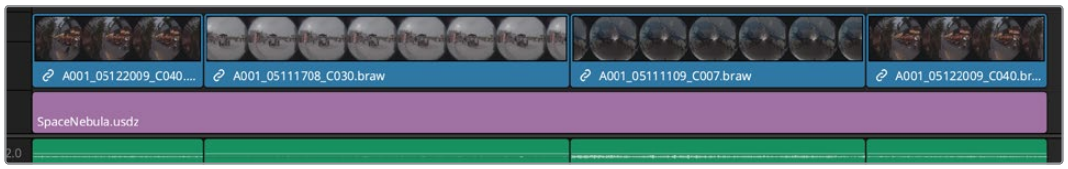
这一区域默认为黑色，但您可以使用背景轨道插入自定义背景图像（采用.usdz格式）。

1 将.usdz背景文件拖放到背景轨道上，从而添加到时间线。

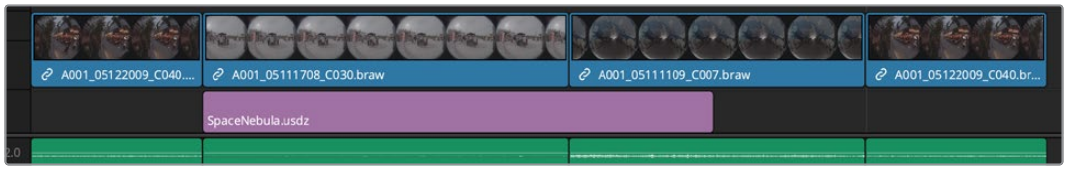
背景图像的时长必须和视频轨道中的片段边界对齐。单个背景文件可以跨一个片段、多个片段或整条时间线，但它不能在沉浸式.braw片段的中间进行剪接（也叫“L剪切”）。



一个片段 — 正确 ✓

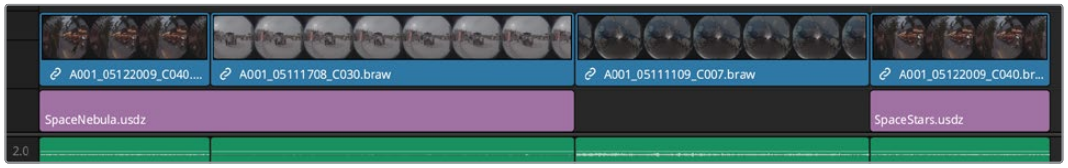


多个片段 — 正确 ✓



L剪切 — 错误 ✗

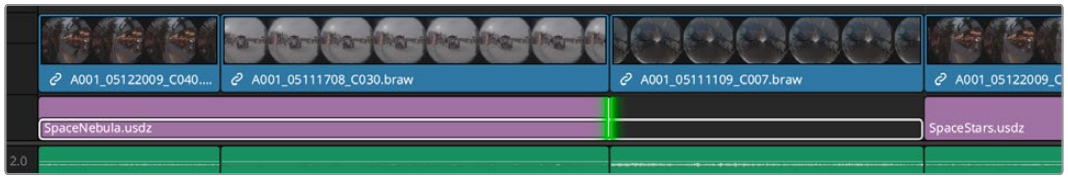
您可以选择在单个时间线上使用多个背景文件，或者只用默认的黑色背景，完全不添加任何图像。



不同图像和空隙 — 正确 ✓

2 使用标准修剪控制来调整背景文件的长度。

拖动.usdz文件的边缘（在“选择模式”或“修剪编辑模式”下）调整长度。



您可以使用“,”（逗号）和“.”（句号）命令，将修剪边缘逐帧挪动。

提示：为了确保背景始终对齐沉浸式视频编辑点，请在剪辑页面工具栏中启用“吸附”模式。

使用检视器

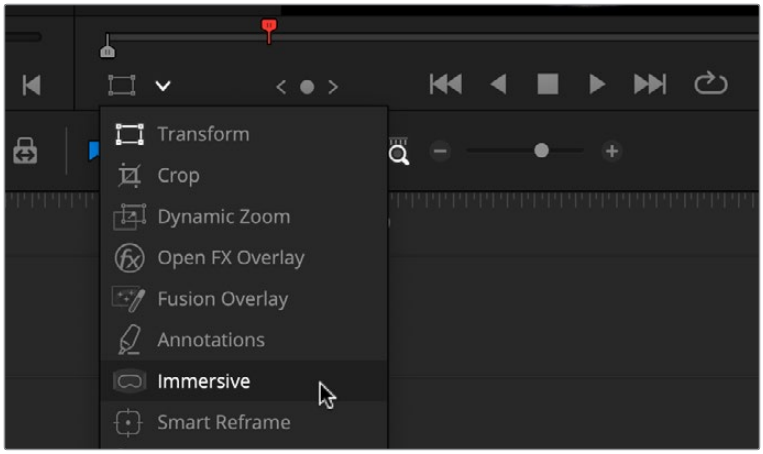
默认情况下，沉浸式片段在DaVinci Resolve的页面检视器中会以圆形的“Lens Space”（镜头空间）显示。用于投影图像的视觉镜头校准源自摄影机，并且对于每个片段都是不同的。



您还可以选择将视图模式更改为LatLong或Viewport模式。

1 点击位于时间线检视器左下角、屏上控制按钮一侧的下拉箭头。

2 选择“Immersive”。



此时，检视器上方会出现一个新的工具栏，里面提供更多查看选项。

3 点击“LatLong”可查看图像的等距柱状投影。这一模式可以对图像进行变形处理，使其更接近传统的2D媒体比例，从而简化部分后期制作过程，例如跟踪和合成。



4 点击“Viewport”可以在正方形画幅内查看沉浸式场景的中心裁切部分。这一模式能帮助您在标准监视器上以正确的比例清晰查看场景内细节，但查看时请注意不要遗漏任何画幅外的内容。

您可以使用顶部沉浸式工具栏中的变换控制对画幅中的立体影像进行竖移和平移操作，并且调整横向和纵向视野。



转场

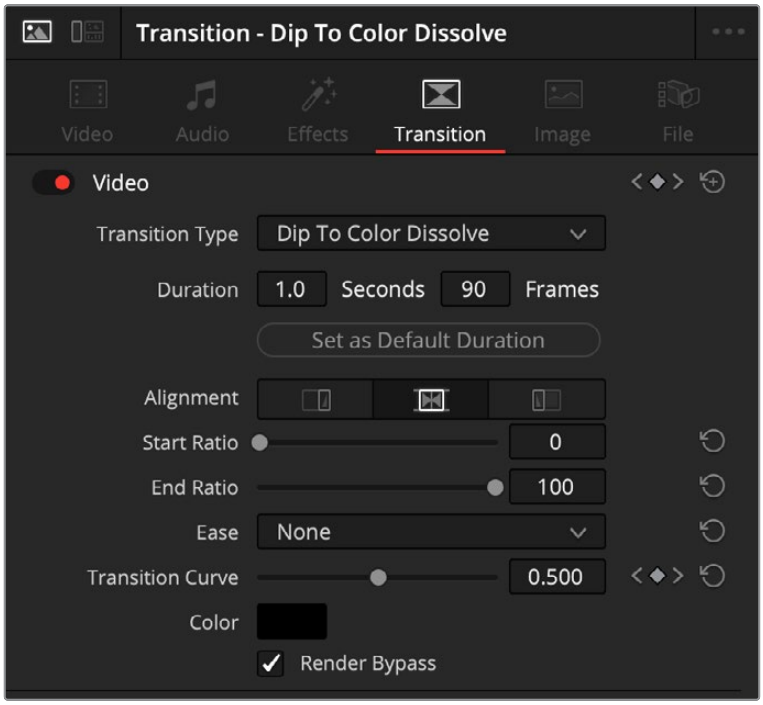
要在两个沉浸式视频片段之间进行转场，您可以执行硬切，使用片段两侧的控制柄淡入或淡出，或者使用“浸入颜色叠化”效果。转场效果不应烧录到素材中，而是应该由Apple Vision Pro在播放过程中自行渲染。

1. 打开“特效库”面板。
2. 在侧边栏中，依次导航进入“工具箱”>“视频转场”。
3. 将“浸入颜色叠化”拖放到两个片段之间的剪切点上。



备注：两个片段的剪切点之外必须具备足够的修剪数据才能应用转场。如果不具备足够的修剪数据（也就是“余量”），就无法应用转场。

4. 在“检查器”面板中，导航到“转场”选项卡，然后选择“Render Bypass”。这样能确保Apple Vision Pro正确渲染叠化效果。
5. 接下来，您可以使用“浸入颜色叠化”检查器中的其他参数来修改转场的时长、对齐方式、动画速度以及色彩等方面。



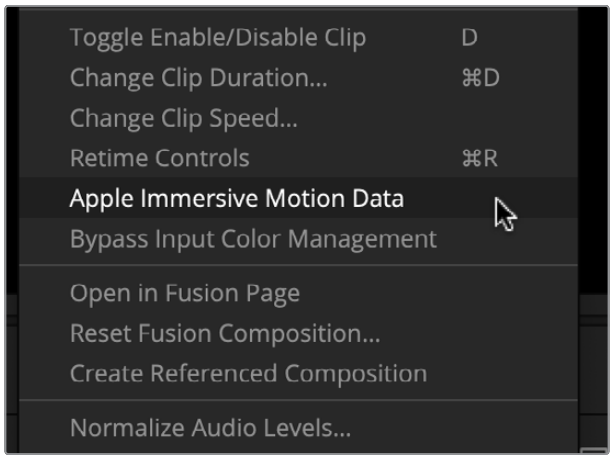
Apple Immersive Motion Data

处理沉浸式内容时，快速或突然的摄影机移动会使观众迷失方向。为了保证观众的舒适感，应避免使用含有快速平移/竖移、旋转运动或不平衡水平线的镜次。

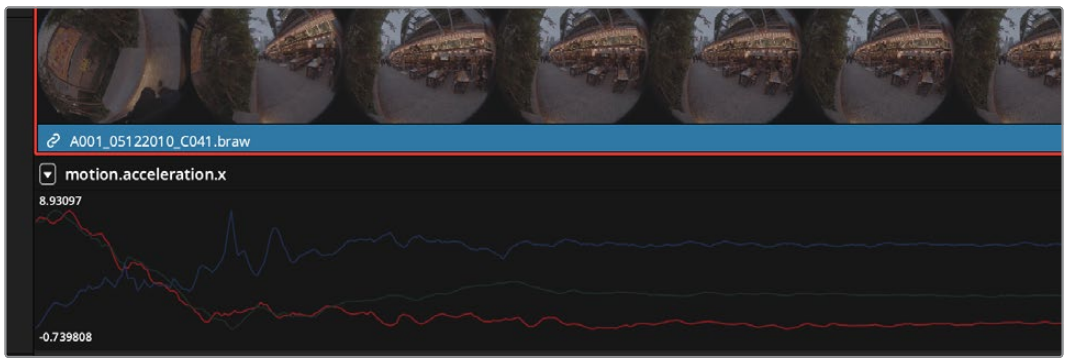
Apple Immersive Motion Data (Apple Immersive运动数据) 功能用图表表示x、y和z加速及陀螺仪位置的值，从而客观反馈摄影机运动和移动过强的地方。剪辑师能够利用这些信息避免使用令人迷失方向的影像部分。

1 在时间线上右键点击一个沉浸式片段。

2 选择“Apple Immersive Motion Data”。

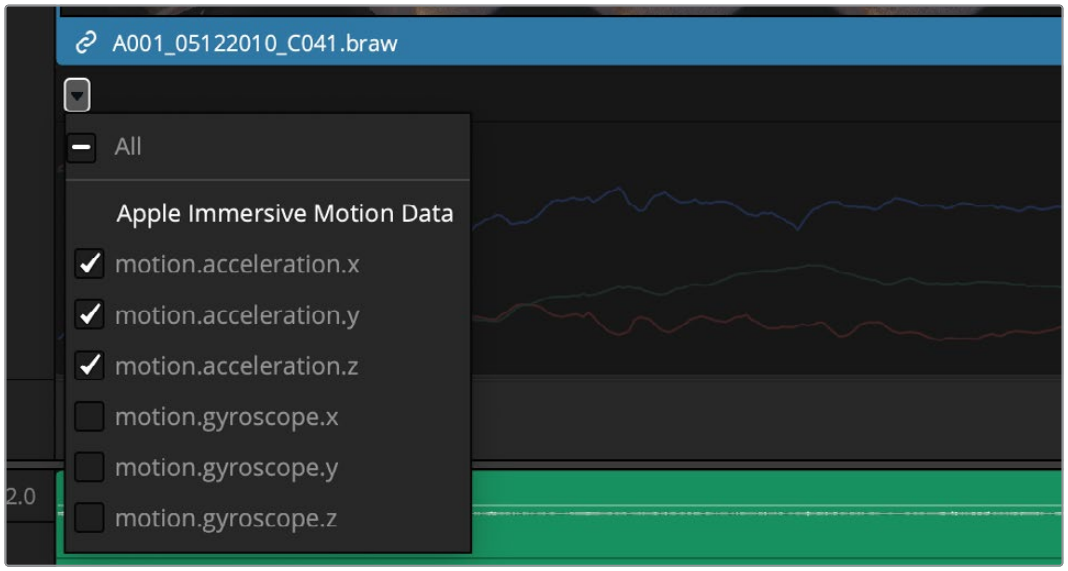


视频轨道中会出现图表，显示x、y、z加速率。



在这个例子中，摄影师在设置摄影机的同时，摄影机处于录制状态。图表中的尖峰表示摄影机被拿起、旋转、放置在他肩膀上时发生的突然移动。理想状态下，您最好使用摄影机放稳后，也就是视频后段读数较平的部分。

3 点击图表左上角的下拉箭头。



您可以根据镜头内容和摄影机运动的类型启用或禁用读数。

标题

您可以在剪辑页面使用“特效库”面板中的任何标题工具为沉浸式场景添加文本，包括文本 (Text) 、文本+ (Text+) 、多文本 (MultiText) 、滚动 (Scroll) 和任何下方字幕条类型。

当放置在沉浸式片段上方时，标题发生器会使用片段的ILPD (沉浸式镜头配置文件数据) 对文本进行准确的扭曲处理，从而匹配场景的镜头变形。

如果没有可用ILPD (比如为黑色背景或沉浸式CGI场景添加文本时) ，将会应用默认的镜头畸变来确保在Apple Vision Pro中观看时，文本仍会显示适当的变形。

1 在“特效库”面板中，导航至“工具箱”>“标题”。

2 将“文本”发生器拖入时间线。将其放置在沉浸式片段上方的视频轨道上，整合到场景中。



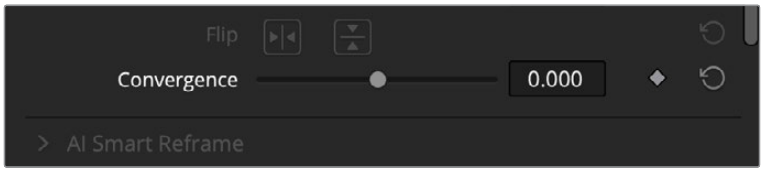
和背景一样，文本发生器不能跨越片段边界。由于每个片段都是用单独的Lens Space镜头空间渲染，任何重叠的标题都有在镜头画面之间变形的风险。

3 将标题修剪到视频编辑点。

4 使用检查器写入标题文本并调整格式。

5 前往“设置”选项卡，使用“会聚”滑块将文本放置在3D空间里。

将滑块向左移动会在空间中把标题推后，向右移动会向前推。



观察会聚最佳的方式是在Apple Vision Pro上监看。如果使用其他监看方式，您可以通过在调色页面修改底层视频图层的Vision模式，更改文本会聚在检视器中的显示方式。

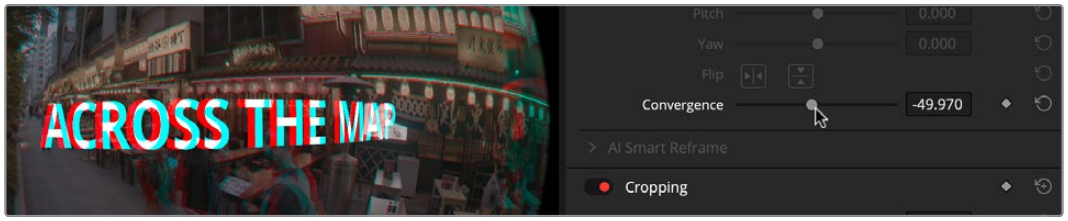
6 前往调色页面。

7 打开3D面板。

面板中间的“Vision”栏可用于更改沉浸式媒体投影时使用的立体格式。

8 选择“Stereo”选项卡。

9 将“Out”模式更改为“Anaglyph (Color)”。



在剪辑页面的检查器面板中对于文本发生器会聚参数的更改，如果通过红蓝 (或anachrome) 眼镜查看会显得更加直观。

创意考量

在Blackmagic URSA CIne Immersive摄影机和Apple Vision Pro制作初期，电影制作者应尝试这种格式，寻找最佳方式，建立新的规定，并发现之前传统电影制作无法想象的工作流程。

出于这样的考量，沉浸式制作的一些方面已经被选定需要特殊考虑，从而确保观众接受度和视觉舒适度的最大化。

1 剪辑应在前期制作阶段开始。由于对镜头长度和观众运动舒适度的特殊考量，电影制作者需要在拍摄场景时就考虑到最终剪辑。

2 避免或限制使用带有快速摄影机移动的片段，因为这些片段可能会让观众迷失方向。应选择使用固定镜头或缓慢平移和Dolly车。

3 失去平衡或极端的摄影机角度也会导致观众迷失方向，应选择与拍摄物体保持水平的镜次，并保持舒适的距离 (不要太近) 。

4 与观看2D场景相比，沉浸式媒体中的新环境需要更久的时间适应。应该放缓切换速度，让观众有时间适应新的环境。

5 在片段之间切换时，应注意观众在上一个镜头中看向哪里。如果在上一个镜头中，观众的注意力集中在画面右侧，那么下一个镜头中的重要内容就不应该发生在画面左侧。

6 空间音频能推动大部分叙事，因此，请将其作为沉浸式制作的另一项特征。

视觉特效合成

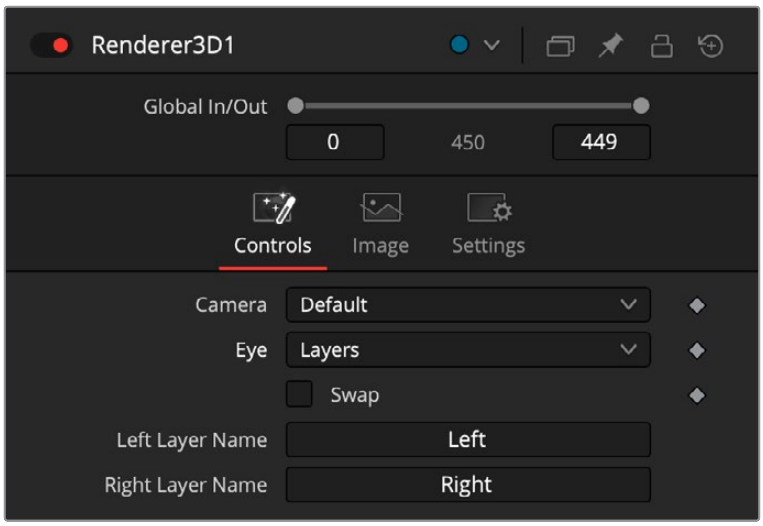
在项目设置中启用Apple Immersive工作流程时，专用的Fusion参数能够实现立体工作流程和符合沉浸式要求的合成。MediaOut（媒体输出）节点拥有左右眼输出选项，Immersive Patcher（沉浸式补丁）工具能够实现等距柱状投影和合成。

媒体输出节点

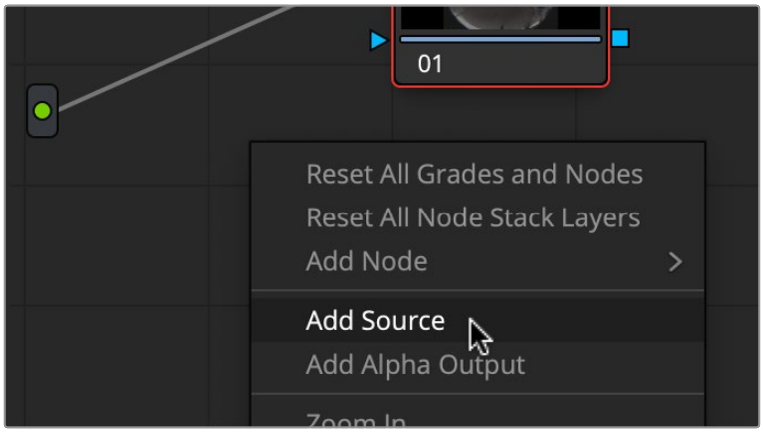
当您处理沉浸式片段首次进入Fusion页面时，节点图包含两个MediaIn（媒体输入）节点，与两个相应的MediaOut（媒体输出）节点连接。



在MediaOut（媒体输出）检查器中，“Stereo Eye”参数可用于指派眼信号输出。



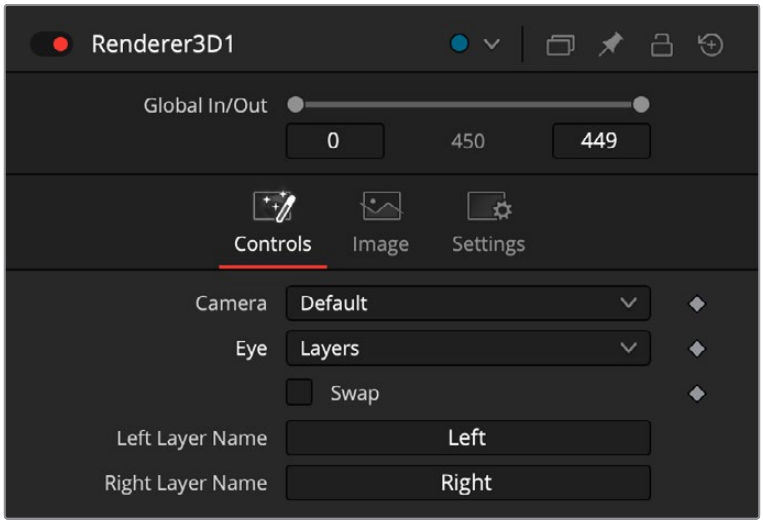
MediaOut1（媒体输出1）被默认设置为左眼，MediaOut2（媒体输出2）被默认设置为右眼。这样的配置让您可以立刻开始组合符合立体要求的合成以及图文叠加。双媒体输出节点还会促使两个信号种类被发送到调色页面的节点编辑器进行进一步图像优化。您可以在节点编辑器中右键点击，然后选择“添加源”，在调色页面中访问MediaOut2（媒体输出2）的信号。



当“Stereo Eye”参数设置为自动时，媒体输出节点会合并信号，并将其指派到时间线上相应的眼睛。这样您就可以使用Fusion的多图层工作流程把左右眼信号发送到单个媒体输出节点。

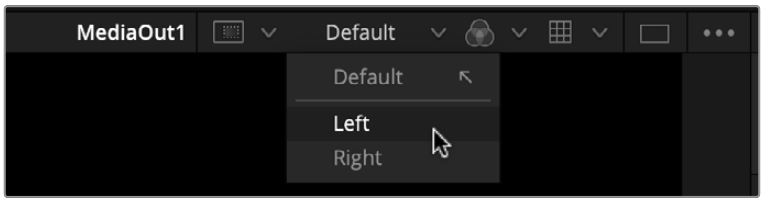
Renderer3D（渲染器3D）

在沉浸式时间线上处理3D图文或CGI时，Renderer3D（渲染器3D）节点拥有多项眼设置，并能够处理左右图层。



这样就可以创建复杂的3D标题或整个CGI场景，然后直接发送到沉浸式项目中的立体时间线。

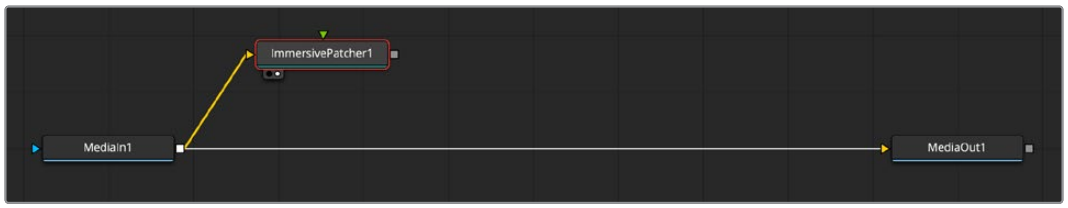
以这种设置开展工作时，您可以使用Fusion检视器右上角的下拉菜单，来回查看合成的左眼和右眼情况。



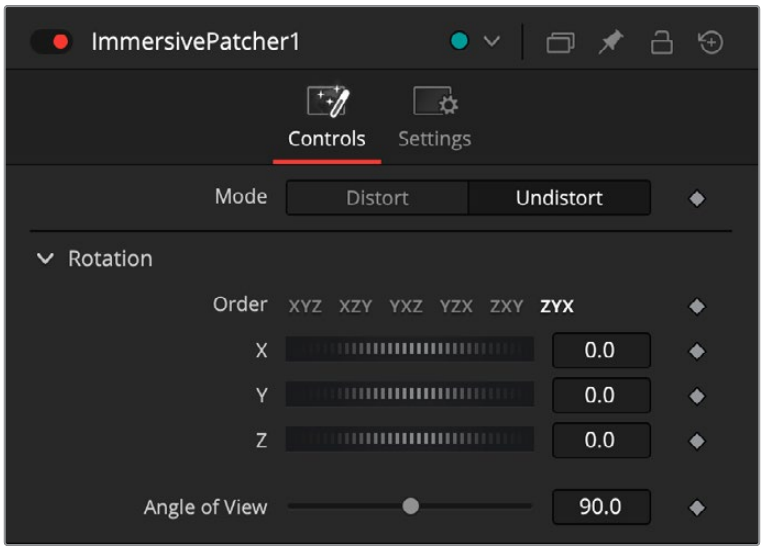
使用Immersive Patcher合成

Immersive Patcher工具让您可以暂时取消沉浸式片段中某部分的变形，从而在一个平面上更容易地跟踪、绘图和合成。完成后您可以使用第二个Immersive Patcher对沉浸式信号进行重新变形，然后把新合成的元素整合进场景中。

1 将Immersive Patcher添加到节点流程中。



2 在检查器中点击“取消扭曲”。



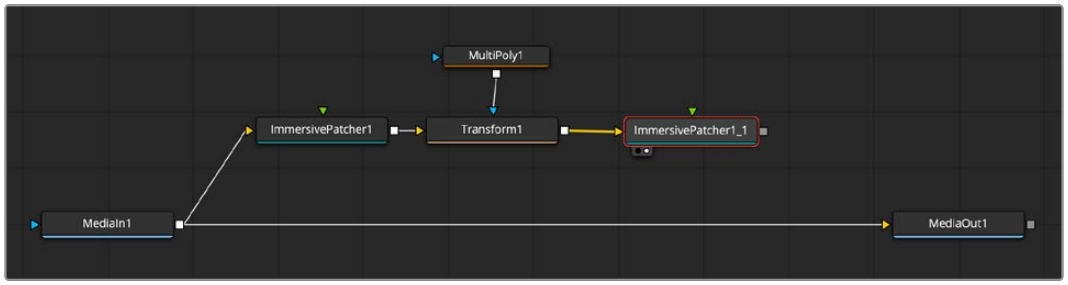
3 使用X、Y和Z旋转参数选择沉浸式图像中的一部分进行展平处理。

4 按照处理标准2D视频信号的方式进行合成。这包括（但不限于）使用标准画笔方法移除不想要的物体，用变换和多边形采样画面中的部分进行场景延伸，或者应用蒙版、文本和外部图像/视频图层来替换标识或屏幕。

完成后您需要对平面图像进行重新变形处理，从而匹配原始沉浸变形。

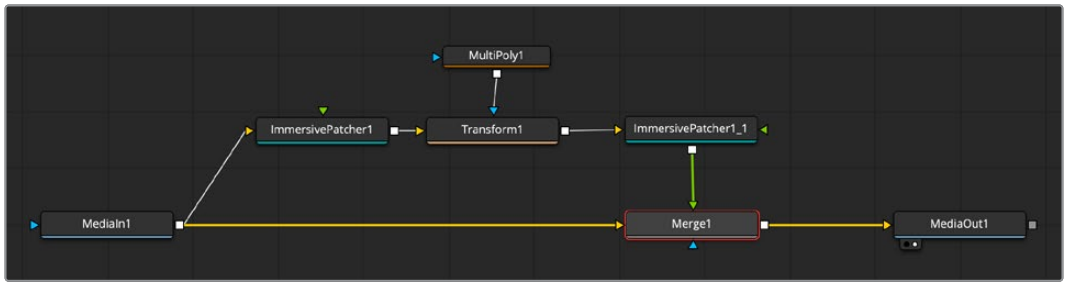
5 复制Immersive Patcher，保留原始旋转和视角信息。

6 在合成节点之后粘贴Immersive Patcher，然后连接到合成流程的末尾。



7 在检查器中点击“扭曲”。

8 将合成补丁与原始片段合并，把特效整合进沉浸式场景中。



如果以协作方式工作，您可以退出Fusion页面，然后在DaVinci Resolve中继续为沉浸式时间线进行剪辑、混合和调色。

如果执行沉浸式交付，将每只眼睛以ACES AP0渲染为EXR，最大限度保留属性进行沉浸式交付。

调色

这部分介绍在DaVinci Resolve调色页面中处理沉浸式媒体时，您可以使用的光学、调色和3D工具。

由于沉浸式内容性质独特，后期制作时不是所有工具都能使用。其他工具目前无法使用，但正在调整过程中以支持沉浸式内容。

降噪

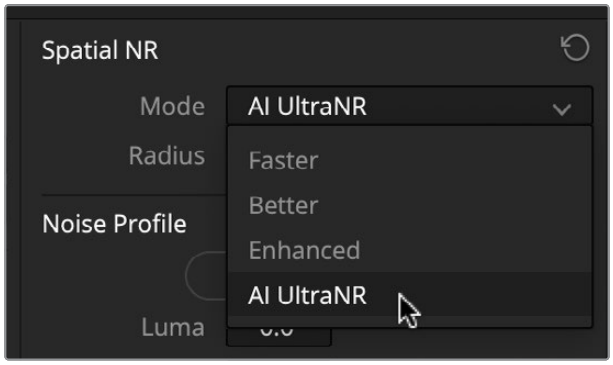
处理沉浸式视频时可以进行降噪处理。“运动特效”面板中的“空域降噪”工具可以分析源媒体的噪波图案，从而将其最大限度降低，同时保留原始图像细节。

1 打开调色页面工具栏的“运动特效”面板。

您可以在“空域降噪”栏中选择希望使用的降噪模式。

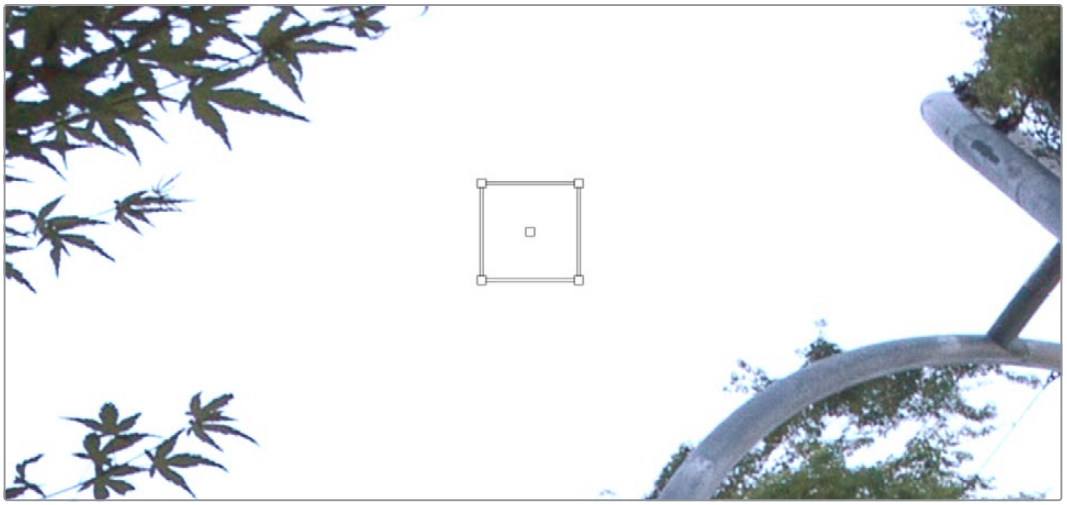
- *“更快”*、*“更好”*、*“更强”*模式可以在给定的半径内执行图案分析，然后根据设定的阈值配置输出。它们会按照处理复杂程度的顺序列出。
- *AI UltraNR*超级降噪是一种AI驱动的噪波图案识别器，它能通过定位并分析画面中“最平”的部分来识别出纯噪点，从信号中提取噪波图案。

2 将空域降噪模式设置为AI UltraNR超级降噪。



3 点击“分析”。

检视器中会出现一个方框，可以识别出给定帧内像素数据“最空”的区域。由于没有实际视觉内容，该方框中的数据非常适合用于提取摄影机传感器产生的数字噪点。



您可以根据需要调整检视器中方框的位置和大小，但通常不建议这么做，因为AI UltraNR超级降噪比人眼更擅长检测最佳的分析区域。

您也可以手动调整“Noise Profile”中的“亮度”和“色度”阈值设置，但同样的，AI UltraNR超级降噪会自动实现理想的阈值混合，完成干净的降噪处理。

一级调色

一级调色的操作方式与普通视频影像一样。在使用一级校色轮、一级Log色轮、HDR调色面板以及HSL曲线进行归一化、平衡和匹配等处理时，请采用同样的思考方式。色轮六矢量切片器和色彩扭曲器面板也如预期一样正常工作。

二级调色

进行沉浸式媒体制作时，限定器等基于像素的选择方法与常规媒体制作无异。

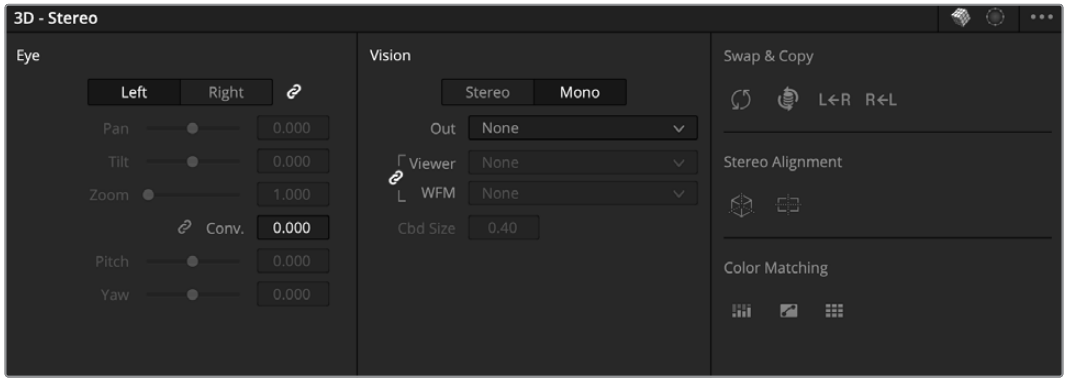
基于对象或基于窗口的选择方法只有在扁平的等距柱状投影（或视口）环境下才能准确运行，也就是说大多数要求精准限定器形状的二级调色方法目前暂不可用。这包括Power Window窗口（以及跟踪器面板）和神奇遮罩工具。

调整大小

沉浸式内容没有可用的调整大小控制工具（比如变焦和平移）。

3D面板

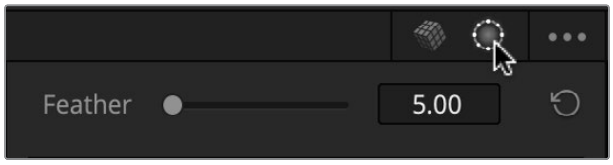
调色页面的3D面板拥有管理多种3D摄影机和图文格式所需的控制。与其他立体格式不同，您不需要调整眼睛的对齐方式，因为Blackmagic URSA Cine Immersive摄影机镜头、焦距和光圈是固定的。



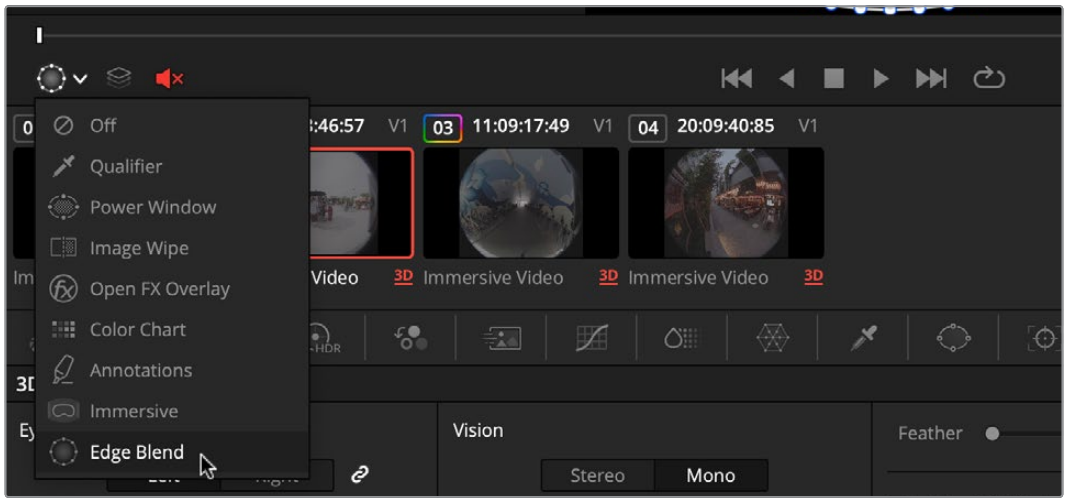
Edge Blend模式仅适用于沉浸式内容，让您可以调整和柔化沉浸式视频画面的边缘。这项功能可以用于隐藏可见的电影设备，并限制观众的视野。超过边缘的部分，观众将看到背景。

1 打开3D面板。

2 在右上角将模式更改为“Edge Blend”。



3 确保检视器的屏上控制设置为“Edge Blend”。



此时，视频画面的外缘应该会出现一圈控制点，中间有一个锚点控制柄。

4 拖动控制点可更改位置。



或者右键点击删除一个点。

5 在3D面板的右侧栏，使用“Feather”参数柔化画面的边缘。

6 选择“visionOS Effect”，确保在Apple Vision Pro上播放的过程中，新的边缘数据被准确处理。如果没有选定，边缘会被烧录到已渲染的视频文件中。

7 在左侧栏，使用“Left”和“Right”左右选项卡在检视器中进行两眼之间的切换。

请注意，3D面板中的许多控制会显示灰色。当不适用于正在处理的媒体时，DaVinci Resolve中的参数会显示灰色，在这个情况下是沉浸式内容。

8 在中间栏，使用“Stereo”和“Mono”选项卡在沉浸式视频片段单眼和双眼镜头检视器之间切换。

CST色彩管理

对调色页面节点编辑器中的节点使用“色彩空间转换” (CST) 效果可实现准确的色彩映射，并对时间线的最大输出尼特亮度进行自定义控制。

推荐的工作流程包含4个CST节点:

1. INPUT CST – 摄影机到工作色彩空间。

   校色发生在这两个节点之间。

2. OUTPUT CST – 工作色彩空间到HDR Mac显示器。
3. MONITOR CST – 1000尼特到250尼特用于HDR监看。
4. AVP CST – 250尼特到108尼特用于Apple Vision Pro监看和交付。



CST时间线映射

首先，您将从摄影机源输入色彩空间映射到“工作”时间线色彩空间。Blackmagic URSA Cine Immersive摄影机以Blackmagic Design Wide Gamut Gen 4 / 5色域运行，并使用DI (达芬奇中间片) 对数曲线。

1. 打开“特效库”面板。
2. 将“色彩空间转换”拖放到节点编辑器中的一个空白节点 (01) 上。

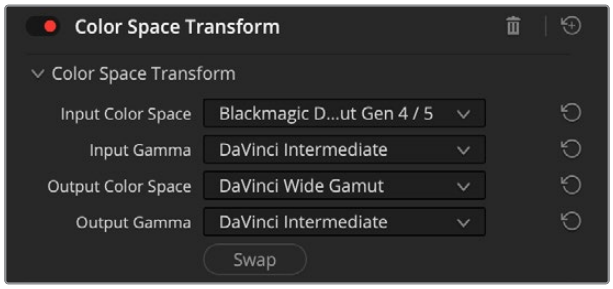
   如果是从输入色彩空间映射到时间线色彩空间，那么该节点就应该是节点树中的首个节点之一，位于任何调色操作之前。

   如果是从输入色彩空间直接映射到监看或交付色彩空间，那么该节点就应该是节点树中最后的节点之一，位于任何调色操作之后。

3. 将节点01标注为INPUT CST。

> **备注**: 操作步骤中使用的节点标签是为了便于参考文档和屏幕截图。您可以使用任何适合您的命名格式，或者直接绕过标注。

4. 将“输入色彩空间”设置为“Blackmagic Design Wide Gamut Gen 4 / 5”。
5. 将“输入伽马”设置为“DI (达芬奇中间片) ”。
6. 将“输出色彩空间”设置为“DWG (达芬奇宽色域) ”。
7. 将“输出伽马”设置为“DI (达芬奇中间片) ”。



8 创建新的串行节点 (02) 。

9 将第一个节点 (01) 复制粘贴到新节点 (02) 上。

10 将节点02标注为OUTPUT CST。

请务必将输出色彩空间和伽马作为新串行节点中的输入，从而确保节点树时间线色彩空间的一致性。

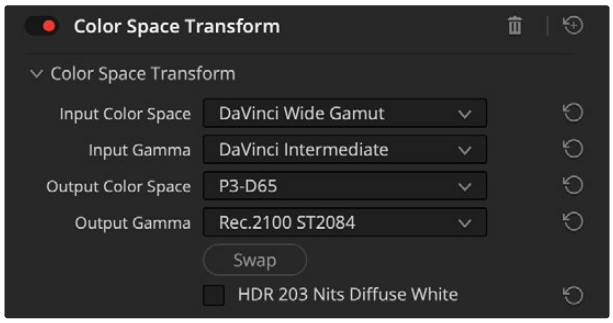
11 点击“交换”按钮切换输入和输出参数。

输入色彩空间和伽马现在适合这个CST节点了。

12 将“输出色彩空间”设置为“P3–D65”。这是当前Mac显示器的默认色域。

13 将“输出伽马”设置为“Rec.2100 ST2084”，也就是1000尼特信号的HDR格式。

所有调色操作将在这两个节点之间发生。使用基于节点的色彩管理进行处理时，您还具有在时间线色彩管理范围之外执行操作的独特能力，许多调色师会利用这种能力添加特定的调色操作、特效或第三方元素。



ICC配置文件

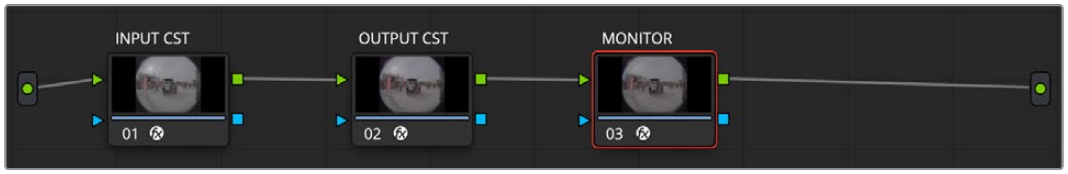
为了进一步确保Blackmagic URSA Cine Immersive信号被准确映射到Mac显示器，请前往“系统设置”>“显示器”，确认“颜色描述文件”预设被设置为“Apple XDR Display (P3–1600 nits)”。

CST调色监视器映射

您可以使用更多CST暂时限制调色过程中监看的最大尼特输出值。

1 在色彩管理“OUTPUT CST” (02) 之后创建新的串行节点 (03) 。

2 将节点03标注为MONITOR。



3 将“色彩空间转换”特效添加到“MONITOR”节点 (03) 。

输入值必须匹配来自之前CST的输出信号。

4 将“输入色彩空间”设置为“P3–D65”。

5 将“输入伽马”设置为“Rec.2100 ST2084”。

这个CST节点的目的只是把图像的亮度映射到更小的尼特值，这意味着输入和输出参数可以保持不变。

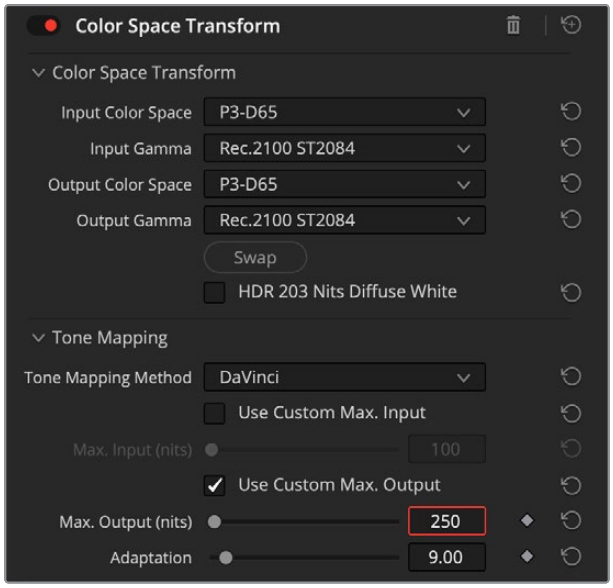
6 将“输出色彩空间”设置为“P3–D65”。

7 将“输出伽马”设置为“Rec.2100 ST2084”。

备注：在一些情况下，如果输入和输出完全相同，就没有必要设置CST的色彩空间或伽马 (因为没有发生映射) 。在这个情况下则有必要，因为自定义尼特映射的应用会受到信号对数曲线值和形状的影响。

8 选择“使用自定义最大输出”。

9 将“最大输出 (尼特) ”设置为250。



现在，1000尼特信号将被映射到250尼特 (使用DaVinci色调映射方式) 。

这样会使画面紧密匹配HDR调色监视器 (比如Flanders XMP310，索尼BVM–HX310等) 上渲染的Apple Vision Pro信号。

然而，如要在Apple Vision Pro上监看项目，您需要执行另一项色调映射操作。

CST Apple Vision Pro监看

在节点流程的末端使用CST节点，可用于实时监看、预览以及向Apple Vision Pro设备最终交付。

1 在“MONITOR”节点 (03) 之后创建新的串行节点 (04) 。

2 将“MONITOR”节点 (03) 复制粘贴到节点04。您将使用同样的输入和输出参数，仅更改尼特值。

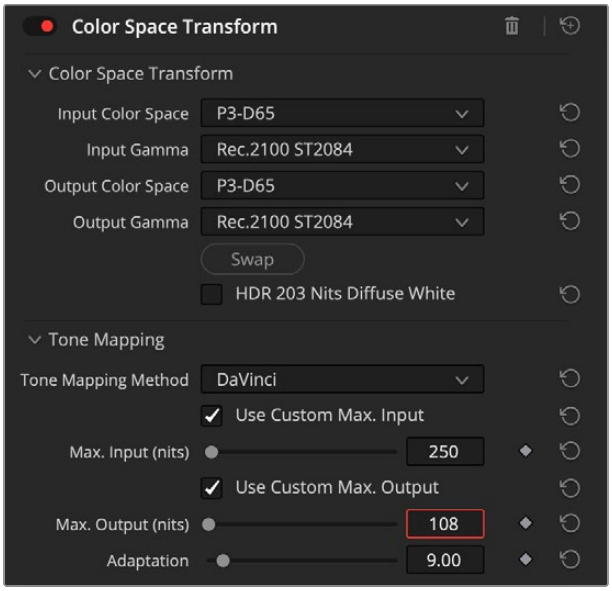
3 将节点04标注为AVP。



4 选择“使用自定义最大输入”。

5 将“最大输入 (尼特) ”设置为250，从而准确解读来自“MONITOR”节点的映射信号。

6 将“最大输出 (尼特) ”设置为108，也就是监看和向Apple Vision Pro交付的最大尼特值。



您现在可以继续调色，根据工作流程的需要禁用CST节点。例如，在最初调色过程中您会用到外部调色监视器，应该禁用“AVP”节点 (04) 来确保在HDR调色监视器上准确看到250尼特信号。切换到使用Apple Vision Pro预览，或者进行最终交付时，重新启用“AVP”节点 (04) 。

CST能否被放置在时间线节点编辑器里?

不能。目前，片段的色调映射取决于色彩空间转换适应性参数的准确应用，该步骤必须以片段为单位逐一进行。

音频

Apple Vision Pro和DaVinci Resolve支持空间和Ambisonic音频，AVP (Apple Vision Pro) 音频及AIA (Apple沉浸音频) 格式。

处理空间音频时，音频后期制作的许多方面保持不变。主要区别之一是与其通过高端扬声器监听，音频工程师应当使用耳机或Airpods获得真正的空间沉浸感，以便更好地再现观众的聆听环境。

在2D混音中，大部分对白会被混合到中央声道，而空间音频需要被混合到与声音来源相对应的位置。处理时，必须使用闪避、均衡和平衡来混合音频元素，避免让听者感到不知所措或困惑。

沉浸式空间元素可以和普通立体声元素结合使用，这样无论听者的头部如何移动，特定的声音元素都能始终保持在固定的位置。这样做能使配乐和旁白等非剧情声清晰感知且容易解读。

在空间音频中，环境声往往会被感知得更为微弱，因此通常需要加大其混音处理的力度，才能被听者感知。但不应以造成听者迷失方向为代价。

混响对沉浸感的帮助巨大，因为这是听者每天都会有的自然体验，在既定环境中产生的每一种声音都会在听者周围的表面上反弹，产生“绽放”的混响质感。因此空间音频设计鼓励使用混响，从而确保完全的沉浸感。

Apple沉浸音频 (AIA) 格式是多种格式的载体：

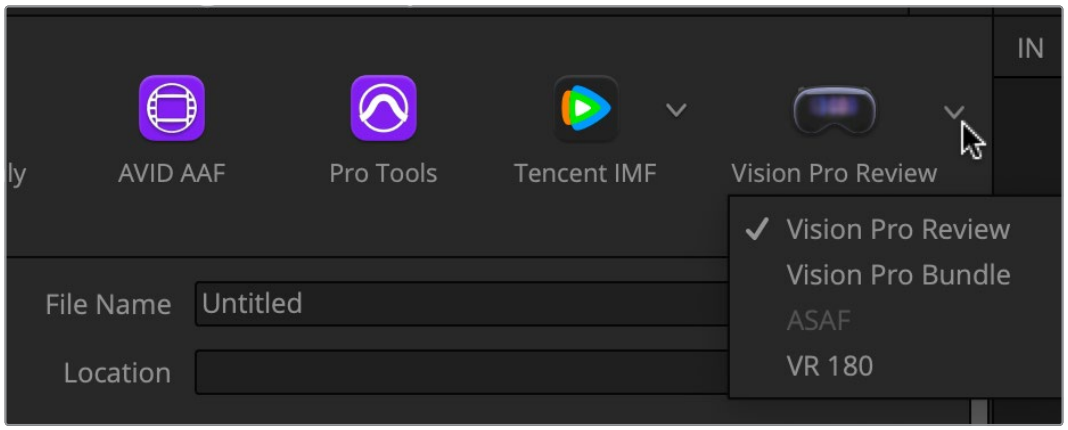
- **Apple空间音频格式 (ASAF)** 是原始的工作音频格式。
- **Apple Positional Audio Codec (APAC)** 是一种可以在支持APAC的设备上播放的.mp4编码格式，比如Airpods，或者来自Apple Vision Pro上的AIVU文件。

Apple空间音频格式 (ASAF)

Apple空间音频格式 (ASAF) 是主要为Apple Vision Pro的VR和AR应用所开发的沉浸式音频格式。ASAF为macOS独有，受到ProTools (通过插件) 和DaVinci Resolve (内置) 支持。ASAF拥有先进的技术能力，包括内置Ambisonics源支持 (称为“Scenes”) 并能通过总线指派进行指派。混合通过虚拟形式完成，没有物理总线，用户可从多种监听格式中选择，包括双耳，通过Apple AirPods的直接整合实现头部追踪。声像处理通过DaVinci Resolve现有的工具完成，包括声像对话框、自动化、SpaceView和检视器内声像工具，检查器中有二级控制。这些功能可以实现声源在空间内的精准放置，包括自动跟踪LatLong视频。ASAF混音能通过“Immersive Audio”>“Export Master File”选项，以WAV或MP4格式被导出（带配套元数据），但目前无法在DaVinci Resolve里被导入或播放。

交付

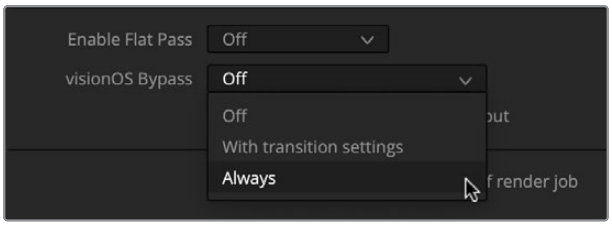
交付页面中的两个渲染设置能够渲染Apple Vision Pro文件用于检查和最终交付。



- **Vision Pro Review** — 创建一个小型MV–HEVC数据包供预览使用。Apple会通过Wi–Fi将最终文件“共享”给Apple Vision Pro，可在Apple Immersive Video实用程序 (AIVU) App中观看，该实用程序可从App Store下载。
- **Vision Pro Bundle** — 创建一个大型ProRes文件包，可用于存档或通过Apple Compressor处理，为Apple TV播放做准备。
- **ASAF** — 以WAV或MP4格式渲染ASAF音频混合。
- **VR180** — 生成VR180规范立体视频文件。

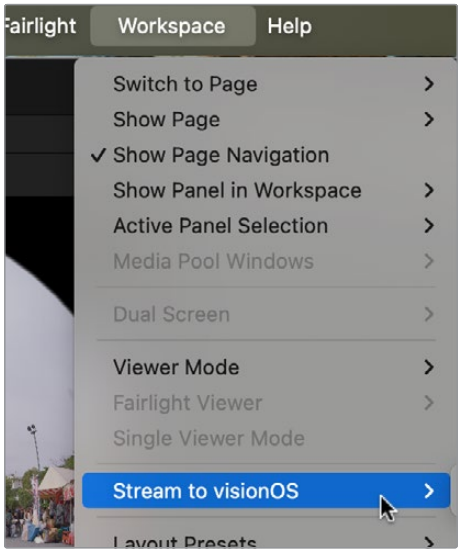
绕过转场

如果沉浸式时间线上添加了转场，将被预渲染，然后变为单一播放轨道。为沉浸式内容生成的XML会描述在这些渲染流之间切换的过程 (影像，然后预渲染转场，然后回到影像)。“渲染设置”面板的“高级设置”中提供了不同的绕过visionOS转场选项，具体包括：始终绕过 (“Always”)，根据在时间线中的状态应用 (“With transition settings”)，或者在时间线中出现时包括 (“Off”)。



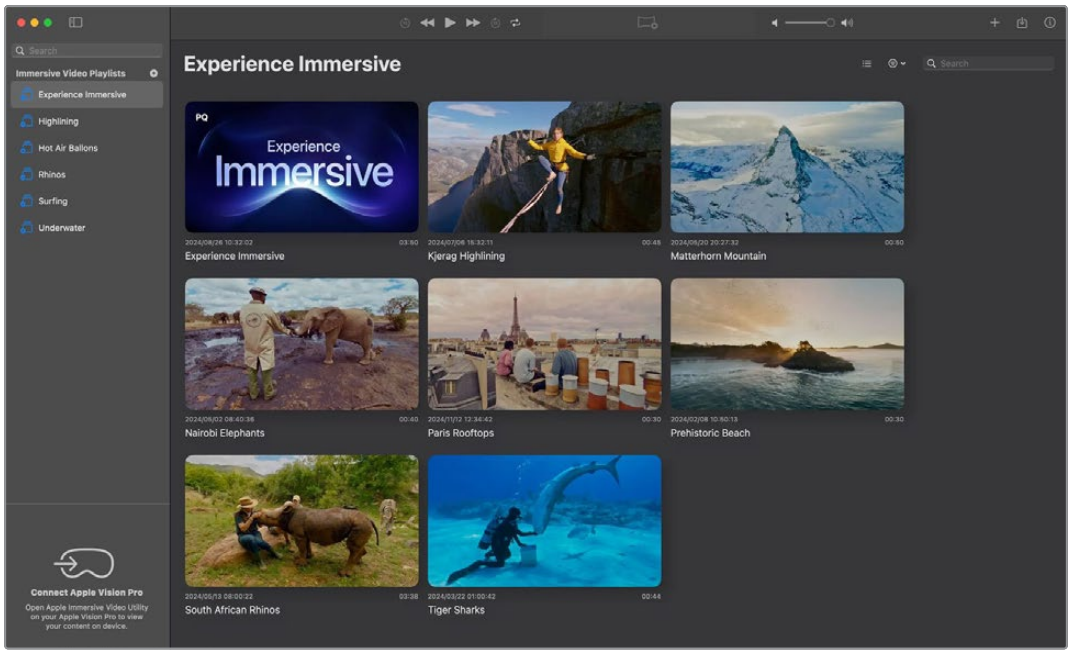
流式传输到Apple Vision Pro

您可以从DaVinci Resolve的剪辑、调色和交付页面流式传输到Apple Vision Pro，前往“工作区”菜单，选择“Stream to visionOS”，然后选择您的AVP设备。推荐使用高速无线网络（最低Wi–Fi 5，首选Wi–Fi 6）获得流畅的流式传输。



关于Apple Immersive Video实用程序

Apple Immersive Video实用程序是macOS和visionOS专用的App，其设计旨在优化Apple沉浸视频内容的创作、管理和播放。该实用程序专为专业电影制作人和沉浸式媒体创作者打造，让用户可以导入、管理、打包和检查带180度视野和空间音频的8K立体视频文件。它支持创建AIVU（Apple Immersive Video Universal）文件，用户可在后期制作流程中进行元数据检查和修改。用户可以创建播放列表，筛选和搜索素材库，甚至将内容直接流式传输到一台或多台Apple Vision Pro设备。



Developer Information

Camera Control REST API

If you are a software or hardware developer you can build custom applications or leverage ready to use tools such as REST client or Postman to seamlessly control and interact with your compatible Blackmagic camera using Camera Control REST API. This API enables you to perform a wide range of operations, such as starting or stopping recordings, accessing disk information and much more. Whether you're developing a custom application tailored to your specific needs or utilizing existing tools, this API empowers you to unlock the full potential of your Blackmagic camera with ease. We look forward to seeing what you come up with!

NOTE It's important to mention that controlling Blackmagic cameras via REST API relies on the web manager being enabled on each compatible Blackmagic camera. Enable the web media manager in the Blackmagic Camera Setup 'network access' settings for each camera you are controlling.

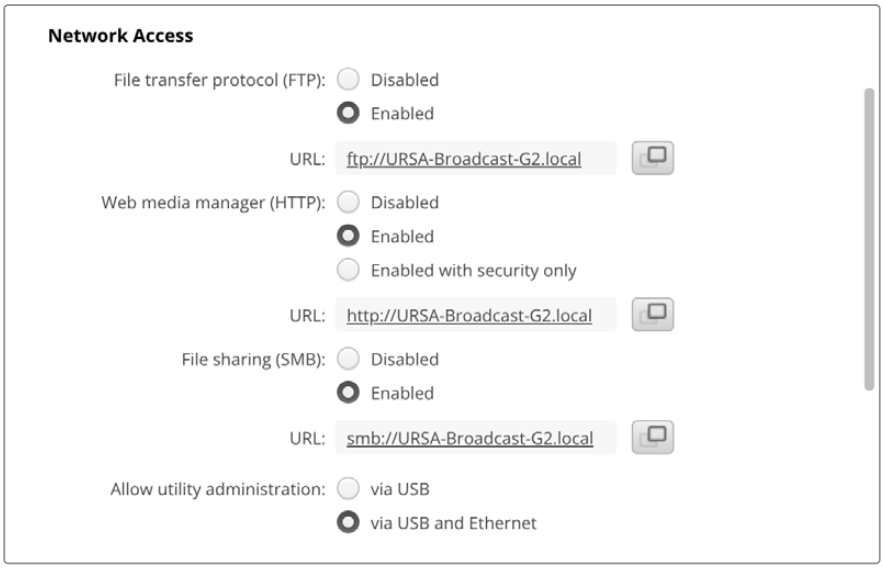
The following Blackmagic cameras are compatible with Camera Control REST API:

- Blackmagic URSA Cine 12K LF
- Blackmagic URSA Cine 17K 65
- Blackmagic URSA Cine Immersive
- Blackmagic PYXIS 6K
- Blackmagic Cinema Camera 6K
- Blackmagic URSA Broadcast G2
- Blackmagic Micro Studio Camera 4K G2
- Blackmagic Studio Camera 4K Plus
- Blackmagic Studio Camera 4K Pro
- Blackmagic Studio Camera 6K Pro
- Blackmagic Studio Camera 4K Plus G2
- Blackmagic Studio Camera 4K Pro G2

Sending API Commands

To send an API command to your camera from a third party application such as Postman, add /control/api/v1/ to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/api/v1/

You can find the Web media manager URL and IP address information in Blackmagic Camera Setup.



The Web media manager URL in Blackmagic Camera Setup

Downloading API's from your Camera

You can download REST API YAML documentation from your camera by adding /control/documentation.html to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/documentation.html

NOTE It's worth noting that changing the camera name in Blackmagic Camera Setup will also change the camera's Web media manager URL.

Livestream Control API

API for controlling Livestreams on Blackmagic Design products.

GET /livestreams/0

Get the livestream's current status.

Response

200 - Livestream's current status.

The response is JSON.

Name	Type	Description
status (required)	string	Possible values are: Idle, Connecting, Streaming, Flushing, Interrupted.
bitrate (required)	integer	Current bitrate (bps).
effectiveVideoFormat (required)	string	Effective video format for the livestream, serialised as a string.
duration	integer	Current stream duration in seconds. Absent if livestream is idle.
cache	integer	Current stream cache usage percentage.

GET /livestreams/0/start

Determine if the livestream is active.

Response

200 - Livestream active status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is active.

PUT /livestreams/0/start

Start the livestream.

Response

204 - Livestream started.

GET /livestreams/0/stop

Determine if the livestream is inactive.

Response

200 - Livestream inactive status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is inactive.

PUT /livestreams/0/stop

Stop the livestream.

Response

204 - Livestream stopped.

GET /livestreams/0/activePlatform

Get the currently selected platform configuration for the livestream.

Response

200 - Livestream active platform configuration.

The response is JSON.

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

PUT /livestreams/0/activePlatform

Set the currently selected platform configuration for the livestream.

Parameters

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

Response

204 - Livestream active platform configuration updated.

400 - Bad Request

GET /livestreams/platforms

Get the list of available platforms.

Response

200 - List of available platforms.

The response is JSON.

Name	Type	Description
	array	List of available platforms names.
[i]	string	Platform name.

GET /livestreams/platforms/{platformName}

Get the service configuration for a platform.

Parameters

Name	Type	Description
{platformName} (required)	string	Name of the platform.

Response

200 - Service configuration for specified platform.

The response is JSON.

Livestream platform service configuration.

Name	Type	Description
platform (required)	string	Corresponding platform name.
key	string	Default stream key.
servers (required)	array	List of server configurations.
servers[i]	object	Server configuration.
servers[i].server (required)	string	Server name.
servers[i].url (required)	string	Livestream destination.
servers[i].srtExtensions	array	Miscellaneous tags used for SRT livestreams.
servers[i].srtExtensions[i]	object	Dictionary object mapping SRT tag strings to values.
servers[i].srtExtensions[i][{key}]	string	SRT tag value.
servers[i].group	string	Logical grouping of the server.
profiles (required)	array	List of profile configurations.
profiles[i]	object	Quality configuration.
profiles[i].profile (required)	string	Quality level name.
profiles[i].configs (required)	array	List of video format configurations.
profiles[i].configs[i]	object	Video format configuration for profiles.
profiles[i].configs[i].resolution (required)	string	Video format serialised as a string.
profiles[i].configs[i].fps (required)	string	Frames per second.
profiles[i].configs[i].bitrate (required)	integer	Pixel bitrate (bps).
profiles[i].configs[i].audioBitrate	integer	Audio bitrate (bps).
profiles[i].configs[i].keyFrameInterval	integer	How often a key frame is sent, in seconds.
profiles[i].configs[i].videoCodecs	array	Supported video encoding algorithm/s.

Name	Type	Description
profiles[i].configs[i].videoCodecs[i]	string	Video encoding algorithm. Possible values are: H264, H265.
profiles[i].lowLatency (required)	boolean	If true, fewer frames will be buffered in the livestream.
defaultProfile	string	Quality level name.
credentials	object	Credientials used for RTMP streams.
credentials.username (required)	string	The username part of the creditials. Only used for RTMP streams.
credentials.password (required)	string	Used for RTMP streams, also used as Passphrase for SRT streams.
customizableUrlEnabled	boolean	True when the server URL is customizable.

400 - Bad Request

GET /livestreams/customPlatforms

Get a list of custom platform files.

Response

200 - List of custom platform files.

The response is JSON.

Name	Type	Description
	array	List of custom platform file names.
[i]	string	Custom platform file name.

DELETE /livestreams/customPlatforms

Remove all custom configuration files.

Response

204 - All custom configuration files removed.

GET /livestreams/customPlatforms/{filename}

Get a custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to get.

Response

200 - Custom platform file.

The response is XML.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

404 - Not Found

PUT /livestreams/customPlatforms/{filename}

Update a custom platform file if it exists, if not, create a new file with the given file name.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to update/create.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

Response

204 - Custom platform file created or updated.

400 - Bad Request

DELETE /livestreams/customPlatforms/{filename}

Remove the given custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to be removed.

Response

204 - Custom platform file removed.

404 - Not Found

Clips Control API

API for listing clips on disk.

GET /clips

Get the list of clips on the active disk.

Response

200 - List of clips on the active disk.

The response is JSON.

List of media clips.

Name	Type	Description
clips (required)	array	
clips[i]	object	Media clip.
clips[i].clipUniqueId (required)	integer	Unique ID used to identify this clip.
clips[i].filePath	string	Path to the file relative to the root of a mount.
clips[i].fileSize	integer	Size of file on disk in bytes.
clips[i].codecFormat	object	
clips[i].codecFormat.codec	string	Currently selected codec.

Name	Type	Description
clips[i].codecFormat.container	string	Multimedia container format.
clips[i].videoFormat	object	Video format configuration.
clips[i].videoFormat.name (required)	string	Video format serialised as a string.
clips[i].videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
clips[i].videoFormat.height	number	Height dimension of video format.
clips[i].videoFormat.width	number	Width dimension of video format.
clips[i].videoFormat.interlaced	boolean	Is the display format interlaced?
clips[i].startTimecode	string	Start timecode of the clip serialised as string.
clips[i].durationTimecode	string	Duration of the clip in timecode format serialised as string.
clips[i].frameCount	integer	Number of frames in clip; duration of the clip in frames.

404 - There is no active disk.

Media Pool Control API

API to manage media pool and handle uploads and project data.

GET /cloud/projects

List all projects within the media pool.

Response

200 - Successfully retrieved the list of all projects.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].libraryID	string	
[i].name	string	
[i].private	boolean	
[i].shared	boolean	
[i].clips	array	List of clips associated with the project.
[i].clips[i]	string	
[i].status	object	
[i].status.numClipsRequested	integer	
[i].status.numClipsComplete	integer	
[i].status.uploadPercent	integer	Percentage of upload completion.
[i].status.numClipsPaused	integer	
[i].status.outOfSpace	boolean	
[i].status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
[i].status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/active

Retrieve data of the actively uploading project.

Response

200 - Successfully retrieved the active project's data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/{projectID}

Retrieve specific project data by project ID.

Parameters

Name	Type	Description
{projectID} (required)	integer	Unique identifier of the project.

Response

200 - Successfully retrieved the project data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	

Name	Type	Description
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

404 - Project not found.

GET /cloud/clips

List all clips within the media pool.

Response

200 - Successfully retrieved the list of all clips.

The response is JSON.

Name	Type	Description
	array	
[i]	string	

GET /cloud/clips/activeUploading

Retrieve data of actively uploading clips.

Response

200 - Successfully retrieved the list of actively uploading clips.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].path	string	
[i].projectID	integer	
[i].status	object	
[i].status.projectID	integer	
[i].status.outOfSpace	boolean	
[i].status.proxyExtension	string	
[i].status.growingFile	boolean	
[i].status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.originalClipTotalSize	integer	
[i].status.proxyClipTotalSize	integer	
[i].status.originalClipCompletedSize	integer	
[i].status.proxyClipCompletedSize	integer	
[i].status.secsRemaining	integer	

GET /cloud/clips/{deviceName}/{path}

Retrieve specific clip data by device and path.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored.
{path} (required)	string	Path to the clip.

Response

200 - Successfully retrieved the clip data.

The response is JSON.

Name	Type	Description
path	string	
projectID	integer	
status	object	
status.projectID	integer	
status.outOfSpace	boolean	
status.proxyExtension	string	
status.growingFile	boolean	
status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.originalClipTotalSize	integer	
status.proxyClipTotalSize	integer	
status.originalClipCompletedSize	integer	
status.proxyClipCompletedSize	integer	
status.secsRemaining	integer	

404 - Clip not found.

Monitoring Control API

API for monitoring and controlling display settings in video equipment.

GET /monitoring/display

Retrieve a list of all display names.

Response

200 - Returns a list of display names.

The response is JSON.

Name	Type	Description
displays	array	List of display names available.
displays[i]	string	

GET /monitoring/{displayName}/cleanFeed

Get the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/cleanFeed

Set the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Clean feed enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/displayLUT

Get the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

400 - Invalid display name.

404 - Display name not found.

PUT /monitoring/{displayName}/displayLUT

Set the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Display LUT enabled/disabled successfully.

400 - Invalid input.

422 - Unprocessable Entity - Unable to process the contained instructions.

GET /monitoring/{displayName}/zebra

Get the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/zebra

Set the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Zebra enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/focusAssist

Get the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/focusAssist

Set the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/focusAssist

Get the focus assist settings.

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

404 - Display name not found.

PUT /monitoring/focusAssist

Set the focus assist settings.

Parameters

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/frameGuide

Get the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame guide enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGuide

Set the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame guide state updated successfully.

422 - Unable to update the frame guide state.

GET /monitoring/frameGuideRatio

Get the current frame guide ratio.

Response

200 - Returns the current frame guide ratio.

The response is JSON.

Name	Type	Description
ratio	string	The frame guide ratio.

PUT /monitoring/frameGuideRatio

Set the frame guide ratio.

Parameters

Name	Type	Description
ratio	string	The frame guide ratio.

Response

204 - Frame guide ratio updated successfully.

422 - Unable to update the frame guide ratio.

GET /monitoring/frameGuideRatio/presets

Get the presets for frame guide ratios.

Response

200 - Returns a list of preset frame guide ratios.

The response is JSON.

Name	Type	Description
presets	array	
presets[i]	string	A frame guide ratio.

GET /monitoring/{displayName}/frameGrids

Get the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame grids enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGrids

Set the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame grids state updated successfully.

422 - Unable to update the frame grids state.

GET /monitoring/frameGrids

Get the global frame grids settings.

Response

200 - Returns the current frame grids settings.

The response is JSON.

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

PUT /monitoring/frameGrids

Set the global frame grids settings.

Parameters

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

Response

204 - Frame grids settings updated successfully.

400 - Invalid input, check the number of frame grids or values.

422 - Unable to update the frame grids settings.

GET /monitoring/{displayName}/safeArea

Get the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the safe area enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/safeArea

Set the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Safe area state updated successfully.

422 - Unable to update the safe area state.

GET /monitoring/safeAreaPercent

Get the current safe area percentage.

Response

200 - Returns the current safe area percentage.

The response is JSON.

Name	Type	Description
percent	integer	Safe area coverage percentage.

PUT /monitoring/safeAreaPercent

Set the safe area percentage.

Parameters

Name	Type	Description
percent	integer	Safe area coverage percentage to set.

Response

204 - Safe area percentage updated successfully.

400 - Invalid percentage value.

422 - Unable to update the safe area percentage.

GET /monitoring/{displayName}/falseColor

Get the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the false color enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/falseColor

Set the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - False color state updated successfully.

422 - Unable to update the false color state.

Event Control API

API For working with built-in websocket.

GET /event/list

Get the list of events that can be subscribed to using the websocket API.

Response

200 - Websocket API events list.

The response is JSON.

Name	Type	Description
events	array	List of events that can be subscribed to using the websocket API.
events[i]	string	

System Control API

API for controlling the System Modes on Blackmagic Design products.

GET /system

Get device system information.

Response

200 - System summary.

The response is JSON.

The properties will be populated only with the values that are supported/implemented by the device in use.

Name	Type	Description
codecFormat	object	Codec format configuration.
codecFormat.codec	string	Codec serialised as string.
codecFormat.container	string	Multimedia container format.
videoFormat	object	Video format configuration.
videoFormat.name (required)	string	Video format serialised as a string.
videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
videoFormat.height	number	Height dimension of video format.
videoFormat.width	number	Width dimension of video format.
videoFormat.interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/product

Get device product information.

Response

200 - Device product information.

The response is JSON.

Product information.

Name	Type	Description
deviceName	string	Name of device as displayed in Setup.
productName	string	Device's product name.
softwareVersion	string	Software version running on device.

501 - This functionality is not implemented for the device in use.

GET /system/supportedCodecFormats

Get the list of supported codecs.

Response

200 - List of supported codec formats.

The response is JSON.

Name	Type	Description
codecs	array	
codecs[i]	object	Codec format configuration.
codecs[i].codec	string	Codec serialised as string.
codecs[i].container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

GET /system/codecFormat

Get the currently selected codec.

Response

200 - Current codec format.

The response is JSON.

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

PUT /system/codecFormat

Update the system codec.

Parameters

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

Response

204 - The codec updated successfully.

400 - The specified codec format is unsupported.

501 - This functionality is not implemented for the device in use.

GET /system/videoFormat

Get the currently selected video format.

Response

200 - Current system video format.

The response is JSON.

Video format configuration.

Name	Type	Description
name (required)	string	Video format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height	number	Height dimension of video format.
width	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

PUT /system/videoFormat

Set the system video format.

Parameters

This parameter can be one of the following types:

Name	Type	Description
name (required)	string	Video format serialised as a string.

Name	Type	Description
frameRate (required)	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height (required)	number	Height dimension of video format.
width (required)	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

Response

204 - The video format updated successfully.

400 - Invalid request.

409 - Operation unsupported in the current state.

501 - This functionality is not implemented for the device in use.

GET /system/supportedVideoFormats

Get the list of supported video formats for the current system state.

Response

200 - List of supported video formats.

The response is JSON.

List of supported video formats.

Name	Type	Description
formats	array	List of video formats.
formats[i]	object	Video format configuration.
formats[i].name (required)	string	Video format serialised as a string.
formats[i].frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
formats[i].height	number	Height dimension of video format.
formats[i].width	number	Width dimension of video format.
formats[i].interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/supportedFormats

Get supported formats.

Response

200 - List of supported formats.

The response is JSON.

Name	Type	Description
supportedFormats	array	
supportedFormats[i]	object	
supportedFormats[i].codecs	array	
supportedFormats[i].codecs[i]	string	
supportedFormats[i].frameRates	array	
supportedFormats[i].frameRates[i]	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
supportedFormats[i]. maxOffSpeedFrameRate	number	
supportedFormats[i]. minOffSpeedFrameRate	number	
supportedFormats[i]. recordResolution	object	
supportedFormats[i]. recordResolution.height	number	Height of the resolution.
supportedFormats[i]. recordResolution.width	number	Width of the resolution.
supportedFormats[i]. sensorResolution	object	
supportedFormats[i]. sensorResolution.height	number	Height of the resolution.
supportedFormats[i]. sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

GET /system/format

Get current format.

Response

200 - Current format.

The response is JSON.

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

PUT /system/format

Set the format.

Parameters

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

Response

204 - System format updated.

501 - This functionality is not implemented for the device in use.

Transport Control API

API for controlling Transport on Blackmagic Design products.

GET /transports/0

Get device's basic transport status.

Response

200 - Transport status.

The response is JSON.

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

PUT /transports/0

Set device's basic transport status.

Parameters

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, Output.

Response

204 - Transport mode was set.

400 - Failed to set transport mode.

GET /transports/0/stop

Determine if transport is stopped.

Response

200 - Transport stop response.

The response is JSON.

Name	Type	Description
	boolean	True when transport mode is InputPreview or when in Output mode and speed is 0.

PUT /transports/0/stop

Stop transport. Deprecated, use POST /transports/0/stop instead.

Response

204 - Transport stopped.

POST /transports/0/stop

Stop transport.

Response

204 - Transport stopped.

GET /transports/0/play

Determine if transport is playing.

Response

200 - Transport play response.

The response is JSON.

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

PUT /transports/0/play

Start playing on transport. Deprecated, use POST /transports/0/play instead.

Response

204 - Transport playing.

400 - Failed to set transport to play.

POST /transports/0/play

Start playing on transport.

Response

204 - Transport playing.

400 - Failed to set transport to play.

GET /transports/0/playback

Get playback state.

Response

200 - Transport playback state.

The response is JSON.

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

PUT /transports/0/playback

Set playback state.

Parameters

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

Response

204 - Updated transport playback state.

400 - Failed to set transport playback state.

GET /transports/0/record

Get record state.

Response

200 - Recording state.

The response is JSON.

Name	Type	Description
recording	boolean	If true, transport is in InputRecord mode.

PUT /transports/0/record

Set record state. Deprecated, use POST /transports/0/record instead.

Parameters

Name	Type	Description
recording (required)	boolean	If true, starts a recording, otherwise stops.
clipName	string	Optional, sets the requested clip name to record to, when “recording” attribute is set to true.

Response

204 - Recording state updated.

400 - Failed to update recording state.

POST /transports/0/record

Start recording.

Parameters

Name	Type	Description
clipName	string	Optional, provides a specific name of clip to record to.

Response

204 - Recording started.

400 - Failed to start recording.

GET /transports/0/clipIndex

Get the clip index of the currently playing clip on the timeline.

Response

200 - Clip index response.

The response is JSON.

Name	Type	Description
clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

GET /transports/0/timecode

Get device timecode.

Response

200 - Timecode response.

The response is JSON.

Name	Type	Description
display	string	The display timecode serialised as a string.
timeline	string	The timeline timecode serialised as a string.

GET /transports/0/timecode/source

Get timecode source selected on device.

Response

200 - Timecode source response.

The response is JSON.

Name	Type	Description
timecode	string	Possible values are: Timeline, Clip.

Timeline Control API

API for controlling playback timeline.

GET /timelines/0

Get the playback timeline.

Response

200 - Playback timeline.

The response is JSON.

Name	Type	Description
clips	array	
clips[i]	object	Timeline clip.
clips[i].clipUniqueId (required)	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).
clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

404 - No timeline / disk available.

DELETE /timelines/0

Clear the current playback timeline. Deprecated, prefer to use POST /timelines/0/clear

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

POST /timelines/0

Add a clip to the timeline.

Parameters

This parameter can be one of the following types:

Add multiple media clips to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips[i].clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips[i].clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips[i].frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add multiple media clips to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add a single clip to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips.clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips.clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips.frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add a single clip to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the timeline as specified.

501 - The operation is not supported on this device.

POST /timelines/0/add

Add a clip to the end of the timeline. Deprecated, use POST /timelines/0 to add clips within the timeline.

Parameters

This parameter can be one of the following types:

Add one clip to the end of the timeline.

Name	Type	Description
clips	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add many clips to the end of the timeline.

Name	Type	Description
clips	array	List of clipUniqueIds of clips to add to end of timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the end of the timeline.

501 - The operation is not supported on this device.

POST /timelines/0/clear

Clear the playback timeline.

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

DELETE /timelines/0/clips/{timelineClipIndex}

Remove the specified clip from the timeline.

Parameters

Name	Type	Description
{timelineClipIndex} (required)	integer	The (0-based) timeline clip index of the clip to remove from the timeline.

Response

204 - The specified clip was removed from the timeline.

501 - The operation is not supported on this device.

Media Control API

API for controlling media devices in Blackmagic Design products.

GET /media/workingset

Get the list of media devices currently in the working set.

Response

200 - The list of media devices in the working set.

The response is JSON.

Name	Type	Description
size	integer	The fixed size of this device's working set.
workingset	array	The device's working set.
workingset[i]	object | null	Device within the working set. null if no device is present within the given working set slot.
workingset[i].volume	string	Volume name.
workingset[i].deviceName	string	Internal device name of this media device.
workingset[i].remainingRecordTime	integer	Remaining record time using current codec and video format in seconds.
workingset[i].totalSpace	integer	Total space on media device in bytes.
workingset[i].remainingSpace	integer	Remaining space on media device in bytes.
workingset[i].clipCount	integer	Number of clips currently on the device.

GET /media/active

Get the currently active media device.

Response

200 - The current active media device.

The response is JSON.

The active media device, or null if there is no active media.

Name	Type	Description
workingsetIndex	integer	Working set index of the active media device.
deviceName	string	Device name of media device.

204 - No media is currently active.

PUT /media/active

Set the currently active media device.

Parameters

Name	Type	Description
workingsetIndex	integer	Working set index of the media to make active.

Response

204 - The active media device was set successfully.

400 - Setting the currently active media device is not possible in the current state.

GET /media/devices/doformatSupportedFilesystems

Get the list of filesystems available to format a media device.

Response

200 - The list of filesystems permitted for formatting.

The response is JSON.

Name	Type	Description
	array	List of filesystems permitted for formatting media.
[i]	string	Filesystem serialised as string.

GET /media/devices/{deviceName}

Get information about a requested device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Information about the requested device.

The response is JSON.

Media device state.

Name	Type	Description
state	string	The current state of the media device. Possible values are: None, Scanning, Mounted, Uninitialised, Formatting, RaidComponent.

400 - Invalid device name.

404 - Device not found.

GET /media/devices/{deviceName}/doformat

Get a format key, used to format the device with a PUT request.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Format prepared.

The response is JSON.

Name	Type	Description
deviceName	string	Device name of media device to format.
key	string	The key required to format this device, provide to PUT /media/devices/{deviceName}/doformat to perform format of media device.

400 - Cannot format the device.

404 - Device not found.

PUT /media/devices/{deviceName}/doformat

Perform a format of the specified media device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by "deviceName" member of GET /media/workingset or GET /media/active.

Name	Type	Description
key	string	The key used to format this device, retrieved from prepare format media request GET /media/devices/{deviceName}/doformat. Format key provided cannot be reused after successful format.
filesystem	string	Filesystem to format to. Supported filesystems can be retrieved with GET /media/devices/doFormatSupportedFilesystems.
volume	string	Volume name to set for the disk after format.

Response

204 - Format successful.

400 - Cannot format the device, invalid filesystem or key.

404 - Device not found.

Slate Control API

API to manage digital slate data.

GET /slates/nextClip

Retrieve the digital slate for the next clip.

Response

200 - Returns the slate data for the next clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	

Name	Type	Description
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Slate data is not available.

PUT /slates/nextClip

Update the slate data for the next clip.

Parameters

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

POST /slates/nextClip/resetProjectData

Reset the project data for the next clip's slate.

Response

200 - Project data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetProjectData

Reset the project data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Project data reset successfully.

POST /slates/nextClip/resetLensData

Reset the lens data for the next clip's slate.

Response

200 - Lens data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetLensData

Reset the lens data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Lens data reset successfully.

GET /slates/clips/{deviceName}/{path}

Retrieve slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Returns the slate data for the specified clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

404 - Clip not found.

PUT /slates/clips/{deviceName}/{path}

Update the slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.

Name	Type	Description
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

Preset Control API

API For controlling the presets on Blackmagic Design products

GET /presets

Get the list of the presets on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presets	array	List of the presets on the camera (.cset files)
presets[i]	string	

POST /presets

Send a preset file to the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presetAdded	string	Name of the preset uploaded (without .cset extension)

400 - Bad request - missing Content-Disposition header or filename

GET /presets/active

Get the currently active preset on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

PUT /presets/active

Set the active preset on the camera

Parameters

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

Response

204 - No Content

404 - Preset file not found

PARAMETERS /presets/{presetName}

GET /presets/{presetName}

Download the preset file

Response

200 - OK

The response is a binary file.

404 - File does not exist

PUT /presets/{presetName}

Save current camera state as a preset

Response

204 - No Content

DELETE /presets/{presetName}

Delete a preset from the camera

Response

204 - No Content

404 - Preset file not found

Audio Control API

API For controlling audio on Blackmagic Design Cameras

GET /audio/channels

Get the total number of audio channels available

Response

200 - Returns the total number of channels

The response is JSON.

Name	Type	Description
channels	integer	Total number of audio channels available

GET /audio/supportedInputs

Get the list of supported audio inputs

Response

200 - List of all supported audio inputs

The response is JSON.

Name	Type	Description
	array	
[i]	string	A supported audio input

GET /audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently selected input

The response is JSON.

Name	Type	Description
input	string	Audio input source and type

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/input

Set the audio input for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
input	string	Audio input source and type

Response

204 - No Content

400 - Invalid audio input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/input/description

Get the description of the current input of the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Description of the current input of the selected channel

The response is JSON.

Name	Type	Description
description	object	
description.gainRange	object	
description.gainRange.Min	number	The minimum gain value in dB
description.gainRange.Max	number	The maximum gain value in dB
description.capabilities	object	
description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter
description.capabilities.Padding	object	
description.capabilities.Padding. available	boolean	Input supports setting of padding
description.capabilities.Padding. forced	boolean	Padding is forced to be set for the input
description.capabilities.Padding. value	number	Value of the padding in dB

404 - Channel does not exist

GET /audio/channel/{channelIndex}/supportedInputs

Get the list of supported inputs and their availability to switch to for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its supported inputs are being queried. (Channels index from 0)

Response

200 - The list of supported inputs

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

404 - Channel does not exist

GET /audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set level for the selected channel

The response is JSON.

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/level

Set the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

Response

204 - No Content

400 - Invalid input or value out of range

404 - Channel does not exist

GET /audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set phantom power for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Phantom power enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/phantomPower

Set the audio phantom power for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Phantom power enabled state

Response

204 - No Content

400 - Phantom power is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set padding for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Padding enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/padding

Set the audio input padding for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Padding enabled state

Response

204 - No Content

400 - Padding is not supported or is forced for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set low cut filter for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Low cut filter enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/lowCutFilter

Set the audio input low cut filter for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Low cut filter enabled state

Response

204 - No Content

400 - Low cut filter is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set availability for the selected channel

The response is JSON.

Name	Type	Description
available	boolean	Whether the input is currently available

404 - Channel does not exist

Lens Control API

API For controlling the lens on Blackmagic Design products

GET /lens/iris

Get lens' aperture

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number

PUT /lens/iris

Set lens' aperture

Parameters

Name	Type	Description
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number
adjustmentStep	integer	Signed value for relative aperture adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens iris is not controllable or is controlled by auto exposure

GET /lens/zoom

Get lens' zoom

Response

200 - OK

The response is JSON.

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value

PUT /lens/zoom

Set lens' zoom

Parameters

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value
adjustmentFocalLength	integer	Signed value for relative focal length adjustment
adjustmentNormalised	number	Signed normalized value for relative zoom adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens zoom is not controllable

GET /lens/focus

Get lens' focus

Response

200 - OK

The response is JSON.

Name	Type	Description
normalised	number	Normalised value

PUT /lens/focus

Set lens' focus

Parameters

Name	Type	Description
normalised	number	Normalised value
focusDistance	integer	Focus distance value

Response

204 - No Content

400 - Bad Request if out of range value is provided

PUT /lens/focus/doAutoFocus

Perform auto focus

Parameters

Name	Type	Description
position	object	
position.x	number	Normalized x coordinate for autofocus ROI
position.y	number	Normalized y coordinate for autofocus ROI

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens focus is not controllable

GET /lens/opticalImageStabilization

Get optical image stabilization status

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Whether optical image stabilization is enabled

501 - Not Implemented if optical image stabilization is not supported on this product

PUT /lens/opticalImageStabilization

Enable or disable optical image stabilization

Parameters

Name	Type	Description
enabled	boolean	Enable or disable optical image stabilization

Response

204 - No Content

501 - Not Implemented if optical image stabilization is not supported on this product

GET /lens/iris/description

Get detailed description of lens' iris capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the iris can be controlled
apertureStop	object	
apertureStop.min	number	Minimum aperture stop
apertureStop.max	number	Maximum aperture stop

GET /lens/zoom/description

Get detailed description of lens' zoom capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the zoom can be controlled
focalLength	object	
focalLength.adjustable	boolean	If focal length is adjustable
focalLength.min	integer	Minimum focal length
focalLength.max	integer	Maximum focal length

GET /lens/focus/description

Get detailed description of lens' focus capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the focus can be controlled
focusDistance	object	
focusDistance.adjustable	boolean	If focus distance is adjustable
focusDistance.min	number	Minimum focus distance
focusDistance.max	number	Maximum focus distance

Video Control API

API For controlling the video on Blackmagic Design products

GET /video/iso

Get current ISO

Response

200 - OK

The response is JSON.

Name	Type	Description
iso	integer	Current ISO value

PUT /video/iso

Set current ISO

Parameters

Name	Type	Description
iso	integer	ISO value to set

Response

204 - No Content

403 - ISO cannot be changed in the current state

GET /video/supportedISOs

Get the list of supported ISO settings

Response

200 - List of supported ISO values

The response is JSON.

Name	Type	Description
supportedISOs	array	Array of supported ISO values
supportedISOs[i]	integer	

GET /video/gain

Get current gain value in decibels

Response

200 - OK

The response is JSON.

Name	Type	Description
gain	integer	Current gain value in decibels

PUT /video/gain

Set current gain value

Parameters

Name	Type	Description
gain	integer	Gain value in decibels to set

Response

204 - No Content

403 - Gain cannot be changed in the current state

GET /video/supportedGains

Get the list of supported gain settings in decibels

Response

200 - List of supported gain values in decibels

The response is JSON.

Name	Type	Description
supportedGains	array	Array of supported gain values in decibels
supportedGains[i]	integer	

GET /video/whiteBalance

Get current white balance

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	integer	Current white balance

PUT /video/whiteBalance

Set current white balance

Parameters

Name	Type	Description
whiteBalance	integer	White balance to set

Response

204 - No Content

400 - Invalid white balance temperature

GET /video/whiteBalance/description

Get white balance range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	object	
whiteBalance.min	integer	Minimum color temperature
whiteBalance.max	integer	Maximum color temperature

PUT /video/whiteBalance/doAuto

Set current white balance automatically

Response

204 - No Content

GET /video/whiteBalanceTint

Get white balance tint

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	integer	Current white balance tint

PUT /video/whiteBalanceTint

Set white balance tint

Parameters

Name	Type	Description
whiteBalanceTint	integer	White balance tint to set

Response

204 - No Content

400 - Invalid white balance tint

GET /video/whiteBalanceTint/description

Get white balance tint range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	object	
whiteBalanceTint.min	integer	Minimum white balance tint
whiteBalanceTint.max	integer	Maximum white balance tint

GET /video/ndFilter

Get ND filter stop

Response

200 - OK

The response is JSON.

Name	Type	Description
stop	number	Current filter power (fStop)

501 - Not implemented for this device

PUT /video/ndFilter

Set ND filter stop

Parameters

Name	Type	Description
stop	number	Filter power (fStop) to set

Response

204 - No Content

400 - Invalid ND filter stop

501 - Not implemented for this device

GET /video/supportedNDFilters

Get the list of available ND filter stops

Response

200 - List of available ND filter stops

The response is JSON.

Name	Type	Description
supportedStops	array	Array of available ND filter stops
supportedStops[i]	number	

501 - Not implemented for this device

GET /video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

Response

200 - List of supported display modes

The response is JSON.

Name	Type	Description
supportedDisplayModes	array	Array of supported display modes
supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

GET /video/ndFilter/displayMode

Get ND filter display mode on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

PUT /video/ndFilter/displayMode

Set ND filter display mode on the camera

Parameters

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

Response

204 - No Content

400 - Invalid display mode for ND filter

501 - Not implemented for this device

GET /video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

Response

200 - Indicates if ND filter is selectable

The response is JSON.

Name	Type	Description
selectable	boolean	True if ND filter adjustments are selectable via a slider

501 - Not implemented for this device

GET /video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

PUT /video/shutter

Set current shutter

Parameters

Name	Type	Description
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

Response

204 - No Content

GET /video/shutter/measurement

Get the current shutter measurement mode

Response

200 - OK

The response is JSON.

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

PUT /video/shutter/measurement

Set the shutter measurement mode

Parameters

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

Response

204 - No Content

400 - Invalid shutter measurement

GET /video/supportedShutters

Get supported shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of supported shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of supported shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of flicker-free shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of flicker-free shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/autoExposure

Get current auto exposure mode

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

PUT /video/autoExposure

Set auto exposure

Parameters

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

Response

204 - No Content

400 - Failed to set auto exposure mode

GET /video/detailSharpening

Get the current state of detail sharpening

Response

200 - Current detail sharpening state

The response is JSON.

Name	Type	Description
enabled	boolean	Whether detail sharpening is enabled

501 - Not implemented for this device

PUT /video/detailSharpening

Enable or disable detail sharpening

Parameters

Name	Type	Description
enabled	boolean	Enable or disable detail sharpening

Response

204 - Detail sharpening state updated

501 - Not implemented for this device

GET /video/detailSharpeningLevel

Get the current detail sharpening level

Response

200 - Current detail sharpening level

The response is JSON.

Name	Type	Description
level	string	Current detail sharpening level Possible values are: Low, Medium, High.

501 - Not implemented for this device

PUT /video/detailSharpeningLevel

Set the detail sharpening level

Parameters

Name	Type	Description
level	string	Desired level of detail sharpening Possible values are: Low, Medium, High.

Response

204 - Detail sharpening level updated

400 - Invalid detail sharpening level

501 - Not implemented for this device

Camera Control API

API For controlling the Camera specific features on Blackmagic Design products

GET /camera/colorBars

Get the status of color bars display

Response

200 - Returns the current status of color bars

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the color bars are currently enabled

PUT /camera/colorBars

Set the status of color bars display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the color bars

Response

204 - Color bars status updated successfully

GET /camera/programFeedDisplay

Get the status of program feed display

Response

200 - Returns the current status of program feed display

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the program feed display is currently enabled

PUT /camera/programFeedDisplay

Set the status of program feed display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the program feed display

Response

204 - Program feed display status updated successfully

GET /camera/tallyStatus

Get the tally status of the camera

Response

200 - Returns the current tally status of the camera

The response is JSON.

Name	Type	Description
status	string	Current tally status of the camera Possible values are: None, Preview, Program.

GET /camera/power

Get the power status of the camera

Response

200 - Returns the current power status

The response is JSON.

Name	Type	Description
source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
batteries	array	List of batteries currently connected to the camera
batteries[i]	object	
batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
batteries[i].chargeRemainingPercent	integer	Remaining battery charge percentage
batteries[i].statusFlags	array	List of battery status flags
batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

GET /camera/power/displayMode

Get the power display mode of the camera

Response

200 - Returns the current power display mode

The response is JSON.

Name	Type	Description
mode	string	Current power display mode Possible values are: Percentage, Voltage.

PUT /camera/power/displayMode

Set the power display mode of the camera

Parameters

Name	Type	Description
mode	string	Power display mode to set Possible values are: Percentage, Voltage.

Response

204 - Power display mode updated successfully

400 - Invalid power display mode

GET /camera/timingReferenceLock

Get the timing reference lock status

Response

200 - Returns the timing reference lock status

The response is JSON.

Name	Type	Description
locked	boolean	Indicates if timing reference is locked

Immersive Control API

API for controlling immersive camera settings on Blackmagic Design cameras

GET /immersive/display/{displayName}/eye

Get the current eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to query (from /monitoring/display endpoint)

Response

200 - OK

The response is JSON.

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

400 - Invalid display name format

404 - Display not found

422 - Failed to get eye view

PUT /immersive/display/{displayName}/eye

Set the eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to control (from /monitoring/display endpoint)

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

Response

204 - No Content

400 - Invalid input or display name format

404 - Display not found

422 - Failed to set eye view

Color Correction Control API

API For controlling the color correction on Blackmagic Design products based on DaVinci Resolve Color Corrector

GET /colorCorrection/lift

Get color correction lift

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

PUT /colorCorrection/lift

Set color correction lift

Parameters

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gamma

Get color correction gamma

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

PUT /colorCorrection/gamma

Set color correction gamma

Parameters

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gain

Get color correction gain

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

PUT /colorCorrection/gain

Set color correction gain

Parameters

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/offset

Get color correction offset

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

PUT /colorCorrection/offset

Set color correction offset

Parameters

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/contrast

Get color correction contrast

Response

200 - OK

The response is JSON.

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/contrast

Set color correction contrast

Parameters

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/color

Get color correction color properties

Response

200 - OK

The response is JSON.

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/color

Set color correction color properties

Parameters

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/lumaContribution

Get color correction luma contribution

Response

200 - OK

The response is JSON.

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

PUT /colorCorrection/lumaContribution

Set color correction luma contribution

Parameters

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

Response

204 - No Content

Notification websocket - 1.0.0

Service that notifies subscribers of device state changes.

messages

Subscribe (The messages from the server/device)

Websocket Opened Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: websocketOpened.
.type	string	Possible values are: event.

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*

Name	Type	Description
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Event Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: propertyValueChanged.

Name	Type	Description
.data.property	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.value	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.type	string	Possible values are: event.

Publish (The messages that user can send to the server/device)

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	

Name	Type	Description
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	

Name	Type	Description
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Device Properties

/media/workingset

The value JSON returned via the eventResponse when the `/media/workingset` property changes on the device:

Name	Type	Description
.size	integer	The fixed size of this device's working set.
.workingset	array	Array of devices within the working set. null if no device is present within the given working set slot.
.workingset[i]		

/media/active

The value JSON returned via the eventResponse when the `/media/active` property changes on the device:

Name	Type	Description
.workingsetIndex	integer	Working set index of the active media device.
.deviceName	string	Internal device name of this media device.

/system

The value JSON returned via the eventResponse when the `/system` property changes on the device:

Name	Type	Description
.codecFormat	object	Codec format configuration.
.codecFormat.codec	string	Codec format serialised as a string.
.codecFormat.container	string	Multimedia container format.
.videoFormat	object	Currently selected video format.
.videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.videoFormat.height	number	Height dimension of video format.
.videoFormat.width	number	Width dimension of video format.
.videoFormat.interlaced	boolean	Is the display format interlaced?.
.videoFormat.name	string	Video format serialised as a string.

/system/codecFormat

Codec format configuration.

The value JSON returned via the eventResponse when the `/system/codecFormat` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.container	string	Multimedia container format.

/system/videoFormat

Currently selected video format.

The value JSON returned via the eventResponse when the `/system/videoFormat` property changes on the device:

Name	Type	Description
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.height	number	Height dimension of video format.
.width	number	Width dimension of video format.
.interlaced	boolean	Is the display format interlaced?.
.name	string	Video format serialised as a string.

/system/format

The value JSON returned via the eventResponse when the `/system/format` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.maxOffSpeedFrameRate	number	
.minOffSpeedFrameRate	number	
.offSpeedEnabled	boolean	
.offSpeedFrameRate	number	
.recordResolution	object	
.recordResolution.height	number	Height of the resolution.
.recordResolution.width	number	Width of the resolution.
.sensorResolution	object	
.sensorResolution.height	number	Height of the resolution.
.sensorResolution.width	number	Width of the resolution.

/system/supportedFormats

The value JSON returned via the eventResponse when the `/system/supportedFormats` property changes on the device:

Name	Type	Description
.supportedFormats	array	
.supportedFormats[i]	object	
.supportedFormats[i].codecs	array	
.supportedFormats[i].codecs[i]	string	
.supportedFormats[i].frameRates	array	
.supportedFormats[i].frameRates[i]	string	Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.supportedFormats[i].maxOffSpeedFrameRate	number	
.supportedFormats[i].minOffSpeedFrameRate	number	
.supportedFormats[i].recordResolution	object	
.supportedFormats[i].recordResolution.height	number	Height of the resolution.
.supportedFormats[i].recordResolution.width	number	Width of the resolution.
.supportedFormats[i].sensorResolution	object	
.supportedFormats[i].sensorResolution.height	number	Height of the resolution.
.supportedFormats[i].sensorResolution.width	number	Width of the resolution.

/timelines/0

The value JSON returned via the eventResponse when the `/timelines/0` property changes on the device:

Name	Type	Description
.clips	array	
.clips[i]	object	Timeline clip.
.clips[i].clipUniqueId	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
.clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
.clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
.clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
.clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).

Name	Type	Description
.clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
.clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

/transports/0

The value JSON returned via the eventResponse when the `/transports/0` property changes on the device:

Name	Type	Description
.mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

/transports/0/stop

true when transport mode is InputPreview or when in Output mode and speed is 0.

The value JSON returned via the eventResponse when the `/transports/0/stop` property changes on the device:

Name	Type	Description
	boolean	true when transport mode is InputPreview or when in Output mode and speed is 0.

/transports/0/play

True when transport is in Output mode and speed is non-zero.

The value JSON returned via the eventResponse when the /transports/0/play property changes on the device:

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

/transports/0/playback

The value JSON returned via the eventResponse when the `/transports/0/playback` property changes on the device:

Name	Type	Description
.type	string	Possible values are: Play, Jog, Shuttle, Var.
.loop	boolean	When true playback loops from the end of the timeline to the beginning of the timeline.
.singleClip	boolean	When true playback loops from the end of the current clip to the beginning of the current clip.
.speed	number	Playback speed, 1.0 for normal forward playback
.position	integer	Playback position on the timeline in units of video frames.

/transports/0/record

The value JSON returned via the eventResponse when the `/transports/0/record` property changes on the device:

Name	Type	Description
.recording	boolean	Is transport in Input Record mode.

/transports/0/timecode

The value JSON returned via the eventResponse when the `/transports/0/timecode` property changes on the device:

Name	Type	Description
.display	string	The display timecode serialised as a string.
.timeline	string	The timeline timecode serialised as a string.

/transports/0/timecode/source

The value JSON returned via the eventResponse when the `/transports/0/timecode/source` property changes on the device:

Name	Type	Description
.timecode	string	Possible values are: Timeline, Clip.

/transports/0/clipIndex

The value JSON returned via the eventResponse when the `/transports/0/clipIndex` property changes on the device:

Name	Type	Description
.clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

/slates/nextClip

The value JSON returned via the eventResponse when the `/slates/nextClip` property changes on the device:

Name	Type	Description
.clip	object	
.clip.clipName	string	
.clip.reel	integer	
.clip.scene	string	
.clip.sceneLocation	string	Possible values are: Interior, Exterior.
.clip.sceneTime	string	Possible values are: Day, Night.
.clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
.clip.take	integer	
.clip.takeType	string	Possible values are: None, PU, VFX, SER.
.clip.goodTake	boolean	
.lens	object	
.lens.lensType	string	
.lens.iris	string	

Name	Type	Description
.lens.focalLength	string	
.lens.distance	string	
.lens.filter	string	
.project	object	
.project.projectName	string	
.project.director	string	
.project.camera	string	
.project.cameraOperator	string	

/monitoring/{displayName}/cleanFeed

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/cleanFeed` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/displayLUT

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/displayLUT` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/zebra

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/zebra` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/focusAssist` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGuide

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGuide` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGrids` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/safeArea

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/safeArea` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/falseColor

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/falseColor` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/focusAssist` property changes on the device:

Name	Type	Description
.mode	string	Possible values are: Peak, ColoredLines.
.color	string	Possible values are: Red, Green, Blue, White, Black.
.intensity	integer	

/monitoring/frameGuideRatio

The value JSON returned via the eventResponse when the `/monitoring/frameGuideRatio` property changes on the device:

Name	Type	Description
.ratio	string	

/monitoring/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/frameGrids` property changes on the device:

Name	Type	Description
.frameGrids	array	
.frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

/monitoring/safeAreaPercent

The value JSON returned via the eventResponse when the `/monitoring/safeAreaPercent` property changes on the device:

Name	Type	Description
.percent	integer	Safe area coverage percentage.

/audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input` property changes on the device:

Name	Type	Description
.input	string	Audio input source and type

/audio/channel/{channelIndex}/supportedInputs

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/supportedInputs` property changes on the device:

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

/audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/level` property changes on the device:

Name	Type	Description
.gain	number	Gain value in dB
.normalised	number	Normalised level value between 0.0 and 1.0

/audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/phantomPower` property changes on the device:

Name	Type	Description
.enabled	boolean	Phantom power enabled state

/audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/padding` property changes on the device:

Name	Type	Description
.enabled	boolean	Padding enabled state

/audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/lowCutFilter` property changes on the device:

Name	Type	Description
.enabled	boolean	Low cut filter enabled state

/audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/available` property changes on the device:

Name	Type	Description
.available	boolean	Whether the input is currently available

/audio/channel/{channelIndex}/input/description

Description of the current input of the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input/description` property changes on the device:

Name	Type	Description
.description	object	
.description.gainRange	object	
.description.gainRange.Min	number	The minimum gain value in dB
.description.gainRange.Max	number	The maximum gain value in dB
.description.capabilities	object	
.description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
.description.capabilities. LowCutFilter	boolean	Input supports setting of low cut filter

Name	Type	Description
.description.capabilities.Padding	object	
.description.capabilities.Padding.available	boolean	Input supports setting of padding
.description.capabilities.Padding.forced	boolean	Padding is forced to be set for the input
.description.capabilities.Padding.value	number	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.

/colorCorrection/lift

Get color correction lift

The value JSON returned via the eventResponse when the `/colorCorrection/lift` property changes on the device:

Name	Type	Description
.red	number	Red lift component
.green	number	Green lift component
.blue	number	Blue lift component
.luma	number	Luma lift component

/colorCorrection/gamma

Get color correction gamma

The value JSON returned via the eventResponse when the `/colorCorrection/gamma` property changes on the device:

Name	Type	Description
.red	number	Red gamma component
.green	number	Green gamma component
.blue	number	Blue gamma component
.luma	number	Luma gamma component

/colorCorrection/gain

Get color correction gain

The value JSON returned via the eventResponse when the `/colorCorrection/gain` property changes on the device:

Name	Type	Description
.red	number	Red gain component
.green	number	Green gain component
.blue	number	Blue gain component
.luma	number	Luma gain component

/colorCorrection/offset

Get color correction offset

The value JSON returned via the eventResponse when the `/colorCorrection/offset` property changes on the device:

Name	Type	Description
.red	number	Red offset component
.green	number	Green offset component
.blue	number	Blue offset component
.luma	number	Luma offset component

/colorCorrection/contrast

Get color correction contrast

The value JSON returned via the eventResponse when the `/colorCorrection/contrast` property changes on the device:

Name	Type	Description
.pivot	number	Contrast pivot point
.adjust	number	Contrast adjustment

/colorCorrection/color

Get color correction color properties

The value JSON returned via the eventResponse when the `/colorCorrection/color` property changes on the device:

Name	Type	Description
.hue	number	Color hue adjustment
.saturation	number	Color saturation adjustment

/colorCorrection/lumaContribution

Get color correction luma contribution

The value JSON returned via the eventResponse when the `/colorCorrection/lumaContribution` property changes on the device:

Name	Type	Description
.lumaContribution	number	Luma contribution value

/lens/iris

Get lens' aperture

The value JSON returned via the eventResponse when the `/lens/iris` property changes on the device:

Name	Type	Description
.continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
.apertureStop	number	Aperture stop value
.normalised	number	Normalised value
.apertureNumber	integer	Aperture number

/lens/iris/description

Get detailed description of lens' iris capabilities

The value JSON returned via the eventResponse when the `/lens/iris/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the iris can be controlled
.apertureStop	object	
.apertureStop.min	number	Minimum aperture stop
.apertureStop.max	number	Maximum aperture stop

/lens/focus

Get lens' focus

The value JSON returned via the eventResponse when the `/lens/focus` property changes on the device:

Name	Type	Description
.normalised	number	Normalised value

/lens/focus/description

Get detailed description of lens' focus capabilities

The value JSON returned via the eventResponse when the `/lens/focus/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the focus can be controlled
.focusDistance	object	
.focusDistance.adjustable	boolean	If focus distance is adjustable
.focusDistance.min	number	Minimum focus distance
.focusDistance.max	number	Maximum focus distance

/lens/zoom

Get lens' zoom

The value JSON returned via the eventResponse when the `/lens/zoom` property changes on the device:

Name	Type	Description
.focalLength	integer	Focal length in mm
.normalised	number	Normalised value

/lens/zoom/description

Get detailed description of lens' zoom capabilities

The value JSON returned via the eventResponse when the `/lens/zoom/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the zoom can be controlled
.focalLength	object	
.focalLength.adjustable	boolean	If focal length is adjustable
.focalLength.min	integer	Minimum focal length
.focalLength.max	integer	Maximum focal length

/presets

Get the list of the presets on the camera

The value JSON returned via the eventResponse when the `/presets` property changes on the device:

Name	Type	Description
.presets	array	List of the presets on the camera (.cset files)
.presets[i]	string	

/presets/active

Get the currently active preset on the camera

The value JSON returned via the eventResponse when the `/presets/active` property changes on the device:

Name	Type	Description
.preset	string	Name of the active preset (with .cset extension, or 'default')

/camera/colorBars

Get the status of color bars display

The value JSON returned via the eventResponse when the `/camera/colorBars` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the color bars are currently enabled

/camera/programFeedDisplay

Get the status of program feed display

The value JSON returned via the eventResponse when the `/camera/programFeedDisplay` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the program feed display is currently enabled

/camera/tallyStatus

Get the tally status of the camera

The value JSON returned via the eventResponse when the `/camera/tallyStatus` property changes on the device:

Name	Type	Description
.status	string	Current tally status of the camera Possible values are: None, Preview, Program.

/camera/power

Get the power status of the camera

The value JSON returned via the eventResponse when the `/camera/power` property changes on the device:

Name	Type	Description
.source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
.milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
.batteries	array	
.batteries[i]	object	
.batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
.batteries[i].chargeRemainingPercent	integer	Remaining battery charge percentage
.batteries[i].statusFlags	array	List of battery status flags
.batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

/camera/power/displayMode

Get the power display mode of the camera

The value JSON returned via the eventResponse when the `/camera/power/displayMode` property changes on the device:

Name	Type	Description
.mode	string	Current power display mode Possible values are: Percentage, Voltage.

/camera/timingReferenceLock

Get the timing reference lock status

The value JSON returned via the eventResponse when the `/camera/timingReferenceLock` property changes on the device:

Name	Type	Description
.locked	boolean	Indicates if timing reference is locked

/video/iso

Get current ISO

The value JSON returned via the eventResponse when the `/video/iso` property changes on the device:

Name	Type	Description
.iso	integer	Current ISO value

/video/supportedISOs

Get the list of supported ISO settings

The value JSON returned via the eventResponse when the `/video/supportedISOs` property changes on the device:

Name	Type	Description
.supportedISOs	array	Array of supported ISO values
.supportedISOs[i]	integer	

/video/gain

Get current gain value in decibels

The value JSON returned via the eventResponse when the `/video/gain` property changes on the device:

Name	Type	Description
.gain	integer	Current gain value in decibels

/video/supportedGains

Get the list of supported gain settings in decibels

The value JSON returned via the eventResponse when the `/video/supportedGains` property changes on the device:

Name	Type	Description
.supportedGains	array	Array of supported gain values in decibels
.supportedGains[i]	integer	

/video/whiteBalance

Get current white balance

The value JSON returned via the eventResponse when the `/video/whiteBalance` property changes on the device:

Name	Type	Description
.whiteBalance	integer	Current white balance

/video/whiteBalance/description

Get white balance range

The value JSON returned via the eventResponse when the `/video/whiteBalance/description` property changes on the device:

Name	Type	Description
.whiteBalance	object	
.whiteBalance.min	integer	Minimum color temperature
.whiteBalance.max	integer	Maximum color temperature

/video/whiteBalanceTint

Get white balance tint

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint` property changes on the device:

Name	Type	Description
.whiteBalanceTint	integer	Current white balance tint

/video/whiteBalanceTint/description

Get white balance tint range

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint/description` property changes on the device:

Name	Type	Description
.whiteBalanceTint	object	
.whiteBalanceTint.min	integer	Minimum white balance tint
.whiteBalanceTint.max	integer	Maximum white balance tint

/video/ndFilter

Get ND filter stop

The value JSON returned via the eventResponse when the `/video/ndFilter` property changes on the device:

Name	Type	Description
.stop	number	Current filter power (fStop)

/video/supportedNDFilters

Get the list of available ND filter stops

The value JSON returned via the eventResponse when the `/video/supportedNDFilters` property changes on the device:

Name	Type	Description
.supportedStops	array	Array of available ND filter stops
.supportedStops[i]	number	

/video/ndFilter/displayMode

Get ND filter display mode on the camera

The value JSON returned via the eventResponse when the `/video/ndFilter/displayMode` property changes on the device:

Name	Type	Description
.displayMode	string	ND filter display mode Possible values are: Stop, Number, Fraction.

/video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

The value JSON returned via the eventResponse when the `/video/supportedNDFilterDisplayModes` property changes on the device:

Name	Type	Description
.supportedDisplayModes	array	Array of supported display modes
.supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

/video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

The value JSON returned via the eventResponse when the `/video/ndFilterSelectable` property changes on the device:

Name	Type	Description
.selectable	boolean	True if ND filter adjustments are selectable via a slider

/video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

The value JSON returned via the eventResponse when the `/video/shutter` property changes on the device:

Name	Type	Description
.continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
.shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
.shutterAngle	number	Shutter angle

/video/shutter/measurement

Get the current shutter measurement mode

The value JSON returned via the eventResponse when the `/video/shutter/measurement` property changes on the device:

Name	Type	Description
.measurement	string	Current shutter measurement mode Possible values are: ShutterAngle, ShutterSpeed.

/video/supportedShutters

Get supported shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/supportedShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of supported shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array of flicker-free shutter speeds
.shutterSpeeds[i]	integer	

/video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/flickerFreeShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of flicker-free shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array
.shutterSpeeds[i]	integer	

/video/autoExposure

Get current auto exposure mode

The value JSON returned via the eventResponse when the `/video/autoExposure` property changes on the device:

Name	Type	Description
.mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
.type	string	Comma-separated list of device types in the auto exposure stack

/video/detailSharpening

Get the current state of detail sharpening

The value JSON returned via the eventResponse when the `/video/detailSharpening` property changes on the device:

Name	Type	Description
.enabled	boolean	Whether detail sharpening is enabled

/video/detailSharpeningLevel

Get the current detail sharpening level

The value JSON returned via the eventResponse when the `/video/detailSharpeningLevel` property changes on the device:

Name	Type	Description
.level	string	Current detail sharpening level of supported shutter speeds Possible values are: Low, Medium, High.

帮助

获得帮助最快捷的途径是访问Blackmagic Design在线支持页面并浏览有关摄影机的最新支持信息和文件。

Blackmagic Design在线支持页面

请登陆Blackmagic Design支持中心获得最新版操作手册、软件以及技术答疑文章，网址：www.blackmagicdesign.com/cn/support

联系Blackmagic Design支持中心

如果我们提供的支持信息无法解答您的疑问，请到支持页面下点击“给我们发送电子邮件”按钮即可发送技术支持请求。或者，您也可以点击支持页面下的“查找您所在地区的支持团队”按钮，致电您所在地区的Blackmagic Design支持中心获得帮助。

查看当前安装的软件版本

要检查您计算机上的Blackmagic Camera Setup的软件版本，请打开“About Blackmagic Camera Setup”窗口查看。

- 在Mac系统下，请到“应用程序”文件夹下打开Blackmagic Camera Setup程序。选择程序菜单中的“About Blackmagic Camera Setup”即可查看版本号。
- 如使用Windows系统，请到开始菜单或开始屏幕下打开Blackmagic Camera Setup实用程序。点击“Help”（帮助）菜单并选择“About Blackmagic Camera Setup”即可查看版本号。

如何获得软件更新

检查完您电脑上安装的Blackmagic Camera Utility软件版本号之后，请登陆网址www.blackmagicdesign.com/cn/support访问Blackmagic Design支持中心查看最新版本。请及时将软件升级到最新版本，但切勿在重要项目制作过程中升级软件。

监管声明

在欧盟范围内处置电子垃圾和电子设备的注意事项。

根据产品所附的提示标志，本设备不得与其它废弃材料共同处置。处置废弃设备时，必须交给指定收集点进行回收。对废弃设备进行单独收集并回收能够节省自然资源，且回收方式不会损害环境和人体健康。获取更多关于废弃设备回收点的信息，请联系您所在城市的回收站，或当时购买设备的经销商。

本设备经过测试，符合FCC规则的第15部分对A类数字设备的限制。这些限制旨在为运行于商业环境中的设备提供合理保护，使其免受有害干扰的影响。本设备可生成、使用且辐射射频能量，如果未按照安装手册来安装和使用本设备，则可能导致对无线电通信的有害干扰。在住宅区运行本产品可能会产生有害干扰，在这种情况下将由用户自行承担消除干扰的费用。

本设备的操作运行必须满足以下两个条件：

1. 本设备或不会造成有害干扰。
2. 本设备必须能够承受任何干扰，包括可能导致意外操作的干扰。

R–R–BMD–20250213001

ICES-3 (A)
NMB-3 (A)

加拿大ISED认证声明

本设备符合加拿大A类数码产品的相关标准。

任何对本产品的改装或预期用途之外的使用均可能导致相关标准认证无效。

必须使用有高品质屏蔽的HDMI电缆连接HDMI接口。

本设备经检测符合商业环境使用要求。在家用环境中，本设备可能会造成无线电干扰。

安全信息

Blackmagic URSA Cine Immersive内附的AC到24V DC电源线必须连接在配有保护地线的电源插座。

Blackmagic URSA Cine Immersive内部没有操作人员可维护的零件。维修服务请联系当地Blackmagic Design服务中心。

将输入电源接口连接到外部DC电池源时，电源接线必须包括限流或熔断。

所使用的电源接线应标记为VW–1或符合IEC 60332或IEC 60695的相关部分。

建议这款摄影机使用24V DC电源。如果使用12V DC电源，所使用的电源接线应足以承受14A电流。

本产品提供有B型电池扣板。建议使用24V (28.8V) 电池或双电压电池。

关于12V (14.4V) 电池，请参阅电池的操作手册或电池上的标记信息，以确定最大连续放电电流。建议最低额定值为12A。

加利福尼亚安全声明

该产品可能会让您暴露在塑料部件中所含的微量多溴化联苯等化学物质下，此类物质已被加州政府认定为可能导致癌症、先天畸形或其他遗传危害的物质。

详情请访问网址：www.P65Warnings.ca.gov

欧洲办事处

Blackmagic Design Europe B.V.
Rijnlanderweg 766, Unit D
2132 NM Hoofddorp
NL

Blackmagic URSA Cine Immersive

在日照环境下，请考虑为摄影机遮光，以防止摄影机或锂电池长时间暴露在阳光下。请勿将锂电池放置在热源附近。

24V“RS”输出接口适合为第三方配件提供电源。请确保电源功耗小于48W。

保修

有限保修

Blackmagic Design保证本产品自购买之日起12个月内不会有材料和工艺上的缺陷。若本产品在保修期内出现质量问题，Blackmagic Design可选择为产品提供免费修理或更换零部件，或者更换缺陷产品。

为确保消费者有权享受本保修条款中的服务，如遇产品质量问题请务必在保修期内联系Blackmagic Design并妥善安排保修事宜。消费者应将缺陷产品包装并运送到Blackmagic Design的指定服务中心进行维修，运费由消费者承担并预先支付。若消费者因任何原因退货，所有运费、保险费、关税等各项税务以及其他费用均由消费者承担。

本保修条款不适用于任何因使用、维护不当或保养不周造成的缺陷、故障或损坏。根据本保修服务，Blackmagic Design的保修范围不包括以下内容：1. 对由非Blackmagic Design专门人员进行的安装、维修或保养所造成的损坏进行维修，2. 对因使用不当或连接到不兼容设备所造成的损坏进行维修，3. 对因使用了非Blackmagic Design生产的零部件所导致的损坏或故障进行维修，及 4. 对经过改装或和其他产品进行组装的产品进行保养维修（因为产品经改装或组装后会增加保养维修所需时间或保养难度）。

由于URSA Viewfinder的光学元件和放大镜原理类似，所以将寻像器直接暴露在阳光下可能会损坏其显示器。当OLED屏幕显示帧导栏等静态或高对比度图像时间过长时，OLED基板可能发生残影或烧屏现象。为避免此现象，长时间不使用寻像器时，切勿遮盖用于探测面部的IR感应器，并确保断开寻像器连接。本产品保修不包括图像残影问题。

本保修条款由BLACKMAGIC DESIGN提供，它可取代所有其他明示或隐含的保修。BLACKMAGIC DESIGN及其供应商对任何有关适销性及就特定用途的适用性等隐含保证不作任何担保。BLACKMAGIC DESIGN负责为消费者提供缺陷产品的维修或更换服务是完整和排他性补救措施。任何间接、特殊、偶然或必然损坏等损坏，不论BLACKMAGIC DESIGN或其供应商是否事先获悉，BLACKMAGIC DESIGN均不予负责。若消费者对本设备进行非法使用，BLACKMAGIC DESIGN概不负责。对因使用本产品造成的损失，BLACKMAGIC DESIGN概不负责。本产品的操作风险由用户自行承担。

© 版权所有 2025 Blackmagic Design。保留一切权利。“Blackmagic Design”、“URSA”、“DeckLink”、“HDLink”、“Workgroup Videohub”、“Multibridge Pro”、“Multibridge Extreme”、“Intensity”以及“Leading the creative video revolution”均为美国及其他国家的注册商标。所有其他公司名称及产品名称可能是其他所有者的注册商标。

“Bluetooth®”（蓝牙）一词，其商标和徽标为Bluetooth SIG, Inc.公司的注册商标，且Blackmagic Design对于上述标识的各类用途均获授权。其他商标和商标名称皆为其各自拥有者所有。

2025년 10월
카메라 가이드

Blackmagicdesign

Blackmagic URSA Cine Immersive



Blackmagic URSA Cine Immersive

목차

Blackmagic URSA Cine Immersive 917
몰입형 포맷이란? 917
Apple Immersive Video 917
URSA Cine Immersive 918
시작하기 919
카메라 전원 연결하기 919
녹화 설정 셋업하기 920
이미지 모니터링하기 922
모니터링 옵션 및 오버레이 922
첫 번째 숏 프레이밍하기 923
카메라 높이 923
카메라 틸트 924
위치 설정 925
시청자 주체성 926
카메라 움직임 926
데이터 전송률 표 927
오디오 및 타임코드 928
Apple Spatial Audio Format(ASAF) 928
앰비소닉 928
핀 마이크 및 초소형 마이크 929
핸드헬드 마이크 929
타임코드 연결하기 929
촬영 931
노출 요구 사항 931
셔터 속도 931
ISO 및 다이나믹 레인지 932
ND 933
오프스피드 녹화 934
히스토그램 935
LCD 모니터 옵션 935
녹화 936
재생 937
클립 전송하기 938
요약 939
DaVinci Resolve Studio 940
워크스테이션 권장 사항 940
프로젝트 셋업 941
마스터 설정 941
비디오 모니터링 942
영상 스케일링 942
색상 관리 943
DaVinci Wide Gamut에 관하여 944
Apple Immersive 프로세싱 945
영상 입수 945
편집 947
취합 947
배경막 947
뷰어 사용하기 949
트랜지션 951
Apple Immersive Motion Data 952
타이틀 953
사용자 조언 954
VFX 합성 955
MediaOut 노드 955
Renderer3D 956
Immersive Patcher를 사용하여 합성하기 957
컬러 그레이딩 958
노이즈 리덕션 958
프라이머리 그레이딩 959
세컨더리 959
크기 조절 960
3D 팔레트 960
CST 컬러 관리 962
CST 타임라인 매핑 962
CST 그레이딩 모니터 매핑 963
CST Apple Vision Pro 모니터링 965
오디오 966
Apple Spatial Audio Format(ASAF) 966
딜리버리 967
트랜지션 바이패스하기 967
Apple Vision Pro로 스트리밍하기 968
Apple Immersive Video Utility에 관하여 968
Developer Information 969
Camera Control REST API 969
Livestream Control API 971
Clips Control API 975
Media Pool Control API 976
Monitoring Control API 979
Event Control API 988
System Control API 988

Transport Control API 994
Timeline Control API 998
Media Control API 1001
Slate Control API 1004
Preset Control API 1010
Audio Control API 1012
Lens Control API 1017
Video Control API 1021
Camera Control API 1029
Immersive Control API 1032
Color Correction Control API 1033
Notification websocket - 1.0.0 1038
messages 1038
도움말 1062
규제 사항 1063
안전 정보 1064
보증 1065

Blackmagic URSA Cine Immersive

Blackmagic URSA Cine Immersive를 구매 또는 대여하신 것을 축하드립니다.

Blackmagic URSA Cine Immersive는 일반적으로 몰입형 영상이라고 하는 180도 스테레오스코픽 VR 푸티지를 촬영하도록 설계되었습니다. 특히, 현존하는 최상급 몰입형 영상 경험을 제공하는 Apple Vision Pro를 위한 Apple Immersive Video 촬영에 최적화되어 있습니다.

본 사용자 가이드는 Blackmagic URSA Cine Immersive를 소개 및 촬영 방법을 소개하며, 주로 카메라 기본 작동 및 Apple Immersive Video 포맷의 딜리버리에 초점을 맞췄습니다. 이 포맷에 대한 이해도를 통해 카메라 또한 쉽게 이해하실 수 있습니다.

다음 페이지에서는 카메라 사용법 뿐만 아니라 몰입형 포맷이 주는 장점에 대한 기술적 조언 및 사용자 조언을 함께 제공합니다.

몰입형 포맷이란?

몰입형 포맷이란 3차원 공간 영상 포맷을 의미합니다. 여러분 대다수는 이와 유사한 기존 기술을 아실 겁니다. 몰입형이 그 기술들과 무엇이 다른지 기존 기술들에 대한 전반적 개요를 알려드리겠습니다.

- 기존의 3D 시네마는 사람의 두 눈 사이 거리를 모방한 축간 거리를 이용해 두 대의 카메라로 각 장면을 촬영하여 관객으로 하여금 프레임 내의 깊이를 경험할 수 있도록 합니다. 이러한 스테레오스코픽 이미지를 통해 시청자가 깊이를 경험할 수 있는 반면, 촬영 감독의 구성 연출이 관객의 시점을 좌우합니다.
- 구형 영상은 3D와 360도 카메라의 원리를 조합하여 관객이 영상의 깊이를 경험하는 동시에 어느 방향으로든 볼 수 있게 합니다. 360도 영상은 이미지의 작은 부분이라도 관객이 선명하게 볼 수 있도록 매우 높은 해상도로 제작해야 합니다. 인체의 한계로 인해 시청자는 고개를 어깨에서 어깨까지만 돌릴 수 있기 때문에, 뒤를 돌아보려면 보통 일어서 있거나 회전 의자에 앉아 있어야 합니다.
- 볼류메트릭(Volumetric) 영상은 관객이 깊이를 경험하고 주위를 돌아보는 경험 외에도 이동하거나 시점을 옮길 수 있습니다. 카메라 어레이를 통해 볼류메트릭 영상을 캡처하는 반면 대부분의 가상 현실 또는 증강 현실은 실시간으로 렌더링한 3D 에셋을 활용하며 참가자에게 안전한 환경을 제공해야 합니다.

몰입형은 기존 시네마와 3D 인터랙티브 포맷을 최고 장점들만 합친 것입니다. 이것은 가장 단순한 형태로 높은 프레임 레이트와 고해상도의 180도 스테레오스코픽 포맷입니다.

Apple Immersive Video

Blackmagic URSA Cine Immersive는 특별히 Apple Immersive Video를 캡처하도록 설계되었습니다. 이 포맷은 Apple Vision Pro용으로 사용되며 180도 시야각에서 고해상도 스테레오스코픽 3D 비디오를 지원합니다.

Apple Immersive Video는 다음과 같은 기능을 지원합니다.

- 초당 90프레임 재생
- 각 눈 당 최대 4320 x 4320 해상도 제공
- 카메라 원본 이미지를 180도 환경으로 매핑하는 특수 메타데이타 시스템
- 3D 사운드 제공을 위한 ASAF(Apple Spatial Audio Format)

Apple Immersive Video는 고해상도의 180도 스테레오스코픽 영상 재생이 가능한 하드웨어를 사용해야 합니다. Apple Vision Pro는 현재 이 포맷을 지원하는 유일한 기기입니다.



Apple Vision Pro

이 헤드셋은 7.5 마이크론 픽셀 피치를 제공하는 3660 x 3200 픽셀의 듀얼 마이크로 OLED 디스플레이를 탑재했습니다. 또한 13개의 트래킹 카메라 및 LiDAR 스캐너, 관성 측정 장치(IMU), 머리와 눈 동작을 형상에 일치시키는 눈 추적 센서를 포함합니다.

M2 프로세서는 8코어 CPU 및 10코어 GPU 성능을 지원하여 최대 100fps로 재생합니다.

URSA Cine Immersive

Blackmagic URSA Cine Immersive는 Apple Vision Pro용 Apple Immersive Video를 캡처하도록 설계되었습니다.



Blackmagic URSA Cine Immersive

URSA Cine Immersive는 두 개의 일치된 58메가픽셀 RGBW 센서를 사용합니다. 이 센서는 거의 정사각형으로, 가로 및 세로 방향으로 180도 시야를 캡처합니다. 각 센서에는 210도의 어안 렌즈가 장착되어 있으며, 렌즈의 중심부는 서로 64mm 간격으로 떨어져 있습니다.

내장 ND 필터는 노출 제어를 위해 최대 8스탑을 지원합니다. 렌즈 조리개는 대략 f4.5에 고정되어 있습니다.

카메라는 두 개의 8K 이미지 스트림을 90fps로 녹화하며, 이로 인해 일반 4K 24p보다 훨씬 높은 데이터 전송률을 지원합니다.

시작하기

전원을 연결하고 카메라를 켜기만 하면 Blackmagic URSA Cine Immersive의 사용 준비가 완료됩니다. 카메라 모듈 베이에 포맷된 Blackmagic Media Module 8TB가 로딩되어 있어 곧바로 녹화를 시작할 수 있습니다.

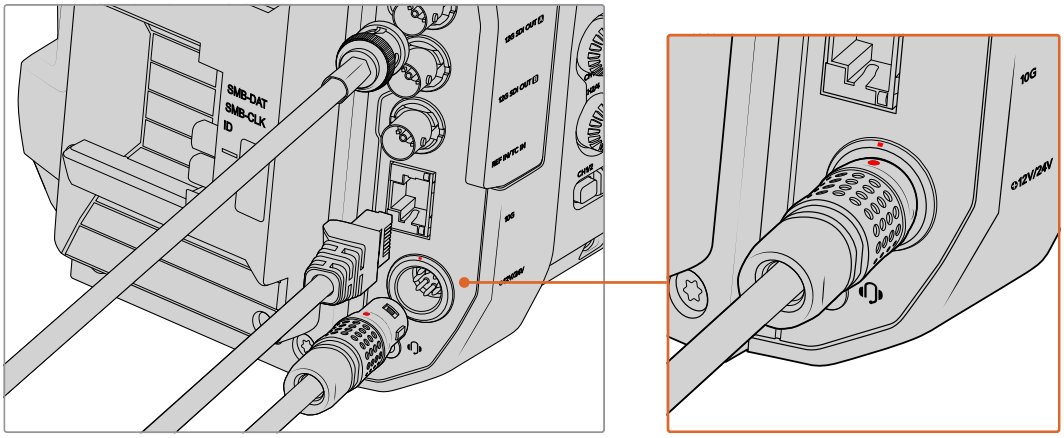
카메라 전원 연결하기

카메라에 전원을 연결하는 가장 빠른 방법은 제품에 포함된 전원 어댑터를 사용하여 외부 전원에 연결하는 것입니다.

외부 전원에 연결하기

1. 우선 먼지 보호 고무 덮개를 제거한 다음 안전한 곳에 보관하세요. 표준 IEC C13 주전원 케이블을 사용해 AC -24V DC 어댑터를 메인 전원 콘센트에 연결하세요.
2. 어댑터의 8핀 커넥터를 카메라 뒷면 패널의 24V DC 전원 커넥터에 연결하세요.

카메라에 배터리를 장착한 채 외부 전원에 연결할 경우, 카메라에서 외부 전력만 사용합니다. 완충된 배터리를 장착한 채 외부 전원을 분리하면 곧바로 카메라 배터리 전원으로 전환됩니다.



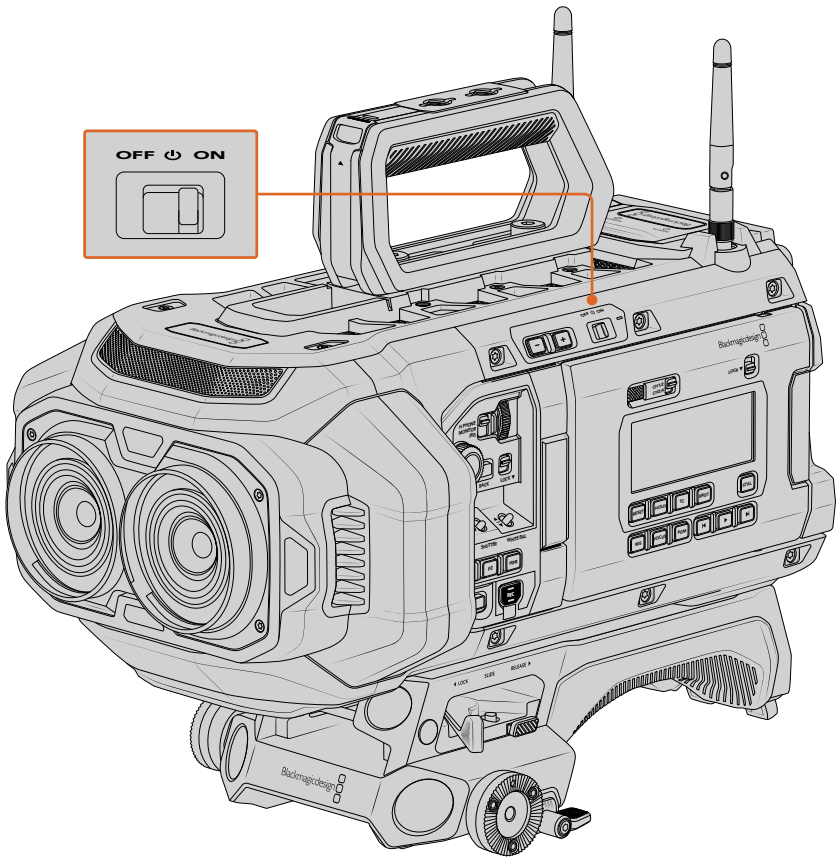
전원을 연결하려면 빨간 점이 정렬되도록 맞춘 다음 커넥터를 제대로 장착될 때까지 부드럽게 밀어 넣으세요. 전원을 분리하려면 커넥터의 금속 슬리브를 잡고 카메라에서 부드럽게 잡아당기세요. 그러면 슬리브가 뒤로 빠져 커넥터를 소켓으로부터 분리할 수 있습니다.

카메라에 포함된 배터리 플레이트는 24V B 마운트 배터리를 위해 설계된 B 마운트 플레이트입니다. 높은 전압을 연결하면 열 손실이 줄어들어 높은 프레임 레이트의 영상을 녹화할 수 있고, 카메라에 연결된 다양한 액세서리에 전원을 공급할 수 있기 때문에 URSA Cine Immersive에는 24~34V의 전원을 연결하는 것이 가장 좋습니다.

옵션 제품인 VLock 및 Gold Mount 플레이트(24V 배터리 호환), 그리고 Gold Mount Plus 플레이트(26V 서드파티 배터리 호환)도 사용할 수 있습니다.

참고 URSA Cine Immersive에 12~18V 전원 공급 장치 또는 배터리를 사용해 전원을 공급하는 경우, 3핀 RS 커넥터 또는 2핀 Lemo 스타일 커넥터를 통한 액세서리 전원 공급은 불가능합니다.

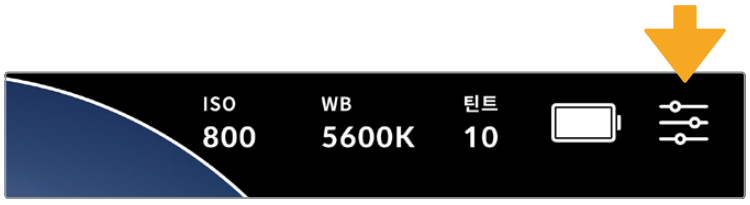
배터리 장착 또는 외부 전원을 연결한 후, URSA Cine Immersive 카메라를 켜세요. 카메라 상단에 있는 카메라 스위치를 ON으로 설정하세요.



전원 스위치를 ON으로 설정하면 카메라에 전원이 공급됩니다.

녹화 설정 셋업하기

이 섹션에서는 몰입형 콘텐츠 촬영을 위해 반드시 알아야 할 고유하거나 중요한 카메라 설정에 관한 정보를 소개합니다. 카메라 설정 메뉴를 열려면 접이식 터치스크린 또는 보조 터치스크린의 우측 상단에 있는 메뉴 설정 아이콘을 누르세요.



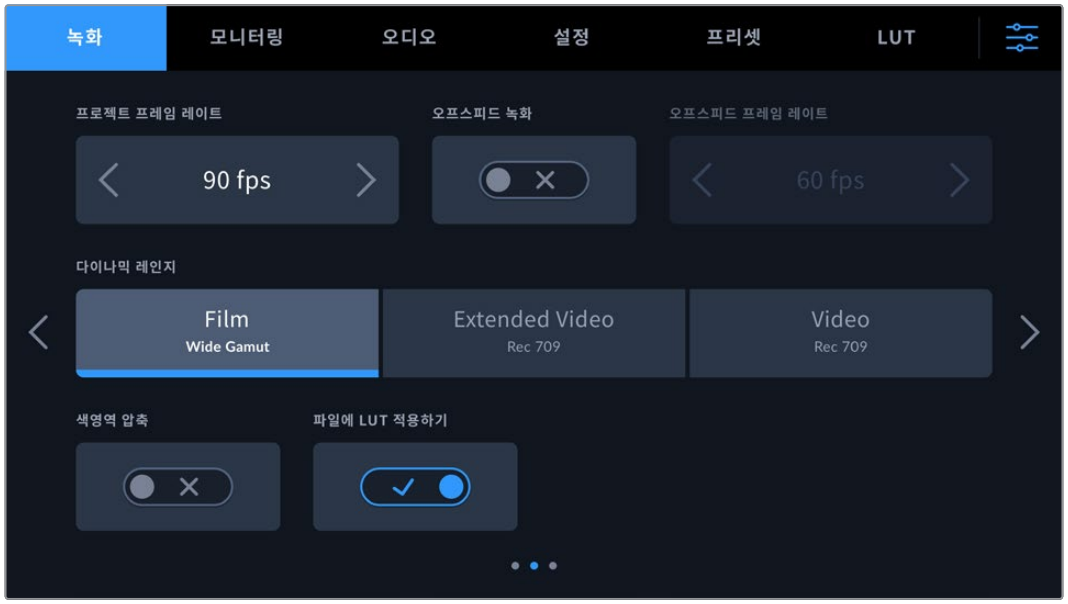
녹화 설정 1

이 페이지의 '코덱 및 화질' 옵션을 통해 작업 중인 프로젝트에 적합한 Blackmagic RAW 코덱을 설정할 수 있습니다. Blackmagic RAW 옵션은 네 개의 '고정 비트레이트'와 '균일 화질' 설정으로 나누어져 있습니다. 만약 Apple Vision Pro을 위해 90fps으로 녹화할 경우, 12:1 또는 Q3 압축 설정을 사용할 것을 강력히 추천합니다. 이 설정은 현재 컴퓨팅 하드웨어를 사용하여 관리할 수 있는 데이터 전송 속도와 파일 크기를 유지하면서 동시에 매우 높은 이미지 품질을 유지할 수 있는 최적의 설정입니다.



녹화 설정 2

카메라 '녹화' 설정의 두 번째 페이지는 프레임 레이트 설정을 포함합니다.



카메라의 프로젝트 프레임 레이트는 Apple Vision Pro의 디스플레이 프레임 레이트에 적합하도록 90fps로 설정되어야 합니다.

다른 시스템을 위한 촬영을 한다면, 60fps 또는 50fps와 같은 더 낮은 프레임 레이트만으로도 충분합니다. 24fps 영화 콘텐츠를 위한 시각 효과 플레이트를 촬영한다면 24fps 또는 25fps와 같은 기존 필름 스타일의 낮은 캡처 프레임 레이트가 적합합니다.

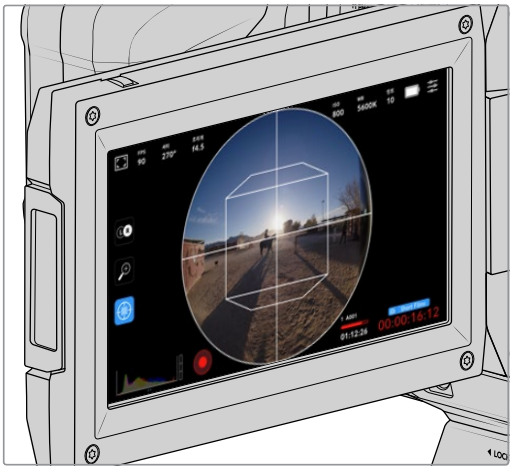
이제 URSA Cine Immersive 사용 준비가 모두 완료되었습니다. 카메라의 미디어 모듈이 이미 포맷되어 있어 녹화를 바로 시작할 수 있습니다.

이미지 모니터링하기

URSA Cine Immersive는 촬영 중 이미지 모니터링을 위한 여러 옵션을 제공합니다.

접이식 터치스크린은 카메라 운영자가 사용하기 위한 것입니다. 카메라의 물리적 부피를 줄이기 위해 스크린을 뒤집어 사용해도 화면을 볼 수 있으며, 또는 자기 자신을 촬영하는 경우 셀프 모니터링을 위해 스크린을 앞쪽으로 돌려서 사용할 수 있습니다.

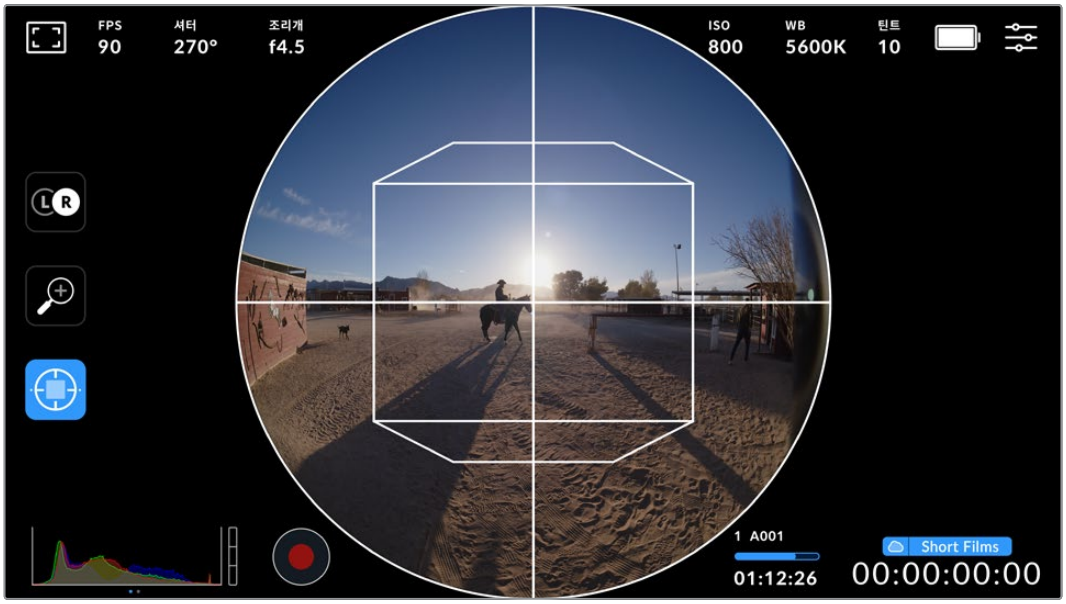
보조 터치스크린을 사용하면 카메라 운영자의 시야를 가리지 않고 다른 스태프가 이미지를 모니터링할 수 있습니다. 카메라의 극단적으로 넓은 시야각으로 인해 보조 터치스크린 사용 시 주위가 필요합니다.





모니터링 옵션 및 오버레이

카메라는 렌즈 공간을 표시하며, 한 번에 한쪽 눈만 볼 수 있습니다. 접이식 터치스크린 및 보조 터치스크린은 180도 뷰를 디스플레이하기 위해 마스킹됩니다. SDI 출력은 각 센서로 캡처한 풀 210도 이미지를 보여줍니다.



모든 모니터링 출력은 같은 눈을 보여줍니다. 두 터치스크린에 모두 표시되는 'L R' 버튼을 눌러 좌안 및 우안을 전환할 수 있습니다.

오버레이 옵션은 터치스크린에 있는 오버레이 버튼을 눌러 활성화할 수 있습니다. 오버레이는 다음과 같은 기능을 포함합니다.

- 180도 외측 경계선
- 프레임 중앙을 표시하는 중앙 십자선
- 내측 가이드는 Apple Vision Pro에서 정면을 똑바로 바라보았을 때 보이는 대략적인 시야각을 나타냅니다.

이러한 도구를 통해 영상 캡처 시 정확한 프레이밍 및 위치 선정이 가능합니다.

첫 번째 숏 프레이밍하기

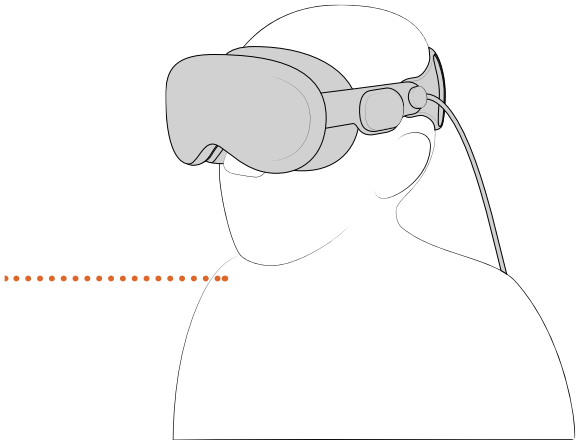
몰입형과 기존 2D 콘텐츠는 공통점이 많지만 뚜렷한 차이점들도 있습니다.

예를 들어, 휴대폰을 너무 빨리 스크롤해서 차 안에서 멀미를 경험하신 적이 있나요? 이는 두뇌에 상충되는 신호가 전달 될 때 느껴지는 감각의 불일치 때문입니다. 예를 들어 사용자의 내이는 몸이 움직이고 있다는 신호를 보내지만 눈은 몸이 정지한 상태라고 신호를 보내는 경우입니다.

몰입형 영상에서 이와 유사한 경험이 쉽게 발생합니다. 예를 들어, 만약 카메라를 패닝하면 시청자의 눈은 몸이 움직이고 있다는 신호를 전송하는 반면, 목과 내이는 시청자의 몸이 정지 상태라는 신호를 전송합니다. 결과적으로, 시청자의 흥미를 유발하는 카메라 움직임을 습득하기 전에 고정 숏 촬영 방법을 먼저 숙지하는 것이 중요합니다.

카메라 높이

기존 2D 시네마를 촬영할 때는 장면 내의 힘의 균형을 표현하기 위해 카메라를 높게 또는 낮게 배치하기도 합니다. 하지만 몰입형 영상을 촬영할 때는 바닥에 카메라를 놓았을 때 어떤 영향이 발생하는지 고려해야 합니다. 이 경우 시청자는 어떤 느낌을 받을까요? 시청자는 마치 목까지 땅에 묻인 느낌을 받을 수 있습니다.



시청자가 인식하는 바닥 높이는 시청 경험에 영향을 미칠 수 있습니다. 낮은 카메라에서는 시청자가 인식하는 바닥 높이가 상식에 어긋날 수 있습니다.



몰입형 촬영 시 낮은 각도는 시청자에게 불편함을 줄 수 있습니다.

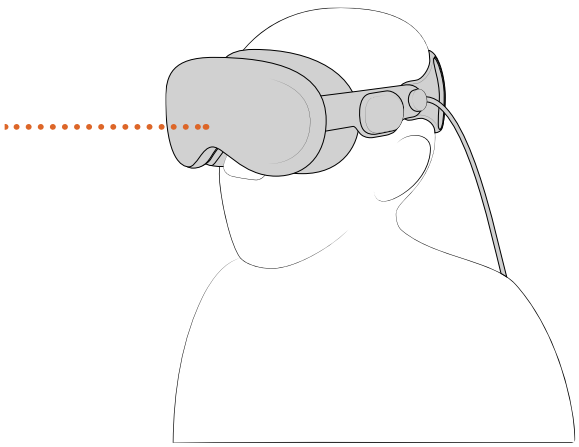
물리적으로 현장에 있는 느낌을 주기 위해서는 카메라 높이와 위치를 장면 내에서 시청자가 서있거나 앉아 있는 높이로 맞추는 게 일반적으로 가장 좋습니다. 이러한 높이에서도 여전히 제작자의 창작 의도를 표현할 수 있는 여지는 있습니다. 인물이 등장하는 장면의 경우, 인물보다 약간 높은 위치의 카메라가 시청자로 하여금 인물을 약간 내려다 볼 수 있는 통제력을 부여할 수 있습니다. 반대로, 인물보다 약간 낮은 위치의 카메라는 시청자로 하여금 인물을 올려다 보도록 만듭니다.

그렇다고 해서 좀 더 과감한 카메라 위치를 사용하면 시청자가 편안함을 느낄 수 없다는 것은 아닙니다. 날으는 카메라, 탑다운 뷰, 로우 앵글 등 많은 다양한 카메라 기법들이 몰입형 영상에 성공적으로 활용되고 있습니다. 알맞은 편집, 움직임, 문맥이 조화를 이룰 때 다양한 카메라 기법은 시청자를 즐겁게 할 수 있습니다.

카메라 틸트

기존의 2D 콘텐츠 촬영에서 카메라를 높게 또는 낮게 배치 시 주로 카메라 틸트를 사용하여 구도를 유지합니다. 이러한 기법을 몰입형 이미지에 사용 시 어떻게 반영되는지 고려해 보세요.

카메라 각도가 아래로 향하게 하면, 시청자는 지평선을 보기 위해 고개를 위로 들어야 합니다. 멀미를 일으키진 않겠지만 현장감이 깨집니다. 현실에서 지평선은 언제나 바로 정면으로 보이기 때문입니다.



몰입형 숏은 지평선이 0도일 때 가장 자연스럽게 느껴질 것입니다.



URSA Cine Immersive 카메라를 틸트했다면, 평상시에 그냥 바라보는 지평선과 지평선을 바라보기 위해 움직이는 감각 사이에서 시청자의 몰입감은 깨질 것입니다.

그런 이유로 대부분의 몰입형 숏은 0도 또는 0도에 가깝게 틸트되어 캡처됩니다. 카메라가 가로 및 세로 방향으로 180+도 시야를 캡처하기 때문에 시청자는 여전히 위를 보거나 아래를 내려다 볼 수 있습니다. 제작자가 특정 방향을 보도록 시청자를 강요하는 것이 아니라 시청자가 통제권을 가집니다.

시청자가 영상의 다양한 부분들을 보도록 유도하기 위해서 조명, 조리개, 프레이밍, 유도선 등 여러 기술을 사용할 수 있습니다.

위치 설정

몰입형은 일반적으로 앉아서 시청합니다. 증강 현실 또는 가상 현실 포맷이 아니기 때문에 시청자는 서거나 움직여 돌아 다니지 않습니다. 현재 앉은 상태인가요? 시험적으로 머리를 좌우로 돌려보고 위아래로 움직여 보세요. 몸을 움직이지 않고 얼마만큼 편안하게 머리를 돌릴 수 있나요? 몰입형 포맷의 시야각은 대부분 사람들이 편안하게 여기는 회전 범위를 초과합니다. 이는 시청자가 좌우로 돌아보아도 여전히 볼 수 있는 영상이 있다는 뜻이기 때문에 바람직합니다. 이 실험을 통해 시야각 내의 어느 위치에 주요 동작을 포함시켜야 하는지 쉽게 이해할 수 있습니다.

일반적인 머리 움직임 범위

회전: 120~160도
고개 숙이기 및 젖히기: 110~130도

눈동자 움직임과 고개 회전이 동시에 이루어지면 어깨 뒤를 쳐다보는 것도 가능하지만, 대부분의 사람들은 그 자세를 유지하는 것을 불편하게 느낍니다.

더 정확한 카메라 구도에 도움에 되도록 오버레이 기능을 활성화하세요. 일반적인 시야를 표시하는 오버레이 선 안에서 촬영하면 콘텐츠 시청 시 가시성에 도움이 됩니다.

프레임 내의 관심 지점(Point of Interest)의 움직임 또한 고려해야 합니다. 좌측 끝에서 우측 끝까지 인물을 추적하려면 머리를 최대로 움직여야 합니다. 이와 유사하게 POI가 이미지 가장자리에 있는 숏에서 반대 가장자리에 POI가 있는 숏으로 전환할 때도 머리를 최대로 움직여야 합니다. 세심한 구도로 촬영하여 창의성을 잃지 않으면서도 시청자의 편안함을 제공해야 합니다.

시청자 주체성

기존의 2D 콘텐츠를 촬영할 때는 프레임 및 렌즈 선택, 초점 설정을 사용해 시청자의 주의를 유도합니다. 몰입형 촬영에서는 180도 시야각으로 인해 영상 정보를 제외하거나 시청자의 주의를 유도하기가 더 어렵습니다. 몰입형은 시청자가 이미지를 탐색할 수 있도록 훨씬 확장된 주체성을 부여합니다. 이런 점이 몰입형 콘텐츠를 아주 매력적이게 만드는 요소 중 하나입니다. 하지만 세심하게 연출된 음향, 조명, 동선, 구도를 사용해 시청자의 주의를 여전히 유도할 수 있습니다.

카메라 움직임

몰입형 카메라는 굳이 고정되어 있어야 할 필요가 없지만 카메라의 움직임이 부자연스럽거나 불편하지 않도록 필히 조심해야 합니다. 몰입형 카메라의 편안한 움직임을 이해하는 데에 좋은 비유로써 자동차를 들 수 있습니다. 대형 차량은 급격한 속도 또는 방향 전환에 저항하는 질량과 관성을 가지고 있습니다. 따라서, 대형 차량은 시간에 따라 가속하거나 감속할 것이고 예측할 수 있고 통제된 방식으로 움직일 것입니다. 고정 숏을 숙달하고 동적인 몰입형 숏으로 넘어갈 때 편안한 움직임으로 제작할 수 있도록 다음 사항을 고려하세요.

- **지속 시간:** 시청자가 카메라 움직임에 적응하고 즐길 수 있도록 시간을 주세요. 짧게 주어진 시간은 방향 감각 상실을 유발할 가능성이 높습니다.
- **강도:** 몰입형 콘텐츠 편집 시 저강도의 움직임이 편집하기 훨씬 쉽습니다. 고강도 움직임도 가능하지만 준비 시간과 정리 시간을 할애하면 시청자가 안정을 찾는데 도움이 될 것입니다.
- **시각적 기준점:** 이는 장면 안에서 고정된 부분입니다. 시각적 기준점이 없는 숏에서는 방향 감각을 상실할 수 있습니다. 차량 내부와 같은 전경 요소 또는 수평선과 같은 원경 요소는 움직이는 숏에서 안정감을 찾는 데 도움이 됩니다.

데이터 전송률 표

아래 표는 90fps 및 24fps 프레임의 8K x 7K 몰입형 콘텐츠의 데이터 전송률을 보여줍니다.

Blackmagic RAW 고정 비트레이트 설정 시 90fps의 최대 데이터 전송률

8K x 7K 몰입형 콘텐츠			
코덱	**픽셀 크기**	**프레임 레이트**	**데이터 전송률**
Blackmagic RAW 5:1	8160 x 7200 스테레오	90	3203 MB/s
Blackmagic RAW 8:1	8160 x 7200 스테레오	90	2002 MB/s
Blackmagic RAW 12:1	8160 x 7200 스테레오	90	1335 MB/s
Blackmagic RAW 18:1	8160 x 7200 스테레오	90	889 MB/s

Blackmagic RAW 고정 비트레이트 설정 시 24fps의 최대 데이터 전송률

8K x 7K 몰입형 콘텐츠			
코덱	**픽셀 크기**	**프레임 레이트**	**데이터 전송률**
Blackmagic RAW 5:1	8160 x 7200 스테레오	24	854 MB/s
Blackmagic RAW 8:1	8160 x 7200 스테레오	24	534 MB/s
Blackmagic RAW 12:1	8160 x 7200 스테레오	24	356 MB/s
Blackmagic RAW 18:1	8160 x 7200 스테레오	24	237 MB/s

오디오 및 타임코드

오디오는 몰입형 경험에서 매우 결정적인 부분입니다. URSA Cine Immersive를 전용 오디오 캡처 장치와 함께 사용하는 것이 중요합니다.

Apple Spatial Audio Format(ASAF)

Apple 공간 오디오 포맷 또는 ASAF는 Apple Immersive Video를 시청 시 3D 오디오를 제공하는 데 사용됩니다. ASAF는 넓은 의미에서 Apple Positional Audio Framework의 일부로 3가지 다른 유형의 오디오 녹음을 저장할 수 있습니다.

- **채널 기반:** 기존의 서라운드 믹스를 ASAF 파일에 임베드할 수 있습니다. 헤드락(head locked) 나레이션 및 음악에 이상적인 방식입니다.
- **객체 기반:** 음향은 3D 공간의 위치를 나타내는 메타데이터를 포함한 개별적인 객체로써 각 스피커 채널과는 분리하여 취급할 수 있습니다. 이러한 음향은 일반적으로 모노 녹음이며 음향의 3D 좌표에 장면 안에서 움직이는 객체를 따르도록 애니메이션 효과를 적용할 수 있습니다. 장면 안에서 시각적인 객체를 가리키는 다이어제틱(Diegetic) 오디오에 이상적입니다.
- **앰비소닉:** 구면 조화 함수를 이용해 전지향성 오디오를 렌더링할 수 있는 완전한 구형 서라운드 포맷입니다. 환경 음향에 이상적입니다.

다음 부분에서는 효과적인 ASAF 파일을 구축하기 위해 필요한 장치 개요를 설명합니다.

앰비소닉

앰비소닉 마이크는 정확한 기하학적인 패턴 안에 정렬된 4개 이상의 캡슐을 포함하고 있습니다. 각 캡슐은 동일한 소리를 소리가 나는 방향에 따라 다른 도착 시간과 강도로 캡처합니다. 이 캡슐은 나중에 컴퓨터에서 프로세싱하기 위해 개별 채널에 녹음됩니다. 소프트웨어는 이 차이들을 비교하여 녹음된 소리의 방향과 속성을 추정합니다.



1차 앰비소닉 마이크

앰비소닉 마이크

앰비소닉 마이크는 포함된 캡슐 수에 따라 분류됩니다. 더 많은 캡슐은 공간 디테일을 더욱 정확하게 재현할 수 있는 마이크라는 의미입니다.

1차: 4채널

2차: 9채널

3차: 16채널

4차: 25채널 이상

앰비소닉 마이크가 카메라에 달려있던 그 옆에 있든 상관없이 마이크의 방향이 카메라의 촬영 방향과 정확하게 일치하는 것이 중요합니다. 이뿐만 아니라, 앰비소닉 마이크를 엔드파이어(endfire), 브로드사이드(broadside), 위/아래 방향(upward or downward) 등 다양한 설정에서 사용할 수 있기 때문에 오디오 전문가는 어떤 설정이 사용되고 있는지 반드시 확인해야 합니다. 이러한 정보는 오디오 레코더의 메타데이터를 통해서 혹은 간단한 메모를 공유해 전달할 수 있습니다.

앰비소닉 녹음은 환경음으로 사용될 수 있습니다. 이 위에 다른 채널 기반 또는 객체 기반 음향 효과들을 겹쳐 쌓아 올릴 수 있습니다. 만약, 앰비소닉 마이크가 카메라와 정확히 정렬되었다면 앰버소닉 녹음은 오디오 엔지니어가 추가적인 사운드 디자인을 더할 시 레퍼런스로 활용할 수도 있습니다.

핀 마이크 및 초소형 마이크

180도 카메라로 촬영할 때는 붐대를 사용 시 장면에 포함되기 때문에 사용할 수 없습니다. 즉, 마이크를 숨기거나 핀 마이크를 사용해 다이어제틱 오디오와 다이이얼로그를 녹음해야 합니다.

다음의 예를 참고하세요.

- 무선 송신기 및 수신기를 사용해 해당 오디오 소스를 URSA Cine Immersive의 두 개의 XLR 입력에 바로 연결하세요.
- 또는 해당 오디오 소스를 외부 오디오 레코더에 연결할 수 있습니다. 이러한 오디오 레코더는 대형 멀티 채널 기기 또는 몸에 부착할 수 있는 소형 기기일 수도 있습니다.

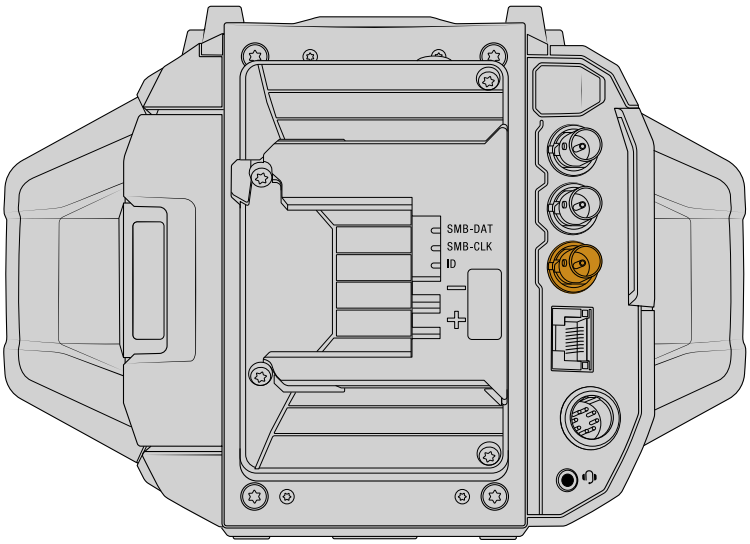
이러한 녹음본은 이후에 채널 기반 또는 객체 기반 사운드와 동기화하여 믹싱할 수 있습니다.

핸드헬드 마이크

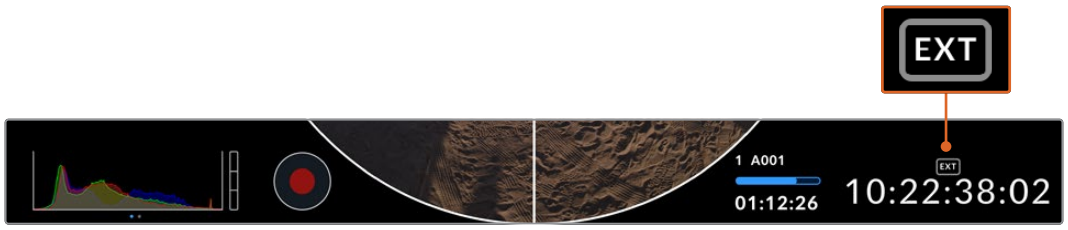
촬영 후 현장에서 추가 음향 효과를 녹음하는 것은 후반 작업 시간을 절약해줄 뿐 아니라 훨씬 현장감 있는 사운드를 담을 수 있습니다. 제작 현장에 핸드헬드 하이퍼카디오이드 또는 이와 유사한 마이크를 가져가면 매우 유용합니다. 마이크를 액션과 최대한 가까이에 배치하여 다시 연기할 수 있기 때문입니다. 이 녹음본은 나중에 채널 기반 또는 객체 기반 사운드로 최종 믹싱됩니다.

타임코드 연결하기

외부 타임코드를 URSA Cine Immersive에 연결하려면, BNC 케이블을 사용해 타임코드 생성기를 카메라 뒷면의 타임코드 입력 단자에 연결하세요. 듀얼 시스템 촬영 시 동기화된 타임코드에 연결하면 영상과 오디오를 후반 작업에서 쉽게 동기화할 수 있습니다.



프레임 레이트가 매칭되는 외부 타임코드를 이 단자에 연결하면 카메라가 이 수신 타임코드에 자동으로 잠깁니다. 외부 타임코드 신호가 잠기면 URSA Cine Immersive의 LCD 타임코드 카운터 옆에 'EXT' 표시가 나타납니다. 케이블을 뽑을 경우, 타임코드가 멈추며 'EXT' 표시가 'INT'로 바뀌어 이제 카메라의 내부 타임코드 시간이 사용되고 있음을 알 수 있습니다.



타임코드 생성기에 연결하면 카메라 타임코드에 'EXT'가 표시됩니다.

참고 SMPTE 타임코드 표준은 23.976, 24, 30 등과 같은 일반적인 프레임 레이트에 맞추어져 있습니다. 90fps를 위한 SMPTE 타임코드 표준은 없습니다. 따라서, URSA Cine Immersive의 프로젝트 프레임 레이트가 90fps에 설정되어 있으면 카메라는 실제로 30fps 타임코드를 사용합니다. SDI 출력도 30p에서 운영합니다. 모든 타임코드 액세서서리 또는 외부 오디오 레코더는 30fps로 설정되어야 합니다.

촬영

이제 Blackmagic URSA Cine Immersive 카메라 작동법과 대부분의 몰입형 포맷에 대한 기본 사항을 배웠으니 이번에는 카메라 심층 작동법에 대해 알아보겠습니다.

노출 요구 사항

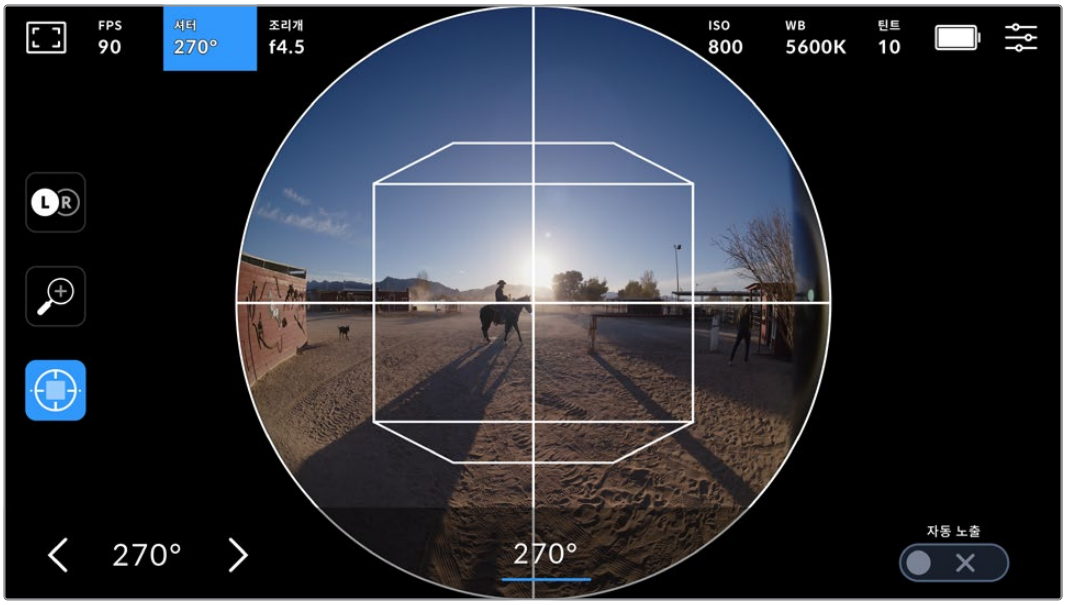
Apple Immersive Video는 기존 24fps 캡처보다 더 높은 노출을 필요로 하므로, 촬영 및 조명에 대한 접근법을 재고해야 할 수도 있습니다. 180도 셔터를 유지한다면, 90fps 캡처에는 동일한 분량임에도 거의 4배 이상 조명이 필요합니다. 또한, 렌즈의 고정 조리개 값이 약 f4.5이기 때문에 일반적인 대구경 렌즈에 비해 최대 3스탑까지 어둡게 됩니다. 즉, 적절한 노출을 확보하기 위해 최대 7스탑 이상의 빛이 더 필요할 수도 있습니다. 그러나 90fps에 촬영 시 270°에서 360° 사이의 넓은 셔터 개각도를 사용하면 높은 프레임 레이트에서 발생하는 빛 손실을 보안하는 동시에 만족스러운 모션 블러도 얻을 수 있습니다.

셔터 속도

'셔터' 상태 정보는 셔터 속도 또는 셔터 개각도를 나타냅니다. '설정' 메뉴에 있는 셔터 측정 설정에서 셔터 정보를 셔터 개각도 또는 셔터 속도로 표시할 수 있습니다.

셔터 속도 변경하기

1. 상단 도구바에 있는 '셔터' 아이콘을 누르세요
2. 좌/우 화살표를 사용해 일반 셔터 속도 프리셋을 순환하세요. 특정 셔터값을 선택하려면 스크린 왼쪽 하단의 셔터값 표시를 두 번 누르세요. 화면에 나타난 키패드를 통해 11.2에서 360도 사이의 최대 소수점 두 자리까지 셔터값을 입력할 수 있습니다.



사용자 지정 프레임 레이트는 온스크린 키패트를 사용해 입력할 수 있습니다.

셔터 속도는 높은 프레임의 몰입형 콘텐츠 영상의 모습에 미치는 영향이 적습니다. 24p 또는 30p 촬영 시에는 180도 셔터를 선호하는 사람들이 많지만, 몰입형 촬영 시에는 노출을 보조하기 위해 셔터 속도를 사용하는 것이 더 일반적입니다.

저조도 촬영 시, 셔터는 270도까지 증가할 수 있으며, 예외의 경우엔 360도 까지도 증가합니다.

밝은 조건에서 촬영 시 셔터 속도는 11.2도까지 낮아질 수 있습니다. 하지만 밝은 조건에서는 보통 사용자가 선호하는 셔터 속도 설정에 맞추고 ND 필터를 통해 노출 조절을 하는 것이 더 일반적입니다.

Apple Immersive Video를 90fps와 같은 높은 프레임에서 촬영할 때는 플리커 현상이 발생할 수도 있다는 점을 유의해야 합니다. 60hz 환경에서 작업 시 270도 셔터 사용을 권장하고 50hz 환경에서는 324도 셔터를 권장합니다.

ISO 및 다이나믹 레인지

ISO는 카메라의 현재 ISO 설정, 또는 감광도를 나타냅니다. 이 표시를 눌러 ISO를 다양한 조명 조건에 맞게 조절할 수 있습니다. URSA Cine Immersive의 ISO 범위는 ISO 200 ~ 3200까지입니다. 최적의 ISO는 800입니다.



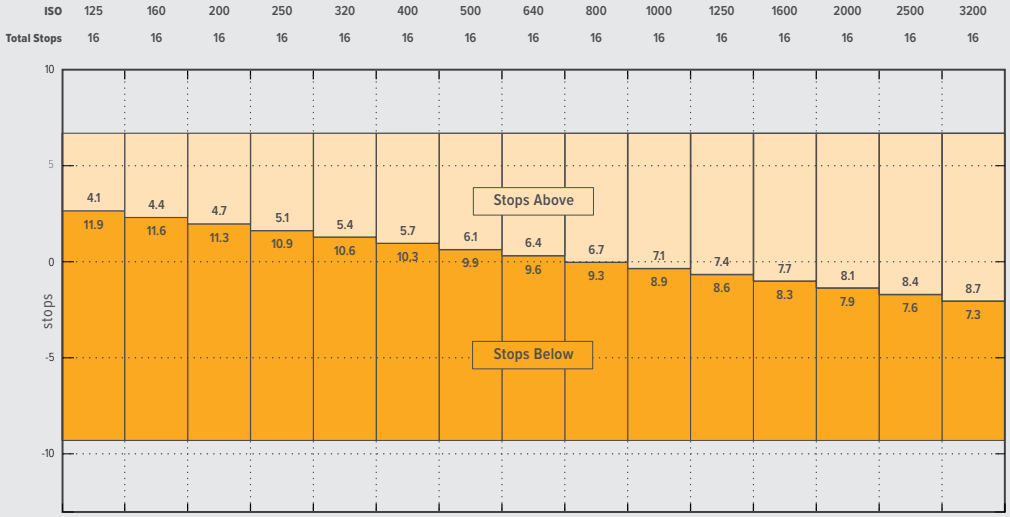
ISO는 'ISO' 표시를 눌러 설정할 수 있습니다.

숫자로 표시되는 프리셋 중 원하는 숫자를 눌러 풀 스탑 단위로 ISO를 사용할 수 있습니다. 프리셋 아래의 슬라이더를 사용하면 이러한 프리셋 사이에서 ISO 설정을 1/3 스톱 단위로 미세 조정할 수 있습니다.

촬영 환경에 따라 ISO를 좀 더 높거나 낮게 설정할 수 있습니다. 예를 들어 저조도 조건에서는 ISO 1600이 적절할 수 있지만, 화면에 노이즈 현상이 발생할 수 있습니다. 밝은 조명 조건에서는 ISO 200으로 촬영하면 풍부한 색상을 얻을 수 있습니다.

Blackmagic 카메라는 넓은 다이나믹 레인지를 제공해 이미지를 캡처할 때 뿐 아니라 DaVinci Resolve 색보정 작업 시에도 높은 유연성을 경험할 수 있습니다. 아래 표에는 URSA Cine Immersive 카메라에서 사용 가능한 다이나믹 레인지를 보여줍니다.

URSA Cine Immersive 다이나믹 레인지



ND

카메라의 ND 필터는 노출 조절의 보통 가장 흔한 방법으로 미리 정해진 노출 스탑 수치로 카메라 센서에 도달하는 광량을 줄일 수 있습니다.



URSA Cine Immersive의 'ND' 설정은 접이식 터치스크린 좌측 상단 위에 있는 버튼을 사용해 조절합니다. 'Clear', '2 Stops', '4 Stops', '6 Stops' 수치로 설정할 수 있습니다.

현재 설정된 'ND'는 터치스크린 좌측 상단에 있는 상태 표시를 통해 나타납니다. 'ND'가 활성화되면 표시는 지속적으로 나타납니다. 'ND'가 'Clear'로 설정되면 표시는 4초 후 사라집니다.



참고 ND 필터 표시 용어를 사용자 편의에 맞게 변경할 수 있습니다. 설정 메뉴를 사용해 ND 필터 번호, 스탑, 비율 중 선택하세요.

오프스피드 녹화

설정 메뉴의 프레젝트 프레임 레이트는 URSA Cine Immersive의 녹화 포맷 프레임 레이트이며 푸티지의 재생 속도를 결정합니다. 대개 프레임 레이트는 후반 제작 워크플로에서 사용하는 재생 속도 및 오디오 싱크와 딜리버리 요구조건에 맞게 설정됩니다.



URSA Cine Immersive의 초당 프레임 표시를 누르면 프레임 레이트 설정 창에 접속할 수 있습니다. Blackmagic URSA Cine Immersive는 초당 23.98, 24, 25, 29.97, 30, 50, 59.94, 60 및 90fps의 총 9가지의 프로젝트 프레임 레이트 설정 옵션을 제공합니다. 몰입형 촬영 시 프로젝트 프레임 레이트는 90fps로 설정해야 합니다.

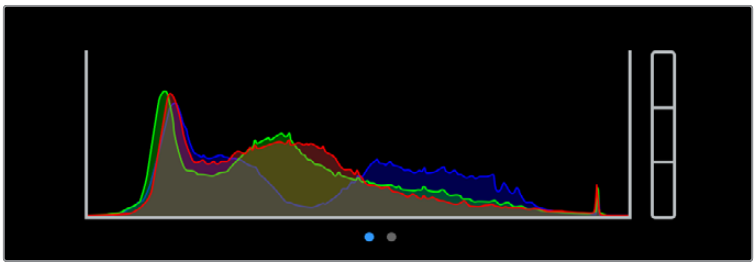
센서 프레임 레이트를 사용해 센서의 이미지 캡처 스피드를 결정합니다. URSA Cine Immersive의 프로젝트 프레임 레이트 및 센서 프레임 레이트는 일반적인 재생 속도에 맞게 기본 설정되어 있습니다. 하지만 카메라의 FPS 메뉴 우측 하단에 있는 '오프스피드 프레임 레이트' 스위치 아이콘을 눌러 센서 프레임 레이트를 개별적으로 설정할 수도 있습니다.

오프스피드 프레임 레이트 스위치가 활성화 시 수동으로 센서 프레임을 설정할 수 있습니다. 초당 90 프레임이 카메라의 최고 프레임 레이트이며 Apple Vision Pro용 촬영에 설정하는 90fps인 프로젝트 프레임 레이트와 일치한다는 것을 명심하세요. 그러나 30 또는 60프레임과 같은 낮은 프로젝트 프레임 레이트 선택하여 초당 90프레임 오프스피드로 촬영하면 아름다운 슬로우 모션 푸티지를 얻을 수 있습니다.

오프스피드 프레임 레이트 스위치가 켜진 상태에서, 스크린 좌측 상단에 있는 프레임 레이트 표시는 센서 프레임 레이트를 보여주고 그 다음으로 프로젝트 프레임 레이트를 보여줍니다.

히스토그램

URSA Cine Immersive의 터치스크린 좌측 하단에서 히스토그램을 확인할 수 있습니다. 히스토그램은 이미지의 밝기 분포도를 나타내는 그래프입니다. 좌측에는 섀도우, 가운데는 미드톤, 우측에는 하이라이트가 나타납니다. 그래프의 높이는 해당 밝기 레벨이 이미지에 얼마나 분포되어 있는지를 나타냅니다.



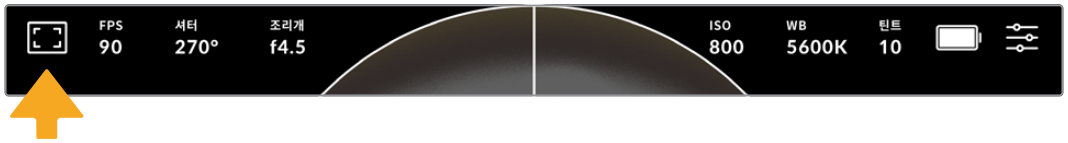
히스토그램은 한눈에 클립 톤 분포도를 볼 수 있도록 제공합니다. 그래프 좌측에 있는 이미지는 노출이 부족할 가능성이 높습니다. 이와 유사하게, 그래프 우측에 있는 이미지는 과다 노출일 가능성이 높습니다. 히스토그램 항시 이미지와 같이 맥락 안에서 해석되어야 하지만 일반적으로 적정 노출된 이미지는 균형 잡힌 그래프로 가운데에 표시됩니다. 노출이 증가하면 히스토그램이 우측으로, 노출이 감소하면 좌측으로 이동합니다.

개별 컬러 채널이 클리핑되면 히스토그램 우측에 있는 표시기로 알 수 있습니다. 일반적으로 어느 채널도 클리핑 되지 않도록 해야 합니다. 그러나 이미지의 미드톤과 섀도우를 보존하기 위함이면 이미지 클리핑이 허용되는 경우가 있습니다.

ETTR(Expose to the Right)은 클리핑되는 컬러 채널 없이 히스토그램을 가능한한 최대한 우측으로 밀어 의도적으로 이미지를 과다 노출하는 기법입니다. 이 기법은 신호를 최대화하고 노이즈를 최소화합니다. 후반 제작에서 노출을 낮추기만 한다면 이 기법은 깨끗한 이미지를 성취하기 위한 훌륭한 방법입니다.

LCD 모니터 옵션

URSA Cine Immersive의 LCD 터치스크린 좌측 상단에 있는 모니터 아이콘을 눌러 LCD 모니터링 설정에 접속할 수 있습니다. 이 메뉴는 모니터 밝기와 같은 유용한 옵션 및 강력한 노출 도구들도 포함합니다.



지브라

'지브라'를 통해 지브라 레벨을 설정할 수 있을 뿐만 아니라, 터치스크린에 나타나는 지브라를 켜고 끌 수 있습니다. 지브라 기능을 사용할 경우, 영상에서 설정된 노출 레벨을 초과하는 부분에 대각선 줄무늬가 나타납니다. 과다 노출 부분은 설정한 퍼센트 수치를 바탕으로 표시됩니다. 이 설정은 과다 노출의 가능성이 있는 부분을 식별하기 위한 것이기 때문에 100%보다는 95%와 같은 수치로 설정하는 것이 일반적입니다. 두 번째 지브라 설정을 통해 중간 회색 지브라 영역을 켜고 끄거나 '중간 회색 +1' 스탑을 선택할 수 있습니다.

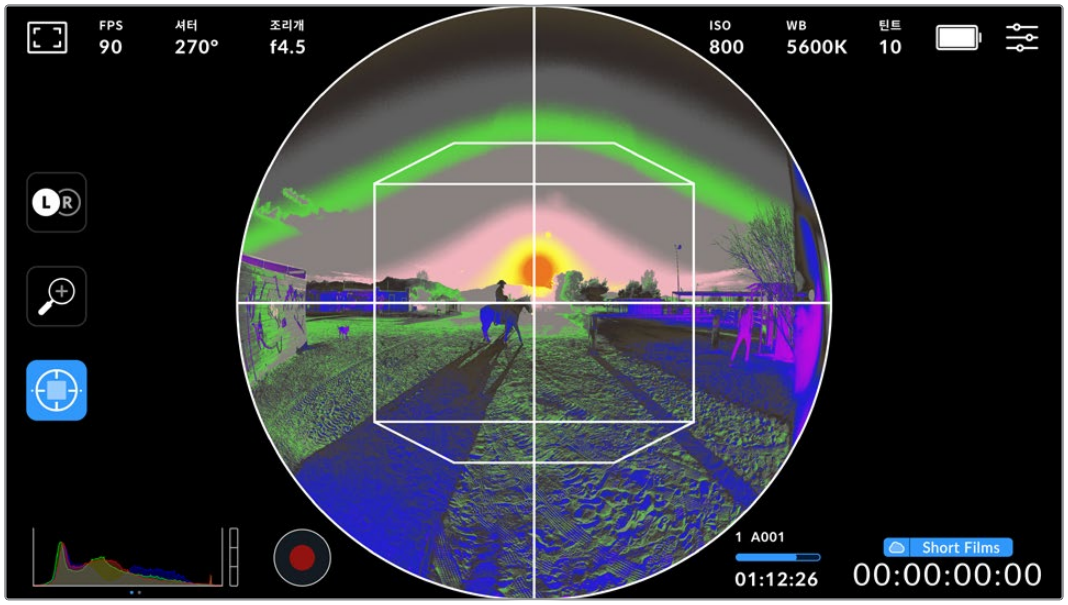
그리드

그리드 설정을 통해 URSA Cine Immersive의 모든 출력에 표시될 오버레이 설정뿐만 아니라, LCD 터치스크린에 나타나는 3분할 선 및 수평계, 십자선, 중심점을 활성화할 수 있습니다. 몰입형 콘텐츠 촬영 시 수평계는 매우 유용합니다. 수평계는 카메라가 완벽하게 수평일 경우 이미지 중앙에 일치된 이중 십자선을 표시하며, 기울어진 경우 어긋난 십자선을 표시합니다.

폴스 컬러

폴스 컬러 기능은 이미지 위에 노출값을 나타내는 여러 가지 색상을 표시해 이미지 속 서로 다른 요소의 노출값을 확인할 수 있는 기능입니다. 예를 들어, 분홍색은 밝은 피부톤에 최적인 노출을, 녹색은 어두운 피부톤에 최적인 노출을 나타냅니다. 인물을 촬영할 때 폴스 컬러의 분홍색 또는 녹색 부분을 모니터링하여 피부톤을 위한 최적의 노출값을 지속적으로 유지할 수 있습니다.

마찬가지로, 이미지 속 피사체가 노란색에서 빨간색으로 변한다면 이는 과다 노출을 의미합니다. 히스토그램과 유사하게, 이 도구를 사용해 아주 신속하고 직관적이게 이미지의 색상 분포도를 이해할 수 있습니다.



녹화

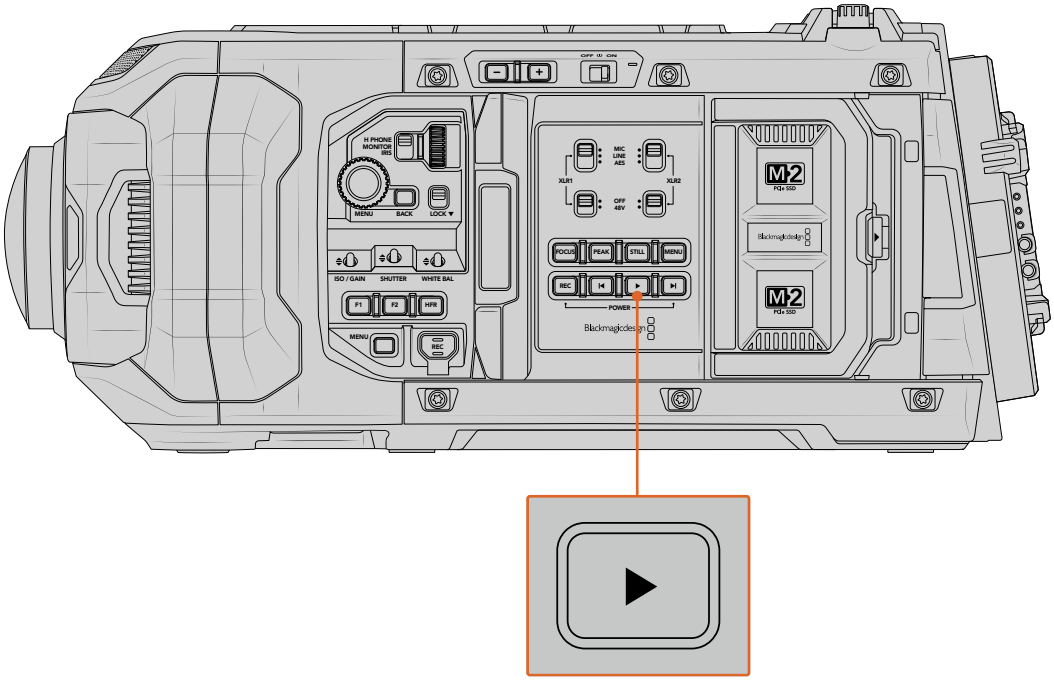
카메라의 빨간색 녹화 버튼 중 하나를 누르면 녹화가 시작됩니다. 녹화 버튼은 전면부 제어 패널과 접이식 터치스크린 뒤쪽의 내부 제어 패널, 보조 스테이션 제어 패널, 그리고 2개의 LCD 터치스크린에 배치되어 있습니다. 녹화 버튼을 다시 누르면 녹화가 중단됩니다.

이 저장 미디어 표시는 잔여 녹화 시간을 나타냅니다. 시간은 선택된 프레임 레이트에 기반하여 시:분:초 순으로 표시합니다. 녹화 시간 위에 표시되는 미디어 바는 현재 상태에 따라 파란색 및 흰색, 또는 빨간색으로 표시되며, 미디어의 사용 공간을 표시합니다.



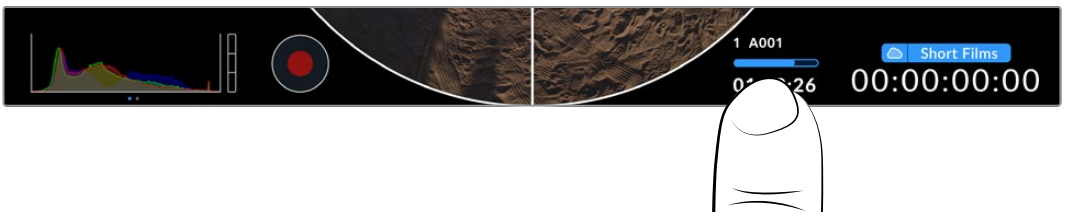
재생

클립을 촬영한 후에, 트랜스포트 컨트롤 버튼을 사용하여 녹화 영상을 재생할 수 있습니다. 녹화한 최신 클립을 재생하기 위해, 카메라 좌측 접이식 터치스크린 뒤쪽에 위치한 재생 버튼을 누르세요. 다른 클립을 검색하기 위해 뒤로 가기 버튼과 앞으로 가기 버튼을 사용할 수 있습니다. 재생 모드에서 나와 녹화를 지속하기 위한 카메라 준비 모드로 전환하려면 녹화 버튼을 한 번 누르세요.

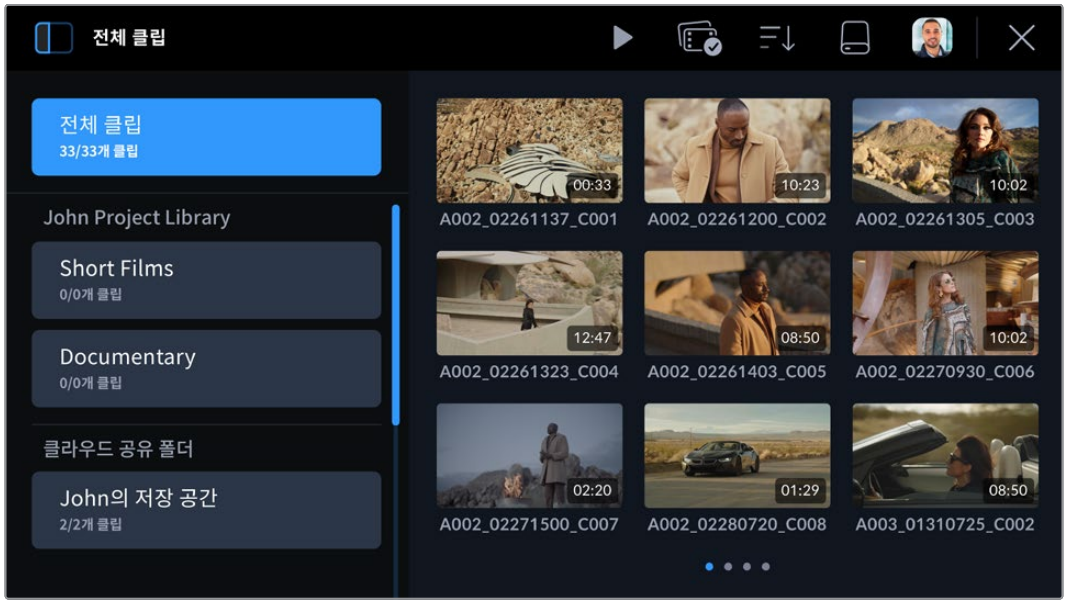


앞으로 가기 버튼과 뒤로 가기 버튼을 눌러 클립의 시작 및 끝부분으로 이동할 수 있습니다. 뒤로 가기 버튼을 누르면 현재 재생 중인 클립의 시작 부분으로 넘어가며, 이 버튼을 두 번 누르면 이전 클립의 시작 부분으로 되돌아갑니다. 앞으로 가기 버튼 또는 뒤로 가기 버튼을 누르고 있으면 2배속으로 정방향/역방향으로 재생됩니다. 앞/뒤로 셔틀 중인 상태에서 앞으로 가기 또는 뒤로 가기 버튼을 2번 누르면 4배속으로, 3번 누르면 8배속으로, 4번 누르면 16배속으로 재생됩니다.

촬영한 푸티지를 더욱 자세히 검토할 필요가 있다면, 미디어 풀을 사용하시길 권장합니다. 미디어 풀을 열려면 카메라 터치스크린 하단에 있는 저장 미디어 표시를 누르세요.



메인 페이지 브라우저는 사용자의 카메라에 연결된 모든 미디어에 저장된 녹화 클립의 썸네일을 모두 표시합니다. 썸네일은 세 줄로 표시되며 클립이 추가됨에 따라 터치스크린에서 쓸어 넘겨 다음 페이지의 썸네일을 확인할 수 있습니다. 디스플레이 하단에 있는 페이지 표시는 볼 수 있는 클립의 페이지 수를 나타냅니다.



URSA Cine Immersive를 Blackmagic Cloud 계정에 연결하여 푸티지를 곧바로 DaVinci Resolve 프로젝트에 전송할 수 있습니다.

언제든 미디어 풀을 나가려면 우측 상단에 있는 'X'를 누르거나 카메라 녹화 버튼을 한 번 누르세요.

클립 전송하기

카메라 스토리지의 클립을 전송할 수 있는 방법은 아주 다양합니다. 예를 들면, URSA Cine Immersive를 컴퓨터에 꽂거나 10G 이더넷 케이블을 통해 네트워크에 연결할 수 있으며, 이더넷을 통해 카메라 스토리지의 클립을 복사할 수도 있습니다. 또한 미디어 모듈을 네트워크나 컴퓨터에 연결된 Blackmagic Media Dock에 로딩할 수 있으며, 이를 통해 최대 3개의 미디어 모듈로 동시에 작업할 수 있습니다.

이더넷을 통해 컴퓨터에 연결할 경우, 컴퓨터는 Blackmagic Cloud Store가 연결되었을 때와 동일한 방식으로 Blackmagic Media Module을 인식합니다.

사용자 카메라는 DHCP로 기본 설정되어 있기 때문에 컴퓨터 또는 네트워크가 카메라를 인식하여 IP 주소를 자동으로 지정합니다.

파일 공유 활성화하기

카메라를 처음으로 셋업할 경우, 컴퓨터나 이더넷을 통한 네트워크에 연결되어 있다면 SMB 파일 공유 기능이 비활성화되도록 기본 설정됩니다. 이는 네트워크에서 다른 사용자들과 파일을 곧바로 공유하기 위한 Blackmagic Cloud Store와는 달리, 주요 촬영을 마친 뒤 마스터 카메라 파일에 접속할 수 있는 접속 권한을 즉시 허용하고 싶지 않을 수 있기 때문입니다. 이런 이유로 SMB 파일 공유 기능은 사용자가 직접 활성화할 수 있도록 설정되어 있습니다.

1. 파일 공유 기능을 활성화하려면, Blackmagic Camera Setup을 컴퓨터에 다운로드하여 설치하세요. 이 셋업 유틸리티는 카메라를 업데이트하고 관리 설정을 변경할 수 있는 관리용 소프트웨어입니다. Blackmagic Camera Setup은 저희 웹사이트에서 다운로드하실 수 있습니다. https://www.blackmagicdesign.com/kr/support/family/professional-cameras
2. 이 셋업 유틸리티를 다운로드 및 설치하고 나면 USB를 통해 카메라를 컴퓨터에 연결한 다음 Blackmagic Camera Setup을 실행하세요. 상단 패널에 있는 USB 포트 중 카메라 뒷면 패널에 가까운 포트를 통해 연결하세요.

3 설정을 열려면 URSA Cine Immersive의 이미지나 설정 아이콘을 클릭하세요.

4 'Network Access' 설정으로 스크롤하여 내린 다음 'File Sharing (SMB)'을 활성화하세요. 그런 다음, 우측 하단에 있는 'Save' 버튼을 눌러 설정을 적용하세요.

5 이제 그 아래쪽의 URL 설정으로 이동해 URL 아이콘을 클릭하면 주소가 자동으로 생성됩니다. URL 주소에 카메라 이름이 로컬 네트워크 위치로 표시됩니다.

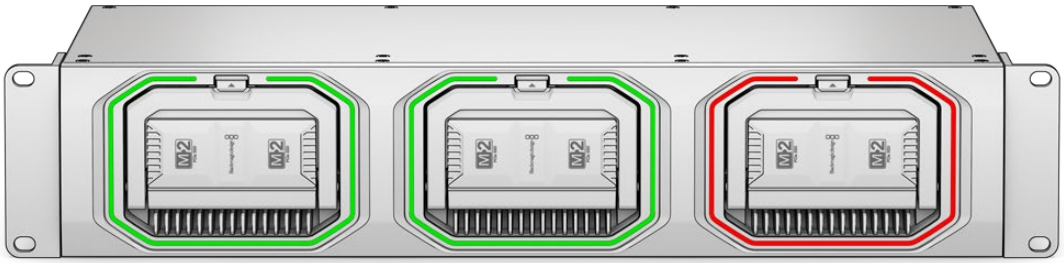
6 'Save' 버튼을 클릭하세요.

곧장 클립으로 이동한 다음 카메라의 URL 이름을 클릭하세요. 그러면 비밀번호 혹은 게스트 접속 여부를 묻는 창이 열립니다. 'Guest' 버튼을 클릭한 다음 'Connect'를 클릭하세요.

7 이제 컴퓨터에서 자동으로 Mac의 'Finder' 또는 Windows의 '파일 탐색기'에 있는 미디어 모듈을 검색합니다. 이 유틸리티에서 이더넷 설정을 기억하기 때문에 처음 셋업할 때 한 번만 설정하면 됩니다. 이후 이더넷을 통해 카메라를 연결하면 다른 설정을 조정하지 않아도 곧바로 미디어 모듈에 접속할 수 있습니다.

Blackmagic Media Dock 사용하기

Blackmagic Media Module은 신속하게 탈부착할 수 있도록 설계되어 외부 스토리지로 클립을 전송하거나 빈 모듈로 교체해 녹화를 이어갈 수 있습니다. Blackmagic Media Dock을 사용해 카메라 없이 Media Module에서 푸티지에 접속할 수 있습니다.



요약

Apple Immersive Video는 시각적 스토리텔링에 대한 새로운 접근법을 소개하며 기존 2D 시네마에 대한 대안을 제시합니다. 이것은 기존 포맷을 대체하려는 것이 아닌 프로젝트 요구에 따라 선택할 수 있는 추가적인 포맷을 제공합니다. 다른 모든 포맷처럼 이야기에 따라 창의적 선택은 달라집니다.

새로운 포맷들이 대체로 복잡한 워크플로를 요구하는 반면, URSA Cine Immersive는 Apple Immersive Video의 캡처 과정을 간소화하도록 설계되었습니다. 이는 몰입형 컨텐츠에 필요한 간단한 프로덕션 경로를 제공하며 시각 효과 및 다른 작업에도 사용될 수 있습니다.

이 가이드 URSA Cine Immersive의 주요 기능 및 워크플로를 간략히 설명하며, 사용자의 출발점이 되도록 의도하였습니다. 모든 카메라 시스템이 그렇듯 특정 프로덕션 요구에 맞는 심도있는 탐구와 검증을 통해 최고의 결과물을 얻을 수 있을 것입니다.

DaVinci Resolve Studio



DaVinci Resolve Studio는 편집/색보정/시각 효과/모션 그래픽/오디오 후반 제작 기능이 매끄럽게 통합된 종합적인 후반 제작 소프트웨어입니다. 초고화질의 스테레오스코픽 비디오와 90fps의 프레임 레이트, 공간 오디오 믹싱 등 Blackmagic RAW Immersive 포맷의 독특한 특성에 맞게 개발된 워크플로를 지원합니다. DaVinci Resolve Studio는 몰입형 콘텐츠 제작자를 위한 영상 제작 및 모니터링, Apple Immersive Video 딜리버리 기능까지 완벽히 갖춘 강력한 도구입니다.

워크스테이션 권장 사항

하드웨어 권장 사항

- Mac Studio M3 Ultra Desktop
- 고주사율 모니터(최소 60Hz)

시스템 최소 요구 사항

- macOS 15 또는 이후 버전
- 64GB 시스템 메모리
- 모니터링을 위한 Blackmagic Design Desktop Video 12.9 또는 이후 버전
- Apple Silicon 기반 컴퓨터 또는 Metal 지원 GPU

몰입형 영상 모니터링 요구 사항

- Apple Vision Pro
- 고속 무선 네트워크(최소 Wi-Fi 5, Wi-Fi 6 권장)

프로젝트 셋업

Blackmagic URSA Cine Immersive의 RAW 미디어를 입수하기 전에, 우선 DaVinci Resolve의 프로젝트 설정으로 가서 미디어가 제대로 디베이어되었는지, 최적의 영상 출력으로 처리되었는지 확인하여 수월한 편집 작업을 수행할 수 있습니다.

마스터 설정

프로젝트 셋업 시 Apple Vision Pro용 콘텐츠를 생성 여부를 설정하는 것이 중요합니다. 그래야 프로젝트 준비, 파일 취합 및 딜리버리 작업 시 필요한 모든 올바른 설정 및 워크플로 옵션이 나타납니다.

1. DaVinci Resolve Studio를 실행하세요.
2. 'Project Manager' 창 우측 하단에 있는 'New Project'를 클릭하세요.
3. 프로젝트 이름을 입력하고 'Media Location'(필요할 경우)을 설정한 다음 'Create'를 클릭하세요.
4. DaVinci Resolve Studio 우측 하단에 있는 톱니바퀴 아이콘을 클릭해 'Project Settings'를 여세요.

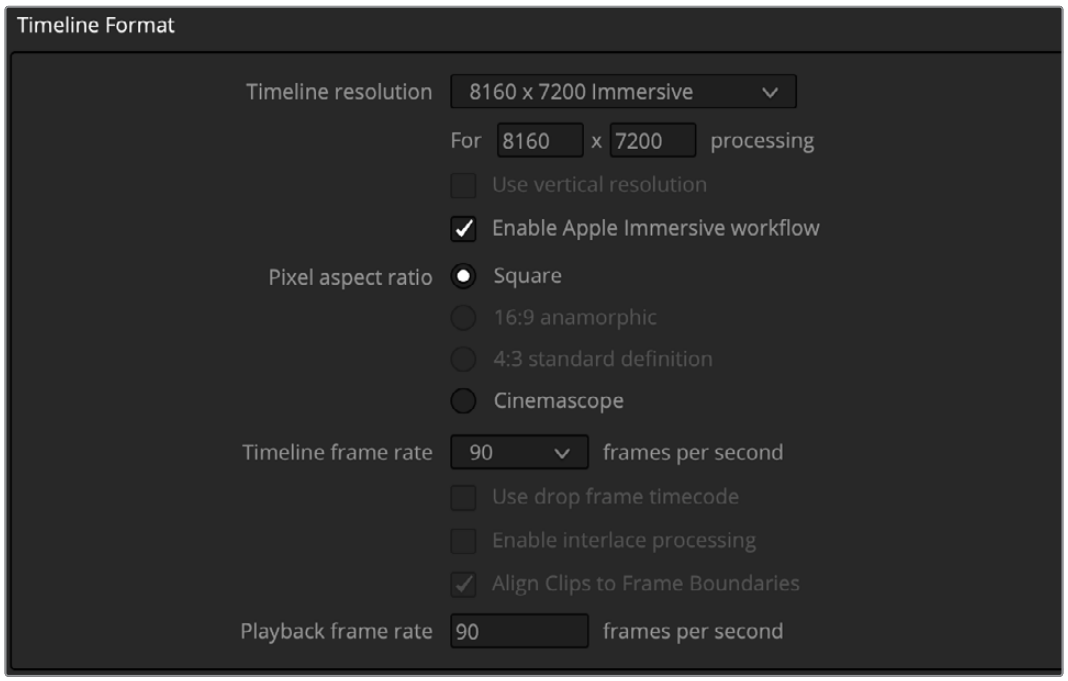
   먼저 프로젝트의 해상도와 프레임 레이트를 설정하세요.
5. 사이드바에 있는 'Master Settings'를 선택하세요.
6. 'Timeline Format' 섹션에서 'Enable Apple Immersive workflow'를 선택하세요.



   이를 통해 DaVinci Resolve의 여러 설정 사항이 변경되어 몰입형 워크플로에 가장 적합한 상태로 바뀝니다. 몰입형 프로젝트 생성 시 항상 이 설정을 먼저 선택하세요.
7. 'Timeline resolution'을 '8160 x 7200 Immersive'로 변경해 클립이 원본 센서 해상도로 처리될 수 있도록 하세요.

   프로세싱 성능이 부족할 경우, 절반에 해당하는 '4080 x 3600 Immersive'를 선택하면 클립이 신속하게 재생됩니다.
8. 'Timeline frame rate'는 반드시 '90 frames per second'로 설정하세요. 이 바로미터는 'Enable Apple Immersive workflow' 활성화 시 자동으로 설정됩니다.
9. 'Playback frame rate'를 '90 frames per second'로 설정하면 몰입형 비디오 콘텐츠가 가장 정확하게 재현됩니다.

   프로세싱 성능이 부족하거나 저주사율 모니터를 사용할 경우, 프레임 레이트를 낮춰 클립이 안정적으로 재생되도록 하세요. 원본 프레임 레이트보다 낮은 30fps 또는 60fps의 프레임 레이트를 선택하는 것이 좋습니다.



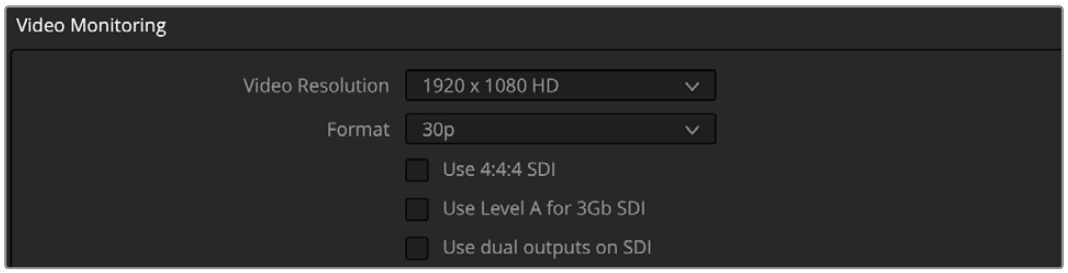
비디오 모니터링

Decklink나 UltraStudio 등의 캡처/재생 장치를 통해 외부 모니터를 사용할 경우, 모니터 사양을 바탕으로 비디오 모니터링 설정을 변경하세요.

1 'Video Monitoring' 항목에서 모니터 해상도에 맞는 비디오 해상도를 설정하세요. 예를 들어, '1920 x 1080 HD'나 '3840 x 2160 Ultra HD'로 설정할 수 있습니다.

몰입형 비디오의 원본 프레임 레이트가 90fps일 수 있지만, 대부분의 방송용 모니터에서는 해당 프레임 레이트가 지원되지 않습니다.

2 'Format' 항목을 모니터의 프레임 레이트에 맞춰 설정하세요. 예를 들어, '30p'로 설정할 수 있습니다.



영상 스케일링

'Image Scaling' 설정을 통해 몰입형 미디어의 해상도 및 종횡비가 뷰어에 올바르게 나타나도록 할 수 있습니다. 이러한 파라미터는 'Master Settings'에서 'Enable Apple Immersive workflow' 옵션을 활성화하면 자동으로 설정되지만, 해당 항목들이 올바르게 설정되어 있는지 확인해야 합니다.

1 'Project Settings' 사이드바에서 'Image Scaling'을 선택하세요.

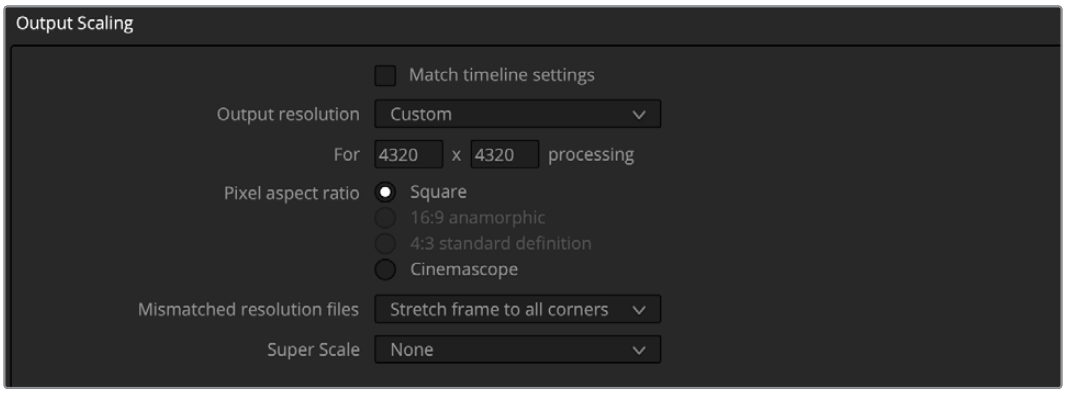
2 'Output Scaling' 항목에서 'Match timeline settings' 선택이 해제되어 있는지 확인하세요.

3 ‘Output resolution’ 항목이 ‘Custom’ 및 ‘4320 x 4320’으로 설정되어 있는지 확인하세요.

소형 디스플레이에서의 빠른 클립 재생을 원할 경우, 절반에 해당하는 2160 x 2160 해상도로 설정하세요.

이러한 정사각형 종횡비 해상도는 Apple Immersive Video 콘텐츠를 DaVinci Resolve Studio 내에서 재생하는 경우나 와이파이 또는 개발자 스트랩을 통해 Apple Vision Pro로 스트리밍할 때 필요합니다.

4 ‘Mismatched resolution files’ 항목이 ‘Stretch frame to all corners’로 설정되어 있는지 확인하세요.



색상 관리

최대 출력 니트 휘도의 커스텀 제어 필요성이 대두됨에 따라 색상 관리 시 컬러 페이지에서 색공간 변환(Color Space Transform, CST) 노드 사용이 권장됩니다. ‘Project Settings’에 포함된 색상 관리 설정의 목표는 컬러 페이지의 타임라인 색공간이나 작업 중인 색공간을 지정하고 렌더링된 파일과 관련된 색공간 메타데이터를 설정하는 것입니다.

1 ‘Project Settings’ 사이드바에서 ‘Color Management’를 선택하세요.

2 ‘Color science’는 ‘DaVinci YRGB’로 남겨두세요.

3 ‘Timeline color space’는 ‘DaVinciWG/Intermediate’로 설정하세요.

이는 HDR 워크플로를 지원해 향후에도 사용 가능한 폭넓은 색공간으로, Blackmagic URSA Cine Immersive 미디어 및 Apple Vision Pro 영상에 사용하기 적합합니다.

4 ‘Output color space’를 ‘P3-D65 ST2084 1000 nit’로 설정하세요.

이 색공간은 Mac Studio Desktop에서 DaVinci Resolve 뷰어를 모니터링하기 적절한 색영역을 제공하는데, 보통 Display P3가 여기에 해당합니다. 색보정용 모니터나 다른 Apple ICC 프로필을 사용할 경우, 현재 사용 중인 디스플레이 색상 표준에 매칭되도록 출력 색공간이 변경됩니다.

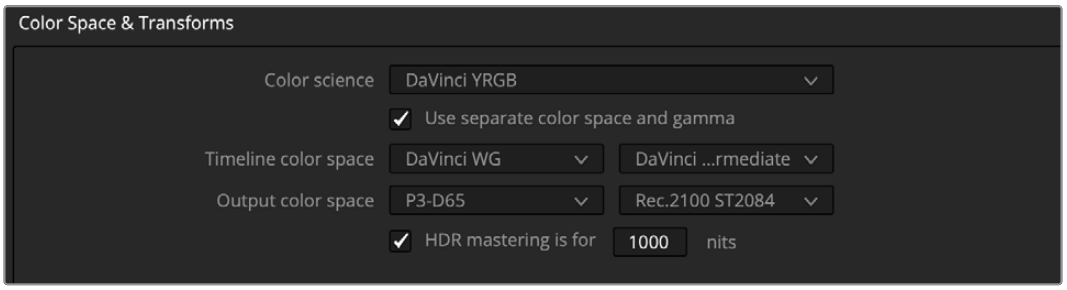
하지만 감마(전송 기능)가 ICC 프로필과 제대로 매칭되도록 변경해야 합니다.

5 색공간 설정 위쪽에 있는 ‘Use separate color space and gamma’를 클릭하세요.

그러면 색공간 및 감마 드롭다운 필드가 분리되어 표시됩니다.

6 오른쪽 칼럼의 출력 감마를 “Rec.2100 ST2084’로 설정하세요.

7 ‘HDR mastering is for’ 항목을 ‘1000’ nits로 설정하세요.



현재 설정 사항을 프리셋으로 저장하여 다음번 Apple Immersive 프로젝트의 설정 시간을 단축할 수 있습니다.

8 이 경우, 'Project Settings' 창 우측 상단에 있는 세 점 옵션 메뉴로 이동해 ‘Save Current Settings as Preset’을 선택하세요.

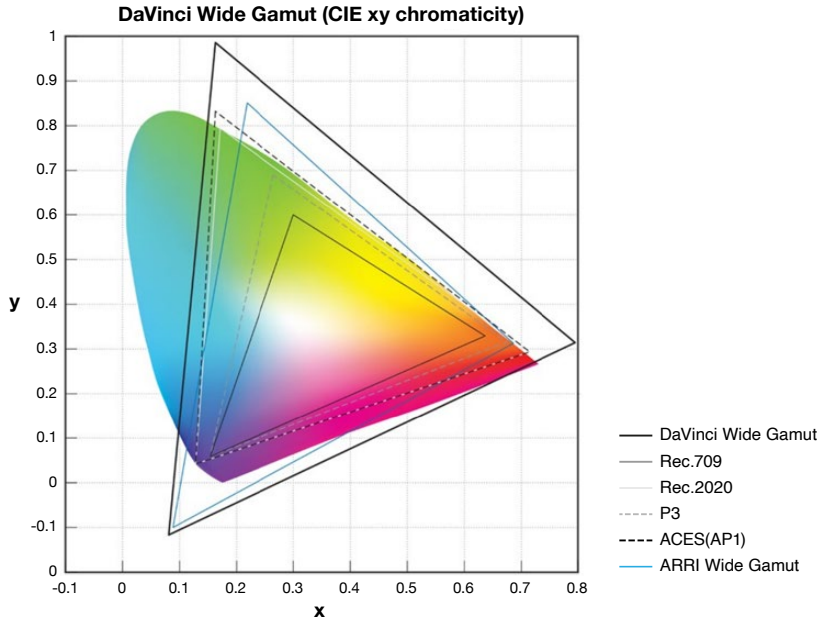
9 프리셋 이름을 입력한 다음 'OK'를 클릭하세요.

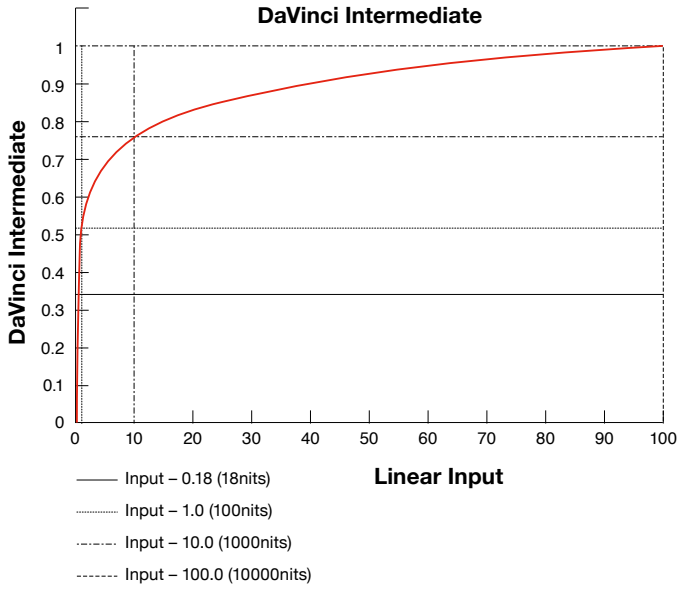
다음번에 몰입형 미디어가 포함된 프로젝트 설정을 신속하게 설정하려면, 'Project Settings' 창을 열고 옵션 메뉴를 클릭한 다음 목록에 저장된 프리셋을 선택하세요.

10 ‘Save’를 클릭하면 'Project Settings' 창이 닫힙니다.

DaVinci Wide Gamut에 관하여

DaVinci Wide Gamut은 DaVinci Resolve를 위해 개발된 Blackmagic Design의 첨단 내부 색공간으로, 아주 폭넓고 정확한 색보정 및 이미지 프로세싱 환경을 제공합니다. DaVinci Intermediate 감마와 함께 통합된 작업 공간을 형성해 다양한 카메라 포맷을 다루는 색상 관리가 간소화되어 입력부터 출력까지 일정한 톤과 채도 매핑이 유지됩니다. HDR 및 몰입형 미디어 작업에 최적인 이 폭넓은 색영역을 통해 컬러리스트가 고충실도 딜리버리 포맷을 마스터링할 때 밴딩 현상이나 색조 클리핑 없이 뉘앙스를 조절할 수 있습니다.





Apple Immersive 프로세싱

Project Settings 사이드바에 있는 Apple Immersive 섹션에서 커스텀 몰입형 렌즈 프로필 데이터(Immersive Lens Profile Data, ILPD)를 설정해 몰입형 콘텐츠가 보정 및 투사되는 방식을 수정할 수 있습니다.

1 'Project Settings' 사이드바에서 'Apple Immersive'를 선택하세요.
2 'Open Calibration Folder'를 클릭하세요.
3 커스텀 ILPD 파일을 추가 또는 업데이트하세요.
4 'Update Lists'를 클릭해 미디어 풀에서 접속할 수 있는 ILPD 콘텐츠를 새로 고침하세요.
5 'Save'를 클릭하면 'Project Settings' 창이 닫힙니다.
6 미디어 풀에서 몰입형 클립을 우클릭하세요.
7 메뉴에서 'Apple Immersive calibration'으로 이동한 다음 소스 ILPD('No Apple Immersive calibration file selected')나 'Custom ILPD' 또는 'Gap ILPD' 중 하나를 선택하세요.

기본적으로 커스텀 ILPD Calibration 폴더는 기본 설정된 LUT 폴더와 함께 ProIM 폴더에 저장되어 있습니다.

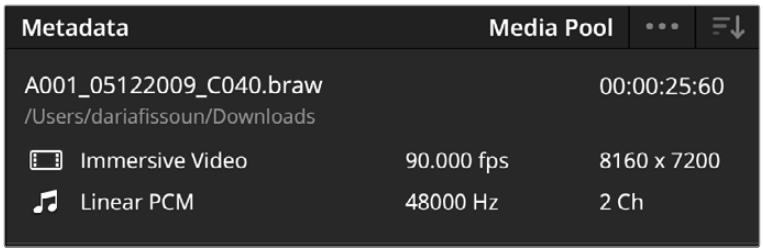
영상 입수

몰입형 콘텐츠 원본을 입수할 경우, 임포트된 .braw 클립에 양안 영상 및 관련된 모든 몰입형 메타데이터가 포함되어 있습니다.

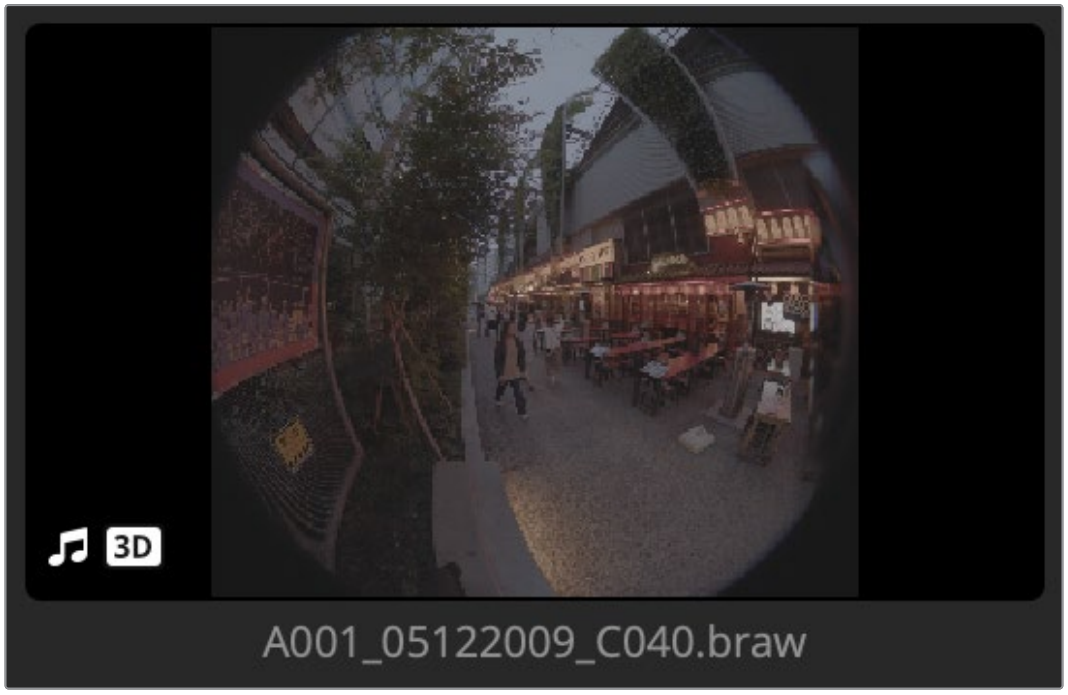
1 미디어 페이지로 이동하세요.
2 'Media Storage' 패널에서 몰입형 비디오 콘텐츠가 있는 위치 경로를 따라 이동하세요.
3 미디어를 아래 미디어 풀로 드래그 앤 드롭하세요.

정보 원본 폴더의 구조를 유지하려면, 'Media Storage' 패널의 폴더를 미디어 풀 사이드바로 드래그하세요. 마스터 빈이나 생성해 둔 다른 빈으로 드래그하면 폴더 및 하위 폴더가 그대로 유지됩니다.

Apple Immersive 비디오 클립을 선택하면 ‘Metadata’ 패널 헤더의 코덱 파라미터에 ‘Immersive Video’라고 표시되며, 원본 프레임 레이트와 해상도도 함께 나타납니다.



이뿐만 아니라 비디오 썸네일 좌측 하단에 3D 아이콘이 나타나 3D 클립이라는 사실을 알려줍니다.



참고 몰입형 미디어를 미디어 풀에 처음으로 드래그 앤 드롭할 경우, ‘Keep Settings’ 또는 ‘Change Settings’ 선택 여부를 묻는 다이얼로그 창이 나타납니다. ‘Change Settings’를 선택하면 프로젝트 설정이 자동으로 소스 미디어의 프레임 레이트와 해상도로 변경됩니다.

편집

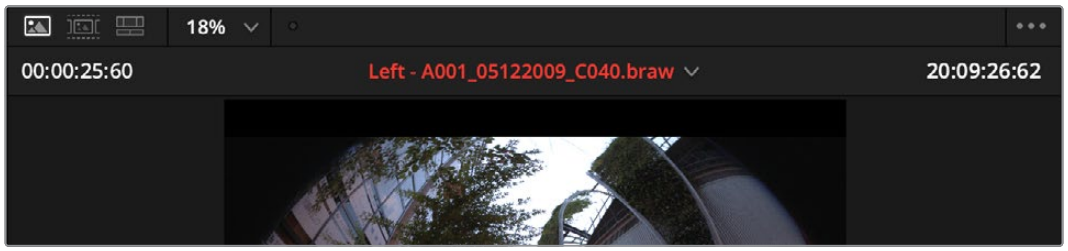
몰입형 콘텐츠 작업 시 편집 도구 및 동작이 대부분 기존의 2D 비디오 포맷을 작업하던 경우와 동일하게 유지됩니다. 다음에 이어지는 내용에서는 몰입형 비디오 워크플로에 특화된 편집 기능 또는 동작에 대해 설명합니다.

취합

타임라인을 취합하는 동안 클립을 미리보기할 경우, 소스 뷰어에 나타나는 영상을 좌안 또는 우안으로 변경할 수 있습니다. 그렇지 않을 경우, 타임라인에 있는 컷을 통합하는 과정이 표준 NLE 워크플로와 동일하게 유지됩니다.

1 편집 페이지로 이동하세요.

2 클립을 더블 클릭하면 해당 클립이 좌측의 소스 뷰어로 로딩됩니다. 뷰어 상단의 라벨에 현재 보고 있는 영상이 좌안용인지 우안용인지 표시됩니다.



3 뷰어에 표시되는 좌/우안 영상을 변경하려면, 미디어 풀의 소스 클립을 우클릭한 다음 'Stereoscopic 3D' > 'Stereoscopic 3D Mode' > 'Left Eye/Right Eye/Both Eyes'를 선택하세요.

> **정보** 소스 뷰어 옵션에서 'Live Media Preview' 항목을 활성화한 경우, 미디어 풀의 썸네일 위로 마우스를 가져가 소스 뷰어의 클립을 훑어보세요.

4 'I' 및 'O' 단축키를 사용해 소스 뷰어의 범위를 지정하세요.

5 도구바에서 'Insert Clip'을 누르거나 F9 단축키를 사용하여 타임라인에 클립을 삽입하세요.

> **정보** 미디어 풀의 클립을 타임라인 패널로 드래그할 수도 있습니다.

6 'Selection' 또는 'Trim Edit' 모드에서 마우스를 사용해 타임라인의 클립을 트리밍하세요.

배경막

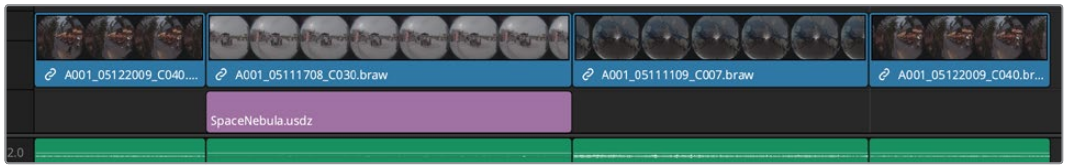
'Project Settings'에서 'Enable Apple Immersive workflow'를 선택한 경우, 'BD'로 표시되는 배경막(Backdrops) 트랙이 'Video 1' 트랙 밑에 자동으로 나타납니다.

배경막이란 시청자가 몰입형 비디오 콘텐츠 둘레를 벗어나는 부분을 볼 때 나타나는 영역을 의미하며, 보통 180~210도의 시야각으로 제한되어 있습니다.

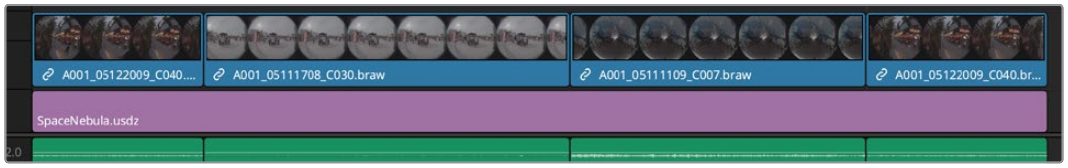
배경막은 보통 검은색으로 기본 설정되어 있지만, 배경막 트랙을 사용하여 .usdz 포맷의 커스텀 배경 이미지를 삽입할 수 있습니다.

1 .usdz 배경막 파일을 배경막 트랙으로 드래그하면 타임라인에 추가됩니다.

배경막 이미지의 지속 길이는 비디오 트랙의 클립 길이와 일치해야 합니다. 하나의 배경막을 한 개의 클립, 여러 개의 클립 또는 전체 타임라인 길이에 맞춰 연장할 수 있지만, ‘L-컷’ 방식처럼 몰입형 .braw 클립 중간에 잘리게 할 수는 없습니다.



한 개의 클립 - 올바른 예시 ✓

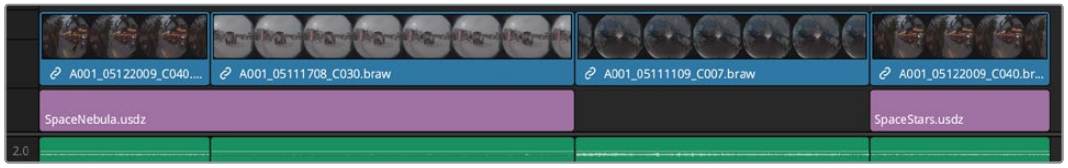


여러 개의 클립 - 올바른 예시 ✓



L-컷 - 잘못된 예시 ✗

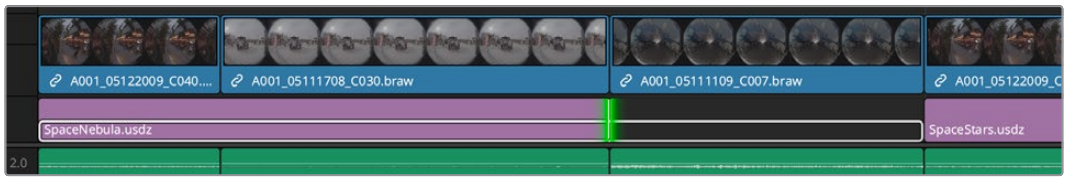
하나의 타임라인 안에 여러 개의 배경막 파일을 사용할 수 있고, 기본 설정된 검은색 배경막에 아무런 이미지를 추가하지 않을 수도 있습니다.



각기 다른 이미지와 빈 부분 - 올바른 예시 ✓

2 표준 트리밍 컨트롤을 사용해 배경막 파일의 길이를 조절하세요.

‘Selection 또는 Trim Edit’ 모드에서 .usdz 파일의 가장자리를 드래그해 길이를 조절하세요.



쉼표(,)나 마침표(.) 명령어를 사용해 한 번에 한 프레임씩 트림 가장자리를 미세 조절할 수도 있습니다.

정보 배경막이 항상 몰입형 비디오 클립과 일치하도록 하려면, 편집 페이지의 도구바에서 ‘Snap’ 모드를 활성화하세요.

뷰어 사용하기

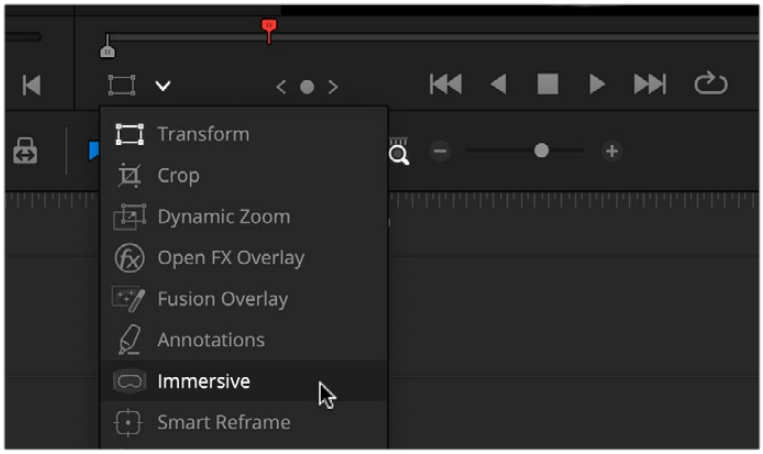
몰입형 클립은 DaVinci Resolve 페이지 뷰어의 구형 '렌즈 공간'에 나타나도록 기본 설정되어 있습니다. 이미지 투사에 사용되는 시각적 렌즈 보정은 카메라에서 시작되며, 이는 각 클립에 따라 달라집니다.



보기 모드를 'LatLong'이나 'Viewport'로 변경할 수 있는 옵션도 제공됩니다.

1 타임라인 뷰어 좌측 하단에서 온스크린 컨트롤 버튼 옆에 있는 드롭다운 화살표를 클릭하세요.

2 'Immersive'를 선택하세요.



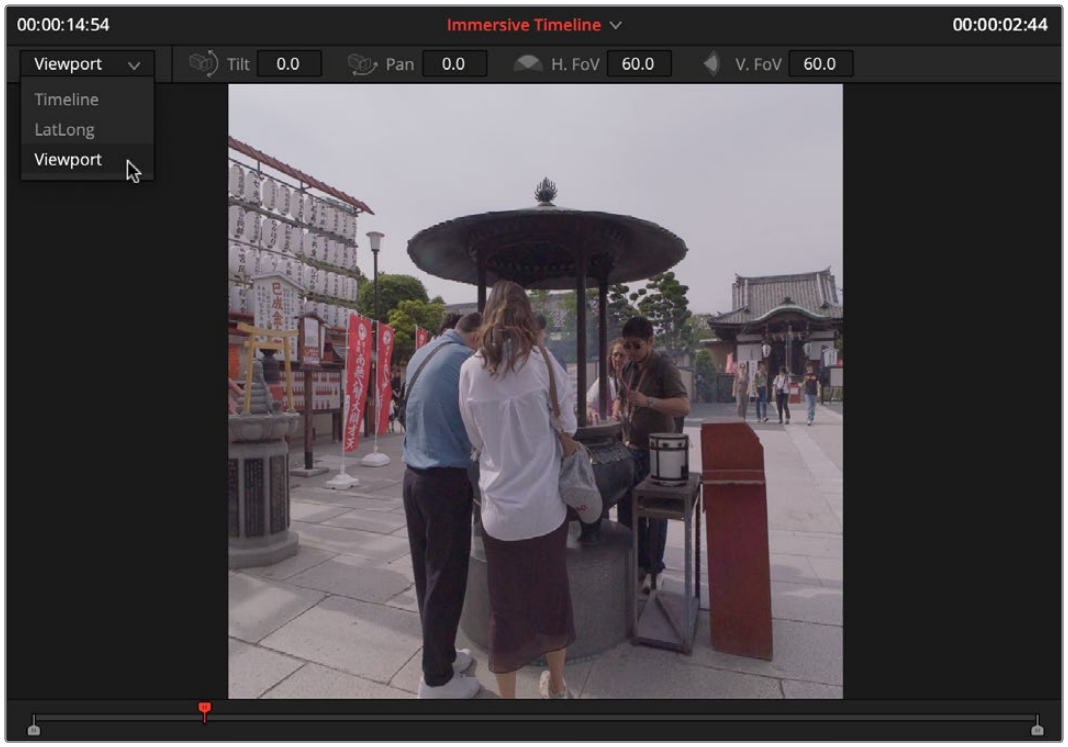
새로운 도구바가 추가적인 보기 옵션과 함께 뷰어 위에 나타납니다.

3 'LatLong'을 클릭하면 이미지의 정방형 투영본을 볼 수 있습니다. 이 모드는 이미지를 왜곡시켜 일반적인 2D 미디어 비율처럼 보이도록 하는데, 이를 통해 트래킹 및 합성 등의 특정 후반 제작 과정이 훨씬 수월해집니다.



4 'Viewport'를 클릭하면 정사각형 프레임으로 잘린 몰입형 장면을 볼 수 있습니다. 이 모드를 통해 표준 모니터에서 해당 장면의 디테일을 정확한 비율로 볼 수 있지만, 프레임 밖의 움직임을 놓치지 않도록 주의해야 합니다.

몰입형 도구바 상단에 있는 변형 컨트롤을 사용해 스테레오스코픽 이미지를 틸트/패닝할 수 있을 뿐 아니라 수평 및 수직 시야각을 조절할 수도 있습니다.



트랜지션

2개의 몰입형 비디오 클립 간 트랜지션이 필요한 경우, 바로 장면을 전환하거나 클립 양쪽의 핸들을 사용해 페이드 인 또는 페이드 아웃 또는 'Dip To Color Dissolve' 효과를 사용할 수 있습니다. 이런 트랜지션 효과는 푸티지에 입혀지는 것이 아니라 Apple Vision Pro에서 영상이 재생되는 동안 렌더링되어 나타난다는 사실을 기억하세요.

1 'Effects' 패널을 여세요.

2 사이드바에서 'Toolbox' > 'Video Transitions'로 이동하세요.

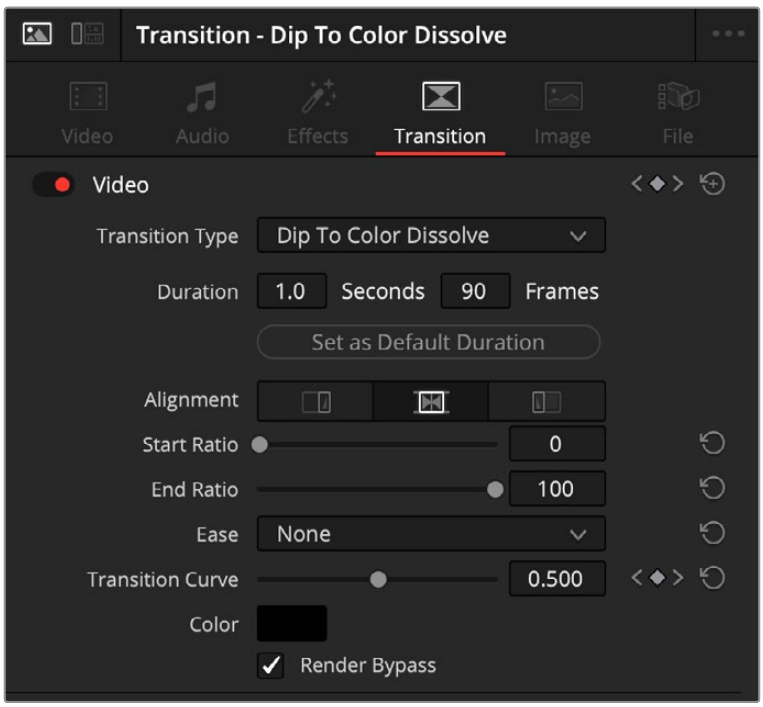
3 'Dip To Color Dissolve'를 두 클립 간의 컷 포인트로 드래그하세요.



참고 트랜지션을 적용하려면 두 클립의 컷 포인트 이상으로 트리밍 데이터가 충분히 있어야 합니다. 트리밍 데이터(또는 핸들 길이)가 충분하지 않을 경우, 트랜지션이 적용되지 않습니다.

4 'Inspector' 패널에서 'Transition' 탭으로 이동한 다음 'Render Bypass'를 선택하세요. 이렇게 하면 Apple Vision Pro에 의해 디졸브가 정확하게 렌더링됩니다.

5 이제 'Dip To Color Dissolve Inspector'에 남아 있는 파라미터를 사용하여 트랜지션의 길이와 위치, 애니메이션 속도, 색상을 조절할 수 있습니다.



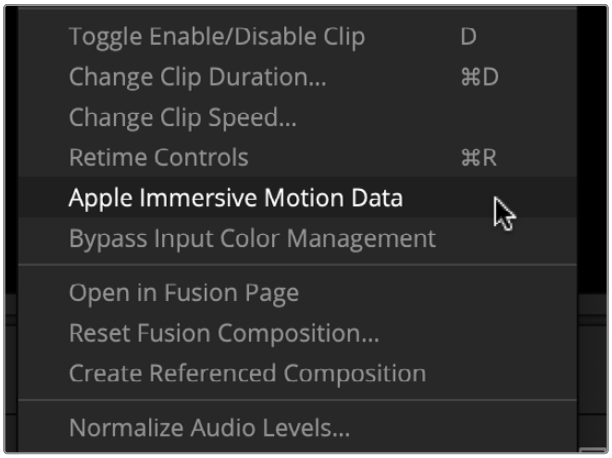
Apple Immersive Motion Data

몰입형 콘텐츠 작업 시, 너무 빠르거나 갑작스러운 카메라 움직임은 시청자로 하여금 방향 감각을 잃게 만들 수 있습니다. 따라서 편안한 시청을 위해 급작스러운 패닝/틸트, 회전 움직임, 또는 편평하지 않은 수평선이 포함된 테이크 사용은 피하는 것이 좋습니다.

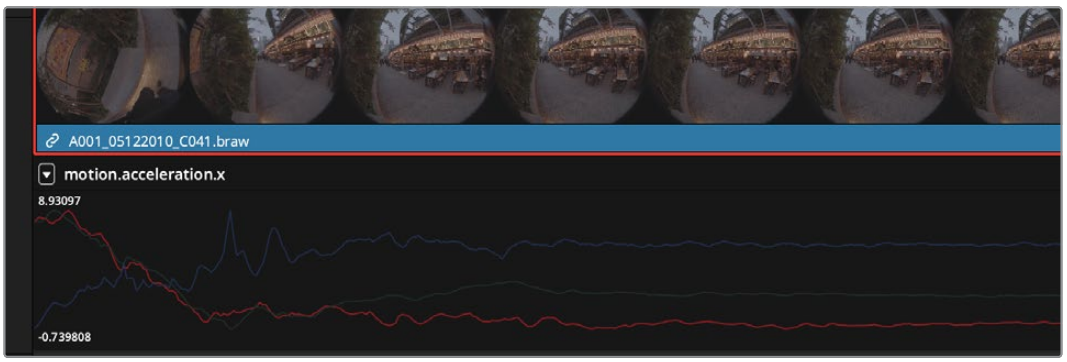
Apple Immersive Motion Data는 x, y, z 가속도와 자이로스코프 위치 값을 그래프로 표시하여 카메라 동작 및 움직임이 너무 강한 부분을 객관적으로 보여줍니다. 편집자는 이러한 정보를 바탕으로 영상에서 방향 감각에 혼돈을 일으키는 부분을 제외할 수 있습니다.

1 타임라인의 몰입형 클립을 우클릭하세요.

2 'Apple Immersive Motion Data'를 선택하세요.

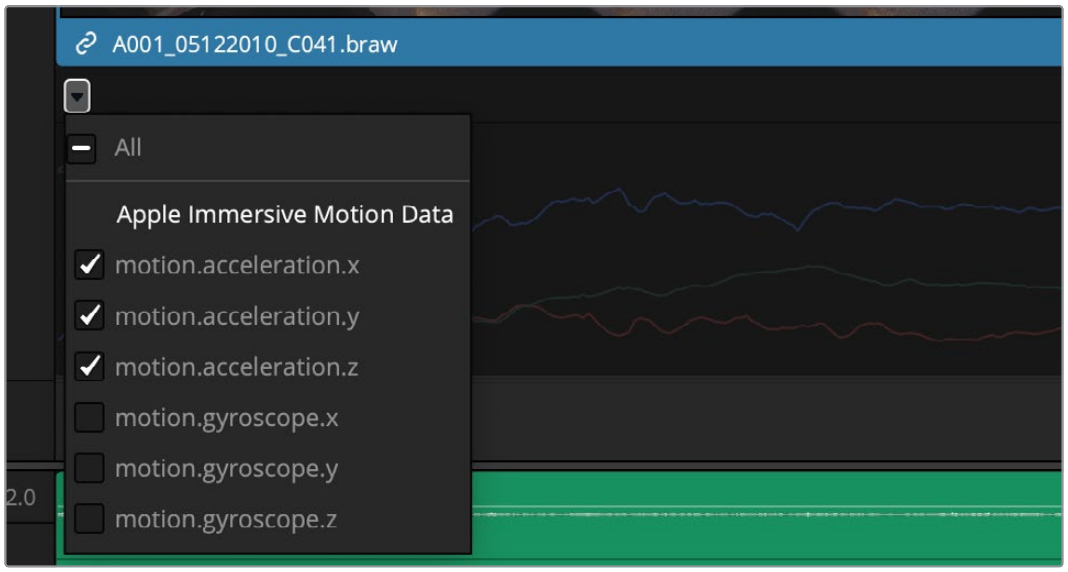


비디오 트랙에 x, y, z 가속도를 보여주는 그래프가 나타납니다.



해당 예시는 카메라 운영자가 카메라를 위로 올리면서 녹화된 모습을 보여줍니다. 그래프에서 급등하는 지점은 카메라를 들어 올려 회전한 다음 어깨에 올려놓는 갑작스러운 동작을 나타냅니다. 이상적으로는 카메라가 안정된 후 평평한 값으로 표시되는 비디오의 후반부를 사용하는 것이 좋습니다.

3 그래프의 좌측 상단에 있는 네모난 화살표 아이콘을 클릭하세요.



영상 콘텐츠 및 카메라 동작의 특성에 따라 판독 기능을 활성화하거나 비활성화할 수 있습니다.

타이틀

편집 페이지에서는 'Effects' 패널의 타이틀 도구(Text 및 Text+, MultiText, Scroll, 하단 1/3 관련 옵션)를 사용하여 몰입형 장면에 텍스트를 추가할 수 있습니다.

타이틀 생성기를 몰입형 클립 위에 배치하면 타이틀 생성기가 클립의 몰입형 렌즈 프로필 데이터인 ILPD(Immersive Lnes Profile Data)를 사용해 텍스트를 정확하게 왜곡하여 해당 장면의 렌즈 워프와 맞춥니다.

검은색 배경막이나 몰입형 CGI 장면에 텍스트를 추가하는 경우처럼 ILPD를 사용할 수 없는 경우에는 기본 렌즈 왜곡 값이 적용되어 Apple Vision Pro에서 볼 때 텍스트가 적절하게 변형되어 표시됩니다.

1 'Effects' 패널에서 'Toolbox' > 'Titles'를 선택하세요.

2 텍스트 생성기를 타임라인으로 드래그하세요. 텍스트가 해당 장면에 통합되도록 텍스트 생성기를 몰입형 클립 위의 비디오 트랙에 배치하세요.



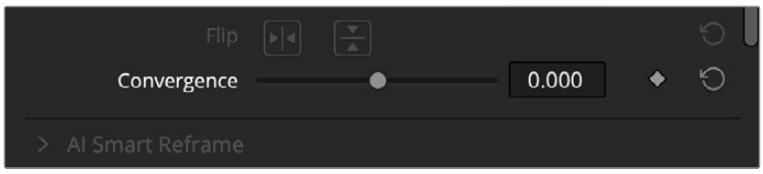
텍스트 생성기는 배경막과 마찬가지로 클립 경계선을 넘을 수 없습니다. 각각의 클립이 개별 렌즈 공간을 사용해 렌더링되기 때문에 타이틀이 클립 경계선을 넘어 걸쳐지게 되면 숏 사이에서 변형될 위험이 있습니다.

3 타이틀을 비디오 편집 포인트에 맞게 트리밍하세요.

4 'Inspector'에서 타이틀 텍스트를 입력하고 포맷을 설정하세요.

5 'Settings' 탭으로 이동하여 'Convergence' 슬라이더를 조절해 3D 공간에서 텍스트 위치를 조절하세요.

이 슬라이더를 왼쪽으로 움직이면 공간 속에서 타이틀을 배경 쪽으로 밀며, 오른쪽으로 움직이면 타이틀을 전경 쪽으로 가지고 옵니다.



Apple Vision Pro에서 모니터링 시 수렴 정도를 가장 잘 확인할 수 있습니다. 다른 모니터링 수단을 사용하는 경우, 컬러 페이지에서 비디오 레이어의 'Vision'(시청) 모드를 변경하여 뷰어에 표시되는 텍스트 수렴 방식을 조정할 수 있습니다.

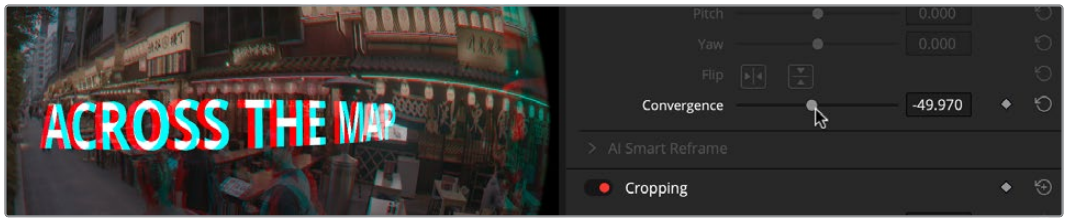
6 컬러 페이지로 이동하세요.

7 3D 팔레트를 여세요.

팔레트의 중앙에 있는 'Vision' 칼럼에서 몰입형 미디어 투사에 사용되는 스테레오스코픽 표준을 변경할 수 있습니다.

8 'Stereo' 탭을 선택하세요.

9 'Out' 모드를 'Anaglyph'(색상)으로 선택하세요.



편집 페이지의 'Inspector' 패널에서 텍스트 생성기의 'Convergence' 파라미터를 조정하면 애너글리프(Anaglyph) 안경 또는 애너크롬(Anachrome) 안경으로 볼 때 더욱 직관적으로 보입니다.

사용자 조언

Blackmagic URSA Cine Immersive 카메라가 출시된 초기에 영화 제작자들이 이 카메라를 직접 테스트하며 가장 효율적인 촬영 방법 및 조언, 그리고 기존 영상 촬영 방식에서는 생각할 수 없었던 새로운 워크플로를 발견해 냈습니다.

이에 따라 몰입형 영상을 시청하는 관객이 최대한의 몰입감과 시각적 편안함을 느낄 수 있도록 몰입형 영상 제작에 관련된 몇 가지 유의 사항을 공유하고자 합니다.

1 편집은 사전 제작 단계부터 시작되어야 합니다. 영상 제작자는 숏 길이와 시청자가 느끼게 될 자연스러운 장면 이동을 위해 최종 편집을 미리 염두에 두고 촬영을 시작해야 합니다.

2 특히 갑작스러운 카메라 움직임은 시청자에게 어지러움을 줄 수 있으므로 클립에 포함되지 않도록 하거나 빈도를 제한해야 합니다. 대신, 고정 숏을 사용하거나 패닝 및 돌리를 천천히 움직이며 촬영하세요.

3 균형이 맞지 않거나 극단적인 각도의 영상 또한 어지러움을 유발할 수 있으므로, 피사체와 동등한 레벨 또는 너무 가깝지 않은 편안한 거리에 맞춰 테이크를 완성하세요.

4 몰입형 미디어에서는 새로운 환경을 받아들이는 데에 2D 장면을 볼 때보다 시간이 더 오래 걸립니다. 시청자가 새로운 환경에 적응할 수 있도록 컷의 속도를 늦춰야 합니다.

5 한 클립에서 다른 클립으로 전환할 때는 이전 숏에서 시청자의 시선이 어디에 머물렀는지 항상 염두에 두세요. 시청자의 시선이 이전 숏에서 오른쪽에 있었다면, 중요한 동작이 프레임의 왼쪽에서 일어나면 안 됩니다.

6 공간 오디오는 스토리텔링에서 중요한 역할을 합니다. 몰입형 영상 제작 시 공간 오디오를 하나의 캐릭터로 생각하고 작업하세요.

VFX 합성

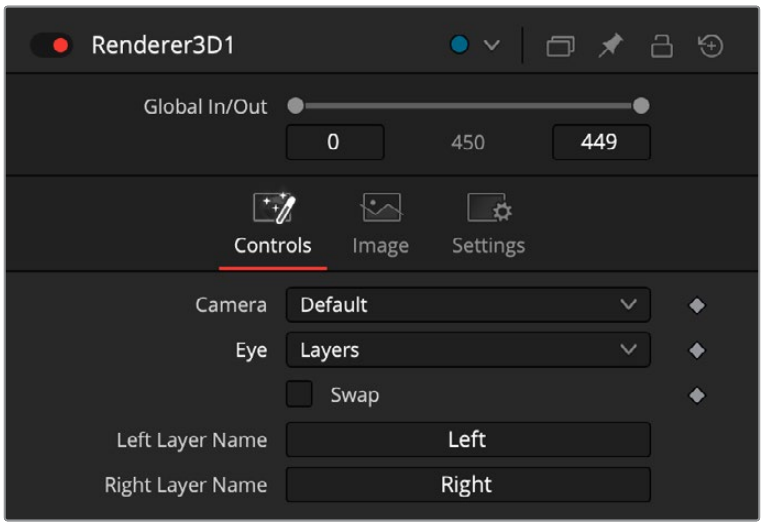
프로젝트 설정에서 'Apple Immersive Workflow'를 활성화하면, 전용 Fusion 파라미터를 통해 매끄러운 스테레오스코픽 워크플로 및 몰입형에 최적화된 합성 작업이 가능해집니다. 'MediaOut' 노드는 좌안 및 우안 출력 옵션을 제공하며, 'Immersive Patcher' 도구를 사용해 구형 투영 및 합성을 작업할 수 있습니다.

MediaOut 노드

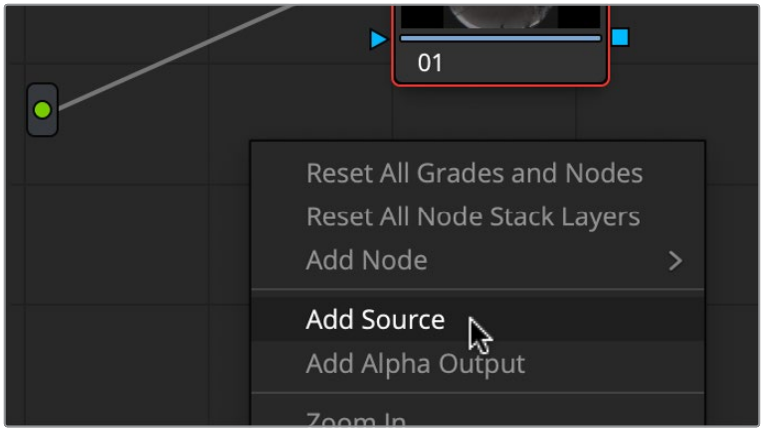
몰입형 클립 작업 시 Fusion 페이지로 처음 진입하면 2개의 'MediaIn' 노드가 각각의 'MediaOut' 노드에 연결된 노드 그래프가 나타납니다.



'MediaOut' 인스펙터에서 'Stereo Eye' 파라미터를 사용하여 각 눈 신호 출력을 라우팅할 수 있습니다.



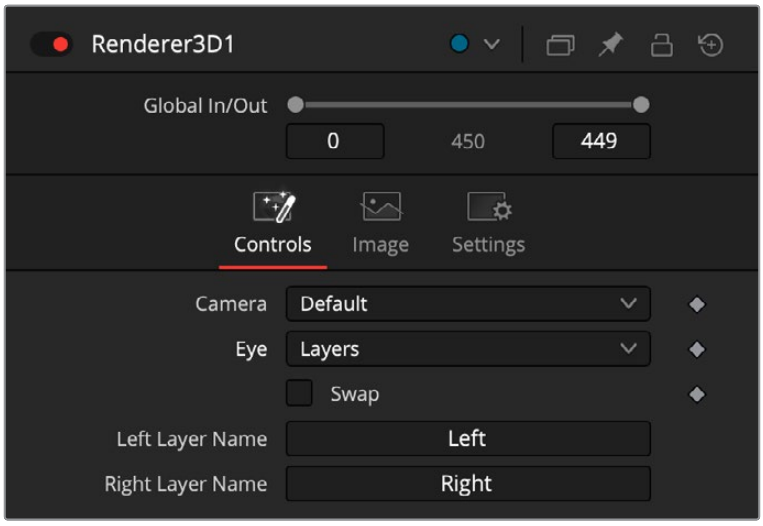
'MediaOut1'은 좌안으로, 'MediaOut2'는 우안으로 기본 설정되어 있습니다. 이 설정을 통해 스테레오에 적합한 컴포지트와 그래픽 오버레이 작업을 즉시 시작할 수 있습니다. 또한 이 2개의 'MediaOut' 노드는 후반 이미지 보정 작업을 위해 컬러 페이지의 노드 편집기에 2개의 신호로 전송됩니다. 노드 편집기에서 우클릭한 후 'Add Source'를 클릭하면 컬러 페이지에서 'MediaOut2' 신호에 접근할 수 있습니다.



'Stereo Eye' 파라미터가 'Auto'로 설정되어 있을 경우에는 'MediaOut' 노드가 두 신호를 합친 다음 타임라인에서 적절한 눈으로 배분합니다. 이를 통해 Fusion 멀티플레이어 워크플로를 사용하여 좌/우 신호를 단일 'MediaOut' 노드로 보낼 수 있습니다.

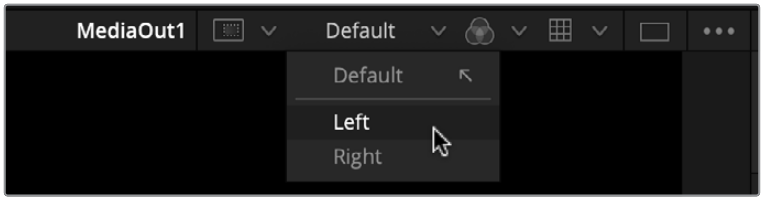
Renderer3D

몰입형 타임라인에서 3D 그래픽 또는 CGI로 작업할 때 'Renderer3D' 노드를 사용해 다양한 눈 설정 및 좌/우 레이어를 처리할 수 있습니다.



이를 통해 복잡한 3D 타이틀이나 전체 CGI 장면을 생성하여 이를 몰입형 프로젝트의 스테레오스코픽 타임라인으로 직접 전송할 수 있습니다.

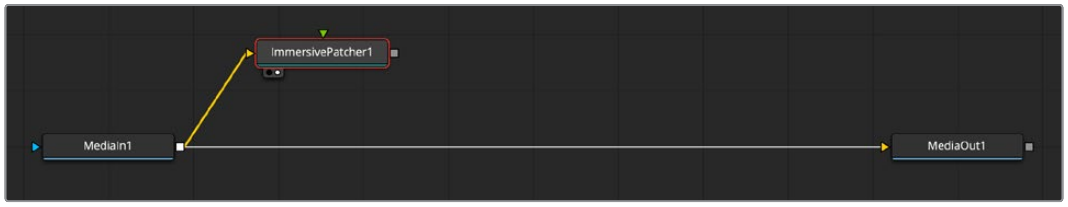
이 셋업을 사용할 경우, Fusion 뷰어의 우측 상단에 있는 드롭다운 메뉴를 통해 컴포지트의 좌안 및 우안을 확인할 수 있습니다.



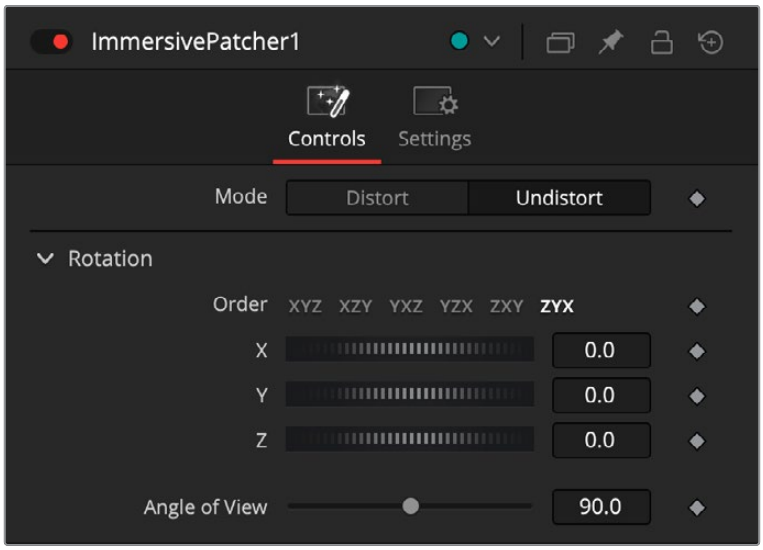
Immersive Patcher를 사용하여 합성하기

Immersive Patcher 도구는 몰입형 클립의 특정 부분을 일시적으로 왜곡 해제하여 평면에서 손쉽게 트래킹, 페인팅, 합성을 할 수 있도록 합니다. 이 작업을 완료한 후, 두 번째 Immersive Patcher를 사용하여 몰입형 신호에 다시 왜곡 적용한 다음, 이 새로운 합성 요소를 장면에 통합할 수 있습니다.

1 'Immersive Patcher'를 노드 파이프라인에 추가하세요.



2 인스펙터에서 'Undistort'를 클릭하세요.



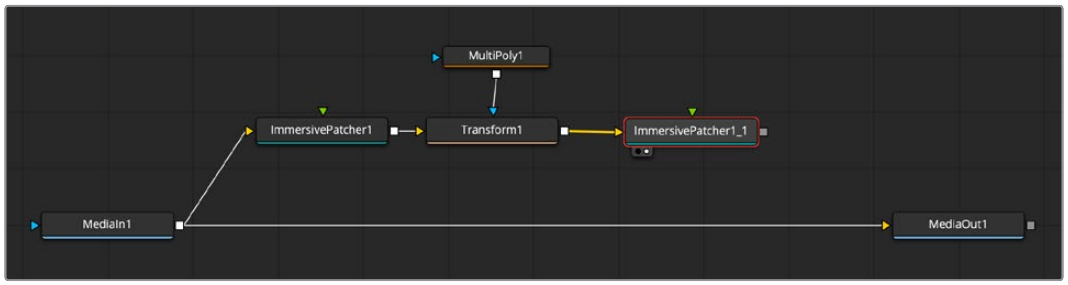
3 'X', 'Y', 'Z' 회전 파라미터를 사용하여 평면화할 몰입형 이미지 영역을 선택하세요.

4 일반 2D 비디오 신호와 마찬가지로 합성을 진행하세요. 다음에 국한되지는 않지만, 원치 않는 객체를 제거하기 위한 표준 페인트 방법이나 변형 및 폴리곤을 사용한 세트 확장용 프레임 영역 샘플링, 표지판이나 스크린을 바꾸기 위한 매트, 텍스트 및 외부 이미지/비디오 레이어 적용 등의 합성을 진행할 수 있습니다.

   완료 후, 평면 이미지를 다시 왜곡 적용하여 원본 몰입형 워프에 맞추세요.

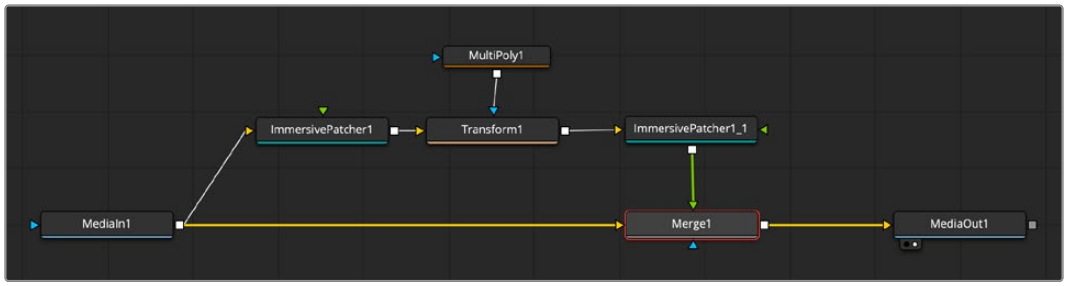
5 'Immersive Patcher'를 복사하여 원래의 회전 및 각도 뷰 정보를 유지하세요.

6 컴포지트 노드 뒤에 ‘Immersive Patcher’를 붙여 넣고 합성 파이프라인의 끝에 연결하세요.



7 인스펙터에서 ‘Distort’를 클릭하세요.

8 합성된 패치를 원본 클립에 병합하여 효과를 해당 몰입형 장면에 통합하세요.



협업하여 작업하는 경우, Fusion 페이지를 빠져나온 후 DaVinci Resolve 내에서 몰입형 타임라인을 편집하거나 믹싱 및 색보정 작업을 계속할 수 있습니다.

몰입형으로 변환하는 경우, 각 눈을 ‘ACES AP0’의 ‘EXR’로 렌더링하여 몰입형 딜리버리를 위한 속성을 최대한 유지할 수 있습니다.

컬러 그레이딩

다음 부분에는 DaVinci Resolve의 컬러 페이지에서 몰입형 미디어로 작업 시 사용할 수 있는 광학 및 컬러, 3D 도구가 소개되어 있습니다.

몰입형 콘텐츠만의 속성으로 인해 후반 제작 도구 중 일부는 사용 불가할 수 있습니다. 현재 사용할 수 없는 도구일지라도 향후 몰입형 콘텐츠 제작을 지원하기 위한 업데이트 과정 중임을 알려드립니다.

노이즈 리덕션

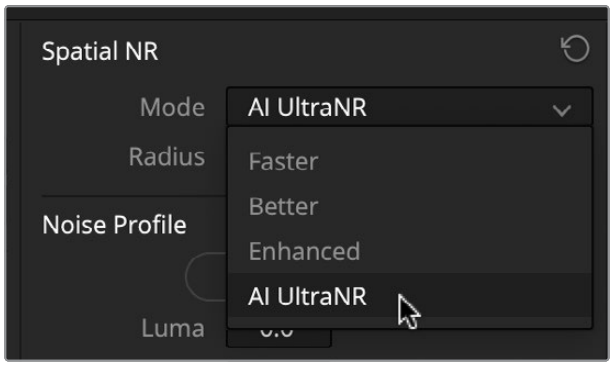
몰입형 비디오로 작업 시 노이즈 리덕션(Noise reduction) 도구를 사용할 수 있습니다. ‘Motion Effects’ 팔레트에 있는 ‘Spatial NR’은 소스 미디어의 노이즈 패턴을 분석해 원본 이미지 디테일을 유지하는 동시에 노이즈를 최대한으로 줄입니다.

1 컬러 페이지의 도구바에서 ‘Motion Effects’ 팔레트를 여세요.

‘Spatial NR’ 칼럼에서 원하는 노이즈 리덕션 모드를 선택하세요.

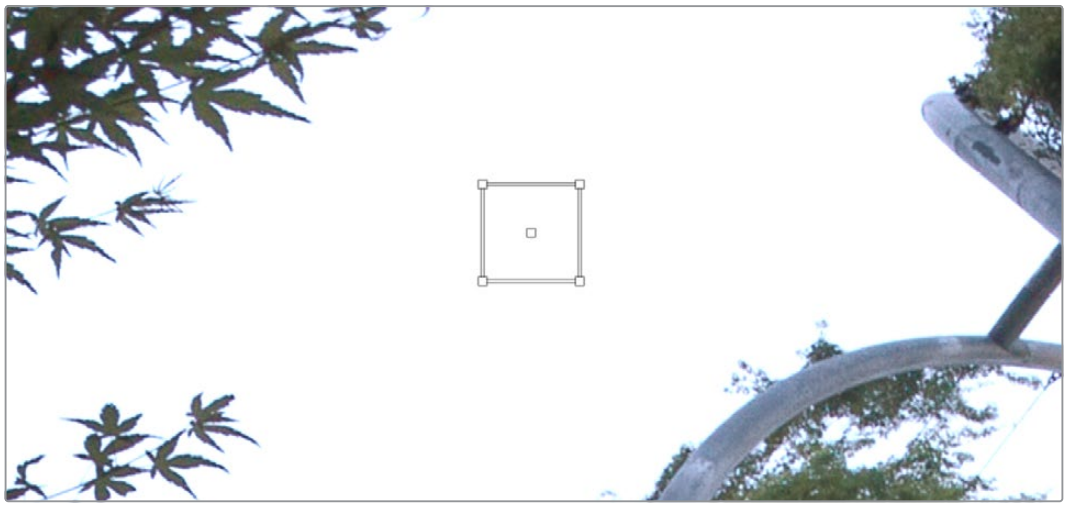
- ‘Faster’ 및 ‘Better’, ‘Enhanced’ 모드는 모두 주어진 반경 크기 내에서 패턴 분석을 실행하고 설정된 임계값에 따라 출력을 구성합니다. 이 모드는 처리가 복잡한 순으로 나열되어 있습니다.
- AI UltraNR은 프레임의 가장 평평한 영역을 찾아 순수한 노이즈를 분석하여 신호에서 노이즈 패턴을 추출하는 AI 기반의 노이즈 패턴 식별 기능입니다.

2 ‘Spatial NR’ 모드를 ‘AI UltraNR’로 설정하세요.



3 ‘Analyze’를 클릭하세요.

뷰어에 상자가 나타나서 주어진 프레임에서 픽셀 데이터의 ‘가장 빈’ 영역을 나타냅니다. 사각형 내부의 데이터는 구체적인 시각적 콘텐츠가 없는 경우에 카메라 센서에서 생성된 디지털 노이즈를 추출하는 데 이상적입니다.



이 상자의 위치나 크기를 변경할 수 있지만 일반적으로 권장되지 않습니다. AI UltraNR는 최적의 분석 영역을 사람의 눈보다 잘 선별하기 때문입니다.

또한 ‘Noise Profile Luma’ 및 ‘Chroma Threshold’ 설정을 수동으로 조절할 수도 있지만, AI UltraNR는 깨끗한 노이즈 제거를 위해 최적의 임계값 믹스를 자동으로 설정할 수 있습니다.

프라이머리 그레이딩

프라이머리 색보정은 일반 비디오 푸티지로 작업할 때와 동일하게 사용할 수 있습니다. 프라이머리 컬러 휠이나 로그 휠, HDR 팔레트, HSL 커브 사용 시 정규화 및 균형, 매칭을 기존 방식과 동일하게 적용하세요. Color Slice 및 Color Warper 팔레트도 기존과 동일하게 작동합니다.

세컨더리

퀄리파이어와 같은 픽셀 기반 선택 방법은 몰입형 미디어에서도 기존과 동일하게 작동합니다.

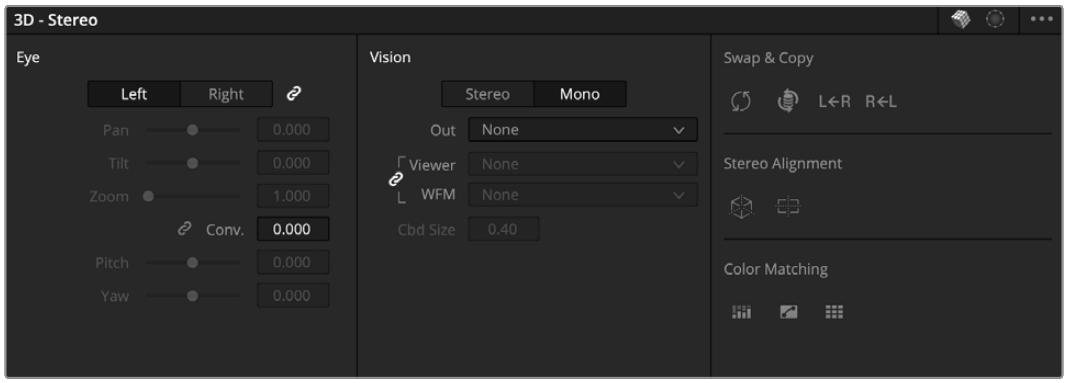
물체 또는 윈도우에 기반한 선택 영역은 구형을 평면(또는 ‘Viewport’)으로 변환한 환경에서만 작동합니다. 이로 인해 모양의 정확성을 요구하는 대부분의 세컨더리 방법은 현재 사용 불가합니다. 여기에는 ‘Power Windows’, 더 나아가 ‘Tracker’ 팔레트와 ‘Magic Mask’ 도구가 해당합니다.

크기 조절

현재 몰입형 콘텐츠에 사용할 수 있는 크기 조절 컨트롤(예: 줌 및 패닝)은 없습니다.

3D 팔레트

컬러 페이지의 3D 팔레트에서는 다양한 3D 카메라 및 그래픽 포맷을 관리하는 컨트롤을 사용할 수 있습니다. Blackmagic URSA Cine Immersive는 카메라 렌즈 및 초점 길이, 조리개가 고정되어 있어, 다른 스테레오스코픽 포맷과는 달리 양쪽 눈의 정렬을 조절할 필요가 없습니다.



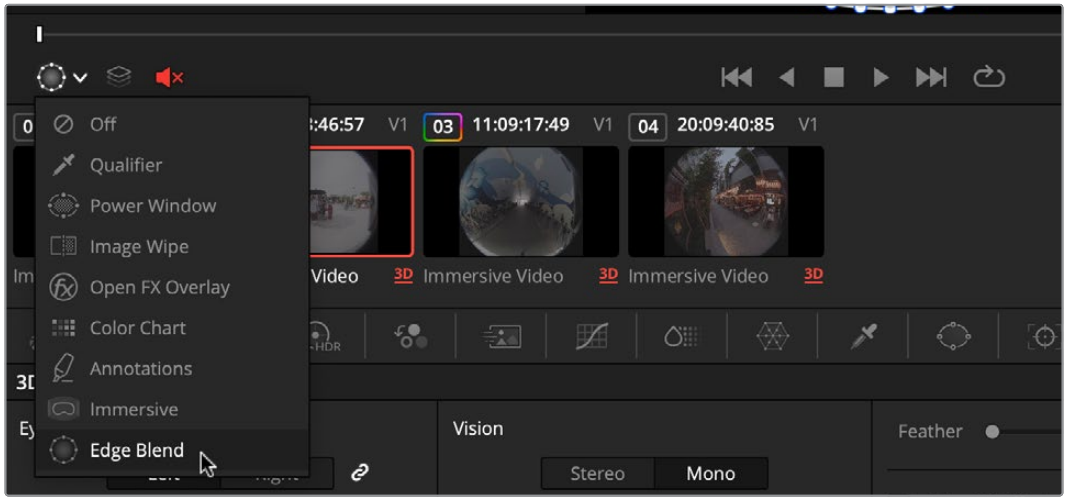
'Edge Blend' 모드는 몰입형 콘텐츠 전용 기능으로, 몰입형 비디오 프레임의 테두리를 부드럽게 조절할 수 있습니다. 이 기능을 사용해 영상에 보이는 촬영 장비를 숨기거나 시청자의 시야각을 제한할 수 있습니다. 시청자 입장에서는 이 경계선 바깥으로는 배경막밖에 보이지 않습니다.

1 3D 팔레트를 여세요.

2 오른쪽 상단에서 모드를 'Edge Blend'로 변경하세요.



3 뷰어의 온스크린 컨트롤을 'Edge Blend'로 변경하세요.



그러면 비디오 프레임 테두리를 원형으로 둘러싼 컨트롤 포인트와 함께 가운데에 고정 핸들이 나타납니다.

4 컨트롤 포인트를 드래그하여 위치를 변경하세요.



아니면 우클릭하여 포인트를 삭제하세요.

5 3D 팔레트의 오른쪽 칼럼에서 'Feather' 파라미터를 사용해 프레임의 테두리를 부드럽게 만드세요.

6 'visionOS Effect'를 선택하여 새로운 테두리 데이터가 Apple Vision Pro에서 재생 시 올바르게 처리될 수 있도록 하세요. 이 항목을 선택하지 않을 시, 테두리가 렌더링된 비디오 파일로 구워집니다.

7 왼쪽 칼럼에 있는 'Left' 및 'Right' 탭을 사용해 뷰어에서 각 눈을 전환하세요.

참고로, 3D 팔레트의 대부분 컨트롤이 회색으로 변경됩니다. DaVinci Resolve의 파라미터는 사용자가 현재 작업 중인 미디어에 적용할 수 없을 때 회색으로 비활성화됩니다. 몰입형 콘텐츠로 작업하는 경우도 이와 동일합니다.

8 가운데 칼럼에서 'Stereo'와 'Mono' 탭을 사용해 몰입형 클립의 단일 뷰어 및 양쪽 렌즈 뷰어를 전환할 수 있습니다.

CST 컬러 관리

컬러 페이지의 노드 편집기에서 노드에 색공간 변형을 뜻하는 'CST'(Color Space Transform)를 사용하여 정확한 색상 매핑 및 타임라인의 최대 출력 니트(nit) 루미넌스를 조절할 수 있습니다.

권장되는 워크플로에는 다음의 4가지 CST 노드가 포함됩니다.

1. INPUT CST - 카메라에서 작업 색공간으로 변환합니다.

   색보정은 이 두 노드 사이에서 진행됩니다.
2. OUTPUT CST - 작업 색공간에서 HDR Mac 디스플레이로 변환합니다.
3. MONITOR CST - HDR 모니터링을 위해 1000 니트에서 250 니트로 변환합니다.
4. AVP CST - Apple Vision Pro 모니터링 및 딜리버리를 위해 250 니트에서 108 니트로 변환합니다.



CST 타임라인 매핑

먼저, 카메라 소스 입력 색공간에서 현재 '작업' 중인 타임라인 색공간으로 매핑을 거칩니다. Blackmagic URSA Cine Immersive 카메라는 DaVinci Intermediate 로그 커브를 사용하는 Blackmagic Design Wide Gamut 4/5세대 색영역에서 작동합니다.

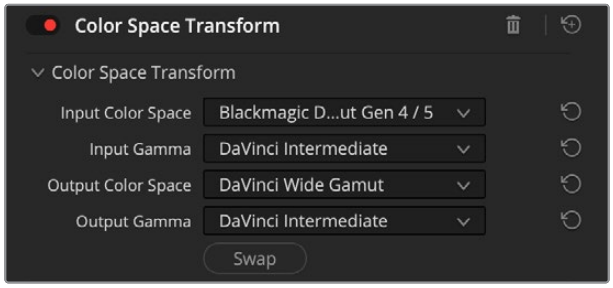
1. 'Effects' 패널을 여세요.
2. CST를 노드 편집기의 빈 노드 01로 드래그하세요.

   입력 색공간에서 타임라인 색공간으로 매핑 시, 이 CST가 다른 어떤 색보정 작업보다 먼저 노드 트리에서 첫 노드로 배치되어야 합니다.

   입력 색공간에서 모니터링 또는 딜리버리 색공간으로 직접 매핑하는 경우, 이 CST는 노드 트리에서 색보정 작업 뒤에 이어지는 마지막 노드여야 합니다.
3. 노드 01을 'INPUT CST'로 명명하세요.

참고 이 설명서에 제안된 노드명은 추후에 참고하거나 스크린숏을 보다 쉽게 사용할 수 있도록 하기 위해 설정되었습니다. 사용자가 원하는 노드명을 사용하거나 노드 명명 과정을 건너뛰어도 무방합니다.

4. 'Input Color Space'를 'Blackmagic Design Wide Gamut Gen 4/5'로 설정하세요.
5. 'Input Gamma'를 'DaVinci Intermediate'로 설정하세요.
6. 'Output Color Space'를 'DaVinci Wide Gamut'으로 설정하세요.
7. 'Output Gamma'를 'DaVinci Intermediate'로 설정하세요.



8 새로운 시리얼 노드 02를 생성하세요.

9 첫 번째 노드 01을 새 노드 02로 복사해 붙여 넣으세요.

10 노드 02를 'OUTPUT CST'로 명명하세요.

노드 트리의 타임라인 색상 공간이 일관성을 유지하도록 하려면 출력 색공간과 출력 감마를 이 새로운 시리얼 노드의 입력으로 지정해야 합니다.

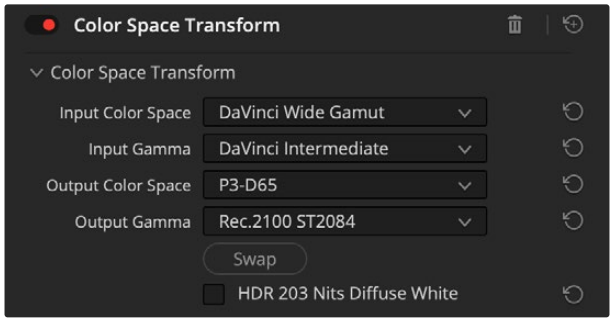
11 'Swap' 버튼을 눌러 입력 및 출력 파라미터를 전환하세요.

입력 색공간과 입력 감마가 이제 이 CST 노드에 적합합니다.

12 'Output Color Space'를 'P3-D65'로 설정하세요. 이는 현재 Mac의 기본 화면 색영역입니다.

13 'Output Gamma'를 'Rec.2100 ST2084'로 설정하세요. 이는 1000 니트 신호에 대한 HDR 표준입니다.

모든 색보정 작업은 이 두 노드 사이에서 진행됩니다. 하지만 노드 기반의 색상 관리를 통해 작업하기 때문에 타임라인 색상 관리의 범주 밖에서 작업할 수 있는 고유한 기능도 사용 가능합니다. 대부분의 컬러리스트는 이를 활용하여 특정 그레이딩 작업이나 효과 또는 서드파티 요소를 적용합니다.



ICC 프로파일

Blackmagic URSA Cine Immersive 신호가 Mac 디스플레이에 올바르게 매핑되었는지 확인하려면 'System Settings' > 'Displays'로 이동하여 컬러 프로파일 프리셋이 'Apple XDR Display(P3-1600 니트)'로 설정되어 있는지 확인하세요.

CST 그레이딩 모니터 매핑

추가 CST를 사용하여 색보정 작업 시 모니터링을 위해 최대 니트 출력값을 일시적으로 제한할 수 있습니다.

1 색상 관리 'OUTPUT CST 02' 뒤에 새로운 시리얼 노드 03를 생성하세요.

2 이 노드 03의 이름을 ‘MONITOR’로 명명하세요.



3 CST 효과를 ‘MONITOR’ 노드 03으로 연결하세요.

입력값은 이전 CST에서 출력되는 신호와 일치해야 합니다.

4 ‘Input Color Space’를 ‘P3-D65’로 설정하세요.

5 ‘Input Gamma’를 ‘Rec.2100 ST2084’로 설정하세요.

이 CST 노드의 목적은 오직 이미지의 루미넌스를 더 작은 니트 값으로 매핑하기 위한 것으로, 입력 및 출력 파라미터는 동일하게 유지됩니다.

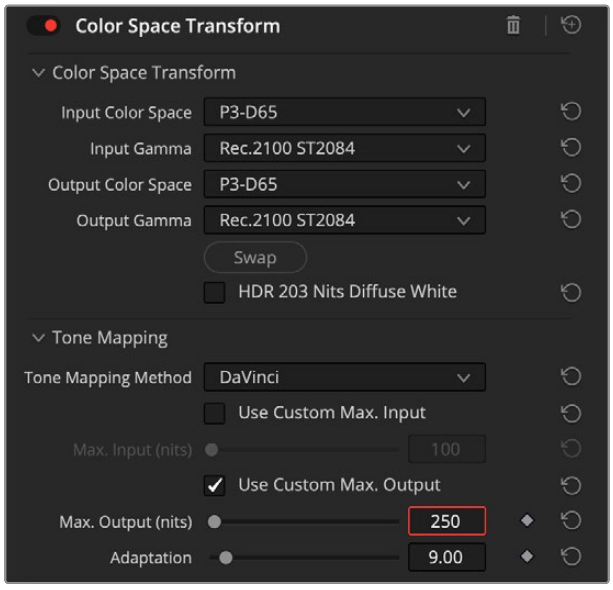
6 ‘Output Color Space’를 ‘P3-D65’로 설정하세요.

7 ‘Output Gamma’를 ‘Rec.2100 ST2084’로 설정하세요.

참고 입력과 출력이 동일하다면 매핑이 발생하지 않으므로 CST의 색공간(Color Space)이나 감마(Gamma)를 설정할 필요가 없습니다. 하지만 본 예시에서는 사용자가 지정한 니트 매핑 적용 시 신호의 로그 곡선 값과 모양의 영향을 받기 때문에 설정이 필요합니다.

8 ‘Use Custom Max. Output’을 선택하세요.

9 ‘Max. Output (nits)’을 ‘250’으로 설정하세요.



1000 니트 신호는 이제 DaVinci 톤 매핑 방식을 통해 250 니트로 매핑됩니다.

이를 통해 HDR 색보정 모니터(예: Flanders XMP310, Sony BVM-HX310 등)에서 렌더링된 Apple Vision Pro 신호와 거의 일치하는 이미지가 생성됩니다.

하지만 Apple Vision Pro에서 프로젝트를 모니터링하려면 톤 매핑 작업을 하나 더 수행해야 합니다.

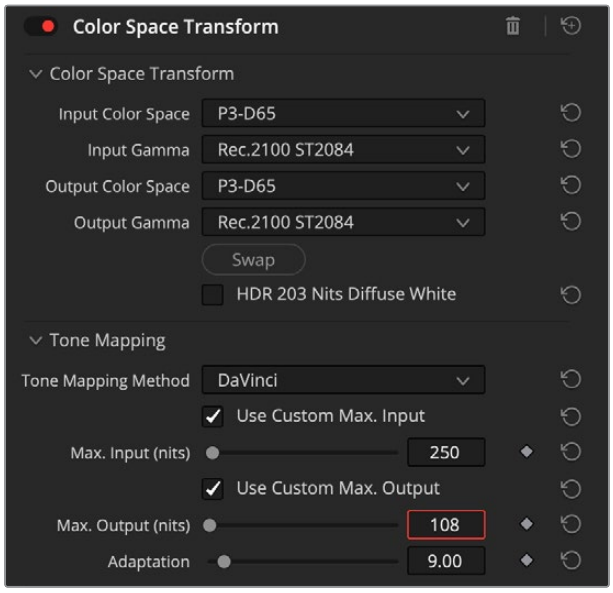
CST Apple Vision Pro 모니터링

최종 CST 노드는 실시간 모니터링 및 미리보기, Apple Vision Pro 기기로 최종 딜리버리하는 데에 사용됩니다.

1. ‘MONITOR’ 노드 03 뒤에 새로운 시리얼 노드 04를 생성하세요.
2. ‘MONITOR’ 노드 03를 노드 04에 복사하여 붙여 넣으세요. 이를 통해 동일한 입력 및 출력 파라미터가 사용되며 니트 숫자만 변경됩니다.
3. 노드 04를 ‘AVP’로 명명하세요.



4. ‘Use Custom Max. Input’을 선택하세요.
5. ‘MONITOR’ 노드에서 매핑된 신호의 정확한 해석을 위해 ‘Max. Input(nits)’을 ‘250’으로 설정하세요.
6. ‘Max. Ouput(nits)’을 ‘108’로 설정하세요. 이는 Apple Vision Pro에 모니터링하고 전달하기 위한 최대 니트값입니다.



이제 색보정을 시작하면 됩니다. 사용자의 워크플로에 따라 필요한 경우, CST 노드들을 비활성화하고 작업할 수도 있습니다. 예를 들어, 1차 색보정 작업을 외부 색보정 모니터를 사용하여 진행할 경우, AVP 노드 04를 비활성화하여 사용자의 HDR 색보정 모니터에서 250 니트 신호가 제대로 표시되도록 하여야 합니다. Apple Vision Pro에서 미리보기로 전환하거나 최종 딜리버리로 전환 시, AVP 노드 04를 다시 활성화하세요.

타임라인 노드 편집기에 CST를 배치할 수 있나요?

아니요. 현재는 클립 톤 매핑 기능이 개별 클립을 기반으로 실행되는 CST 적응 파라미터의 정확한 적용에 의존하고 있기 때문에 불가능합니다.

오디오

Apple Vision Pro와 DaVinci Resolve는 공간 오디오 및 앰비소닉 오디오, AVP (Apple Vision Pro) 오디오, AIA (Apple Immersive Audio) 포맷을 지원합니다.

공간 오디오로 작업 시 오디오 후반 제작의 상당 부분이 기존과 변동 없이 진행됩니다. 가장 큰 차이점은 첨단 스피커로 모니터링하는 것이 아니라, 오디오 엔지니어가 정확한 공간 몰입형 오디오를 확인하고 보다 나은 청자의 청음 환경을 재현하기 위해 헤드폰이나 Airpod를 사용해야 한다는 점입니다.

한편, 2D 믹스의 경우, 다이얼로그의 대부분은 중앙 채널로 믹싱되며, 공간 오디오는 해당 사운드가 발생하는 곳으로 믹싱되어야 합니다. 또한 반드시 더킹 및 이퀄라이제이션, 밸런싱을 사용하여 청자에게 부담을 주거나 혼란을 주지 않는 방향으로 오디오 스템(Audio Stem)을 믹싱해야 합니다.

몰입형 공간 요소는 일반 스테레오 스템과 합칠 수 있습니다. 이 경우, 청자의 머리가 향하는 방향과 상관없이 특정 음양이 일정하게 유지됩니다. 이는 영화 음악이나 보이스오버 등의 배경으로 쓰이는 사운드를 명확하고 해석하기 쉽게 유지하는 데 도움이 됩니다.

공간 오디오에서는 주변 환경 소리가 더 조용하게 해석되는 경향이 있으므로, 이를 더 적극적으로 반영해 믹싱해야 할 경우가 많습니다. 하지만 이 때문에 듣는 사람이 방향성에 혼란을 느끼면 안 됩니다.

리버브는 청자가 매일 자연스럽게 경험하는 것이므로 몰입감을 높이는 데 크게 이바지합니다. 주어진 환경에서 생성된 모든 소리는 청자를 둘러싼 표면에 반사되어 풍부한 리버브 품질을 만들어냅니다. 따라서 공간 오디오 설계 시 리버브를 반영하여 풍부한 몰입감을 더하는 것을 권장합니다.

Apple Immersive Audio(AIA) 포맷은 여러 포맷을 위한 컨테이너와도 같습니다.

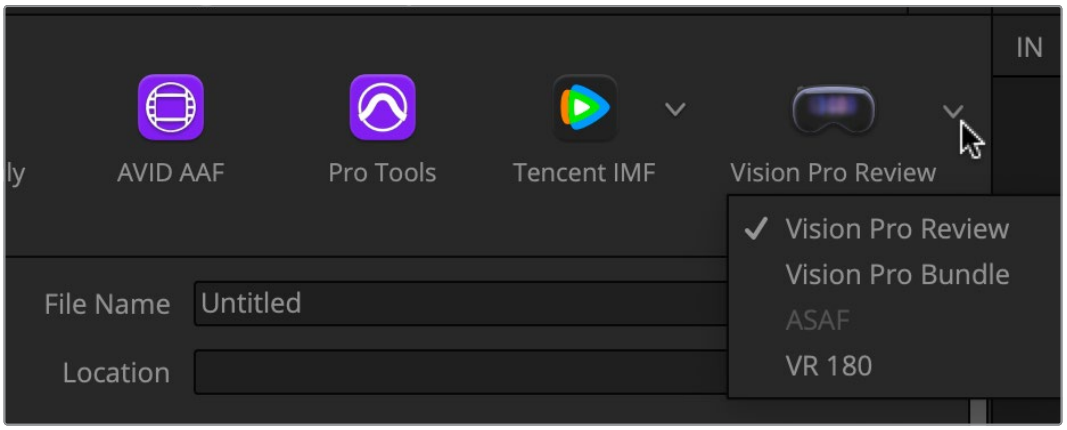
- **Apple Spatial Audio Format(ASAF)** 은 가공을 거치지 않은 작업용 오디오 포맷입니다.
- **Apple Positional Audio Codec(APAC)** 은 Airpods와 같은 APAC 지원 기기나 Apple Vision Pro의 AIVU 파일에서 재생할 수 있는 인코딩된 .mp4 포맷입니다.

Apple Spatial Audio Format(ASAF)

Apple의 공간 오디오 포맷인 ASAF(Apple Spatial Audio Format)는 주로 Apple Vision Pro의 VR 및 AR 앱에서 사용할 수 있도록 개발된 몰입형 오디오 포맷입니다. macOS에서만 지원되는 ASAF는 플러그인을 통한 ProTools 및 DaVinci Resolve의 내장 지원 기능을 통해 지원됩니다. ASAF는 Scenes이라 부르는 앰비소닉 소스에 대한 내장 지원 기능 및 버스 할당을 통한 라우팅 등의 첨단 기술 기능을 지원합니다. 믹싱은 물리적 버스 없이 가상으로 진행되며, 사용자는 Apple AirPod와 직접 연동되는 헤드 트래킹 기능을 갖춘 바이노럴 등의 다양한 모니터링 포맷을 선택할 수 있습니다. 패닝 기능은 DaVinci Resolve의 기존 도구(패닝 다이얼로그, 자동화, SpaceView, 뷰어 내 패닝 도구 등)를 통해 처리되며, 인스펙터에서 보조 컨트롤을 사용할 수 있습니다. 이를 통해 수직-수평(Lat-Long) 비디오에 대한 자동 트래킹 등 소스의 정확한 공간적 배치가 가능합니다. ASAF 믹스는 'Immersive Audio' > 'Export Master File' 옵션을 통해 WAV 또는 MP4 포맷(메타데이터 포함)으로 엑스포트할 수 있지만, 현재는 DaVinci Resolve에 임포트하거나 재생할 수 없습니다.

딜리버리

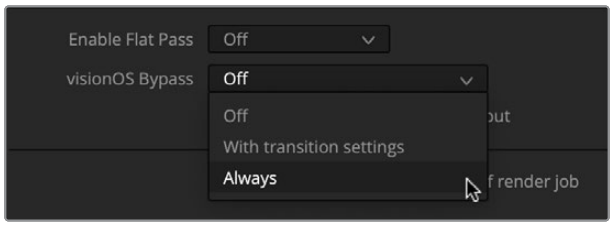
딜리버리 페이지에서 두 가지 렌더링 설정으로 Apple Vision Pro 파일을 렌더링하여 검토 및 딜리버리할 수 있습니다.



- **Vision Pro Review** - 미리보기 목적을 위해 작은 MV-HEVC 패키지를 생성합니다. Apple은 와이파이를 통해 Apple Vision Pro에 결과 파일을 공유하며, App Store에서 다운로드할 수 있는 Apple Immersive Video Utility(AIVU) 앱에서 해당 파일을 시청할 수 있습니다.
- **Vision Pro Bundle**- Apple TV에서 재생할 수 있도록 Apple Compressor에 저장 또는 처리 가능한 대용량 ProRes 파일 번들을 생성합니다.
- **ASAF** - WAV 또는 MP4 형식으로 ASAF 오디오 믹스를 렌더링합니다.
- **VR180** - VR180 규격을 준수하는 스테레오스코픽 영상 파일을 생성합니다.

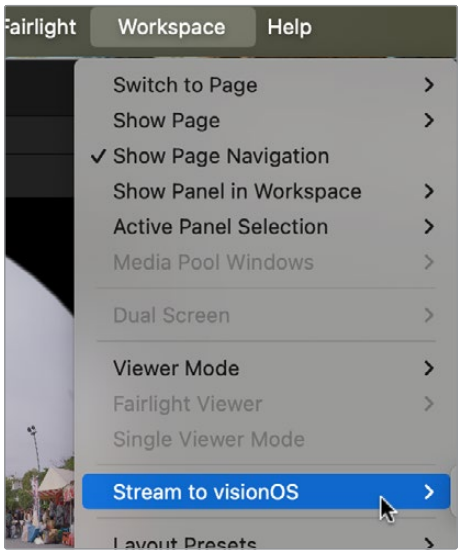
트랜지션 바이패스하기

몰입형 타임라인에 트랜지션을 추가하면 사전 렌더링되어 단일 재생 트랙으로 변환됩니다. 몰입형 콘텐츠를 위해 생성된 XML은 푸티지에서 사전 렌더링된 트랜지션, 그리고 다시 푸티지로 이어지는 이러한 렌더링된 스트림 간의 전환 과정을 명시합니다. 'Render Settings' 패널의 'Advanced Settings' 설정에서는 visionOS 전환 무시(항상), 타임라인의 상태에 따라 적용('전환 설정 포함'), 타임라인에 나타나는 대로 포함(끄기)하도록 할 수 있는 옵션이 포함되어 있습니다.



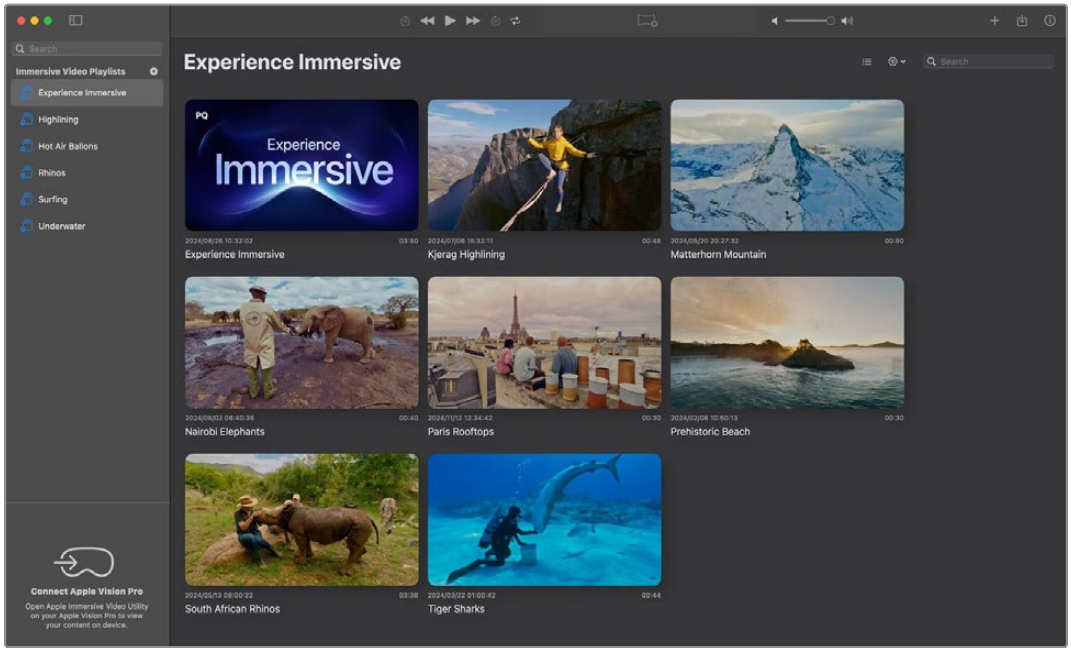
Apple Vision Pro로 스트리밍하기

DaVinci Resolve의 편집 및 컬러, 딜리버리 페이지에서 Apple Vision Pro로 직접 스트리밍할 수 있습니다. ‘Workspace’ 메뉴로 가서 ‘Streaming to visionOS’를 선택한 다음 사용자의 AVP 장치를 선택하세요 원활한 스트리밍을 위해 빠른 무선 네트워크(최소 Wi-Fi 5 이상, Wi-Fi 6 권장) 사용을 권장합니다.



Apple Immersive Video Utility에 관하여

Apple Immersive Video Utility는 Apple Immersive Video 콘텐츠의 제작, 관리 및 재생을 간소화하도록 설계된 macOS 및 visionOS 전용 앱입니다. 전문 영화 제작자와 몰입형 미디어 제작자를 위해 제작된 이 유틸리티를 사용하면 사용자가 180도 시야각과 공간 오디오를 포함한 8K 스테레오스코픽 비디오 파일을 임포트, 구성, 패키징 및 검토할 수 있습니다. 이 유틸리티는 AIVU(Apple Immersive Video Universal) 파일 생성을 지원하여 후반 작업 워크플로를 위한 메타데이터 검사 및 수정이 가능합니다. 사용자는 재생 목록을 만들고, 라이브러리를 정렬 및 검색할 수 있으며, 한 대 이상의 Apple Vision Pro 기기로 콘텐츠를 직접 스트리밍할 수도 있습니다.



Developer Information

Camera Control REST API

If you are a software or hardware developer you can build custom applications or leverage ready to use tools such as REST client or Postman to seamlessly control and interact with your compatible Blackmagic camera using Camera Control REST API. This API enables you to perform a wide range of operations, such as starting or stopping recordings, accessing disk information and much more. Whether you're developing a custom application tailored to your specific needs or utilizing existing tools, this API empowers you to unlock the full potential of your Blackmagic camera with ease. We look forward to seeing what you come up with!

NOTE It's important to mention that controlling Blackmagic cameras via REST API relies on the web manager being enabled on each compatible Blackmagic camera. Enable the web media manager in the Blackmagic Camera Setup 'network access' settings for each camera you are controlling.

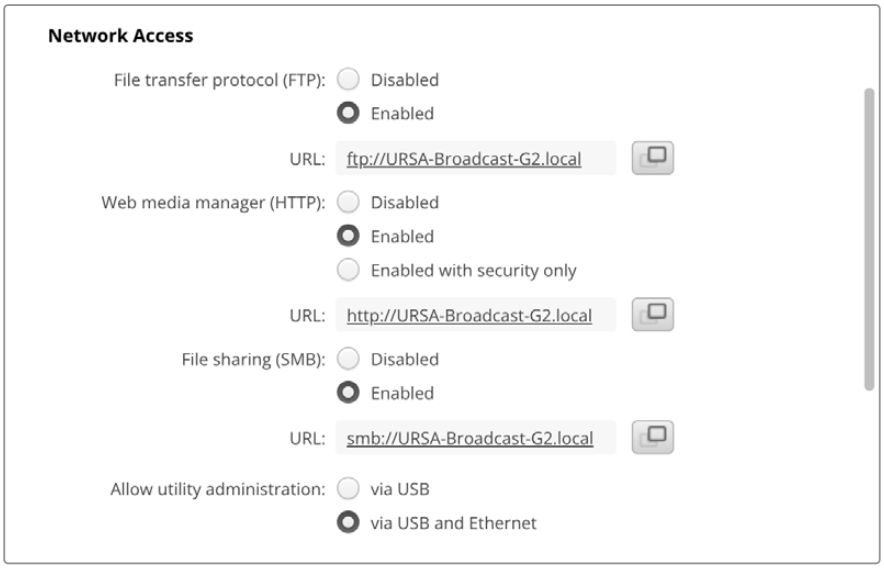
The following Blackmagic cameras are compatible with Camera Control REST API:

- Blackmagic URSA Cine 12K LF
- Blackmagic URSA Cine 17K 65
- Blackmagic URSA Cine Immersive
- Blackmagic PYXIS 6K
- Blackmagic Cinema Camera 6K
- Blackmagic URSA Broadcast G2
- Blackmagic Micro Studio Camera 4K G2
- Blackmagic Studio Camera 4K Plus
- Blackmagic Studio Camera 4K Pro
- Blackmagic Studio Camera 6K Pro
- Blackmagic Studio Camera 4K Plus G2
- Blackmagic Studio Camera 4K Pro G2

Sending API Commands

To send an API command to your camera from a third party application such as Postman, add /control/api/v1/ to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/api/v1/

You can find the Web media manager URL and IP address information in Blackmagic Camera Setup.



The Web media manager URL in Blackmagic Camera Setup

Downloading API's from your Camera

You can download REST API YAML documentation from your camera by adding /control/documentation.html to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/documentation.html

NOTE It's worth noting that changing the camera name in Blackmagic Camera Setup will also change the camera's Web media manager URL.

Livestream Control API

API for controlling Livestreams on Blackmagic Design products.

GET /livestreams/0

Get the livestream's current status.

Response

200 - Livestream's current status.

The response is JSON.

Name	Type	Description
status (required)	string	Possible values are: Idle, Connecting, Streaming, Flushing, Interrupted.
bitrate (required)	integer	Current bitrate (bps).
effectiveVideoFormat (required)	string	Effective video format for the livestream, serialised as a string.
duration	integer	Current stream duration in seconds. Absent if livestream is idle.
cache	integer	Current stream cache usage percentage.

GET /livestreams/0/start

Determine if the livestream is active.

Response

200 - Livestream active status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is active.

PUT /livestreams/0/start

Start the livestream.

Response

204 - Livestream started.

GET /livestreams/0/stop

Determine if the livestream is inactive.

Response

200 - Livestream inactive status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is inactive.

PUT /livestreams/0/stop

Stop the livestream.

Response

204 - Livestream stopped.

GET /livestreams/0/activePlatform

Get the currently selected platform configuration for the livestream.

Response

200 - Livestream active platform configuration.

The response is JSON.

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

PUT /livestreams/0/activePlatform

Set the currently selected platform configuration for the livestream.

Parameters

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

Response

204 - Livestream active platform configuration updated.

400 - Bad Request

GET /livestreams/platforms

Get the list of available platforms.

Response

200 - List of available platforms.

The response is JSON.

Name	Type	Description
	array	List of available platforms names.
[i]	string	Platform name.

GET /livestreams/platforms/{platformName}

Get the service configuration for a platform.

Parameters

Name	Type	Description
{platformName} (required)	string	Name of the platform.

Response

200 - Service configuration for specified platform.

The response is JSON.

Livestream platform service configuration.

Name	Type	Description
platform (required)	string	Corresponding platform name.
key	string	Default stream key.
servers (required)	array	List of server configurations.
servers[i]	object	Server configuration.
servers[i].server (required)	string	Server name.
servers[i].url (required)	string	Livestream destination.
servers[i].srtExtensions	array	Miscellaneous tags used for SRT livestreams.
servers[i].srtExtensions[i]	object	Dictionary object mapping SRT tag strings to values.
servers[i].srtExtensions[i][{key}]	string	SRT tag value.
servers[i].group	string	Logical grouping of the server.
profiles (required)	array	List of profile configurations.
profiles[i]	object	Quality configuration.
profiles[i].profile (required)	string	Quality level name.
profiles[i].configs (required)	array	List of video format configurations.
profiles[i].configs[i]	object	Video format configuration for profiles.
profiles[i].configs[i].resolution (required)	string	Video format serialised as a string.
profiles[i].configs[i].fps (required)	string	Frames per second.
profiles[i].configs[i].bitrate (required)	integer	Pixel bitrate (bps).
profiles[i].configs[i].audioBitrate	integer	Audio bitrate (bps).
profiles[i].configs[i].keyFrameInterval	integer	How often a key frame is sent, in seconds.
profiles[i].configs[i].videoCodecs	array	Supported video encoding algorithm/s.

Name	Type	Description
profiles[i].configs[i].videoCodecs[i]	string	Video encoding algorithm. Possible values are: H264, H265.
profiles[i].lowLatency (required)	boolean	If true, fewer frames will be buffered in the livestream.
defaultProfile	string	Quality level name.
credentials	object	Credientials used for RTMP streams.
credentials.username (required)	string	The username part of the creditials. Only used for RTMP streams.
credentials.password (required)	string	Used for RTMP streams, also used as Passphrase for SRT streams.
customizableUrlEnabled	boolean	True when the server URL is customizable.

400 - Bad Request

GET /livestreams/customPlatforms

Get a list of custom platform files.

Response

200 - List of custom platform files.

The response is JSON.

Name	Type	Description
	array	List of custom platform file names.
[i]	string	Custom platform file name.

DELETE /livestreams/customPlatforms

Remove all custom configuration files.

Response

204 - All custom configuration files removed.

GET /livestreams/customPlatforms/{filename}

Get a custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to get.

Response

200 - Custom platform file.

The response is XML.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

404 - Not Found

PUT /livestreams/customPlatforms/{filename}

Update a custom platform file if it exists, if not, create a new file with the given file name.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to update/create.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

Response

204 - Custom platform file created or updated.

400 - Bad Request

DELETE /livestreams/customPlatforms/{filename}

Remove the given custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to be removed.

Response

204 - Custom platform file removed.

404 - Not Found

Clips Control API

API for listing clips on disk.

GET /clips

Get the list of clips on the active disk.

Response

200 - List of clips on the active disk.

The response is JSON.

List of media clips.

Name	Type	Description
clips (required)	array	
clips[i]	object	Media clip.
clips[i].clipUniqueId (required)	integer	Unique ID used to identify this clip.
clips[i].filePath	string	Path to the file relative to the root of a mount.
clips[i].fileSize	integer	Size of file on disk in bytes.
clips[i].codecFormat	object	
clips[i].codecFormat.codec	string	Currently selected codec.

Name	Type	Description
clips[i].codecFormat.container	string	Multimedia container format.
clips[i].videoFormat	object	Video format configuration.
clips[i].videoFormat.name (required)	string	Video format serialised as a string.
clips[i].videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
clips[i].videoFormat.height	number	Height dimension of video format.
clips[i].videoFormat.width	number	Width dimension of video format.
clips[i].videoFormat.interlaced	boolean	Is the display format interlaced?
clips[i].startTimecode	string	Start timecode of the clip serialised as string.
clips[i].durationTimecode	string	Duration of the clip in timecode format serialised as string.
clips[i].frameCount	integer	Number of frames in clip; duration of the clip in frames.

404 - There is no active disk.

Media Pool Control API

API to manage media pool and handle uploads and project data.

GET /cloud/projects

List all projects within the media pool.

Response

200 - Successfully retrieved the list of all projects.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].libraryID	string	
[i].name	string	
[i].private	boolean	
[i].shared	boolean	
[i].clips	array	List of clips associated with the project.
[i].clips[i]	string	
[i].status	object	
[i].status.numClipsRequested	integer	
[i].status.numClipsComplete	integer	
[i].status.uploadPercent	integer	Percentage of upload completion.
[i].status.numClipsPaused	integer	
[i].status.outOfSpace	boolean	
[i].status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
[i].status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/active

Retrieve data of the actively uploading project.

Response

200 - Successfully retrieved the active project's data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/{projectID}

Retrieve specific project data by project ID.

Parameters

Name	Type	Description
{projectID} (required)	integer	Unique identifier of the project.

Response

200 - Successfully retrieved the project data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	

Name	Type	Description
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

404 - Project not found.

GET /cloud/clips

List all clips within the media pool.

Response

200 - Successfully retrieved the list of all clips.

The response is JSON.

Name	Type	Description
	array	
[i]	string	

GET /cloud/clips/activeUploading

Retrieve data of actively uploading clips.

Response

200 - Successfully retrieved the list of actively uploading clips.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].path	string	
[i].projectID	integer	
[i].status	object	
[i].status.projectID	integer	
[i].status.outOfSpace	boolean	
[i].status.proxyExtension	string	
[i].status.growingFile	boolean	
[i].status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.originalClipTotalSize	integer	
[i].status.proxyClipTotalSize	integer	
[i].status.originalClipCompletedSize	integer	
[i].status.proxyClipCompletedSize	integer	
[i].status.secsRemaining	integer	

GET /cloud/clips/{deviceName}/{path}

Retrieve specific clip data by device and path.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored.
{path} (required)	string	Path to the clip.

Response

200 - Successfully retrieved the clip data.

The response is JSON.

Name	Type	Description
path	string	
projectID	integer	
status	object	
status.projectID	integer	
status.outOfSpace	boolean	
status.proxyExtension	string	
status.growingFile	boolean	
status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.originalClipTotalSize	integer	
status.proxyClipTotalSize	integer	
status.originalClipCompletedSize	integer	
status.proxyClipCompletedSize	integer	
status.secsRemaining	integer	

404 - Clip not found.

Monitoring Control API

API for monitoring and controlling display settings in video equipment.

GET /monitoring/display

Retrieve a list of all display names.

Response

200 - Returns a list of display names.

The response is JSON.

Name	Type	Description
displays	array	List of display names available.
displays[i]	string	

GET /monitoring/{displayName}/cleanFeed

Get the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/cleanFeed

Set the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Clean feed enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/displayLUT

Get the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

400 - Invalid display name.

404 - Display name not found.

PUT /monitoring/{displayName}/displayLUT

Set the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Display LUT enabled/disabled successfully.

400 - Invalid input.

422 - Unprocessable Entity - Unable to process the contained instructions.

GET /monitoring/{displayName}/zebra

Get the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/zebra

Set the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Zebra enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/focusAssist

Get the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/focusAssist

Set the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/focusAssist

Get the focus assist settings.

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

404 - Display name not found.

PUT /monitoring/focusAssist

Set the focus assist settings.

Parameters

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/frameGuide

Get the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame guide enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGuide

Set the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame guide state updated successfully.

422 - Unable to update the frame guide state.

GET /monitoring/frameGuideRatio

Get the current frame guide ratio.

Response

200 - Returns the current frame guide ratio.

The response is JSON.

Name	Type	Description
ratio	string	The frame guide ratio.

PUT /monitoring/frameGuideRatio

Set the frame guide ratio.

Parameters

Name	Type	Description
ratio	string	The frame guide ratio.

Response

204 - Frame guide ratio updated successfully.

422 - Unable to update the frame guide ratio.

GET /monitoring/frameGuideRatio/presets

Get the presets for frame guide ratios.

Response

200 - Returns a list of preset frame guide ratios.

The response is JSON.

Name	Type	Description
presets	array	
presets[i]	string	A frame guide ratio.

GET /monitoring/{displayName}/frameGrids

Get the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame grids enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGrids

Set the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame grids state updated successfully.

422 - Unable to update the frame grids state.

GET /monitoring/frameGrids

Get the global frame grids settings.

Response

200 - Returns the current frame grids settings.

The response is JSON.

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

PUT /monitoring/frameGrids

Set the global frame grids settings.

Parameters

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

Response

204 - Frame grids settings updated successfully.

400 - Invalid input, check the number of frame grids or values.

422 - Unable to update the frame grids settings.

GET /monitoring/{displayName}/safeArea

Get the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the safe area enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/safeArea

Set the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Safe area state updated successfully.

422 - Unable to update the safe area state.

GET /monitoring/safeAreaPercent

Get the current safe area percentage.

Response

200 - Returns the current safe area percentage.

The response is JSON.

Name	Type	Description
percent	integer	Safe area coverage percentage.

PUT /monitoring/safeAreaPercent

Set the safe area percentage.

Parameters

Name	Type	Description
percent	integer	Safe area coverage percentage to set.

Response

204 - Safe area percentage updated successfully.

400 - Invalid percentage value.

422 - Unable to update the safe area percentage.

GET /monitoring/{displayName}/falseColor

Get the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the false color enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/falseColor

Set the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - False color state updated successfully.

422 - Unable to update the false color state.

Event Control API

API For working with built-in websocket.

GET /event/list

Get the list of events that can be subscribed to using the websocket API.

Response

200 - Websocket API events list.

The response is JSON.

Name	Type	Description
events	array	List of events that can be subscribed to using the websocket API.
events[i]	string	

System Control API

API for controlling the System Modes on Blackmagic Design products.

GET /system

Get device system information.

Response

200 - System summary.

The response is JSON.

The properties will be populated only with the values that are supported/implemented by the device in use.

Name	Type	Description
codecFormat	object	Codec format configuration.
codecFormat.codec	string	Codec serialised as string.
codecFormat.container	string	Multimedia container format.
videoFormat	object	Video format configuration.
videoFormat.name (required)	string	Video format serialised as a string.
videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
videoFormat.height	number	Height dimension of video format.
videoFormat.width	number	Width dimension of video format.
videoFormat.interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/product

Get device product information.

Response

200 - Device product information.

The response is JSON.

Product information.

Name	Type	Description
deviceName	string	Name of device as displayed in Setup.
productName	string	Device's product name.
softwareVersion	string	Software version running on device.

501 - This functionality is not implemented for the device in use.

GET /system/supportedCodecFormats

Get the list of supported codecs.

Response

200 - List of supported codec formats.

The response is JSON.

Name	Type	Description
codecs	array	
codecs[i]	object	Codec format configuration.
codecs[i].codec	string	Codec serialised as string.
codecs[i].container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

GET /system/codecFormat

Get the currently selected codec.

Response

200 - Current codec format.

The response is JSON.

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

PUT /system/codecFormat

Update the system codec.

Parameters

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

Response

204 - The codec updated successfully.

400 - The specified codec format is unsupported.

501 - This functionality is not implemented for the device in use.

GET /system/videoFormat

Get the currently selected video format.

Response

200 - Current system video format.

The response is JSON.

Video format configuration.

Name	Type	Description
name (required)	string	Video format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height	number	Height dimension of video format.
width	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

PUT /system/videoFormat

Set the system video format.

Parameters

This parameter can be one of the following types:

Name	Type	Description
name (required)	string	Video format serialised as a string.

Name	Type	Description
frameRate (required)	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height (required)	number	Height dimension of video format.
width (required)	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

Response

204 - The video format updated successfully.

400 - Invalid request.

409 - Operation unsupported in the current state.

501 - This functionality is not implemented for the device in use.

GET /system/supportedVideoFormats

Get the list of supported video formats for the current system state.

Response

200 - List of supported video formats.

The response is JSON.

List of supported video formats.

Name	Type	Description
formats	array	List of video formats.
formats[i]	object	Video format configuration.
formats[i].name (required)	string	Video format serialised as a string.
formats[i].frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
formats[i].height	number	Height dimension of video format.
formats[i].width	number	Width dimension of video format.
formats[i].interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/supportedFormats

Get supported formats.

Response

200 - List of supported formats.

The response is JSON.

Name	Type	Description
supportedFormats	array	
supportedFormats[i]	object	
supportedFormats[i].codecs	array	
supportedFormats[i].codecs[i]	string	
supportedFormats[i].frameRates	array	
supportedFormats[i].frameRates[i]	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
supportedFormats[i]. maxOffSpeedFrameRate	number	
supportedFormats[i]. minOffSpeedFrameRate	number	
supportedFormats[i]. recordResolution	object	
supportedFormats[i]. recordResolution.height	number	Height of the resolution.
supportedFormats[i]. recordResolution.width	number	Width of the resolution.
supportedFormats[i]. sensorResolution	object	
supportedFormats[i]. sensorResolution.height	number	Height of the resolution.
supportedFormats[i]. sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

GET /system/format

Get current format.

Response

200 - Current format.

The response is JSON.

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

PUT /system/format

Set the format.

Parameters

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

Response

204 - System format updated.

501 - This functionality is not implemented for the device in use.

Transport Control API

API for controlling Transport on Blackmagic Design products.

GET /transports/0

Get device's basic transport status.

Response

200 - Transport status.

The response is JSON.

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

PUT /transports/0

Set device's basic transport status.

Parameters

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, Output.

Response

204 - Transport mode was set.

400 - Failed to set transport mode.

GET /transports/0/stop

Determine if transport is stopped.

Response

200 - Transport stop response.

The response is JSON.

Name	Type	Description
	boolean	True when transport mode is InputPreview or when in Output mode and speed is 0.

PUT /transports/0/stop

Stop transport. Deprecated, use POST /transports/0/stop instead.

Response

204 - Transport stopped.

POST /transports/0/stop

Stop transport.

Response

204 - Transport stopped.

GET /transports/0/play

Determine if transport is playing.

Response

200 - Transport play response.

The response is JSON.

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

PUT /transports/0/play

Start playing on transport. Deprecated, use POST /transports/0/play instead.

Response

204 - Transport playing.

400 - Failed to set transport to play.

POST /transports/0/play

Start playing on transport.

Response

204 - Transport playing.

400 - Failed to set transport to play.

GET /transports/0/playback

Get playback state.

Response

200 - Transport playback state.

The response is JSON.

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

PUT /transports/0/playback

Set playback state.

Parameters

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

Response

204 - Updated transport playback state.

400 - Failed to set transport playback state.

GET /transports/0/record

Get record state.

Response

200 - Recording state.

The response is JSON.

Name	Type	Description
recording	boolean	If true, transport is in InputRecord mode.

PUT /transports/0/record

Set record state. Deprecated, use POST /transports/0/record instead.

Parameters

Name	Type	Description
recording (required)	boolean	If true, starts a recording, otherwise stops.
clipName	string	Optional, sets the requested clip name to record to, when “recording” attribute is set to true.

Response

204 - Recording state updated.

400 - Failed to update recording state.

POST /transports/0/record

Start recording.

Parameters

Name	Type	Description
clipName	string	Optional, provides a specific name of clip to record to.

Response

204 - Recording started.

400 - Failed to start recording.

GET /transports/0/clipIndex

Get the clip index of the currently playing clip on the timeline.

Response

200 - Clip index response.

The response is JSON.

Name	Type	Description
clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

GET /transports/0/timecode

Get device timecode.

Response

200 - Timecode response.

The response is JSON.

Name	Type	Description
display	string	The display timecode serialised as a string.
timeline	string	The timeline timecode serialised as a string.

GET /transports/0/timecode/source

Get timecode source selected on device.

Response

200 - Timecode source response.

The response is JSON.

Name	Type	Description
timecode	string	Possible values are: Timeline, Clip.

Timeline Control API

API for controlling playback timeline.

GET /timelines/0

Get the playback timeline.

Response

200 - Playback timeline.

The response is JSON.

Name	Type	Description
clips	array	
clips[i]	object	Timeline clip.
clips[i].clipUniqueId (required)	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).
clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

404 - No timeline / disk available.

DELETE /timelines/0

Clear the current playback timeline. Deprecated, prefer to use POST /timelines/0/clear

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

POST /timelines/0

Add a clip to the timeline.

Parameters

This parameter can be one of the following types:

Add multiple media clips to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips[i].clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips[i].clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips[i].frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add multiple media clips to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add a single clip to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips.clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips.clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips.frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add a single clip to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the timeline as specified.

501 - The operation is not supported on this device.

POST /timelines/0/add

Add a clip to the end of the timeline. Deprecated, use POST /timelines/0 to add clips within the timeline.

Parameters

This parameter can be one of the following types:

Add one clip to the end of the timeline.

Name	Type	Description
clips	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add many clips to the end of the timeline.

Name	Type	Description
clips	array	List of clipUniqueIds of clips to add to end of timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the end of the timeline.

501 - The operation is not supported on this device.

POST /timelines/0/clear

Clear the playback timeline.

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

DELETE /timelines/0/clips/{timelineClipIndex}

Remove the specified clip from the timeline.

Parameters

Name	Type	Description
{timelineClipIndex} (required)	integer	The (0-based) timeline clip index of the clip to remove from the timeline.

Response

204 - The specified clip was removed from the timeline.

501 - The operation is not supported on this device.

Media Control API

API for controlling media devices in Blackmagic Design products.

GET /media/workingset

Get the list of media devices currently in the working set.

Response

200 - The list of media devices in the working set.

The response is JSON.

Name	Type	Description
size	integer	The fixed size of this device's working set.
workingset	array	The device's working set.
workingset[i]	object | null	Device within the working set. null if no device is present within the given working set slot.
workingset[i].volume	string	Volume name.
workingset[i].deviceName	string	Internal device name of this media device.
workingset[i].remainingRecordTime	integer	Remaining record time using current codec and video format in seconds.
workingset[i].totalSpace	integer	Total space on media device in bytes.
workingset[i].remainingSpace	integer	Remaining space on media device in bytes.
workingset[i].clipCount	integer	Number of clips currently on the device.

GET /media/active

Get the currently active media device.

Response

200 - The current active media device.

The response is JSON.

The active media device, or null if there is no active media.

Name	Type	Description
workingsetIndex	integer	Working set index of the active media device.
deviceName	string	Device name of media device.

204 - No media is currently active.

PUT /media/active

Set the currently active media device.

Parameters

Name	Type	Description
workingsetIndex	integer	Working set index of the media to make active.

Response

204 - The active media device was set successfully.

400 - Setting the currently active media device is not possible in the current state.

GET /media/devices/doformatSupportedFilesystems

Get the list of filesystems available to format a media device.

Response

200 - The list of filesystems permitted for formatting.

The response is JSON.

Name	Type	Description
	array	List of filesystems permitted for formatting media.
[i]	string	Filesystem serialised as string.

GET /media/devices/{deviceName}

Get information about a requested device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Information about the requested device.

The response is JSON.

Media device state.

Name	Type	Description
state	string	The current state of the media device. Possible values are: None, Scanning, Mounted, Uninitialised, Formatting, RaidComponent.

400 - Invalid device name.

404 - Device not found.

GET /media/devices/{deviceName}/doformat

Get a format key, used to format the device with a PUT request.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Format prepared.

The response is JSON.

Name	Type	Description
deviceName	string	Device name of media device to format.
key	string	The key required to format this device, provide to PUT /media/devices/{deviceName}/doformat to perform format of media device.

400 - Cannot format the device.

404 - Device not found.

PUT /media/devices/{deviceName}/doformat

Perform a format of the specified media device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by "deviceName" member of GET /media/workingset or GET /media/active.

Name	Type	Description
key	string	The key used to format this device, retrieved from prepare format media request GET /media/devices/{deviceName}/doformat. Format key provided cannot be reused after successful format.
filesystem	string	Filesystem to format to. Supported filesystems can be retrieved with GET /media/devices/doFormatSupportedFilesystems.
volume	string	Volume name to set for the disk after format.

Response

204 - Format successful.

400 - Cannot format the device, invalid filesystem or key.

404 - Device not found.

Slate Control API

API to manage digital slate data.

GET /slates/nextClip

Retrieve the digital slate for the next clip.

Response

200 - Returns the slate data for the next clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	

Name	Type	Description
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Slate data is not available.

PUT /slates/nextClip

Update the slate data for the next clip.

Parameters

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

POST /slates/nextClip/resetProjectData

Reset the project data for the next clip's slate.

Response

200 - Project data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetProjectData

Reset the project data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Project data reset successfully.

POST /slates/nextClip/resetLensData

Reset the lens data for the next clip's slate.

Response

200 - Lens data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetLensData

Reset the lens data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Lens data reset successfully.

GET /slates/clips/{deviceName}/{path}

Retrieve slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Returns the slate data for the specified clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

404 - Clip not found.

PUT /slates/clips/{deviceName}/{path}

Update the slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.

Name	Type	Description
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

Preset Control API

API For controlling the presets on Blackmagic Design products

GET /presets

Get the list of the presets on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presets	array	List of the presets on the camera (.cset files)
presets[i]	string	

POST /presets

Send a preset file to the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presetAdded	string	Name of the preset uploaded (without .cset extension)

400 - Bad request - missing Content-Disposition header or filename

GET /presets/active

Get the currently active preset on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

PUT /presets/active

Set the active preset on the camera

Parameters

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

Response

204 - No Content

404 - Preset file not found

PARAMETERS /presets/{presetName}

GET /presets/{presetName}

Download the preset file

Response

200 - OK

The response is a binary file.

404 - File does not exist

PUT /presets/{presetName}

Save current camera state as a preset

Response

204 - No Content

DELETE /presets/{presetName}

Delete a preset from the camera

Response

204 - No Content

404 - Preset file not found

Audio Control API

API For controlling audio on Blackmagic Design Cameras

GET /audio/channels

Get the total number of audio channels available

Response

200 - Returns the total number of channels

The response is JSON.

Name	Type	Description
channels	integer	Total number of audio channels available

GET /audio/supportedInputs

Get the list of supported audio inputs

Response

200 - List of all supported audio inputs

The response is JSON.

Name	Type	Description
	array	
[i]	string	A supported audio input

GET /audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently selected input

The response is JSON.

Name	Type	Description
input	string	Audio input source and type

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/input

Set the audio input for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
input	string	Audio input source and type

Response

204 - No Content

400 - Invalid audio input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/input/description

Get the description of the current input of the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Description of the current input of the selected channel

The response is JSON.

Name	Type	Description
description	object	
description.gainRange	object	
description.gainRange.Min	number	The minimum gain value in dB
description.gainRange.Max	number	The maximum gain value in dB
description.capabilities	object	
description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter
description.capabilities.Padding	object	
description.capabilities.Padding. available	boolean	Input supports setting of padding
description.capabilities.Padding. forced	boolean	Padding is forced to be set for the input
description.capabilities.Padding. value	number	Value of the padding in dB

404 - Channel does not exist

GET /audio/channel/{channelIndex}/supportedInputs

Get the list of supported inputs and their availability to switch to for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its supported inputs are being queried. (Channels index from 0)

Response

200 - The list of supported inputs

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

404 - Channel does not exist

GET /audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set level for the selected channel

The response is JSON.

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/level

Set the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

Response

204 - No Content

400 - Invalid input or value out of range

404 - Channel does not exist

GET /audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set phantom power for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Phantom power enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/phantomPower

Set the audio phantom power for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Phantom power enabled state

Response

204 - No Content

400 - Phantom power is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set padding for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Padding enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/padding

Set the audio input padding for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Padding enabled state

Response

204 - No Content

400 - Padding is not supported or is forced for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set low cut filter for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Low cut filter enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/lowCutFilter

Set the audio input low cut filter for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Low cut filter enabled state

Response

204 - No Content

400 - Low cut filter is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set availability for the selected channel

The response is JSON.

Name	Type	Description
available	boolean	Whether the input is currently available

404 - Channel does not exist

Lens Control API

API For controlling the lens on Blackmagic Design products

GET /lens/iris

Get lens' aperture

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number

PUT /lens/iris

Set lens' aperture

Parameters

Name	Type	Description
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number
adjustmentStep	integer	Signed value for relative aperture adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens iris is not controllable or is controlled by auto exposure

GET /lens/zoom

Get lens' zoom

Response

200 - OK

The response is JSON.

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value

PUT /lens/zoom

Set lens' zoom

Parameters

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value
adjustmentFocalLength	integer	Signed value for relative focal length adjustment
adjustmentNormalised	number	Signed normalized value for relative zoom adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens zoom is not controllable

GET /lens/focus

Get lens' focus

Response

200 - OK

The response is JSON.

Name	Type	Description
normalised	number	Normalised value

PUT /lens/focus

Set lens' focus

Parameters

Name	Type	Description
normalised	number	Normalised value
focusDistance	integer	Focus distance value

Response

204 - No Content

400 - Bad Request if out of range value is provided

PUT /lens/focus/doAutoFocus

Perform auto focus

Parameters

Name	Type	Description
position	object	
position.x	number	Normalized x coordinate for autofocus ROI
position.y	number	Normalized y coordinate for autofocus ROI

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens focus is not controllable

GET /lens/opticalImageStabilization

Get optical image stabilization status

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Whether optical image stabilization is enabled

501 - Not Implemented if optical image stabilization is not supported on this product

PUT /lens/opticalImageStabilization

Enable or disable optical image stabilization

Parameters

Name	Type	Description
enabled	boolean	Enable or disable optical image stabilization

Response

204 - No Content

501 - Not Implemented if optical image stabilization is not supported on this product

GET /lens/iris/description

Get detailed description of lens' iris capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the iris can be controlled
apertureStop	object	
apertureStop.min	number	Minimum aperture stop
apertureStop.max	number	Maximum aperture stop

GET /lens/zoom/description

Get detailed description of lens' zoom capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the zoom can be controlled
focalLength	object	
focalLength.adjustable	boolean	If focal length is adjustable
focalLength.min	integer	Minimum focal length
focalLength.max	integer	Maximum focal length

GET /lens/focus/description

Get detailed description of lens' focus capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the focus can be controlled
focusDistance	object	
focusDistance.adjustable	boolean	If focus distance is adjustable
focusDistance.min	number	Minimum focus distance
focusDistance.max	number	Maximum focus distance

Video Control API

API For controlling the video on Blackmagic Design products

GET /video/iso

Get current ISO

Response

200 - OK

The response is JSON.

Name	Type	Description
iso	integer	Current ISO value

PUT /video/iso

Set current ISO

Parameters

Name	Type	Description
iso	integer	ISO value to set

Response

204 - No Content

403 - ISO cannot be changed in the current state

GET /video/supportedISOs

Get the list of supported ISO settings

Response

200 - List of supported ISO values

The response is JSON.

Name	Type	Description
supportedISOs	array	Array of supported ISO values
supportedISOs[i]	integer	

GET /video/gain

Get current gain value in decibels

Response

200 - OK

The response is JSON.

Name	Type	Description
gain	integer	Current gain value in decibels

PUT /video/gain

Set current gain value

Parameters

Name	Type	Description
gain	integer	Gain value in decibels to set

Response

204 - No Content

403 - Gain cannot be changed in the current state

GET /video/supportedGains

Get the list of supported gain settings in decibels

Response

200 - List of supported gain values in decibels

The response is JSON.

Name	Type	Description
supportedGains	array	Array of supported gain values in decibels
supportedGains[i]	integer	

GET /video/whiteBalance

Get current white balance

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	integer	Current white balance

PUT /video/whiteBalance

Set current white balance

Parameters

Name	Type	Description
whiteBalance	integer	White balance to set

Response

204 - No Content

400 - Invalid white balance temperature

GET /video/whiteBalance/description

Get white balance range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	object	
whiteBalance.min	integer	Minimum color temperature
whiteBalance.max	integer	Maximum color temperature

PUT /video/whiteBalance/doAuto

Set current white balance automatically

Response

204 - No Content

GET /video/whiteBalanceTint

Get white balance tint

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	integer	Current white balance tint

PUT /video/whiteBalanceTint

Set white balance tint

Parameters

Name	Type	Description
whiteBalanceTint	integer	White balance tint to set

Response

204 - No Content

400 - Invalid white balance tint

GET /video/whiteBalanceTint/description

Get white balance tint range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	object	
whiteBalanceTint.min	integer	Minimum white balance tint
whiteBalanceTint.max	integer	Maximum white balance tint

GET /video/ndFilter

Get ND filter stop

Response

200 - OK

The response is JSON.

Name	Type	Description
stop	number	Current filter power (fStop)

501 - Not implemented for this device

PUT /video/ndFilter

Set ND filter stop

Parameters

Name	Type	Description
stop	number	Filter power (fStop) to set

Response

204 - No Content

400 - Invalid ND filter stop

501 - Not implemented for this device

GET /video/supportedNDFilters

Get the list of available ND filter stops

Response

200 - List of available ND filter stops

The response is JSON.

Name	Type	Description
supportedStops	array	Array of available ND filter stops
supportedStops[i]	number	

501 - Not implemented for this device

GET /video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

Response

200 - List of supported display modes

The response is JSON.

Name	Type	Description
supportedDisplayModes	array	Array of supported display modes
supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

GET /video/ndFilter/displayMode

Get ND filter display mode on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

PUT /video/ndFilter/displayMode

Set ND filter display mode on the camera

Parameters

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

Response

204 - No Content

400 - Invalid display mode for ND filter

501 - Not implemented for this device

GET /video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

Response

200 - Indicates if ND filter is selectable

The response is JSON.

Name	Type	Description
selectable	boolean	True if ND filter adjustments are selectable via a slider

501 - Not implemented for this device

GET /video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

PUT /video/shutter

Set current shutter

Parameters

Name	Type	Description
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

Response

204 - No Content

GET /video/shutter/measurement

Get the current shutter measurement mode

Response

200 - OK

The response is JSON.

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

PUT /video/shutter/measurement

Set the shutter measurement mode

Parameters

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

Response

204 - No Content

400 - Invalid shutter measurement

GET /video/supportedShutters

Get supported shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of supported shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of supported shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of flicker-free shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of flicker-free shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/autoExposure

Get current auto exposure mode

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

PUT /video/autoExposure

Set auto exposure

Parameters

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

Response

204 - No Content

400 - Failed to set auto exposure mode

GET /video/detailSharpening

Get the current state of detail sharpening

Response

200 - Current detail sharpening state

The response is JSON.

Name	Type	Description
enabled	boolean	Whether detail sharpening is enabled

501 - Not implemented for this device

PUT /video/detailSharpening

Enable or disable detail sharpening

Parameters

Name	Type	Description
enabled	boolean	Enable or disable detail sharpening

Response

204 - Detail sharpening state updated

501 - Not implemented for this device

GET /video/detailSharpeningLevel

Get the current detail sharpening level

Response

200 - Current detail sharpening level

The response is JSON.

Name	Type	Description
level	string	Current detail sharpening level Possible values are: Low, Medium, High.

501 - Not implemented for this device

PUT /video/detailSharpeningLevel

Set the detail sharpening level

Parameters

Name	Type	Description
level	string	Desired level of detail sharpening Possible values are: Low, Medium, High.

Response

204 - Detail sharpening level updated

400 - Invalid detail sharpening level

501 - Not implemented for this device

Camera Control API

API For controlling the Camera specific features on Blackmagic Design products

GET /camera/colorBars

Get the status of color bars display

Response

200 - Returns the current status of color bars

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the color bars are currently enabled

PUT /camera/colorBars

Set the status of color bars display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the color bars

Response

204 - Color bars status updated successfully

GET /camera/programFeedDisplay

Get the status of program feed display

Response

200 - Returns the current status of program feed display

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the program feed display is currently enabled

PUT /camera/programFeedDisplay

Set the status of program feed display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the program feed display

Response

204 - Program feed display status updated successfully

GET /camera/tallyStatus

Get the tally status of the camera

Response

200 - Returns the current tally status of the camera

The response is JSON.

Name	Type	Description
status	string	Current tally status of the camera Possible values are: None, Preview, Program.

GET /camera/power

Get the power status of the camera

Response

200 - Returns the current power status

The response is JSON.

Name	Type	Description
source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
batteries	array	List of batteries currently connected to the camera
batteries[i]	object	
batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
batteries[i].chargeRemainingPercent	integer	Remaining battery charge percentage
batteries[i].statusFlags	array	List of battery status flags
batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

GET /camera/power/displayMode

Get the power display mode of the camera

Response

200 - Returns the current power display mode

The response is JSON.

Name	Type	Description
mode	string	Current power display mode Possible values are: Percentage, Voltage.

PUT /camera/power/displayMode

Set the power display mode of the camera

Parameters

Name	Type	Description
mode	string	Power display mode to set Possible values are: Percentage, Voltage.

Response

204 - Power display mode updated successfully

400 - Invalid power display mode

GET /camera/timingReferenceLock

Get the timing reference lock status

Response

200 - Returns the timing reference lock status

The response is JSON.

Name	Type	Description
locked	boolean	Indicates if timing reference is locked

Immersive Control API

API for controlling immersive camera settings on Blackmagic Design cameras

GET /immersive/display/{displayName}/eye

Get the current eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to query (from /monitoring/display endpoint)

Response

200 - OK

The response is JSON.

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

400 - Invalid display name format

404 - Display not found

422 - Failed to get eye view

PUT /immersive/display/{displayName}/eye

Set the eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to control (from /monitoring/display endpoint)

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

Response

204 - No Content

400 - Invalid input or display name format

404 - Display not found

422 - Failed to set eye view

Color Correction Control API

API For controlling the color correction on Blackmagic Design products based on DaVinci Resolve Color Corrector

GET /colorCorrection/lift

Get color correction lift

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

PUT /colorCorrection/lift

Set color correction lift

Parameters

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gamma

Get color correction gamma

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

PUT /colorCorrection/gamma

Set color correction gamma

Parameters

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gain

Get color correction gain

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

PUT /colorCorrection/gain

Set color correction gain

Parameters

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/offset

Get color correction offset

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

PUT /colorCorrection/offset

Set color correction offset

Parameters

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/contrast

Get color correction contrast

Response

200 - OK

The response is JSON.

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/contrast

Set color correction contrast

Parameters

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/color

Get color correction color properties

Response

200 - OK

The response is JSON.

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/color

Set color correction color properties

Parameters

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/lumaContribution

Get color correction luma contribution

Response

200 - OK

The response is JSON.

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

PUT /colorCorrection/lumaContribution

Set color correction luma contribution

Parameters

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

Response

204 - No Content

Notification websocket - 1.0.0

Service that notifies subscribers of device state changes.

messages

Subscribe (The messages from the server/device)

Websocket Opened Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: websocketOpened.
.type	string	Possible values are: event.

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*

Name	Type	Description
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Event Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: propertyValueChanged.

Name	Type	Description
.data.property	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.value	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.type	string	Possible values are: event.

Publish (The messages that user can send to the server/device)

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	

Name	Type	Description
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	

Name	Type	Description
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Device Properties

/media/workingset

The value JSON returned via the eventResponse when the `/media/workingset` property changes on the device:

Name	Type	Description
.size	integer	The fixed size of this device's working set.
.workingset	array	Array of devices within the working set. null if no device is present within the given working set slot.
.workingset[i]		

/media/active

The value JSON returned via the eventResponse when the `/media/active` property changes on the device:

Name	Type	Description
.workingsetIndex	integer	Working set index of the active media device.
.deviceName	string	Internal device name of this media device.

/system

The value JSON returned via the eventResponse when the `/system` property changes on the device:

Name	Type	Description
.codecFormat	object	Codec format configuration.
.codecFormat.codec	string	Codec format serialised as a string.
.codecFormat.container	string	Multimedia container format.
.videoFormat	object	Currently selected video format.
.videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.videoFormat.height	number	Height dimension of video format.
.videoFormat.width	number	Width dimension of video format.
.videoFormat.interlaced	boolean	Is the display format interlaced?.
.videoFormat.name	string	Video format serialised as a string.

/system/codecFormat

Codec format configuration.

The value JSON returned via the eventResponse when the `/system/codecFormat` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.container	string	Multimedia container format.

/system/videoFormat

Currently selected video format.

The value JSON returned via the eventResponse when the `/system/videoFormat` property changes on the device:

Name	Type	Description
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.height	number	Height dimension of video format.
.width	number	Width dimension of video format.
.interlaced	boolean	Is the display format interlaced?.
.name	string	Video format serialised as a string.

/system/format

The value JSON returned via the eventResponse when the `/system/format` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.maxOffSpeedFrameRate	number	
.minOffSpeedFrameRate	number	
.offSpeedEnabled	boolean	
.offSpeedFrameRate	number	
.recordResolution	object	
.recordResolution.height	number	Height of the resolution.
.recordResolution.width	number	Width of the resolution.
.sensorResolution	object	
.sensorResolution.height	number	Height of the resolution.
.sensorResolution.width	number	Width of the resolution.

/system/supportedFormats

The value JSON returned via the eventResponse when the `/system/supportedFormats` property changes on the device:

Name	Type	Description
.supportedFormats	array	
.supportedFormats[i]	object	
.supportedFormats[i].codecs	array	
.supportedFormats[i].codecs[i]	string	
.supportedFormats[i].frameRates	array	
.supportedFormats[i].frameRates[i]	string	Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.supportedFormats[i].maxOffSpeedFrameRate	number	
.supportedFormats[i].minOffSpeedFrameRate	number	
.supportedFormats[i].recordResolution	object	
.supportedFormats[i].recordResolution.height	number	Height of the resolution.
.supportedFormats[i].recordResolution.width	number	Width of the resolution.
.supportedFormats[i].sensorResolution	object	
.supportedFormats[i].sensorResolution.height	number	Height of the resolution.
.supportedFormats[i].sensorResolution.width	number	Width of the resolution.

/timelines/0

The value JSON returned via the eventResponse when the `/timelines/0` property changes on the device:

Name	Type	Description
.clips	array	
.clips[i]	object	Timeline clip.
.clips[i].clipUniqueId	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
.clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
.clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
.clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
.clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).

Name	Type	Description
.clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
.clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

/transports/0

The value JSON returned via the eventResponse when the `/transports/0` property changes on the device:

Name	Type	Description
.mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

/transports/0/stop

true when transport mode is InputPreview or when in Output mode and speed is 0.

The value JSON returned via the eventResponse when the `/transports/0/stop` property changes on the device:

Name	Type	Description
	boolean	true when transport mode is InputPreview or when in Output mode and speed is 0.

/transports/0/play

True when transport is in Output mode and speed is non-zero.

The value JSON returned via the eventResponse when the /transports/0/play property changes on the device:

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

/transports/0/playback

The value JSON returned via the eventResponse when the `/transports/0/playback` property changes on the device:

Name	Type	Description
.type	string	Possible values are: Play, Jog, Shuttle, Var.
.loop	boolean	When true playback loops from the end of the timeline to the beginning of the timeline.
.singleClip	boolean	When true playback loops from the end of the current clip to the beginning of the current clip.
.speed	number	Playback speed, 1.0 for normal forward playback
.position	integer	Playback position on the timeline in units of video frames.

/transports/0/record

The value JSON returned via the eventResponse when the `/transports/0/record` property changes on the device:

Name	Type	Description
.recording	boolean	Is transport in Input Record mode.

/transports/0/timecode

The value JSON returned via the eventResponse when the `/transports/0/timecode` property changes on the device:

Name	Type	Description
.display	string	The display timecode serialised as a string.
.timeline	string	The timeline timecode serialised as a string.

/transports/0/timecode/source

The value JSON returned via the eventResponse when the `/transports/0/timecode/source` property changes on the device:

Name	Type	Description
.timecode	string	Possible values are: Timeline, Clip.

/transports/0/clipIndex

The value JSON returned via the eventResponse when the `/transports/0/clipIndex` property changes on the device:

Name	Type	Description
.clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

/slates/nextClip

The value JSON returned via the eventResponse when the `/slates/nextClip` property changes on the device:

Name	Type	Description
.clip	object	
.clip.clipName	string	
.clip.reel	integer	
.clip.scene	string	
.clip.sceneLocation	string	Possible values are: Interior, Exterior.
.clip.sceneTime	string	Possible values are: Day, Night.
.clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
.clip.take	integer	
.clip.takeType	string	Possible values are: None, PU, VFX, SER.
.clip.goodTake	boolean	
.lens	object	
.lens.lensType	string	
.lens.iris	string	

Name	Type	Description
.lens.focalLength	string	
.lens.distance	string	
.lens.filter	string	
.project	object	
.project.projectName	string	
.project.director	string	
.project.camera	string	
.project.cameraOperator	string	

/monitoring/{displayName}/cleanFeed

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/cleanFeed` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/displayLUT

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/displayLUT` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/zebra

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/zebra` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/focusAssist` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGuide

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGuide` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGrids` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/safeArea

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/safeArea` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/falseColor

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/falseColor` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/focusAssist` property changes on the device:

Name	Type	Description
.mode	string	Possible values are: Peak, ColoredLines.
.color	string	Possible values are: Red, Green, Blue, White, Black.
.intensity	integer	

/monitoring/frameGuideRatio

The value JSON returned via the eventResponse when the `/monitoring/frameGuideRatio` property changes on the device:

Name	Type	Description
.ratio	string	

/monitoring/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/frameGrids` property changes on the device:

Name	Type	Description
.frameGrids	array	
.frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

/monitoring/safeAreaPercent

The value JSON returned via the eventResponse when the `/monitoring/safeAreaPercent` property changes on the device:

Name	Type	Description
.percent	integer	Safe area coverage percentage.

/audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input` property changes on the device:

Name	Type	Description
.input	string	Audio input source and type

/audio/channel/{channelIndex}/supportedInputs

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/supportedInputs` property changes on the device:

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

/audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/level` property changes on the device:

Name	Type	Description
.gain	number	Gain value in dB
.normalised	number	Normalised level value between 0.0 and 1.0

/audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/phantomPower` property changes on the device:

Name	Type	Description
.enabled	boolean	Phantom power enabled state

/audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/padding` property changes on the device:

Name	Type	Description
.enabled	boolean	Padding enabled state

/audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/lowCutFilter` property changes on the device:

Name	Type	Description
.enabled	boolean	Low cut filter enabled state

/audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/available` property changes on the device:

Name	Type	Description
.available	boolean	Whether the input is currently available

/audio/channel/{channelIndex}/input/description

Description of the current input of the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input/description` property changes on the device:

Name	Type	Description
.description	object	
.description.gainRange	object	
.description.gainRange.Min	number	The minimum gain value in dB
.description.gainRange.Max	number	The maximum gain value in dB
.description.capabilities	object	
.description.capabilities.PhantomPower	boolean	Input supports setting of phantom power
.description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter

Name	Type	Description
.description.capabilities.Padding	object	
.description.capabilities.Padding.available	boolean	Input supports setting of padding
.description.capabilities.Padding.forced	boolean	Padding is forced to be set for the input
.description.capabilities.Padding.value	number	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.

/colorCorrection/lift

Get color correction lift

The value JSON returned via the eventResponse when the `/colorCorrection/lift` property changes on the device:

Name	Type	Description
.red	number	Red lift component
.green	number	Green lift component
.blue	number	Blue lift component
.luma	number	Luma lift component

/colorCorrection/gamma

Get color correction gamma

The value JSON returned via the eventResponse when the `/colorCorrection/gamma` property changes on the device:

Name	Type	Description
.red	number	Red gamma component
.green	number	Green gamma component
.blue	number	Blue gamma component
.luma	number	Luma gamma component

/colorCorrection/gain

Get color correction gain

The value JSON returned via the eventResponse when the `/colorCorrection/gain` property changes on the device:

Name	Type	Description
.red	number	Red gain component
.green	number	Green gain component
.blue	number	Blue gain component
.luma	number	Luma gain component

/colorCorrection/offset

Get color correction offset

The value JSON returned via the eventResponse when the `/colorCorrection/offset` property changes on the device:

Name	Type	Description
.red	number	Red offset component
.green	number	Green offset component
.blue	number	Blue offset component
.luma	number	Luma offset component

/colorCorrection/contrast

Get color correction contrast

The value JSON returned via the eventResponse when the `/colorCorrection/contrast` property changes on the device:

Name	Type	Description
.pivot	number	Contrast pivot point
.adjust	number	Contrast adjustment

/colorCorrection/color

Get color correction color properties

The value JSON returned via the eventResponse when the `/colorCorrection/color` property changes on the device:

Name	Type	Description
.hue	number	Color hue adjustment
.saturation	number	Color saturation adjustment

/colorCorrection/lumaContribution

Get color correction luma contribution

The value JSON returned via the eventResponse when the `/colorCorrection/lumaContribution` property changes on the device:

Name	Type	Description
.lumaContribution	number	Luma contribution value

/lens/iris

Get lens' aperture

The value JSON returned via the eventResponse when the `/lens/iris` property changes on the device:

Name	Type	Description
.continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
.apertureStop	number	Aperture stop value
.normalised	number	Normalised value
.apertureNumber	integer	Aperture number

/lens/iris/description

Get detailed description of lens' iris capabilities

The value JSON returned via the eventResponse when the `/lens/iris/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the iris can be controlled
.apertureStop	object	
.apertureStop.min	number	Minimum aperture stop
.apertureStop.max	number	Maximum aperture stop

/lens/focus

Get lens' focus

The value JSON returned via the eventResponse when the `/lens/focus` property changes on the device:

Name	Type	Description
.normalised	number	Normalised value

/lens/focus/description

Get detailed description of lens' focus capabilities

The value JSON returned via the eventResponse when the `/lens/focus/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the focus can be controlled
.focusDistance	object	
.focusDistance.adjustable	boolean	If focus distance is adjustable
.focusDistance.min	number	Minimum focus distance
.focusDistance.max	number	Maximum focus distance

/lens/zoom

Get lens' zoom

The value JSON returned via the eventResponse when the `/lens/zoom` property changes on the device:

Name	Type	Description
.focalLength	integer	Focal length in mm
.normalised	number	Normalised value

/lens/zoom/description

Get detailed description of lens' zoom capabilities

The value JSON returned via the eventResponse when the `/lens/zoom/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the zoom can be controlled
.focalLength	object	
.focalLength.adjustable	boolean	If focal length is adjustable
.focalLength.min	integer	Minimum focal length
.focalLength.max	integer	Maximum focal length

/presets

Get the list of the presets on the camera

The value JSON returned via the eventResponse when the `/presets` property changes on the device:

Name	Type	Description
.presets	array	List of the presets on the camera (.cset files)
.presets[i]	string	

/presets/active

Get the currently active preset on the camera

The value JSON returned via the eventResponse when the `/presets/active` property changes on the device:

Name	Type	Description
.preset	string	Name of the active preset (with .cset extension, or 'default')

/camera/colorBars

Get the status of color bars display

The value JSON returned via the eventResponse when the `/camera/colorBars` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the color bars are currently enabled

/camera/programFeedDisplay

Get the status of program feed display

The value JSON returned via the eventResponse when the `/camera/programFeedDisplay` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the program feed display is currently enabled

/camera/tallyStatus

Get the tally status of the camera

The value JSON returned via the eventResponse when the `/camera/tallyStatus` property changes on the device:

Name	Type	Description
.status	string	Current tally status of the camera Possible values are: None, Preview, Program.

/camera/power

Get the power status of the camera

The value JSON returned via the eventResponse when the `/camera/power` property changes on the device:

Name	Type	Description
.source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
.milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
.batteries	array	
.batteries[i]	object	
.batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
.batteries[i]. chargeRemainingPercent	integer	Remaining battery charge percentage
.batteries[i].statusFlags	array	List of battery status flags
.batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

/camera/power/displayMode

Get the power display mode of the camera

The value JSON returned via the eventResponse when the `/camera/power/displayMode` property changes on the device:

Name	Type	Description
.mode	string	Current power display mode Possible values are: Percentage, Voltage.

/camera/timingReferenceLock

Get the timing reference lock status

The value JSON returned via the eventResponse when the `/camera/timingReferenceLock` property changes on the device:

Name	Type	Description
.locked	boolean	Indicates if timing reference is locked

/video/iso

Get current ISO

The value JSON returned via the eventResponse when the `/video/iso` property changes on the device:

Name	Type	Description
.iso	integer	Current ISO value

/video/supportedISOs

Get the list of supported ISO settings

The value JSON returned via the eventResponse when the `/video/supportedISOs` property changes on the device:

Name	Type	Description
.supportedISOs	array	Array of supported ISO values
.supportedISOs[i]	integer	

/video/gain

Get current gain value in decibels

The value JSON returned via the eventResponse when the `/video/gain` property changes on the device:

Name	Type	Description
.gain	integer	Current gain value in decibels

/video/supportedGains

Get the list of supported gain settings in decibels

The value JSON returned via the eventResponse when the `/video/supportedGains` property changes on the device:

Name	Type	Description
.supportedGains	array	Array of supported gain values in decibels
.supportedGains[i]	integer	

/video/whiteBalance

Get current white balance

The value JSON returned via the eventResponse when the `/video/whiteBalance` property changes on the device:

Name	Type	Description
.whiteBalance	integer	Current white balance

/video/whiteBalance/description

Get white balance range

The value JSON returned via the eventResponse when the `/video/whiteBalance/description` property changes on the device:

Name	Type	Description
.whiteBalance	object	
.whiteBalance.min	integer	Minimum color temperature
.whiteBalance.max	integer	Maximum color temperature

/video/whiteBalanceTint

Get white balance tint

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint` property changes on the device:

Name	Type	Description
.whiteBalanceTint	integer	Current white balance tint

/video/whiteBalanceTint/description

Get white balance tint range

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint/description` property changes on the device:

Name	Type	Description
.whiteBalanceTint	object	
.whiteBalanceTint.min	integer	Minimum white balance tint
.whiteBalanceTint.max	integer	Maximum white balance tint

/video/ndFilter

Get ND filter stop

The value JSON returned via the eventResponse when the `/video/ndFilter` property changes on the device:

Name	Type	Description
.stop	number	Current filter power (fStop)

/video/supportedNDFilters

Get the list of available ND filter stops

The value JSON returned via the eventResponse when the `/video/supportedNDFilters` property changes on the device:

Name	Type	Description
.supportedStops	array	Array of available ND filter stops
.supportedStops[i]	number	

/video/ndFilter/displayMode

Get ND filter display mode on the camera

The value JSON returned via the eventResponse when the `/video/ndFilter/displayMode` property changes on the device:

Name	Type	Description
.displayMode	string	ND filter display mode Possible values are: Stop, Number, Fraction.

/video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

The value JSON returned via the eventResponse when the `/video/supportedNDFilterDisplayModes` property changes on the device:

Name	Type	Description
.supportedDisplayModes	array	Array of supported display modes
.supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

/video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

The value JSON returned via the eventResponse when the `/video/ndFilterSelectable` property changes on the device:

Name	Type	Description
.selectable	boolean	True if ND filter adjustments are selectable via a slider

/video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

The value JSON returned via the eventResponse when the `/video/shutter` property changes on the device:

Name	Type	Description
.continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
.shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
.shutterAngle	number	Shutter angle

/video/shutter/measurement

Get the current shutter measurement mode

The value JSON returned via the eventResponse when the `/video/shutter/measurement` property changes on the device:

Name	Type	Description
.measurement	string	Current shutter measurement mode Possible values are: ShutterAngle, ShutterSpeed.

/video/supportedShutters

Get supported shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/supportedShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of supported shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array of flicker-free shutter speeds
.shutterSpeeds[i]	integer	

/video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/flickerFreeShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of flicker-free shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array
.shutterSpeeds[i]	integer	

/video/autoExposure

Get current auto exposure mode

The value JSON returned via the eventResponse when the `/video/autoExposure` property changes on the device:

Name	Type	Description
.mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
.type	string	Comma-separated list of device types in the auto exposure stack

/video/detailSharpening

Get the current state of detail sharpening

The value JSON returned via the eventResponse when the `/video/detailSharpening` property changes on the device:

Name	Type	Description
.enabled	boolean	Whether detail sharpening is enabled

/video/detailSharpeningLevel

Get the current detail sharpening level

The value JSON returned via the eventResponse when the `/video/detailSharpeningLevel` property changes on the device:

Name	Type	Description
.level	string	Current detail sharpening level of supported shutter speeds Possible values are: Low, Medium, High.

도움말

가장 빠르게 지원 받을 수 있는 방법은 Blackmagic Design 온라인 고객지원 페이지에 접속하여 카메라 관련 최신 지원 정보를 이용하는 것입니다.

Blackmagic Design 온라인 고객 지원 페이지

최신 사용 설명서와 소프트웨어, 지원 노트는 Blackmagic 고객 지원 센터에서 확인하실 수 있습니다.
www.blackmagicdesign.com/kr/support

Blackmagic Design 고객 지원에 문의하기

고객지원 페이지에서 원하는 정보를 얻지 못한 경우에는 '고객지원' 페이지에 있는 '이메일 보내기' 버튼을 클릭하여 지원 요청 이메일을 보내주세요. 다른 방법으로는, '고객지원' 페이지의 '지역별 고객 지원팀 찾기' 버튼을 클릭하여 가장 가까운 Blackmagic Design 고객지원 사무실에 문의하세요.

현재 설치된 소프트웨어 버전 확인하기

컴퓨터에 설치된 Blackmagic Camera Setup 유틸리티의 소프트웨어 버전을 확인하려면 'About Blackmagic Camera Setup' 창을 여세요.

- Mac에서는 '애플리케이션' 폴더에 있는 Blackmagic Camera Setup 유틸리티를 실행하세요. '애플리케이션' 메뉴에서 About Blackmagic Camera Setup 유틸리티를 선택하면 버전 번호가 나타납니다.
- Windows에서는 '시작 메뉴' 또는 시작 스크린에서 Blackmagic Camera Setup 유틸리티를 실행합니다. '도움말' 메뉴를 클릭한 뒤 About Blackmagic Camera Setup 유틸리티를 선택하고 버전을 확인하세요.

최신 버전 소프트웨어로 업데이트하기

컴퓨터에 설치된 Blackmagic Camera Utility 소프트웨어 버전을 확인한 뒤, Blackmagic 고객 지원 센터 (www.blackmagicdesign.com/kr/support)를 방문하여 최신 업데이트를 확인하세요. 최신 버전으로 업데이트하는 것을 권장하지만, 중요한 프로젝트를 실행하는 도중에는 소프트웨어 업데이트를 하지 않는 것이 좋습니다.

규제 사항

유럽 연합 국가 내의 전기전자제품 폐기물 처리 기준.

제품에 부착된 기호는 해당 제품을 다른 폐기물과는 별도로 처리되어야 함을 나타냅니다. 제품을 폐기하려면 반드시 재활용 지정 수거 장소에 폐기해야 합니다. 폐기물 제품을 분리수거 및 재활용으로 처리하는 것은 자연 자원을 보전하고 인간의 건강과 환경을 보호할 수 있도록 폐기물을 재활용할 수 있는 방법입니다. 재활용을 위한 제품 폐기물 장소에 관한 자세한 정보는 해당 지역 시청의 재활용 센터 혹은 해당 제품을 구입한 상점으로 문의하세요.

본 제품은 테스트 결과 FCC 규정 제15항에 따라 A급 디지털 기기 제한 사항을 준수하는 것으로 확인되었습니다. 해당 제한 사항은 본 제품을 상업적 환경에서 사용할 시 발생할 수 있는 유해 혼선으로부터 적절한 보호를 제공하기 위함입니다. 이 제품은 무선 주파수를 생성 및 사용, 방출할 수 있습니다. 따라서 설명서의 안내에 따라 제품을 설치 및 사용하지 않을 시, 무선 통신을 방해하는 전파 혼선을 일으킬 수 있습니다. 해당 제품을 주거 지역에서 사용할 경우, 유해 전파 혼선이 발생할 가능성이 있으며 이 경우, 사용자는 자체 비용으로 전파 혼선 문제를 해결해야 합니다.

제품 작동은 다음 두 가지 조건을 전제로 합니다.

1. 본 기기는 유해 혼신을 일으키지 않습니다.
2. 본 기기는 원치 않는 작동을 일으킬 수 있는 혼신 등의 모든 혼신을 수용합니다.

R-R-BMD-20250213001

ICES-3 (A)
NMB-3 (A)

ISED 캐나다 성명

본 기기는 캐나다 표준 A급 디지털 장치 규정을 준수합니다.

정해진 사용 목적 이외의 다른 목적의 사용 또는 제품 변경은 표준 규정 위반으로 간주할 수 있습니다.

HDMI 인터페이스 연결 시에는 반드시 고품질의 쉴드 HDMI 케이블을 사용해야 합니다.

이 기기는 업무용 환경에서 사용할 목적으로 적합성 평가를 거쳤습니다. 가정용 환경에서 사용하는 경우 전파간섭의 우려가 있습니다.

안전 정보

Blackmagic URSA Cine Immersive와 함께 제공된 AC-24V DC 전원 공급 장치는 보호 접지 연결이 있는 주 전원 콘센트에 연결해야 합니다.

Blackmagic URSA Cine Immersive의 내부에는 사용자가 수리할 수 있는 부품이 없습니다. 제품 수리는 해당 지역 Blackmagic Design 서비스 센터에 문의하세요.

입력 전원 커넥터를 외부 DC 배터리 소스에 연결할 때, 공급 배선에는 전류 제한 또는 퓨즈 장치가 포함되어야 합니다.

사용되는 배선은 VW-1로 표시되어 있거나 IEC 60332 또는 IEC 60695의 관련 부분을 준수해야 합니다.

이 카메라의 DC 전원은 24V를 권장합니다. 12V DC 전원을 사용하는 경우, 14A 전류를 충분히 감당할 수 있는 배선을 사용해야 합니다.

이 제품은 B-마운트 배터리 플레이트와 함께 제공됩니다. 24V(28.8V) 배터리 또는 듀얼 전압 배터리 사용을 권장합니다.

12V(14.4V) 배터리의 경우, 배터리 설명서나 표시를 참조하여 최대 연속 방전 전류를 확인하세요. 최소 12A 배터리 사용을 권장합니다.

캘리포니아주 성명

본 제품을 사용하는 사용자는 제품의 플라스틱 내 폴리브롬화 비페닐에 노출될 수 있으며 캘리포니아주에서는 해당 물질이 암, 선천적 결손증, 기타 생식기능의 손상을 유발하는 것으로 알려져 있습니다.

더욱 자세한 정보는 www.P65Warnings.ca.gov를 확인하세요.

유럽 사무실

Blackmagic Design Europe B.V.
Rijnlanderweg 766, Unit D
2132 NM Hoofddorp
NL

Blackmagic URSA Cine

일광 조건에서는 카메라나 리튬 배터리가 햇빛에 장시간 노출되지 않도록 그늘에 보관하세요. 리튬 배터리에 열기가 닿지 않도록 주의하세요.

24V ‘RS’ 출력 커넥터는 서드파티 액세서리에 전원을 공급하기 적합합니다. 전력 소비량이 48W 미만인지 확인하세요.

보증

한정 보증

Blackmagic Design은 본 제품의 부품 및 제조에 어떠한 결함도 없음을 제품 구매일로부터 12개월 동안 보증합니다. 보증 기간 내에 결함이 발견될 경우, Blackmagic Design은 당사의 결정에 따라 무상 수리 또는 새로운 제품으로 교환해드립니다.

구매 고객은 반드시 보증 기간이 만료되기 전에 결함 사실을 Blackmagic Design에 통지해야 적절한 보증 서비스를 제공받을 수 있습니다. 구매 고객은 지정된 Blackmagic Design 서비스 센터로 결함 제품을 포장 및 운송할 책임이 있으며, 운송 비용은 선불로 지급되어야 합니다. 구매 고객은 또한 이유를 불문하고 제품 반송에 대한 운송료, 보험, 관세, 세금, 기타 비용을 부담해야 합니다.

이 보증은 부적절한 사용, 관리 및 취급으로 인한 파손, 고장, 결함에는 적용되지 않습니다. Blackmagic Design은 다음과 같은 경우에 보증 서비스를 제공할 의무가 없습니다. a) Blackmagic Design 판매 대리인이 아닌 개인에 의해 발생한 제품 손상. b) 부적절한 사용 및 호환하지 않는 장비와의 연결로 인한 제품 손상. c) Blackmagic Design사의 부품 및 공급품이 아닌 것을 사용하여 발생한 손상 및 고장. d) 제품을 개조하거나 다른 제품과 통합하여 제품 작동 시간 증가 및 기능 저하가 발생한 경우.

URSA Viewfinder를 직사광선에 노출시키면 뷰파인더 광학 장치가 돋보기 역할을 하기 때문에 뷰파인더 디스플레이가 손상될 수 있습니다. 프레임 가이드와 같은 고정 이미지 또는 고대비 이미지가 패널에 장기간 표시되면 OLED 패널에서 잔상 또는 번인이 발생할 수 있습니다. 이를 방지하기 위해 얼굴 감지용 IR 센서가 다른 물체에 가려지지 않도록 하며, 장기간 사용하지 않을 때는 뷰파인더를 분리하세요. 이미지 유지 성능은 이 제품 보증에 포함되지 않습니다.

Blackmagic Design에서 제공하는 제품 보증은 다른 모든 명시적 또는 묵시적 보증을 대신합니다. Blackmagic Design과 관련 판매 회사는 상품성 및 특정 목적의 적합성과 관련된 모든 묵시적 보증을 부인합니다. Blackmagic Design의 결함 제품 수리 및 교환 관련 책임은 구매 고객에게 제공되는 유일한 배상 수단입니다. Blackmagic Design은 자사 또는 판매 회사에서 관련 위험의 소지에 대한 사전 통보의 여부와 관계없이 모든 간접적, 특별, 우발적, 결과적 손해에 대한 책임을 지지 않습니다. Blackmagic Design은 장비의 불법적 사용과 관련하여 어떤 법적 책임도 지지 않습니다. Blackmagic Design은 본 제품의 사용으로 인해 발생하는 손해에 대해서는 어떤 법적 책임도 지지 않습니다. 제품 사용으로 인해 발생할 수 있는 위험에 대한 책임은 본인에게 있습니다.

© Copyright 2025 Blackmagic Design. 모든 저작권은 Blackmagic Design에 있습니다. 'Blackmagic Design', 'URSA', 'DeckLink', 'HDLink', 'Workgroup Videohub', 'Multibridge Pro', 'Multibridge Extreme', 'Intensity', 'Leading the creative video revolution'은 모두 미국 및 기타 국가에 등록된 상표입니다. 모든 다른 회사명 및 제품 이름은 관련 회사의 등록 상표일 수 있습니다.

Bluetooth® 워드마크와 로고는 Bluetooth SIG, Inc. 소유의 등록상표이며, Blackmagic Design은 허가에 따라 이 상표를 사용합니다. 다른 등록상표 및 상품명은 해당 업체에서 소유하는 자산입니다.

Октябрь 2025 г.

Руководство по работе с камерой

Blackmagicdesign

Blackmagic URSA Cine Immersive



Содержание

Blackmagic URSA Cine Immersive 1069
Свойства иммерсивного видео 1069
Apple Immersive Video 1069
URSA Cine Immersive 1070
Подготовка к работе 1071
Подключение питания 1071
Установка настроек записи 1072
Мониторинг изображения 1074
Опции мониторинга и вывод параметров 1074
Кадрирование первого плана 1075
Высота камеры 1075
Наклон камеры вверх или вниз 1076
Позиционирование 1077
Свобода зрителя 1078
Движение камеры 1078
Скорость передачи данных 1079
Звук и тайм-код 1080
Формат Apple Spatial Audio (ASAF) 1080
Объемный звук 1080
Петличные и миниатюрные микрофоны 1081
Ручные микрофоны 1081
Подключение тайм-кода 1081
Съемка 1083
Требования к экспозиции 1083
Выдержка затвора 1083
ISO и динамический диапазон 1084
Светофильтры ND 1085
Изменение скорости записи 1086
Гистограмма 1087
Опции мониторинга на ЖК-дисплее 1087
Запись 1088
Воспроизведение 1089
Передача клипов 1090
Заключение 1091
DaVinci Resolve Studio 1092
Рекомендуемые рабочие станции 1092
Настройки проекта 1093
Основные настройки 1093
Вывод на видеомонитор 1094
Масштабирование изображения 1094
Управление цветом 1095
Пространство DaVinci Wide Gamut 1096
Обработка контента в формате Apple Immersive 1097
Загрузка 1097
Монтаж 1099
Сборка 1099
Оформление фона 1099
Окно просмотра 1101
Переходы 1103
Данные о движении Apple Immersive Motion Data 1104
Титры 1105
Творческие соображения 1106
Композитинг с использованием визуальных эффектов 1107
Узлы MediaOut 1107
Узел Renderer3D 1108
Композитинг с помощью инструмента Immersive Patcher 1109
Цветокоррекция 1110
Шумоподавление 1110
Первичный грейдинг 1111
Выборочная цветоустановка 1111
Управление размером 1112
Панель 3D-инструментов 1112
Управление цветом в CST-режиме 1114
Назначение временной шкалы в CST-режиме 1114
Настройка монитора для грейдинга в CST-режиме 1115
Настройка параметров при мониторинге в гарнитуре Apple Vision Pro в CST-режиме 1117
Аудио 1118
Формат Apple Spatial Audio (ASAF) 1118
Конечный продукт 1119
Работа с переходами 1119
Стриминг на гарнитуру Apple Vision Pro 1120
Утилита Apple Immersive Video 1120
Developer Information 1121
Camera Control REST API 1121
Livestream Control API 1123
Clips Control API 1127
Media Pool Control API 1128
Monitoring Control API 1131
Event Control API 1140
System Control API 1140
Transport Control API 1146

Timeline Control API 1150
Media Control API 1153
Slate Control API 1156
Preset Control API 1162
Audio Control API 1164
Lens Control API 1169
Video Control API 1173
Camera Control API 1181
Immersive Control API 1184
Color Correction Control API 1185
Notification websocket - 1.0.0 1190
messages 1190
Помощь 1214
Соблюдение нормативных требований 1215
Правила безопасности 1216
Гарантия 1217

Blackmagic URSA Cine Immersive

Поздравляем вас с приобретением или взятием напрокат камеры Blackmagic URSA Cine Immersive.

Эта модель предназначена для съемки стереоскопического VR-контента, который из-за охвата на 180 градусов обычно называют иммерсивным. В частности, камера позволяет создавать материал с эффектом погружения для гарнитуры Apple Vision Pro в наилучшем на сегодня качестве.

Это руководство содержит краткий обзор модели Blackmagic URSA Cine Immersive, а также описание ее базовых принципов работы и уникального формата Apple Immersive Video. Понимание особенностей такого материала позволит лучше освоить и саму камеру.

Ниже содержатся рекомендации технического и творческого характера, с помощью которых можно не только вести съемку, но и узнать о преимуществах иммерсивного формата.

Свойства иммерсивного видео

Иммерсивный контент представляет собой трехмерное пространственное видео. У него есть свои индивидуальные характеристики, хотя похожие форматы уже существуют. Для понимания отличий приведем краткое описание других технологий.

- **Традиционное 3D-кино** позволяет ощущать глубину в кадре благодаря съемке каждой сцены двумя камерами на дистанции, которая имитирует расстояние между глазами человека. Хотя стереоскопическое изображение создает пространственный эффект, зрительская перспектива контролируется операторским построением композиции.
- **Сферическое видео** основывается на принципах 3D-кино и панорамных камер, позволяя зрителю смотреть в любом направлении и ощущать глубину. Изображение с охватом на 360 градусов требует очень высокого разрешения, чтобы малые фрагменты изображения имели достаточную четкость. Поскольку человеческая физиология допускает только поворот головы от одного плеча до другого, для взгляда назад необходимо стоять или сидеть во вращающемся кресле.
- **Объемное видео** позволяет зрителю не только ощущать глубину и смотреть вокруг, но также двигаться и менять перспективу. Хотя его можно получить с помощью массива камер, большинство контента виртуальной или дополненной реальности создают с использованием визуализированного 3D-материала, а для съемки требуется безопасная среда.

Иммерсивное видео сочетает в себе преимущества традиционного кино и интерактивных 3D-форматов. По существу, это стереоскопический материал с охватом на 180 градусов, высокими кадровой частотой и разрешением.

Apple Immersive Video

Камера Blackmagic URSA Cine Immersive разработана специально для съемки в формате Apple Immersive Video. Используемый с гарнитурой Apple Vision Pro, он поддерживает высокое разрешение, стереоскопическое 3D-изображение и поле зрения 180 градусов.

Характеристики формата Apple Immersive Video

- Воспроизведение с частотой 90 к/с.
- Распространение в разрешении до 4320 x 4320 на канал.
- Уникальная система метаданных для переноса оригинального изображения камеры в среду с охватом 180 градусов.
- Формат Apple Spatial Audio (ASAF) для 3D-звука.

Формат Apple Immersive Video требует аппаратного обеспечения, которое позволяет воспроизводить стереоскопическое видео 180 градусов в высоком разрешении. На сегодня единственным таким устройством является гарнитура Apple Vision Pro.



Apple Vision Pro

Гарнитура имеет два дисплея Micro OLED 3660 x 3200 пикселей с шагом 7,5 микрон. Она содержит 13 камер отслеживания, сканер LiDAR, гиростабилизаторы и сенсоры трекинга для ориентации изображения в соответствии с движением головы и глаз.

Чип M2 с 8-ядерным центральным и 10-ядерным графическим процессорами обеспечивает воспроизведение с частотой до 100 к/с.

URSA Cine Immersive

Модель Blackmagic URSA Cine Immersive позволяет вести съемку в формате Apple Immersive Video для гарнитуры Apple Vision Pro.



Blackmagic URSA Cine Immersive

Камера использует две согласованные RGBW-матрицы по 58 Мпк. Они имеют почти квадратную форму, чтобы обеспечивать съемку с полем зрения 180 градусов по вертикали и горизонтали. Каждая из них оснащена широкоугольным объективом 210°, расстояние между которыми составляет 64 мм.

Встроенные ND-фильтры обеспечивают контроль света в диапазоне до 8 ступеней. Диафрагма объектива имеет фиксированное значение, составляющее примерно f4.5.

Камера ведет запись двух 8K-потоков с частотой 90 к/с. В результате скорость передачи данных существенно возрастает по сравнению со съемкой в 4K 24p.

Подготовка к работе

Чтобы начать работу с Blackmagic URSA Cine Immersive, достаточно подать питание на камеру и включить ее. Отформатированный накопитель Blackmagic Media Module 8TB установлен в специальный отсек и готов к записи.

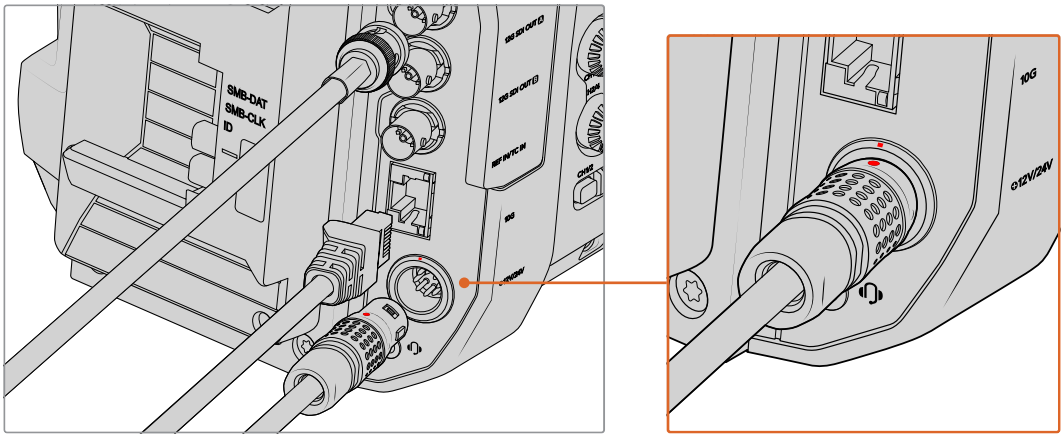
Подключение питания

Самый простой способ — использовать прилагаемый адаптер.

Подключение внешнего источника питания

1. Сначала снимите защитную резиновую крышку и уберите ее в надежное место. Подключите адаптер 24 В к сетевой розетке с помощью силового кабеля по стандарту IEC C13.
2. Соедините 8-контактный разъем адаптера с разъемом 24 В на тыльной стороне камеры.

При одновременном подключении внешнего источника питания и аккумуляторной батареи используется первый из них. Если его отсоединить, при наличии установленной и полностью заряженной батареи камера сразу перейдет на питание от нее.



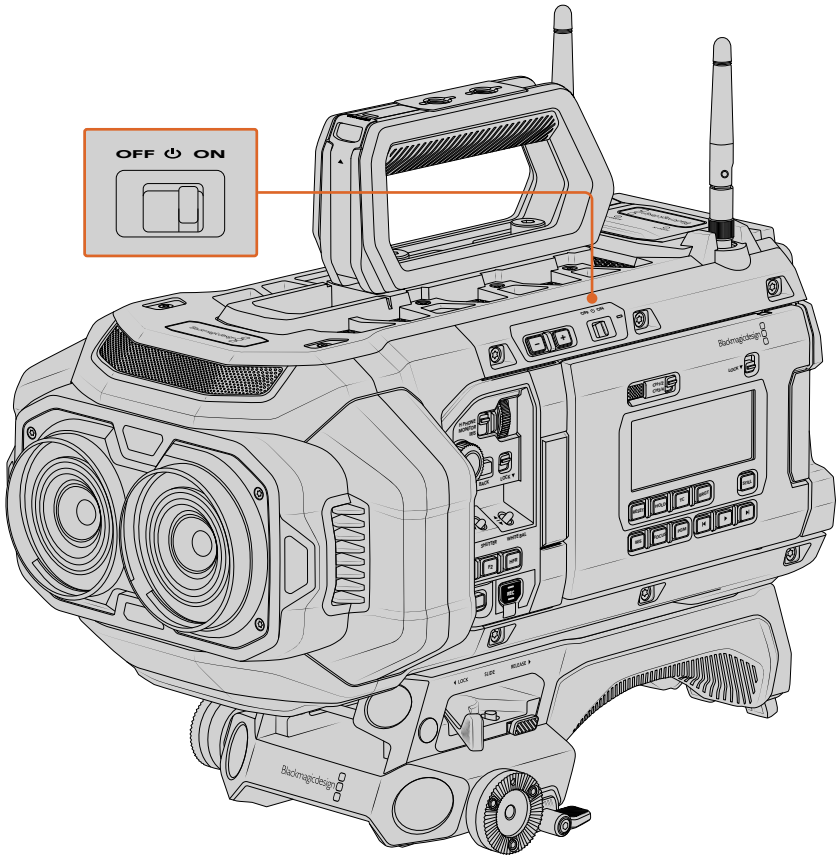
Чтобы подключить камеру к питанию, совместите красные точки и аккуратно вставьте кабель в разъем до конца. Для отключения потяните на себя металлическое основание кабеля и извлеките его из гнезда.

В комплект поставки включена площадка B-Mount, предназначенная для установки аккумуляторов 24 В соответствующей конструкции. Для работы с URSA Cine Immersive идеально подходит диапазон напряжения 24–34 В, так как при нем снижаются тепловые потери, что позволяет вести запись с высокой кадровой частотой и питать от камеры несколько аксессуаров.

Кроме того, доступны дополнительные площадки VLock и Gold Mount (24 В), а также Gold Mount Plus (26 В) под батареи сторонних производителей.

ПРИМЕЧАНИЕ. Когда питание камеры URSA Cine Immersive осуществляется от источников или батарей 12–18 В, подача энергии через 3- и 2-контактные разъемы (RS и LEMO соответственно) невозможна.

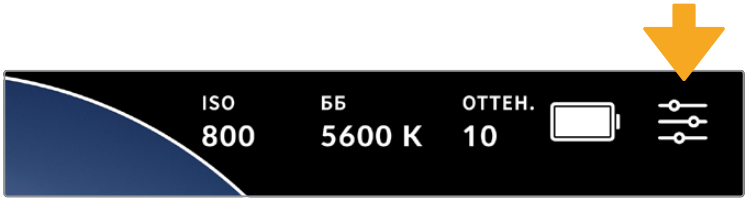
После установки батареи или подсоединения внешнего источника питания включите камеру URSA Cine Immersive. Установите расположенный сверху тумблер в положение ON (Вкл.).



Чтобы включить камеру, установите тумблер в положение ON (Вкл.)

Установка настроек записи

Этот раздел содержит информацию о настройках, которые являются уникальными или важными при съемке иммерсивного контента. Чтобы открыть меню, коснитесь соответствующего значка в правом верхнем углу откидного или дополнительного сенсорного дисплея.



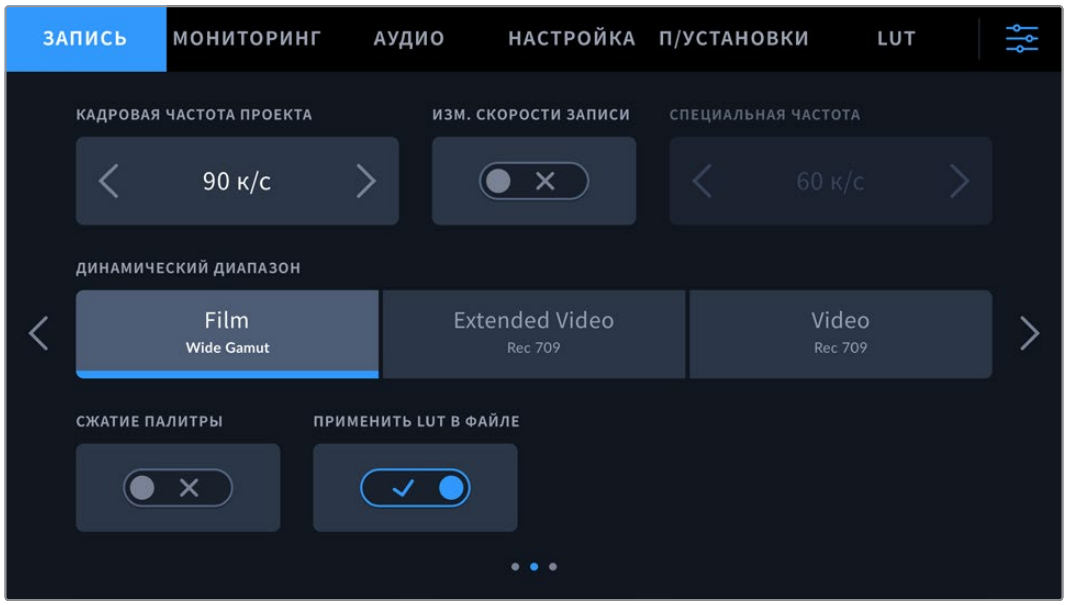
Вкладка «ЗАПИСЬ» (стр. 1)

Секция «КОДЕК И КАЧЕСТВО» на этой странице позволяет выбрать настройку Blackmagic RAW в зависимости от специфики проекта. Доступны четыре варианта постоянного битрейта и сохранение с постоянным качеством. При записи контента для Apple Vision Pro с частотой 90 к/с настоятельно рекомендуется использовать опцию 12:1 или Q3. Они обеспечивают оптимальный баланс между скоростью передачи данных с приемлемым размером файла и исключительно высокой четкостью изображения.



Вкладка «ЗАПИСЬ» (стр. 2)

Вторая страница вкладки «ЗАПИСЬ» содержит настройки кадровой частоты.



Для корректного отображения на гарнитуре Apple Vision Pro кадровая частота проекта должна составлять 90 к/с.

При съемке для других систем может быть достаточным более низкое значение (60 или 50 к/с). Традиционные для кино кадровые частоты, такие как 24 или 25 к/с, подойдут при создании фонового изображения под визуальные эффекты в фильмах.

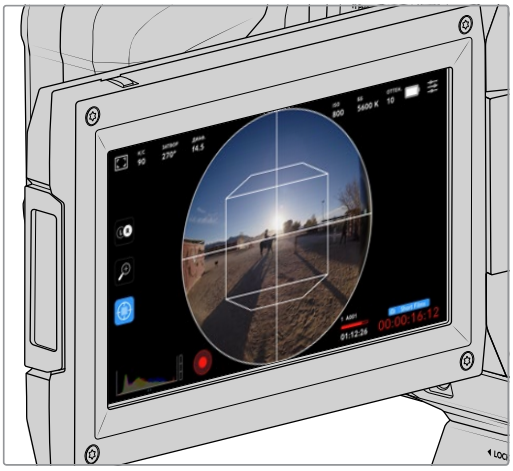
Подготовка к началу работы с URSA Cine Immersive закончена. Накопитель Media Module уже отформатирован и позволяет сразу приступать к записи.

Мониторинг изображения

Камера URSA Cine Immersive имеет несколько вариантов мониторинга изображения во время съемки.

Откидной сенсорный дисплей предназначен для использования оператором. Его можно повернуть назад, чтобы уменьшить физические размеры камеры с сохранением видимости, или вперед для мониторинга при съемке самого себя.

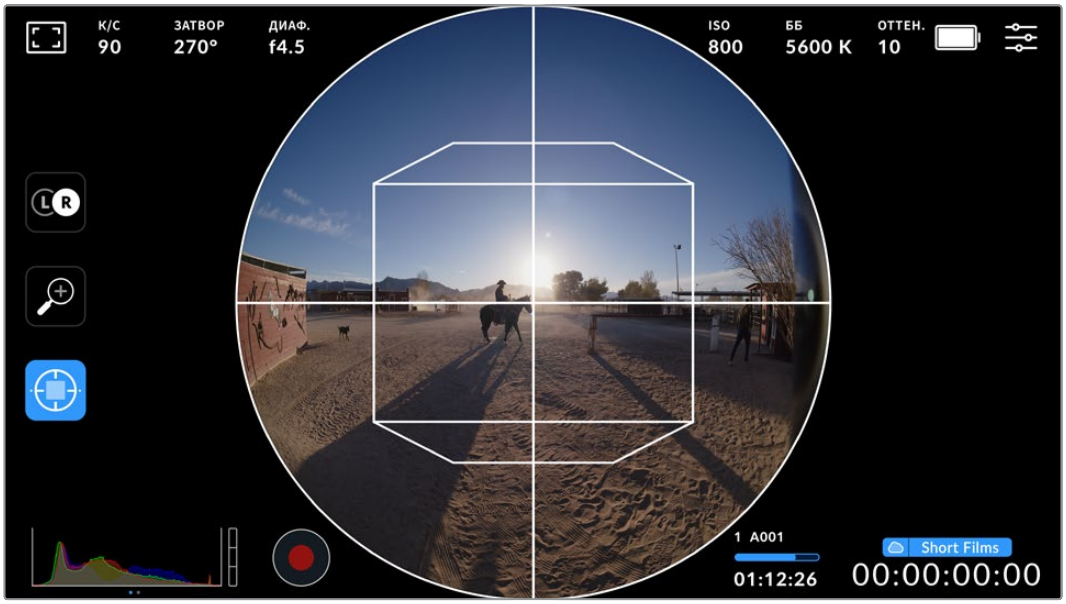
Вспомогательный сенсорный дисплей позволяет другому члену команды отслеживать изображение без помех для оператора. При его использовании следует соблюдать осторожность из-за исключительно широкого поля зрения камеры.





Опции мониторинга и вывод параметров

Единовременно камера показывает охватываемое объективом пространство для одного глаза. Откидной и вспомогательный сенсорные дисплеи скрыты для отображения обзора на 180 градусов. На SDI-выходы поступает полный материал, фиксируемый каждой матрицей (210 градусов).



Все выходы мониторинга отображают одинаковый канал. Переключение между правым и левым выполняют касанием кнопки L R на любом сенсорном дисплее.

Вывод дополнительных параметров на сенсорные дисплеи можно активировать соответствующей кнопкой. В этом случае будут отображаться:

- внешняя граница поля 180 градусов;
- перекрестие в центре кадра;
- внутренняя рамка, показывающая приблизительное поле зрения на Apple Vision Pro при взгляде прямо вперед.

Эти инструменты помогают с точным кадрированием и позиционированием при съемке иммерсивного контента.

Кадрирование первого плана

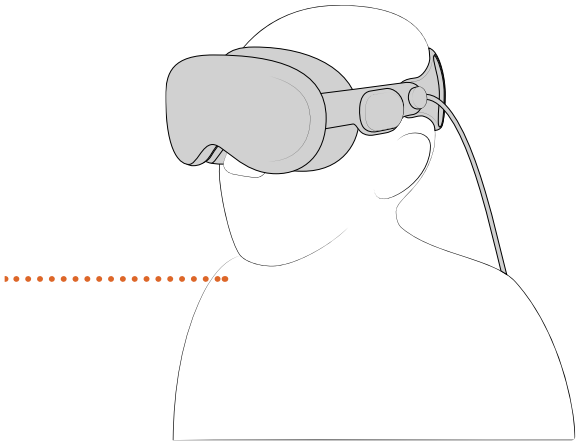
Иммерсивный контент и традиционное 2D-видео имеют много общего, но в то же время есть четкие различия.

Например, при длительном просмотре экрана мобильного телефона в автомобиле может возникать чувство тошноты. Оно появляется из-за того, что мозг человека получает противоречивые сигналы. Органы слуха говорят о том, что он движется, тогда как глаза свидетельствуют о стационарном положении.

Иммерсивное видео может создавать схожий эффект. При панорамировании камеры зрительно возникает чувство движения, однако органы слуха и положение шеи говорят о стационарном состоянии. По этой причине перед работой над динамичными сценами лучше овладеть приемами съемки статичных планов.

Высота камеры

В традиционном 2D-кино высокое или низкое размещение камеры может показывать баланс силы в сцене. Если при съемке иммерсивного контента ее расположить на уровне пола, у зрителя возникнет чувство, что он закопан в землю по шею.



Воспринимаемая высота может оказать влияние на вызываемые при просмотре ощущения. При размещении камеры внизу у зрителя возникает чувство, не имеющее общего с реальностью.



При съемке иммерсивного контента низкие ракурсы могут вызывать у зрителя неприятные ощущения

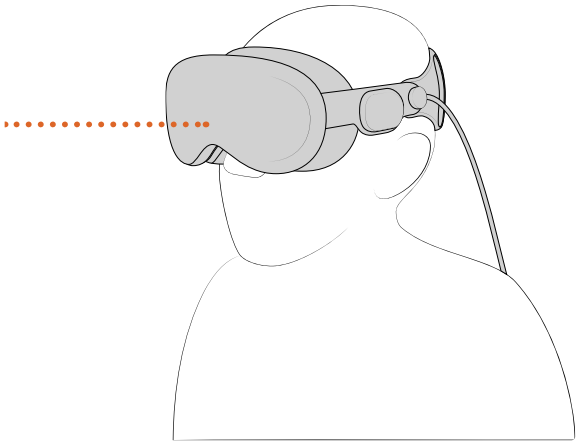
Чтобы создать ощущение присутствия, чаще всего лучше выбирать такие высоту и положение камеры, при которых зрителю кажется, будто он стоит или сидит. Вместе с тем по-прежнему остается пространство для творческих экспериментов. Если в сцене есть люди, повышение ракурса дает им чувство превосходства, позволяя смотреть на других сверху. В противном случае, когда ракурс ниже, возникает обратный эффект.

Иногда комфортный для просмотра материал можно получить и при более широкой амплитуде высоты камеры. Съемку с воздуха, сверху, из нижнего положения и многие другие приемы успешно применяют при создании иммерсивного контента. В сочетании с хорошим монтажом, движением и контекстом будет достигнут надлежащий эффект.

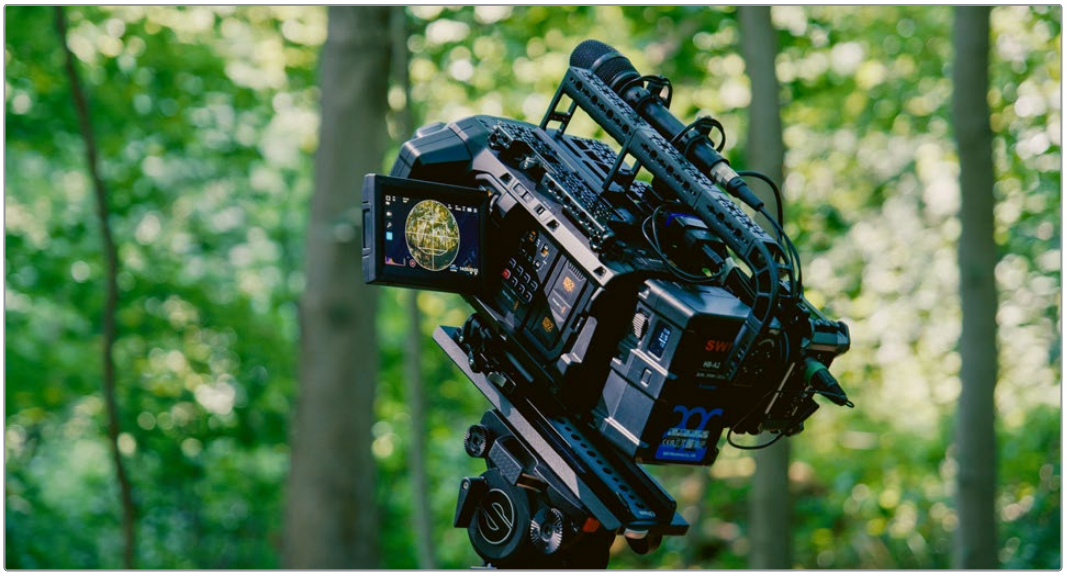
Наклон камеры вверх или вниз

При съемке традиционного 2D-видео и высоком или низком размещении камеры оператор часто наклоняет ее для сохранения композиции. В случае с иммерсивным контентом необходимо учитывать эффект таких действий.

При направлении камеры вниз зрителю придется поднять голову вверх, чтобы видеть горизонт. Это вряд ли вызовет неприятные ощущения вроде тошноты, но нарушит чувство реальности, поскольку горизонт всегда расположен прямо по линии взгляда.



В наиболее выигрышных иммерсивных планах угол горизонта составляет 0 градусов



При наклоне камеры URSA Cine Immersive у зрителя возникнет несоответствие из-за разных положений головы, при которых они видят горизонт сейчас и привыкли видеть во всех остальных случаях

По этой причине большинство иммерсивных планов снимают при наклоне 0 градусов или очень близко к нему. Так как камера фиксирует поле зрения, которое составляет более 180° по горизонтали и вертикали, зритель по-прежнему может переводить взгляд вверх и вниз. Именно он, а не оператор, решает, куда смотреть.

Чтобы побудить зрителя перевести взгляд на другие участки изображения, можно использовать такие приемы, как освещение, апертурное кадрирование и подводящие линии.

Позиционирование

Обычно иммерсивный контент смотрят сидя. Так как он не предполагает дополненной или виртуальной реальности, не нужно стоять и двигаться. Находясь в сидячем положении, можно провести эксперимент и повернуть голову из стороны в сторону, а затем вверх и вниз. Насколько далеко удастся отвести взгляд без смещения тела? Поле зрения иммерсивного материала превышает комфортный диапазон поворота у большинства людей. В этом заключается его преимущество, потому что при отклонении влево или вправо данный формат все равно обеспечивает периферическое восприятие. Проведенный эксперимент позволяет понять основной принцип построения композиции.

Типичный диапазон движения головы

Поворот: 120–160 градусов
Сгибание и вытягивание: 110–130 градусов

Когда поворот головы сопровождается движением глаз, можно посмотреть назад через плечо, однако для большинства людей долго соблюдать такое положение будет некомфортно.

Чтобы облегчить построение композиции, активируйте вывод параметров камеры. Когда действие сосредоточено в границах внутренних меток, показывающих типичное поле зрения, контент будет более удобен для восприятия.

Также необходимо учитывать движение объектов в рамках кадра. Следование за человеком, который перемещается из крайнего левого в крайнее правое положение, или переход между планами с искомым предметом в противоположных концах потребует сильного поворота головы у зрителя. Тщательное построение композиции при создании художественного образа позволит создать комфортный для просмотра материал.

Свобода зрителя

При съемке традиционного 2D-видео управлять вниманием зрителя можно с помощью кадра, выбора оптики и наведения фокуса. В случае с иммерсивным контентом эта задача усложняется из-за поля зрения 180 градусов, которое затрудняет исключение тех или иных деталей. По этой причине у зрителя появляется гораздо больше свободы в изучении изображения, что делает такой формат очень эффектным. Для управления взглядом используют тщательно выверенные направленный звук, освещение, блокировку и композицию.

Движение камеры

Камеру для съемки иммерсивного контента не обязательно держать в статичном положении, однако ее движение должно быть осторожным. Хорошей аналогией будет пример с грузовым автомобилем, масса и инерция которого предотвращают внезапное изменение скорости или направления. Разгон и торможение осуществляются предсказуемым и контролируемым образом. После освоения статичных планов и перехода к динамичным сценам учет перечисленных ниже факторов позволит качественно показать движение в кадре.

- **Продолжительность.** Зрителю требуется время, чтобы адаптироваться к движению камеры. Резкие скачки способны вызвать дезориентацию.
- **Интенсивность.** Движения низкой интенсивности проще интегрировать в иммерсивный контент. Высокая интенсивность не воспрещается, однако для комфорта зрителю потребуются адаптация до и после.
- **Визуальные якоря.** Ими являются статичные объекты сцены. План без визуальных якорей может дезориентировать. Элементы переднего плана (например, салон автомобиля) или такие удаленные объекты, как горизонт, обеспечивают укоренение динамичной сцены.

Скорость передачи данных

В таблице ниже указана скорость передачи для иммерсивного контента 8K x 7K при 90 к/с и 24 к/с.

Максимальная скорость передачи данных при 90 к/с в кодеке Blackmagic RAW с постоянным битрейтом

8K x 7K иммерсивный контент			
Кодек	**Разрешение**	**Частота кадров**	**Скорость**
Blackmagic RAW 5:1	8160 x 7200 стерео	90	3203 МБ/с
Blackmagic RAW 8:1	8160 x 7200 стерео	90	2002 МБ/с
Blackmagic RAW 12:1	8160 x 7200 стерео	90	1335 МБ/с
Blackmagic RAW 18:1	8160 x 7200 стерео	90	889 МБ/с

Максимальная скорость передачи данных при 24 к/с в кодеке Blackmagic RAW с постоянным битрейтом

8K x 7K иммерсивный контент			
Кодек	**Разрешение**	**Частота кадров**	**Скорость**
Blackmagic RAW 5:1	8160 x 7200 стерео	24	854 МБ/с
Blackmagic RAW 8:1	8160 x 7200 стерео	24	534 МБ/с
Blackmagic RAW 12:1	8160 x 7200 стерео	24	356 МБ/с
Blackmagic RAW 18:1	8160 x 7200 стерео	24	237 МБ/с

Звук и тайм-код

В иммерсивном контенте аудиодорожка играет очень важную роль. По этой причине камеру URSA Cine Immersive необходимо использовать с соответствующим оснащением записи звука.

Формат Apple Spatial Audio (ASAF)

Формат Apple Spatial Audio (ASAF) обеспечивает 3D-звук при просмотре иммерсивного видео на гарнитуре Apple. Он является частью более широкой платформы Apple Positional Audio Framework и позволяет хранить три типа записи.

- **На основе каналов.** В файл ASAF можно интегрировать традиционный объемный звук. Такой вариант хорошо подходит для материала с фиксированным положением источника голоса и музыкой.
- **На основе объектов.** Звук рассматривается как совокупность индивидуальных объектов, метаданные которых описывают их положение в 3D-пространстве вне зависимости от отдельных каналов. Обычно это монофоническая запись. Трехмерные координаты можно анимировать, чтобы следить за движущимися в сцене объектами. Такой вариант хорошо подходит для диегетического звука, чьи источники видны на экране.
- **Объемный звук.** Формат объемного аудио, когда с помощью сферической гармоники получают эффект амбиофонии. Он хорошо подходит для создания окружающего звука.

Ниже содержится обзор оборудования, необходимого для получения оптимального файла ASAF.

Объемный звук

Микрофон для объемного звука содержит четыре и более капсул в строго заданной геометрической форме. Каждая из них записывает один и тот же аудиопоток с разными начальным временем и интенсивностью в зависимости от направления, из которого он поступает. Сохранение выполняется на дискретных каналах для последующей обработки на компьютере. Путем сравнения различий устанавливаются траектория и природа записанного звука.



Микрофон для амбиофонии первого порядка

Микрофоны для записи объемного звука

Микрофоны для создания амбиофонии классифицируют по количеству имеющихся капсул: чем их больше, тем точнее можно воспроизвести пространственную детализацию.

Первый порядок:	четыре канала
Второй порядок:	девять каналов
Третий порядок:	16 каналов
Четвертый порядок:	25 и более каналов

Независимо от расположения микрофона (на камере или сбоку), он должен совпадать с линией съемки. Кроме того, из-за возможности использования в разных конфигурациях (осевое, боковое, верхнее или нижнее направление) эту информацию необходимо довести до сведения звукорежиссера. Ее передают посредством метаданных на устройстве записи или с помощью сопроводительных комментариев.

Объемную запись можно использовать как основу для окружающего звука. На нее легко добавить другие каналы и эффекты различных объектов. При правильной ориентации микрофона относительно камеры она также служит отправной точкой для звукорежиссера по мере создания аудиодорожки.

Петличные и миниатюрные микрофоны

При съемке на камеру с полем 180 градусов невозможно использовать микрофонную удочку, потому что она попадет в объектив. По этой причине для записи диегетического звука и диалогов потребуются скрытые и петличные микрофоны.

Возможные варианты

- Подключение источников звука напрямую к двум XLR-входам камеры URSA Cine Immersive с помощью беспроводных передатчиков сигнала.
- Подключение к внешнему звуковому рекордеру. Он может представлять собой большое многоканальное или миниатюрное портативное устройство.

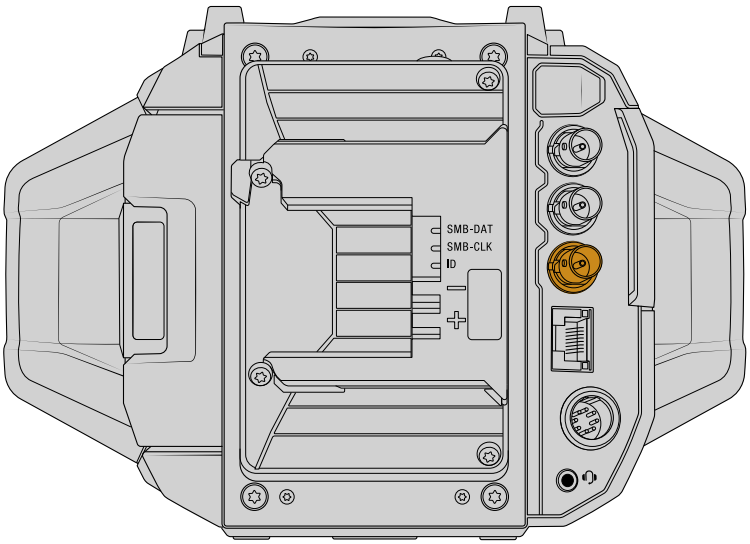
Записанные звуковые дорожки можно синхронизировать и микшировать на основе каналов или объектов.

Ручные микрофоны

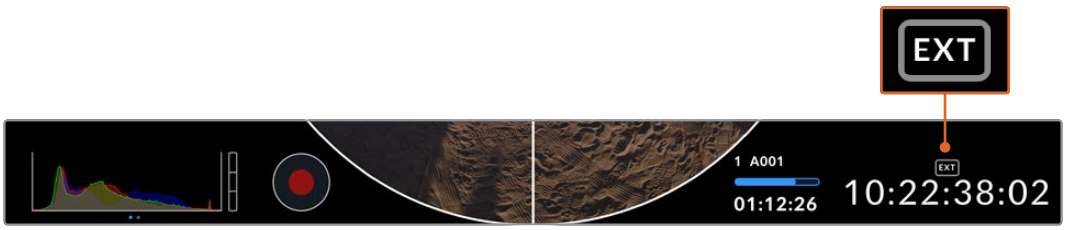
Запись дополнительных звуковых эффектов на локации после остановки камеры не только экономит время при постобработке, но и обеспечивает создание более аутентичной фонограммы. Использование ручных гиперкардиоидных или схожих микрофонов дает весомые преимущества, позволяя располагать их максимально близко при повторной съемке сцен. Записанные звуковые дорожки можно включить в конечный микс на основе каналов или объектов.

Подключение тайм-кода

Чтобы получать тайм-код извне, соедините его генератор с соответствующим входом на тыльной стороне камеры URSA Cine Immersive посредством BNC-кабеля. При съемке двухпоточного видео это позволяет синхронизировать звук и изображение на этапе постобработки.



Если кадровая частота сигнала из источника внешнего тайм-кода совпадает с заданными настройками, данный тайм-код будет выбран автоматически. В этом случае на ЖК-дисплеях модели URSA Cine Immersive рядом со счетчиком тайм-кода появляется обозначение «ВНЕШ.». Если отсоединить кабель, обозначение изменится на «ВНУТ.», что указывает на использование внутреннего тайм-кода камеры.



При подключении к генератору тайм-кода отображается соответствующее обозначение

ПРИМЕЧАНИЕ. Тайм-коды по стандартам SMPTE предусмотрены для типичных кадровых частот (23,976; 24; 30 к/с и т. д.), а для 90 к/с общепринятого формата нет. По этой причине при проектной кадровой частоте 90 к/с камера URSA Cine Immersive использует тайм-код для 30 к/с. На SDI-выходах, как и на дополнительном оснащении и внешних звуковых рекордерах, тоже применяется данная настройка.

Съемка

Эта глава позволяет углубленно изучить функционал камеры Blackmagic URSA Cine Immersive после освоения базовых принципов работы с ней и понимания преимуществ создаваемого с ее помощью контента.

Требования к экспозиции

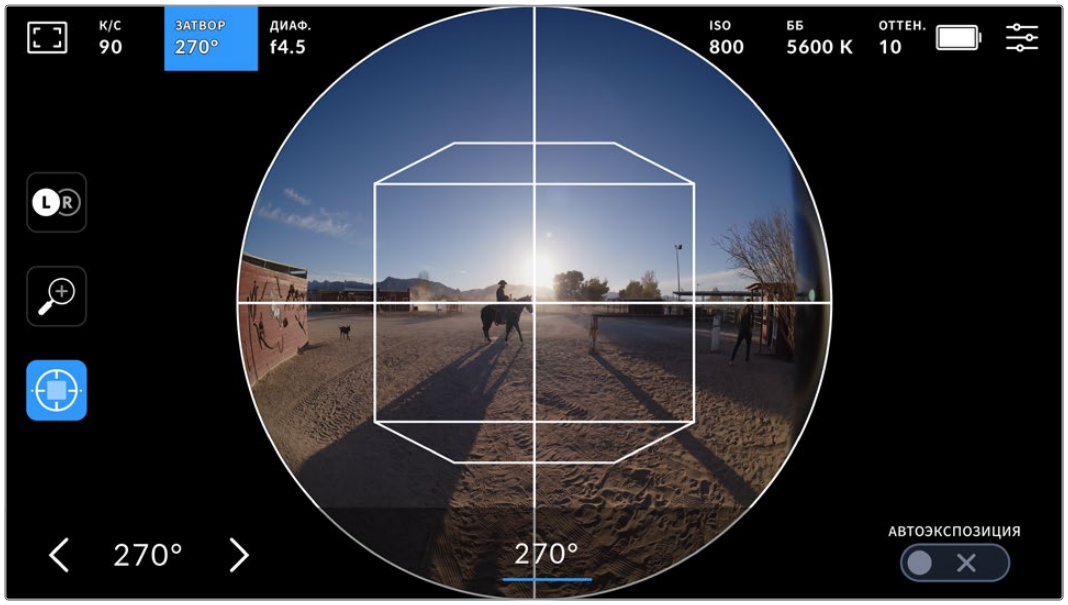
Формат Apple Immersive Video требует более высокой экспозиции по сравнению с изображением 24 к/с, в связи с чем необходимо изменить подход к съемке. Если настройка затвора составляет 180 градусов, то при записи с частотой 90 к/с для сопоставимого результата нужно почти в четыре раза больше света. Кроме того, кадры объективов с фиксированной диафрагмой приблизительно f4.5 до трех ступеней темнее, чем у оптики с очень высокой светосилой. Как следствие, для достижения достаточной экспозиции нужно до семи дополнительных ступеней. Однако если при 90 к/с использовать широкий угол затвора между 270° и 360°, это поможет компенсировать утрату света из-за повышения частоты и вместе с тем получить хороший визуальный ряд.

Выдержка затвора

Индикатор «ЗАТВОР» показывает угол раскрытия или выдержку затвора. В меню «НАСТРОЙКА» для данного параметра предусмотрено два способа расчета и отображения.

Порядок изменения выдержки затора

1. Коснитесь индикатора «ЗАТВОР» на верхней панели инструментов.
2. Используя левую или правую стрелку, выберите одно из доступных значений. Для установки индивидуальной настройки затвора дважды коснитесь текущей величины в нижнем левом углу экрана и на появившейся клавиатуре введите необходимое число с точностью до двух десятичных знаков в диапазоне от 11,2 до 360 градусов.



Индивидуальную кадровую частоту можно задать с помощью экранной клавиатуры

Выдержка затора оказывает не такое большое влияние на визуальный ряд иммерсивного контента с высокой кадровой частотой. Хотя при 24 или 30 к/с многие по-прежнему используют значение 180 градусов, при съемке в данном формате эту настройку обычно используют для помощи с экспозицией.

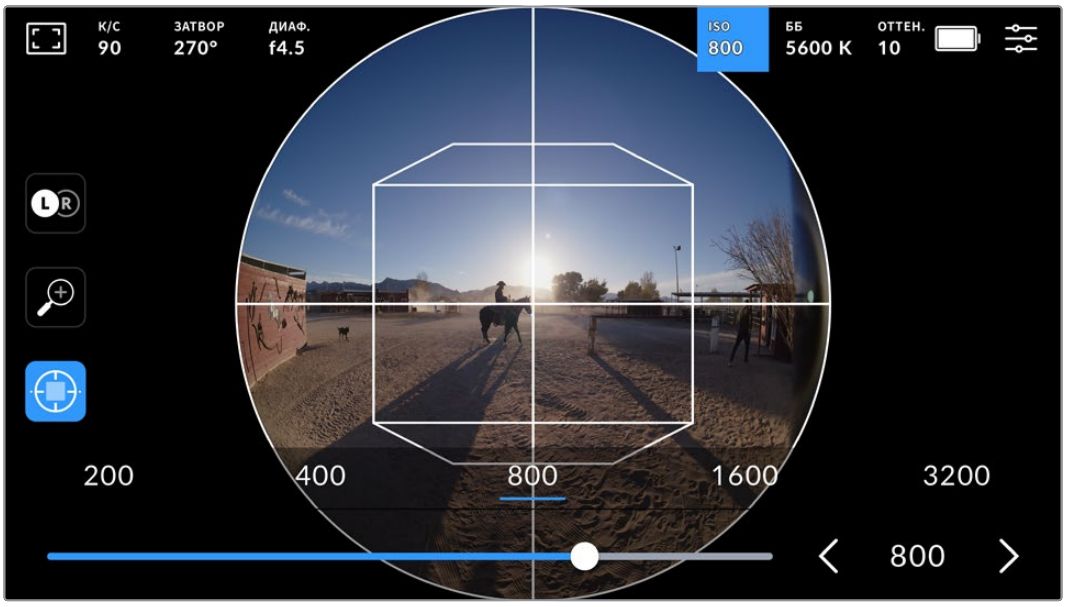
В условиях слабого освещения угол затвора можно довести до 270°, а в исключительных обстоятельствах — до 360°.

При ярком освещении допускается уменьшение значения затвора до 11,2 градусов. Тем не менее в таких условиях его чаще оставляют на уровне предпочтительной настройки, а для контроля экспозиции используют ND-фильтры.

Следует отметить, что при съемке в формате Apple Immersive Video с частотой 90 к/с могут возникать проблемы из-за мерцающего освещения. В сетях 60 Гц рекомендуется использовать настройку 270°, а при 50 Гц — 324°.

ISO и динамический диапазон

Индикатор ISO показывает текущую настройку световой чувствительности камеры. Коснитесь индикатора, чтобы изменить ее в соответствии с условиями съемки. На URSA Cine Immersive диапазон ISO составляет от 200 до 3200, оптимальное значение — 800.



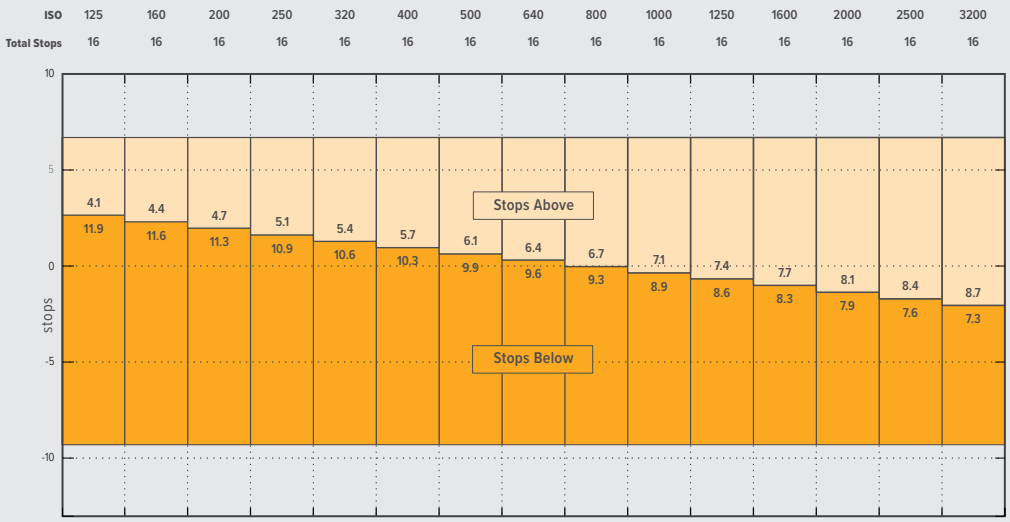
Световую чувствительность можно задать с помощью индикатора ISO

Чтобы выбрать целое число из доступных предустановок, коснитесь соответствующего значения ISO. Слайдер под ними позволяет изменить ISO с шагом 1/3 ступени.

В зависимости от конкретных условий съемки можно выбрать более высокое или низкое значение ISO. Например, для слабого освещения подойдет настройка ISO 1600, хотя при ее использовании вероятно появление заметного электронного шума. При ярком свете для передачи насыщенных цветов лучше использовать ISO 200.

Благодаря широкому динамическому диапазону съемочная техника Blackmagic Design обеспечивает исключительную гибкость на этапе создания изображения и его грейдинга в системе DaVinci Resolve. В таблице ниже показаны соответствующие доступные значения на камере URSA Cine Immersive.

Динамический диапазон камеры URSA Cine Immersive



Светофильтры ND

Самым распространенным способом контроля экспозиции является использование ND-фильтров камеры. Они позволяют уменьшить количество попадающего на матрицу света в соответствии с заданным шагом экспозиции.



Для работы с ND-фильтрами камеры URSA Cine Immersive используют кнопки с левой стороны корпуса над откидным сенсорным дисплеем. Доступные настройки: без фильтра, две, четыре, шесть и восемь ступеней.

Индикатор в левом верхнем углу сенсорного дисплея показывает текущую настройку ND-фильтра. При использовании фильтра он отображается постоянно, а в противном случае гаснет через четыре секунды.



ПРИМЕЧАНИЕ. Для удобства обозначения, используемые при работе с ND-фильтрами, можно изменить. Меню настройки позволяет выбрать число, ступень или дробь.

Изменение скорости записи

Кадровая частота проекта, заданная в меню настройки камеры URSA Cine Immersive, определяет формат записи материала. Обычно выбирают значение, которое соответствует скорости воспроизведения и звукового сопровождения в приложении постобработки.



Чтобы изменить настройку кадровой частоты на URSA Cine Immersive, коснитесь значка К/С. Камера имеет девять опций для съемки проектов: 23,98; 24; 25; 29,97; 30; 50; 59,94; 60 и 90 к/с. При создании иммерсивного контента кадровая частота проекта должна составлять 90 к/с.

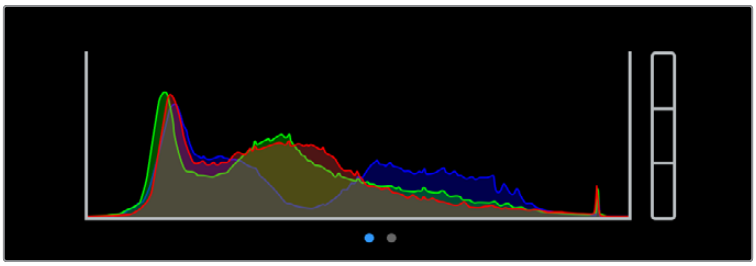
Кадровая частота матрицы определяет скорость съемки изображения. По умолчанию на URSA Cine Immersive используются одинаковые значения кадровой частоты матрицы и проекта для воспроизведения в обычном режиме. При необходимости можно установить независимую кадровую частоту матрицы. Для этого коснитесь значка «СПЕЦ. ЧАСТОТА» в правом нижнем углу меню «К/С».

При активации специальной кадровой частоты матрицы ее настройку можно задать вручную. Максимальное значение составляет 90 к/с и соответствует кадровой частоте проекта при создании контента для Apple Vision Pro. Если кадровая частота проекта меньше (например, 30 или 60 к/с), а съемка идет со скоростью 90 к/с, будет получен материал с эффектом замедленного движения.

При изменении скорости записи индикатор в левом верхнем углу экрана показывает кадровую частоту матрицы, а после нее — кадровую частоту проекта.

Гистограмма

В левом нижнем углу сенсорного дисплея на камере URSA Cine Immersive выводится гистограмма, которая показывает распределение яркости в видео. Левая сторона соответствует темным тонам, середина — средним, а правая — светлым. Высота графика обозначает, какая часть изображения имеет такой уровень яркости.



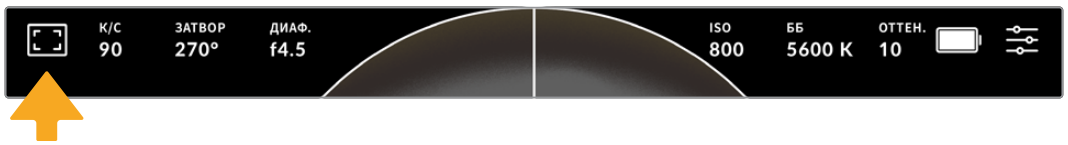
Гистограмма дает наглядное представление о распределении тонов в клипе. Если ее основная часть находится слева, имеет место недостаточная экспозиция, а если справа — избыточная. Интерпретировать визуальную индикацию следует в контексте самого изображения, но обычно при должной экспозиции график сбалансирован и располагается в середине. При увеличении экспозиции гистограмма смещается вправо, при уменьшении — влево.

Индикатор справа от гистограммы показывает утрату детализации в отдельных цветовых каналах. Обычно ее следует избегать, однако в отдельных случаях она допускается, если это помогает сохранить средние тона и тени изображения.

Метод экспонирования вправо заключается в намеренном использовании избыточной экспозиции с предельно возможным сдвигом гистограммы в соответствующем направлении без утраты детализации в цветовых каналах. Такой прием увеличивает до максимума сигнал и сводит к минимуму шум. При условии уменьшения экспозиции во время постобработки это позволяет получить четкое изображение.

Опции мониторинга на ЖК-дисплее

Чтобы получить доступ к настройкам мониторинга, коснитесь значка экрана в левом верхнем углу сенсорного ЖК-дисплея на URSA Cine Immersive. Это меню позволяет задать яркость и содержит мощные инструменты установки экспозиции.



Зебра

Данной настройкой выбирают способ вывода зебры на ЖК-дисплей, а также задают ее уровень. Она показывает диагональные линии в тех областях, где экспозиция превышает установленный предел. Величина в процентах используется для пометки участков с избыточной экспозицией. Обычным является значение 95%, а не 100%, потому что с его помощью идентифицируют фрагменты изображения, поддающиеся коррекции. Второе значение позволяет активировать зоны среднего серого или среднего серого плюс один шаг.

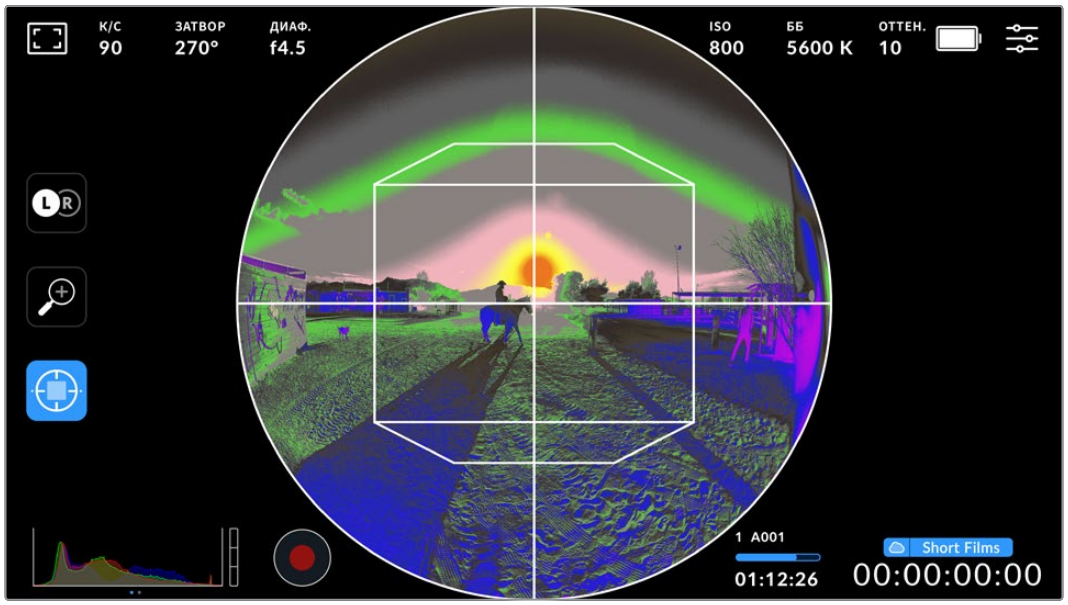
Сетка

Эта настройка позволяет использовать один из видов сетки («Трети», «Горизонт», «Крестик» и «Точка») на ЖК-дисплее, а также выбирать ее для вывода на всех выходах модели URSA Cine Immersive. При съемке иммерсивного контента индикатор горизонта бывает очень полезен. Он показывает двойное перекрестие в центре изображения, которое выровнено при точно горизонтальном положении камеры и расходится при смещении.

Условный цвет

Эта функция добавляет к изображению различные цвета, которые показывают уровень экспозиции для его отдельных элементов. Например, розовый является показателем оптимальной экспозиции для достоверной передачи тонов светлой кожи, зеленый — более темной. Во время съемки людей подобная функция помогает обеспечить однородность визуального ряда.

Если его элементы меняют цвет с желтого на красный, это означает избыточную экспозицию. Как и гистограмма, данный инструмент позволяет очень быстро и наглядно оценить распределение тонов в изображении.



Запись

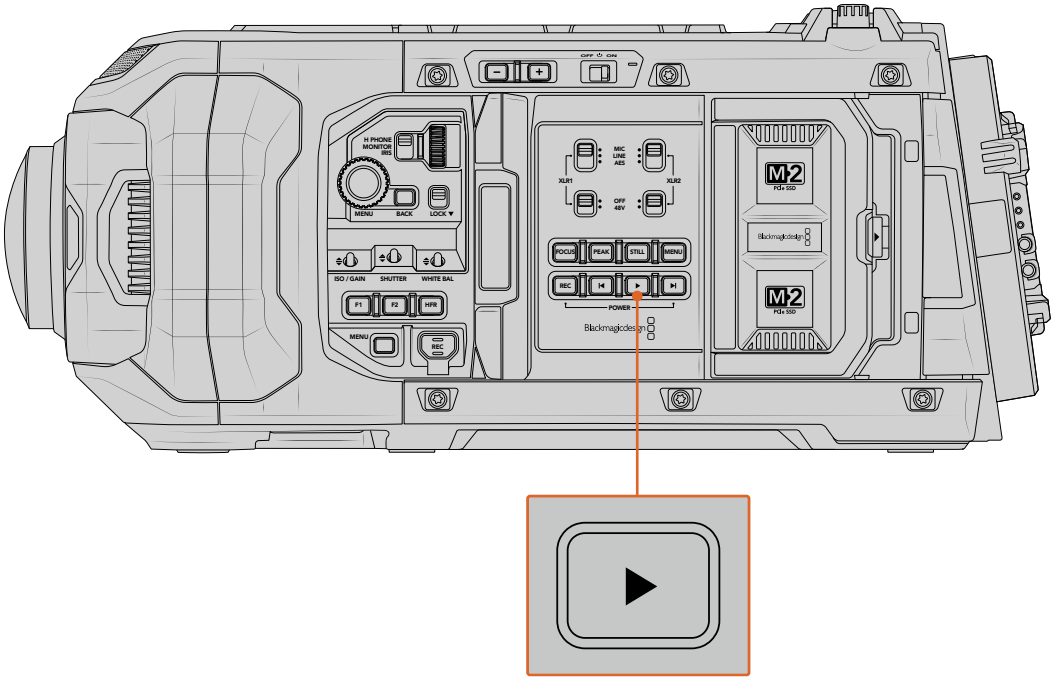
Для начала записи нажмите любую из красных кнопок на боковой панели управления, внутренней панели управления за откидным сенсорным экраном, панели управления вспомогательной станции и двух ЖК-дисплеях. Для остановки записи нажмите кнопку еще раз.

Индикаторы накопителей показывают оставшееся время записи. Оно отображается как часы:минуты:секунды с учетом выбранной кадровой частоты. В зависимости от состояния накопителя полоса, расположенная над значением времени, может иметь синий, белый или красный цвет, а степень заливки визуально показывает уровень заполнения.



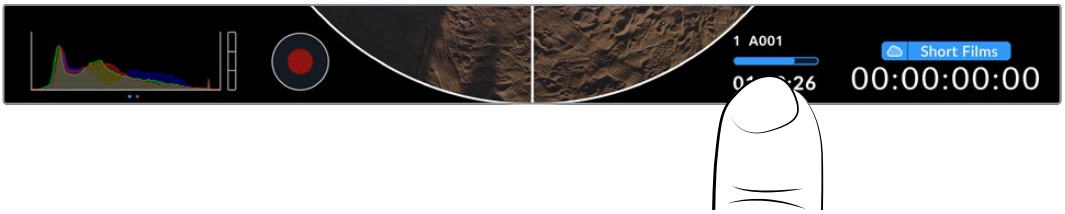
Воспроизведение

После того как клипы записаны, их можно просмотреть с помощью органов управления воспроизведением. Для вывода последнего из них нажмите соответствующую аппаратную кнопку с левой стороны камеры за откидным сенсорным дисплеем. Клавиши предыдущего и следующего клипов позволяют выполнить переход назад или вперед. Чтобы выйти из режима воспроизведения и продолжить съемку, нажмите кнопку записи один раз.

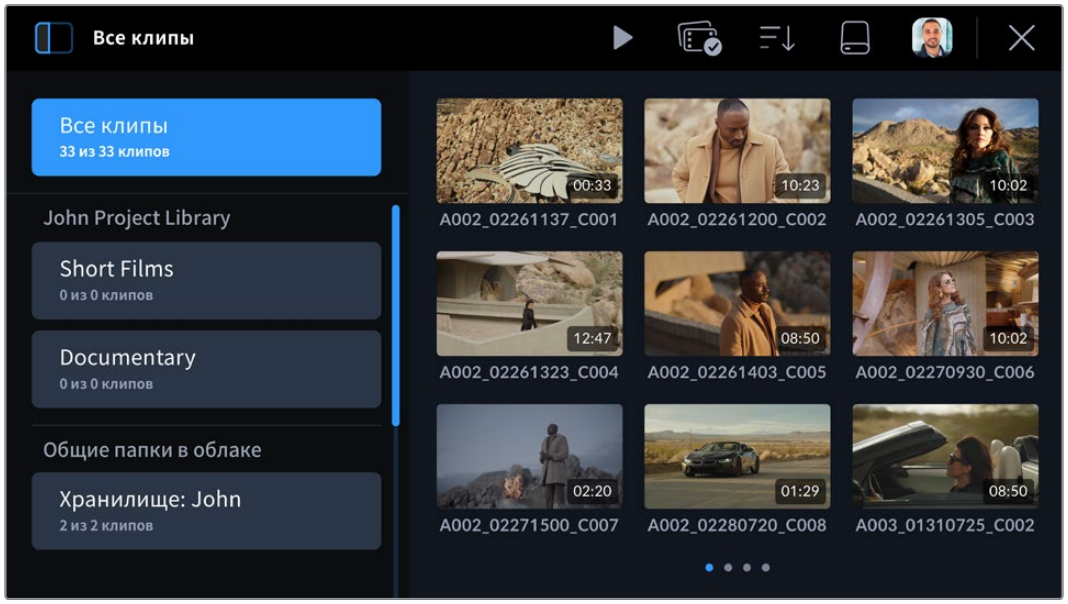


Для перехода к началу или окончанию клипа нажмите кнопку перемотки вперед либо назад. Для перехода к началу текущего клипа нажмите кнопку обратной перемотки один раз, к началу предыдущего клипа — два раза. Чтобы выполнить обычное или обратное воспроизведение с удвоенной скоростью, нажмите и удерживайте кнопку перемотки вперед либо назад. Если в режиме перемотки нажать одну из этих кнопок два, три или четыре раза, скорость воспроизведения увеличится в 4, 8 и 16 раз соответственно.

Когда требуется более детальное изучение отснятого материала, можно использовать медиатеку. Чтобы открыть ее, коснитесь одного из индикаторов накопителей в нижней части сенсорного дисплея.



На главной странице браузера отображаются эскизы всех клипов, сохраненных на подключенных к камере накопителях. Они расположены в три ряда, и по мере добавления нового материала к следующей группе видео легко перейти с помощью органов управления сенсорного дисплея. Индикаторы в нижней части экрана показывают точное число страниц с контентом.



Выполните вход с камеры URSA Cine Immersive в учетную запись на Blackmagic Cloud и направьте материал непосредственно в проект DaVinci Resolve.

Если нужно выйти из медиатеки, коснитесь крестика в правом верхнем углу или нажмите любую из кнопок записи на камере один раз.

Передача клипов

Предусмотрено несколько способов передачи записанных файлов с URSA Cine Immersive. Чтобы скопировать материал, достаточно подключить камеру к компьютеру по сети 10G Ethernet с помощью соответствующего кабеля. Кроме того, можно использовать устройство Blackmagic Media Dock, подсоединенное к сети или ПК и позволяющее одновременно использовать до трех накопителей.

При подключении к компьютеру по Ethernet блок Blackmagic Media Module распознается по аналогии с решением Blackmagic Cloud Store.

По умолчанию используется протокол DHCP с автоматическим присвоением камере IP-адреса.

Общий доступ к файлам

При первой настройке камеры и подключении к компьютеру или сети по Ethernet использование протокола SMB по умолчанию отключено. В отличие от работы с решением Blackmagic Cloud Store, когда требуется быстрый обмен файлами с другими членами команды, подобный уровень доступа к оригиналам после важной съемки может быть неуместен. Таким образом, активировать данный функционал необходимо самостоятельно.

1. Для этого необходимо загрузить и установить на компьютере утилиту Blackmagic Camera Setup. Она предназначена для изменения настроек администрирования и обновления внутреннего ПО съемочной техники. Загрузить утилиту Blackmagic Camera Setup можно по адресу https://www.blackmagicdesign.com/ru/support/family/professional-cameras.
2. После установки утилиты подключите камеру к компьютеру с помощью кабеля USB, а затем откройте ее. Используйте порт USB, расположенный на верхней панели корпуса ближе к его тыльной стороне.

3 Щелкните кнопкой мыши по изображению URSA Cine Immersive или значку настроек, чтобы получить доступ к меню.

4 Перейдите к настройкам сетевого доступа и активируйте опцию обмена файлами по протоколу SMB. Чтобы изменения вступили в силу, нажмите Save.

5 Затем нажмите на значок URL для автоматического заполнения адреса. Имя камеры появится в качестве адреса локальной сети.

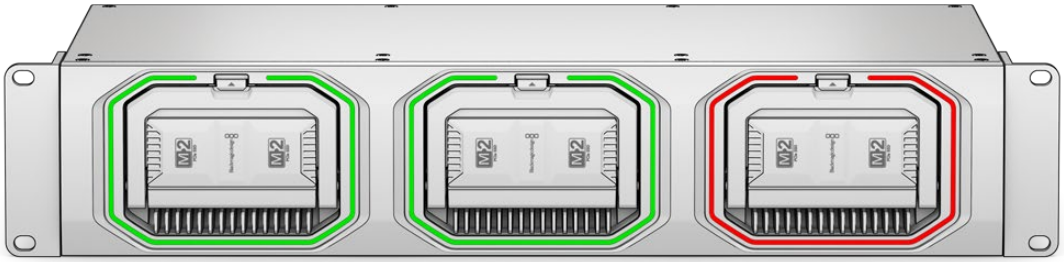
6 Для сохранения нажмите Save.

Для доступа к клипам нажмите на URL-адрес камеры. Откроется диалоговое окно для ввода пароля или гостевого подключения. Выберите вторую опцию и нажмите Connect.

7 Затем в приложении Finder или «Проводник» для компьютеров Mac и Windows соответственно автоматически появится накопитель. В утилите сохранятся выбранные настройки Ethernet, поэтому в следующий раз при подключении камеры по данному протоколу можно сразу перейти к накопителю без необходимости снова вводить параметры.

Работа с Blackmagic Media Dock

Накопитель Blackmagic Media Module легко установить и извлечь, когда необходимо выгрузить файлы на внешний носитель, а также при заполнении памяти для продолжения работы. Для доступа к отснятому материалу без камеры можно использовать решение Blackmagic Media Dock.



Заключение

Формат Apple Immersive Video предлагает новый подход к созданию визуального контента, который является альтернативой традиционному 2D-кино. Он призван не заменить, а дополнить доступные варианты съемки, выбираемые в зависимости от специфики проекта. Как и в других случаях, творческие решения будут варьироваться исходя из характера истории.

Хотя новые форматы часто подразумевают наличие комплексного рабочего процесса, камера URSA Cine Immersive облегчает создание иммерсивного контента. Ее также можно использовать при работе над визуальными эффектами и в других областях.

Это руководство содержит обзор базового функционала камеры URSA Cine Immersive и служит отправной точкой в ее освоении. Как и при работе с другой съемочной техникой, дальнейшее изучение и тестирование помогут получить наилучшие результаты с учетом конкретных требований проекта.

DaVinci Resolve Studio



Приложение DaVinci Resolve Studio является комплексным программным пакетом, объединившим в себе инструменты монтажа и грейдинга изображения, добавления визуальных эффектов и сведения звука. Оно поддерживает уникальные характеристики формата Blackmagic RAW Immersive, в том числе стереоскопическое видео в сверхвысоком разрешении, кадровую частоту 90 к/с и сведение пространственного звука. Данная система позволяет создавать, обрабатывать и экспортировать файлы Apple Immersive Video, что превращает ее в мощный ресурс для генерации контента с эффектом погружения.

Рекомендуемые рабочие станции

Рекомендуемое аппаратное обеспечение

- Компьютер Mac Studio M3 Ultra
- Монитор с высокой частотой обновления (минимум 60 Гц)

Минимальные требования к системе

- Платформа macOS 15 или более поздние версии
- Системная память 64 ГБ
- Приложение Blackmagic Design Desktop Video 12.9 или его более поздние версии для мониторинга
- Компьютер на основе процессора Apple Silicon или графический процессор, поддерживающий технологию Metal

Требования для мониторинга иммерсивного контента

- Гарнитура Apple Vision Pro
- Высокоскоростное беспроводное интернет-соединение (минимальный стандарт Wi-Fi 5, предпочтительнее Wi-Fi 6)

Настройки проекта

В целях оптимального визуального качества и удобства монтажа перед загрузкой файлов формата Blackmagic URSA Cine Immersive RAW рекомендуется проверить настройки проекта DaVinci Resolve, чтобы точно выполнить дебайеризацию и обработку исходного материала.

Основные настройки

В самом начале принципиально важно указать, что создаваемый контент предназначен для гарнитуры Apple Vision Pro. Это откроет доступ к соответствующим настройкам и опциям на этапах подготовки, сборки и экспорта проекта.

1. Запустите приложение DaVinci Resolve Studio.
2. В нижнем правом углу окна Project Manager (Менеджер проектов) перейдите к новому проекту.
3. Укажите имя, выберите расположение (если требуется) и нажмите Create (Создать).
4. Откройте настройки проекта щелчком по значку шестеренки в нижнем правом углу приложения DaVinci Resolve Studio.

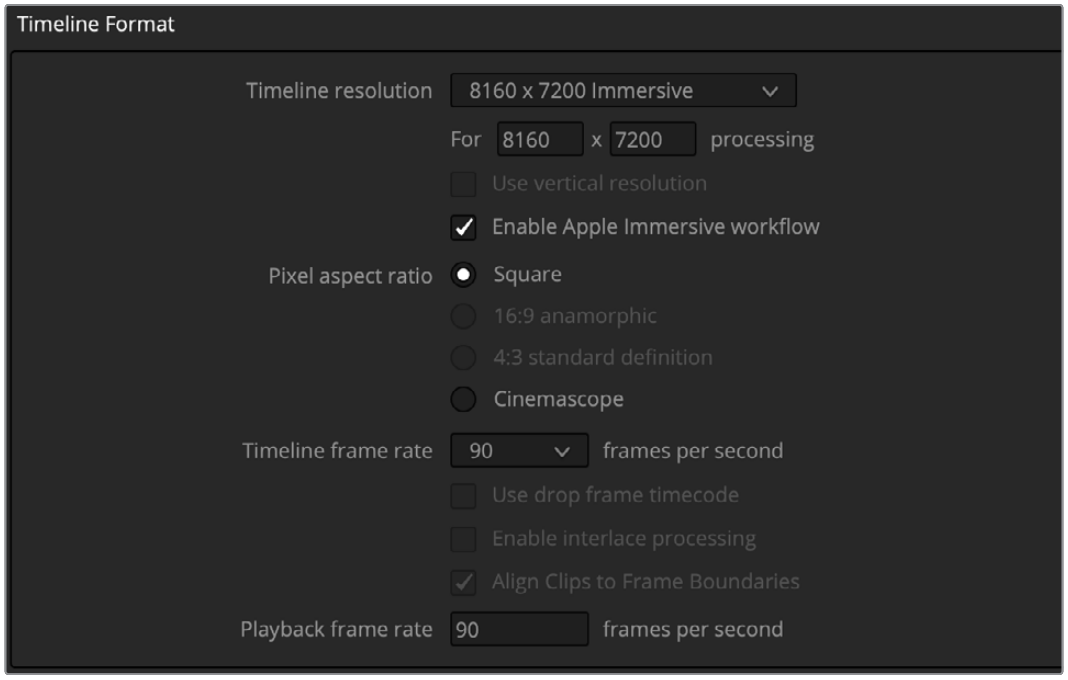
   Сперва необходимо задать разрешение и частоту кадров проекта.
5. На боковой панели перейдите в секцию Master Settings (Основные настройки).
6. В разделе формата временной шкалы активируйте опцию иммерсивного контента Enable Apple Immersive workflow.



   Это позволит отобразить настройки DaVinci Resolve, предназначенные для работы с трехмерным изображением. На начальном этапе создания подобного проекта важно отметить данный режим.
7. Выберите разрешение 8160 x 7200 Immersive для обработки клипов в оригинальном формате.

   При ограниченной мощности можно задать 4080 x 3600 Immersive для более быстрого воспроизведения.
8. Частота кадров временной шкалы должна составлять 90 к/с. При активации режима обработки иммерсивного контента это значение устанавливается автоматически.
9. В целях максимально точного отображения иммерсивных файлов частота воспроизведения должна быть 90 к/с.

   При ограниченной мощности обработки данных (или невысокой частоте обновления монитора) необходимо снизить значение для повышения стабильности системы. Рекомендуется выбрать величину, являющуюся дробной частью оригинальной кадровой частоты, например 30 или 60 к/с.



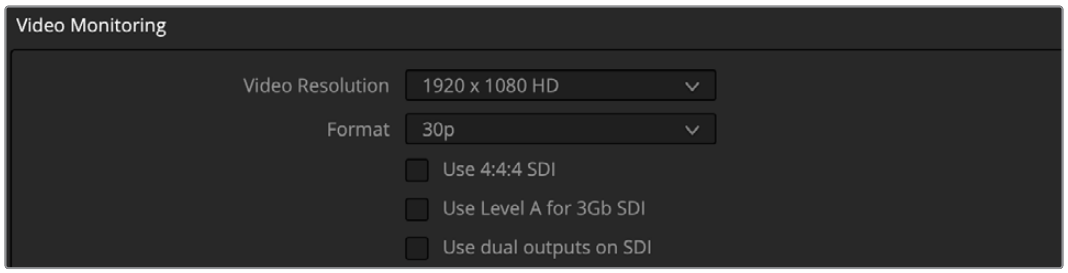
Вывод на видеомонитор

При подключении внешнего дисплея (например, через платы захвата и вывода DeckLink или устройство UltraStudio) настройки мониторинга зависят от его технических характеристик.

1 В секции Video Monitoring (Вывод на видеомонитор) выберите соответствующее разрешение (1920 x 1080 HD или 3840 x 2160 Ultra HD).

   Оригинальная кадровая частота иммерсивного контента составляет 90 к/с, но не все вещательные дисплеи ее поддерживают.

2 В поле формата укажите соответствующее значение (например, 30p).



Масштабирование изображения

Данные параметры обеспечивают корректное отображение разрешения и пропорций выводимого в окно просмотра контента. Они должны автоматически обновиться после активации режима иммерсивного видео в секции основных настроек, но пользователям рекомендуется это проверить.

1 В боковой секции настроек проекта выберите Image Scaling (Масштабирование изобр.).

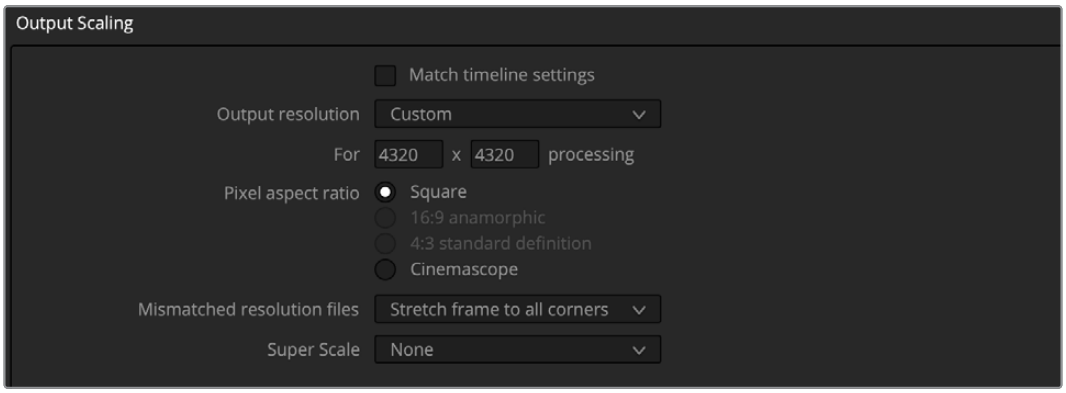
2 В поле Output Scaling (Масштабирование вывода) отмените выбор опции Match timeline settings (Следовать настройкам временной шкалы).

3 Задайте в поле Output resolution (Выходное разрешение) вариант Custom со значениями **4320 x 4320**.

На небольших мониторах и для ускоренного воспроизведения их можно уменьшить вдвое (**2160 x 2160**).

Подобные квадратные пропорции необходимы для просмотра контента в формате Apple Immersive в программной среде DaVinci Resolve Studio и на гарнитуре Apple Vision Pro (по беспроводному интернет-соединению или с помощью ремешка Apple Vision Pro Developer Strap).

4 В поле Mismatched resolution files (Файлы с несоответств. разрешением) необходимо выбрать опцию Stretch frame to all corners (Растянуть кадр во все углы).



Управление цветом

С учетом необходимости индивидуального контроля за максимальной яркостью рекомендуется использовать узлы преобразования цветового пространства на соответствующей странице. Данная секция в настройках проекта предназначена для выбора цветового (или «рабочего») пространства для временной шкалы и сохранения соответствующих метаданных, интегрируемых в финальный файл.

1 На боковой панели настроек проекта перейдите в секцию Color Management (Управление цветом).

2 В поле цветокодировки оставьте вариант DaVinci YRGB.

3 Для цветового пространства временной шкалы укажите DaVinciWG/Intermediate.

Поддерживающий HDR-форматы, данный вариант является идеальным для обработки файлов, записанных на камеру Blackmagic URSA Cine Immersive и создаваемых для просмотра на гарнитуре Apple Vision Pro.

4 В поле Output color space (Выходное цветовое пространство) выберите P3-D65 ST2084 1000 nits.

Данное пространство содержит подходящую палитру для мониторинга контента в приложении DaVinci Resolve на компьютере Mac Studio Desktop, где обычно установлена версия Display P3. При использовании дисплея, предназначенного специально для грейдинга, или другого цветового профиля Apple ICC необходимо выбрать соответствующую опцию.

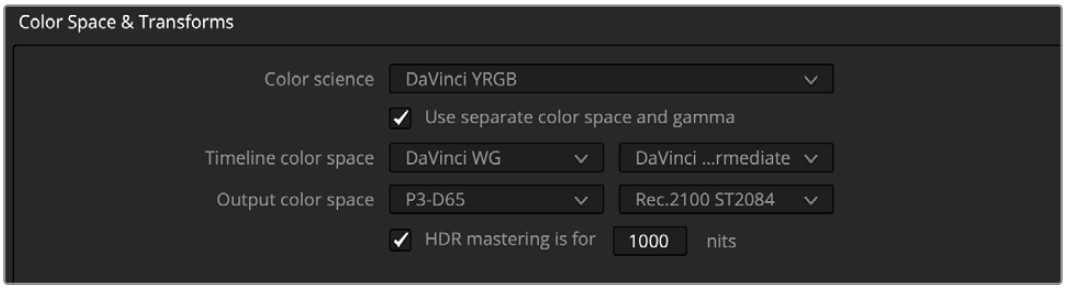
При этом потребуется изменить гамму (функцию передачи) в целях корректной настройки под профиль ICC.

5 Поставьте галочку в поле Use separate color space and gamma (Выбирать цветовое пространство и гамму отдельно).

Внизу отобразятся два раскрывающихся меню для введения соответствующих параметров.

6 Для выходной гаммы (в правой колонке) укажите вариант Rec.2100 ST2084.

7 Отметьте опцию HDR mastering is for 1000 nits (HDR-мастеринг с ярк. 1000 нт).



В целях экономии времени при настройке следующего проекта в формате Apple Immersive можно сохранить задаваемые параметры в виде предустановки.

8 Нажмите на обозначенное тремя точками меню в верхнем правом углу окна и выберите опцию Set Current Settings as Default Preset.

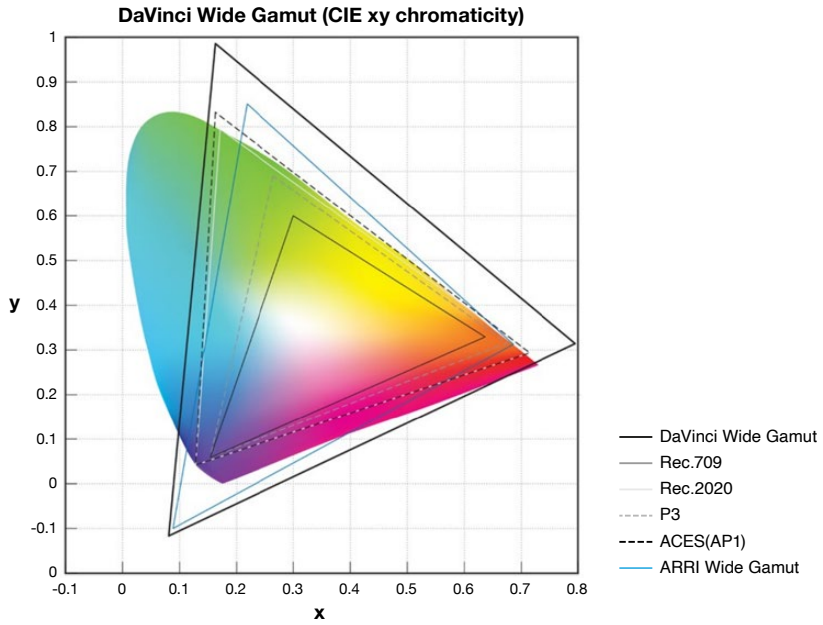
9 Укажите соответствующее имя и нажмите OK.

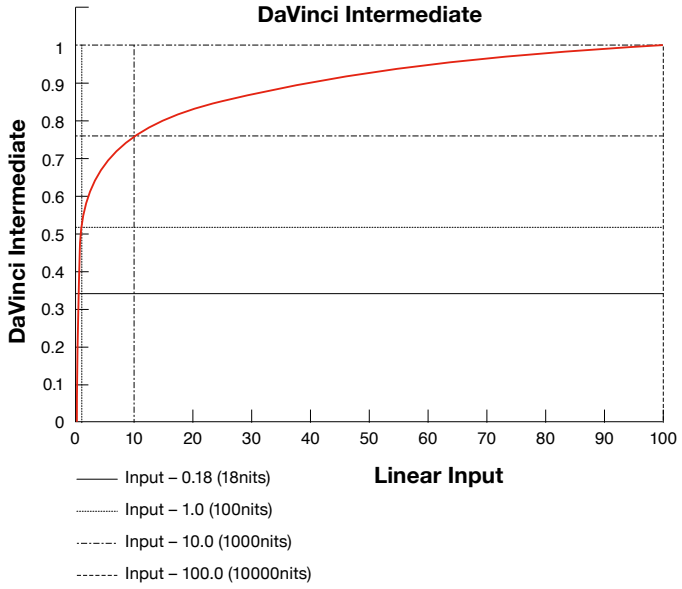
Для быстрой конфигурации следующего проекта в формате иммерсивного видео откройте окно настроек проекта, перейдите в меню опций и выберите появившуюся в списке предустановку.

10 Нажмите Save (Сохранить), чтобы закрыть настройки проекта.

Пространство DaVinci Wide Gamut

Специально созданное для приложения DaVinci Resolve, цветовое пространство DaVinci Wide Gamut является высокоэффективной моделью с расширенным диапазоном палитры для грейдинга и обработки изображений. В сочетании с гаммой DaVinci Intermediate оно обеспечивает единую рабочую систему, оптимизирующую управление цветом при использовании разных форматов, а также согласование тонов и насыщенности на всех этапах, от загрузки и до экспорта финального материала. Данная система идеально подходит для HDR- и иммерсивного контента, позволяя колористам выполнять тонкую коррекцию без появления полос или утраты тональной детализации при создании версий в высоком качестве.





Обработка контента в формате Apple Immersive

На боковой панели настроек проекта в секции Apple Immersive можно задать индивидуальные параметры профиля объектива ILPD (Immersive Lens Profile Data), чтобы модифицировать калибровку и проекцию соответствующих данных.

1. На боковой панели настроек проекта перейдите в секцию Apple Immersive.
2. Нажмите Open Calibration Folder.
3. Добавьте или обновите индивидуальный файл ILPD.
4. Нажмите Update Lists (Обновить списки) для актуализации доступной в медиатеке информации ILPD.
5. Нажмите Save (Сохранить), чтобы закрыть настройки проекта.
6. В секции Media Pool (Медиатека) правой кнопкой щелкните на иммерсивном клипе.
7. В контекстном меню перейдите к опции Apple Immersive Calibration и выберите исходный (No Apple Immersive Calibration File Selected), индивидуальный или промежуточный набор ILPD.

По умолчанию папка ILPD Calibration находится в архиве ProIM там же, где и LUT-директория.

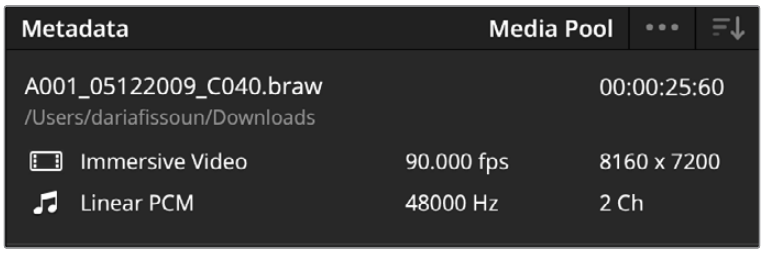
Загрузка

При импорте иммерсивного RAW-контента вместе с файлом сохраняются все соответствующие метаданные, в том числе под оба глаза.

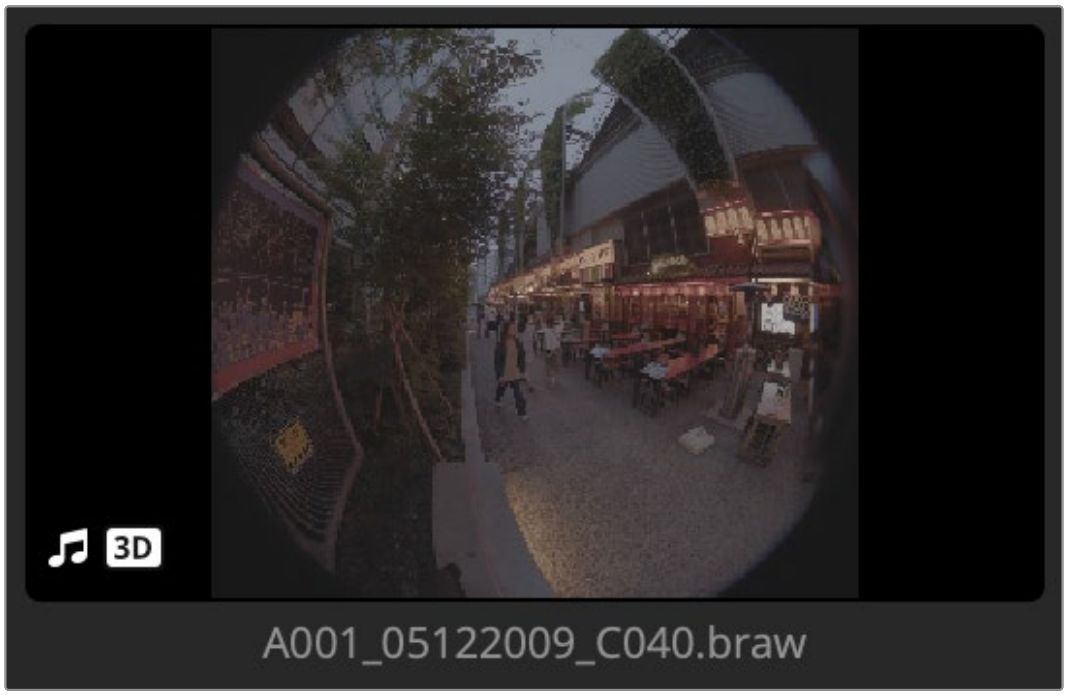
1. Перейдите на страницу Media (Мультимедиа).
2. В разделе Media Storage (Хранилище) найдите иммерсивный файл.
3. Перетащите клип в секцию Media Pool (Медиатека).

СОВЕТ. Чтобы не нарушить оригинальную структуру директории, необходимо перетащить папки из хранилища мультимедиа на боковую панель медиатеки. При переносе в основной ящик (или любой другой созданный пользователем) существующая архитектура не меняется.

Если выбрать клип в формате Apple Immersive, то в заголовке панели метаданных отобразится соответствующий маркер вместе с оригинальной частотой кадров и разрешением.



Кроме того, данный статус видеоклипа дополнительно подтверждается значком 3D в нижнем левом углу пиктограммы.



ПРИМЕЧАНИЕ. При первом перетаскивании иммерсивного видео в медиатеку может открыться диалоговое окно с предложением сохранить или изменить настройки. В последнем случае проекту автоматически присваиваются параметры исходного клипа.

Монтаж

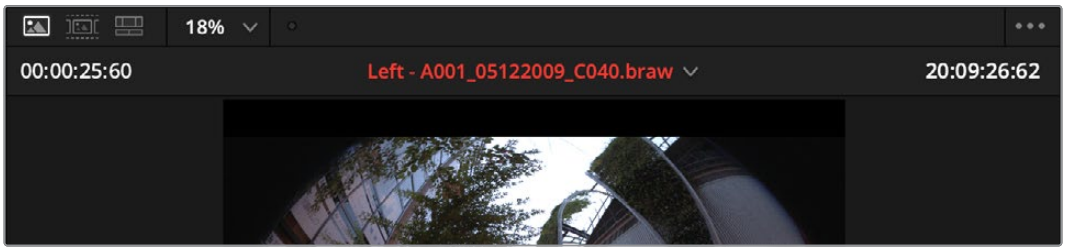
Обработка иммерсивного контента выполняется с помощью таких же инструментов и действий, которые применяются для стандартных форматов двухмерного видео. Ниже описаны операции монтажа, используемые при постпроизводстве исключительно иммерсивного контента.

Сборка

При просмотре клипов в процессе формирования временной шкалы в окне просмотра можно выбирать левый или правый канал. Во всем остальном процесс сборки не отличается от нелинейного монтажа.

1 Перейдите на страницу монтажа.

2 Дважды щелкните кнопкой мыши на клипе, чтобы открыть его в окне источника, расположенном слева. Сверху указано, какой канал выведен на экран.



3 Чтобы сменить угол зрения, правым щелчком нажмите на исходный клип в медиатеке и выберите Stereoscopic 3D > Stereoscopic 3D Mode > Left Eye/Right Eye/Both Eyes.

СОВЕТ. Дополнительно можно навести курсор на эскиз в медиатеке для прокрутки клипов в окне источника, если вы активировали опцию интерактивного просмотра клипов в соответствующем меню.

4 Используйте кнопки I и O, чтобы задать нужный диапазон в окне источника.

5 Нажмите Insert Clip (Вставить клип) на панели инструментов (или используйте клавишу F9) для добавления фрагментов на временную шкалу.

СОВЕТ. Клипы из медиатеки также можно перетаскивать на временную шкалу.

6 Используйте мышь (в режиме выбора диапазона или подгонки клипов) для обработки контента на тайм-линии.

Оформление фона

Если в настройках проекта активирована опция Enable Apple Immersive workflow, то под треком Video 1 автоматически появится фоновая дорожка.

Данный элемент представляет собой область, находящуюся за пределами иммерсивного видеоконтента, который обычно охватывает поле зрения 180°–210°.

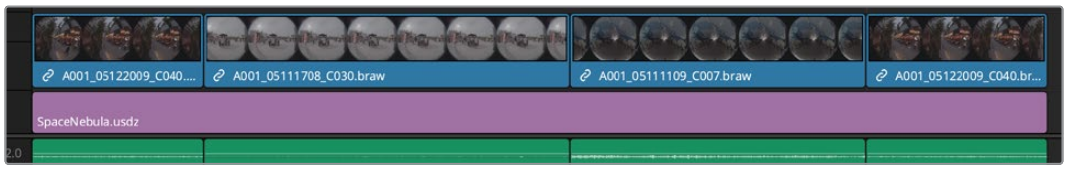
По умолчанию такой фон является черным, но на специальной дорожке можно добавить собственные файлы (в формате .usdz).

1 Перетаскиванием разместите на временной шкале фоновый файл .usdz.

Продолжительность фона должна соответствовать границам клипа на видеодорожке. Один такой файл может использоваться для единичного клипа, их группы или всей временной шкалы, но его нельзя вставить в середину иммерсивного клипа .braw (в качестве косой склейки).



Один клип — можно ✓

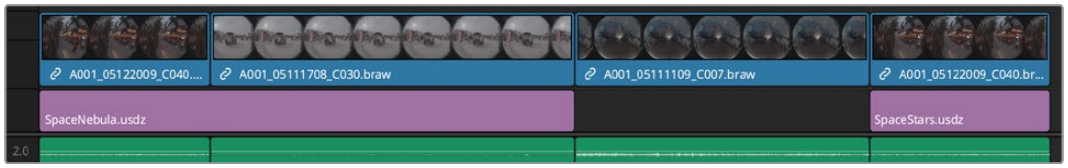


Группа клипов — можно ✓



Косая склейка — нельзя ✗

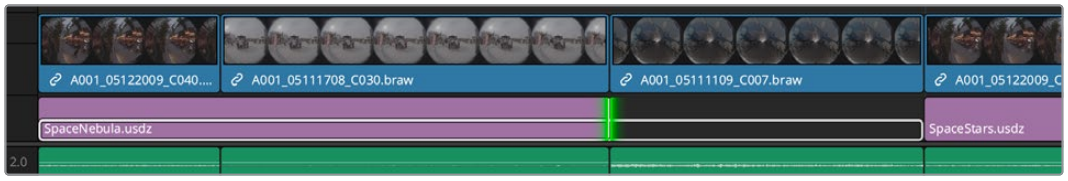
На одной временной шкале можно разместить несколько фоновых дорожек или вообще их не использовать. Тогда по умолчанию будет наложен черный задний план.



Разные кадры с промежутками — можно ✓

2 Продолжительность фонового файла регулируется с помощью стандартных инструментов обрезки.

Перетаскиванием края клипа .usdz (в режиме выбора или подгонки) отрегулируйте его продолжительность.



Для смещения по одному кадру можно также использовать кнопки с запятой («,») и точкой («.»).

СОВЕТ. Чтобы фоновые файлы всегда были совмещены с правками иммерсивного контента, активируйте режим прикрепления на панели инструментов.

Окно просмотра

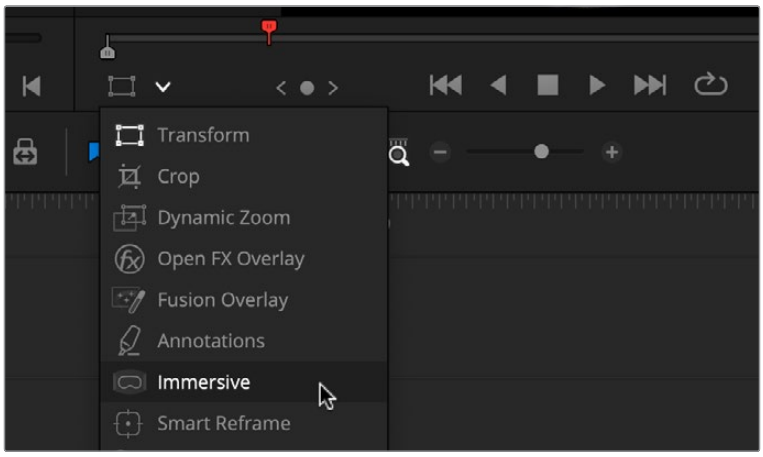
По умолчанию иммерсивные клипы отображаются в приложении DaVinci Resolve в виде сферической проекции. Процесс визуальной калибровки при выводе видео выполняется на камере по отдельности для каждого клипа.



Можно выбрать режим просмотра LatLong (Шир./долг.) или Viewport (в виде настраиваемого окна).

1 В нижнем левом углу окна просмотра временной шкалы нажмите на стрелку раскрывающегося меню рядом с кнопкой экранных органов управления.

2 Выберите Immersive.



Вверху появится новая панель инструментов с дополнительными вариантами отображения контента.

3 Для равнопромежуточной проекции клипа выберите опцию LatLong. Данный режим деформирует изображение для более точного воспроизведения традиционных двухмерных пропорций, что упрощает некоторые процессы постобработки, например отслеживание объектов и композитинг.



4 Выберите опцию Viewport для просмотра центрального фрагмента иммерсивного видео в квадратной рамке. Этот режим позволяет отобразить с правильными пропорциями сцену на стандартном мониторе, при этом важно не забыть об объектах за пределами кадра.

На панели инструментов для иммерсивного контента можно использовать элементы управления трансформацией для наклона и панорамирования стереоскопического изображения, а также для горизонтальной или вертикальной ориентации поля зрения.



Переходы

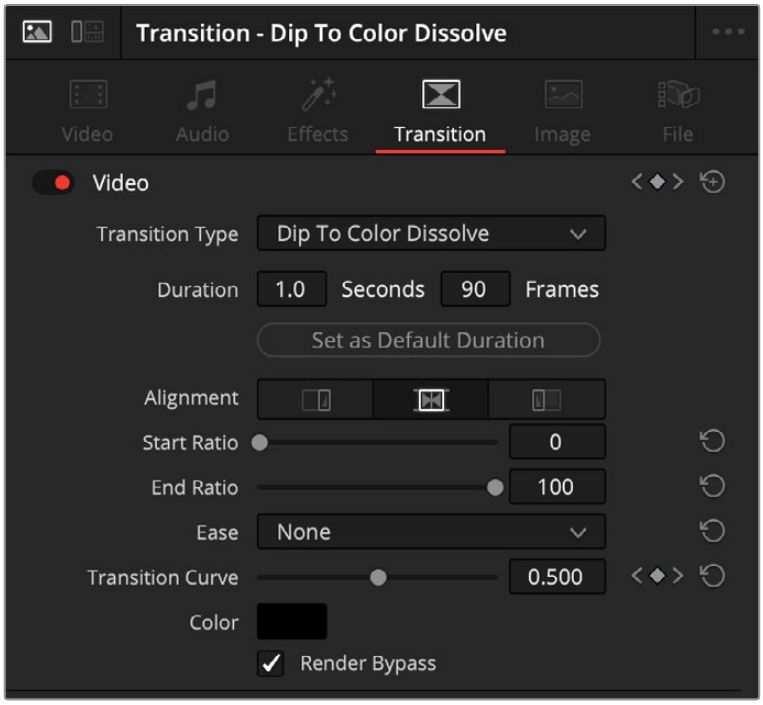
Переход между иммерсивными клипами можно выполнить прямой склейкой, используя буферы с любой стороны для эффекта проявления или затухания, а также погружением в цвет. Их рекомендуется не встраивать в материал, а визуализировать на гарнитуре Apple Vision Pro во время воспроизведения.

1. Откройте панель эффектов.
2. На боковой панели перейдите в меню Toolbox (Панель элементов) > Video Transitions (Видеопереходы)
3. Перетащите эффект Dip To Color Dissolve (Погружение в цвет) на монтажный стык между клипами.



ПРИМЕЧАНИЕ. По обе стороны стыка необходимо иметь достаточно данных (буфер) для подгонки при наложении перехода, иначе он не будет добавлен.

4. На панели Inspector (Инспектор) перейдите на вкладку Transition (Переход) и отметьте опцию Render Bypass. Это обеспечит корректное отображение эффекта на гарнитуре Apple Vision Pro.
5. Затем с помощью других параметров в секции Dip To Color Dissolve (Погружение в цвет) окна Inspector (Инспектор) можно изменить длительность, выравнивание, скорость анимации и цвет перехода.

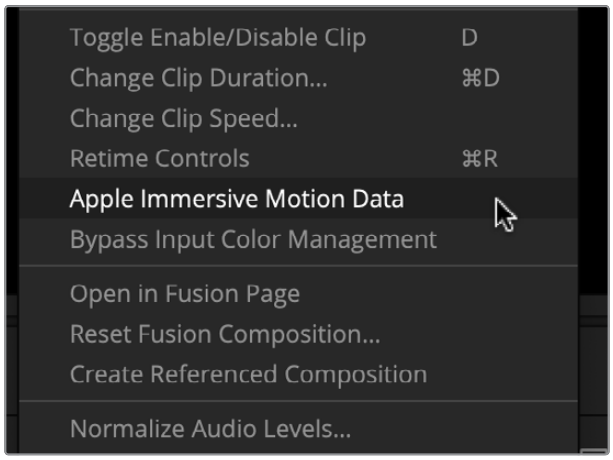


Данные о движении Apple Immersive Motion Data

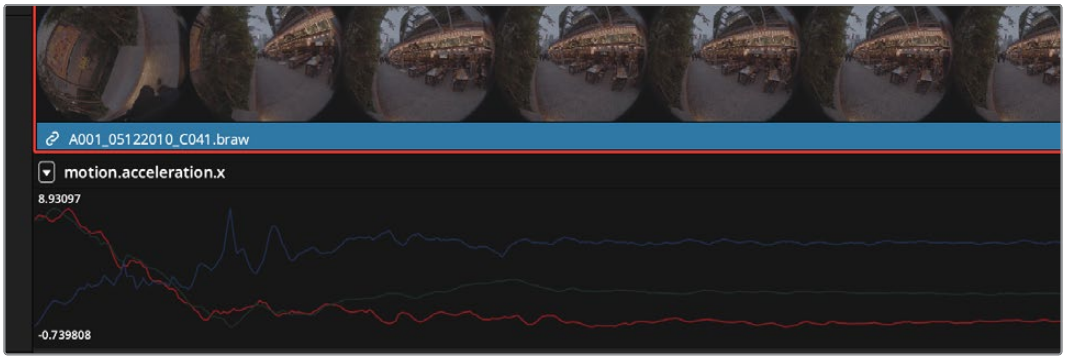
При работе с иммерсивным контентом быстрые либо внезапные движения камеры могут дезориентировать зрителя. Чтобы обеспечить комфорт при просмотре контента, избегайте резких наклонов или внезапного панорамирования плюс вращений и неровной линии горизонта.

На графиках Apple Immersive Motion Data показана информация об ускорении по осям x, y и z, а также отображаются положения гироскопа, чтобы объективно продемонстрировать, где движения камеры и ее перемещение в пространстве отличаются слишком высокой интенсивностью. Редакторы могут использовать эту информацию для выбора подходящих фрагментов видеоматериала.

1. Щелкните правой кнопкой мыши по иммерсивному клипу на временной шкале.
2. Выберите Apple Immersive Motion Data.

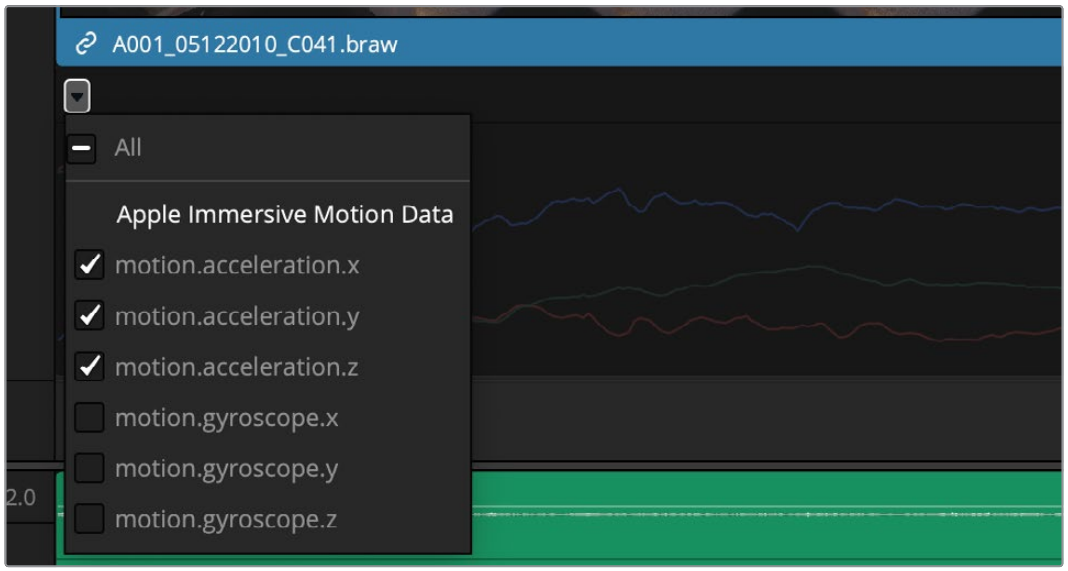


На видеодорожке появится график, отображающий ускорение по осям x, y и z.



В данном примере камера вела запись контента в процессе настройки. Пики на графике указывают на внезапное движение, когда ее поднимали, поворачивали и размещали на плече. В идеале следует использовать последнюю часть видео с более стабильными значениями, после того как для съемочной техники выбрали определенное положение.

3 В левом верхнем углу графика щелкните по стрелке смещения.



Считывание данных можно активировать или отключить в зависимости от содержания кадра и характера движения камеры.

Титры

На странице монтажа в иммерсивную сцену можно добавить текст, используя любой из инструментов на панели эффектов, например Scroll (Прокрутка), Text (Текст), Text+ (Текст+) или MultiText, а также любой из вариантов бегущей строки.

При размещении поверх иммерсивного клипа генератор титров будет полагаться на данные объектива для точного искажения надписей в соответствии с оптической деформацией.

Если такие данные недоступны (например, при добавлении текста поверх черного фона или иммерсивной сцены с компьютерной графикой), будет применено искажение по умолчанию, чтобы обеспечить корректный вид изображения при просмотре в гарнитуре Apple Vision Pro.

1 В разделе эффектов выберите Toolbox (Панель элементов) > Titles (Титры).

2 Перетащите генератор текста на временную шкалу и разместите на видеодорожке над иммерсивным клипом, чтобы включить его в сцену.



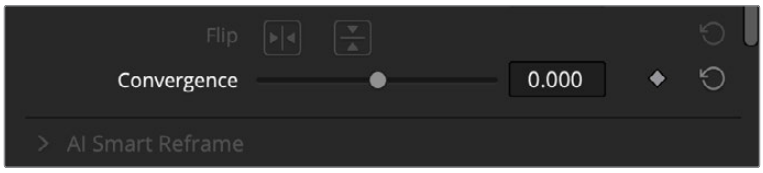
Как и в случае с фонами, генераторы текста не могут пересекать границы клипа. Поскольку рендеринг каждого фрагмента предусматривает различные данные объектива, существует риск искажения титров между кадрами.

3 Обрежьте заголовок по точкам редактирования видео.

4 Используйте панель Inspector (Инспектор) для ввода и форматирования текста.

5 Перейдите на вкладку настроек и используйте слайдер Convergence (Схождение), чтобы расположить текст в трехмерном пространстве.

Переместите его влево или вправо, чтобы сдвинуть заголовок назад либо вперед.



Схождение лучше всего видно при отслеживании изображения на гарнитуре Apple Vision Pro. В ходе использования других средств мониторинга его можно отрегулировать в окне просмотра, изменив режим Vision (Изображение) базового видеослоя на странице цвета.

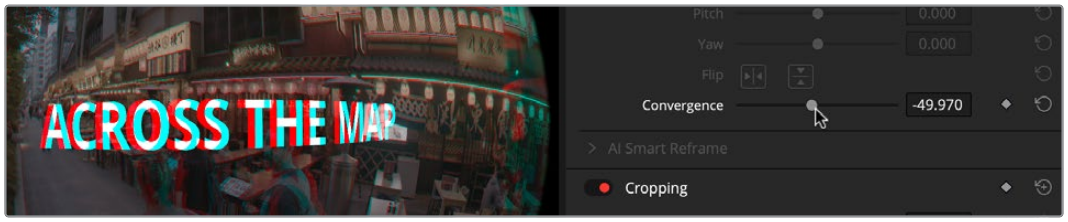
6 Перейдите на страницу цвета.

7 Откройте панель 3D-инструментов.

Столбец Vision (Изображение) в центре позволяет изменить стереоскопический стандарт, используемый для проецирования иммерсивного фрагмента.

8 Выберите вкладку Stereo (Стерео).

9 Измените режим Out (Выход) на Anaglyph (Color) (Анаглиф (Цвет).



Изменить параметр схождения на панели Inspector (Инспектор) в разделе монтажа проще при просмотре изображения через анаглифные очки.

Творческие соображения

На ранних этапах работы с камерой Blackmagic URSA Cine Immersive и гарнитурой Apple Vision Pro кинематографисты могут свободно экспериментировать с форматом съемки, чтобы определить наиболее оптимальные подходы к созданию контента и расширить границы традиционного производства.

Некоторые аспекты получения иммерсивного изображения уже отмечены как требующие особого внимания для обеспечения максимального усвоения материала аудиторией и визуального комфорта.

1 Редактирование контента нужно начинать на этапе подготовки к производству. Из-за особых требований к продолжительности плана и уровню визуального комфорта при движении камеры кинематографистам необходимо снимать сцены с учетом финальной версии монтажа.

2 Избегайте использования клипов с быстрыми движениями камеры, поскольку они могут дезориентировать зрителя. Вместо этого отдайте предпочтение кадрам, снятым из фиксированного положения, при медленном перемещении объектива или с применением операторских тележек.

3 Несбалансированные или слишком экстремальные ракурсы также могут дезориентировать. Выбирайте дубли, где камера находится на одном уровне с объектом съемки и на удобном расстоянии от него (не слишком близко).

4 Освоение среды при съемке иммерсивного контента длится дольше по сравнению с созданием 2D-материала. Переходы нужно выполнять в более медленном режиме, чтобы у зрителей было достаточно времени для адаптации к новой обстановке.

5 При переходе между клипами стоит учитывать, куда был направлен взгляд зрителя на предыдущих кадрах. Ключевые события не должны разворачиваться в левой части экрана, если ранее внимание аудитории было приковано к области справа.

6 Пространственный звук во многом определяет повествование. Отнеситесь к нему как к еще одному персонажу при создании захватывающих проектов.

Композитинг с использованием визуальных эффектов

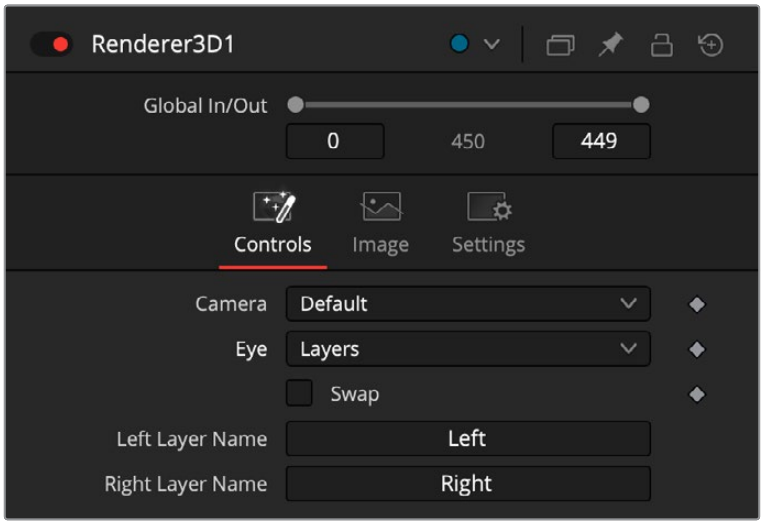
Если в настройках проекта активирован режим Apple Immersive, специальные параметры Fusion служат для обеспечения работы в стереоскопическом формате и иммерсивного композитинга. Узлы MediaOut позволяют выводить изображение для левого и правого глаза, а инструмент Immersive Patcher дает возможность получения равнопромежуточной проекции и соответствующей композиции.

Узлы MediaOut

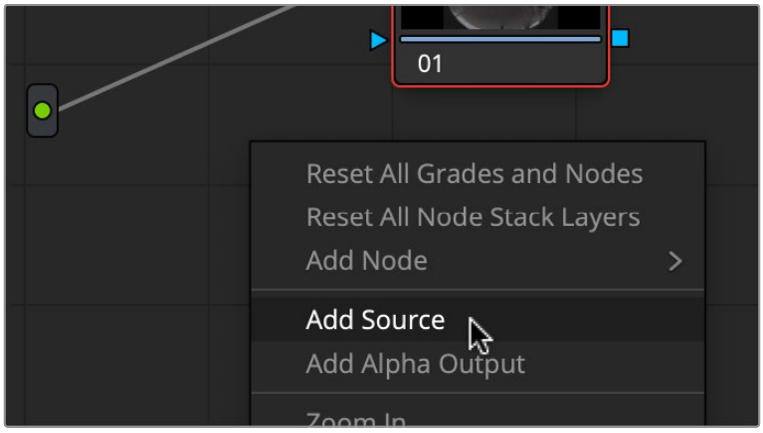
Когда страницу Fusion впервые открывают во время работы над иммерсивным клипом, на графике отображаются два узла MediaIn, соединенные с двумя соответствующими элементами MediaOut.



На панели MediaOut Inspector параметр Stereo Eye можно настроить для вывода сигнала глаза.



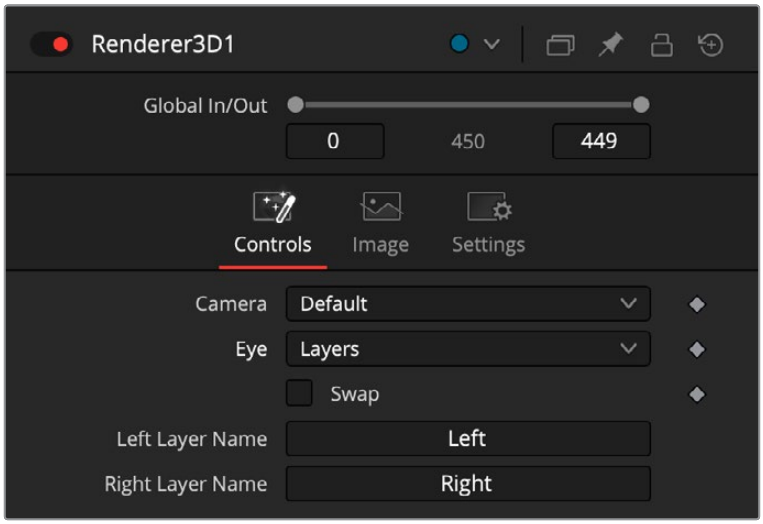
По умолчанию узел MediaOut1 соответствует левому глазу, а MediaOut2 — правому. Такая конфигурация позволяет сразу приступить к созданию стереокомпозиций и наложению графики. Двойные узлы MediaOut также обеспечивают отправку пары вариантов сигнала в редактор узлов на странице цвета для дальнейшей корректировки изображения. Доступ к сигналу MediaOut2 в данном разделе можно получить, щелкнув правой кнопкой мыши в редакторе узлов и выбрав опцию Add Source (Добавить источник).



Если для параметра Stereo Eye задан автоматический режим, узел MediaOut объединяет сигналы и назначает каждый из них для соответствующего глаза на временной шкале. Это позволяет использовать многослойный рабочий процесс Fusion для их отправки на один узел Media Out.

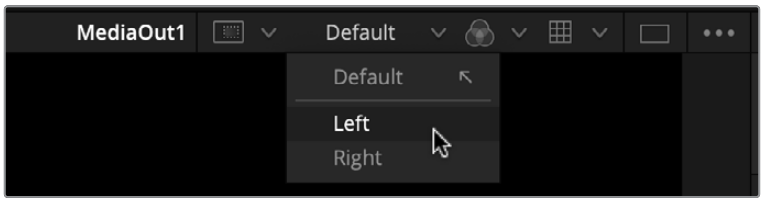
Узел Renderer3D

При работе с 3D- или компьютерной графикой на иммерсивной временной шкале с помощью настроек узла Renderer3D можно выполнять обработку левого и правого слоев.



Это позволяет создавать сложные 3D-титры или целые сцены на базе компьютерной графики и выгружать их непосредственно на стереоскопическую временную шкалу в ходе подготовки проектов с эффектом погружения.

В подобной ситуации можно использовать раскрывающееся меню в правом верхнем углу окна Fusion для переключения между левым и правым каналами композиции.



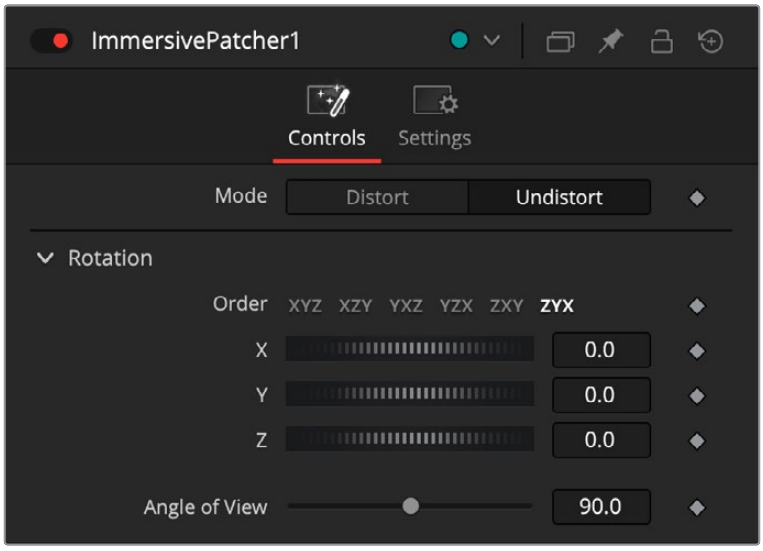
Композитинг с помощью инструмента Immersive Patcher

Инструмент Immersive Patcher позволяет временно устранить искажения в определенном фрагменте иммерсивного клипа, что упрощает отслеживание, рисование и композитинг в плоском формате. После завершения процесса можно еще раз прибегнуть к этому же инструменту для повторной деформации иммерсивного сигнала и добавления полученных элементов в сцену.

1 Добавьте инструмент Immersive Patcher в конфигурацию узлов.



2 На панели Inspector (Инспектор) щелкните Undistort.



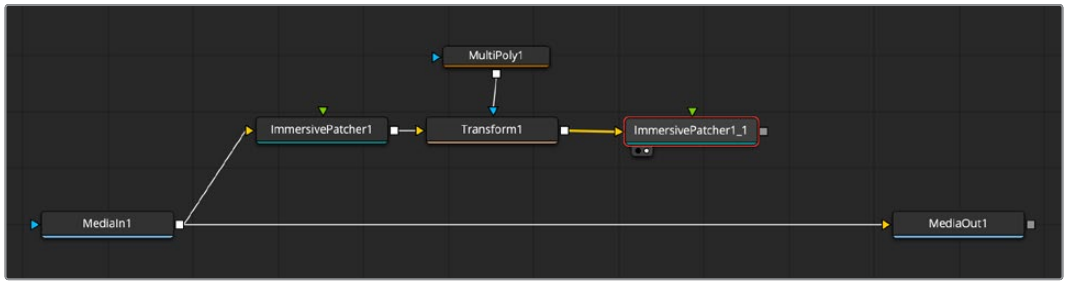
3 Используйте параметры вращения X, Y и Z, чтобы выбрать фрагмент иммерсивного изображения, который необходимо сделать плоским.

4 Перейдите к настройкам комбинированного изображения так же, как и при работе со стандартным 2D-сигналом. В список доступных опций входят использование стандартных методов рисования для удаления нежелательных объектов плюс выборка участков кадра с помощью средств трансформации и полигонов для расширения сцены либо добавления масок, текста и внешних видеослоев при замене знаков или экранов.

После завершения необходимо повторно исказить плоское изображение, чтобы оно соответствовало параметрам исходной иммерсивной деформации.

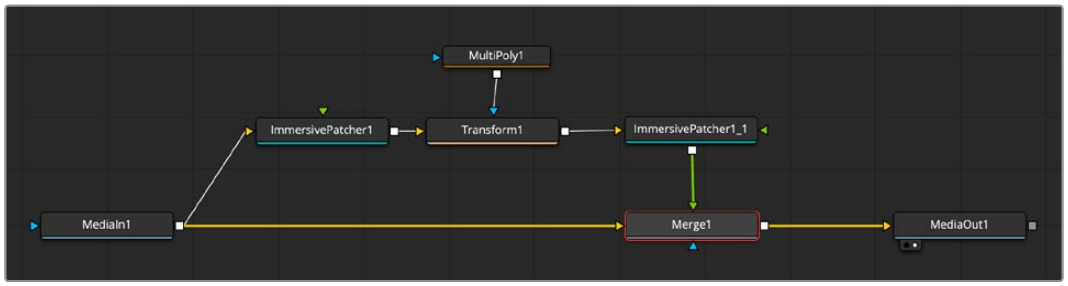
5 Скопируйте данные узла Immersive Patcher, чтобы сохранить исходную информацию о вращении и угле обзора.

6 Вставьте соответствующие данные после узлов композитинга и выполните соединение с конечным элементом конфигурации.



7 На панели Inspector (Инспектор) щелкните Distort.

8 Объедините комбинированный фрагмент с исходным клипом, чтобы добавить эффект в иммерсивную сцену.



При совместной работе после этого можно закрыть страницу Fusion и продолжить монтаж, микширование и грейдинг изображения на иммерсивной временной шкале в приложении DaVinci Resolve.

При выполнении операций в иммерсивном режиме используйте формат EXR в цветовом пространстве ACES AP0 для рендеринга сигнала каждого глаза, чтобы сохранить максимальное количество информации при экспорте изображения.

Цветокоррекция

В этом разделе рассматриваются оптические, цветовые и 3D-инструменты, которые можно использовать при работе с иммерсивным изображением на странице цвета в приложении DaVinci Resolve.

Ввиду уникальных свойств такого контента не все инструменты доступны для применения на этапе постобработки. Ряд средств в настоящее время находится в процессе адаптации для поддержки данного формата.

Шумоподавление

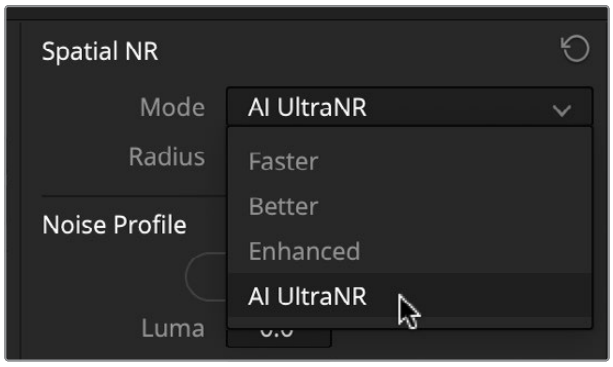
При работе с иммерсивным видео можно применить шумоподавление. На панели Motion Effects (Эффекты движения) инструмент Spatial NR (Пространств. шумоподавление) анализирует помехи исходного контента, чтобы максимально устранить их, сохранив при этом исходную детализацию изображения.

1 Откройте секцию Motion Effects (Эффекты движения) на панели инструментов страницы цвета.

В столбце Spatial NR (Пространств. шумоподавление) можно выбрать желаемый режим.

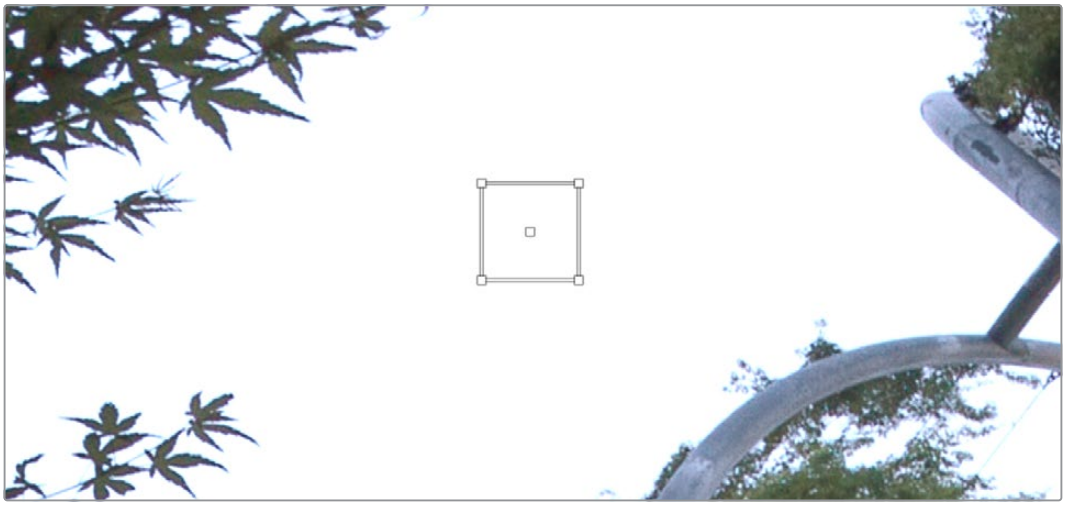
- В режимах Faster (Быстрее), Better (Качественнее) и Enhanced (Улучшенный) будет выполнен анализ помех в пределах заданного радиуса, причем его результат соответствует установленному пороговому значению. Они перечислены в порядке сложности обработки.
- Инструмент AI UltraNR представляет собой идентификатор на базе искусственного интеллекта, который извлекает помехи из сигнала, обнаруживая и анализируя самый плоский участок кадра на предмет чистого шума.

2 Выберите режим пространственного шумоподавления AI UltraNR.



3 Нажмите Analyze (Анализировать).

В окне просмотра появится рамка, определяющая самую пустую область пиксельных данных в кадре. Поскольку внутри нее не содержится визуальный контент, она идеально подходит для извлечения цифрового шума, создаваемого матрицей камеры.



При желании можно изменить положение и размер рамки в окне просмотра, но обычно это не рекомендуется, поскольку инструмент AI UltraNR лучше справляется с обнаружением оптимальных областей для анализа, чем глаза.

Параметры порогов яркости и цветности для профиля шума можно настроить вручную, хотя инструмент AI UltraNR обычно автоматически определяет идеальное сочетание значений для наилучшего результата.

Первичный грейдинг

Первичную цветокоррекцию иммерсивного изображения, включая его нормализацию, балансировку и согласование на основе кругов цветового и логарифмического грейдинга, выполняют так же, как и обработку 2D-контента. То же самое касается панелей Color Slice и Color Warper (Тональный корректор).

Выборочная цветоустановка

Методы выборки на основе пикселей, такие как квалификатор, тоже можно использовать при работе с иммерсивным контентом.

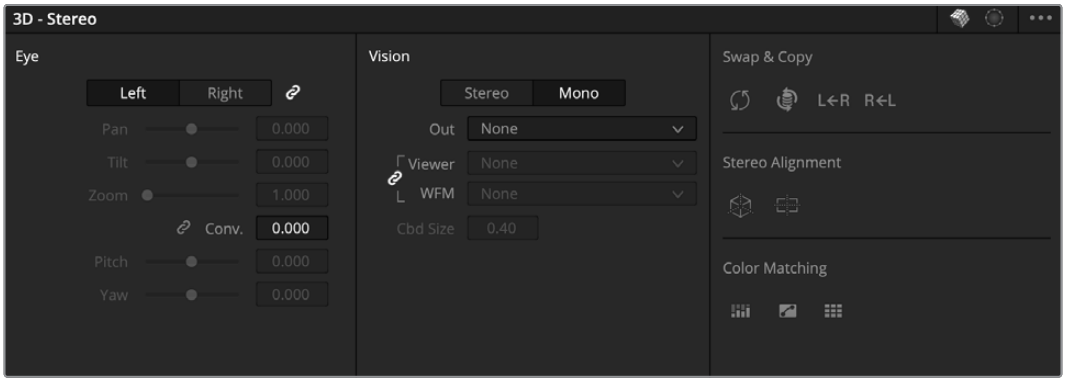
Точная селекция на основе объектов или окон поддерживается только в плоской равнопромежуточной среде, а это означает, что большинство средств продвинутой цветоустановки, включая Power-зоны и другие инструменты на панели отслеживания плюс волшебную маску, в настоящее время недоступны.

Управление размером

Инструменты управления размером (например, масштабирование и панорамирование) недоступны при работе с иммерсивным контентом.

Панель 3D-инструментов

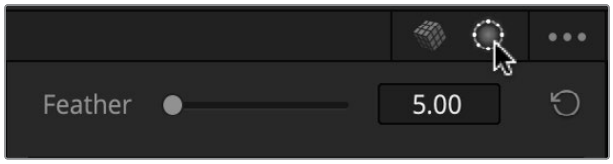
На панели 3D-инструментов содержатся опции управления различными 3D-форматами камеры и графики. В отличие от других стереоскопических стандартов, здесь не нужно настраивать положение глаз, поскольку параметры оптики, фокусного расстояния и диафрагмы на модели Blackmagic URSA Cine Immersive фиксированы.



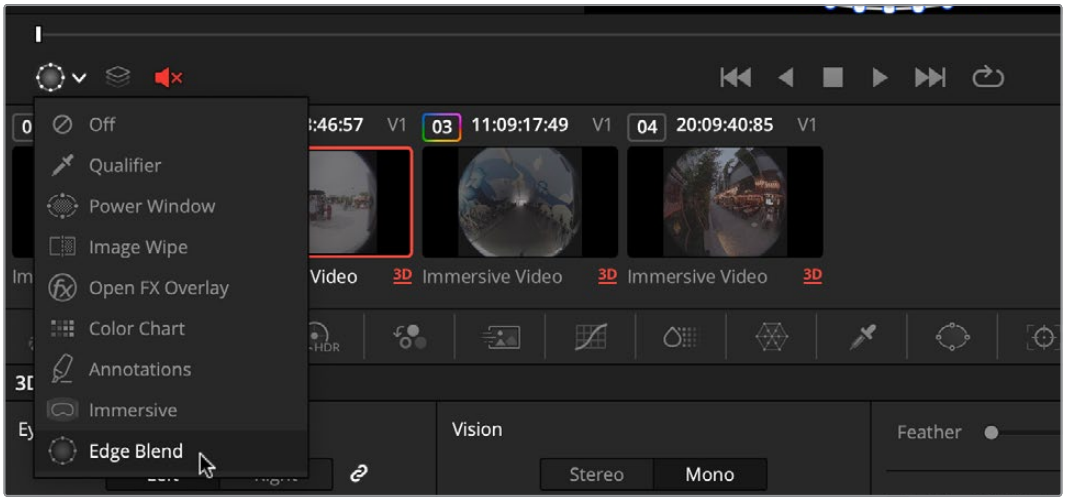
Режим Edge Blend доступен только для иммерсивного контента. Он позволяет регулировать и смягчать периметр соответствующего видеокадра. Его можно использовать, чтобы скрыть кинооборудование, находящееся на площадке, и ограничить поле зрения наблюдателя. За заданной границей будет отображаться фон.

1 Откройте панель 3D-инструментов.

2 В правом верхнем углу измените режим на Edge Blend.



3 Убедитесь, что для экранных элементов управления выбран соответствующий режим.



По периметру видеокадра должен располагаться круг контрольных точек с меткой в центре.

4 Перетащите контрольную точку, чтобы изменить ее положение.



Также можно щелкнуть по ней правой кнопкой мыши, чтобы удалить точку.

5 В правом столбце панели 3D-инструментов используйте параметр Feather (Растушевка), чтобы смягчить края рамки.

6 Выберите настройку visionOS Effect для обеспечения корректной обработки новых данных во время воспроизведения на гарнитуре Apple Vision Pro. В противном случае информация будет встроена в видеофайл для рендеринга.

7 В левом столбце используйте соответствующие вкладки для переключения между глазами в окне просмотра.

Обратите внимание, что многие элементы управления на панели 3D-инструментов неактивны. Это происходит, когда их невозможно использовать при работе с текущим материалом, например иммерсивным контентом.

8 В центральном столбце используйте вкладки Stereo (Стерео) и Mono (Моно) для переключения между соответствующими режимами оптики при просмотре иммерсивного видеоклипа.

Управление цветом в CST-режиме

Использование эффекта преобразования цветового пространства (CST) при работе с узлами на странице цвета позволит выполнить точное назначение палитры и отрегулировать максимальный уровень яркости временной шкалы.

Рекомендуемый рабочий процесс основан на четырех перечисленных ниже узлах.

1. INPUT CST – преобразование между камерой и используемым цветовым пространством.

   Грейдинг выполняется между этими двумя узлами.

2. OUTPUT CST – преобразование между используемым цветовым пространством и HDR-дисплеем Mac.
3. MONITOR CST – преобразование между 1000 и 250 нт для HDR-мониторинга.
4. AVP CST – преобразование между 250 и 108 нт для мониторинга с помощью гарнитуры Apple Vision Pro и экспорта финальной версии.



Назначение временной шкалы в CST-режиме

Сначала выполняется согласование цветового пространства камеры с соответствующими параметрами временной шкалы. Модель Blackmagic URSA Cine Immersive использует палитру Blackmagic Design Wide Gamut Gen 4/5 с логарифмической кривой DaVinci Intermediate.

1. Откройте панель эффектов.
2. Перетащите эффект Color Space Transform (Преобразование цветового пространства) на пустой узел (01) в соответствующем редакторе.

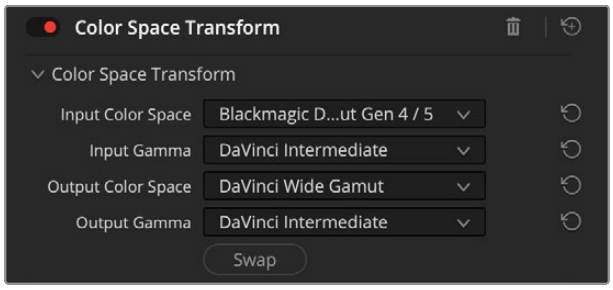
   При согласовании параметров источника и временной шкалы этот узел должен находится в начале конфигурации элементов и перед любыми операциями грейдинга.

   При согласовании параметров источника с настройками цветового пространства, используемого для мониторинга или экспорта, этот узел должен находится в конце конфигурации элементов после любых операций грейдинга.

3. Обозначьте узел 01 как **INPUT CST**.

ПРИМЕЧАНИЕ. Идентификаторы узлов, используемые в данном руководстве, сделаны для удобства ссылок на документацию и снимки экрана. Можно выбрать любой принцип присвоения имен или вообще отказаться от маркировки.

4. Для настройки Input Color Space (Входное цветовое пространство) выберите опцию Blackmagic Design Wide Gamut Gen 4/5.
5. Для настройки Input Gamma (Входная гамма) задайте параметр DaVinci Intermediate.
6. Для настройки Output Color Space (Выходное цветовое пространство) выберите опцию DaVinci Wide Gamut.
7. Для настройки Output Gamma (Выходная гамма) задайте параметр DaVinci Intermediate.



8 Добавьте новый последовательный узел (02).

9 Скопируйте и вставьте первый узел (01) в новый узел (02).

10 Обозначьте узел 02 как **OUTPUT CST**.

Параметры цветового пространства и палитры на выходе должны использоваться в качестве входных данных в этом последовательном узле, чтобы гарантировать необходимое согласование на временной шкале узловой структуры.

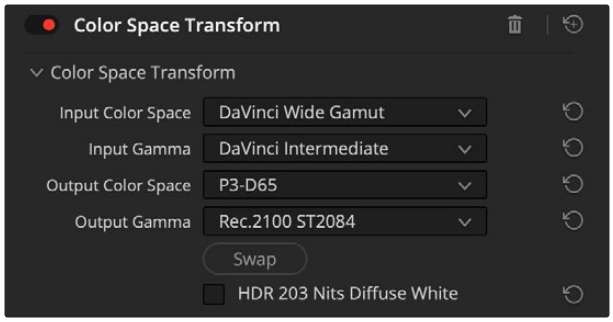
11 Нажмите кнопку Swap (Поменять), чтобы поменять местами параметры на входе и выходе.

Параметры цветового пространства и гаммы на входе теперь соответствуют данному узлу CST.

12 Для настройки Output Color Space (Выходное цветовое пространство) выберите опцию P3-D65. Ее по умолчанию применяют для дисплеев Mac.

13 Для настройки Output Gamma (Выходная гамма) выберите опцию Rec.2100 ST2084 (HDR-стандарт для сигнала 1000 нт).

Все операции грейдинга будут происходить между этими двумя узлами. При работе с системой управления цветом на основе узлов также появляется уникальная возможность выполнения сторонних операций вне временной шкалы, охватывающих некоторые аспекты цветокоррекции, а также добавление эффектов и графики.



Профили ICC

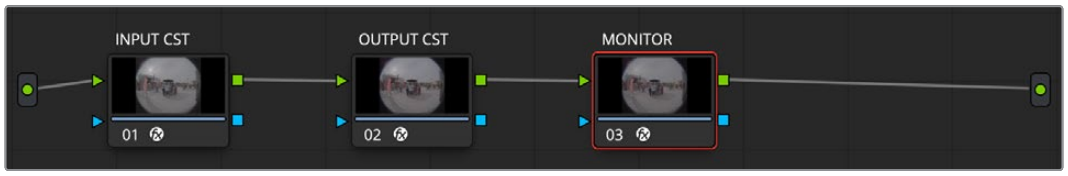
Чтобы убедиться, что сигнал камеры URSA Cine Immersive корректно отображается на дисплее Mac, перейдите в раздел «Системные настройки» > «Дисплеи» и задайте для цветового профиля значение Apple XDR Display (P3-1600 нт).

Настройка монитора для грейдинга в CST-режиме

Для временного ограничения максимальных значений нт при мониторинге в процессе грейдинга можно использовать дополнительные инструменты CST.

1 Создайте новый последовательный узел (03) после узла управления цветом OUTPUT CST (02).

2 Обозначьте узел 03 как **MONITOR**.



3 Добавьте эффект преобразования цветового пространства к узлу MONITOR (03).

Входные значения должны соответствовать параметрам исходящего сигнала предыдущего узла CST.

4 Для настройки Input Color Space (Выходное цветовое пространство) выберите опцию P3-D65.

5 Для настройки Input Gamma (Входная гамма) задайте Rec.2100 ST2084.

Целью данного узла является исключительно понижение яркости изображения в нт, поэтому входные и выходные параметры могут оставаться прежними.

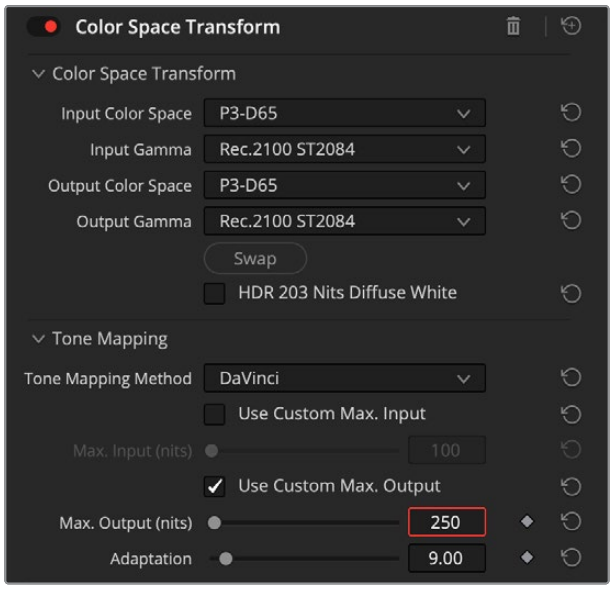
6 Для настройки Output Color Space (Выходное цветовое пространство) выберите опцию P3-D65.

7 Для настройки Output Gamma (Выходная гамма) задайте Rec.2100 ST2084.

ПРИМЕЧАНИЕ. В некоторых случаях не требуется отдельно устанавливать параметры цветового пространства или гаммы, поскольку они идентичны на входе и выходе, и их согласование не производится. В данном случае это необходимо, так как применение пользовательской настройки яркости зависит от значения и формы логарифмической кривой сигнала.

8 Выберите Use Custom Max. Output (Использовать настраиваемый макс. выход).

9 Установите максимальную яркость в нт на **250**.



После этого с использованием инструмента DaVinci Tone Mapping Method произойдет преобразование контента 1000 нт в 250 нт.

В результате будет получено изображение, максимально соответствующее сигналу гарнитуры Apple Vision Pro на HDR-мониторе для грейдинга (например, Flanders XMP310 или Sony BVM-HX310).

Тем не менее для мониторинга проекта с помощью решения Apple Vision Pro потребуется выполнить еще одну операцию.

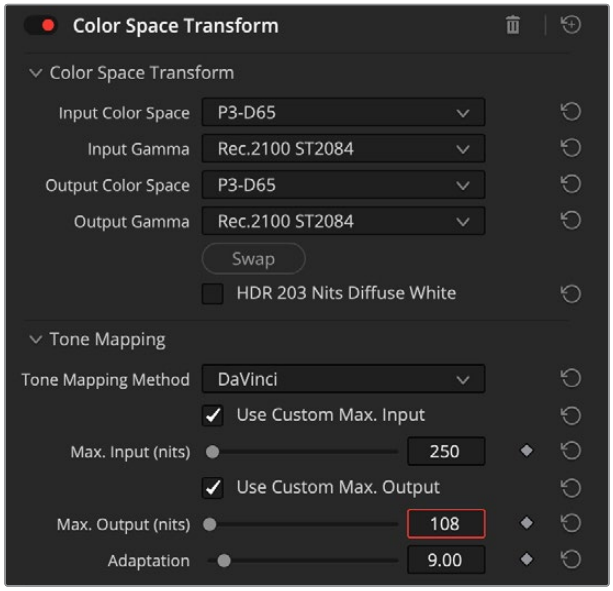
Настройка параметров при мониторинге в гарнитуре Apple Vision Pro в CST-режиме

Финальный узел CST можно использовать для мониторинга в реальном времени, предварительного просмотра изображения и подготовки окончательной версии контента с применением устройства Apple Vision Pro.

1. Создайте новый последовательный узел (04) после узла MONITOR (03).
2. Скопируйте и вставьте узел MONITOR (03) в узел (04). Будут использоваться одинаковые входные и выходные параметры. Изменится только значение нт.
3. Обозначьте узел 04 как AVP.



4. Выберите Use Custom Max. Input (Использовать настраиваемый макс. вход).
5. Установите максимальный входной сигнал на **250** нт, чтобы корректно преобразовать сигнал узла MONITOR.
6. Установите максимальный выходной сигнал на **108** нт, что является максимальным значением для мониторинга и подготовки финальной версии с использованием гарнитуры Apple Vision Pro.



Теперь можно приступить к грейдингу, отключая узлы CST, если это необходимо. Например, во время начального этапа цветокоррекции, в ходе которого используется внешний монитор, следует отключить узел AVP (04), чтобы обеспечить корректное отображение сигнала яркостью 250 нт на HDR-мониторе. При предварительном просмотре изображения в гарнитуре Apple Vision Pro или для подготовки окончательной версии повторно активируйте узел AVP (04).

Можно ли поместить элементы CST в редактор узлов на временной шкале?

Нет. В настоящее время соответствующий функционал полагается на точное применение параметра Color Space Transform Adaptation, которое выполняется для каждого клипа отдельно.

Аудио

Решение Apple Vision Pro и приложение DaVinci Resolve поддерживают пространственный и объемный звук, а также форматы AVP (Apple Vision Pro) и AIA (Apple Immersive Audio).

При работе с пространственным звуком многие аспекты постобработки остаются неизменными. Одно из главных отличий заключается в том, что вместо мониторинга с помощью профессиональных динамиков необходимо использовать наушники или модель AirPods для настоящего погружения в среду и качественного воспроизведения соответствующих характеристик.

В то время как при работе с 2D-аудио сведение большей части диалогов происходит на базе центральных каналов, для пространственного звука требуются данные исходного окружения. Для микширования сигналов таким образом, чтобы они не перегружали и не дезориентировали слушателя, необходимо использовать приглушение, эквализацию и балансировку.

Элементы погружения в пространство можно сочетать с обычной стереогруппой, обеспечивая последовательность определенных звуков независимо от положения головы. Это полезно для сохранения ясности и легкости интерпретации недиегетических звуков, таких как музыкальная партитура и закадровый голос.

Звуки окружающей среды в пространственном аудио обычно становятся более тихими. В ходе микширования их часто приходится усиливать, однако это не должно происходить за счет дезориентации слушателя.

Реверберация, которую рекомендуется применять при работе с пространственным аудио, в значительной степени способствует созданию эффекта погружения, поскольку слушатели естественным образом ощущают ее каждый день. Каждый звук в определенной среде отражается от окружающих поверхностей, обеспечивая соответствующее качество дорожки.

Формат Apple Immersive Audio (AIA) содержит приведенные ниже стандарты.

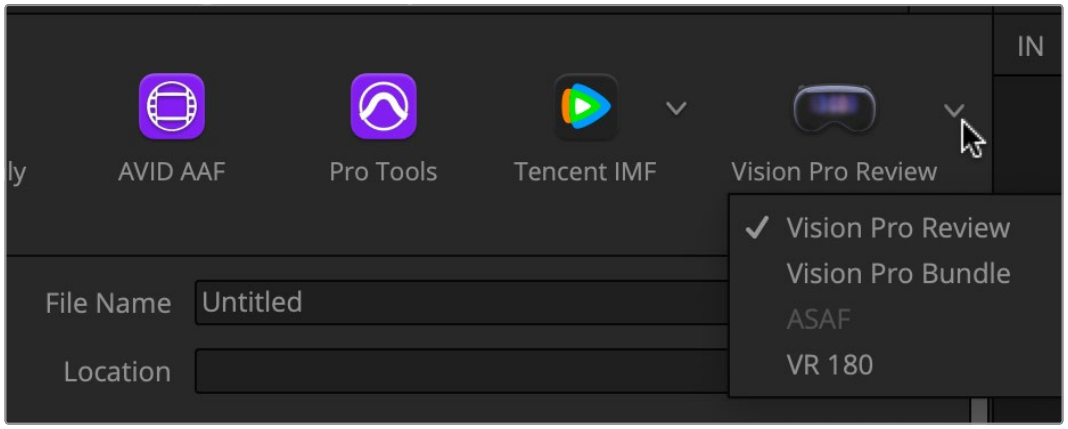
- **Apple Spatial Audio Format (ASAF)** — рабочий аудиоформат RAW.
- **Apple Positional Audio Codec (APAC)** — закодированный формат .mp4, воспроизводимый на устройствах с поддержкой APAC, таких как наушники AirPods или файлы AIVU на гарнитуре Apple Vision Pro.

Формат Apple Spatial Audio (ASAF)

Стандарт Apple Spatial Audio Format (ASAF) представляет собой формат иммерсивного звука, разработанный в первую очередь для приложений виртуальной и дополненной реальности при использовании гарнитуры Apple Vision Pro. Он поддерживается только на платформе macOS, и для работы с ним требуются станция ProTools в сочетании с соответствующими плагинами и приложение DaVinci Resolve. Формат обладает расширенными техническими возможностями, включая встроенную поддержку объемных источников и маршрутизацию посредством назначения шин. Микширование происходит в виртуальном режиме без физических шин, и пользователи могут выбирать различные варианты мониторинга, например бинауральный с отслеживанием положения головы посредством прямой интеграции с наушниками AirPods. Панорамирование осуществляется с помощью инструментов приложения DaVinci Resolve, в том числе диалогового окна, автоматизации, SpaceView и средств в окне просмотра, а также дополнительных элементов управления на панели Inspector (Инспектор). Функционал позволяет точно размещать источники в пространстве, включая автоматическое отслеживание по широтно-долготному видео. Файлы ASAF можно выгрузить в форматах WAV или MP4 (с соответствующими метаданными) с помощью опции Immersive Audio (Объемный звук) > Export Master File (Экспортировать мастер-файл), хотя в настоящее время их нельзя импортировать или воспроизвести в приложении DaVinci Resolve.

Конечный продукт

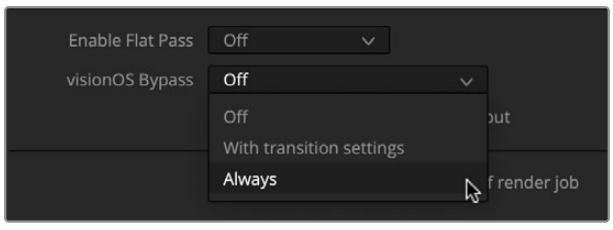
На странице экспорта два параметра рендеринга позволяют подготовить файлы Apple Vision Pro для проверки и создания окончательной версии.



- **Формат Vision Pro Review** — создает небольшой пакет MV-HEVC для предварительного просмотра. Система Apple передает полученный файл на гарнитуру Apple Vision Pro через Wi-Fi и обеспечивает его воспроизведение в приложении Apple Immersive Video Utility (AIVU), которое доступно в магазине App Store.
- **Формат Vision Pro Bundle** — создает большой пакет файлов ProRes, который можно архивировать или обрабатывать в приложении Apple Compressor, чтобы подготовить его для воспроизведения на Apple TV.
- **Формат ASAF** — воспроизводит аудиодорожку ASAF в форматах WAV или MP4.
- **Формат VR180** — создает стереоскопический видеофайл, совместимый со стандартом VR180.

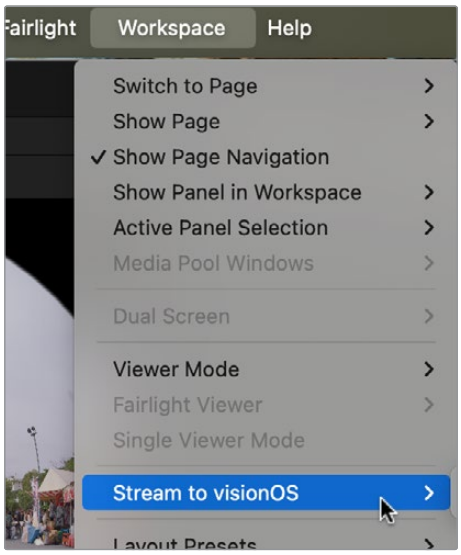
Работа с переходами

Если переход добавляется на иммерсивную временную шкалу, будут выполнены его предварительный рендеринг и преобразование в одну дорожку воспроизведения. Файл XML, сгенерированный для подобного контента, описывает процесс переключения между этими потоками (визуальным рядом, затем переходом и опять визуальным рядом). В разделе Advanced Settings (Дополнительные настройки) на панели настроек рендеринга есть опция постоянного обхода эффектов visionOS, применения их в соответствии со статусом на тайм-линии или включения по мере появления на временной шкале.



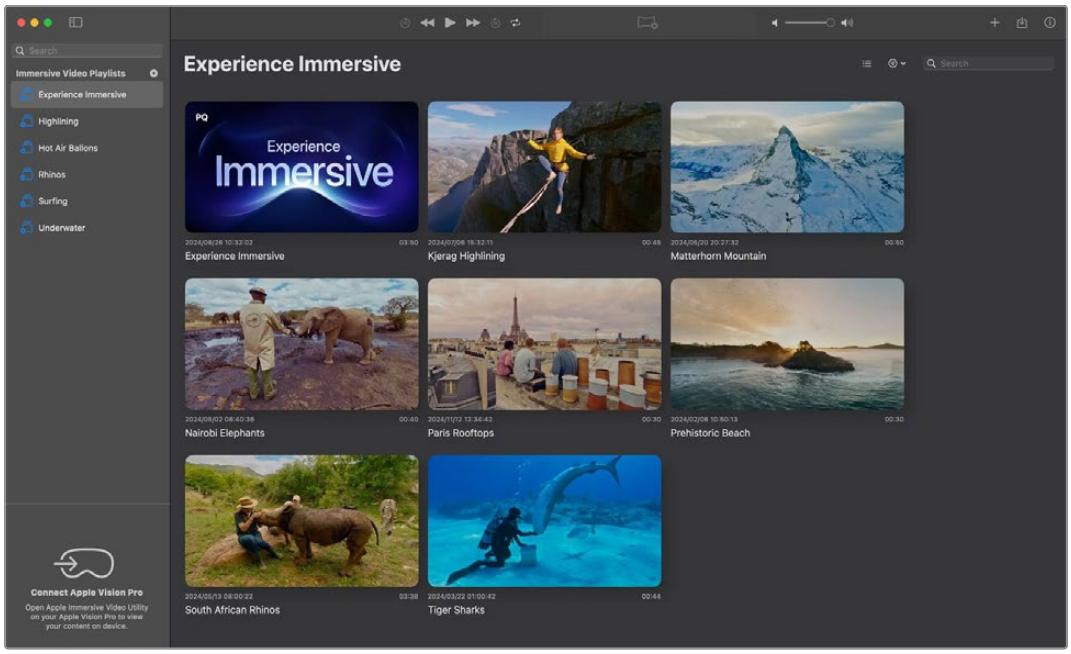
Стриминг на гарнитуру Apple Vision Pro

Стриминг на гарнитуру Apple Vision Pro можно вести со страниц монтажа, цвета и экспорта в приложении DaVinci Resolve, перейдя в меню Workspace (Рабочая область) и выбрав опцию Stream to visionOS, а затем указав устройство AVP. Для плавной трансляции рекомендуется использовать быструю беспроводную сеть (минимум Wi-Fi 5, предпочтительно Wi-Fi 6).



Утилита Apple Immersive Video

Утилита Apple Immersive Video представляет собой специализированное программное обеспечение для macOS и visionOS, предназначенное для производства, изменения параметров и воспроизведения соответствующего контента. Разработанная для профессиональных кинематографистов и создателей иммерсивного материала, она позволяет пользователям импортировать, систематизировать, архивировать и просматривать 8K-файлы стереоскопического видео с углом обзора 180 градусов и пространственным звуком. Есть поддержка файлов AIVU (Apple Immersive Video Universal), что позволяет проверять и редактировать метаданные в ходе постобработки. Пользователи могут генерировать плейлисты, сортировать информацию и осуществлять поиск в библиотеках, а также выводить поток напрямую на одно или несколько устройств Apple Vision Pro.



Developer Information

Camera Control REST API

If you are a software or hardware developer you can build custom applications or leverage ready to use tools such as REST client or Postman to seamlessly control and interact with your compatible Blackmagic camera using Camera Control REST API. This API enables you to perform a wide range of operations, such as starting or stopping recordings, accessing disk information and much more. Whether you're developing a custom application tailored to your specific needs or utilizing existing tools, this API empowers you to unlock the full potential of your Blackmagic camera with ease. We look forward to seeing what you come up with!

NOTE It's important to mention that controlling Blackmagic cameras via REST API relies on the web manager being enabled on each compatible Blackmagic camera. Enable the web media manager in the Blackmagic Camera Setup 'network access' settings for each camera you are controlling.

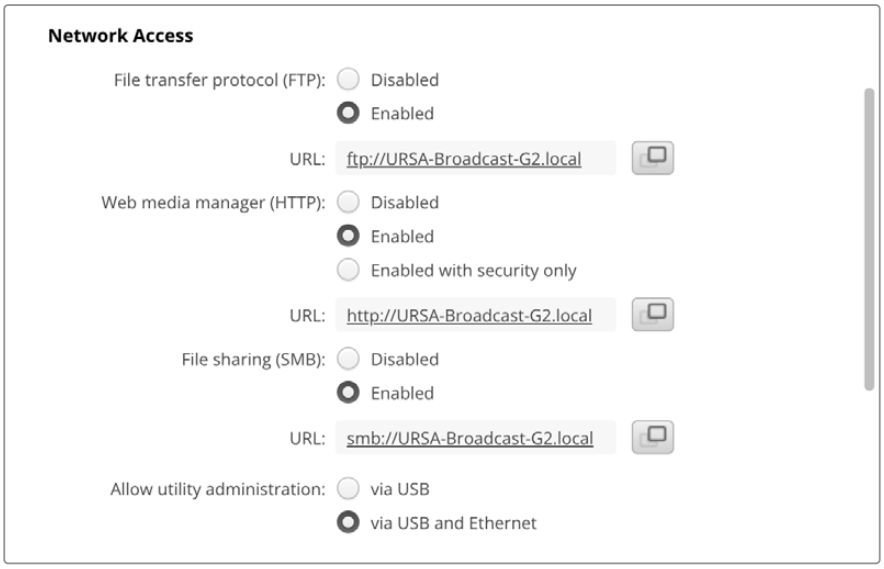
The following Blackmagic cameras are compatible with Camera Control REST API:

- Blackmagic URSA Cine 12K LF
- Blackmagic URSA Cine 17K 65
- Blackmagic URSA Cine Immersive
- Blackmagic PYXIS 6K
- Blackmagic Cinema Camera 6K
- Blackmagic URSA Broadcast G2
- Blackmagic Micro Studio Camera 4K G2
- Blackmagic Studio Camera 4K Plus
- Blackmagic Studio Camera 4K Pro
- Blackmagic Studio Camera 6K Pro
- Blackmagic Studio Camera 4K Plus G2
- Blackmagic Studio Camera 4K Pro G2

Sending API Commands

To send an API command to your camera from a third party application such as Postman, add /control/api/v1/ to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/api/v1/

You can find the Web media manager URL and IP address information in Blackmagic Camera Setup.



The Web media manager URL in Blackmagic Camera Setup

Downloading API's from your Camera

You can download REST API YAML documentation from your camera by adding /control/documentation.html to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/documentation.html

NOTE It's worth noting that changing the camera name in Blackmagic Camera Setup will also change the camera's Web media manager URL.

Livestream Control API

API for controlling Livestreams on Blackmagic Design products.

GET /livestreams/0

Get the livestream's current status.

Response

200 - Livestream's current status.

The response is JSON.

Name	Type	Description
status (required)	string	Possible values are: Idle, Connecting, Streaming, Flushing, Interrupted.
bitrate (required)	integer	Current bitrate (bps).
effectiveVideoFormat (required)	string	Effective video format for the livestream, serialised as a string.
duration	integer	Current stream duration in seconds. Absent if livestream is idle.
cache	integer	Current stream cache usage percentage.

GET /livestreams/0/start

Determine if the livestream is active.

Response

200 - Livestream active status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is active.

PUT /livestreams/0/start

Start the livestream.

Response

204 - Livestream started.

GET /livestreams/0/stop

Determine if the livestream is inactive.

Response

200 - Livestream inactive status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is inactive.

PUT /livestreams/0/stop

Stop the livestream.

Response

204 - Livestream stopped.

GET /livestreams/0/activePlatform

Get the currently selected platform configuration for the livestream.

Response

200 - Livestream active platform configuration.

The response is JSON.

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

PUT /livestreams/0/activePlatform

Set the currently selected platform configuration for the livestream.

Parameters

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

Response

204 - Livestream active platform configuration updated.

400 - Bad Request

GET /livestreams/platforms

Get the list of available platforms.

Response

200 - List of available platforms.

The response is JSON.

Name	Type	Description
	array	List of available platforms names.
[i]	string	Platform name.

GET /livestreams/platforms/{platformName}

Get the service configuration for a platform.

Parameters

Name	Type	Description
{platformName} (required)	string	Name of the platform.

Response

200 - Service configuration for specified platform.

The response is JSON.

Livestream platform service configuration.

Name	Type	Description
platform (required)	string	Corresponding platform name.
key	string	Default stream key.
servers (required)	array	List of server configurations.
servers[i]	object	Server configuration.
servers[i].server (required)	string	Server name.
servers[i].url (required)	string	Livestream destination.
servers[i].srtExtensions	array	Miscellaneous tags used for SRT livestreams.
servers[i].srtExtensions[i]	object	Dictionary object mapping SRT tag strings to values.
servers[i].srtExtensions[i][{key}]	string	SRT tag value.
servers[i].group	string	Logical grouping of the server.
profiles (required)	array	List of profile configurations.
profiles[i]	object	Quality configuration.
profiles[i].profile (required)	string	Quality level name.
profiles[i].configs (required)	array	List of video format configurations.
profiles[i].configs[i]	object	Video format configuration for profiles.
profiles[i].configs[i].resolution (required)	string	Video format serialised as a string.
profiles[i].configs[i].fps (required)	string	Frames per second.
profiles[i].configs[i].bitrate (required)	integer	Pixel bitrate (bps).
profiles[i].configs[i].audioBitrate	integer	Audio bitrate (bps).
profiles[i].configs[i].keyFrameInterval	integer	How often a key frame is sent, in seconds.
profiles[i].configs[i].videoCodecs	array	Supported video encoding algorithm/s.

Name	Type	Description
profiles[i].configs[i].videoCodecs[i]	string	Video encoding algorithm. Possible values are: H264, H265.
profiles[i].lowLatency (required)	boolean	If true, fewer frames will be buffered in the livestream.
defaultProfile	string	Quality level name.
credentials	object	Credientials used for RTMP streams.
credentials.username (required)	string	The username part of the creditials. Only used for RTMP streams.
credentials.password (required)	string	Used for RTMP streams, also used as Passphrase for SRT streams.
customizableUrlEnabled	boolean	True when the server URL is customizable.

400 - Bad Request

GET /livestreams/customPlatforms

Get a list of custom platform files.

Response

200 - List of custom platform files.

The response is JSON.

Name	Type	Description
	array	List of custom platform file names.
[i]	string	Custom platform file name.

DELETE /livestreams/customPlatforms

Remove all custom configuration files.

Response

204 - All custom configuration files removed.

GET /livestreams/customPlatforms/{filename}

Get a custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to get.

Response

200 - Custom platform file.

The response is XML.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

404 - Not Found

PUT /livestreams/customPlatforms/{filename}

Update a custom platform file if it exists, if not, create a new file with the given file name.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to update/create.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

Response

204 - Custom platform file created or updated.

400 - Bad Request

DELETE /livestreams/customPlatforms/{filename}

Remove the given custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to be removed.

Response

204 - Custom platform file removed.

404 - Not Found

Clips Control API

API for listing clips on disk.

GET /clips

Get the list of clips on the active disk.

Response

200 - List of clips on the active disk.

The response is JSON.

List of media clips.

Name	Type	Description
clips (required)	array	
clips[i]	object	Media clip.
clips[i].clipUniqueId (required)	integer	Unique ID used to identify this clip.
clips[i].filePath	string	Path to the file relative to the root of a mount.
clips[i].fileSize	integer	Size of file on disk in bytes.
clips[i].codecFormat	object	
clips[i].codecFormat.codec	string	Currently selected codec.

Name	Type	Description
clips[i].codecFormat.container	string	Multimedia container format.
clips[i].videoFormat	object	Video format configuration.
clips[i].videoFormat.name (required)	string	Video format serialised as a string.
clips[i].videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
clips[i].videoFormat.height	number	Height dimension of video format.
clips[i].videoFormat.width	number	Width dimension of video format.
clips[i].videoFormat.interlaced	boolean	Is the display format interlaced?
clips[i].startTimecode	string	Start timecode of the clip serialised as string.
clips[i].durationTimecode	string	Duration of the clip in timecode format serialised as string.
clips[i].frameCount	integer	Number of frames in clip; duration of the clip in frames.

404 - There is no active disk.

Media Pool Control API

API to manage media pool and handle uploads and project data.

GET /cloud/projects

List all projects within the media pool.

Response

200 - Successfully retrieved the list of all projects.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].libraryID	string	
[i].name	string	
[i].private	boolean	
[i].shared	boolean	
[i].clips	array	List of clips associated with the project.
[i].clips[i]	string	
[i].status	object	
[i].status.numClipsRequested	integer	
[i].status.numClipsComplete	integer	
[i].status.uploadPercent	integer	Percentage of upload completion.
[i].status.numClipsPaused	integer	
[i].status.outOfSpace	boolean	
[i].status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
[i].status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/active

Retrieve data of the actively uploading project.

Response

200 - Successfully retrieved the active project's data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/{projectID}

Retrieve specific project data by project ID.

Parameters

Name	Type	Description
{projectID} (required)	integer	Unique identifier of the project.

Response

200 - Successfully retrieved the project data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	

Name	Type	Description
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

404 - Project not found.

GET /cloud/clips

List all clips within the media pool.

Response

200 - Successfully retrieved the list of all clips.

The response is JSON.

Name	Type	Description
	array	
[i]	string	

GET /cloud/clips/activeUploading

Retrieve data of actively uploading clips.

Response

200 - Successfully retrieved the list of actively uploading clips.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].path	string	
[i].projectID	integer	
[i].status	object	
[i].status.projectID	integer	
[i].status.outOfSpace	boolean	
[i].status.proxyExtension	string	
[i].status.growingFile	boolean	
[i].status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.originalClipTotalSize	integer	
[i].status.proxyClipTotalSize	integer	
[i].status.originalClipCompletedSize	integer	
[i].status.proxyClipCompletedSize	integer	
[i].status.secsRemaining	integer	

GET /cloud/clips/{deviceName}/{path}

Retrieve specific clip data by device and path.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored.
{path} (required)	string	Path to the clip.

Response

200 - Successfully retrieved the clip data.

The response is JSON.

Name	Type	Description
path	string	
projectID	integer	
status	object	
status.projectID	integer	
status.outOfSpace	boolean	
status.proxyExtension	string	
status.growingFile	boolean	
status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.originalClipTotalSize	integer	
status.proxyClipTotalSize	integer	
status.originalClipCompletedSize	integer	
status.proxyClipCompletedSize	integer	
status.secsRemaining	integer	

404 - Clip not found.

Monitoring Control API

API for monitoring and controlling display settings in video equipment.

GET /monitoring/display

Retrieve a list of all display names.

Response

200 - Returns a list of display names.

The response is JSON.

Name	Type	Description
displays	array	List of display names available.
displays[i]	string	

GET /monitoring/{displayName}/cleanFeed

Get the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/cleanFeed

Set the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Clean feed enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/displayLUT

Get the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

400 - Invalid display name.

404 - Display name not found.

PUT /monitoring/{displayName}/displayLUT

Set the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Display LUT enabled/disabled successfully.

400 - Invalid input.

422 - Unprocessable Entity - Unable to process the contained instructions.

GET /monitoring/{displayName}/zebra

Get the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/zebra

Set the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Zebra enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/focusAssist

Get the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/focusAssist

Set the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/focusAssist

Get the focus assist settings.

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

404 - Display name not found.

PUT /monitoring/focusAssist

Set the focus assist settings.

Parameters

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/frameGuide

Get the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame guide enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGuide

Set the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame guide state updated successfully.

422 - Unable to update the frame guide state.

GET /monitoring/frameGuideRatio

Get the current frame guide ratio.

Response

200 - Returns the current frame guide ratio.

The response is JSON.

Name	Type	Description
ratio	string	The frame guide ratio.

PUT /monitoring/frameGuideRatio

Set the frame guide ratio.

Parameters

Name	Type	Description
ratio	string	The frame guide ratio.

Response

204 - Frame guide ratio updated successfully.

422 - Unable to update the frame guide ratio.

GET /monitoring/frameGuideRatio/presets

Get the presets for frame guide ratios.

Response

200 - Returns a list of preset frame guide ratios.

The response is JSON.

Name	Type	Description
presets	array	
presets[i]	string	A frame guide ratio.

GET /monitoring/{displayName}/frameGrids

Get the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame grids enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGrids

Set the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame grids state updated successfully.

422 - Unable to update the frame grids state.

GET /monitoring/frameGrids

Get the global frame grids settings.

Response

200 - Returns the current frame grids settings.

The response is JSON.

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

PUT /monitoring/frameGrids

Set the global frame grids settings.

Parameters

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

Response

204 - Frame grids settings updated successfully.

400 - Invalid input, check the number of frame grids or values.

422 - Unable to update the frame grids settings.

GET /monitoring/{displayName}/safeArea

Get the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the safe area enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/safeArea

Set the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Safe area state updated successfully.

422 - Unable to update the safe area state.

GET /monitoring/safeAreaPercent

Get the current safe area percentage.

Response

200 - Returns the current safe area percentage.

The response is JSON.

Name	Type	Description
percent	integer	Safe area coverage percentage.

PUT /monitoring/safeAreaPercent

Set the safe area percentage.

Parameters

Name	Type	Description
percent	integer	Safe area coverage percentage to set.

Response

204 - Safe area percentage updated successfully.

400 - Invalid percentage value.

422 - Unable to update the safe area percentage.

GET /monitoring/{displayName}/falseColor

Get the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the false color enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/falseColor

Set the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - False color state updated successfully.

422 - Unable to update the false color state.

Event Control API

API For working with built-in websocket.

GET /event/list

Get the list of events that can be subscribed to using the websocket API.

Response

200 - Websocket API events list.

The response is JSON.

Name	Type	Description
events	array	List of events that can be subscribed to using the websocket API.
events[i]	string	

System Control API

API for controlling the System Modes on Blackmagic Design products.

GET /system

Get device system information.

Response

200 - System summary.

The response is JSON.

The properties will be populated only with the values that are supported/implemented by the device in use.

Name	Type	Description
codecFormat	object	Codec format configuration.
codecFormat.codec	string	Codec serialised as string.
codecFormat.container	string	Multimedia container format.
videoFormat	object	Video format configuration.
videoFormat.name (required)	string	Video format serialised as a string.
videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
videoFormat.height	number	Height dimension of video format.
videoFormat.width	number	Width dimension of video format.
videoFormat.interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/product

Get device product information.

Response

200 - Device product information.

The response is JSON.

Product information.

Name	Type	Description
deviceName	string	Name of device as displayed in Setup.
productName	string	Device's product name.
softwareVersion	string	Software version running on device.

501 - This functionality is not implemented for the device in use.

GET /system/supportedCodecFormats

Get the list of supported codecs.

Response

200 - List of supported codec formats.

The response is JSON.

Name	Type	Description
codecs	array	
codecs[i]	object	Codec format configuration.
codecs[i].codec	string	Codec serialised as string.
codecs[i].container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

GET /system/codecFormat

Get the currently selected codec.

Response

200 - Current codec format.

The response is JSON.

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

PUT /system/codecFormat

Update the system codec.

Parameters

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

Response

204 - The codec updated successfully.

400 - The specified codec format is unsupported.

501 - This functionality is not implemented for the device in use.

GET /system/videoFormat

Get the currently selected video format.

Response

200 - Current system video format.

The response is JSON.

Video format configuration.

Name	Type	Description
name (required)	string	Video format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height	number	Height dimension of video format.
width	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

PUT /system/videoFormat

Set the system video format.

Parameters

This parameter can be one of the following types:

Name	Type	Description
name (required)	string	Video format serialised as a string.

Name	Type	Description
frameRate (required)	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height (required)	number	Height dimension of video format.
width (required)	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

Response

204 - The video format updated successfully.

400 - Invalid request.

409 - Operation unsupported in the current state.

501 - This functionality is not implemented for the device in use.

GET /system/supportedVideoFormats

Get the list of supported video formats for the current system state.

Response

200 - List of supported video formats.

The response is JSON.

List of supported video formats.

Name	Type	Description
formats	array	List of video formats.
formats[i]	object	Video format configuration.
formats[i].name (required)	string	Video format serialised as a string.
formats[i].frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
formats[i].height	number	Height dimension of video format.
formats[i].width	number	Width dimension of video format.
formats[i].interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/supportedFormats

Get supported formats.

Response

200 - List of supported formats.

The response is JSON.

Name	Type	Description
supportedFormats	array	
supportedFormats[i]	object	
supportedFormats[i].codecs	array	
supportedFormats[i].codecs[i]	string	
supportedFormats[i].frameRates	array	
supportedFormats[i].frameRates[i]	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
supportedFormats[i]. maxOffSpeedFrameRate	number	
supportedFormats[i]. minOffSpeedFrameRate	number	
supportedFormats[i]. recordResolution	object	
supportedFormats[i]. recordResolution.height	number	Height of the resolution.
supportedFormats[i]. recordResolution.width	number	Width of the resolution.
supportedFormats[i]. sensorResolution	object	
supportedFormats[i]. sensorResolution.height	number	Height of the resolution.
supportedFormats[i]. sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

GET /system/format

Get current format.

Response

200 - Current format.

The response is JSON.

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

PUT /system/format

Set the format.

Parameters

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

Response

204 - System format updated.

501 - This functionality is not implemented for the device in use.

Transport Control API

API for controlling Transport on Blackmagic Design products.

GET /transports/0

Get device's basic transport status.

Response

200 - Transport status.

The response is JSON.

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

PUT /transports/0

Set device's basic transport status.

Parameters

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, Output.

Response

204 - Transport mode was set.

400 - Failed to set transport mode.

GET /transports/0/stop

Determine if transport is stopped.

Response

200 - Transport stop response.

The response is JSON.

Name	Type	Description
	boolean	True when transport mode is InputPreview or when in Output mode and speed is 0.

PUT /transports/0/stop

Stop transport. Deprecated, use POST /transports/0/stop instead.

Response

204 - Transport stopped.

POST /transports/0/stop

Stop transport.

Response

204 - Transport stopped.

GET /transports/0/play

Determine if transport is playing.

Response

200 - Transport play response.

The response is JSON.

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

PUT /transports/0/play

Start playing on transport. Deprecated, use POST /transports/0/play instead.

Response

204 - Transport playing.

400 - Failed to set transport to play.

POST /transports/0/play

Start playing on transport.

Response

204 - Transport playing.

400 - Failed to set transport to play.

GET /transports/0/playback

Get playback state.

Response

200 - Transport playback state.

The response is JSON.

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

PUT /transports/0/playback

Set playback state.

Parameters

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

Response

204 - Updated transport playback state.

400 - Failed to set transport playback state.

GET /transports/0/record

Get record state.

Response

200 - Recording state.

The response is JSON.

Name	Type	Description
recording	boolean	If true, transport is in InputRecord mode.

PUT /transports/0/record

Set record state. Deprecated, use POST /transports/0/record instead.

Parameters

Name	Type	Description
recording (required)	boolean	If true, starts a recording, otherwise stops.
clipName	string	Optional, sets the requested clip name to record to, when “recording” attribute is set to true.

Response

204 - Recording state updated.

400 - Failed to update recording state.

POST /transports/0/record

Start recording.

Parameters

Name	Type	Description
clipName	string	Optional, provides a specific name of clip to record to.

Response

204 - Recording started.

400 - Failed to start recording.

GET /transports/0/clipIndex

Get the clip index of the currently playing clip on the timeline.

Response

200 - Clip index response.

The response is JSON.

Name	Type	Description
clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

GET /transports/0/timecode

Get device timecode.

Response

200 - Timecode response.

The response is JSON.

Name	Type	Description
display	string	The display timecode serialised as a string.
timeline	string	The timeline timecode serialised as a string.

GET /transports/0/timecode/source

Get timecode source selected on device.

Response

200 - Timecode source response.

The response is JSON.

Name	Type	Description
timecode	string	Possible values are: Timeline, Clip.

Timeline Control API

API for controlling playback timeline.

GET /timelines/0

Get the playback timeline.

Response

200 - Playback timeline.

The response is JSON.

Name	Type	Description
clips	array	
clips[i]	object	Timeline clip.
clips[i].clipUniqueId (required)	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).
clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

404 - No timeline / disk available.

DELETE /timelines/0

Clear the current playback timeline. Deprecated, prefer to use POST /timelines/0/clear

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

POST /timelines/0

Add a clip to the timeline.

Parameters

This parameter can be one of the following types:

Add multiple media clips to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips[i].clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips[i].clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips[i].frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add multiple media clips to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add a single clip to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips.clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips.clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips.frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add a single clip to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the timeline as specified.

501 - The operation is not supported on this device.

POST /timelines/0/add

Add a clip to the end of the timeline. Deprecated, use POST /timelines/0 to add clips within the timeline.

Parameters

This parameter can be one of the following types:

Add one clip to the end of the timeline.

Name	Type	Description
clips	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add many clips to the end of the timeline.

Name	Type	Description
clips	array	List of clipUniqueIds of clips to add to end of timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the end of the timeline.

501 - The operation is not supported on this device.

POST /timelines/0/clear

Clear the playback timeline.

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

DELETE /timelines/0/clips/{timelineClipIndex}

Remove the specified clip from the timeline.

Parameters

Name	Type	Description
{timelineClipIndex} (required)	integer	The (0-based) timeline clip index of the clip to remove from the timeline.

Response

204 - The specified clip was removed from the timeline.

501 - The operation is not supported on this device.

Media Control API

API for controlling media devices in Blackmagic Design products.

GET /media/workingset

Get the list of media devices currently in the working set.

Response

200 - The list of media devices in the working set.

The response is JSON.

Name	Type	Description
size	integer	The fixed size of this device's working set.
workingset	array	The device's working set.
workingset[i]	object | null	Device within the working set. null if no device is present within the given working set slot.
workingset[i].volume	string	Volume name.
workingset[i].deviceName	string	Internal device name of this media device.
workingset[i].remainingRecordTime	integer	Remaining record time using current codec and video format in seconds.
workingset[i].totalSpace	integer	Total space on media device in bytes.
workingset[i].remainingSpace	integer	Remaining space on media device in bytes.
workingset[i].clipCount	integer	Number of clips currently on the device.

GET /media/active

Get the currently active media device.

Response

200 - The current active media device.

The response is JSON.

The active media device, or null if there is no active media.

Name	Type	Description
workingsetIndex	integer	Working set index of the active media device.
deviceName	string	Device name of media device.

204 - No media is currently active.

PUT /media/active

Set the currently active media device.

Parameters

Name	Type	Description
workingsetIndex	integer	Working set index of the media to make active.

Response

204 - The active media device was set successfully.

400 - Setting the currently active media device is not possible in the current state.

GET /media/devices/doformatSupportedFilesystems

Get the list of filesystems available to format a media device.

Response

200 - The list of filesystems permitted for formatting.

The response is JSON.

Name	Type	Description
	array	List of filesystems permitted for formatting media.
[i]	string	Filesystem serialised as string.

GET /media/devices/{deviceName}

Get information about a requested device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Information about the requested device.

The response is JSON.

Media device state.

Name	Type	Description
state	string	The current state of the media device. Possible values are: None, Scanning, Mounted, Uninitialised, Formatting, RaidComponent.

400 - Invalid device name.

404 - Device not found.

GET /media/devices/{deviceName}/doformat

Get a format key, used to format the device with a PUT request.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Format prepared.

The response is JSON.

Name	Type	Description
deviceName	string	Device name of media device to format.
key	string	The key required to format this device, provide to PUT /media/devices/{deviceName}/doformat to perform format of media device.

400 - Cannot format the device.

404 - Device not found.

PUT /media/devices/{deviceName}/doformat

Perform a format of the specified media device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Name	Type	Description
key	string	The key used to format this device, retrieved from prepare format media request GET /media/devices/{deviceName}/doformat. Format key provided cannot be reused after successful format.
filesystem	string	Filesystem to format to. Supported filesystems can be retrieved with GET /media/devices/doFormatSupportedFilesystems.
volume	string	Volume name to set for the disk after format.

Response

204 - Format successful.

400 - Cannot format the device, invalid filesystem or key.

404 - Device not found.

Slate Control API

API to manage digital slate data.

GET /slates/nextClip

Retrieve the digital slate for the next clip.

Response

200 - Returns the slate data for the next clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	

Name	Type	Description
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Slate data is not available.

PUT /slates/nextClip

Update the slate data for the next clip.

Parameters

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

POST /slates/nextClip/resetProjectData

Reset the project data for the next clip's slate.

Response

200 - Project data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetProjectData

Reset the project data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Project data reset successfully.

POST /slates/nextClip/resetLensData

Reset the lens data for the next clip's slate.

Response

200 - Lens data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetLensData

Reset the lens data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Lens data reset successfully.

GET /slates/clips/{deviceName}/{path}

Retrieve slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Returns the slate data for the specified clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

404 - Clip not found.

PUT /slates/clips/{deviceName}/{path}

Update the slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.

Name	Type	Description
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

Preset Control API

API For controlling the presets on Blackmagic Design products

GET /presets

Get the list of the presets on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presets	array	List of the presets on the camera (.cset files)
presets[i]	string	

POST /presets

Send a preset file to the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presetAdded	string	Name of the preset uploaded (without .cset extension)

400 - Bad request - missing Content-Disposition header or filename

GET /presets/active

Get the currently active preset on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

PUT /presets/active

Set the active preset on the camera

Parameters

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

Response

204 - No Content

404 - Preset file not found

PARAMETERS /presets/{presetName}

GET /presets/{presetName}

Download the preset file

Response

200 - OK

The response is a binary file.

404 - File does not exist

PUT /presets/{presetName}

Save current camera state as a preset

Response

204 - No Content

DELETE /presets/{presetName}

Delete a preset from the camera

Response

204 - No Content

404 - Preset file not found

Audio Control API

API For controlling audio on Blackmagic Design Cameras

GET /audio/channels

Get the total number of audio channels available

Response

200 - Returns the total number of channels

The response is JSON.

Name	Type	Description
channels	integer	Total number of audio channels available

GET /audio/supportedInputs

Get the list of supported audio inputs

Response

200 - List of all supported audio inputs

The response is JSON.

Name	Type	Description
	array	
[i]	string	A supported audio input

GET /audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently selected input

The response is JSON.

Name	Type	Description
input	string	Audio input source and type

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/input

Set the audio input for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
input	string	Audio input source and type

Response

204 - No Content

400 - Invalid audio input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/input/description

Get the description of the current input of the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Description of the current input of the selected channel

The response is JSON.

Name	Type	Description
description	object	
description.gainRange	object	
description.gainRange.Min	number	The minimum gain value in dB
description.gainRange.Max	number	The maximum gain value in dB
description.capabilities	object	
description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter
description.capabilities.Padding	object	
description.capabilities.Padding. available	boolean	Input supports setting of padding
description.capabilities.Padding. forced	boolean	Padding is forced to be set for the input
description.capabilities.Padding. value	number	Value of the padding in dB

404 - Channel does not exist

GET /audio/channel/{channelIndex}/supportedInputs

Get the list of supported inputs and their availability to switch to for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its supported inputs are being queried. (Channels index from 0)

Response

200 - The list of supported inputs

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

404 - Channel does not exist

GET /audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set level for the selected channel

The response is JSON.

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/level

Set the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

Response

204 - No Content

400 - Invalid input or value out of range

404 - Channel does not exist

GET /audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set phantom power for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Phantom power enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/phantomPower

Set the audio phantom power for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Phantom power enabled state

Response

204 - No Content

400 - Phantom power is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set padding for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Padding enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/padding

Set the audio input padding for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Padding enabled state

Response

204 - No Content

400 - Padding is not supported or is forced for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set low cut filter for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Low cut filter enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/lowCutFilter

Set the audio input low cut filter for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Low cut filter enabled state

Response

204 - No Content

400 - Low cut filter is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set availability for the selected channel

The response is JSON.

Name	Type	Description
available	boolean	Whether the input is currently available

404 - Channel does not exist

Lens Control API

API For controlling the lens on Blackmagic Design products

GET /lens/iris

Get lens' aperture

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number

PUT /lens/iris

Set lens' aperture

Parameters

Name	Type	Description
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number
adjustmentStep	integer	Signed value for relative aperture adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens iris is not controllable or is controlled by auto exposure

GET /lens/zoom

Get lens' zoom

Response

200 - OK

The response is JSON.

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value

PUT /lens/zoom

Set lens' zoom

Parameters

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value
adjustmentFocalLength	integer	Signed value for relative focal length adjustment
adjustmentNormalised	number	Signed normalized value for relative zoom adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens zoom is not controllable

GET /lens/focus

Get lens' focus

Response

200 - OK

The response is JSON.

Name	Type	Description
normalised	number	Normalised value

PUT /lens/focus

Set lens' focus

Parameters

Name	Type	Description
normalised	number	Normalised value
focusDistance	integer	Focus distance value

Response

204 - No Content

400 - Bad Request if out of range value is provided

PUT /lens/focus/doAutoFocus

Perform auto focus

Parameters

Name	Type	Description
position	object	
position.x	number	Normalized x coordinate for autofocus ROI
position.y	number	Normalized y coordinate for autofocus ROI

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens focus is not controllable

GET /lens/opticalImageStabilization

Get optical image stabilization status

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Whether optical image stabilization is enabled

501 - Not Implemented if optical image stabilization is not supported on this product

PUT /lens/opticalImageStabilization

Enable or disable optical image stabilization

Parameters

Name	Type	Description
enabled	boolean	Enable or disable optical image stabilization

Response

204 - No Content

501 - Not Implemented if optical image stabilization is not supported on this product

GET /lens/iris/description

Get detailed description of lens' iris capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the iris can be controlled
apertureStop	object	
apertureStop.min	number	Minimum aperture stop
apertureStop.max	number	Maximum aperture stop

GET /lens/zoom/description

Get detailed description of lens' zoom capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the zoom can be controlled
focalLength	object	
focalLength.adjustable	boolean	If focal length is adjustable
focalLength.min	integer	Minimum focal length
focalLength.max	integer	Maximum focal length

GET /lens/focus/description

Get detailed description of lens' focus capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the focus can be controlled
focusDistance	object	
focusDistance.adjustable	boolean	If focus distance is adjustable
focusDistance.min	number	Minimum focus distance
focusDistance.max	number	Maximum focus distance

Video Control API

API For controlling the video on Blackmagic Design products

GET /video/iso

Get current ISO

Response

200 - OK

The response is JSON.

Name	Type	Description
iso	integer	Current ISO value

PUT /video/iso

Set current ISO

Parameters

Name	Type	Description
iso	integer	ISO value to set

Response

204 - No Content

403 - ISO cannot be changed in the current state

GET /video/supportedISOs

Get the list of supported ISO settings

Response

200 - List of supported ISO values

The response is JSON.

Name	Type	Description
supportedISOs	array	Array of supported ISO values
supportedISOs[i]	integer	

GET /video/gain

Get current gain value in decibels

Response

200 - OK

The response is JSON.

Name	Type	Description
gain	integer	Current gain value in decibels

PUT /video/gain

Set current gain value

Parameters

Name	Type	Description
gain	integer	Gain value in decibels to set

Response

204 - No Content

403 - Gain cannot be changed in the current state

GET /video/supportedGains

Get the list of supported gain settings in decibels

Response

200 - List of supported gain values in decibels

The response is JSON.

Name	Type	Description
supportedGains	array	Array of supported gain values in decibels
supportedGains[i]	integer	

GET /video/whiteBalance

Get current white balance

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	integer	Current white balance

PUT /video/whiteBalance

Set current white balance

Parameters

Name	Type	Description
whiteBalance	integer	White balance to set

Response

204 - No Content

400 - Invalid white balance temperature

GET /video/whiteBalance/description

Get white balance range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	object	
whiteBalance.min	integer	Minimum color temperature
whiteBalance.max	integer	Maximum color temperature

PUT /video/whiteBalance/doAuto

Set current white balance automatically

Response

204 - No Content

GET /video/whiteBalanceTint

Get white balance tint

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	integer	Current white balance tint

PUT /video/whiteBalanceTint

Set white balance tint

Parameters

Name	Type	Description
whiteBalanceTint	integer	White balance tint to set

Response

204 - No Content

400 - Invalid white balance tint

GET /video/whiteBalanceTint/description

Get white balance tint range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	object	
whiteBalanceTint.min	integer	Minimum white balance tint
whiteBalanceTint.max	integer	Maximum white balance tint

GET /video/ndFilter

Get ND filter stop

Response

200 - OK

The response is JSON.

Name	Type	Description
stop	number	Current filter power (fStop)

501 - Not implemented for this device

PUT /video/ndFilter

Set ND filter stop

Parameters

Name	Type	Description
stop	number	Filter power (fStop) to set

Response

204 - No Content

400 - Invalid ND filter stop

501 - Not implemented for this device

GET /video/supportedNDFilters

Get the list of available ND filter stops

Response

200 - List of available ND filter stops

The response is JSON.

Name	Type	Description
supportedStops	array	Array of available ND filter stops
supportedStops[i]	number	

501 - Not implemented for this device

GET /video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

Response

200 - List of supported display modes

The response is JSON.

Name	Type	Description
supportedDisplayModes	array	Array of supported display modes
supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

GET /video/ndFilter/displayMode

Get ND filter display mode on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

PUT /video/ndFilter/displayMode

Set ND filter display mode on the camera

Parameters

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

Response

204 - No Content

400 - Invalid display mode for ND filter

501 - Not implemented for this device

GET /video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

Response

200 - Indicates if ND filter is selectable

The response is JSON.

Name	Type	Description
selectable	boolean	True if ND filter adjustments are selectable via a slider

501 - Not implemented for this device

GET /video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

PUT /video/shutter

Set current shutter

Parameters

Name	Type	Description
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

Response

204 - No Content

GET /video/shutter/measurement

Get the current shutter measurement mode

Response

200 - OK

The response is JSON.

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

PUT /video/shutter/measurement

Set the shutter measurement mode

Parameters

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

Response

204 - No Content

400 - Invalid shutter measurement

GET /video/supportedShutters

Get supported shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of supported shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of supported shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of flicker-free shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of flicker-free shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/autoExposure

Get current auto exposure mode

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

PUT /video/autoExposure

Set auto exposure

Parameters

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

Response

204 - No Content

400 - Failed to set auto exposure mode

GET /video/detailSharpening

Get the current state of detail sharpening

Response

200 - Current detail sharpening state

The response is JSON.

Name	Type	Description
enabled	boolean	Whether detail sharpening is enabled

501 - Not implemented for this device

PUT /video/detailSharpening

Enable or disable detail sharpening

Parameters

Name	Type	Description
enabled	boolean	Enable or disable detail sharpening

Response

204 - Detail sharpening state updated

501 - Not implemented for this device

GET /video/detailSharpeningLevel

Get the current detail sharpening level

Response

200 - Current detail sharpening level

The response is JSON.

Name	Type	Description
level	string	Current detail sharpening level Possible values are: Low, Medium, High.

501 - Not implemented for this device

PUT /video/detailSharpeningLevel

Set the detail sharpening level

Parameters

Name	Type	Description
level	string	Desired level of detail sharpening Possible values are: Low, Medium, High.

Response

204 - Detail sharpening level updated

400 - Invalid detail sharpening level

501 - Not implemented for this device

Camera Control API

API For controlling the Camera specific features on Blackmagic Design products

GET /camera/colorBars

Get the status of color bars display

Response

200 - Returns the current status of color bars

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the color bars are currently enabled

PUT /camera/colorBars

Set the status of color bars display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the color bars

Response

204 - Color bars status updated successfully

GET /camera/programFeedDisplay

Get the status of program feed display

Response

200 - Returns the current status of program feed display

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the program feed display is currently enabled

PUT /camera/programFeedDisplay

Set the status of program feed display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the program feed display

Response

204 - Program feed display status updated successfully

GET /camera/tallyStatus

Get the tally status of the camera

Response

200 - Returns the current tally status of the camera

The response is JSON.

Name	Type	Description
status	string	Current tally status of the camera Possible values are: None, Preview, Program.

GET /camera/power

Get the power status of the camera

Response

200 - Returns the current power status

The response is JSON.

Name	Type	Description
source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
batteries	array	List of batteries currently connected to the camera
batteries[i]	object	
batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
batteries[i].chargeRemainingPercent	integer	Remaining battery charge percentage
batteries[i].statusFlags	array	List of battery status flags
batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

GET /camera/power/displayMode

Get the power display mode of the camera

Response

200 - Returns the current power display mode

The response is JSON.

Name	Type	Description
mode	string	Current power display mode Possible values are: Percentage, Voltage.

PUT /camera/power/displayMode

Set the power display mode of the camera

Parameters

Name	Type	Description
mode	string	Power display mode to set Possible values are: Percentage, Voltage.

Response

204 - Power display mode updated successfully

400 - Invalid power display mode

GET /camera/timingReferenceLock

Get the timing reference lock status

Response

200 - Returns the timing reference lock status

The response is JSON.

Name	Type	Description
locked	boolean	Indicates if timing reference is locked

Immersive Control API

API for controlling immersive camera settings on Blackmagic Design cameras

GET /immersive/display/{displayName}/eye

Get the current eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to query (from /monitoring/display endpoint)

Response

200 - OK

The response is JSON.

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

400 - Invalid display name format

404 - Display not found

422 - Failed to get eye view

PUT /immersive/display/{displayName}/eye

Set the eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to control (from /monitoring/display endpoint)

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

Response

204 - No Content

400 - Invalid input or display name format

404 - Display not found

422 - Failed to set eye view

Color Correction Control API

API For controlling the color correction on Blackmagic Design products based on DaVinci Resolve Color Corrector

GET /colorCorrection/lift

Get color correction lift

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

PUT /colorCorrection/lift

Set color correction lift

Parameters

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gamma

Get color correction gamma

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

PUT /colorCorrection/gamma

Set color correction gamma

Parameters

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gain

Get color correction gain

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

PUT /colorCorrection/gain

Set color correction gain

Parameters

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/offset

Get color correction offset

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

PUT /colorCorrection/offset

Set color correction offset

Parameters

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/contrast

Get color correction contrast

Response

200 - OK

The response is JSON.

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/contrast

Set color correction contrast

Parameters

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/color

Get color correction color properties

Response

200 - OK

The response is JSON.

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/color

Set color correction color properties

Parameters

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/lumaContribution

Get color correction luma contribution

Response

200 - OK

The response is JSON.

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

PUT /colorCorrection/lumaContribution

Set color correction luma contribution

Parameters

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

Response

204 - No Content

Notification websocket - 1.0.0

Service that notifies subscribers of device state changes.

messages

Subscribe (The messages from the server/device)

Websocket Opened Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: websocketOpened.
.type	string	Possible values are: event.

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*

Name	Type	Description
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Event Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: propertyValueChanged.

Name	Type	Description
.data.property	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.value	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.type	string	Possible values are: event.

Publish (The messages that user can send to the server/device)

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	

Name	Type	Description
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	

Name	Type	Description
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Device Properties

/media/workingset

The value JSON returned via the eventResponse when the `/media/workingset` property changes on the device:

Name	Type	Description
.size	integer	The fixed size of this device's working set.
.workingset	array	Array of devices within the working set. null if no device is present within the given working set slot.
.workingset[i]		

/media/active

The value JSON returned via the eventResponse when the `/media/active` property changes on the device:

Name	Type	Description
.workingsetIndex	integer	Working set index of the active media device.
.deviceName	string	Internal device name of this media device.

/system

The value JSON returned via the eventResponse when the `/system` property changes on the device:

Name	Type	Description
.codecFormat	object	Codec format configuration.
.codecFormat.codec	string	Codec format serialised as a string.
.codecFormat.container	string	Multimedia container format.
.videoFormat	object	Currently selected video format.
.videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.videoFormat.height	number	Height dimension of video format.
.videoFormat.width	number	Width dimension of video format.
.videoFormat.interlaced	boolean	Is the display format interlaced?.
.videoFormat.name	string	Video format serialised as a string.

/system/codecFormat

Codec format configuration.

The value JSON returned via the eventResponse when the `/system/codecFormat` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.container	string	Multimedia container format.

/system/videoFormat

Currently selected video format.

The value JSON returned via the eventResponse when the `/system/videoFormat` property changes on the device:

Name	Type	Description
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.height	number	Height dimension of video format.
.width	number	Width dimension of video format.
.interlaced	boolean	Is the display format interlaced?.
.name	string	Video format serialised as a string.

/system/format

The value JSON returned via the eventResponse when the `/system/format` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.maxOffSpeedFrameRate	number	
.minOffSpeedFrameRate	number	
.offSpeedEnabled	boolean	
.offSpeedFrameRate	number	
.recordResolution	object	
.recordResolution.height	number	Height of the resolution.
.recordResolution.width	number	Width of the resolution.
.sensorResolution	object	
.sensorResolution.height	number	Height of the resolution.
.sensorResolution.width	number	Width of the resolution.

/system/supportedFormats

The value JSON returned via the eventResponse when the `/system/supportedFormats` property changes on the device:

Name	Type	Description
.supportedFormats	array	
.supportedFormats[i]	object	
.supportedFormats[i].codecs	array	
.supportedFormats[i].codecs[i]	string	
.supportedFormats[i].frameRates	array	
.supportedFormats[i].frameRates[i]	string	Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.supportedFormats[i].maxOffSpeedFrameRate	number	
.supportedFormats[i].minOffSpeedFrameRate	number	
.supportedFormats[i].recordResolution	object	
.supportedFormats[i].recordResolution.height	number	Height of the resolution.
.supportedFormats[i].recordResolution.width	number	Width of the resolution.
.supportedFormats[i].sensorResolution	object	
.supportedFormats[i].sensorResolution.height	number	Height of the resolution.
.supportedFormats[i].sensorResolution.width	number	Width of the resolution.

/timelines/0

The value JSON returned via the eventResponse when the `/timelines/0` property changes on the device:

Name	Type	Description
.clips	array	
.clips[i]	object	Timeline clip.
.clips[i].clipUniqueId	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
.clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
.clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
.clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
.clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).

Name	Type	Description
.clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
.clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

/transports/0

The value JSON returned via the eventResponse when the `/transports/0` property changes on the device:

Name	Type	Description
.mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

/transports/0/stop

true when transport mode is InputPreview or when in Output mode and speed is 0.

The value JSON returned via the eventResponse when the `/transports/0/stop` property changes on the device:

Name	Type	Description
	boolean	true when transport mode is InputPreview or when in Output mode and speed is 0.

/transports/0/play

True when transport is in Output mode and speed is non-zero.

The value JSON returned via the eventResponse when the /transports/0/play property changes on the device:

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

/transports/0/playback

The value JSON returned via the eventResponse when the `/transports/0/playback` property changes on the device:

Name	Type	Description
.type	string	Possible values are: Play, Jog, Shuttle, Var.
.loop	boolean	When true playback loops from the end of the timeline to the beginning of the timeline.
.singleClip	boolean	When true playback loops from the end of the current clip to the beginning of the current clip.
.speed	number	Playback speed, 1.0 for normal forward playback
.position	integer	Playback position on the timeline in units of video frames.

/transports/0/record

The value JSON returned via the eventResponse when the `/transports/0/record` property changes on the device:

Name	Type	Description
.recording	boolean	Is transport in Input Record mode.

/transports/0/timecode

The value JSON returned via the eventResponse when the `/transports/0/timecode` property changes on the device:

Name	Type	Description
.display	string	The display timecode serialised as a string.
.timeline	string	The timeline timecode serialised as a string.

/transports/0/timecode/source

The value JSON returned via the eventResponse when the `/transports/0/timecode/source` property changes on the device:

Name	Type	Description
.timecode	string	Possible values are: Timeline, Clip.

/transports/0/clipIndex

The value JSON returned via the eventResponse when the `/transports/0/clipIndex` property changes on the device:

Name	Type	Description
.clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

/slates/nextClip

The value JSON returned via the eventResponse when the `/slates/nextClip` property changes on the device:

Name	Type	Description
.clip	object	
.clip.clipName	string	
.clip.reel	integer	
.clip.scene	string	
.clip.sceneLocation	string	Possible values are: Interior, Exterior.
.clip.sceneTime	string	Possible values are: Day, Night.
.clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
.clip.take	integer	
.clip.takeType	string	Possible values are: None, PU, VFX, SER.
.clip.goodTake	boolean	
.lens	object	
.lens.lensType	string	
.lens.iris	string	

Name	Type	Description
.lens.focalLength	string	
.lens.distance	string	
.lens.filter	string	
.project	object	
.project.projectName	string	
.project.director	string	
.project.camera	string	
.project.cameraOperator	string	

/monitoring/{displayName}/cleanFeed

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/cleanFeed` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/displayLUT

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/displayLUT` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/zebra

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/zebra` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/focusAssist` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGuide

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGuide` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGrids` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/safeArea

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/safeArea` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/falseColor

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/falseColor` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/focusAssist` property changes on the device:

Name	Type	Description
.mode	string	Possible values are: Peak, ColoredLines.
.color	string	Possible values are: Red, Green, Blue, White, Black.
.intensity	integer	

/monitoring/frameGuideRatio

The value JSON returned via the eventResponse when the `/monitoring/frameGuideRatio` property changes on the device:

Name	Type	Description
.ratio	string	

/monitoring/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/frameGrids` property changes on the device:

Name	Type	Description
.frameGrids	array	
.frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

/monitoring/safeAreaPercent

The value JSON returned via the eventResponse when the `/monitoring/safeAreaPercent` property changes on the device:

Name	Type	Description
.percent	integer	Safe area coverage percentage.

/audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input` property changes on the device:

Name	Type	Description
.input	string	Audio input source and type

/audio/channel/{channelIndex}/supportedInputs

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/supportedInputs` property changes on the device:

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

/audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/level` property changes on the device:

Name	Type	Description
.gain	number	Gain value in dB
.normalised	number	Normalised level value between 0.0 and 1.0

/audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/phantomPower` property changes on the device:

Name	Type	Description
.enabled	boolean	Phantom power enabled state

/audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/padding` property changes on the device:

Name	Type	Description
.enabled	boolean	Padding enabled state

/audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/lowCutFilter` property changes on the device:

Name	Type	Description
.enabled	boolean	Low cut filter enabled state

/audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/available` property changes on the device:

Name	Type	Description
.available	boolean	Whether the input is currently available

/audio/channel/{channelIndex}/input/description

Description of the current input of the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input/description` property changes on the device:

Name	Type	Description
.description	object	
.description.gainRange	object	
.description.gainRange.Min	number	The minimum gain value in dB
.description.gainRange.Max	number	The maximum gain value in dB
.description.capabilities	object	
.description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
.description.capabilities. LowCutFilter	boolean	Input supports setting of low cut filter

Name	Type	Description
.description.capabilities.Padding	object	
.description.capabilities.Padding.available	boolean	Input supports setting of padding
.description.capabilities.Padding.forced	boolean	Padding is forced to be set for the input
.description.capabilities.Padding.value	number	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.

/colorCorrection/lift

Get color correction lift

The value JSON returned via the eventResponse when the `/colorCorrection/lift` property changes on the device:

Name	Type	Description
.red	number	Red lift component
.green	number	Green lift component
.blue	number	Blue lift component
.luma	number	Luma lift component

/colorCorrection/gamma

Get color correction gamma

The value JSON returned via the eventResponse when the `/colorCorrection/gamma` property changes on the device:

Name	Type	Description
.red	number	Red gamma component
.green	number	Green gamma component
.blue	number	Blue gamma component
.luma	number	Luma gamma component

/colorCorrection/gain

Get color correction gain

The value JSON returned via the eventResponse when the `/colorCorrection/gain` property changes on the device:

Name	Type	Description
.red	number	Red gain component
.green	number	Green gain component
.blue	number	Blue gain component
.luma	number	Luma gain component

/colorCorrection/offset

Get color correction offset

The value JSON returned via the eventResponse when the `/colorCorrection/offset` property changes on the device:

Name	Type	Description
.red	number	Red offset component
.green	number	Green offset component
.blue	number	Blue offset component
.luma	number	Luma offset component

/colorCorrection/contrast

Get color correction contrast

The value JSON returned via the eventResponse when the `/colorCorrection/contrast` property changes on the device:

Name	Type	Description
.pivot	number	Contrast pivot point
.adjust	number	Contrast adjustment

/colorCorrection/color

Get color correction color properties

The value JSON returned via the eventResponse when the `/colorCorrection/color` property changes on the device:

Name	Type	Description
.hue	number	Color hue adjustment
.saturation	number	Color saturation adjustment

/colorCorrection/lumaContribution

Get color correction luma contribution

The value JSON returned via the eventResponse when the `/colorCorrection/lumaContribution` property changes on the device:

Name	Type	Description
.lumaContribution	number	Luma contribution value

/lens/iris

Get lens' aperture

The value JSON returned via the eventResponse when the `/lens/iris` property changes on the device:

Name	Type	Description
.continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
.apertureStop	number	Aperture stop value
.normalised	number	Normalised value
.apertureNumber	integer	Aperture number

/lens/iris/description

Get detailed description of lens' iris capabilities

The value JSON returned via the eventResponse when the `/lens/iris/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the iris can be controlled
.apertureStop	object	
.apertureStop.min	number	Minimum aperture stop
.apertureStop.max	number	Maximum aperture stop

/lens/focus

Get lens' focus

The value JSON returned via the eventResponse when the `/lens/focus` property changes on the device:

Name	Type	Description
.normalised	number	Normalised value

/lens/focus/description

Get detailed description of lens' focus capabilities

The value JSON returned via the eventResponse when the `/lens/focus/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the focus can be controlled
.focusDistance	object	
.focusDistance.adjustable	boolean	If focus distance is adjustable
.focusDistance.min	number	Minimum focus distance
.focusDistance.max	number	Maximum focus distance

/lens/zoom

Get lens' zoom

The value JSON returned via the eventResponse when the `/lens/zoom` property changes on the device:

Name	Type	Description
.focalLength	integer	Focal length in mm
.normalised	number	Normalised value

/lens/zoom/description

Get detailed description of lens' zoom capabilities

The value JSON returned via the eventResponse when the `/lens/zoom/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the zoom can be controlled
.focalLength	object	
.focalLength.adjustable	boolean	If focal length is adjustable
.focalLength.min	integer	Minimum focal length
.focalLength.max	integer	Maximum focal length

/presets

Get the list of the presets on the camera

The value JSON returned via the eventResponse when the `/presets` property changes on the device:

Name	Type	Description
.presets	array	List of the presets on the camera (.cset files)
.presets[i]	string	

/presets/active

Get the currently active preset on the camera

The value JSON returned via the eventResponse when the `/presets/active` property changes on the device:

Name	Type	Description
.preset	string	Name of the active preset (with .cset extension, or 'default')

/camera/colorBars

Get the status of color bars display

The value JSON returned via the eventResponse when the `/camera/colorBars` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the color bars are currently enabled

/camera/programFeedDisplay

Get the status of program feed display

The value JSON returned via the eventResponse when the `/camera/programFeedDisplay` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the program feed display is currently enabled

/camera/tallyStatus

Get the tally status of the camera

The value JSON returned via the eventResponse when the `/camera/tallyStatus` property changes on the device:

Name	Type	Description
.status	string	Current tally status of the camera Possible values are: None, Preview, Program.

/camera/power

Get the power status of the camera

The value JSON returned via the eventResponse when the `/camera/power` property changes on the device:

Name	Type	Description
.source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
.milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
.batteries	array	
.batteries[i]	object	
.batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
.batteries[i]. chargeRemainingPercent	integer	Remaining battery charge percentage
.batteries[i].statusFlags	array	List of battery status flags
.batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

/camera/power/displayMode

Get the power display mode of the camera

The value JSON returned via the eventResponse when the `/camera/power/displayMode` property changes on the device:

Name	Type	Description
.mode	string	Current power display mode Possible values are: Percentage, Voltage.

/camera/timingReferenceLock

Get the timing reference lock status

The value JSON returned via the eventResponse when the `/camera/timingReferenceLock` property changes on the device:

Name	Type	Description
.locked	boolean	Indicates if timing reference is locked

/video/iso

Get current ISO

The value JSON returned via the eventResponse when the `/video/iso` property changes on the device:

Name	Type	Description
.iso	integer	Current ISO value

/video/supportedISOs

Get the list of supported ISO settings

The value JSON returned via the eventResponse when the `/video/supportedISOs` property changes on the device:

Name	Type	Description
.supportedISOs	array	Array of supported ISO values
.supportedISOs[i]	integer	

/video/gain

Get current gain value in decibels

The value JSON returned via the eventResponse when the `/video/gain` property changes on the device:

Name	Type	Description
.gain	integer	Current gain value in decibels

/video/supportedGains

Get the list of supported gain settings in decibels

The value JSON returned via the eventResponse when the `/video/supportedGains` property changes on the device:

Name	Type	Description
.supportedGains	array	Array of supported gain values in decibels
.supportedGains[i]	integer	

/video/whiteBalance

Get current white balance

The value JSON returned via the eventResponse when the `/video/whiteBalance` property changes on the device:

Name	Type	Description
.whiteBalance	integer	Current white balance

/video/whiteBalance/description

Get white balance range

The value JSON returned via the eventResponse when the `/video/whiteBalance/description` property changes on the device:

Name	Type	Description
.whiteBalance	object	
.whiteBalance.min	integer	Minimum color temperature
.whiteBalance.max	integer	Maximum color temperature

/video/whiteBalanceTint

Get white balance tint

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint` property changes on the device:

Name	Type	Description
.whiteBalanceTint	integer	Current white balance tint

/video/whiteBalanceTint/description

Get white balance tint range

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint/description` property changes on the device:

Name	Type	Description
.whiteBalanceTint	object	
.whiteBalanceTint.min	integer	Minimum white balance tint
.whiteBalanceTint.max	integer	Maximum white balance tint

/video/ndFilter

Get ND filter stop

The value JSON returned via the eventResponse when the `/video/ndFilter` property changes on the device:

Name	Type	Description
.stop	number	Current filter power (fStop)

/video/supportedNDFilters

Get the list of available ND filter stops

The value JSON returned via the eventResponse when the `/video/supportedNDFilters` property changes on the device:

Name	Type	Description
.supportedStops	array	Array of available ND filter stops
.supportedStops[i]	number	

/video/ndFilter/displayMode

Get ND filter display mode on the camera

The value JSON returned via the eventResponse when the `/video/ndFilter/displayMode` property changes on the device:

Name	Type	Description
.displayMode	string	ND filter display mode Possible values are: Stop, Number, Fraction.

/video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

The value JSON returned via the eventResponse when the `/video/supportedNDFilterDisplayModes` property changes on the device:

Name	Type	Description
.supportedDisplayModes	array	Array of supported display modes
.supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

/video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

The value JSON returned via the eventResponse when the `/video/ndFilterSelectable` property changes on the device:

Name	Type	Description
.selectable	boolean	True if ND filter adjustments are selectable via a slider

/video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

The value JSON returned via the eventResponse when the `/video/shutter` property changes on the device:

Name	Type	Description
.continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
.shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
.shutterAngle	number	Shutter angle

/video/shutter/measurement

Get the current shutter measurement mode

The value JSON returned via the eventResponse when the `/video/shutter/measurement` property changes on the device:

Name	Type	Description
.measurement	string	Current shutter measurement mode Possible values are: ShutterAngle, ShutterSpeed.

/video/supportedShutters

Get supported shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/supportedShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of supported shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array of flicker-free shutter speeds
.shutterSpeeds[i]	integer	

/video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/flickerFreeShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of flicker-free shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array
.shutterSpeeds[i]	integer	

/video/autoExposure

Get current auto exposure mode

The value JSON returned via the eventResponse when the `/video/autoExposure` property changes on the device:

Name	Type	Description
.mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
.type	string	Comma-separated list of device types in the auto exposure stack

/video/detailSharpening

Get the current state of detail sharpening

The value JSON returned via the eventResponse when the `/video/detailSharpening` property changes on the device:

Name	Type	Description
.enabled	boolean	Whether detail sharpening is enabled

/video/detailSharpeningLevel

Get the current detail sharpening level

The value JSON returned via the eventResponse when the `/video/detailSharpeningLevel` property changes on the device:

Name	Type	Description
.level	string	Current detail sharpening level of supported shutter speeds Possible values are: Low, Medium, High.

Помощь

Самый быстрый способ получить помощь — обратиться к страницам поддержки на сайте Blackmagic Design и проверить наличие последних справочных материалов по камере.

Раздел поддержки на сайте Blackmagic Design

Последние версии руководства по эксплуатации и программного обеспечения, а также дополнительную информацию можно найти в разделе поддержки Blackmagic Design на странице www.blackmagicdesign.com/ru/support.

Обращение в службу поддержки Blackmagic Design

Если с помощью доступных справочных материалов решить проблему не удалось, воспользуйтесь формой «Отправить нам сообщение» в разделе поддержки. Там же можно найти телефон ближайшего представительства компании Blackmagic Design.

Проверка используемой версии программного обеспечения

Чтобы узнать версию утилиты Blackmagic Camera Setup, установленную на вашем компьютере, откройте окно About Blackmagic Camera Setup.

- На компьютере с операционной системой Mac откройте утилиту Blackmagic Camera Setup в папке «Приложения». В меню выберите About Blackmagic Camera Setup, чтобы узнать номер версии.
- На компьютере с операционной системой Windows откройте ПО Blackmagic Camera Setup в меню или на экране «Пуск». В меню «Помощь» выберите About Blackmagic Camera Setup, чтобы узнать номер версии.

Загрузка последних версий программного обеспечения

Узнав установленную версию утилиты Blackmagic Camera Setup, перейдите в раздел поддержки Blackmagic Design на странице www.blackmagicdesign.com/support, чтобы проверить наличие последних релизов. Рекомендуется всегда использовать актуальную версию программного обеспечения, однако обновление лучше всего выполнять после завершения текущего проекта.

Соблюдение нормативных требований



Утилизация электрооборудования и электронной аппаратуры в Европейском Союзе

Изделие содержит маркировку, в соответствии с которой его запрещается утилизировать вместе с бытовыми отходами. Непригодное для эксплуатации оборудование необходимо передать в пункт вторичной переработки. Раздельный сбор отходов и их повторное использование позволяют беречь природные ресурсы, охранять окружающую среду и защищать здоровье человека. Чтобы получить подробную информацию о порядке утилизации, обратитесь в местные муниципальные органы или к дилеру, у которого вы приобрели это изделие.



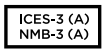
Данное оборудование протестировано по требованиям для цифровых устройств класса A (раздел 15 спецификаций FCC) и признано соответствующим всем предъявляемым критериям. Соблюдение упомянутых нормативов обеспечивает достаточную защиту от вредного излучения при работе оборудования в нежилых помещениях. Так как это изделие генерирует и излучает радиоволны, при неправильной установке оно может становиться источником радиопомех. Если оборудование эксплуатируется в жилых помещениях, высока вероятность возникновения помех, влияние которых в этом случае пользователь должен устранить самостоятельно.

До эксплуатации допускается оборудование, соответствующее двум главным требованиям.

1. Оборудование или устройство не должно быть источником вредных помех.
2. Оборудование или устройство должно быть устойчивым к помехам, включая помехи, которые могут вызвать сбой в работе.



R-R-BMD-20250213001



Соответствие требованиям ISED (Канада)

Данное оборудование соответствует канадским стандартам для цифровых устройств класса A.

Любая модификация или использование изделия не по назначению может повлечь за собой аннулирование заявления о соответствии этим стандартам.

Подключение к HDMI-интерфейсу должно выполняться с помощью качественного экранированного кабеля.

Данное оборудование протестировано по требованиям, предъявляемым к устройствам при работе в нежилых помещениях. При использовании в бытовых условиях оно может становиться источником помех для радиосигнала.

Правила безопасности

Прилагаемый блок питания 24 В постоянного тока для камеры Blackmagic URSA Cine необходимо подключить к электрической розетке, которая должна иметь заземляющий контакт.

Внутри корпуса камеры Blackmagic URSA Cine не содержатся детали, подлежащие обслуживанию. Для выполнения ремонтных работ обратитесь в местный сервисный центр Blackmagic Design.

При соединении входного разъема питания с внешним источником (DC) аккумулятор должен поддерживать режим ограничения тока или иметь предохранитель.

Используемая проводка должна иметь маркировку VW-1 либо соответствовать стандартам IEC 60332 или IEC 60695.

Мы рекомендуем, чтобы напряжение источника (DC) для камеры составляло 24 В. Если оно находится на отметке 12 В, используемая проводка должна быть рассчитана на ток силой 14 А.

В комплект поставки данного изделия входит пластина под батарею B-Mount. Мы рекомендуем использовать аккумулятор напряжением 24 В (28,8 В) или батарею с двойным напряжением.

При работе с батареей 12 В (14,4 В) обратитесь к руководству по ее эксплуатации или маркировке, чтобы определить максимальный непрерывный ток разряда. Мы рекомендуем минимальное значение 12 А.

Уведомление для жителей штата Калифорния

При работе с этим оборудованием существует возможность контакта с содержащимися в пластмассе микропримесями многобромистого бифенила, который в штате Калифорния признан канцерогеном и увеличивает риск врожденных дефектов и пороков репродуктивной системы.

Подробнее см. информацию на сайте www.P65Warnings.ca.gov.

Европейское представительство

Blackmagic Design Europe B.V.
Rijnlanderweg 766, Unit D
2,132 NM Hoofddorp
NL

Blackmagic URSA Cine

Камера и аккумулятор не должны подвергаться продолжительному воздействию прямых солнечных лучей, а литиевые батареи следует хранить вдали от источников тепла.

Выходной разъем 24 В RS подходит для подачи питания на сторонние аксессуары, такие как электроприводы фокусировки. Убедитесь, что потребляемая мощность составляет менее 48 Вт.

Гарантия

Ограниченная гарантия

Компания Blackmagic Design гарантирует отсутствие в данном изделии дефектов материала и производственного брака в течение 12 месяцев с даты продажи. Если во время гарантийного срока будут выявлены дефекты, Blackmagic Design по своему усмотрению выполнит ремонт неисправного изделия без оплаты стоимости запчастей и трудозатрат или заменит такое изделие новым.

Чтобы воспользоваться настоящей гарантией, потребитель обязан уведомить компанию Blackmagic Design о дефекте до окончания гарантийного срока и обеспечить условия для предоставления необходимых услуг. Потребитель несет ответственность за упаковку и доставку неисправного изделия в соответствующий сервисный центр Blackmagic Design с оплатой почтовых расходов. Потребитель обязан оплатить все расходы по доставке и страхованию, пошлины, налоги и иные сборы в связи с возвратом изделия вне зависимости от причины возврата.

Настоящая гарантия не распространяется на дефекты, отказы и повреждения, возникшие из-за ненадлежащего использования, неправильного ухода или обслуживания. Компания Blackmagic Design не обязана по настоящей гарантии: а) устранять повреждения, возникшие в результате действий по установке, ремонту или обслуживанию изделия лицами, которые не являются персоналом Blackmagic Design; б) устранять повреждения, возникшие в результате ненадлежащего использования или подключения к несовместимому оборудованию; в) устранять повреждения или дефекты, вызванные использованием запчастей или материалов других производителей; г) обслуживать изделие, если оно было модифицировано или интегрировано с другим оборудованием, когда такая модификация или интеграция увеличивает время или повышает сложность обслуживания изделия.

Воздействие прямых солнечных лучей может привести к повреждению экрана URSA Viewfinder, так как его оптика обладает свойствами увеличительного стекла. При длительном выводе на экран статичных или высококонтрастных объектов (например, рамок кадрирования) дисплей может выгорать или сохранять остаточное изображение. Чтобы избежать этого, не закрывайте инфракрасный датчик. Если видоискатель не используется в течение длительного времени, его следует отключить. Настоящая гарантия не распространяется на случаи возникновения остаточного изображения.

НАСТОЯЩАЯ ГАРАНТИЯ ПРЕДОСТАВЛЯЕТСЯ КОМПАНИЕЙ BLACKMAGIC DESIGN ВМЕСТО ЛЮБЫХ ДРУГИХ ПРЯМО ВЫРАЖЕННЫХ ИЛИ ПОДРАЗУМЕВАЕМЫХ ГАРАНТИЙ. КОМПАНИЯ BLACKMAGIC DESIGN И ЕЕ ДИЛЕРЫ ОТКАЗЫВАЮТСЯ ОТ ЛЮБЫХ ПОДРАЗУМЕВАЕМЫХ ГАРАНТИЙ КОММЕРЧЕСКОЙ ЦЕННОСТИ ИЛИ ПРИГОДНОСТИ ДЛЯ КАКОЙ-ЛИБО ОПРЕДЕЛЕННОЙ ЦЕЛИ. ОТВЕТСТВЕННОСТЬ BLACKMAGIC DESIGN ПО РЕМОНТУ ИЛИ ЗАМЕНЕ НЕИСПРАВНЫХ ИЗДЕЛИЙ В ТЕЧЕНИЕ ГАРАНТИЙНОГО СРОКА ОБСЛУЖИВАНИЯ ЯВЛЯЕТСЯ ПОЛНЫМ И ИСКЛЮЧИТЕЛЬНЫМ СРЕДСТВОМ ВОЗМЕЩЕНИЯ, ПРЕДОСТАВЛЯЕМЫМ ПОТРЕБИТЕЛЮ. КОМПАНИЯ BLACKMAGIC DESIGN НЕ НЕСЕТ ОТВЕТСТВЕННОСТИ ЗА КОСВЕННЫЕ, ФАКТИЧЕСКИЕ, СОПУТСТВУЮЩИЕ ИЛИ ПОСЛЕДУЮЩИЕ УБЫТКИ, ВНЕ ЗАВИСИМОСТИ ОТ ТОГО, БЫЛА ИЛИ НЕТ ОНА (ЛИБО ЕЕ ДИЛЕР) ПРЕДВАРИТЕЛЬНО ИЗВЕЩЕНА О ВОЗМОЖНОСТИ ТАКИХ УБЫТКОВ. КОМПАНИЯ BLACKMAGIC DESIGN НЕ НЕСЕТ ОТВЕТСТВЕННОСТИ ЗА ПРОТИВОПРАВНОЕ ИСПОЛЬЗОВАНИЕ ОБОРУДОВАНИЯ СО СТОРОНЫ ПОТРЕБИТЕЛЯ И ЗА УБЫТКИ, ВОЗНИКАЮЩИЕ ВСЛЕДСТВИЕ ПРИМЕНЕНИЯ ЭТОГО ИЗДЕЛИЯ. РИСКИ, СВЯЗАННЫЕ С ЕГО ЭКСПЛУАТАЦИЕЙ, ВОЗЛАГАЮТСЯ НА ПОТРЕБИТЕЛЯ.

© Copyright 2025 Blackmagic Design. Все права защищены. Товарные знаки Blackmagic Design, URSA, DeckLink, HDLink, Workgroup Videohub, Multibridge Pro, Multibridge Extreme, Intensity и "Leading the creative video revolution" зарегистрированы как товарные знаки в США и других странах. Названия других компаний и наименования продуктов могут являться товарными знаками соответствующих правообладателей.

Обозначение и логотипы Bluetooth® являются зарегистрированными товарными знаками Bluetooth SIG, Inc., а их использование со стороны компании Blackmagic Design осуществляется по лицензии. Прочие товарные знаки и торговые марки являются собственностью соответствующих владельцев.

Ottobre 2025

Guida alla camera

Blackmagicdesign

Blackmagic

URSA Cine Immersive



Blackmagic URSA Cine Immersive

Indice

Blackmagic URSA Cine Immersive 1221
Che cos'è il video immersivo? 1221
Apple Immersive Video 1221
URSA Cine Immersive 1222
Operazioni preliminari 1223
Alimentare la camera 1223
Impostare la camera per la registrazione 1224
Monitoraggio immagini 1226
Opzioni di monitoraggio e sovrimpressioni 1226
Inquadrare la prima ripresa 1227
Altezza della camera 1227
Inclinazione della camera 1228
Posizionamento 1229
Il ruolo attivo del pubblico 1230
Movimento della camera 1230
Tabelle di data rate 1231
Audio e timecode 1232
Apple Spatial Audio Format 1232
Microfoni ambisonici 1232
Microfoni lavalier e in miniatura 1233
Microfoni a mano 1233
Collegare il timecode 1233
Riprendere video immersivo 1235
Requisiti di esposizione 1235
Velocità dell'otturatore 1235
ISO e gamma dinamica 1236
Filtri ND 1237
Registrazione off speed 1238
Istogramma 1239
Opzioni di monitoraggio con il display LCD 1239
Registrazione 1240
Riproduzione 1241
Trasferire le clip 1242
Riepilogo 1243
DaVinci Resolve Studio 1244
Requisiti della workstation 1244
Configurazione del progetto 1245
Impostazioni principali 1245
Monitoraggio video 1246
Ridimensionamento 1246
Gestione del colore 1247
Informazioni su DaVinci Wide Gamut 1248
Elaborazione Apple Immersive 1249
Acquisizione 1249
Montaggio 1251
Assemblaggio 1251
Sfondi 1251
Utilizzare il visore 1253
Transizioni 1255
Apple Immersive Motion Data 1256
Titoli 1257
Considerazioni creative 1258
Compositing VFX 1259
Nodi MediaOut 1259
Renderer3D 1260
Compositing con Immersive Patcher 1261
Color grading 1262
Riduzione rumore 1262
Grading primario 1263
Grading secondario 1263
Dimensioni 1264
Palette 3D 1264
Gestione del colore CST 1266
Mappatura timeline CST 1266
Mappatura CST per monitorare il grading 1267
Monitoraggio CST con Apple Vision Pro 1269
Audio 1270
Apple Spatial Audio Format (ASAF) 1270
Consegna 1271
Bypassare le transizioni 1271
Trasmettere su Apple Vision Pro 1272
Informazioni su About Apple Immersive Video Utility 1272
Developer Information 1273
Camera Control REST API 1273

Livestream Control API 1275
Clips Control API 1279
Media Pool Control API 1280
Monitoring Control API 1283
Event Control API 1292
System Control API 1292
Transport Control API 1298
Timeline Control API 1302
Media Control API 1305
Slate Control API 1308
Preset Control API 1314
Audio Control API 1316
Lens Control API 1321
Video Control API 1325
Camera Control API 1333
Immersive Control API 1336
Color Correction Control API 1337
Notification websocket - 1.0.0 1342
messages 1342
Assistenza 1366
Normative 1367
Sicurezza 1368
Garanzia 1369

Blackmagic URSA Cine Immersive

Grazie per aver acquistato o noleggiato Blackmagic URSA Cine Immersive.

Blackmagic URSA Cine Immersive è progettata per girare contenuti di realtà virtuale in stereoscopico a 180 gradi, ovvero il video immersivo. È stata ottimizzata per filmare Apple Immersive Video per il visore Apple Vision Pro, attualmente la migliore esperienza immersiva sul mercato.

Questa guida offre una panoramica sulla camera Blackmagic URSA Cine Immersive. Ci concentreremo sulle operazioni di base per le riprese e sull'innovativo formato per il quale è stata progettata: Apple Immersive Video. Comprendere il formato ti aiuterà a capire meglio la camera.

Nelle pagine seguenti troverai svariati consigli tecnici e creativi per utilizzare al meglio la camera e per capire i benefici del formato immersivo.

Che cos'è il video immersivo?

Il video immersivo è un formato video spaziale tridimensionale. Avrai senz'altro sentito parlare di altre tecnologie simili che circolano da qualche tempo. Per capire cosa rende diverso il video immersivo, vediamo prima in cosa consistono le altre tecnologie:

- **Cinema 3D tradizionale:** cattura ogni scena con due camere con un interasse che simula la distanza tra gli occhi umani. Queste immagini stereoscopiche creano l'impressione di profondità, ma la prospettiva del pubblico è controllata dalla composizione dell'operatore di ripresa.
- **Video sferico:** combina i princìpi delle camere 3D e a 360° creando profondità e consentendo al pubblico di guardare in qualsiasi direzione. Le immagini a 360° richiedono una risoluzione altissima per far sì che la porzione dell'immagine visualizzata dallo spettatore venga riprodotta con sufficiente fedeltà. La fisiologia umana permette di girare la testa solo da spalla a spalla, dunque per guardarsi indietro il pubblico deve necessariamente stare in piedi o essere seduto su una sedia girevole.
- **Video volumetrico** oltre ad offrire profondità e movimento allo sguardo del pubblico, consente di spostarsi e di cambiare prospettiva. Anche se una configurazione ad array è in grado di catturare il video volumetrico, la maggior parte della realtà virtuale o della realtà aumentata si serve di asset 3D renderizzati in modo dinamico e dunque richiede che i partecipanti si trovino in un ambiente sicuro.

Il video immersivo combina il meglio del cinema tradizionale e dei formati 3D interattivi. Nella sua forma più semplice, è un formato stereoscopico a 180° ad alta risoluzione e frame rate elevato.

Apple Immersive Video

Blackmagic URSA Cine Immersive è progettata appositamente per la cattura in Apple Immersive Video. Questo formato si utilizza con il visore Apple Vision Pro ed è compatibile con il video stereoscopico 3D ad alta definizione in un campo visivo di 180°.

Le caratteristiche di Apple Immersive Video sono:

- Riproduzione a 90 fps
- Distribuzione con risoluzioni fino a 4320 x 4320 per occhio
- Un sistema di metadati unico per mappare l'immagine originale della camera in un ambiente a 180°
- Il formato Apple Spatial Audio Format per l'audio 3D.

Apple Immersive Video richiede un hardware in grado di gestire la riproduzione di video stereoscopico a 180° ad alta risoluzione, e Apple Vision Pro è al momento l'unico dispositivo capace di supportare questo formato.



Apple Vision Pro

Questo visore include due display micro OLED da 3660 x 3200 pixel con un pixel pitch di 7,5 micron. Include 13 camere per il tracciamento, uno scanner LiDAR, misurazione di unità inerziale (IMU), e sensori di tracciamento dell'occhio per allineare le immagini al movimento di testa e occhi.

Un processore M2 con CPU a 8 core e GPU a 10 core consente la riproduzione fino a 100 fps.

URSA Cine Immersive

Blackmagic URSA Cine Immersive è progettata per catturare Apple Immersive Video per Apple Vision Pro.



Blackmagic URSA Cine Immersive

Utilizza due sensori RGBW identici da 58 megapixel, allineati per catturare l'intero campo visivo di 180°, sia in verticale che in orizzontale. Ogni sensore è abbinato a un obiettivo fisheye a 210°, e la distanza tra i due centri delle lenti è 64 mm.

I filtri ND integrati consentono fino a 8 stop di controllo della luce, e l'apertura è fissa su f4,5 circa.

La camera registra due flussi 8K a 90 fps, richiedendo data rate notevolmente più alti delle registrazioni 4K 24p standard.

Operazioni preliminari

Per cominciare, collega l'alimentazione a Blackmagic URSA Cine Immersive e accendi la camera. Blackmagic Media Module 8TB è già inserito nella camera e formattato per cominciare subito a girare.

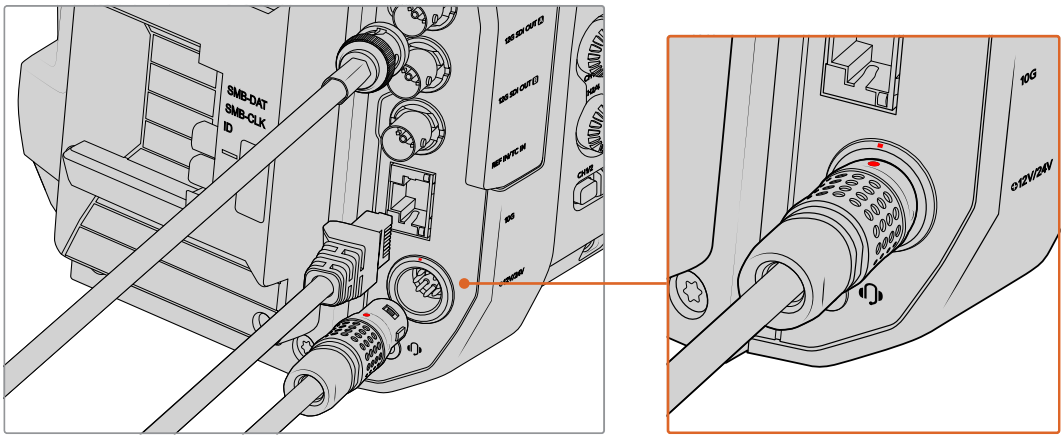
Alimentare la camera

La soluzione più veloce è utilizzare l'adattatore in dotazione.

Per collegare l'alimentazione esterna:

1. Rimuovi il tappo coprilente in gomma e mettilo da parte. Inserisci l'adattatore da AC a DC 24V nella presa di corrente con un cavo IEC C13 standard.
2. Collega l'adattatore a 8 pin al connettore DC 24V sul retro della camera.

Se sono collegate le batterie e l'alimentazione esterna, verrà usata solo quest'ultima. Se disconnetti l'alimentazione esterna mentre è collegata una batteria carica, la camera passerà a quest'ultima senza interruzioni.



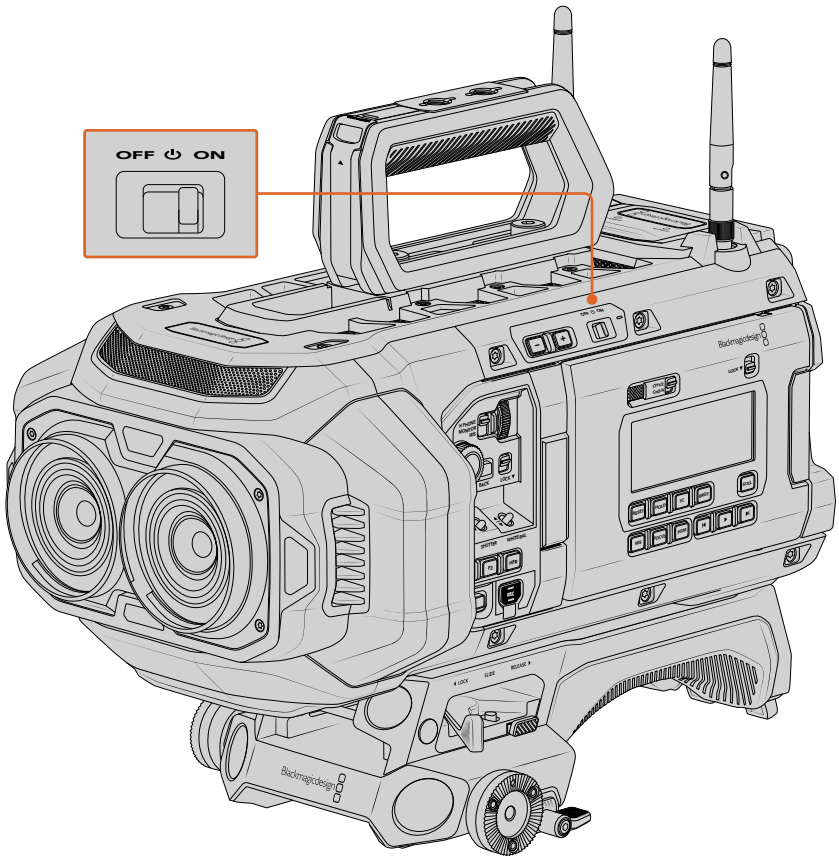
Per collegare l'alimentazione, allinea i puntini rossi e inserisci delicatamente il connettore fino a sentire un clic. Per rimuoverlo, fai presa sulla guaina in metallo del connettore e tiralo verso di te. La guaina si sposterà indietro sganciando il connettore dalla presa

La piastra per batteria sulla camera è per le batterie con attacco B da 24V. L'intervallo di tensione da 24V a 34V è ideale per operare URSA Cine Immersive perché riduce eventuali perdite del sistema termico consentendo la cattura con frame rate elevati e l'alimentazione degli accessori collegati alla camera.

Sono disponibili anche piastre opzionali VLock e Gold Mount per 24V e Gold Mount Plus per batterie di terzi da 26V.

NOTA Quando si utilizza URSA Cine Immersive con alimentazione o batterie da 12V a 18V, non è possibile alimentare anche gli accessori tramite i connettori RS a 3 pin o in stile Lemo a 2 pin.

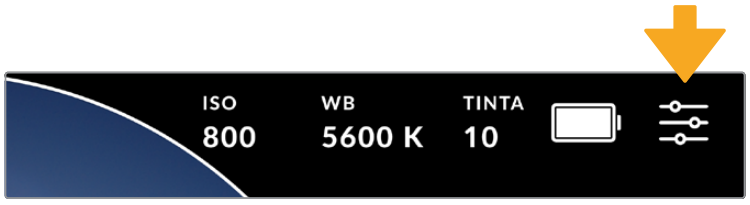
Una volta collegata una batteria o l'alimentazione esterna, accendi URSA Cine Immersive. Sposta l'interruttore sulla parte superiore della camera su **On**.



Sposta l'interruttore su On per accendere la camera

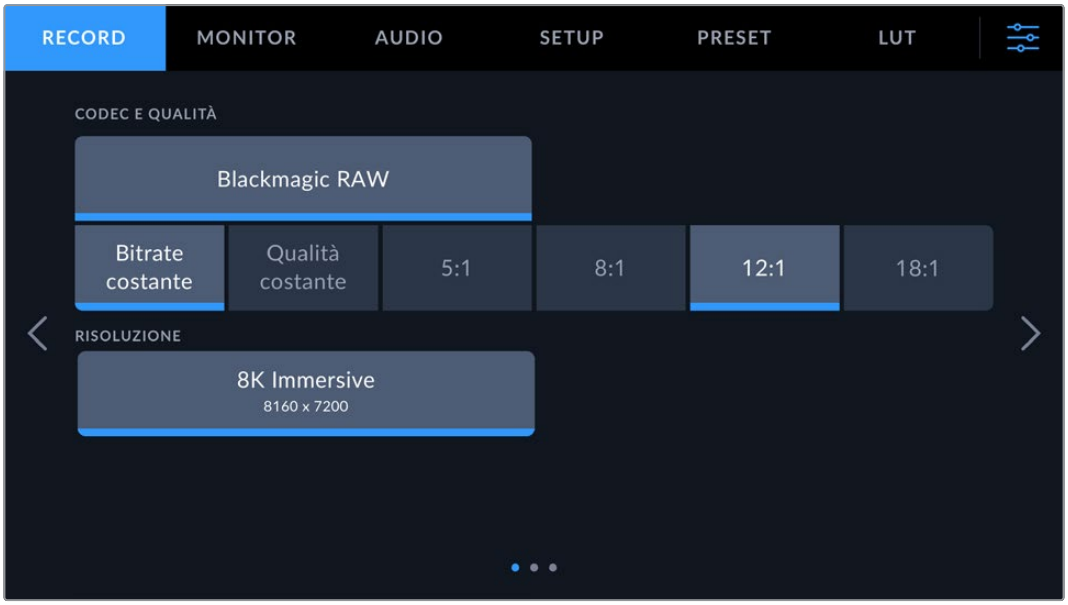
Impostare la camera per la registrazione

In questa sezione mostriamo delle impostazioni specifiche della camera che riguardano esclusivamente i contenuti immersivi. Per aprire il menù impostazioni della camera, tocca l'icona nell'angolo in alto a destra del touchscreen richiudibile o del touchscreen di monitoraggio.



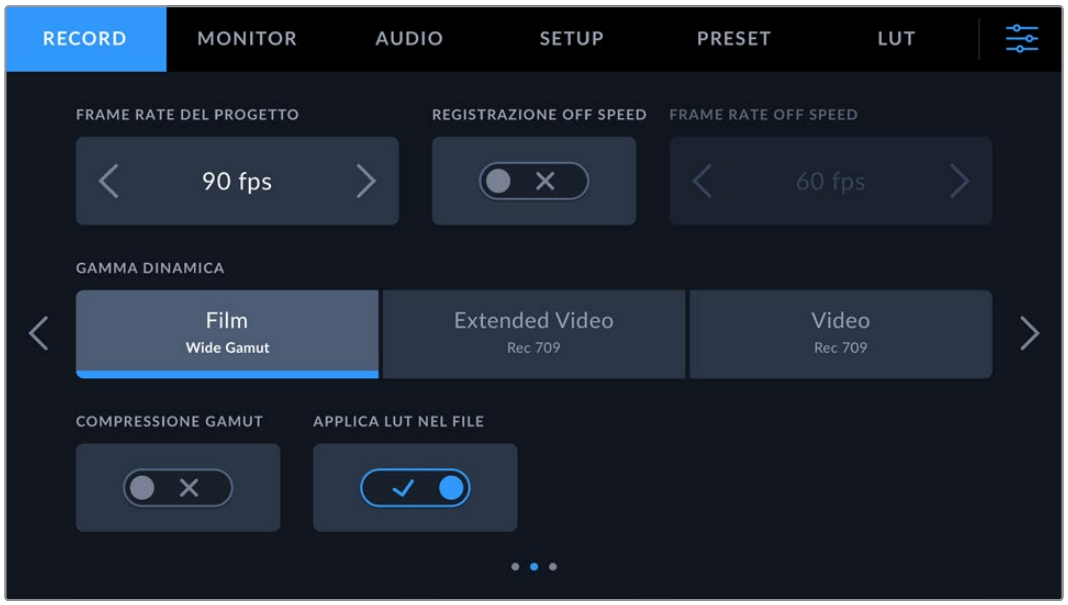
Record - Pagina 1

Scegli il codec Blackmagic RAW adatto al tuo progetto tra le quattro opzioni di **Codec e qualità** contenute in **Bitrate constante** e **Qualità costante**. Consigliamo le compressioni 12:1 o Q3 quando si registrano contenuti a 90 fps per Apple Vision Pro. Queste due impostazioni sono il perfetto compromesso tra data rate sufficienti e dimensioni di file contenute mantenendo allo stesso tempo una qualità dell'immagine altissima.



Record - Pagina 2

Qui trovi le impostazioni di frame rate.



Il frame rate del progetto della camera dev'essere impostato su 90 fps per conformarsi al frame rate del display di Apple Vision Pro.

Se stai filmando per altri sistemi, potrebbero essere sufficienti frame rate più bassi, come 60 fps o 50 fps. I frame rate tradizionali di cattura in stile film di 24 fps o 25 fps sono adatti per le riprese di sfondi VFX per contenuti cinematografici 24p.

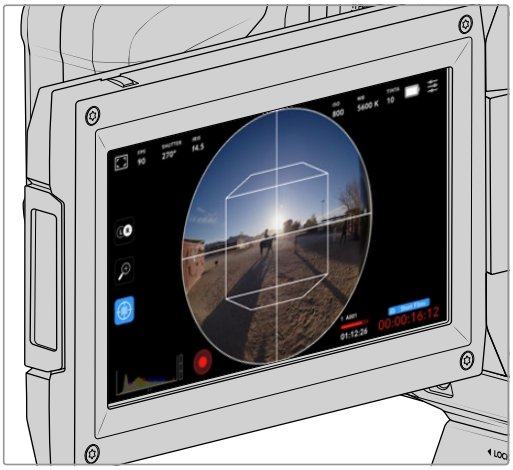
Hai completato con successo la configurazione iniziale di URSA Cine Immersive! Il Media Module è già formattato e inserito nella camera, quindi potrai cominciare subito a registrare.

Monitoraggio immagini

URSA Cine Immersive include varie opzioni per monitorare le immagini in fase di cattura.

Il touchscreen richiudibile è pensato per l'operatore di ripresa: si può invertire per ridurre il profilo della camera continuando a visualizzare lo schermo, oppure si può spostare in avanti per monitorarsi quando si riprende sé stessi.

Il touchscreen di monitoraggio consente ad altri membri della troupe di monitorare le immagini senza ostruire la visuale dell'operatore. Bisogna però fare attenzione, la camera ha un campo visivo estremamente ampio!





Opzioni di monitoraggio e sovrimpressioni

La camera mostra lo spazio visivo delle lenti, un occhio alla volta. Lo schermo richiudibile e quello di monitoraggio sono impostati per mostrare una visuale di 180°. Le uscite SDI mostrano l'immagine intera a 210° catturata da ogni sensore.



Tutte le uscite di monitoraggio mostrano lo stesso occhio, e puoi passare dall'occhio destro a quello sinistro toccando la dicitura **L** (sinistra) o **R** (destra) sui touchscreen.

Puoi abilitare un overlay opzionale sui touchscreen toccando il pulsante apposito, scegliendo tra:

- Un bordo esterno ai 180°
- Un mirino per marcare il centro dell'inquadratura
- Una guida interna che rappresenta il campo visivo approssimativo quando si guarda davanti a sé con il visore Apple Vision Pro.

Questi strumenti sono progettati per aiutarti a inquadrare e a posizionarti per la cattura immersiva.

Inquadrare la prima ripresa

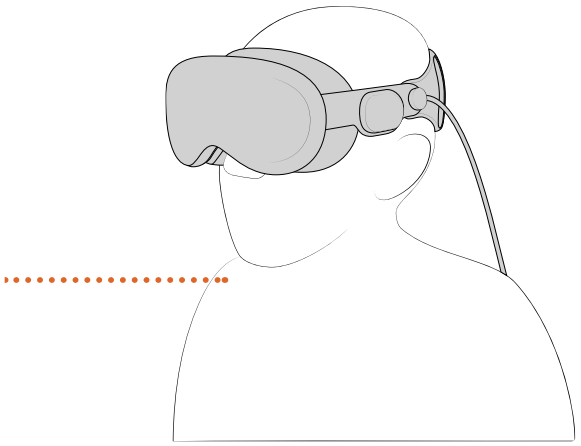
I contenuti immersivi e 2D tradizionali hanno molto in comune ma presentano anche notevoli differenze.

Per esempio, hai mai provato una sensazione di nausea mentre scrolli lo schermo del telefono in macchina? È causata dalla discordanza sensoriale tra i segnali ricevuti dal cervello: l'orecchio interno indica al corpo che ti stai muovendo, mentre i tuoi occhi ti dicono che sei immobile.

La stessa cosa può capitare con il video immersivo. Ad esempio, se muovi la camera in linea orizzontale, gli occhi indicano allo spettatore che si sta muovendo mentre il collo e l'orecchio interno che è fermo. Quindi è consigliabile cominciare a far pratica con le riprese stazionarie prima di provare a creare movimenti della camera piacevoli per il pubblico.

Altezza della camera

Nel cinema 2D tradizionale, girare con la camera ad altezze diverse determina l'equilibrio del potere in una scena. Quando si filma un video immersivo, se si posiziona la camera rasente al suolo, che impatto pensi che avrebbe sullo spettatore? Probabilmente darebbe la sensazione di essere sottoterra fino al collo.



La percezione dell'altezza del pavimento ha un forte impatto sull'esperienza visiva del pubblico. Una ripresa dal basso darebbe una prospettiva che non ha molto senso



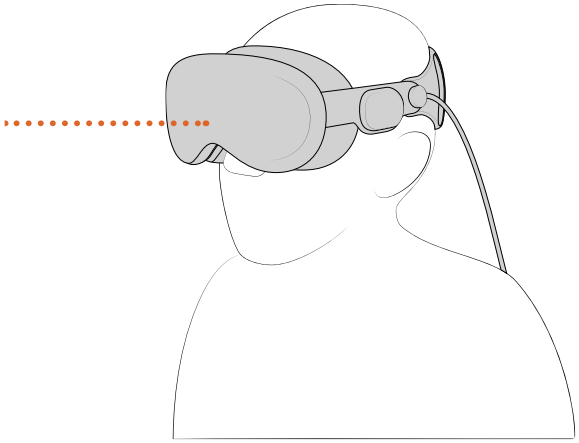
Quando si girano video immersivi, un'angolazione bassa può creare una sensazione di disagio

Per aiutare il pubblico a sentirsi fisicamente presente, è consigliabile posizionare la camera a un'altezza che lo faccia sentire seduto o in piedi nell'ambiente. C'è comunque spazio per esprimersi in modo creativo! Quando sono presenti altre persone nella scena, alzare leggermente la camera può dare allo spettatore una sensazione di potere perché le osserva dall'alto. Allo stesso modo, abbassando leggermente la camera si è obbligati a guardare le persone dal basso.

Detto questo, è possibile mantenere il pubblico a proprio agio posizionando le camere in modo più avventuroso. Ci sono numerose tecniche di ripresa applicabili alle produzioni immersive, come camere volanti, visuali dall'alto e da angolazioni basse. Con il giusto montaggio, movimento e contesto queste tecniche possono essere utilizzate per divertire il pubblico.

Inclinazione della camera

Quando si filmano contenuti 2D tradizionali e si posiziona la camera in alto o in basso, la regista spesso inclina la camera per mantenere la composizione. Applicato al video immersivo, se si inclina la camera verso il basso il pubblico viene forzato a spostare la testa verso l'alto per vedere l'orizzonte. Questo movimento non dà la sensazione di nausea, ma compromette il senso di presenza perché nella vita reale l'orizzonte è sempre direttamente davanti a noi.



Le riprese immersive hanno più senso quando l'orizzonte si trova a 0°



Inclinare URSA Cine Immersive crea confusione per il pubblico perché è costretto a guardare in una direzione insolita per vedere l'orizzonte

Per questo motivo la maggior parte delle riprese immersive vengono catturate a 0°, o con un'inclinazione vicina allo zero. La camera cattura con un campo visivo di oltre 180°, in orizzontale e in verticale, consentendo al pubblico di guardare in alto e in basso. Invece di essere la cineasta a forzare lo sguardo in una determinata direzione, il pubblico ne ha pieno controllo.

Se si vuole attirare l'attenzione dello spettatore verso parti specifiche dell'immagine, consigliamo di impiegare altre tecniche come illuminazione, apertura ridotta e linee guida.

Posizionamento

Il video immersivo viene solitamente visionato da seduti. Non essendo un formato di realtà aumentata o di realtà virtuale, lo spettatore non deve stare in piedi o muoversi. Se sei seduta al momento, facciamo un esperimento. Muovi la testa a destra e a sinistra, poi in su e in giù. Quanto riesci a girarla senza muovere il corpo? Il campo visivo del formato immersivo eccede la rotazione massima della maggior parte delle persone. Ed è un vantaggio, perché quando muovi la testa verso destra o sinistra il formato ti dà anche una visione periferica. Questo piccolo esperimento dovrebbe indicarti dove è meglio posizionare l'azione principale all'interno del campo visivo.

Raggio di movimento della testa (approssimativo)

Rotazione: 120 - 160 gradi
Flessione ed estensione: 110 - 130 gradi

Quando la rotazione della testa viene unita al movimento degli occhi, è possibile guardare oltre la propria spalla ma per molte persone non è uno sforzo sostenibile a lungo.

Per migliorare la composizione, abilita gli overlay della camera. Le guide interne che mostrano il normale campo visivo aiutano a mantenere la visibilità dei contenuti.

Un'altra considerazione è il movimento dei punti di interesse nell'inquadratura. Tracciare una persona che cammina da un estremo all'altro dell'inquadratura richiede un notevole movimento della testa. Allo stesso modo, passare da una scena in cui il punto d'interesse è al bordo dell'immagine a una scena in cui si trova nel lato opposto richiede un movimento della testa estremo. Dunque un'attenta composizione può mettere a proprio agio il pubblico e allo stesso tempo comunicare in modo creativo.

Il ruolo attivo del pubblico

Quando si girano contenuti 2D, si possono utilizzare inquadratura, obiettivi e messa a fuoco per guidare lo sguardo dello spettatore. Nel caso del video immersivo, il campo visivo di 180° rende più difficile escludere informazioni e pilotare l'attenzione del pubblico. Il video immersivo dà allo spettatore molta più libertà di esplorare le immagini, ed è uno dei motivi per cui questi contenuti sono così coinvolgenti. È comunque possibile attirare l'attenzione del pubblico con un uso attento di suono direzionale, illuminazione, pianificazione del movimento e composizione.

Movimento della camera

La camera immersiva non deve per forza rimanere immobile, ma è importante assicurarsi che il movimento sia naturale e metta a proprio agio lo spettatore. L'esempio di un veicolo in movimento dovrebbe aiutare a comprendere questo concetto. Un veicolo di grandi dimensioni ha una massa e inerzia tali da resistere a cambiamenti improvvisi di velocità o direzione. Ecco perché accelera e frena nel tempo e si muove in modo prevedibile e controllato. Una volta perfezionate le riprese statiche, quando ti senti pronto a filmare scene immersive con movimento, tieni in considerazione questi fattori:

- **Durata:** lascia che il pubblico abbia abbastanza tempo per abituarsi al movimento della camera per sentirsi a proprio agio. I movimenti brevi possono disorientare lo spettatore.
- **Intensità:** i movimenti a bassa intensità sono più facilmente integrabili nel montaggio del video immersivo. È possibile inserire movimenti ad alta intensità, ma è consigliabile preparare il pubblico prima e lasciare un po' di spazio dopo.
- **Ancoraggio visivo:** sono elementi della scena che non si muovono, essenziali in ogni inquadratura per orientare il pubblico. Gli elementi in primo piano come l'interno di un veicolo oppure oggetti distanti come l'orizzonte aiutano ad ancorare le riprese con movimento.

Tabelle di data rate

Le tabelle seguenti mostrano i data rate per i contenuti immersivi 8K x 7K a 90 fps e 24 fps.

Data rate massimi a 90 fps con Blackmagic RAW a bitrate costante

8K x 7K immersivo			
Codec	**Dimensioni dei pixel**	**Frame rate**	**Data rate**
Blackmagic RAW 5:1	8160 x 7200 stereo	90	3203 MB/s
Blackmagic RAW 8:1	8160 x 7200 stereo	90	2002 MB/s
Blackmagic RAW 12:1	8160 x 7200 stereo	90	1335 MB/s
Blackmagic RAW 18:1	8160 x 7200 stereo	90	889 MB/s

Data rate massimi a 24 fps con Blackmagic RAW a bitrate costante

8K x 7K immersivo			
Codec	**Dimensioni dei pixel**	**Frame rate**	**Data rate**
Blackmagic RAW 5:1	8160 x 7200 stereo	24	854 MB/s
Blackmagic RAW 8:1	8160 x 7200 stereo	24	534 MB/s
Blackmagic RAW 12:1	8160 x 7200 stereo	24	356 MB/s
Blackmagic RAW 18:1	8160 x 7200 stereo	24	237 MB/s

Audio e timecode

L'audio è un elemento essenziale dell'esperienza immersiva, dunque è importante che la tua URSA Cine Immersive sia equipaggiata con attrezzatura audio dedicata.

Apple Spatial Audio Format

Apple Spatial Audio Format (ASAF) si utilizza per accedere all'audio 3D con Apple Immersive Video. ASAF fa parte dell'Apple Positional Audio Framework ed è in grado di archiviare tre tipi diversi di registrazioni audio.

- **A livello di canale:** puoi integrare un mix surround tradizionale nel file ASAF. È l'opzione ideale per la narrazione da un particolare punto di vista e per la musica.
- **A livello di oggetto:** i suoni si possono trattare come oggetti individuali con metadati che descrivono la loro posizione nello spazio 3D, e sono indipendenti dai canali di ogni parlante. Si tratta solitamente di registrazioni mono. Le coordinate 3D si possono animare per seguire oggetti che si muovono all'interno della scena. È ideale per l'audio proveniente da oggetti visibili nella scena.
- **Ambisonico:** formato surround sferico in grado di distribuire il suono in qualsiasi direzione utilizzando armoniche sferiche. È ideale per il suono ambientale.

Questa sezione continua con una panoramica sull'attrezzatura necessaria per registrare un file ASAF di qualità.

Microfoni ambisonici

I microfoni ambisonici sono composti da quattro o più capsule organizzate geometricamente. Ogni capsula cattura lo stesso suono ma con tempi e intensità diversi in base alla provenienza e direzione del suono. Il canale di ogni capsula viene registrato separatamente per l'elaborazione del computer in un secondo tempo. Il software mette a confronto queste differenze per stimare la direzione e la tipologia dei suoni registrati.



Un microfono ambisonico di primo ordine

Microfoni ambisonici

I microfoni ambisonici vengono classificati in base al numero di capsule di cui sono composti. Più capsule sono presenti, più dettagli spaziali saranno riprodotti dal microfono.

Primo ordine:	4 canali
Secondo ordine:	9 canali
Terzo ordine:	16 canali
Quarto ordine:	25 canali e oltre

Il microfono ambisonico si può installare sulla camera o a lato della camera, l'importante è che sia correttamente allineato alla direzione di ripresa. Inoltre, essendo i microfoni ambisonici impiegati in una varietà di configurazioni (cardioide, laterale, verso l'alto o verso il basso, ecc.), è importante assicurarsi che l'ingegnere del suono sappia in che modo è stato utilizzato. Queste informazioni si possono condividere tramite i metadati del registratore di audio o come note.

Le registrazioni ambisoniche si possono utilizzare come bed ambientale, e poi sovrapporre altri canali ed effetti sonori legati ad oggetti. Se i microfoni sono allineati correttamente alla camera, le registrazioni ambisoniche possono anche servire come riferimento per l'ingegnere del suono.

Microfoni lavalier e in miniatura

Quando si riprende con una camera a 180° è impossibile usare un microfono boom montato su asta, perché sarebbe visibile nella scena! È dunque necessario utilizzare microfoni nascosti e in miniatura per registrare l'audio e il dialogo nella scena.

Queste sono alcune opzioni:

- Collega le fonti audio direttamente ai due ingressi XLR di URSA Cine Immersive utilizzando trasmettitori e ricevitori wireless.
- Collegali a un registratore audio esterno, ad esempio un dispositivo multicanale o uno miniaturizzato da indossare.

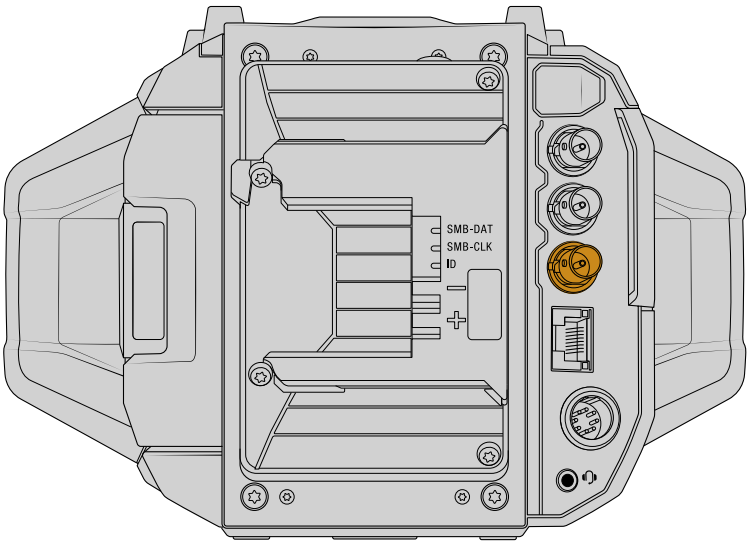
Le registrazioni verranno poi sincronizzate e mixate come canali o come suoni legati a particolari oggetti.

Microfoni a mano

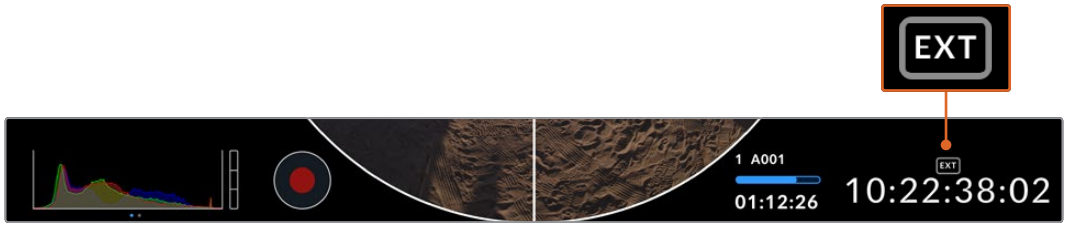
Registrare effetti sonori aggiuntivi in loco dopo che le riprese sono state completate non solo fa risparmiare tempo ma consente di catturare un audio più autentico. Avere a disposizione un microfono ipercardioide in fase di produzione è molto utile, perché si può replicare l'azione con il microfono posizionato il più vicino possibile. Le registrazioni verranno poi montate nel mix finale come canali o come suoni legati a particolari oggetti.

Collegare il timecode

Per utilizzare il timecode esterno con URSA Cine Immersive, con un cavo BNC collega un generatore di timecode all'ingresso timecode situato nel retro della camera. Collegarsi a un timecode sincronizzato quando suono e immagini vengono registrati separatamente garantisce che audio e video si possano facilmente unire in fase di post produzione.



Quando a questo ingresso è connesso un timecode esterno con lo stesso frame rate, la camera vi si sincronizza automaticamente. L'etichetta **EXT** appare sul touchscreen di URSA Cine Immersive quando utilizza un timecode esterno. Se disconnetti il cavo, il timecode rimane sincronizzato con quello interno della camera, e il touchscreen mostra la dicitura **INT**.



La dicitura EXT appare sul timecode se la camera è connessa a un generatore di timecode

NOTA Esistono standard timecode SMPTE per i comuni frame rate come 23.976, 24 e 30, ma non per 90 fps. Ecco perché quando il frame rate del progetto di URSA Cine Immersive è impostato su 90 fps, la camera utilizza lo stesso timecode del 30 fps. Anche le uscite SDI operano a 30p. Qualsiasi accessorio o registratore esterno di audio va configurato su 30 fps.

Riprendere video immersivo

Ora che hai una conoscenza di base del funzionamento di Blackmagic URSA Cine Immersive e di come sfruttare al meglio il formato immersivo, in questa sezione imparerai a operare la camera.

Requisiti di esposizione

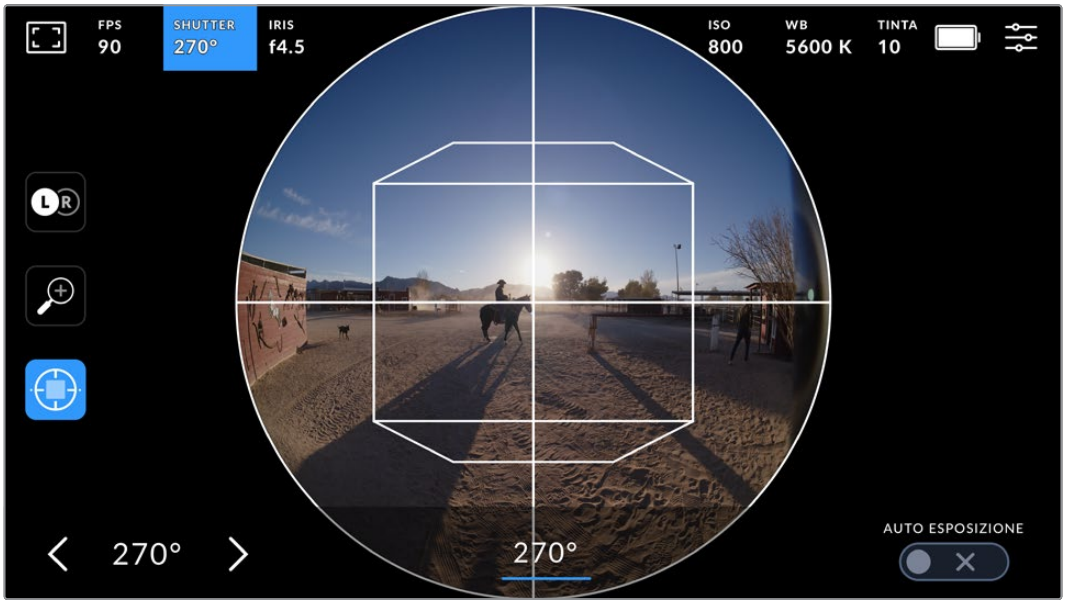
Apple Immersive Video richiede una maggiore esposizione rispetto alla cattura tradizionale a 24 fps. Dovrai dunque rivalutare il tuo modo di riprendere e di illuminare la scena. Se mantieni lo shutter a 180 gradi, la cattura a 90 fps richiede un'esposizione quasi quattro volte maggiore per la stessa scena. Inoltre l'obiettivo ha il diaframma fisso su un valore di circa f4,5, ovvero tre stop più scuro delle comuni lenti super veloci. Significa che serviranno fino a sette stop di luce in più per ottenere la giusta esposizione. In ogni caso, se riprendi a 90 fps, utilizzando un angolo dell'otturatore ampio tra 270° e 360° aiuta a compensare per la perdita di luce degli alti frame rate e allo stesso tempo ottenere una piacevole sfocatura di movimento.

Velocità dell'otturatore

L'indicatore **Shutter** mostra l'angolo o la velocità dell'otturatore. È possibile visualizzare le informazioni di shutter come angolo dell'otturatore o come velocità dell'otturatore dal menù SETUP.

Per cambiare la velocità dell'otturatore:

1. Tocca l'icona **Shutter** nella barra degli strumenti superiore.
2. Usa la freccia destra o sinistra per passare tra i valori più comuni di shutter. Se preferisci inserire un valore manualmente, tocca due volte il valore corrente tra le due frecce in basso a sinistra e usa la tastiera che appare per inserire fino a due decimali tra 11,2 e 360 gradi.



Il frame rate desiderato può essere inserito con la tastiera a schermo

La velocità dell'otturatore ha meno impatto su come appare il video immersivo ad alto frame rate. Anche se un valore di 180° tende ad essere lo standard per chi riprende a 24 o 30p, è sempre più comune utilizzare la velocità dell'otturatore per regolare l'esposizione quando si cattura il video immersivo.

Per le riprese con poca luce, il valore dello shutter si può aumentare a 270°, o in casi eccezionali anche a 360°.

Per le riprese con molta luce, il valore dello shutter si può diminuire fino a 11,2°, anche se è più comune lasciare la velocità dell'otturatore sull'impostazione da te scelta per il movimento e utilizzare i filtri ND per controllare l'esposizione.

Un altro fattore da prendere in considerazione è che il frame rate elevato di 90 fps di Apple Immersive Video può causare problemi con lo sfarfallio delle luci. Consigliamo di impostare lo shutter su 270° negli ambienti con frequenza di 60Hz, e su 324° per i 50Hz.

ISO e gamma dinamica

L'indicatore ISO mostra le impostazioni ISO della camera, ovvero la sensibilità alla luce. Tocca questo indicatore per regolare l'ISO secondo le condizioni di luce. I valori di ISO di URSA Cine Immersive vanno da 200 a 3200, e 800 è quello ottimale.



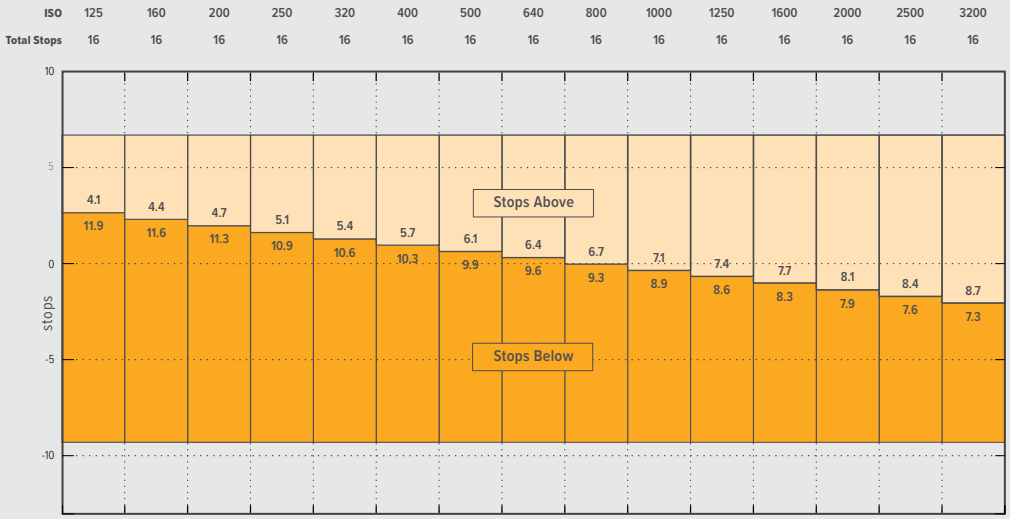
Tocca l'indicatore ISO per configurarlo

Puoi anche toccare il numero del valore predefinito come stop completo. Lo slider sotto i predefiniti ti permette di regolare l'ISO con precisione in incrementi di 1/3 di stop.

Scegli l'impostazione in base alle condizioni di luce. Per esempio, con poca illuminazione un ISO di 1600 è adatto, ma può causare la comparsa di rumore. In condizioni di elevata luminosità, un ISO di 200 è ideale per catturare colori più intensi.

Le camere Blackmagic catturano immagini con un'ampia gamma dinamica, consentendoti un enorme margine di manovra sia nelle riprese che in fase di color grading su DaVinci Resolve. La tabella sottostante mostra la gamma dinamica disponibile su URSA Cine Immersive.

Gamma dinamica di URSA Cine Immersive



Filtri ND

I filtri ND della camera sono il modo più comunemente usato per controllare l'esposizione, perché riducono la quantità di luce che raggiunge il sensore della camera di un numero predefinito di stop.



I filtri ND di URSA Cine Immersive si regolano con i pulsanti sulla parte sinistra sopra il touchscreen richiudibile. Scegli tra nessun filtro, **2**, **4**, **6** e **8** stop.

L'impostazione ND attuale viene confermata dall'indicatore in alto a sinistra del touchscreen. Quando viene utilizzato, l'indicatore rimane acceso. Se è impostato su nessun filtro, l'indicatore scomparirà dopo 4 secondi.



NOTA Il valore del filtro ND si può visualizzare come numero ND, stop, o frazione.

Registrazione off speed

Il frame rate del progetto nel menù SETUP indica il formato di registrazione di URSA Cine Immersive e definisce la velocità di riproduzione dei filmati. Dunque normalmente asseconda la velocità di riproduzione e la sincronizzazione audio adoperate in post produzione, oltre ai requisiti di consegna.



Tocca l'indicatore FPS per accedere alle impostazioni del frame rate di URSA Cine Immersive. Le 8 opzioni di frame rate del progetto sono 23.98, 24, 25, 29.97, 30, 50, 59.94, 60 e 90 fps. Per le riprese immersive va impostato su 90 fps.

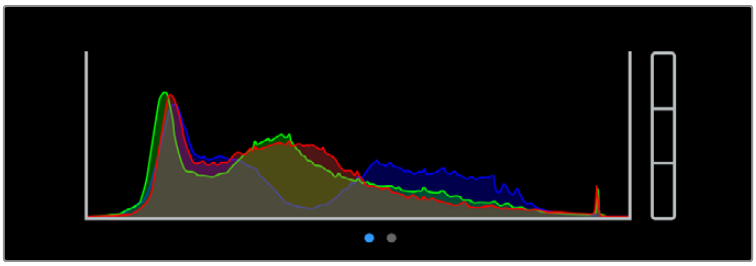
Invece il frame rate off speed definisce la velocità di cattura del sensore. I frame rate di progetto e del sensore di URSA Cine Immersive combaciano di default per riprodurre a velocità normale. Per modificare solo il frame rate del sensore, attiva l'interruttore **Frame rate off speed** in basso a destra del menù FPS.

Quando è abilitato, puoi impostare manualmente il valore di frame rate off speed. Ricorda che 90 fps è il frame rate più alto offerto dalla camera e combacia perfettamente con il frame rate del progetto di 90 fps delle riprese per Apple Vision Pro. In ogni caso, scegliendo un frame rate del progetto più basso come 30 o 60 fps e poi girando off speed a 90 fps si catturano meravigliosi filmati al rallentatore.

Quando l'interruttore Frame Rate Off Speed è abilitato, l'indicatore di frame rate in alto a sinistra dello schermo mostra il frame rate del sensore seguito dal frame rate del progetto.

Istogramma

In basso a sinistra del touchscreen di URSA Cine Immersive trovi l'istogramma, un grafico che mostra la distribuzione della luminosità nell'immagine. La parte sinistra rappresenta le ombre, quella centrale i mezzitoni, e la parte destra le luci. L'altezza del grafico indica la porzione dell'immagine a quel livello di luminosità.



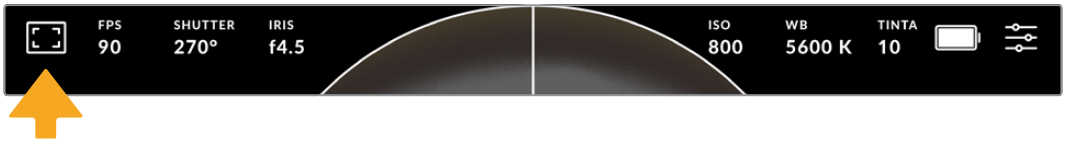
L'istogramma fornisce la distribuzione tonale della clip a colpo d'occhio. Un'immagine in cui il grafico si mostra sulla parte sinistra è probabilmente sottoesposta; allo stesso modo, se il grafico si mostra sulla parte destra l'immagine è in tutta probabilità sovraesposta. L'istogramma va sempre interpretato nel contesto dell'immagine, ma in generale se un'immagine è ben esposta il grafico è distribuito nella zona centrale. Aumentando l'esposizione vedrai il grafico dell'istogramma spostarsi verso destra, diminuendola, verso sinistra.

A destra dell'istogramma c'è un indicatore che mostra il clipping dei singoli canali di colore. Normalmente il clipping va evitato, anche se in determinate situazioni è accettabile eccedere i livelli per preservare i mezzitoni e le ombre.

Per esempio la tecnica dell'esposizione a destra prevede di sovraesporre l'immagine spingendo l'istogramma più a destra possibile senza clippare i canali di colore. In questo modo si sfrutta al massimo il segnale riducendo al minimo il rumore. Basta solo abbassare l'esposizione in fase di post, e si otterranno immagini nitidissime!

Opzioni di monitoraggio con il display LCD

Tocca l'icona del monitor in alto a sinistra del touchscreen LCD di URSA Cine Immersive per accedere alle funzioni di monitoraggio. Questo menù contiene impostazioni utili come la luminosità del monitor, oltre a potenti strumenti per regolare l'esposizione.



Zebra

Abilita o disabilita la funzione Zebra sul touchscreen e regolane la percentuale. Questa funzione copre con linee diagonali le aree dell'immagine che eccedono la percentuale di esposizione impostata. Questa percentuale aiuta a mostrare quali parti dell'immagine sono sovraesposte, e solitamente viene impostata su 95% invece che su 100% perché consente di identificare le aree dell'immagine che si possono salvare dalla sovraesposizione. La seconda opzione di zebra ti permette di abilitare o disabilitare i livelli del grigio medio o del grigio medio +1 stop.

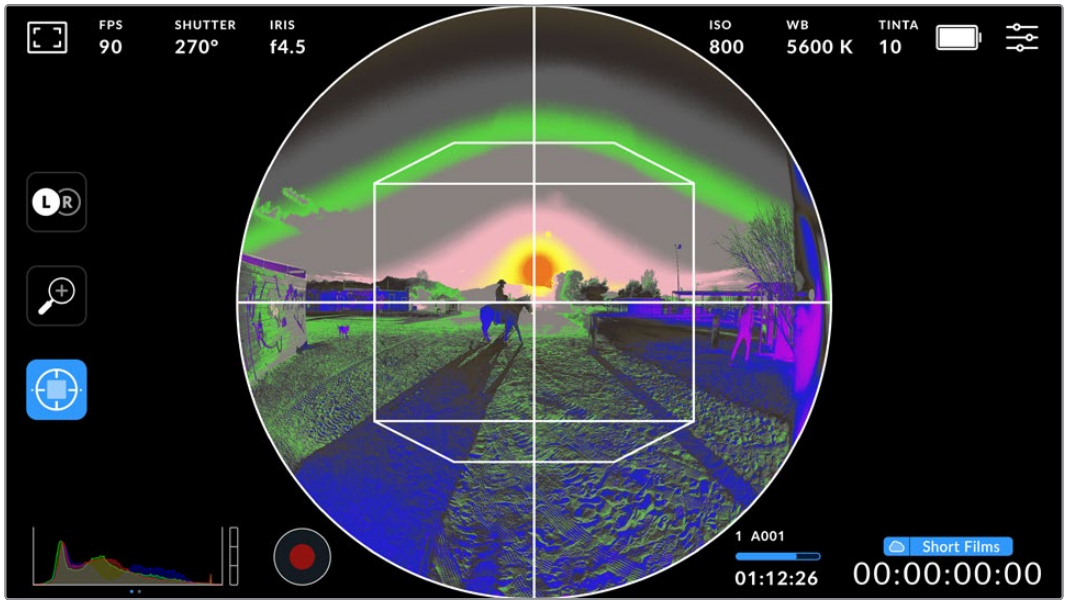
Griglie

Le griglie dividono lo schermo in terzi, oppure mostrano un mirino, un punto centrale o l'orizzonte sul touchscreen LCD e su tutte le uscite di URSA Cine Immersive. Quando si riprende il video immersivo, l'indicatore dell'orizzonte è molto utile. Mostra un doppio mirino al centro dell'immagine che si allinea solo quando la camera è perfettamente in piano.

Falsi colori

Questa funzione copre vari elementi nell'immagine di colori che rappresentano il loro livello di esposizione ideale. Per esempio il rosa rappresenta l'esposizione ottimale per la pelle chiara, mentre il verde si riferisce alle tonalità più scure. In questo caso, osservando il rosa o il verde durante le riprese potrai mantenere costante l'esposizione delle tonalità della pelle.

Se gli elementi dell'immagine cambiano dal giallo al rosso, significa che sono sovraesposti. Come l'istogramma, questo strumento palesa a colpo d'occhio la distribuzione tonale dell'immagine.



Registrazione

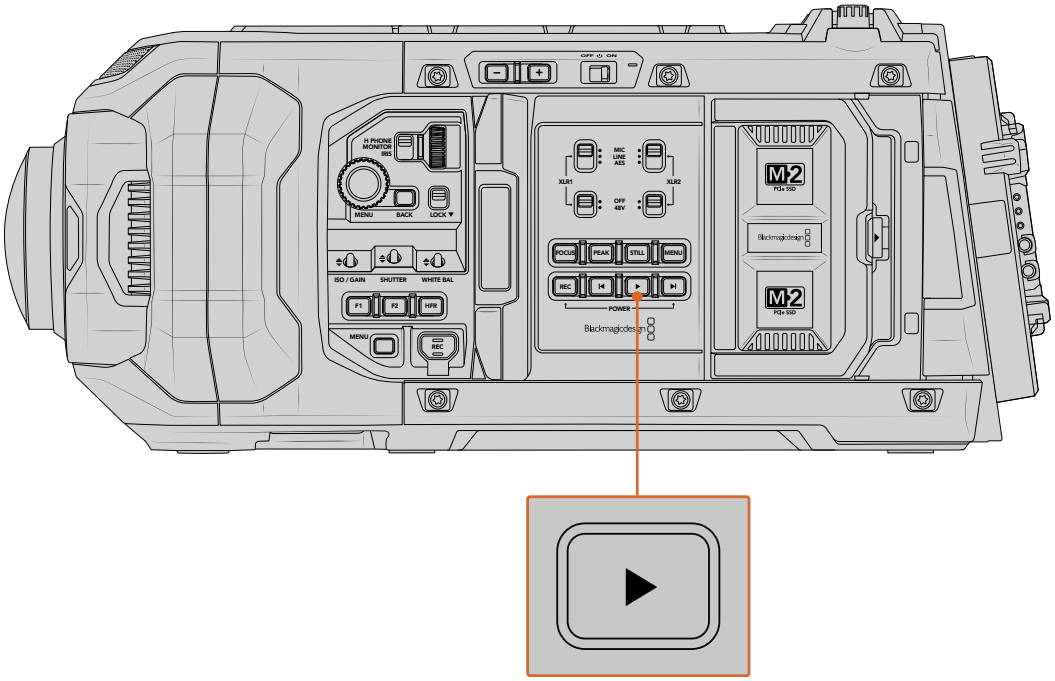
Premi uno dei pulsanti REC sulla camera per iniziare a registrare. I pulsanti di registrazione si trovano sul pannello frontale, sul pannello di controllo interno dietro al touchscreen richiudibile, nella sezione di monitoraggio e su entrambi i touchscreen LCD. Premi di nuovo il pulsante REC per interrompere la registrazione.

Gli indicatori dei supporti di memoria confermano il tempo residuo di registrazione, in ore:minuti:secondi secondo il frame rate selezionato. La barra sopra il tempo residuo è blu, bianca o rossa in base allo stato del supporto di memoria, e indica lo spazio utilizzato.



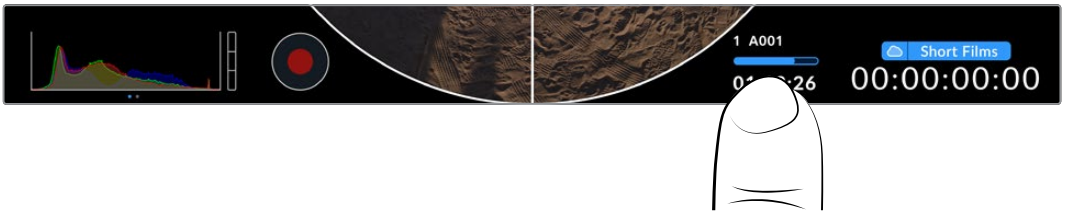
Riproduzione

Una volta registrate le clip, utilizza i pulsanti di trasporto per riprodurle. Per riprodurre l'ultima clip registrata, premi il pulsante Play sul lato sinistro della camera, dietro il touchscreen richiudibile. Con i pulsanti Avanti e Indietro si passa alle altre clip. Per uscire dalla riproduzione e preparare la camera per ricominciare a registrare, premi una volta REC.

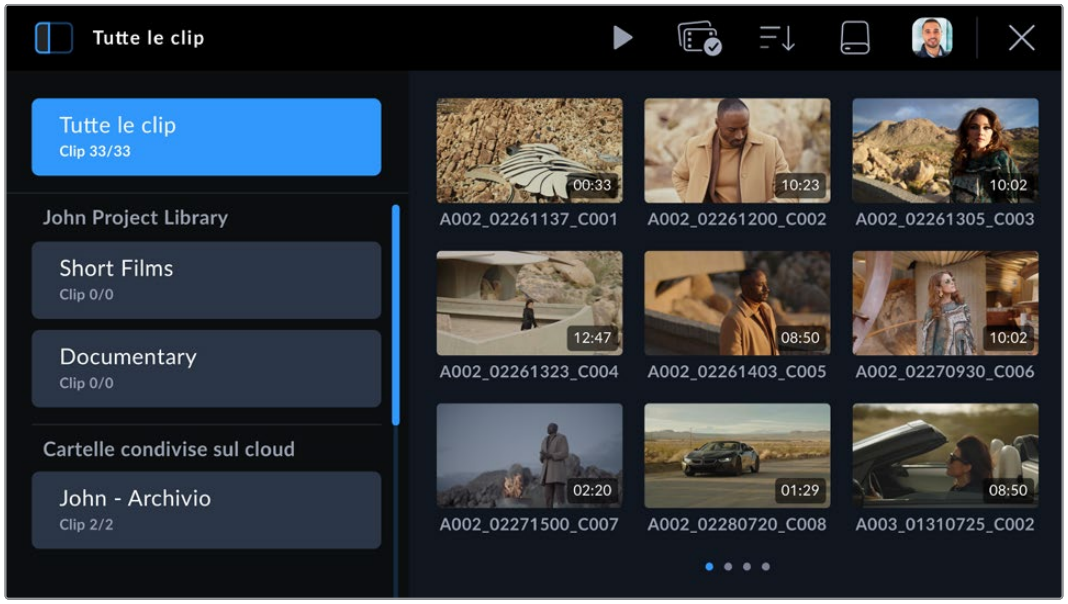


Premi i pulsanti Avanti e Indietro per andare all'inizio o alla fine delle clip. Premendo Indietro una volta si torna all'inizio della clip corrente. Premendolo due volte si salta all'inizio della clip precedente. Tieni premuto uno o l'altro per riprodurre o tornare indietro a velocità 2x. Sempre in questa modalità premi due volte i rispettivi pulsanti per aumentare la velocità di 4x, tre volte per 8x, e quattro volte per 16x.

Se è necessaria una revisione più approfondita del girato, è consigliato l'uso dell'archivio multimediale. Per aprire l'archivio multimediale, tocca uno degli indicatori dei supporti di memoria in basso nel touchscreen.



Il browser della pagina principale dell'archivio multimediale mostra le miniature delle clip registrate su tutti i supporti di memoria collegati alla camera. Le miniature sono disposte su tre file e, man mano che vengono aggiunte altre clip, basta sfogliare con il dito per andare alla pagina successiva. Il numero di pagine disponibili è indicato dai puntini centrali in basso nel touchscreen.



Connetti URSA Cine Immersive a un account Blackmagic Cloud e invia il girato direttamente al progetto di DaVinci Resolve.

Per uscire dall'archivio multimediale in qualsiasi momento, tocca la X nell'angolo in alto a destra o premi uno dei pulsanti REC.

Trasferire le clip

Ci sono vari modi per trasferire le clip dall'archivio della camera. Ad esempio, collega URSA Cine Immersive al computer o a una rete tramite un cavo 10G Ethernet e copia le clip dall'archivio della tua camera tramite internet. È anche possibile estrarre il Media Module e inserirlo in un Blackmagic Media Dock connesso alla tua rete o computer, così da accedere fino a tre Media Module contemporaneamente.

Se colleghi la camera al computer tramite ethernet, il computer riconoscerà il Blackmagic Media Module come un qualsiasi Blackmagic Cloud Store.

La camera è impostata su DHCP di default, quindi il tuo computer o rete le assegneranno automaticamente un indirizzo IP.

Abilitare la condivisione dei file

Durante la configurazione iniziale della camera quando viene collegata al computer o alla rete tramite ethernet, la condivisione dei file mediante SMB è disabilitata di default per salvaguardare i file master della camera dopo una giornata di riprese. Infatti i file non saranno subito accessibili come quelli file salvati in un Blackmagic Cloud Store per la condivisione immediata con gli altri utenti nella rete. Per questo motivo lasciamo a te la scelta di abilitare la condivisione dei file tramite SMB.

1. Per farlo, scarica e installa sul computer l'utilità Blackmagic Camera Setup. Con questa applicazione potrai anche aggiornare il software della camera e cambiare le impostazioni da amministratore. Blackmagic Camera Setup è disponibile per il download alla pagina https://www.blackmagicdesign.com/it/support/family/professional-cameras
2. Una volta scaricata e installata l'utilità, collega la camera al computer tramite USB e apri Blackmagic Camera Setup. Connetti la porta USB sulla parte superiore della camera, verso il retro.

3 Per aprire le impostazioni, clicca sull'immagine della URSA Cine Immersive o sull'icona delle impostazioni.

4 Scorri alla sezione **Network access** > **File sharing (SMB)** e abilita con **Enable**. Poi seleziona **Save** in basso a destra per rendere effettive le impostazioni.

5 Ora clicca sull'icona con il quadratino alla voce URL per compilare il campo automaticamente. L'indirizzo URL mostrerà il nome della camera come destinazione di rete locale.

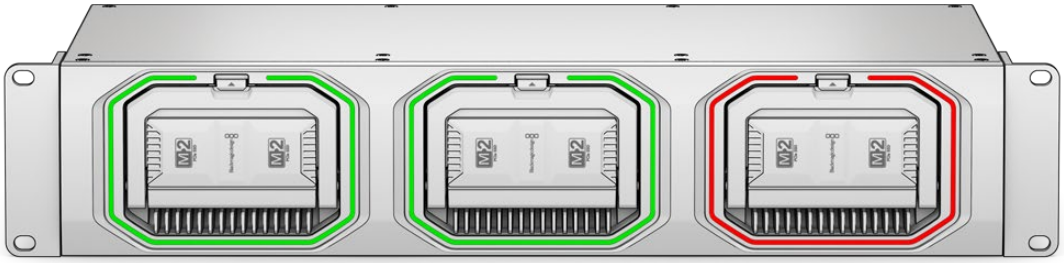
6 Conferma con **Save**.

Per accedere immediatamente alle clip, clicca sull'URL della camera. Si aprirà una finestra chiedendo di inserire una password o di accedere come guest. Seleziona **Guest** e clicca **Connect**.

7 Il computer navigherà automaticamente nel Media Module dal Finder su Mac e con Windows Explorer su Windows. L'utilità si ricorderà dell'impostazione ethernet, quindi ti basterà occupartene una volta sola in fase di configurazione iniziale. Ogni volta che la camera si connette tramite ethernet potrai accedere subito al Media Module senza dover riaprire queste impostazioni.

Utilizzare Blackmagic Media Dock

Blackmagic Media Module è stato progettato per essere inserito e rimosso velocemente, per trasferire le clip su un archivio esterno o per sostituirlo con un modulo vuoto per continuare a registrare. Blackmagic Media Dock si può utilizzare per accedere al girato dal Media Module senza la camera.



Riepilogo

Apple Immersive Video introduce un approccio innovativo allo storytelling, offrendo un'alternativa al cinema 2D tradizionale. Non vuole prenderne il posto ma piuttosto proporre un nuovo formato selezionabile in base alle esigenze del progetto. Come per ogni formato, le scelte creative cambiano a seconda della storia che si vuole raccontare.

Spesso i nuovi formati richiedono flussi di lavoro più complessi, ma URSA Cine Immersive è stata sviluppata per semplificare il processo di cattura di Apple Immersive Video. Oltre a snellire la produzione di contenuti immersivi, si può anche utilizzare per gli effetti visivi e altre applicazioni.

Questa guida descrive le funzioni e i flussi di lavoro principali di URSA Cine Immersive, ed è intesa come punto di partenza. Come con qualsiasi impianto camera, sono necessari ulteriori ricerche e test per ottenere i migliori risultati per le proprie esigenze di produzione.

DaVinci Resolve Studio



DaVinci Resolve Studio è un software di post produzione completo che integra perfettamente montaggio, correzione colore, effetti visivi, grafica in movimento e post produzione audio. Supporta flussi di lavoro adatti alle caratteristiche uniche del formato Blackmagic RAW Immersive, tra cui video stereoscopico ad altissima risoluzione, frame rate di 90 fps e mixaggio di audio spaziale. DaVinci Resolve Studio è progettato per gestire produzione, monitoraggio e consegna di materiale Apple Immersive Video, ed è dunque uno strumento formidabile per i creatori di contenuti immersivi.

Requisiti della workstation

Requisiti hardware

- Mac Studio M3 Ultra Desktop
- Monitor ad alta frequenza di aggiornamento (minimo 60Hz)

Requisiti minimi di sistema

- macOS 15 o successivo
- 64 GB di memoria di sistema
- Per il monitoraggio, Blackmagic Design Desktop Video 12.9 o successivo
- Computer dotati di processore Apple Silicon o GPU compatibile con Metal

Requisiti per il monitoraggio immersivo

- Apple Vision Pro
- Rete wireless ad alta velocità (minimo Wi-Fi 5, Wi-Fi 6 consigliato)

Configurazione del progetto

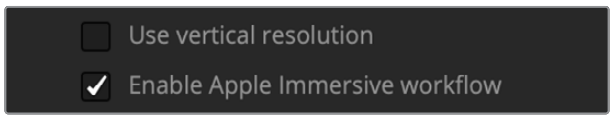
Prima di iniziare ad acquisire file RAW con Blackmagic URSA Cine Immersive, consigliamo di configurare le impostazioni di progetto di DaVinci Resolve per garantire che i file siano debayerizzati ed elaborati correttamente per ottenere risultati visivi ottimali e facilitarne il montaggio.

Impostazioni principali

È fondamentale indicare che stai per creare contenuti per Apple Vision Pro durante la configurazione iniziale di un progetto, in modo che tu possa vedere tutte le opzioni giuste nelle fasi di preparazione, assemblaggio e consegna.

1. Avvia DaVinci Resolve Studio.
2. Nell'angolo in basso a destra della finestra di **Gestione del progetto,** clicca su **Nuovo progetto**.
3. Inserisci un nome per il progetto, imposta la posizione del materiale (se necessario) e clicca su **Crea**.
4. Apri le impostazioni del progetto cliccando sull'icona con l'ingranaggio nell'angolo in basso a destra di DaVinci Resolve Studio.

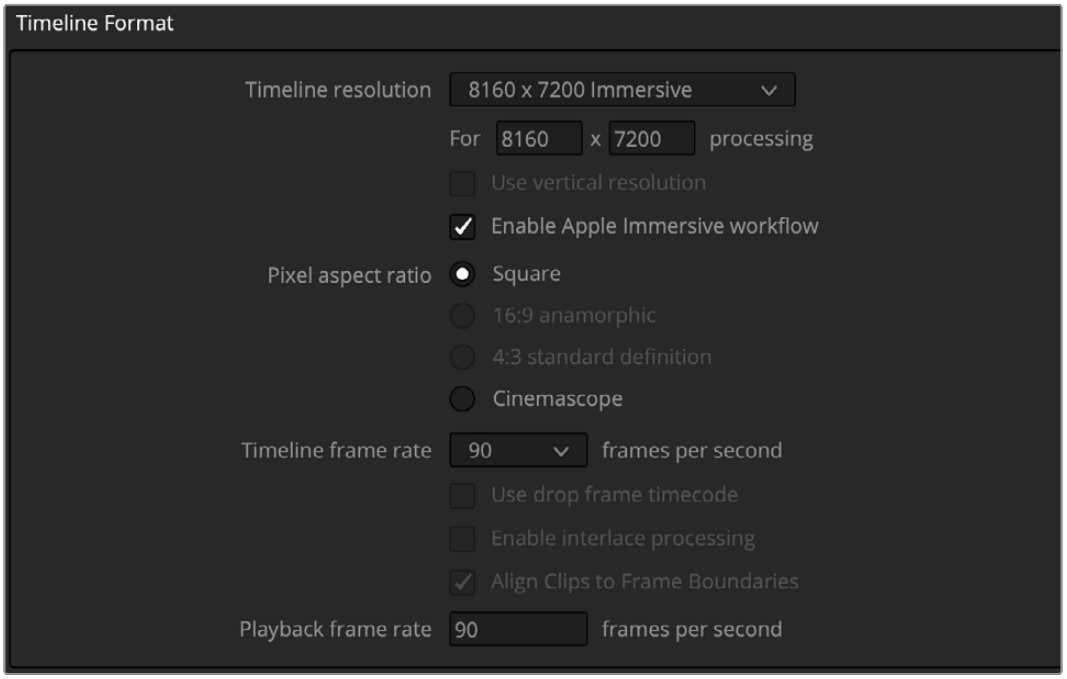
   Innanzitutto devi impostare la risoluzione e il frame rate del progetto.
5. Seleziona la voce **Principali** nella barra laterale a sinistra.
6. Nella sezione Formato della timeline, spunta la casella **Enable Apple Immersive workflow** in modo da configurare le impostazioni ottimali per i flussi di lavoro immersivi. Quando crei un progetto immersivo, assicurati di selezionare quest'opzione per prima.



7. All'opzione **Risoluzione timeline** seleziona **8160 x 7200 Immersive** per elaborare le clip nella risoluzione nativa del sensore.

   Se stai lavorando con un sistema dalle prestazioni limitate, scegli **4080 x 3600 Immersive** per dimezzare la risoluzione e velocizzare la riproduzione delle clip.
8. L'opzione **Frame rate timeline** dovrebbe essere impostata su **90 fps**. Questo parametro viene configurato automaticamente quando abiliti il workflow Apple Immersive.
9. Imposta l'opzione **Frame rate riproduzione** su **90 fps**. per ottenere la riproduzione più fedele possibile dei tuoi contenuti video immersivi.

   Se stai lavorando con un sistema dalle prestazioni limitate (o un monitor con una bassa frequenza di aggiornamento), diminuisci il frame rate per ottenere una riproduzione più stabile. Consigliamo di scegliere una frazione del frame rate nativo, ad esempio 30 fps o 60 fps.



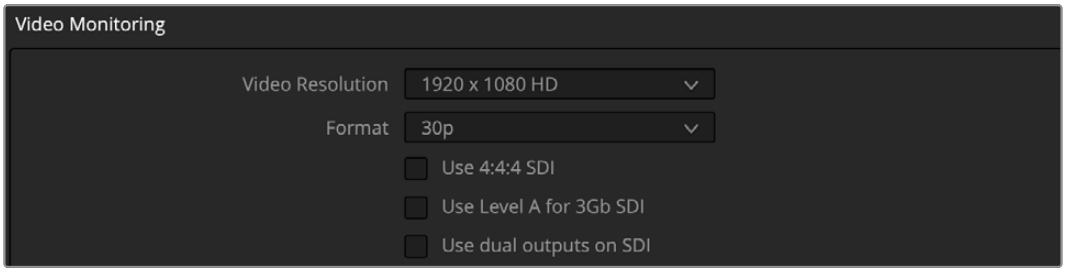
Monitoraggio video

Se utilizzi un monitor esterno (con un dispositivo di acquisizione/riproduzione come Decklink o UltraStudio), configura le impostazioni di monitoraggio video in base alle specifiche tecniche del tuo monitor.

1. Nella sezione **Monitoraggio video**, imposta la voce **Risoluzione video** in modo che combaci con la risoluzione del tuo monitor, ad esempio 1920 x 1080 HD o 3840 x 2160 Ultra HD.

   Sebbene il frame rate nativo del video immersivo sia 90 fps, non è un frame rate compatibile con la maggior parte dei monitor broadcast.

2. Imposta l'opzione **Formato** sul frame rate del tuo monitor, ad esempio **30p**.



Ridimensionamento

Le impostazioni di ridimensionamento fanno in modo che la risoluzione e il rapporto d'aspetto dei file immersivi vengano mostrati correttamente sul visore. Questi parametri vengono configurati automaticamente quando selezioni l'opzione **Enable Apple Immersive workflow** nelle impostazioni principali, ma è consigliabile verificare che siano corretti.

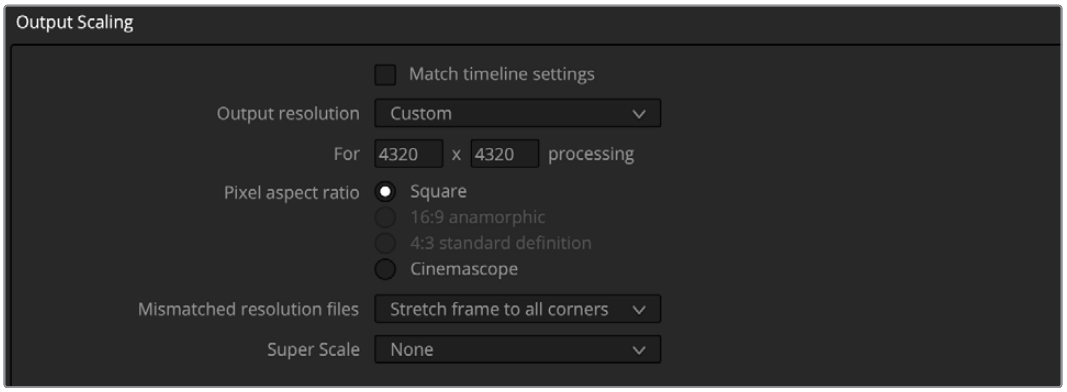
1. Seleziona la voce **Ridimensionamento** nella barra laterale della finestra Impostazione del progetto.
2. Nella sezione **Ridimensionamento di uscita**, assicurati che l'opzione **Rispecchia impostazioni della timeline** non sia selezionata.

3 Verifica che l'opzione **Risoluzione di uscita** sia su **Custom** e impostata su **4320 x 4320**.

Dimezza la risoluzione (**2160 x 2160**) per i monitor più piccoli e/o per ottenere una riproduzione più veloce.

Queste risoluzioni con rapporto d'aspetto quadrato sono necessarie per riprodurre i contenuti Apple Immersive Video all'interno di DaVinci Resolve Studio e per trasmettere in streaming all'Apple Vision Pro (tramite Wi-Fi o Developer Strap).

4 Assicurati che l'opzione **File in risoluzioni differenti** sia impostata su **Dilata fotogramma e riempi**.



Gestione del colore

Visto che è necessario un controllo personalizzato per la luminanza massima in nit, consigliamo di gestire il colore utilizzando i nodi Trasformazione di spazio colore nella pagina Color. L'obiettivo delle opzioni di gestione del colore nelle impostazioni di progetto è quello di specificare lo spazio colore della timeline nella pagina Color e di impostare i metadati dello spazio colore che saranno associati al file renderizzato.

1 Seleziona la voce **Gestione del colore** nella barra laterale a sinistra.

2 Mantieni **Scienza colore** su **DaVinci YRGB**.

3 Imposta **Spazio colore timeline** su **DaVinciWG/Intermediate**.

Questo è uno spazio colore a prova di futuro che supporta i flussi di lavoro HDR, risultando ideale per i file di Blackmagic URSA Cine Immersive e i formati di consegna di Apple Vision Pro.

4 Imposta **Spazio colore uscita** su **P3-D65 ST2084 1000 nit**.

Questo spazio colore offre il gamut adatto al monitoraggio dei visori DaVinci Resolve su un computer Mac Studio, solitamente Display P3. Se utilizzi un monitor professionale per il grading o un altro profilo Apple ICC, bisogna cambiare lo spazio colore dell'uscita sullo standard del colore del tuo schermo.

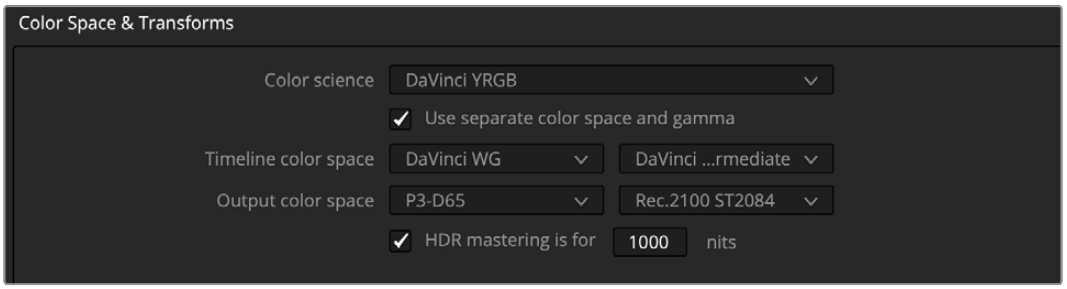
Tuttavia dovrai modificare la gamma (funzione di trasferimento) per mappare correttamente il profilo ICC.

5 Spunta la casella **Separa spazio colore e gamma** sopra le opzioni dello spazio colore.

Ora i menù a discesa dello spazio colore e della gamma appaiono separati.

6 Imposta lo spazio colore uscita (nella colonna di destra) su **Rec.2100 ST2084**.

7 Seleziona **Mastering HDR per 1000 nit**.



Per velocizzare la creazione dei progetti Apple Immersive successivi, puoi salvare le impostazioni attuali come predefinito.

8 Clicca sull'icona con i tre punti nell'angolo in alto a destra della finestra delle impostazioni del progetto e seleziona **Salva le impostazioni come predefinito**.

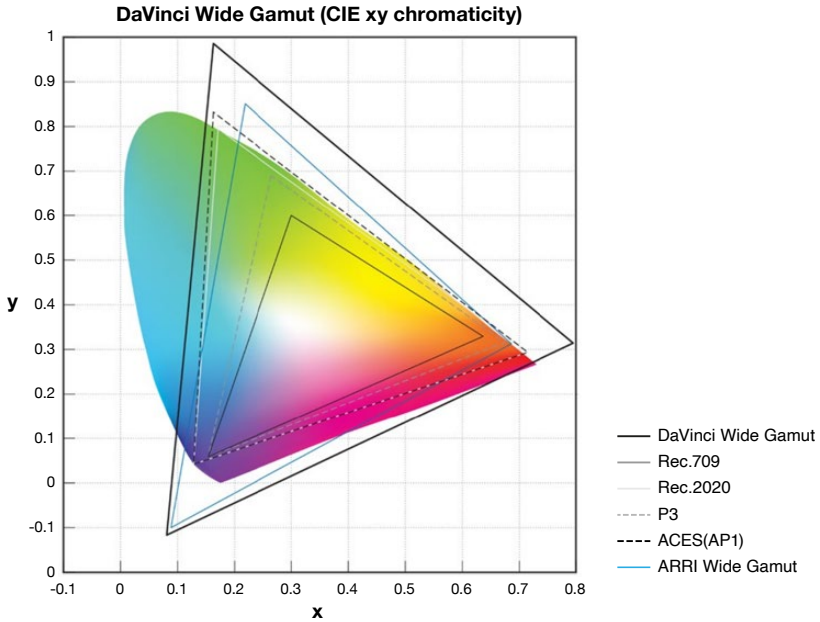
9 Dai un nome al predefinito e clicca OK.

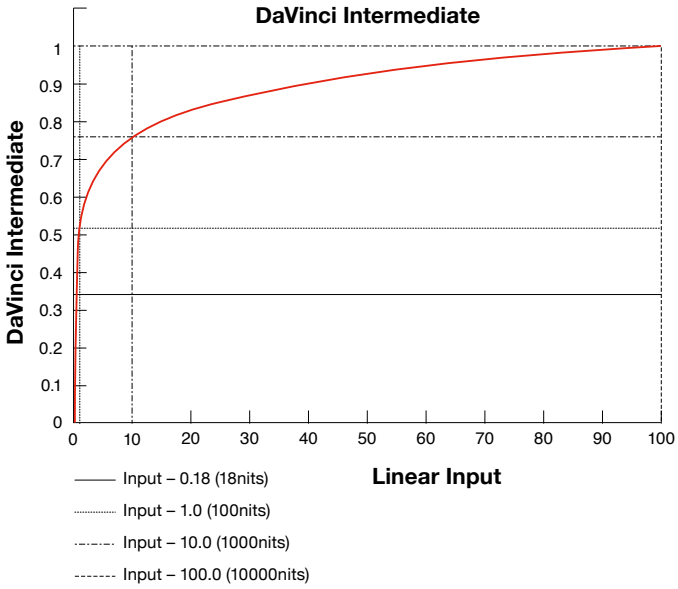
Per configurare velocemente le impostazioni del progetto la prossima volta che lavori con contenuti immersivi, apri la finestra delle impostazioni del progetto, clicca il menù delle opzioni e scegli il predefinito.

10 Clicca **Salva** per chiudere le impostazioni del progetto.

Informazioni su DaVinci Wide Gamut

DaVinci Wide Gamut è lo spazio colore interno avanzato sviluppato da Blackmagic Design per DaVinci Resolve, che offre un ambiente incredibilmente ampio e preciso per il color grading e l'elaborazione delle immagini. In combinazione con la gamma DaVinci Intermediate, forma uno spazio di lavoro unificato che semplifica la gestione del colore su diversi formati di ripresa, garantendo una mappatura uniforme della tonalità e della saturazione dall'ingresso all'uscita. Questa ampia gamma dinamica è ideale per lavorare con l'HDR e i contenuti immersivi, e permette ai coloristi di regolare con la massima precisione senza il rischio di bande indesiderate o clipping dei toni quando si masterizza in formati ad alta fedeltà.





Elaborazione Apple Immersive

Nella barra laterale delle impostazioni del progetto, la sezione Apple Immersive ti permette di impostare ILPD (Immersive Lens Profile Data) personalizzati per modificare la calibrazione e la proiezione di contenuti immersivi.

1. Seleziona Apple Immersive nella barra laterale delle impostazioni del progetto.
2. Clicca **Open Calibration Folder**.
3. Aggiungi o aggiorna un file ILPD custom.
4. Clicca **Update Lists** per aggiornare i contenuti ILPD accessibili nella Biblioteca.
5. Clicca **Salva** per chiudere le impostazioni del progetto.
6. Nella Biblioteca fai clic col pulsante destro su una clip immersiva.
7. Nel menù contestuale vai alla voce **Apple Immersive Calibration** e scegli tra ILPD sorgente **No Apple Immersive Calibration File Selected**, ILPD custom o ILPD gap.

Di default, la cartella Calibrazione ILPD custom si trova nella cartella ProIM, accanto alla cartella LUT di default.

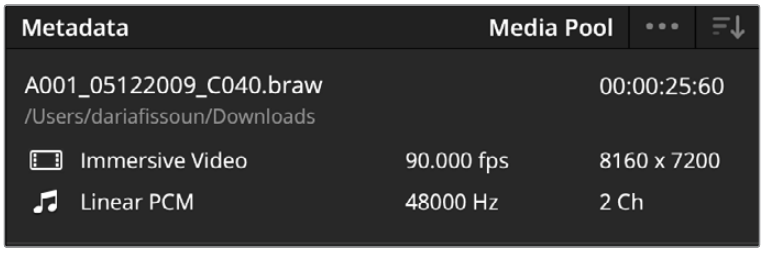
Acquisizione

Quando si acquisisce materiale immersivo RAW, sia i metadati degli occhi che il resto dei metadati associati accompagnano la clip BRAW importata.

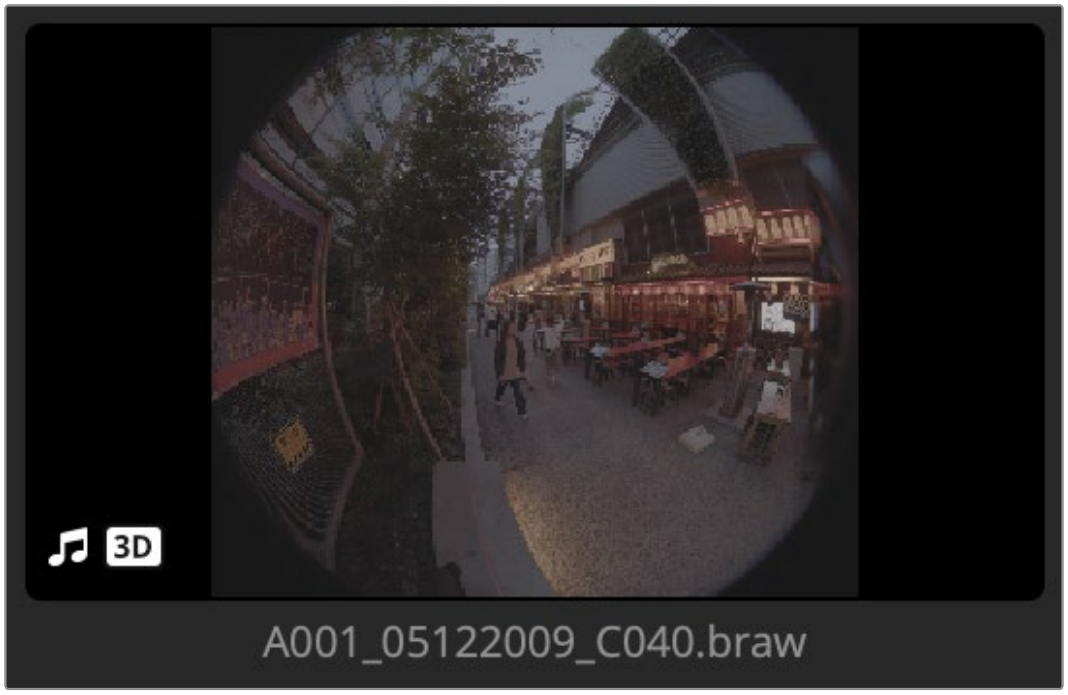
1. Apri la pagina Media.
2. Nel pannello Archivio cerca la posizione dei contenuti video immersivi.
3. Trascina i file nella Biblioteca in basso.

SUGGERIMENTO Per mantenere la struttura della cartella originale, trascina le cartelle dal pannello Archivio nella barra laterale della Biblioteca. Quando si trascinano nel Bin master (o nei bin che hai creato), le cartelle e le sottocartelle vengono mantenute.

Quando viene selezionata una clip video Apple Immersive, il parametro codec nell'intestazione del pannello Metadati indicherà che si tratta di un video immersivo, specificandone anche frame rate nativo e risoluzione.



Inoltre comparirà un'icona 3D nell'angolo in basso a sinistra della miniatura del video per confermare che è una clip 3D.



NOTA Quando trascini per la prima volta file immersivi nella Biblioteca, una finestra di dialogo ti chiederà se intendi mantenere o cambiare le impostazioni. Se scegli di cambiarle, le impostazioni di progetto adotteranno automaticamente il frame rate e la risoluzione del materiale sorgente.

Montaggio

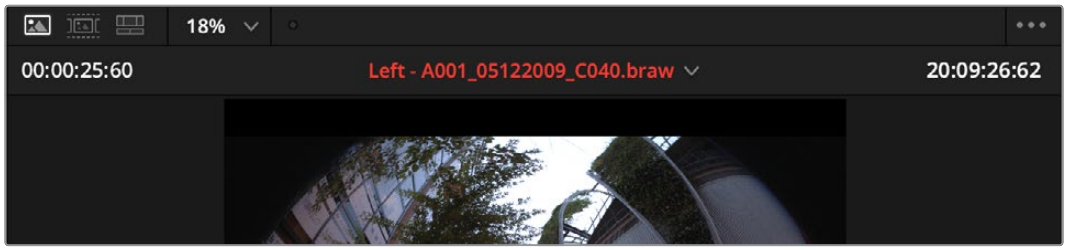
Quando lavori con contenuti immersivi, le azioni e gli strumenti di montaggio restano perlopiù le stesse di quelle dei formati video 2D convenzionali. Le istruzioni fornite qui sotto descrivono funzioni o azioni speciali dei flussi di lavoro immersivi.

Assemblaggio

Quando visualizzi in anteprima le clip durante l'assemblaggio della timeline, puoi cambiare l'occhio (destro o sinistro) che vedi nel visore sorgente. A parte questo, le procedure di assemblaggio della timeline sono le stesse dei normali software di montaggio non lineare.

1 Apri la pagina **Edit**.

2 Fai doppio clic su una clip nel visore sorgente a sinistra. L'etichetta in cima al visore indica quale occhio stai osservando.



3 Per cambiare l'occhio del visore, clicca col pulsante destro destro la clip sorgente nella Biblioteca e seleziona **3D stereoscopico** > **Modalità 3D stereoscopico** > **Occhio sinistro/Occhio destro/Entrambi gli occhi**.

SUGGERIMENTO Puoi anche posizionare il cursore sulle miniature nella Biblioteca per vedere l'anteprima delle clip nel visore sorgente se hai attivato l'opzione Anteprima istantanea nelle impostazioni del visore sorgente.

4 Premi i tasti di scelta rapida I e O per determinare un intervallo nel visore sorgente.

5 Clicca su **Inserisci clip** nella barra degli strumenti (o usa la scorciatoia con il tasto F9) per inserire le clip nella timeline.

SUGGERIMENTO Puoi anche trascinare le clip dalla Biblioteca al pannello della timeline.

6 Usa il mouse (in modalità Selezione o Ritaglio) per ritagliare le clip nella timeline.

Sfondi

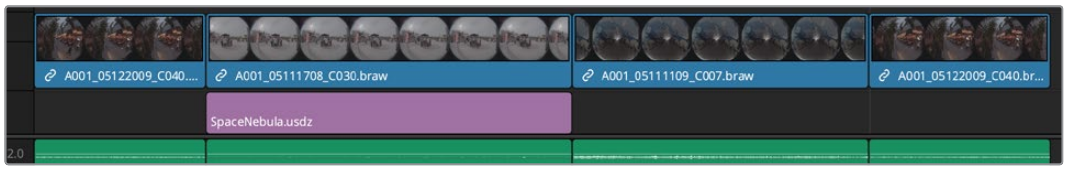
Quando selezioni **Enable Apple Immersive workflow** nelle impostazioni di progetto, compare automaticamente una traccia Sfondi **BD** (dall'inglese Backdrop) sotto la traccia Video 1.

Gli sfondi interessano l'area che sarà visibile allo spettatore quando volge lo sguardo al di là del perimetro dei contenuti video immersivi, solitamente limitati a un campo visivo da 180/210 gradi.

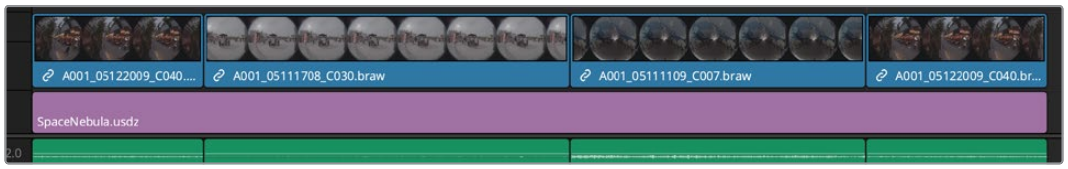
Quest'area è nera di default, ma puoi inserire immagini di sfondo personalizzate (in formato .usdz) utilizzando la traccia Sfondi.

1 Trascina un file di sfondo .usdz nella traccia Sfondi per aggiungerlo alla timeline.

La durata delle immagini di sfondo deve combaciare con i confini della clip nelle tracce video. Un singolo file di sfondo può coprire una clip, diverse clip o l'intera timeline, ma non può tagliare in due una clip immersiva .BRAW (noto anche come "taglio a L").



Una clip - corretto ✓

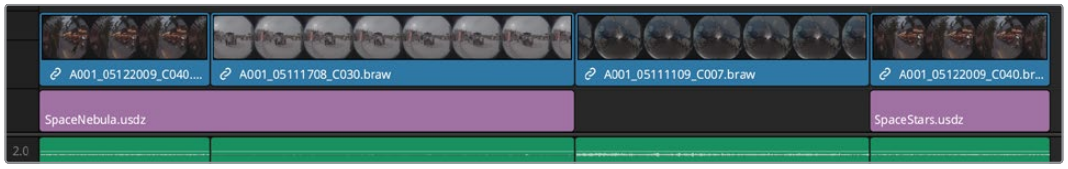


Diverse clip - corretto ✓



Taglio a L - errato ✗

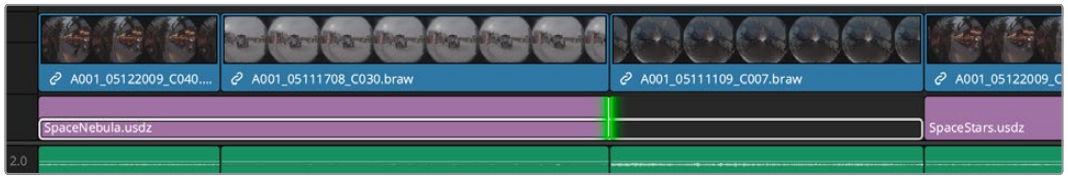
Puoi scegliere di avere diversi file di sfondo all'interno di una singola timeline o di non aggiungere alcuna immagine per lo sfondo nero di default.



Immagini e intervalli diversi - corretto ✓

2 Usa i normali controlli di taglio per regolare la lunghezza del file di sfondo.

Trascina l'estremità del file .usdz (in modalità Selezione o Ritaglio) per regolarne la lunghezza.



Puoi anche utilizzare i comandi , (virgola) e . (punto) per aumentare o diminuire il taglio un fotogramma alla volta.

SUGGERIMENTO Per assicurarti che gli sfondi siano sempre allineati agli edit di video immersivo, attiva la modalità Calamita nella barra degli strumenti della pagina Edit.

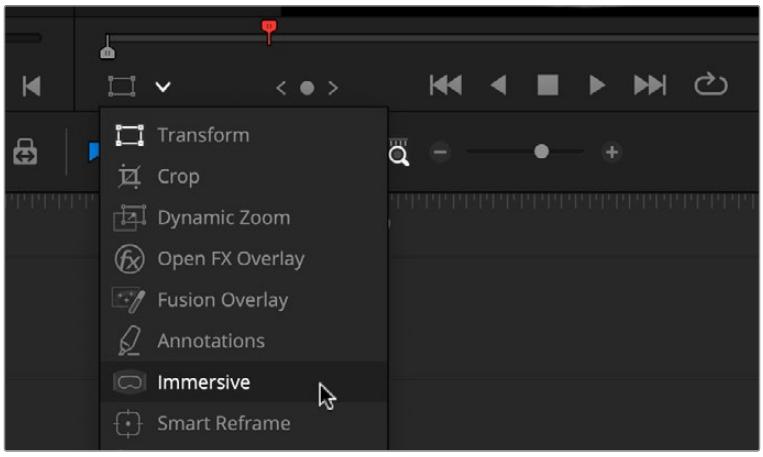
Utilizzare il visore

Di default le clip immersive vengono visualizzate nei visori di DaVinci Resolve in uno spazio circolare. La calibrazione della lente utilizzata per proiettare l'immagine deriva dalla camera stessa ed è unica per ogni clip.



Hai anche la possibilità di scegliere la modalità di visualizzazione, in LatLong o in una finestra Viewport.

1. Nell'angolo in basso a sinistra del visore della timeline, clicca sulla freccia rivolta verso il basso accanto all'icona dei controlli in sovrimpressione.
2. Seleziona **Immersive**.



Sopra il visore comparirà una barra degli strumenti con nuove opzioni di visualizzazione.

3. Clicca su **LatLong** per visualizzare una proiezione equirettangolare dell'immagine. Questa modalità distorce l'immagine per farla somigliare di più alle proporzioni dei file 2D tradizionali, così da semplificare alcune procedure di post come il tracciamento e il compositing.



4 Clicca su **Viewport** per vedere un ritaglio centrale della scena immersiva in un fotogramma quadrato. Questa modalità ti aiuta a vedere chiaramente i dettagli della scena con le proporzioni giuste su un monitor standard, anche se devi fare attenzione a eventuali particolari tagliati fuori dal riquadro.

Puoi utilizzare i controlli di trasformazione nella barra degli strumenti immersivi in alto per modificare l'inclinazione e la panoramica dell'immagine stereoscopica nell'inquadratura, oltre a regolare il campo visivo orizzontale e verticale.



Transizioni

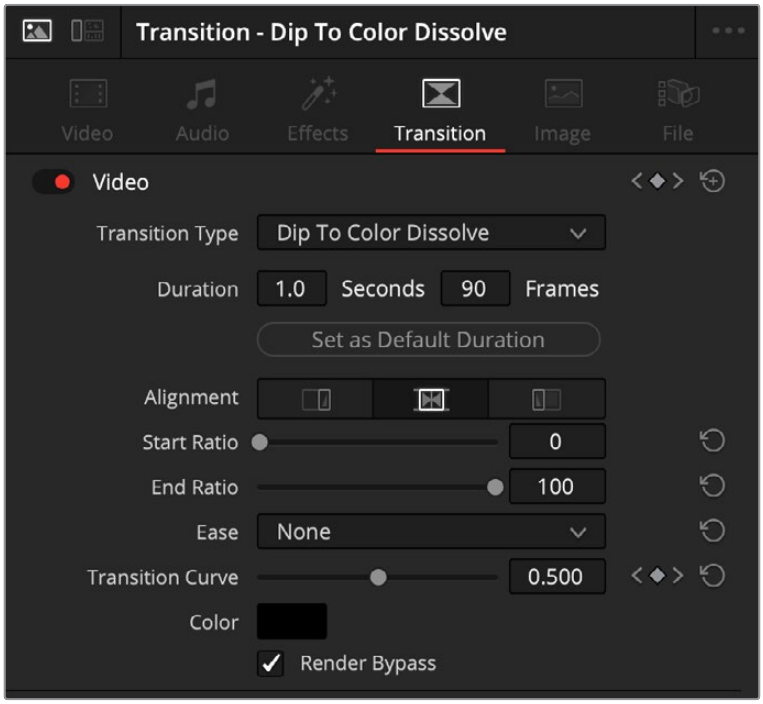
Per effettuare una transizione tra due clip di video immersivo, puoi optare per uno stacco netto, usare le maniglie ai lati della clip per inserire effetti di assolvenza e dissolvenza oppure utilizzare l'effetto Dissolvenza a colori. Le transizioni non vanno integrate nei filmati, ma piuttosto vengono renderizzate da Apple Vision Pro durante la riproduzione.

1. Apri il pannello **Effetti**.
2. Nella barra laterale, vai alla voce **Strumenti** > **Transizioni video**.
3. Trascina la transizione **Dissolvenza a colori** su un punto di taglio tra due clip.



NOTA: Devi avere abbastanza spazio oltre ai punti di taglio di entrambe le clip per applicare la transizione. Se questo spazio (tra le "maniglie") è insufficiente, la transizione non verrà applicata.

4. Nel pannello **Esplora**, vai alla tab **Transizioni** e spunta la casella **Render Bypass** per assicurarti che la dissolvenza sia renderizzata correttamente da Apple Vision Pro.
5. Puoi utilizzare il resto dei parametri nella finestra Esplora della Dissolvenza a colori per modificarne durata, allineamento, velocità di animazione e colore della transizione.



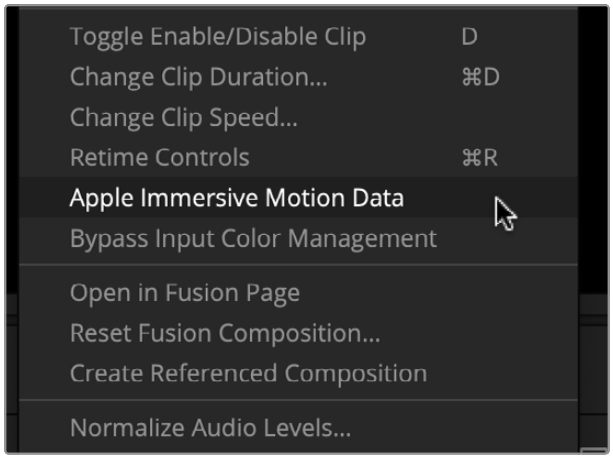
Apple Immersive Motion Data

Quando si lavora con contenuti immersivi, lo spettatore può essere disorientato da movimenti improvvisi e repentini della camera. Per garantire il massimo comfort dell'utente, evita di usare riprese con panoramiche o inclinazioni veloci, movimenti rotatori o orizzonti non allineati.

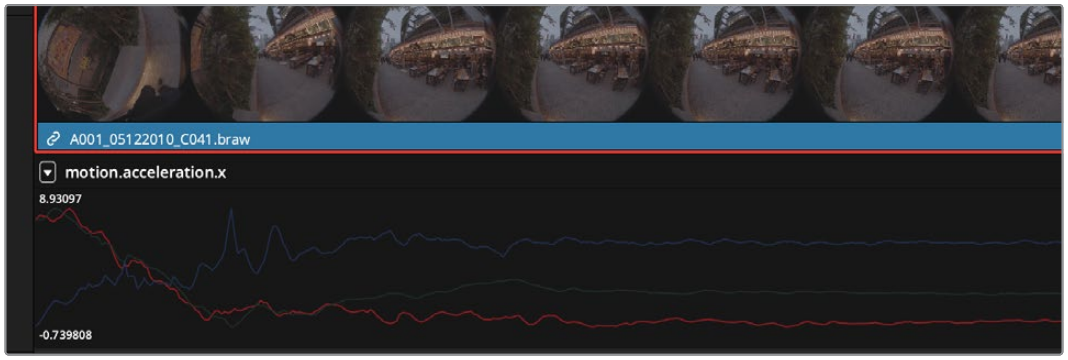
Gli Apple Immersive Motion Data mostrano grafici con valori di accelerazione x, y, z e di posizione giroscopica per visualizzare in maniera obiettiva i punti in cui i movimenti della camera possono risultare troppo accentuati. Chi lavora al montaggio può sfruttare queste informazioni per evitare di usare sezioni di filmati che possono disorientare gli spettatori.

1 Clicca col pulsante destro su una clip immersiva nella timeline.

2 Seleziona **Apple Immersive Motion Data**.

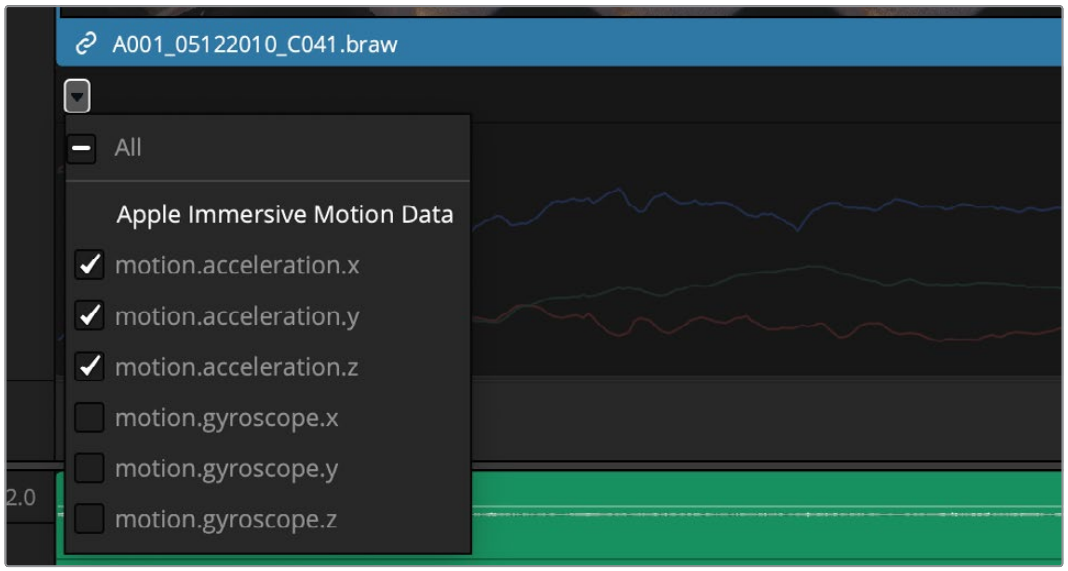


Comparirà un grafico nella traccia video che mostra i tassi di accelerazione x, y, z.



In questo esempio la camera registrava mentre l'operatore la stava preparando per le riprese. I picchi nel grafico indicano i movimenti improvvisi della camera mentre veniva sollevata, ruotata e posizionata sulla spalla dell'operatore. Sarebbe meglio utilizzare la seconda parte del video in cui la camera è in una posizione stabile con i valori più bassi.

3 Nell'angolo in alto a sinistra del grafico, clicca sulla freccia verso il basso.



Puoi mostrare o nascondere i valori in base ai contenuti del filmato e al tipo di movimento della camera.

Titoli

Nella pagina Edit puoi aggiungere del testo in una scena immersiva usando qualsiasi titolo nel pannello Effetti: Testo, Testo+, MultiText, Scorrimento e qualunque titolo Terzo inferiore.

Quando viene posizionato sopra una clip immersiva, il generatore di titoli userà gli ILPD (Immersive Lens Profile Data) per distorcere con precisione il testo in linea con la distorsione della lente nella scena.

In caso non sia disponibile alcun ILPD (per esempio quando si aggiunge un testo sopra uno sfondo nero o una scena immersiva in computer grafica), la distorsione predefinita della lente verrà applicata per assicurare che il testo appaia correttamente distorto quando è visualizzato con Apple Vision Pro.

1 Nel pannello **Effetti**, vai su **Strumenti** > **Titoli**.

2 Trascina il generatore di testo sulla timeline. Posizionalo su una traccia video sopra la clip immersiva per incorporarlo nella scena.



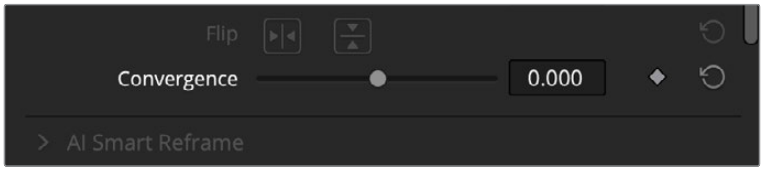
Come nel caso degli sfondi, i generatori di testo non possono dividere a metà le clip, dato che ogni clip è renderizzata con un'immagine spaziale singola e i titoli sovrapposti rischiano di essere distorti tra una clip e l'altra.

3 Ritaglia il titolo fino al punto (o punti) di montaggio della clip.

4 Usa il pannello Esplora per scrivere e formattare il testo del titolo.

5 Vai alla tab **Impostazioni** e sposta lo slider **Convergenza** per posizionare il testo nello spazio 3D.

Se sposti lo slider verso sinistra il titolo si allontana nello spazio, mentre se sposti lo slider verso destra lo avvicini.



La funzione Convergenza si osserva in modo ottimale si monitorando con il visore Apple Vision Pro. Se utilizzi un altro metodo di monitoraggio, puoi cambiare il modo in cui la convergenza di testo appare nel visore modificando la modalità Vista del livello video sottostante nella pagina Color.

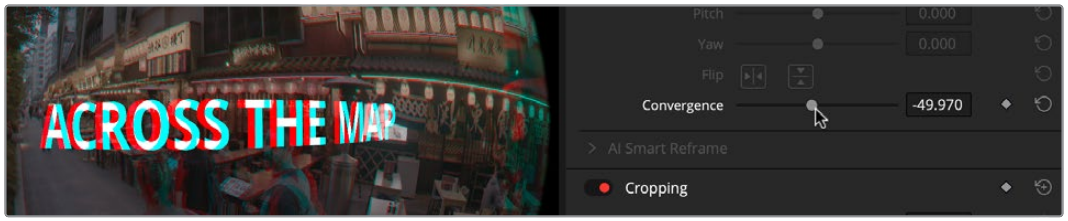
6 Vai alla pagina **Color**.

7 Apri la palette **3D**.

La colonna **Vista al centro** della tavolozza ti permette di modificare lo standard stereoscopico utilizzato per proiettare i contenuti immersivi.

8 Seleziona la tab **Stereo**.

9 Alla voce Uscita, seleziona **Anaglifo (colori)**.



Ora modificare il parametro Convergenza del generatore di testo nel pannello Esplora della pagina Edit risulterà più intuitivo se visualizzato con occhiali anaglifici (o anacromatici).

Considerazioni creative

Questi sono gli albori delle produzioni immersive Blackmagic URSA Cine Immersive e Apple Vision Pro, e incoraggiamo i filmmaker a sperimentare questo nuovo formato per identificare le procedure ottimali, stabilire nuove regole e scoprire metodi di lavoro finora impensabili nelle produzioni cinematografiche tradizionali.

Tenendo a mente questa considerazione, sono già stati identificati alcuni aspetti delle produzioni immersive per assicurare il massimo coinvolgimento e comfort visivo del pubblico:

1 Inizia il montaggio in fase di pre-produzione. Per via delle particolari considerazioni relative al comfort del pubblico con la durata delle riprese e con il movimento, è necessario che le scene vengano filmate tenendo presente il montaggio finale.

2 Evita o limita l'utilizzo di clip in cui sono presenti movimenti rapidi della camera, poiché tendono a disorientare lo spettatore. È preferibile optare per inquadrature fisse, panoramiche lente e carrellate.

3 Riprendi allo stesso livello del soggetto e a una distanza ragionevole (non troppo ravvicinata), perché anche angolazioni di ripresa estreme o sbilanciate possono provocare disorientamento.

4 Rispetto alle scene in 2D, nei contenuti immersivi è necessario più tempo per assorbire i nuovi ambienti, perciò inserisci tagli più lenti per permettere allo spettatore di adattarsi alla nuova scena.

5 Quando si passa da una clip all'altra, presta attenzione al punto in cui lo spettatore stava guardando nelle riprese precedenti: le azioni importanti non dovrebbero verificarsi nella parte sinistra dell'inquadratura se prima l'attenzione del pubblico era rivolta verso la parte destra della scena.

6 L'audio spaziale gioca un ruolo cruciale nella narrazione e nelle produzioni immersive e va dunque considerato alla stregua di un personaggio.

Compositing VFX

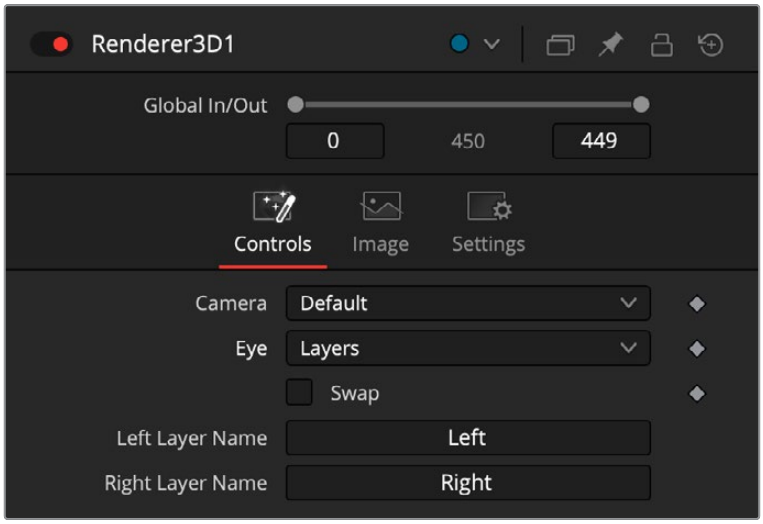
Quando l'opzione **Enable Apple Immersive workflow** è attivata nelle impostazioni di progetto, i parametri dei contenuti immersivi di Fusion permettono flussi di lavoro stereoscopici e compositing conformi a questo tipo di file. I nodi MediaOut offrono opzioni di uscita per l'occhio destro e sinistro, mentre lo strumento Immersive Patcher permette di regolare la proiezione equirettangolare e la composizione.

Nodi MediaOut

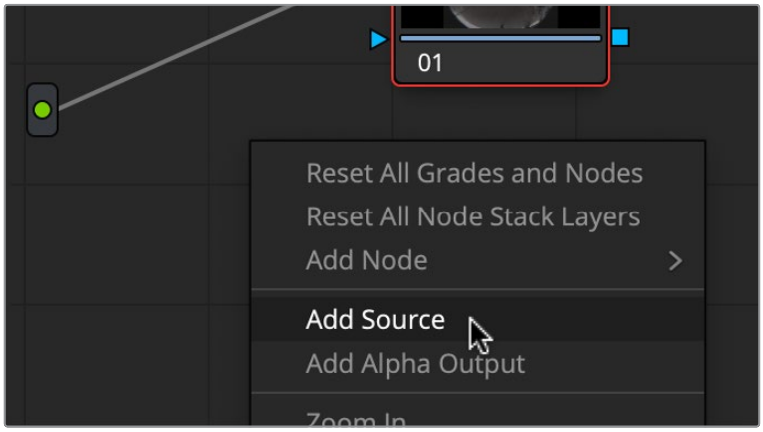
La prima volta che apri la pagina Fusion quando lavori con una clip immersiva, il grafico dei nodi contiene due nodi MediaIn connessi ad altrettanti nodi MediaOut.



Nel pannello Esplora di MediaOut, puoi usare il parametro **Stereo Eye** per instradare l'uscita dell'occhio.



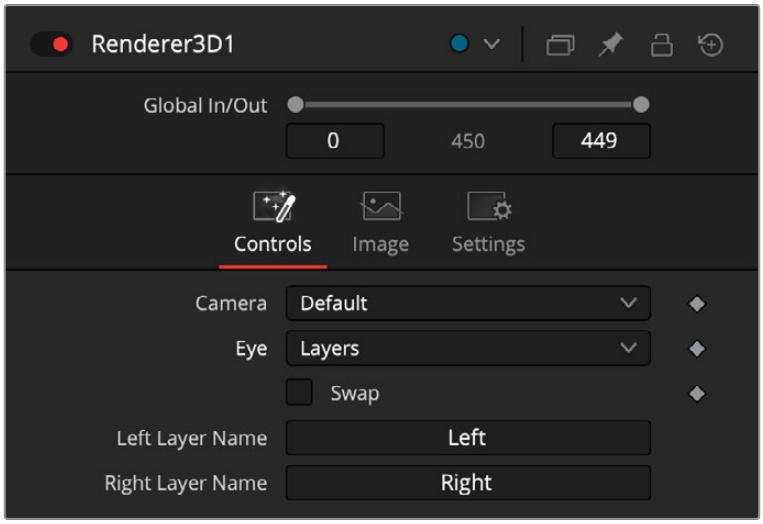
Di default il nodo MediaOut1 è assegnato all'occhio sinistro e MediaOut2 a quello destro. Questa configurazione ti consente di iniziare subito ad assemblare composizioni e sovrimpressioni grafiche conformi al formato stereo. I due nodi MediaOut hanno come risultato due varianti del segnale inviate all'editor dei nodi della pagina Color per perfezionare ulteriormente l'immagine. Puoi accedere al segnale MediaOut2 nella pagina Color cliccando col pulsante destro nell'editor dei nodi e selezionando **Aggiungi sorgente**.



Se il parametro Stereo Eye è impostato su **Auto**, il nodo MediaOut combina i segnali e li assegna agli occhi corrispondenti nella timeline, in modo da utilizzare il flusso di lavoro multilivello di Fusion per inviare i segnali sinistro e destro a un unico nodo MediaOut.

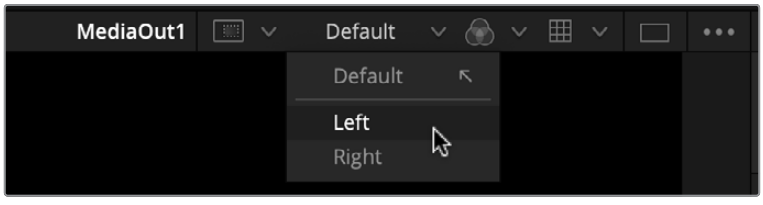
Renderer3D

Quando si lavora con grafiche 3D o CGI in una timeline immersiva, il nodo Renderer3D offre diverse configurazioni per gli occhi e la possibilità di elaborare i livelli del sinistro e del destro.



In questo modo è possibile realizzare complessi titoli 3D o intere scene in CGI e trasferirle direttamente sulla timeline stereoscopica di un progetto immersivo.

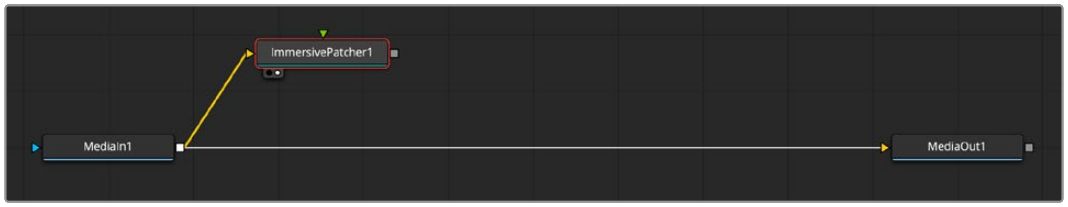
Quando lavori con questa configurazione puoi usare il menù a discesa nell'angolo in alto a destra dei visori di Fusion per alternare la visuale con l'occhio destro o sinistro della tua composizione.



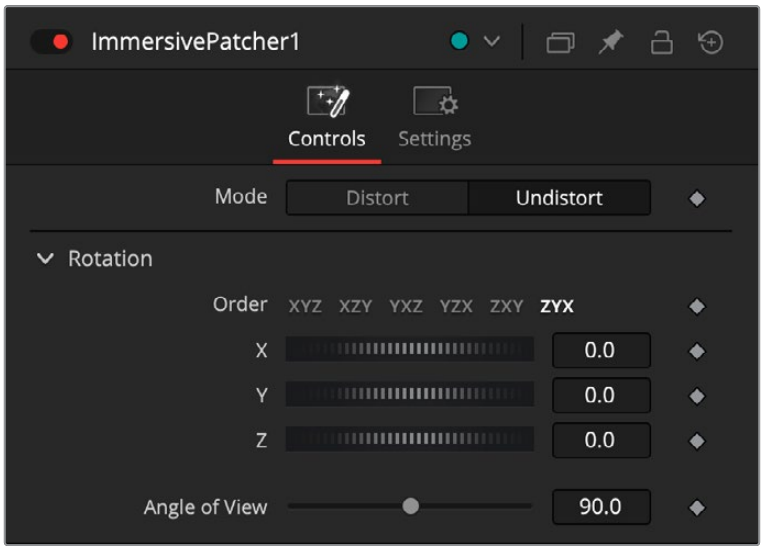
Compositing con Immersive Patcher

Lo strumento Immersive Patcher ti consente di “appiattire” temporaneamente la distorsione di un segmento di una clip immersiva così da facilitare tracciamento, pittura e compositing su una superficie piana. Una volta finito, usa un secondo Immersive Patcher per distorcere nuovamente il segnale immersivo e integrare i nuovi elementi composti nella scena.

1 Aggiungi **Immersive Patcher** alla struttura a nodi.



2 Nel pannello Esplora clicca **Undistort**.



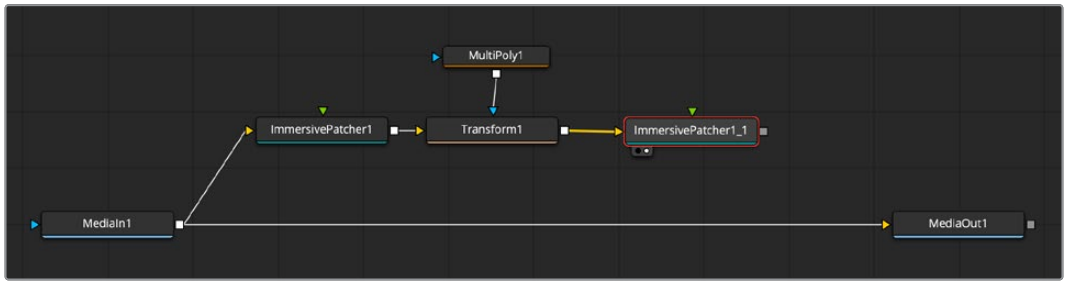
3 Usa i parametri di rotazione X, Y e Z per scegliere una sezione dell’immagine immersiva da appiattire.

4 Effettua il compositing come con un normale segnale video 2D, ovvero rimuovi oggetti indesiderati con metodi di pittura standard, estendi il set campionando sezioni dell’inquadratura con trasformazioni e poligoni, applica matte, testo, e livelli con video o immagini esterne per sostituire insegne o schermi, e altro ancora.

Una volta terminato, devi riportare l’immagine appiattita nella distorsione immersiva originale.

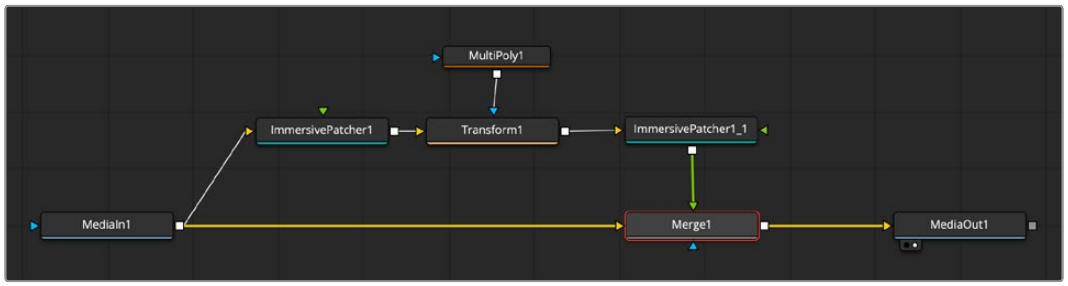
5 Copia l’Immersive Patcher per mantenere la rotazione e le informazioni dell’angolo di visione originali.

6 Incolla l'Immersive Patcher dopo i nodi compositi e connettilo alla fine della struttura a nodi del compositing.



7 Nel pannello Esplora clicca su **Distort**.

8 Accorpa la selezione composta alla clip originale per integrare l'effetto nella scena immersiva.



Se stai lavorando in collaborazione con altri, puoi uscire dalla pagina Fusion e continuare a montare, mixare e correggere la timeline immersiva su DaVinci Resolve.

Se stai preparando la consegna dei file immersivi, renderizza i singoli occhi come file EXR in ACES AP0 per mantenere il maggior numero di proprietà.

Color grading

Questa sezione descrive gli strumenti ottici, di colore e 3D che puoi impiegare quando lavori con i file immersivi nella pagina Color di DaVinci Resolve.

Per via delle particolarità uniche dei contenuti immersivi, in fase di post produzione non sarà possibile utilizzare tutti i consueti strumenti di lavoro. Alcuni strumenti non sono al momento disponibili ma verranno adattati per essere compatibili con i file immersivi.

Riduzione rumore

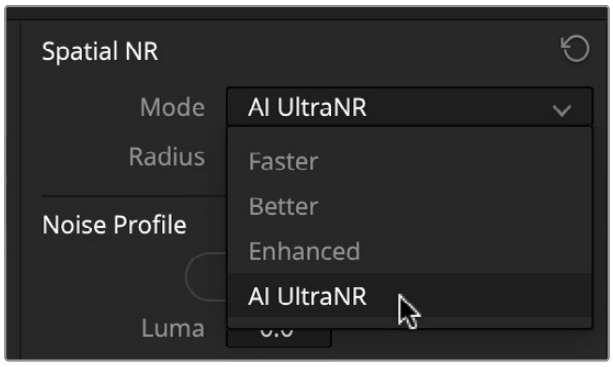
È possibile ridurre il rumore quando si lavora con i video immersivi. Nel menù Effetti di movimento, lo strumento RR Spaziale analizza il modello di rumore per ridurlo il più possibile mantenendo al contempo i dettagli dell'immagine originale.

1 Nella pagina Color, apri il menù **Effetti di movimento**.

Nella colonna **RR Spaziale** seleziona la modalità di riduzione del rumore che intendi usare.

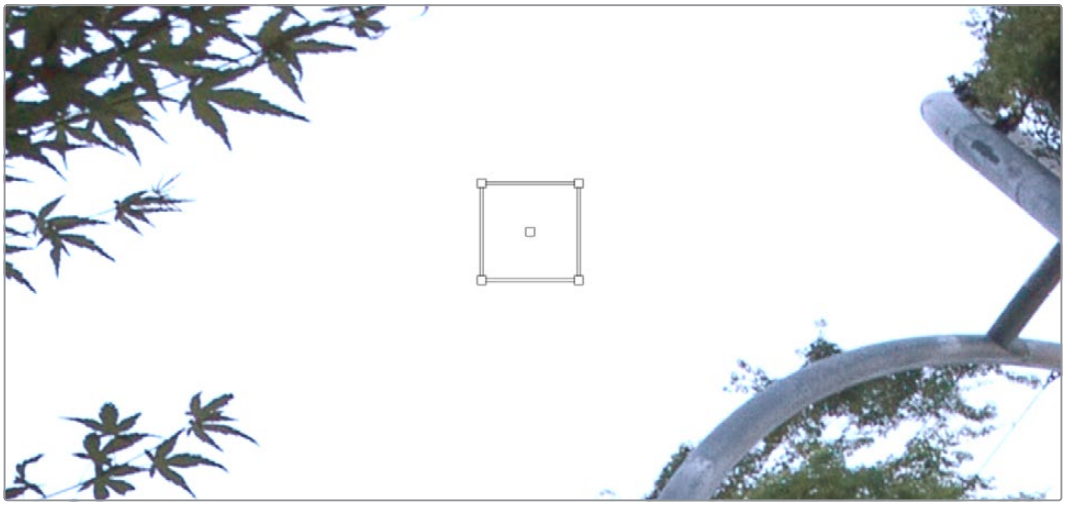
- Le modalità **Veloce**, **Migliore** e **Ottimizzato** analizzeranno il rumore entro un determinato raggio e configureranno la riduzione in base alla soglia impostata. Sono elencati in ordine di complessità computazionale.
- La modalità **AI UltraNR** sfrutta l'intelligenza artificiale per identificare ed estrarre il modello di rumore dal segnale localizzandolo e analizzando la sezione più "piatta" del fotogramma.

2 Imposta la modalità di riduzione del rumore spaziale su **AI UltraNR**.



3 Clicca **Analizza**.

Nel visore comparirà un perimetro che identifica l'area più "vuota" in termini di dati dei pixel nel fotogramma. Senza contenuti visivi tangibili, i dati all'interno del perimetro sono ideali per estrarre il rumore digitale prodotto dal sensore della camera.



Puoi riposizionare e cambiare le dimensioni del perimetro nel visore, ma di solito è sconsigliato perché la funzione AI UltraNR è in grado di rilevare le aree ottimali con una maggiore precisione rispetto ai nostri occhi.

Puoi anche regolare manualmente le Soglie Luma e Chroma nella sezione Noise Profile, ma anche in questo caso AI UltraNR tende a ottenere automaticamente i parametri ottimali di riduzione del rumore.

Grading primario

Il color grading primario funziona allo stesso modo di quello con le normali clip video, perciò puoi usare le consuete pratiche di normalizzazione, bilanciamento e concordanza quando si usano le ruote primarie e Log, la palette HDR e le curve HSL. Sono utilizzabili anche le palette ColorSlice e Distorsore di colore.

Grading secondario

I metodi di selezione basati su pixel, come il Selettore, funzionano in modo intuitivo con i file immersivi.

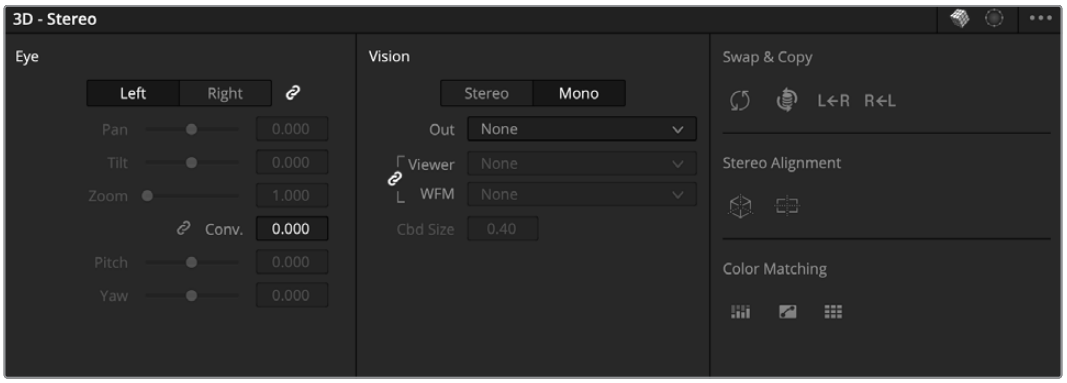
Invece le selezioni basate su oggetti o finestre funzionano con precisione solo in un ambiente piatto equirettangolare (o Viewport). Dunque la maggior parte dei metodi di grading secondario che richiedono una selezione precisa non sono attualmente disponibili, tra cui le Power Window (dunque anche la palette Tracciatore) e lo strumento Magic Mask.

Dimensioni

Non sono disponibili controlli di ridimensionamento (come ad esempio zoom e panoramica) per i contenuti immersivi.

Palette 3D

La palette 3D della pagina Color permette di gestire una grande varietà di formati grafici e di camere 3D. A differenza degli altri formati stereoscopici, non c'è bisogno di regolare l'allineamento degli occhi perché le lenti, la lunghezza focale e il diaframma di Blackmagic URSA Cine Immersive sono fisse.



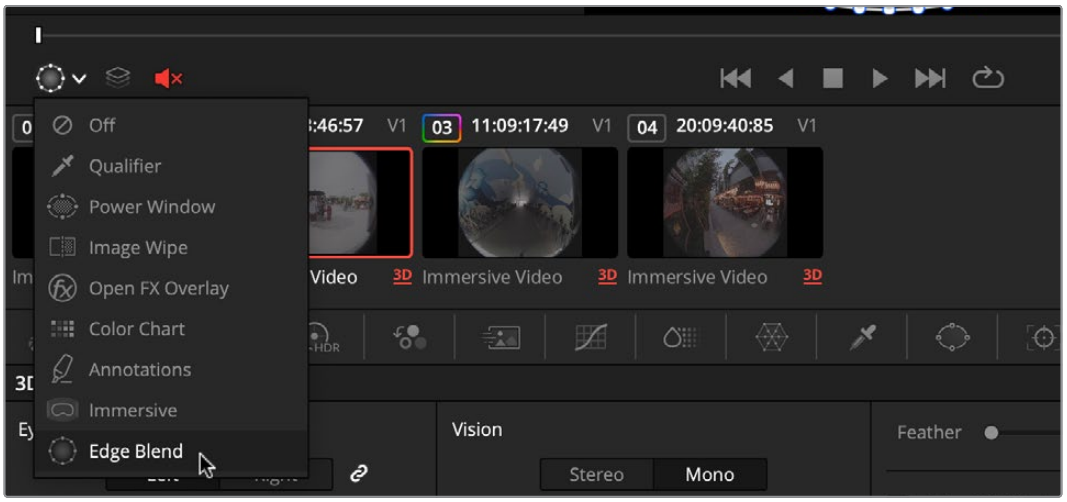
La modalità Edge Blend è disponibile solo per i contenuti immersivi e ti permette di regolare e sfumare il perimetro del fotogramma video immersivo, così da poter nascondere le attrezzature visibili nell'inquadratura e limitare il campo visivo dell'osservatore. Al di là di questo bordo, l'osservatore vedrà lo sfondo.

1 Apri la palette **3D**.

2 Nell'angolo in alto a destra passa alla modalità **Edge Blend**.



3 Assicurati che i controlli del visore siano impostati su Edge Blend.



Dovresti vedere un cerchio di punti di controllo intorno al perimetro del fotogramma video, con una maniglia di ancoraggio al centro.

4 Trascina un punto di controllo per cambiarne la posizione,



oppure cliccaci sopra col pulsante destro per eliminarlo.

5 Nella colonna destra della palette 3D, usa il parametro **Feather** per sfumare il bordo dell'inquadratura.

6 Spunta la casella **visionOS Effect** per garantire che i nuovi lavori del bordo siano elaborati correttamente durante la riproduzione su Apple Vision Pro. Se non selezioni questa casella, il bordo sarà integrato nel file video renderizzato.

7 Nella colonna di sinistra, usa le tab Sinistra e Destra per cambiare l'occhio mostrato nel visore.

Noterai che diversi controlli della palette 3D non sono disponibili, questo perché non si possono applicare ai contenuti immersivi.

8 Nella colonna centrale, usa le tab **Stereo** e **Mono** per mostrare la clip video immersiva con un visore singolo o un visore doppio.

Gestione del colore CST

L'effetto Trasformazione di spazio colore (CST) sui nodi nell'editor dei nodi della pagina Color consente una mappatura accurata dei colori e un controllo personalizzato sulla luminanza massima in nit dell'output della timeline.

Il flusso di lavoro consigliato prevede quattro nodi CST:

1. INPUT CST – dalla camera allo spazio colore.

   La correzione colore avviene tra questi due nodi.

2. OUTPUT CST – dallo spazio colore al display HDR del Mac.
3. MONITOR CST – da 1000 nit a 250 nit per monitoraggio HDR.
4. AVP CST – da 250 nit a 108 nit per monitoraggio e consegna con Apple Vision Pro.



Mappatura timeline CST

Innanzitutto devi mappare lo spazio colore della camera sorgente allo spazio colore della timeline "di lavoro". Blackmagic URSA Cine Immersive opera nel gamut di colore Blackmagic Design Wide Gamut Gen 4/5 con la curva log DaVinci Intermediate.

1. Apri il pannello **Effetti**.
2. Trascina l'effetto **Trasformazione di spazio colore** su un nodo vuoto (01) nell'editor dei nodi.

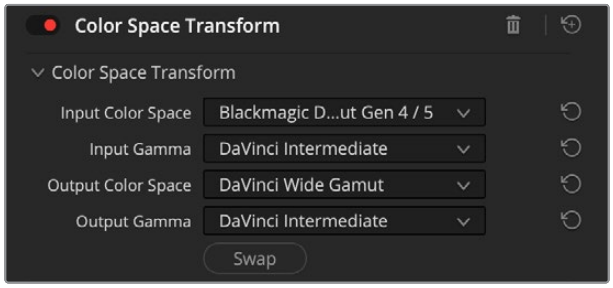
   Quando si effettua la mappatura dallo spazio colore di ingresso allo spazio colore della timeline, questo dovrebbe essere uno dei primi nodi presenti nella struttura a nodi, prima di iniziare qualunque operazione di grading.

   In caso tu stia mappando lo spazio colore di ingresso direttamente allo spazio colore di monitoraggio o consegna, questo nodo dovrà essere uno degli ultimi della struttura a nodi, dopo le operazioni di grading.

3. Rinomina l'etichetta del nodo 01 in **INPUT CST**.

> **NOTA** Le etichette dei nodi usate in queste istruzioni sono state scelte per motivi di chiarezza. Puoi ovviamente scegliere la denominazione che preferisci o evitare del tutto di usarle.

4. Imposta **Spazio colore ingresso** su **Blackmagic Design Wide Gamut Gen 4/5**.
5. Imposta **Gamma ingresso** su **DaVinci Intermediate**.
6. Imposta **Spazio colore uscita** su **DaVinci Wide Gamut**.
7. Imposta **Gamma uscita** su **DaVinci Intermediate**.



8 Clicca col destro sul nodo INPUT CST e aggiungi un nuovo nodo seriale (02).

9 Copia e incolla il primo nodo (01) sul nuovo nodo (02).

10 Rinomina l'etichetta del nodo 02 in OUTPUT CST.

È necessario che la gamma e lo spazio colore dell'uscita siano identici a quelli dell'ingresso in questo nuovo nodo seriale per mantenere uniforme lo spazio colore della timeline della struttura a nodi.

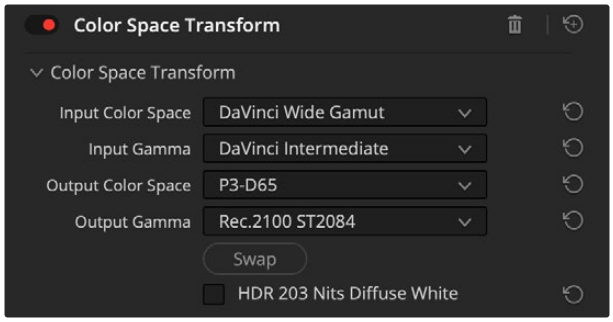
11 Premi il pulsante **Inverti** per invertire i parametri di ingresso e di uscita.

Ora i parametri **Gamma ingresso** e **Spazio colore ingresso** sono giusti per questo nodo CST.

12 Imposta **Spazio colore uscita** su **P3-D65**. Questo è l'attuale gamut predefinito dei display Mac.

13 Imposta **Gamma uscita** su **Rec.2100 ST2084**, che rappresenta lo standard per un segnale a 1000 nit.

Tutte le operazioni di grading avvengono tra questi due nodi. Tuttavia, se stai lavorando con una gestione del colore basata sui nodi, puoi anche effettuare operazioni al di fuori dei parametri della gestione del colore della timeline, una funzione esclusiva sfruttata da molte coloriste e coloristi per applicare specifiche operazioni di grading, effetti o elementi di terzi.



Profili ICC

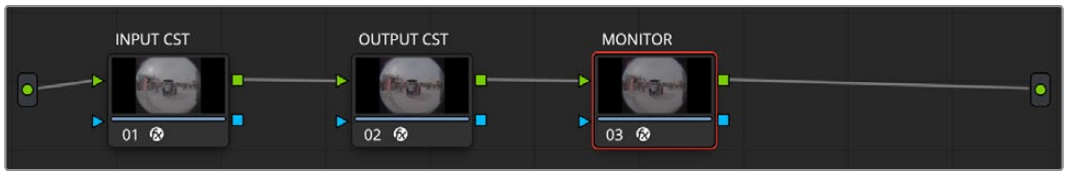
Per accertarti ulteriormente che il segnale di Blackmagic URSA Cine Immersive sia mappato correttamente sul tuo display Mac, apri le Impostazioni di Sistema > Schermi e controlla che il profilo colore sia impostato su Apple XDR Display (P3-1600 nits).

Mappatura CST per monitorare il grading

Puoi utilizzare CST aggiuntivi per limitare temporaneamente i valori massimi di output di nit per il monitoraggio durante il grading.

1 Aggiungi un nuovo nodo seriale (03) dopo il nodo OUTPUT CST (02) di gestione del colore.

2 Rinomina l'etichetta del nodo 03 in **MONITOR**.



3 Aggiungi l'effetto **Trasformazione di spazio colore** al nodo MONITOR (03).

I valori di ingresso devono combaciare con il segnale in uscita del nodo CST precedente.

4 Imposta **Spazio colore ingresso** su **P3-D65**.

5 Imposta **Gamma ingresso** su **Rec.2100 ST2084**.

L'unico scopo di questo nodo CST è quello di mappare la luminanza dell'immagine su un valore minore di nit, dunque i parametri di ingresso e di uscita possono restare gli stessi.

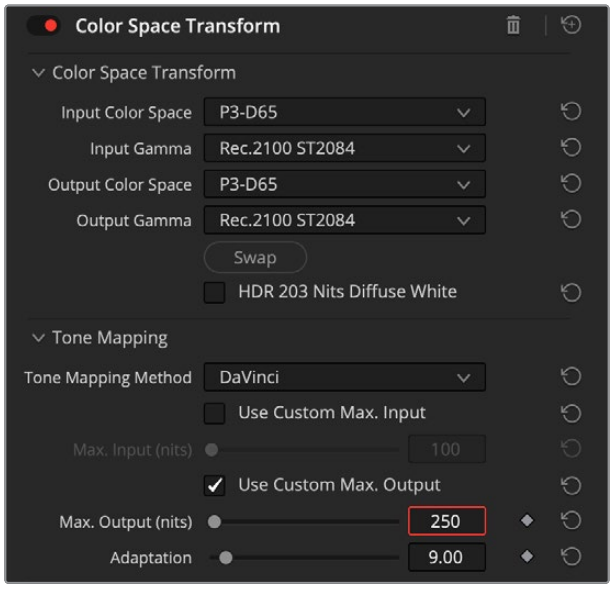
6 Imposta **Spazio colore uscita** su **P3-D65**.

7 Imposta **Gamma uscita** su **Rec.2100 ST2084**.

NOTA In alcune circostanze non è necessario impostare lo spazio colore o la gamma di un CST se ingresso e uscita sono identici (in quanto non avviene alcuna mappatura). In questo caso è necessario perché l'applicazione di una mappatura nit personalizzata è influenzata dal valore e dalla forma della curva log del segnale.

8 Spunta la casella **Personalizza uscita max** .

9 Imposta il parametro **Uscita max (nit)** su **250**.



Ora il segnale da 1000 nit sarà mappato (utilizzando il metodo di mappatura di toni DaVinci) su 250 nit, risultando in un'immagine molto simile al segnale Apple Vision Pro renderizzato su un monitor per il grading (come Flanders XMP310, Sony BVM-HX310, eccetera).

Tuttavia per monitorare il tuo progetto su un dispositivo Apple Vision Pro devi effettuare ancora un'operazione di mappatura dei toni.

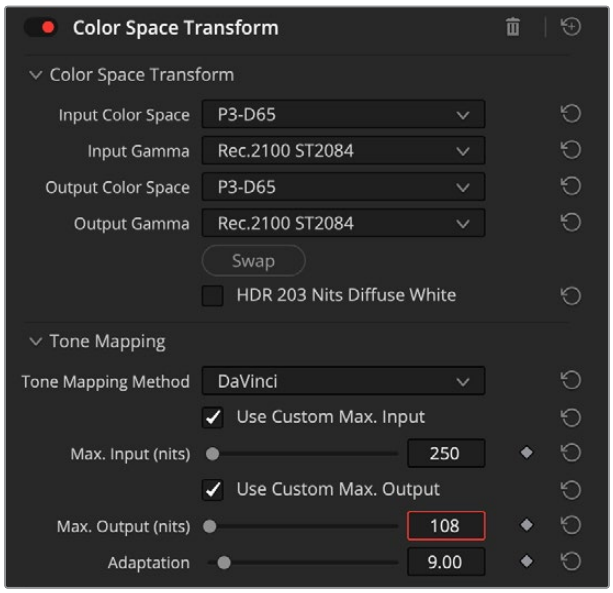
Monitoraggio CST con Apple Vision Pro

Puoi utilizzare un nodo CST finale per consentire il monitoraggio in tempo reale, l'anteprima e la consegna finale ai dispositivi Apple Vision Pro.

1. Aggiungi un nuovo nodo seriale (04) dopo il nodo MONITOR (03).
2. Copia e incolla il nodo MONITOR (03) sul nodo (04). Utilizzerai gli stessi parametri di ingresso e di uscita, modificando soltanto il valore di nit.
3. Rinomina l'etichetta del nodo 04 in **AVP**.



4. Spunta la casella **Personalizza ingresso max**.
5. Imposta il parametro **Ingresso max (nit)** su **250** per interpretare correttamente il segnale mappato dal nodo MONITOR.
6. Imposta il parametro **Uscita max (nit)** su **108**, ovvero il valore massimo di nit per la consegna e il monitoraggio con Apple Vision Pro.



Adesso puoi proseguire con il grading, disattivando i nodi CST se (e quando) necessario in base al tuo flusso di lavoro. Ad esempio, durante la fase iniziale di grading nella quale utilizzerai il tuo monitor di grading esterno, dovresti disattivare il nodo AVP (04) così da assicurarti di vedere correttamente il segnale da 250 nit nel tuo monitor di grading HDR. Poi, quando passi all'anteprima con l'Apple Vision Pro o per la consegna finale, riattiva il nodo AVP (04).

Si possono posizionare i CST nell'editor dei nodi della timeline?

No, al momento la mappatura dei toni delle clip fa affidamento su una precisa applicazione del parametro Adattamento della Trasformazione di spazio colore, che deve essere eseguita una clip alla volta.

Audio

Sia Apple Vision Pro sia DaVinci Resolve supportano i formati audio ambisonico e spaziale, AVP (Apple Vision Pro) e AIA (Apple Immersive Audio).

Diversi aspetti della post produzione audio rimangono inalterati quando si lavora con l'audio spaziale. Una delle differenze più grandi è che i tecnici del suono, anziché usare speaker di alta fascia per il monitoraggio, dovrebbero utilizzare cuffie o AirPods per un'immersione spaziale autentica e una migliore riproduzione dell'ambiente sonoro in cui si troverà il pubblico.

Nei mix 2D la maggior parte dei dialoghi viene mixata sui canali centrali, mentre l'audio spaziale deve essere mixato nell'area in cui ha origine il suono. Ducking, equalizzazione e bilanciamento devono essere utilizzati per mixare i file stem per evitare di confondere o bombardare di suoni l'ascoltatore.

Gli elementi spaziali immersivi possono essere combinati con i consueti file stem stereo, in modo che alcuni suoni restino costanti a prescindere dal movimento della testa. Questo può tornare utile per far sì che i suoni extradiegetici, come la colonna sonora e le voci fuori campo, risultino chiari e facili da interpretare.

I suoni ambientali tendono a essere percepiti di meno nell'audio spaziale, perciò spesso devono essere mixati in modo più calcato per essere notati, senza tuttavia disorientare l'ascoltatore.

I riverberi contribuiscono notevolmente al senso di immersione perché gli ascoltatori lo vivono tutti i giorni: ogni suono prodotto all'interno di un ambiente rimbalza sulle superfici che circondano l'ascoltatore, andando a creare un effetto di "blooming". Per tale motivo il riverbero è un elemento che incoraggiamo a inserire durante il design dell'audio spaziale per assicurare un'immersione totale.

Il formato Apple Immersive Audio (AIA) include a sua volta diversi altri formati:

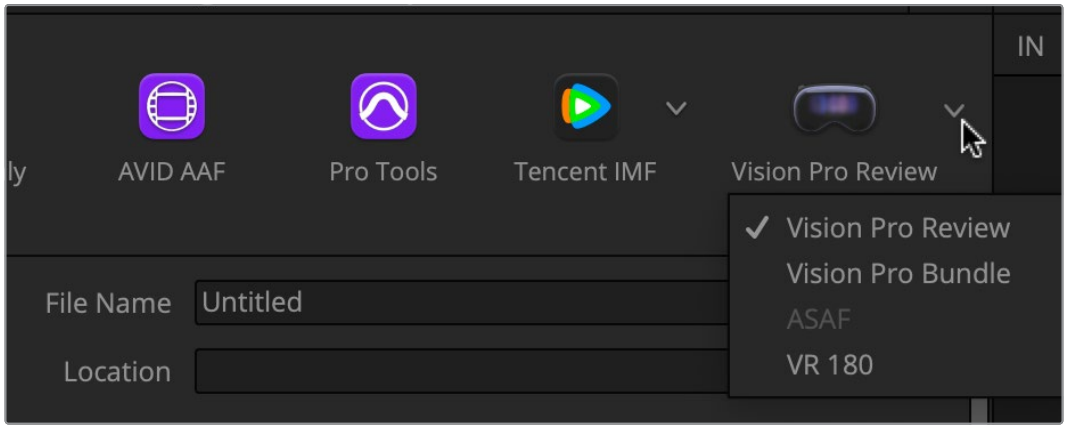
- **Apple Spatial Audio Format (ASAF)** è il formato audio RAW.
- **Apple Positional Audio Codec (APAC)** è il formato .mp4 codificato riproducibile con i dispositivi compatibili, ad esempio gli AirPods, o da un file AIVU con Apple Vision Pro.

Apple Spatial Audio Format (ASAF)

Questo è un formato audio immersivo sviluppato principalmente per le applicazioni in realtà virtuale e aumentata su Apple Vision Pro. ASAF è un formato esclusivo per macOS. È supportato su ProTools (con plugin) e DaVinci Resolve (integrato), e offre funzioni avanzate tra cui il supporto per sorgenti Ambisonics (chiamate "scene") e instradamento tramite assegnazione di bus. Il mixaggio è virtuale, senza bus, e gli utenti possono scegliere tra diversi formati di monitoraggio, tra cui binaurale con tracciamento della testa grazie all'integrazione diretta con Apple AirPod. Il panning è gestito dagli strumenti esistenti di DaVinci Resolve come la finestra di panning, automazione, SpaceView e gli strumenti di panning all'interno del visore, con i controlli secondari disponibili nel pannello Esplora. Questi strumenti permettono un posizionamento spaziale preciso delle sorgenti, tra cui il tracciamento automatico su video LatLong. È possibile esportare i mix ASAF nei formati WAV o MP4 (con i relativi metadati) con l'opzione Audio immersivo > Esporta file master, sebbene al momento non possano essere importati o riprodotti su DaVinci Resolve.

Consegna

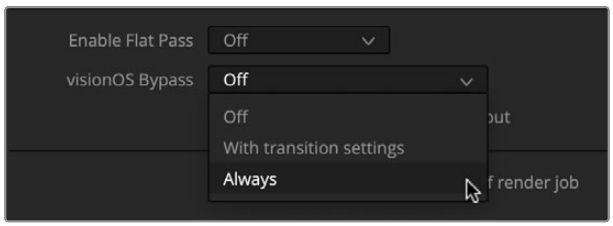
Nelle impostazioni di rendering della pagina Deliver sono presenti due opzioni per renderizzare i file Apple Vision Pro per la revisione e la consegna finale.



- **Vision Pro Review** – crea un piccolo pacchetto MV-HEVC per le anteprime. Apple “condivide” il file generato con Apple Vision Pro tramite Wi-Fi ed è possibile guardarlo nell’app Apple Immersive Video Utility (AIVU), disponibile nell’App Store.
- **Vision Pro Bundle** – crea un grande pacchetto di file ProRes che può essere archiviato o elaborato con Apple Compressor per la riproduzione su Apple TV.
- **ASAF** – renderizza un mix audio ASAF nei formati WAV o MP4.
- **VR180** – genera un file video stereoscopico conforme al VR180.

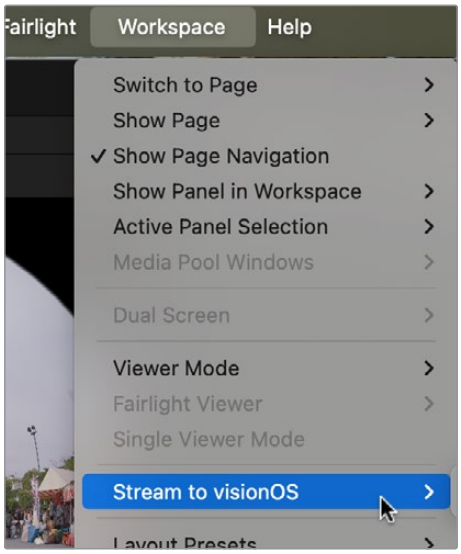
Bypassare le transizioni

Se aggiungi una transizione a una timeline immersiva, sarà pre-renderizzata e trasformata in una singola traccia di riproduzione. Il file XML generato per i contenuti immersivi descrive il passaggio tra questi due flussi renderizzati (filmato, poi la transizione pre-renderizzata, e poi di nuovo il filmato). Nelle impostazioni avanzate del pannello **Impostazioni rendering**, alla voce **visionOS Bypass** puoi scegliere di bypassare le transizioni visionOS selezionando **Always**, applicarle in base al loro stato nella timeline con **With transition settings**, o di includerle così come sono visibili nella timeline con **Off**.



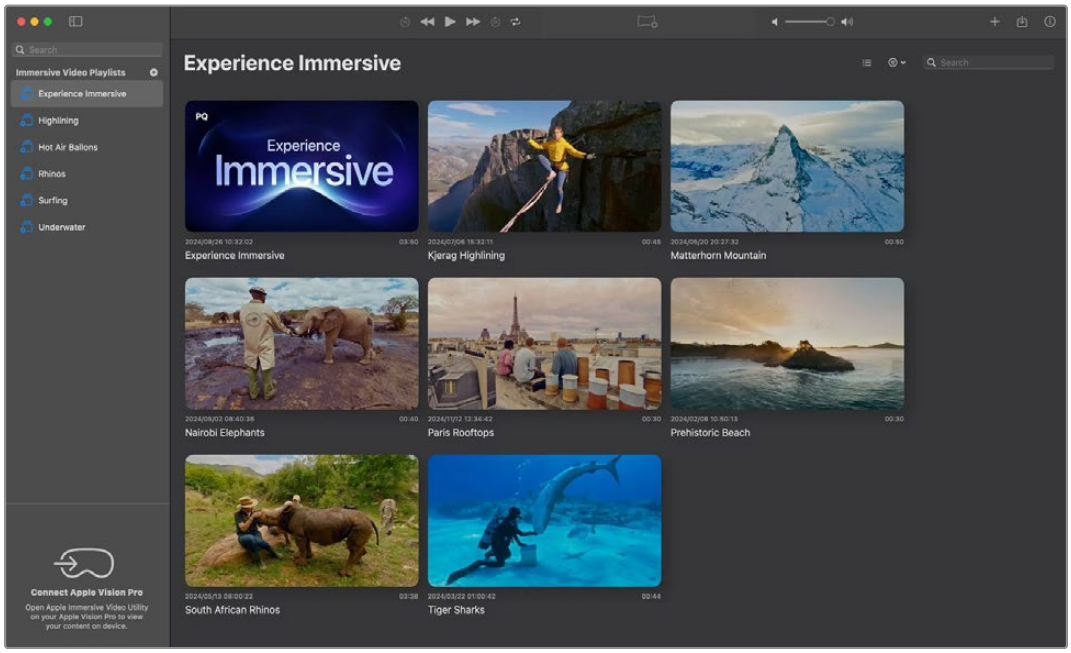
Trasmettere su Apple Vision Pro

È possibile trasmettere in streaming su Apple Vision Pro dalle pagine Edit, Color e Delivery aprendo il menù a discesa **Spazio lavoro** > **Stream to visionOS** e selezionando il tuo dispositivo AVP. Consigliamo di utilizzare una rete wireless veloce (minimo Wi-Fi 5, Wi-Fi 6 consigliato) per una trasmissione senza interruzioni.



Informazioni su About Apple Immersive Video Utility

Questa è una app per macOS e visionOS progettata per facilitare la creazione, gestione e riproduzione di contenuti Apple Immersive Video. È stata realizzata per i filmmaker professionisti e creatori di contenuti immersivi, e consente agli utenti di importare, organizzare, creare pacchetti e rivedere file video stereoscopici in 8K con un campo visivo di 180 gradi e audio spaziale. Supporta la creazione di file AIVU (Apple Immersive Video Universal), permettendo di verificare e di modificare i metadati per i flussi di lavoro di post produzione. Gli utenti possono creare playlist, organizzare e consultare le proprie librerie e perfino trasmettere i contenuti in streaming a uno o più dispositivi Apple Vision Pro.



Developer Information

Camera Control REST API

If you are a software or hardware developer you can build custom applications or leverage ready to use tools such as REST client or Postman to seamlessly control and interact with your compatible Blackmagic camera using Camera Control REST API. This API enables you to perform a wide range of operations, such as starting or stopping recordings, accessing disk information and much more. Whether you're developing a custom application tailored to your specific needs or utilizing existing tools, this API empowers you to unlock the full potential of your Blackmagic camera with ease. We look forward to seeing what you come up with!

NOTE It's important to mention that controlling Blackmagic cameras via REST API relies on the web manager being enabled on each compatible Blackmagic camera. Enable the web media manager in the Blackmagic Camera Setup 'network access' settings for each camera you are controlling.

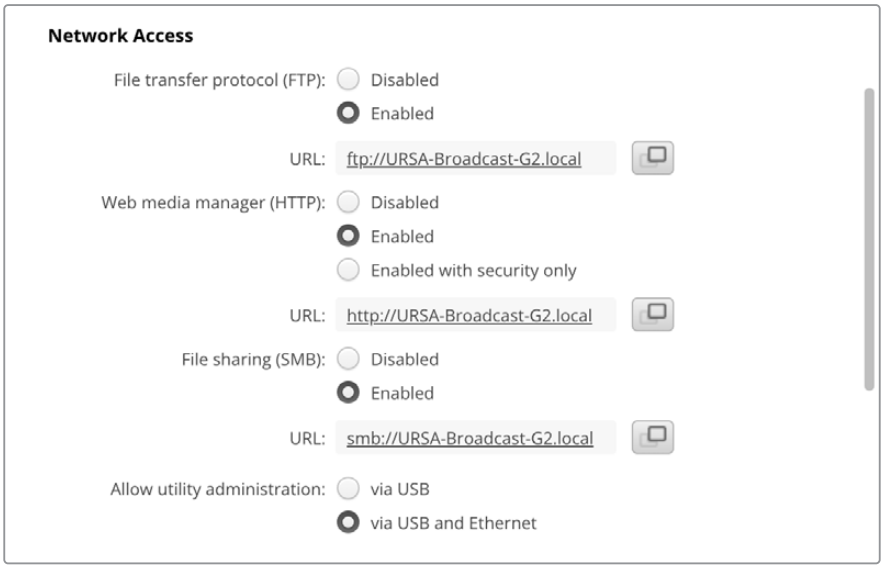
The following Blackmagic cameras are compatible with Camera Control REST API:

- Blackmagic URSA Cine 12K LF
- Blackmagic URSA Cine 17K 65
- Blackmagic URSA Cine Immersive
- Blackmagic PYXIS 6K
- Blackmagic Cinema Camera 6K
- Blackmagic URSA Broadcast G2
- Blackmagic Micro Studio Camera 4K G2
- Blackmagic Studio Camera 4K Plus
- Blackmagic Studio Camera 4K Pro
- Blackmagic Studio Camera 6K Pro
- Blackmagic Studio Camera 4K Plus G2
- Blackmagic Studio Camera 4K Pro G2

Sending API Commands

To send an API command to your camera from a third party application such as Postman, add /control/api/v1/ to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/api/v1/

You can find the Web media manager URL and IP address information in Blackmagic Camera Setup.



The Web media manager URL in Blackmagic Camera Setup

Downloading API's from your Camera

You can download REST API YAML documentation from your camera by adding /control/documentation.html to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/documentation.html

NOTE It's worth noting that changing the camera name in Blackmagic Camera Setup will also change the camera's Web media manager URL.

Livestream Control API

API for controlling Livestreams on Blackmagic Design products.

GET /livestreams/0

Get the livestream's current status.

Response

200 - Livestream's current status.

The response is JSON.

Name	Type	Description
status (required)	string	Possible values are: Idle, Connecting, Streaming, Flushing, Interrupted.
bitrate (required)	integer	Current bitrate (bps).
effectiveVideoFormat (required)	string	Effective video format for the livestream, serialised as a string.
duration	integer	Current stream duration in seconds. Absent if livestream is idle.
cache	integer	Current stream cache usage percentage.

GET /livestreams/0/start

Determine if the livestream is active.

Response

200 - Livestream active status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is active.

PUT /livestreams/0/start

Start the livestream.

Response

204 - Livestream started.

GET /livestreams/0/stop

Determine if the livestream is inactive.

Response

200 - Livestream inactive status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is inactive.

PUT /livestreams/0/stop

Stop the livestream.

Response

204 - Livestream stopped.

GET /livestreams/0/activePlatform

Get the currently selected platform configuration for the livestream.

Response

200 - Livestream active platform configuration.

The response is JSON.

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

PUT /livestreams/0/activePlatform

Set the currently selected platform configuration for the livestream.

Parameters

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

Response

204 - Livestream active platform configuration updated.

400 - Bad Request

GET /livestreams/platforms

Get the list of available platforms.

Response

200 - List of available platforms.

The response is JSON.

Name	Type	Description
	array	List of available platforms names.
[i]	string	Platform name.

GET /livestreams/platforms/{platformName}

Get the service configuration for a platform.

Parameters

Name	Type	Description
{platformName} (required)	string	Name of the platform.

Response

200 - Service configuration for specified platform.

The response is JSON.

Livestream platform service configuration.

Name	Type	Description
platform (required)	string	Corresponding platform name.
key	string	Default stream key.
servers (required)	array	List of server configurations.
servers[i]	object	Server configuration.
servers[i].server (required)	string	Server name.
servers[i].url (required)	string	Livestream destination.
servers[i].srtExtensions	array	Miscellaneous tags used for SRT livestreams.
servers[i].srtExtensions[i]	object	Dictionary object mapping SRT tag strings to values.
servers[i].srtExtensions[i][{key}]	string	SRT tag value.
servers[i].group	string	Logical grouping of the server.
profiles (required)	array	List of profile configurations.
profiles[i]	object	Quality configuration.
profiles[i].profile (required)	string	Quality level name.
profiles[i].configs (required)	array	List of video format configurations.
profiles[i].configs[i]	object	Video format configuration for profiles.
profiles[i].configs[i].resolution (required)	string	Video format serialised as a string.
profiles[i].configs[i].fps (required)	string	Frames per second.
profiles[i].configs[i].bitrate (required)	integer	Pixel bitrate (bps).
profiles[i].configs[i].audioBitrate	integer	Audio bitrate (bps).
profiles[i].configs[i].keyFrameInterval	integer	How often a key frame is sent, in seconds.
profiles[i].configs[i].videoCodecs	array	Supported video encoding algorithm/s.

Name	Type	Description
profiles[i].configs[i].videoCodecs[i]	string	Video encoding algorithm. Possible values are: H264, H265.
profiles[i].lowLatency (required)	boolean	If true, fewer frames will be buffered in the livestream.
defaultProfile	string	Quality level name.
credentials	object	Credientials used for RTMP streams.
credentials.username (required)	string	The username part of the creditials. Only used for RTMP streams.
credentials.password (required)	string	Used for RTMP streams, also used as Passphrase for SRT streams.
customizableUrlEnabled	boolean	True when the server URL is customizable.

400 - Bad Request

GET /livestreams/customPlatforms

Get a list of custom platform files.

Response

200 - List of custom platform files.

The response is JSON.

Name	Type	Description
	array	List of custom platform file names.
[i]	string	Custom platform file name.

DELETE /livestreams/customPlatforms

Remove all custom configuration files.

Response

204 - All custom configuration files removed.

GET /livestreams/customPlatforms/{filename}

Get a custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to get.

Response

200 - Custom platform file.

The response is XML.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

404 - Not Found

PUT /livestreams/customPlatforms/{filename}

Update a custom platform file if it exists, if not, create a new file with the given file name.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to update/create.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

Response

204 - Custom platform file created or updated.

400 - Bad Request

DELETE /livestreams/customPlatforms/{filename}

Remove the given custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to be removed.

Response

204 - Custom platform file removed.

404 - Not Found

Clips Control API

API for listing clips on disk.

GET /clips

Get the list of clips on the active disk.

Response

200 - List of clips on the active disk.

The response is JSON.

List of media clips.

Name	Type	Description
clips (required)	array	
clips[i]	object	Media clip.
clips[i].clipUniqueId (required)	integer	Unique ID used to identify this clip.
clips[i].filePath	string	Path to the file relative to the root of a mount.
clips[i].fileSize	integer	Size of file on disk in bytes.
clips[i].codecFormat	object	
clips[i].codecFormat.codec	string	Currently selected codec.

Name	Type	Description
clips[i].codecFormat.container	string	Multimedia container format.
clips[i].videoFormat	object	Video format configuration.
clips[i].videoFormat.name (required)	string	Video format serialised as a string.
clips[i].videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
clips[i].videoFormat.height	number	Height dimension of video format.
clips[i].videoFormat.width	number	Width dimension of video format.
clips[i].videoFormat.interlaced	boolean	Is the display format interlaced?
clips[i].startTimecode	string	Start timecode of the clip serialised as string.
clips[i].durationTimecode	string	Duration of the clip in timecode format serialised as string.
clips[i].frameCount	integer	Number of frames in clip; duration of the clip in frames.

404 - There is no active disk.

Media Pool Control API

API to manage media pool and handle uploads and project data.

GET /cloud/projects

List all projects within the media pool.

Response

200 - Successfully retrieved the list of all projects.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].libraryID	string	
[i].name	string	
[i].private	boolean	
[i].shared	boolean	
[i].clips	array	List of clips associated with the project.
[i].clips[i]	string	
[i].status	object	
[i].status.numClipsRequested	integer	
[i].status.numClipsComplete	integer	
[i].status.uploadPercent	integer	Percentage of upload completion.
[i].status.numClipsPaused	integer	
[i].status.outOfSpace	boolean	
[i].status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
[i].status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/active

Retrieve data of the actively uploading project.

Response

200 - Successfully retrieved the active project's data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/{projectID}

Retrieve specific project data by project ID.

Parameters

Name	Type	Description
{projectID} (required)	integer	Unique identifier of the project.

Response

200 - Successfully retrieved the project data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	

Name	Type	Description
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

404 - Project not found.

GET /cloud/clips

List all clips within the media pool.

Response

200 - Successfully retrieved the list of all clips.

The response is JSON.

Name	Type	Description
	array	
[i]	string	

GET /cloud/clips/activeUploading

Retrieve data of actively uploading clips.

Response

200 - Successfully retrieved the list of actively uploading clips.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].path	string	
[i].projectID	integer	
[i].status	object	
[i].status.projectID	integer	
[i].status.outOfSpace	boolean	
[i].status.proxyExtension	string	
[i].status.growingFile	boolean	
[i].status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.originalClipTotalSize	integer	
[i].status.proxyClipTotalSize	integer	
[i].status.originalClipCompletedSize	integer	
[i].status.proxyClipCompletedSize	integer	
[i].status.secsRemaining	integer	

GET /cloud/clips/{deviceName}/{path}

Retrieve specific clip data by device and path.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored.
{path} (required)	string	Path to the clip.

Response

200 - Successfully retrieved the clip data.

The response is JSON.

Name	Type	Description
path	string	
projectID	integer	
status	object	
status.projectID	integer	
status.outOfSpace	boolean	
status.proxyExtension	string	
status.growingFile	boolean	
status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.originalClipTotalSize	integer	
status.proxyClipTotalSize	integer	
status.originalClipCompletedSize	integer	
status.proxyClipCompletedSize	integer	
status.secsRemaining	integer	

404 - Clip not found.

Monitoring Control API

API for monitoring and controlling display settings in video equipment.

GET /monitoring/display

Retrieve a list of all display names.

Response

200 - Returns a list of display names.

The response is JSON.

Name	Type	Description
displays	array	List of display names available.
displays[i]	string	

GET /monitoring/{displayName}/cleanFeed

Get the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/cleanFeed

Set the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Clean feed enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/displayLUT

Get the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

400 - Invalid display name.

404 - Display name not found.

PUT /monitoring/{displayName}/displayLUT

Set the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Display LUT enabled/disabled successfully.

400 - Invalid input.

422 - Unprocessable Entity - Unable to process the contained instructions.

GET /monitoring/{displayName}/zebra

Get the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/zebra

Set the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Zebra enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/focusAssist

Get the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/focusAssist

Set the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/focusAssist

Get the focus assist settings.

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

404 - Display name not found.

PUT /monitoring/focusAssist

Set the focus assist settings.

Parameters

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/frameGuide

Get the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame guide enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGuide

Set the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame guide state updated successfully.

422 - Unable to update the frame guide state.

GET /monitoring/frameGuideRatio

Get the current frame guide ratio.

Response

200 - Returns the current frame guide ratio.

The response is JSON.

Name	Type	Description
ratio	string	The frame guide ratio.

PUT /monitoring/frameGuideRatio

Set the frame guide ratio.

Parameters

Name	Type	Description
ratio	string	The frame guide ratio.

Response

204 - Frame guide ratio updated successfully.

422 - Unable to update the frame guide ratio.

GET /monitoring/frameGuideRatio/presets

Get the presets for frame guide ratios.

Response

200 - Returns a list of preset frame guide ratios.

The response is JSON.

Name	Type	Description
presets	array	
presets[i]	string	A frame guide ratio.

GET /monitoring/{displayName}/frameGrids

Get the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame grids enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGrids

Set the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame grids state updated successfully.

422 - Unable to update the frame grids state.

GET /monitoring/frameGrids

Get the global frame grids settings.

Response

200 - Returns the current frame grids settings.

The response is JSON.

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

PUT /monitoring/frameGrids

Set the global frame grids settings.

Parameters

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

Response

204 - Frame grids settings updated successfully.

400 - Invalid input, check the number of frame grids or values.

422 - Unable to update the frame grids settings.

GET /monitoring/{displayName}/safeArea

Get the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the safe area enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/safeArea

Set the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Safe area state updated successfully.

422 - Unable to update the safe area state.

GET /monitoring/safeAreaPercent

Get the current safe area percentage.

Response

200 - Returns the current safe area percentage.

The response is JSON.

Name	Type	Description
percent	integer	Safe area coverage percentage.

PUT /monitoring/safeAreaPercent

Set the safe area percentage.

Parameters

Name	Type	Description
percent	integer	Safe area coverage percentage to set.

Response

204 - Safe area percentage updated successfully.

400 - Invalid percentage value.

422 - Unable to update the safe area percentage.

GET /monitoring/{displayName}/falseColor

Get the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the false color enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/falseColor

Set the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - False color state updated successfully.

422 - Unable to update the false color state.

Event Control API

API For working with built-in websocket.

GET /event/list

Get the list of events that can be subscribed to using the websocket API.

Response

200 - Websocket API events list.

The response is JSON.

Name	Type	Description
events	array	List of events that can be subscribed to using the websocket API.
events[i]	string	

System Control API

API for controlling the System Modes on Blackmagic Design products.

GET /system

Get device system information.

Response

200 - System summary.

The response is JSON.

The properties will be populated only with the values that are supported/implemented by the device in use.

Name	Type	Description
codecFormat	object	Codec format configuration.
codecFormat.codec	string	Codec serialised as string.
codecFormat.container	string	Multimedia container format.
videoFormat	object	Video format configuration.
videoFormat.name (required)	string	Video format serialised as a string.
videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
videoFormat.height	number	Height dimension of video format.
videoFormat.width	number	Width dimension of video format.
videoFormat.interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/product

Get device product information.

Response

200 - Device product information.

The response is JSON.

Product information.

Name	Type	Description
deviceName	string	Name of device as displayed in Setup.
productName	string	Device's product name.
softwareVersion	string	Software version running on device.

501 - This functionality is not implemented for the device in use.

GET /system/supportedCodecFormats

Get the list of supported codecs.

Response

200 - List of supported codec formats.

The response is JSON.

Name	Type	Description
codecs	array	
codecs[i]	object	Codec format configuration.
codecs[i].codec	string	Codec serialised as string.
codecs[i].container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

GET /system/codecFormat

Get the currently selected codec.

Response

200 - Current codec format.

The response is JSON.

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

PUT /system/codecFormat

Update the system codec.

Parameters

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

Response

204 - The codec updated successfully.

400 - The specified codec format is unsupported.

501 - This functionality is not implemented for the device in use.

GET /system/videoFormat

Get the currently selected video format.

Response

200 - Current system video format.

The response is JSON.

Video format configuration.

Name	Type	Description
name (required)	string	Video format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height	number	Height dimension of video format.
width	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

PUT /system/videoFormat

Set the system video format.

Parameters

This parameter can be one of the following types:

Name	Type	Description
name (required)	string	Video format serialised as a string.

Name	Type	Description
frameRate (required)	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height (required)	number	Height dimension of video format.
width (required)	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

Response

204 - The video format updated successfully.

400 - Invalid request.

409 - Operation unsupported in the current state.

501 - This functionality is not implemented for the device in use.

GET /system/supportedVideoFormats

Get the list of supported video formats for the current system state.

Response

200 - List of supported video formats.

The response is JSON.

List of supported video formats.

Name	Type	Description
formats	array	List of video formats.
formats[i]	object	Video format configuration.
formats[i].name (required)	string	Video format serialised as a string.
formats[i].frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
formats[i].height	number	Height dimension of video format.
formats[i].width	number	Width dimension of video format.
formats[i].interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/supportedFormats

Get supported formats.

Response

200 - List of supported formats.

The response is JSON.

Name	Type	Description
supportedFormats	array	
supportedFormats[i]	object	
supportedFormats[i].codecs	array	
supportedFormats[i].codecs[i]	string	
supportedFormats[i].frameRates	array	
supportedFormats[i].frameRates[i]	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
supportedFormats[i]. maxOffSpeedFrameRate	number	
supportedFormats[i]. minOffSpeedFrameRate	number	
supportedFormats[i]. recordResolution	object	
supportedFormats[i]. recordResolution.height	number	Height of the resolution.
supportedFormats[i]. recordResolution.width	number	Width of the resolution.
supportedFormats[i]. sensorResolution	object	
supportedFormats[i]. sensorResolution.height	number	Height of the resolution.
supportedFormats[i]. sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

GET /system/format

Get current format.

Response

200 - Current format.

The response is JSON.

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

PUT /system/format

Set the format.

Parameters

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

Response

204 - System format updated.

501 - This functionality is not implemented for the device in use.

Transport Control API

API for controlling Transport on Blackmagic Design products.

GET /transports/0

Get device's basic transport status.

Response

200 - Transport status.

The response is JSON.

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

PUT /transports/0

Set device's basic transport status.

Parameters

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, Output.

Response

204 - Transport mode was set.

400 - Failed to set transport mode.

GET /transports/0/stop

Determine if transport is stopped.

Response

200 - Transport stop response.

The response is JSON.

Name	Type	Description
	boolean	True when transport mode is InputPreview or when in Output mode and speed is 0.

PUT /transports/0/stop

Stop transport. Deprecated, use POST /transports/0/stop instead.

Response

204 - Transport stopped.

POST /transports/0/stop

Stop transport.

Response

204 - Transport stopped.

GET /transports/0/play

Determine if transport is playing.

Response

200 - Transport play response.

The response is JSON.

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

PUT /transports/0/play

Start playing on transport. Deprecated, use POST /transports/0/play instead.

Response

204 - Transport playing.

400 - Failed to set transport to play.

POST /transports/0/play

Start playing on transport.

Response

204 - Transport playing.

400 - Failed to set transport to play.

GET /transports/0/playback

Get playback state.

Response

200 - Transport playback state.

The response is JSON.

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

PUT /transports/0/playback

Set playback state.

Parameters

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

Response

204 - Updated transport playback state.

400 - Failed to set transport playback state.

GET /transports/0/record

Get record state.

Response

200 - Recording state.

The response is JSON.

Name	Type	Description
recording	boolean	If true, transport is in InputRecord mode.

PUT /transports/0/record

Set record state. Deprecated, use POST /transports/0/record instead.

Parameters

Name	Type	Description
recording (required)	boolean	If true, starts a recording, otherwise stops.
clipName	string	Optional, sets the requested clip name to record to, when “recording” attribute is set to true.

Response

204 - Recording state updated.

400 - Failed to update recording state.

POST /transports/0/record

Start recording.

Parameters

Name	Type	Description
clipName	string	Optional, provides a specific name of clip to record to.

Response

204 - Recording started.

400 - Failed to start recording.

GET /transports/0/clipIndex

Get the clip index of the currently playing clip on the timeline.

Response

200 - Clip index response.

The response is JSON.

Name	Type	Description
clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

GET /transports/0/timecode

Get device timecode.

Response

200 - Timecode response.

The response is JSON.

Name	Type	Description
display	string	The display timecode serialised as a string.
timeline	string	The timeline timecode serialised as a string.

GET /transports/0/timecode/source

Get timecode source selected on device.

Response

200 - Timecode source response.

The response is JSON.

Name	Type	Description
timecode	string	Possible values are: Timeline, Clip.

Timeline Control API

API for controlling playback timeline.

GET /timelines/0

Get the playback timeline.

Response

200 - Playback timeline.

The response is JSON.

Name	Type	Description
clips	array	
clips[i]	object	Timeline clip.
clips[i].clipUniqueId (required)	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).
clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

404 - No timeline / disk available.

DELETE /timelines/0

Clear the current playback timeline. Deprecated, prefer to use POST /timelines/0/clear

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

POST /timelines/0

Add a clip to the timeline.

Parameters

This parameter can be one of the following types:

Add multiple media clips to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips[i].clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips[i].clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips[i].frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add multiple media clips to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add a single clip to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips.clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips.clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips.frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add a single clip to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the timeline as specified.

501 - The operation is not supported on this device.

POST /timelines/0/add

Add a clip to the end of the timeline. Deprecated, use POST /timelines/0 to add clips within the timeline.

Parameters

This parameter can be one of the following types:

Add one clip to the end of the timeline.

Name	Type	Description
clips	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add many clips to the end of the timeline.

Name	Type	Description
clips	array	List of clipUniqueIds of clips to add to end of timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the end of the timeline.

501 - The operation is not supported on this device.

POST /timelines/0/clear

Clear the playback timeline.

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

DELETE /timelines/0/clips/{timelineClipIndex}

Remove the specified clip from the timeline.

Parameters

Name	Type	Description
{timelineClipIndex} (required)	integer	The (0-based) timeline clip index of the clip to remove from the timeline.

Response

204 - The specified clip was removed from the timeline.

501 - The operation is not supported on this device.

Media Control API

API for controlling media devices in Blackmagic Design products.

GET /media/workingset

Get the list of media devices currently in the working set.

Response

200 - The list of media devices in the working set.

The response is JSON.

Name	Type	Description
size	integer	The fixed size of this device's working set.
workingset	array	The device's working set.
workingset[i]	object | null	Device within the working set. null if no device is present within the given working set slot.
workingset[i].volume	string	Volume name.
workingset[i].deviceName	string	Internal device name of this media device.
workingset[i].remainingRecordTime	integer	Remaining record time using current codec and video format in seconds.
workingset[i].totalSpace	integer	Total space on media device in bytes.
workingset[i].remainingSpace	integer	Remaining space on media device in bytes.
workingset[i].clipCount	integer	Number of clips currently on the device.

GET /media/active

Get the currently active media device.

Response

200 - The current active media device.

The response is JSON.

The active media device, or null if there is no active media.

Name	Type	Description
workingsetIndex	integer	Working set index of the active media device.
deviceName	string	Device name of media device.

204 - No media is currently active.

PUT /media/active

Set the currently active media device.

Parameters

Name	Type	Description
workingsetIndex	integer	Working set index of the media to make active.

Response

204 - The active media device was set successfully.

400 - Setting the currently active media device is not possible in the current state.

GET /media/devices/doformatSupportedFilesystems

Get the list of filesystems available to format a media device.

Response

200 - The list of filesystems permitted for formatting.

The response is JSON.

Name	Type	Description
	array	List of filesystems permitted for formatting media.
[i]	string	Filesystem serialised as string.

GET /media/devices/{deviceName}

Get information about a requested device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Information about the requested device.

The response is JSON.

Media device state.

Name	Type	Description
state	string	The current state of the media device. Possible values are: None, Scanning, Mounted, Uninitialised, Formatting, RaidComponent.

400 - Invalid device name.

404 - Device not found.

GET /media/devices/{deviceName}/doformat

Get a format key, used to format the device with a PUT request.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Format prepared.

The response is JSON.

Name	Type	Description
deviceName	string	Device name of media device to format.
key	string	The key required to format this device, provide to PUT /media/devices/{deviceName}/doformat to perform format of media device.

400 - Cannot format the device.

404 - Device not found.

PUT /media/devices/{deviceName}/doformat

Perform a format of the specified media device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by "deviceName" member of GET /media/workingset or GET /media/active.

Name	Type	Description
key	string	The key used to format this device, retrieved from prepare format media request GET /media/devices/{deviceName}/doformat. Format key provided cannot be reused after successful format.
filesystem	string	Filesystem to format to. Supported filesystems can be retrieved with GET /media/devices/doFormatSupportedFilesystems.
volume	string	Volume name to set for the disk after format.

Response

204 - Format successful.

400 - Cannot format the device, invalid filesystem or key.

404 - Device not found.

Slate Control API

API to manage digital slate data.

GET /slates/nextClip

Retrieve the digital slate for the next clip.

Response

200 - Returns the slate data for the next clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	

Name	Type	Description
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Slate data is not available.

PUT /slates/nextClip

Update the slate data for the next clip.

Parameters

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

POST /slates/nextClip/resetProjectData

Reset the project data for the next clip's slate.

Response

200 - Project data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetProjectData

Reset the project data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Project data reset successfully.

POST /slates/nextClip/resetLensData

Reset the lens data for the next clip's slate.

Response

200 - Lens data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetLensData

Reset the lens data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Lens data reset successfully.

GET /slates/clips/{deviceName}/{path}

Retrieve slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Returns the slate data for the specified clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

404 - Clip not found.

PUT /slates/clips/{deviceName}/{path}

Update the slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.

Name	Type	Description
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

Preset Control API

API For controlling the presets on Blackmagic Design products

GET /presets

Get the list of the presets on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presets	array	List of the presets on the camera (.cset files)
presets[i]	string	

POST /presets

Send a preset file to the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presetAdded	string	Name of the preset uploaded (without .cset extension)

400 - Bad request - missing Content-Disposition header or filename

GET /presets/active

Get the currently active preset on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

PUT /presets/active

Set the active preset on the camera

Parameters

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

Response

204 - No Content

404 - Preset file not found

PARAMETERS /presets/{presetName}

GET /presets/{presetName}

Download the preset file

Response

200 - OK

The response is a binary file.

404 - File does not exist

PUT /presets/{presetName}

Save current camera state as a preset

Response

204 - No Content

DELETE /presets/{presetName}

Delete a preset from the camera

Response

204 - No Content

404 - Preset file not found

Audio Control API

API For controlling audio on Blackmagic Design Cameras

GET /audio/channels

Get the total number of audio channels available

Response

200 - Returns the total number of channels

The response is JSON.

Name	Type	Description
channels	integer	Total number of audio channels available

GET /audio/supportedInputs

Get the list of supported audio inputs

Response

200 - List of all supported audio inputs

The response is JSON.

Name	Type	Description
	array	
[i]	string	A supported audio input

GET /audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently selected input

The response is JSON.

Name	Type	Description
input	string	Audio input source and type

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/input

Set the audio input for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
input	string	Audio input source and type

Response

204 - No Content

400 - Invalid audio input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/input/description

Get the description of the current input of the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Description of the current input of the selected channel

The response is JSON.

Name	Type	Description
description	object	
description.gainRange	object	
description.gainRange.Min	number	The minimum gain value in dB
description.gainRange.Max	number	The maximum gain value in dB
description.capabilities	object	
description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter
description.capabilities.Padding	object	
description.capabilities.Padding. available	boolean	Input supports setting of padding
description.capabilities.Padding. forced	boolean	Padding is forced to be set for the input
description.capabilities.Padding. value	number	Value of the padding in dB

404 - Channel does not exist

GET /audio/channel/{channelIndex}/supportedInputs

Get the list of supported inputs and their availability to switch to for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its supported inputs are being queried. (Channels index from 0)

Response

200 - The list of supported inputs

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

404 - Channel does not exist

GET /audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set level for the selected channel

The response is JSON.

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/level

Set the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

Response

204 - No Content

400 - Invalid input or value out of range

404 - Channel does not exist

GET /audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set phantom power for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Phantom power enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/phantomPower

Set the audio phantom power for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Phantom power enabled state

Response

204 - No Content

400 - Phantom power is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set padding for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Padding enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/padding

Set the audio input padding for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Padding enabled state

Response

204 - No Content

400 - Padding is not supported or is forced for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set low cut filter for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Low cut filter enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/lowCutFilter

Set the audio input low cut filter for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Low cut filter enabled state

Response

204 - No Content

400 - Low cut filter is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set availability for the selected channel

The response is JSON.

Name	Type	Description
available	boolean	Whether the input is currently available

404 - Channel does not exist

Lens Control API

API For controlling the lens on Blackmagic Design products

GET /lens/iris

Get lens' aperture

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number

PUT /lens/iris

Set lens' aperture

Parameters

Name	Type	Description
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number
adjustmentStep	integer	Signed value for relative aperture adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens iris is not controllable or is controlled by auto exposure

GET /lens/zoom

Get lens' zoom

Response

200 - OK

The response is JSON.

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value

PUT /lens/zoom

Set lens' zoom

Parameters

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value
adjustmentFocalLength	integer	Signed value for relative focal length adjustment
adjustmentNormalised	number	Signed normalized value for relative zoom adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens zoom is not controllable

GET /lens/focus

Get lens' focus

Response

200 - OK

The response is JSON.

Name	Type	Description
normalised	number	Normalised value

PUT /lens/focus

Set lens' focus

Parameters

Name	Type	Description
normalised	number	Normalised value
focusDistance	integer	Focus distance value

Response

204 - No Content

400 - Bad Request if out of range value is provided

PUT /lens/focus/doAutoFocus

Perform auto focus

Parameters

Name	Type	Description
position	object	
position.x	number	Normalized x coordinate for autofocus ROI
position.y	number	Normalized y coordinate for autofocus ROI

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens focus is not controllable

GET /lens/opticalImageStabilization

Get optical image stabilization status

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Whether optical image stabilization is enabled

501 - Not Implemented if optical image stabilization is not supported on this product

PUT /lens/opticalImageStabilization

Enable or disable optical image stabilization

Parameters

Name	Type	Description
enabled	boolean	Enable or disable optical image stabilization

Response

204 - No Content

501 - Not Implemented if optical image stabilization is not supported on this product

GET /lens/iris/description

Get detailed description of lens' iris capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the iris can be controlled
apertureStop	object	
apertureStop.min	number	Minimum aperture stop
apertureStop.max	number	Maximum aperture stop

GET /lens/zoom/description

Get detailed description of lens' zoom capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the zoom can be controlled
focalLength	object	
focalLength.adjustable	boolean	If focal length is adjustable
focalLength.min	integer	Minimum focal length
focalLength.max	integer	Maximum focal length

GET /lens/focus/description

Get detailed description of lens' focus capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the focus can be controlled
focusDistance	object	
focusDistance.adjustable	boolean	If focus distance is adjustable
focusDistance.min	number	Minimum focus distance
focusDistance.max	number	Maximum focus distance

Video Control API

API For controlling the video on Blackmagic Design products

GET /video/iso

Get current ISO

Response

200 - OK

The response is JSON.

Name	Type	Description
iso	integer	Current ISO value

PUT /video/iso

Set current ISO

Parameters

Name	Type	Description
iso	integer	ISO value to set

Response

204 - No Content

403 - ISO cannot be changed in the current state

GET /video/supportedISOs

Get the list of supported ISO settings

Response

200 - List of supported ISO values

The response is JSON.

Name	Type	Description
supportedISOs	array	Array of supported ISO values
supportedISOs[i]	integer	

GET /video/gain

Get current gain value in decibels

Response

200 - OK

The response is JSON.

Name	Type	Description
gain	integer	Current gain value in decibels

PUT /video/gain

Set current gain value

Parameters

Name	Type	Description
gain	integer	Gain value in decibels to set

Response

204 - No Content

403 - Gain cannot be changed in the current state

GET /video/supportedGains

Get the list of supported gain settings in decibels

Response

200 - List of supported gain values in decibels

The response is JSON.

Name	Type	Description
supportedGains	array	Array of supported gain values in decibels
supportedGains[i]	integer	

GET /video/whiteBalance

Get current white balance

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	integer	Current white balance

PUT /video/whiteBalance

Set current white balance

Parameters

Name	Type	Description
whiteBalance	integer	White balance to set

Response

204 - No Content

400 - Invalid white balance temperature

GET /video/whiteBalance/description

Get white balance range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	object	
whiteBalance.min	integer	Minimum color temperature
whiteBalance.max	integer	Maximum color temperature

PUT /video/whiteBalance/doAuto

Set current white balance automatically

Response

204 - No Content

GET /video/whiteBalanceTint

Get white balance tint

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	integer	Current white balance tint

PUT /video/whiteBalanceTint

Set white balance tint

Parameters

Name	Type	Description
whiteBalanceTint	integer	White balance tint to set

Response

204 - No Content

400 - Invalid white balance tint

GET /video/whiteBalanceTint/description

Get white balance tint range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	object	
whiteBalanceTint.min	integer	Minimum white balance tint
whiteBalanceTint.max	integer	Maximum white balance tint

GET /video/ndFilter

Get ND filter stop

Response

200 - OK

The response is JSON.

Name	Type	Description
stop	number	Current filter power (fStop)

501 - Not implemented for this device

PUT /video/ndFilter

Set ND filter stop

Parameters

Name	Type	Description
stop	number	Filter power (fStop) to set

Response

204 - No Content

400 - Invalid ND filter stop

501 - Not implemented for this device

GET /video/supportedNDFilters

Get the list of available ND filter stops

Response

200 - List of available ND filter stops

The response is JSON.

Name	Type	Description
supportedStops	array	Array of available ND filter stops
supportedStops[i]	number	

501 - Not implemented for this device

GET /video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

Response

200 - List of supported display modes

The response is JSON.

Name	Type	Description
supportedDisplayModes	array	Array of supported display modes
supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

GET /video/ndFilter/displayMode

Get ND filter display mode on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

PUT /video/ndFilter/displayMode

Set ND filter display mode on the camera

Parameters

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

Response

204 - No Content

400 - Invalid display mode for ND filter

501 - Not implemented for this device

GET /video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

Response

200 - Indicates if ND filter is selectable

The response is JSON.

Name	Type	Description
selectable	boolean	True if ND filter adjustments are selectable via a slider

501 - Not implemented for this device

GET /video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

PUT /video/shutter

Set current shutter

Parameters

Name	Type	Description
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

Response

204 - No Content

GET /video/shutter/measurement

Get the current shutter measurement mode

Response

200 - OK

The response is JSON.

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

PUT /video/shutter/measurement

Set the shutter measurement mode

Parameters

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

Response

204 - No Content

400 - Invalid shutter measurement

GET /video/supportedShutters

Get supported shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of supported shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of supported shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of flicker-free shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of flicker-free shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/autoExposure

Get current auto exposure mode

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

PUT /video/autoExposure

Set auto exposure

Parameters

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

Response

204 - No Content

400 - Failed to set auto exposure mode

GET /video/detailSharpening

Get the current state of detail sharpening

Response

200 - Current detail sharpening state

The response is JSON.

Name	Type	Description
enabled	boolean	Whether detail sharpening is enabled

501 - Not implemented for this device

PUT /video/detailSharpening

Enable or disable detail sharpening

Parameters

Name	Type	Description
enabled	boolean	Enable or disable detail sharpening

Response

204 - Detail sharpening state updated

501 - Not implemented for this device

GET /video/detailSharpeningLevel

Get the current detail sharpening level

Response

200 - Current detail sharpening level

The response is JSON.

Name	Type	Description
level	string	Current detail sharpening level Possible values are: Low, Medium, High.

501 - Not implemented for this device

PUT /video/detailSharpeningLevel

Set the detail sharpening level

Parameters

Name	Type	Description
level	string	Desired level of detail sharpening Possible values are: Low, Medium, High.

Response

204 - Detail sharpening level updated

400 - Invalid detail sharpening level

501 - Not implemented for this device

Camera Control API

API For controlling the Camera specific features on Blackmagic Design products

GET /camera/colorBars

Get the status of color bars display

Response

200 - Returns the current status of color bars

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the color bars are currently enabled

PUT /camera/colorBars

Set the status of color bars display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the color bars

Response

204 - Color bars status updated successfully

GET /camera/programFeedDisplay

Get the status of program feed display

Response

200 - Returns the current status of program feed display

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the program feed display is currently enabled

PUT /camera/programFeedDisplay

Set the status of program feed display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the program feed display

Response

204 - Program feed display status updated successfully

GET /camera/tallyStatus

Get the tally status of the camera

Response

200 - Returns the current tally status of the camera

The response is JSON.

Name	Type	Description
status	string	Current tally status of the camera Possible values are: None, Preview, Program.

GET /camera/power

Get the power status of the camera

Response

200 - Returns the current power status

The response is JSON.

Name	Type	Description
source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
batteries	array	List of batteries currently connected to the camera
batteries[i]	object	
batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
batteries[i].chargeRemainingPercent	integer	Remaining battery charge percentage
batteries[i].statusFlags	array	List of battery status flags
batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

GET /camera/power/displayMode

Get the power display mode of the camera

Response

200 - Returns the current power display mode

The response is JSON.

Name	Type	Description
mode	string	Current power display mode Possible values are: Percentage, Voltage.

PUT /camera/power/displayMode

Set the power display mode of the camera

Parameters

Name	Type	Description
mode	string	Power display mode to set Possible values are: Percentage, Voltage.

Response

204 - Power display mode updated successfully

400 - Invalid power display mode

GET /camera/timingReferenceLock

Get the timing reference lock status

Response

200 - Returns the timing reference lock status

The response is JSON.

Name	Type	Description
locked	boolean	Indicates if timing reference is locked

Immersive Control API

API for controlling immersive camera settings on Blackmagic Design cameras

GET /immersive/display/{displayName}/eye

Get the current eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to query (from /monitoring/display endpoint)

Response

200 - OK

The response is JSON.

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

400 - Invalid display name format

404 - Display not found

422 - Failed to get eye view

PUT /immersive/display/{displayName}/eye

Set the eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to control (from /monitoring/display endpoint)

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

Response

204 - No Content

400 - Invalid input or display name format

404 - Display not found

422 - Failed to set eye view

Color Correction Control API

API For controlling the color correction on Blackmagic Design products based on DaVinci Resolve Color Corrector

GET /colorCorrection/lift

Get color correction lift

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

PUT /colorCorrection/lift

Set color correction lift

Parameters

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gamma

Get color correction gamma

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

PUT /colorCorrection/gamma

Set color correction gamma

Parameters

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gain

Get color correction gain

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

PUT /colorCorrection/gain

Set color correction gain

Parameters

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/offset

Get color correction offset

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

PUT /colorCorrection/offset

Set color correction offset

Parameters

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/contrast

Get color correction contrast

Response

200 - OK

The response is JSON.

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/contrast

Set color correction contrast

Parameters

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/color

Get color correction color properties

Response

200 - OK

The response is JSON.

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/color

Set color correction color properties

Parameters

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/lumaContribution

Get color correction luma contribution

Response

200 - OK

The response is JSON.

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

PUT /colorCorrection/lumaContribution

Set color correction luma contribution

Parameters

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

Response

204 - No Content

Notification websocket - 1.0.0

Service that notifies subscribers of device state changes.

messages

Subscribe (The messages from the server/device)

Websocket Opened Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: websocketOpened.
.type	string	Possible values are: event.

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*

Name	Type	Description
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Event Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: propertyValueChanged.

Name	Type	Description
.data.property	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.value	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.type	string	Possible values are: event.

Publish (The messages that user can send to the server/device)

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	

Name	Type	Description
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	

Name	Type	Description
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Device Properties

/media/workingset

The value JSON returned via the eventResponse when the `/media/workingset` property changes on the device:

Name	Type	Description
.size	integer	The fixed size of this device's working set.
.workingset	array	Array of devices within the working set. null if no device is present within the given working set slot.
.workingset[i]		

/media/active

The value JSON returned via the eventResponse when the `/media/active` property changes on the device:

Name	Type	Description
.workingsetIndex	integer	Working set index of the active media device.
.deviceName	string	Internal device name of this media device.

/system

The value JSON returned via the eventResponse when the `/system` property changes on the device:

Name	Type	Description
.codecFormat	object	Codec format configuration.
.codecFormat.codec	string	Codec format serialised as a string.
.codecFormat.container	string	Multimedia container format.
.videoFormat	object	Currently selected video format.
.videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.videoFormat.height	number	Height dimension of video format.
.videoFormat.width	number	Width dimension of video format.
.videoFormat.interlaced	boolean	Is the display format interlaced?.
.videoFormat.name	string	Video format serialised as a string.

/system/codecFormat

Codec format configuration.

The value JSON returned via the eventResponse when the `/system/codecFormat` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.container	string	Multimedia container format.

/system/videoFormat

Currently selected video format.

The value JSON returned via the eventResponse when the `/system/videoFormat` property changes on the device:

Name	Type	Description
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.height	number	Height dimension of video format.
.width	number	Width dimension of video format.
.interlaced	boolean	Is the display format interlaced?.
.name	string	Video format serialised as a string.

/system/format

The value JSON returned via the eventResponse when the `/system/format` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.maxOffSpeedFrameRate	number	
.minOffSpeedFrameRate	number	
.offSpeedEnabled	boolean	
.offSpeedFrameRate	number	
.recordResolution	object	
.recordResolution.height	number	Height of the resolution.
.recordResolution.width	number	Width of the resolution.
.sensorResolution	object	
.sensorResolution.height	number	Height of the resolution.
.sensorResolution.width	number	Width of the resolution.

/system/supportedFormats

The value JSON returned via the eventResponse when the `/system/supportedFormats` property changes on the device:

Name	Type	Description
.supportedFormats	array	
.supportedFormats[i]	object	
.supportedFormats[i].codecs	array	
.supportedFormats[i].codecs[i]	string	
.supportedFormats[i].frameRates	array	
.supportedFormats[i].frameRates[i]	string	Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.supportedFormats[i].maxOffSpeedFrameRate	number	
.supportedFormats[i].minOffSpeedFrameRate	number	
.supportedFormats[i].recordResolution	object	
.supportedFormats[i].recordResolution.height	number	Height of the resolution.
.supportedFormats[i].recordResolution.width	number	Width of the resolution.
.supportedFormats[i].sensorResolution	object	
.supportedFormats[i].sensorResolution.height	number	Height of the resolution.
.supportedFormats[i].sensorResolution.width	number	Width of the resolution.

/timelines/0

The value JSON returned via the eventResponse when the `/timelines/0` property changes on the device:

Name	Type	Description
.clips	array	
.clips[i]	object	Timeline clip.
.clips[i].clipUniqueId	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
.clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
.clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
.clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
.clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).

Name	Type	Description
.clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
.clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

/transports/0

The value JSON returned via the eventResponse when the `/transports/0` property changes on the device:

Name	Type	Description
.mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

/transports/0/stop

true when transport mode is InputPreview or when in Output mode and speed is 0.

The value JSON returned via the eventResponse when the `/transports/0/stop` property changes on the device:

Name	Type	Description
	boolean	true when transport mode is InputPreview or when in Output mode and speed is 0.

/transports/0/play

True when transport is in Output mode and speed is non-zero.

The value JSON returned via the eventResponse when the /transports/0/play property changes on the device:

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

/transports/0/playback

The value JSON returned via the eventResponse when the `/transports/0/playback` property changes on the device:

Name	Type	Description
.type	string	Possible values are: Play, Jog, Shuttle, Var.
.loop	boolean	When true playback loops from the end of the timeline to the beginning of the timeline.
.singleClip	boolean	When true playback loops from the end of the current clip to the beginning of the current clip.
.speed	number	Playback speed, 1.0 for normal forward playback
.position	integer	Playback position on the timeline in units of video frames.

/transports/0/record

The value JSON returned via the eventResponse when the `/transports/0/record` property changes on the device:

Name	Type	Description
.recording	boolean	Is transport in Input Record mode.

/transports/0/timecode

The value JSON returned via the eventResponse when the `/transports/0/timecode` property changes on the device:

Name	Type	Description
.display	string	The display timecode serialised as a string.
.timeline	string	The timeline timecode serialised as a string.

/transports/0/timecode/source

The value JSON returned via the eventResponse when the `/transports/0/timecode/source` property changes on the device:

Name	Type	Description
.timecode	string	Possible values are: Timeline, Clip.

/transports/0/clipIndex

The value JSON returned via the eventResponse when the `/transports/0/clipIndex` property changes on the device:

Name	Type	Description
.clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

/slates/nextClip

The value JSON returned via the eventResponse when the `/slates/nextClip` property changes on the device:

Name	Type	Description
.clip	object	
.clip.clipName	string	
.clip.reel	integer	
.clip.scene	string	
.clip.sceneLocation	string	Possible values are: Interior, Exterior.
.clip.sceneTime	string	Possible values are: Day, Night.
.clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
.clip.take	integer	
.clip.takeType	string	Possible values are: None, PU, VFX, SER.
.clip.goodTake	boolean	
.lens	object	
.lens.lensType	string	
.lens.iris	string	

Name	Type	Description
.lens.focalLength	string	
.lens.distance	string	
.lens.filter	string	
.project	object	
.project.projectName	string	
.project.director	string	
.project.camera	string	
.project.cameraOperator	string	

/monitoring/{displayName}/cleanFeed

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/cleanFeed` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/displayLUT

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/displayLUT` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/zebra

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/zebra` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/focusAssist` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGuide

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGuide` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGrids` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/safeArea

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/safeArea` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/falseColor

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/falseColor` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/focusAssist` property changes on the device:

Name	Type	Description
.mode	string	Possible values are: Peak, ColoredLines.
.color	string	Possible values are: Red, Green, Blue, White, Black.
.intensity	integer	

/monitoring/frameGuideRatio

The value JSON returned via the eventResponse when the `/monitoring/frameGuideRatio` property changes on the device:

Name	Type	Description
.ratio	string	

/monitoring/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/frameGrids` property changes on the device:

Name	Type	Description
.frameGrids	array	
.frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

/monitoring/safeAreaPercent

The value JSON returned via the eventResponse when the `/monitoring/safeAreaPercent` property changes on the device:

Name	Type	Description
.percent	integer	Safe area coverage percentage.

/audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input` property changes on the device:

Name	Type	Description
.input	string	Audio input source and type

/audio/channel/{channelIndex}/supportedInputs

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/supportedInputs` property changes on the device:

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

/audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/level` property changes on the device:

Name	Type	Description
.gain	number	Gain value in dB
.normalised	number	Normalised level value between 0.0 and 1.0

/audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/phantomPower` property changes on the device:

Name	Type	Description
.enabled	boolean	Phantom power enabled state

/audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/padding` property changes on the device:

Name	Type	Description
.enabled	boolean	Padding enabled state

/audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/lowCutFilter` property changes on the device:

Name	Type	Description
.enabled	boolean	Low cut filter enabled state

/audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/available` property changes on the device:

Name	Type	Description
.available	boolean	Whether the input is currently available

/audio/channel/{channelIndex}/input/description

Description of the current input of the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input/description` property changes on the device:

Name	Type	Description
.description	object	
.description.gainRange	object	
.description.gainRange.Min	number	The minimum gain value in dB
.description.gainRange.Max	number	The maximum gain value in dB
.description.capabilities	object	
.description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
.description.capabilities. LowCutFilter	boolean	Input supports setting of low cut filter

Name	Type	Description
.description.capabilities.Padding	object	
.description.capabilities.Padding.available	boolean	Input supports setting of padding
.description.capabilities.Padding.forced	boolean	Padding is forced to be set for the input
.description.capabilities.Padding.value	number	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.

/colorCorrection/lift

Get color correction lift

The value JSON returned via the eventResponse when the `/colorCorrection/lift` property changes on the device:

Name	Type	Description
.red	number	Red lift component
.green	number	Green lift component
.blue	number	Blue lift component
.luma	number	Luma lift component

/colorCorrection/gamma

Get color correction gamma

The value JSON returned via the eventResponse when the `/colorCorrection/gamma` property changes on the device:

Name	Type	Description
.red	number	Red gamma component
.green	number	Green gamma component
.blue	number	Blue gamma component
.luma	number	Luma gamma component

/colorCorrection/gain

Get color correction gain

The value JSON returned via the eventResponse when the `/colorCorrection/gain` property changes on the device:

Name	Type	Description
.red	number	Red gain component
.green	number	Green gain component
.blue	number	Blue gain component
.luma	number	Luma gain component

/colorCorrection/offset

Get color correction offset

The value JSON returned via the eventResponse when the `/colorCorrection/offset` property changes on the device:

Name	Type	Description
.red	number	Red offset component
.green	number	Green offset component
.blue	number	Blue offset component
.luma	number	Luma offset component

/colorCorrection/contrast

Get color correction contrast

The value JSON returned via the eventResponse when the `/colorCorrection/contrast` property changes on the device:

Name	Type	Description
.pivot	number	Contrast pivot point
.adjust	number	Contrast adjustment

/colorCorrection/color

Get color correction color properties

The value JSON returned via the eventResponse when the `/colorCorrection/color` property changes on the device:

Name	Type	Description
.hue	number	Color hue adjustment
.saturation	number	Color saturation adjustment

/colorCorrection/lumaContribution

Get color correction luma contribution

The value JSON returned via the eventResponse when the `/colorCorrection/lumaContribution` property changes on the device:

Name	Type	Description
.lumaContribution	number	Luma contribution value

/lens/iris

Get lens' aperture

The value JSON returned via the eventResponse when the `/lens/iris` property changes on the device:

Name	Type	Description
.continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
.apertureStop	number	Aperture stop value
.normalised	number	Normalised value
.apertureNumber	integer	Aperture number

/lens/iris/description

Get detailed description of lens' iris capabilities

The value JSON returned via the eventResponse when the `/lens/iris/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the iris can be controlled
.apertureStop	object	
.apertureStop.min	number	Minimum aperture stop
.apertureStop.max	number	Maximum aperture stop

/lens/focus

Get lens' focus

The value JSON returned via the eventResponse when the `/lens/focus` property changes on the device:

Name	Type	Description
.normalised	number	Normalised value

/lens/focus/description

Get detailed description of lens' focus capabilities

The value JSON returned via the eventResponse when the `/lens/focus/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the focus can be controlled
.focusDistance	object	
.focusDistance.adjustable	boolean	If focus distance is adjustable
.focusDistance.min	number	Minimum focus distance
.focusDistance.max	number	Maximum focus distance

/lens/zoom

Get lens' zoom

The value JSON returned via the eventResponse when the `/lens/zoom` property changes on the device:

Name	Type	Description
.focalLength	integer	Focal length in mm
.normalised	number	Normalised value

/lens/zoom/description

Get detailed description of lens' zoom capabilities

The value JSON returned via the eventResponse when the `/lens/zoom/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the zoom can be controlled
.focalLength	object	
.focalLength.adjustable	boolean	If focal length is adjustable
.focalLength.min	integer	Minimum focal length
.focalLength.max	integer	Maximum focal length

/presets

Get the list of the presets on the camera

The value JSON returned via the eventResponse when the `/presets` property changes on the device:

Name	Type	Description
.presets	array	List of the presets on the camera (.cset files)
.presets[i]	string	

/presets/active

Get the currently active preset on the camera

The value JSON returned via the eventResponse when the `/presets/active` property changes on the device:

Name	Type	Description
.preset	string	Name of the active preset (with .cset extension, or 'default')

/camera/colorBars

Get the status of color bars display

The value JSON returned via the eventResponse when the `/camera/colorBars` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the color bars are currently enabled

/camera/programFeedDisplay

Get the status of program feed display

The value JSON returned via the eventResponse when the `/camera/programFeedDisplay` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the program feed display is currently enabled

/camera/tallyStatus

Get the tally status of the camera

The value JSON returned via the eventResponse when the `/camera/tallyStatus` property changes on the device:

Name	Type	Description
.status	string	Current tally status of the camera Possible values are: None, Preview, Program.

/camera/power

Get the power status of the camera

The value JSON returned via the eventResponse when the `/camera/power` property changes on the device:

Name	Type	Description
.source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
.milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
.batteries	array	
.batteries[i]	object	
.batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
.batteries[i]. chargeRemainingPercent	integer	Remaining battery charge percentage
.batteries[i].statusFlags	array	List of battery status flags
.batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

/camera/power/displayMode

Get the power display mode of the camera

The value JSON returned via the eventResponse when the `/camera/power/displayMode` property changes on the device:

Name	Type	Description
.mode	string	Current power display mode Possible values are: Percentage, Voltage.

/camera/timingReferenceLock

Get the timing reference lock status

The value JSON returned via the eventResponse when the `/camera/timingReferenceLock` property changes on the device:

Name	Type	Description
.locked	boolean	Indicates if timing reference is locked

/video/iso

Get current ISO

The value JSON returned via the eventResponse when the `/video/iso` property changes on the device:

Name	Type	Description
.iso	integer	Current ISO value

/video/supportedISOs

Get the list of supported ISO settings

The value JSON returned via the eventResponse when the `/video/supportedISOs` property changes on the device:

Name	Type	Description
.supportedISOs	array	Array of supported ISO values
.supportedISOs[i]	integer	

/video/gain

Get current gain value in decibels

The value JSON returned via the eventResponse when the `/video/gain` property changes on the device:

Name	Type	Description
.gain	integer	Current gain value in decibels

/video/supportedGains

Get the list of supported gain settings in decibels

The value JSON returned via the eventResponse when the `/video/supportedGains` property changes on the device:

Name	Type	Description
.supportedGains	array	Array of supported gain values in decibels
.supportedGains[i]	integer	

/video/whiteBalance

Get current white balance

The value JSON returned via the eventResponse when the `/video/whiteBalance` property changes on the device:

Name	Type	Description
.whiteBalance	integer	Current white balance

/video/whiteBalance/description

Get white balance range

The value JSON returned via the eventResponse when the `/video/whiteBalance/description` property changes on the device:

Name	Type	Description
.whiteBalance	object	
.whiteBalance.min	integer	Minimum color temperature
.whiteBalance.max	integer	Maximum color temperature

/video/whiteBalanceTint

Get white balance tint

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint` property changes on the device:

Name	Type	Description
.whiteBalanceTint	integer	Current white balance tint

/video/whiteBalanceTint/description

Get white balance tint range

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint/description` property changes on the device:

Name	Type	Description
.whiteBalanceTint	object	
.whiteBalanceTint.min	integer	Minimum white balance tint
.whiteBalanceTint.max	integer	Maximum white balance tint

/video/ndFilter

Get ND filter stop

The value JSON returned via the eventResponse when the `/video/ndFilter` property changes on the device:

Name	Type	Description
.stop	number	Current filter power (fStop)

/video/supportedNDFilters

Get the list of available ND filter stops

The value JSON returned via the eventResponse when the `/video/supportedNDFilters` property changes on the device:

Name	Type	Description
.supportedStops	array	Array of available ND filter stops
.supportedStops[i]	number	

/video/ndFilter/displayMode

Get ND filter display mode on the camera

The value JSON returned via the eventResponse when the `/video/ndFilter/displayMode` property changes on the device:

Name	Type	Description
.displayMode	string	ND filter display mode Possible values are: Stop, Number, Fraction.

/video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

The value JSON returned via the eventResponse when the `/video/supportedNDFilterDisplayModes` property changes on the device:

Name	Type	Description
.supportedDisplayModes	array	Array of supported display modes
.supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

/video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

The value JSON returned via the eventResponse when the `/video/ndFilterSelectable` property changes on the device:

Name	Type	Description
.selectable	boolean	True if ND filter adjustments are selectable via a slider

/video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

The value JSON returned via the eventResponse when the `/video/shutter` property changes on the device:

Name	Type	Description
.continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
.shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
.shutterAngle	number	Shutter angle

/video/shutter/measurement

Get the current shutter measurement mode

The value JSON returned via the eventResponse when the `/video/shutter/measurement` property changes on the device:

Name	Type	Description
.measurement	string	Current shutter measurement mode Possible values are: ShutterAngle, ShutterSpeed.

/video/supportedShutters

Get supported shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/supportedShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of supported shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array of flicker-free shutter speeds
.shutterSpeeds[i]	integer	

/video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/flickerFreeShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of flicker-free shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array
.shutterSpeeds[i]	integer	

/video/autoExposure

Get current auto exposure mode

The value JSON returned via the eventResponse when the `/video/autoExposure` property changes on the device:

Name	Type	Description
.mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
.type	string	Comma-separated list of device types in the auto exposure stack

/video/detailSharpening

Get the current state of detail sharpening

The value JSON returned via the eventResponse when the `/video/detailSharpening` property changes on the device:

Name	Type	Description
.enabled	boolean	Whether detail sharpening is enabled

/video/detailSharpeningLevel

Get the current detail sharpening level

The value JSON returned via the eventResponse when the `/video/detailSharpeningLevel` property changes on the device:

Name	Type	Description
.level	string	Current detail sharpening level of supported shutter speeds Possible values are: Low, Medium, High.

Assistenza

Per ricevere assistenza e consultare il materiale di supporto più recente della camera, visita la pagina Supporto del sito web di Blackmagic Design.

Pagina di supporto online

Il manuale, il software e le note di supporto più recenti sono disponibili alla pagina www.blackmagicdesign.com/it/support

Contattare Blackmagic Design

Se il materiale di supporto disponibile online non risponde alle tue domande, clicca su **Inviaci una email** dalla pagina Supporto, oppure su **Trova un team di supporto** per telefonare direttamente al team Blackmagic Design più vicino a te.

Verificare la versione del software installato

Per verificare quale versione del software Blackmagic Camera Setup è installata sul tuo computer puoi consultare la finestra About Blackmagic Camera Setup.

- Su Mac, apri la cartella **Applicazioni** e seleziona **Blackmagic Camera Setup**, poi **About Blackmagic Camera Setup** per vedere il numero della versione installata.
- Su Windows, apri **Blackmagic Camera Setup** dal menù **Start** poi clicca su **Help** e seleziona **About Blackmagic Camera Setup** per vedere il numero della versione installata.

Scaricare gli aggiornamenti

Dopo aver controllato quale versione dell'utilità Blackmagic Camera Setup è installata sul tuo computer, visita il Centro assistenza di Blackmagic Design su www.blackmagicdesign.com/it/support per scaricare gli aggiornamenti più recenti. Consigliamo di non aggiornare il software nel pieno svolgimento di un progetto importante.

Normative

Smaltimento di apparecchiature elettriche ed elettroniche nell'Unione Europea

Questo simbolo indica che il dispositivo non deve essere scartato insieme agli altri rifiuti, ma consegnato a uno degli appositi centri di raccolta e riciclaggio. La raccolta e lo smaltimento differenziato corretto di questo tipo di dispositivo evitano lo spreco di risorse e contribuiscono alla sostenibilità ambientale e umana. Per tutte le informazioni sui centri di raccolta e riciclaggio, contatta gli uffici del tuo comune di residenza o il punto vendita presso cui hai acquistato il prodotto.

Questo dispositivo è stato testato e dichiarato conforme ai limiti relativi ai dispositivi digitali di classe A, secondo la Parte 15 del regolamento FCC. Tali limiti sono stati stabiliti con lo scopo di fornire protezione ragionevole da interferenze dannose in ambienti commerciali. Questo dispositivo genera, usa e può irradiare energia a radiofrequenza e, se non è installato o usato in conformità alle istruzioni, può causare interferenze dannose che compromettono le comunicazioni radio. Operare questo dispositivo in ambienti residenziali può causare interferenze dannose, nella cui evenienza l'utente dovrà porvi rimedio a proprie spese.

Il funzionamento di questo dispositivo è soggetto alle due condizioni seguenti:

1. L'apparecchiatura non deve causare interferenze dannose.
2. Questo dispositivo o apparecchiatura deve accettare eventuali interferenze ricevute, incluse le interferenze che possono causare un funzionamento indesiderato.

R-R-BMD-20250213001

ICES-3 (A)
NMB-3 (A)

Dichiarazione ISED (Canada)

Questo dispositivo è conforme agli standard canadesi sui dispositivi digitali di classe A.

Qualsiasi modifica o utilizzo del dispositivo al di fuori di quello previsto potrebbero invalidare la conformità a tali standard.

Consigliamo di connettere le interfacce HDMI usando cavi schermati HDMI di alta qualità.

Questo dispositivo è stato testato per l'uso in ambienti commerciali. Se utilizzato in ambienti domestici, può causare interferenze radio.

Sicurezza

L'alimentatore da AC a DC 24V per URSA Cine deve essere collegato a una presa di corrente con messa a terra.

Le parti all'interno di URSA Cine non sono riparabili dall'utente. Contattare un centro assistenza Blackmagic Design per le operazioni di manutenzione.

Quando si connette l'ingresso di alimentazione a una batteria esterna DC, il cavo di alimentazione deve includere un limitatore di corrente o protezione da fusibile.

Inoltre deve essere di tipo VW-1 o conforme alle parti pertinenti delle norme IEC 60332 o IEC 60695.

Consigliamo che la fonte DC per questa camera sia di 24V. Se la fonte utilizzata è 12V DC, il cavo deve essere sufficiente per una corrente di 14A.

Questo dispositivo ha in dotazione una piastra per batteria con attacco B. Si consiglia l'utilizzo di una batteria 24V (28,8V) o a doppio voltaggio.

Per una batteria 12V (14,4V), consulta il manuale della batteria stessa o i suoi contrassegni per determinare la sua massima corrente di scarica continua. Consigliamo che abbia un valore nominale minimo di 12A.

Dichiarazione dello Stato della California

Questo prodotto può esporre l'utente a sostanze chimiche, per esempio tracce di bifenili polibromurati nelle parti in plastica, che nello Stato della California sono considerati causa di cancro e difetti congeniti o altri danni riproduttivi.

Per maggiori informazioni, visitare la pagina www.P65Warnings.ca.gov.

Ufficio in Europa

Blackmagic Design Europe B.V.
Rijnlanderweg 766, Unit D
2132 NM Hoofddorp
Paesi Bassi

Blackmagic URSA Cine

In condizioni di forte illuminazione, si consiglia di coprire il dispositivo per evitare che la camera e la batteria al litio siano esposte al sole per lunghi periodi. Mantenere le batterie al litio lontano dalle sorgenti di calore.

Il connettore RS 24V è adatto per alimentare accessori di terzi. Assicurarsi che il consumo energetico non superi i 48W.

Garanzia

Garanzia limitata

Blackmagic Design garantisce che questo prodotto è fornito privo di difetti nei materiali e nella manifattura per un periodo di 12 mesi a partire dalla data d'acquisto. Durante il periodo di garanzia, Blackmagic Design, a sua scelta, riparerà il prodotto difettoso, senza costi aggiuntivi per le parti e la manodopera, o offrirà una sostituzione, purché il prodotto difettoso venga restituito.

Per ottenere il servizio previsto dalla presente garanzia, il/la Cliente deve notificare Blackmagic Design del difetto entro il periodo di garanzia e accordarsi sulla prestazione del servizio. Il/la Cliente è responsabile del costo di imballaggio e di spedizione del prodotto al centro di assistenza indicato da Blackmagic Design, con spese di spedizione prepagate. Le spese di spedizione, l'assicurazione, le tasse, la dogana e altre spese pertinenti la resa del prodotto a Blackmagic Design sono a carico del/la Cliente.

Questa garanzia perde di validità per difetti, malfunzionamenti o danni causati da un utilizzo improprio o da manutenzione e cura inadeguate del prodotto. Blackmagic Design non è obbligata sotto questa garanzia a: a) riparare danni causati da tentativi di installazione, riparazione o manutenzione da parte di personale che non sia autorizzato da Blackmagic Design, b) riparare danni causati da uso improprio o connessione ad attrezzatura incompatibile, c) riparare danni o malfunzionamenti causati dall'uso di parti o ricambi non originali Blackmagic Design, o d) fare manutenzione se il prodotto è stato modificato o integrato ad altri prodotti con il risultato di allungare i tempi della manutenzione o di renderla più difficoltosa.

Esporre URSA Viewfinder alla luce diretta del sole può danneggiare il display, poiché le ottiche del viewfinder si comportano come una lente d'ingrandimento. È possibile che si verifichi una ritenzione di immagine o che l'immagine rimanga impressa sui pannelli OLED quando visualizzano immagini statiche o ad alto contrasto, per esempio le guide di inquadratura, per lunghi periodi. Per evitare questo problema, evita di coprire deliberatamente il sensore di movimento a infrarossi e disconnetti il viewfinder quando non è in uso per lunghi periodi. La ritenzione di immagine non è coperta dalla garanzia.

LA PRESENTE GARANZIA DI BLACKMAGIC DESIGN SOSTITUISCE QUALSIASI ALTRA GARANZIA, ESPLICITA O IMPLICITA. BLACKMAGIC DESIGN E I SUOI FORNITORI ESCLUDONO QUALSIASI ALTRA GARANZIA IMPLICITA DI COMMERCIABILITÀ O DI IDONEITÀ AD UN USO SPECIFICO. L'INTERA RESPONSABILITÀ DI BLACKMAGIC DESIGN DI RIPARARE O SOSTITUIRE I PRODOTTI DIFETTOSI DURANTE IL PERIODO DI GARANZIA È L'UNICO ED ESCLUSIVO RIMEDIO FORNITO AL/LA CLIENTE. BLACKMAGIC DESIGN NON SI ASSUME ALCUNA RESPONSABILITÀ PER QUALSIASI DANNO ARRECATO DI NATURA INDIRETTA, SPECIFICA, ACCIDENTALE O CONSEQUENZIALE, ANCHE QUALORA BLACKMAGIC DESIGN O IL FORNITORE FOSSERO STATI AVVERTITI DELLA POSSIBILITÀ DI TALI DANNI. BLACKMAGIC DESIGN NON SI ASSUME ALCUNA RESPONSABILITÀ PER QUALSIASI USO ILLEGALE DEL DISPOSITIVO DA PARTE DEL/LA CLIENTE. BLACKMAGIC DESIGN NON SI ASSUME ALCUNA RESPONSABILITÀ PER DANNI DERIVANTI DALL'USO DI QUESTO PRODOTTO. IL/LA CLIENTE UTILIZZA QUESTO PRODOTTO A PROPRIO RISCHIO.

© Copyright 2025 Blackmagic Design. Tutti i diritti riservati. "Blackmagic Design", "URSA", "DeckLink", "HDLink", "Workgroup Videohub", "Multibridge Pro", "Multibridge Extreme", "Intensity" e "Leading the creative video revolution" sono marchi registrati negli Stati Uniti e in altri Paesi. Altri nomi di prodotti e aziende qui contenuti potrebbero essere marchi dei rispettivi proprietari.

Il nome, marchio e logo Bluetooth® sono registrati e appartengono a Bluetooth SIG, Inc. Qualsiasi uso da parte di Blackmagic Design di questi marchi registrati è sotto licenza. Altri marchi di fabbrica e nomi commerciali ivi contenuti appartengono ai rispettivi proprietari.

Outubro 2025

Guia de Câmera

Blackmagicdesign

Blackmagic

URSA Cine Immersive



Blackmagic URSA Cine Immersive

Índice

Blackmagic URSA Cine Immersive 1373
O que é Imersivo? 1373
Apple Immersive Video 1373
URSA Cine Immersive 1374
Primeiros Passos 1375
Ligar a Câmera 1375
Ajustar as Configurações de Gravação 1376
Monitorar sua Imagem 1378
Opções de Monitoramento e Sobreposições 1378
Enquadrar seu Primeiro Plano 1379
Altura da Câmera 1379
Inclinação da Câmera 1380
Posicionamento 1381
Autonomia do Público 1382
Movimento da Câmera 1382
Tabelas de Taxas de Dados 1383
Áudio e Código de Tempo 1384
Formato de Áudio Espacial da Apple 1384
Ambissônico 1384
Microfones de Lapela e Miniatura 1385
Microfones Portáteis 1385
Conectar o Código de Tempo 1385
Filmagem 1387
Requisitos de Exposição 1387
Velocidade do Obturador 1387
ISO e Faixa Dinâmica 1388
ND 1389
Gravação Off-Speed 1390
Histograma 1391
Opções de Monitoramento LCD 1391
Gravação 1392
Reprodução 1393
Transferir Clipes 1394
Resumo 1395
DaVinci Resolve Studio 1396
Recomendações de Estação de Trabalho 1396
Ajuste de Projeto 1397
Definições Máster 1397
Monitoramento de Vídeo 1398
Tamanho de Imagem 1398
Gerenciamento de Cor 1399
Sobre o DaVinci Wide Gamut 1400
Processamento em Apple Immersive 1401
Ingestão 1401
Edição 1403
Montagem 1403
Fundos 1403
Usar o Visualizador 1405
Transições 1407
Apple Immersive Motion Data 1408
Títulos 1409
Considerações Criativas 1410
Composições VFX 1411
Nós MediaOut 1411
Renderizador 3D 1412
Composições com o Immersive Patcher 1413
Tratamento de Cores 1414
Redução de Ruído 1414
Gradação Primária 1415
Secundárias 1415
Tamanho 1416
Paleta 3D 1416
Gerenciamento de Cores CST 1418
Mapeamento de Linha de Tempo CST 1418
Mapeamento de Monitor de Gradação CST 1419
Monitoramento Apple Vision Pro CST 1421
Áudio 1422
Apple Spatial Audio Format (ASAF) 1422
Entrega 1423
Ignorar Transições 1423
Transmitir para o Apple Vision Pro 1424
Sobre o Apple Immersive Video Utility 1424
Developer Information 1425
Camera Control REST API 1425
Livestream Control API 1427
Clips Control API 1431
Media Pool Control API 1432

Monitoring Control API 1435
Event Control API 1444
System Control API 1444
Transport Control API 1450
Timeline Control API 1454
Media Control API 1457
Slate Control API 1460
Preset Control API 1466
Audio Control API 1468
Lens Control API 1473
Video Control API 1477
Camera Control API 1485
Immersive Control API 1488
Color Correction Control API 1489
Notification websocket - 1.0.0 1494
messages 1494
Ajuda 1518
Informações Regulatórias 1519
Informações de Segurança 1520
Garantia 1521

Blackmagic URSA Cine Immersive

Parabéns pela compra ou aluguel da Blackmagic URSA Cine Immersive.

A Blackmagic URSA Cine Immersive foi projetada para gravar conteúdo de realidade virtual estereoscópica de 180 graus, também chamado de vídeo imersivo, ela foi otimizada especialmente para gravar no formato Apple Immersive Video para o Apple Vision Pro, a experiência imersiva de mais alta qualidade atualmente disponível.

Este guia apresenta a Blackmagic URSA Cine Immersive e mostra como usá-la em filmagens, focando nos aspectos básicos da operação da câmera e no formato exclusivo criado para o Apple Immersive Video. Compreender o formato ajudará a entender melhor a câmera.

Nas páginas a seguir, você encontrará uma combinação de orientações técnicas e criativas que ajudarão não apenas a usar a câmera, mas também a compreender os benefícios do formato imersivo.

O que é Imersivo?

Imersivo é um formato de vídeo espacial tridimensional. Provavelmente você já sabe que outras tecnologias semelhantes existem há algum tempo. Para entender o que torna o imersivo diferente, segue uma visão geral de outras tecnologias:

- **Cinema 3D Tradicional** permite ao público perceber profundidade dentro do quadro ao captar cada cena com duas câmeras posicionadas a uma distância interaxial que imita a distância entre os olhos humanos. Embora as imagens estereoscópicas permitam ao público perceber profundidade, a perspectiva do espectador é controlada pela composição do diretor de fotografia.
- **Vídeo Esférico** combina os princípios das câmeras 3D e 360 graus, permitindo que o público olhe em qualquer direção enquanto percebe profundidade. As imagens em 360 graus exigem resoluções muito altas para assegurar que a pequena porção visualizada pelo espectador tenha fidelidade suficiente. Como a fisiologia humana só permite girar a cabeça de ombro a ombro, para olhar para trás o público geralmente permanece em pé ou sentado em uma cadeira giratória.
- **Vídeo Volumétrico** permite que o público não apenas perceba profundidade e olhe ao redor, mas também se mova e altere sua perspectiva. Embora matrizes de câmeras possam captar vídeo volumétrico, a maioria das experiências de realidade virtual ou aumentada faz uso de elementos 3D renderizados dinamicamente e exige um ambiente seguro para os participantes.

O imersivo combina o melhor do cinema tradicional e dos formatos 3D interativos. Em sua forma mais simples, é um formato estereoscópico de 180 graus com alta taxa de quadros e alta resolução.

Apple Immersive Video

A Blackmagic URSA Cine Immersive foi projetada especificamente para captar em Apple Immersive Video, formato usado com o Apple Vision Pro que oferece suporte a vídeo 3D estereoscópico de alta resolução em um campo de visão de 180 graus.

O Apple Immersive Video tem as seguintes características:

- Reprodução a 90 quadros por segundo.
- Pode ser distribuído em resoluções de até 4320 x 4320 por olho.
- Um sistema exclusivo de metadados para mapear a imagem original da câmera no ambiente de 180 graus.
- Apple Spatial Audio Format para fornecer som 3D.

O Apple Immersive Video requer um hardware capaz de reprodução estereoscópica de 180 graus em alta resolução. Atualmente, o Apple Vision Pro é o único dispositivo que oferece suporte a este formato.



Apple Vision Pro

O headset conta com duas telas micro OLED de 3660 x 3200 pixels com passo de pixel de 7,5 mícrons. Inclui 13 câmeras de rastreamento, um escâner LiDAR, unidades internas de medição e sensores de rastreamento ocular para alinhar as imagens com os movimentos da cabeça e dos olhos.

Um processador M2 com CPU de 8 núcleos e GPU de 10 núcleos processa a reprodução em até 100 quadros por segundo.

URSA Cine Immersive

A Blackmagic URSA Cine Immersive foi projetada para captar Apple Immersive Video para o Apple Vision Pro.



Blackmagic URSA Cine Immersive

A URSA Cine Immersive utiliza dois sensores RGBW de 58 megapixels combinados. Os sensores são quase quadrados para captar um campo de visão completo de 180 graus, tanto horizontal quanto verticalmente. Cada sensor é pareado com uma lente olho de peixe de 210 graus. Os centros das lentes ficam a 64 mm de distância.

Filtros ND integrados oferecem até 8 stops de controle de luz. A abertura das lentes é fixa em aproximadamente f4.5.

A câmera grava dois fluxos de imagem 8K a 90 quadros por segundo, resultando em taxas de dados significativamente mais altas do que uma gravação padrão em 4K 24p.

Primeiros Passos

Começar a usar sua Blackmagic URSA Cine Immersive é tão simples quanto ligar a câmera. O Blackmagic Media Module 8TB é inserido no compartimento do módulo da câmera, formatado e pronto para gravar clipes imediatamente.

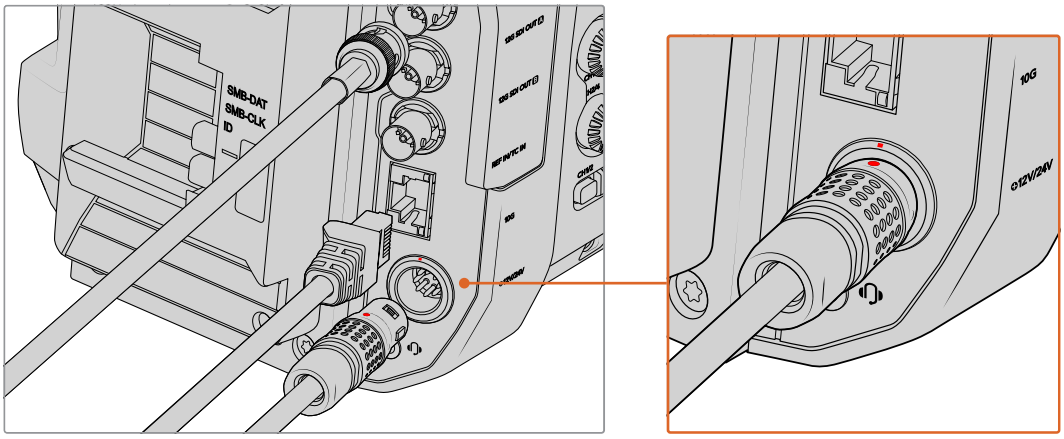
Ligar a Câmera

A maneira mais rápida de ligar a câmera é conectar a fonte de alimentação externa usando o adaptador de alimentação fornecido.

Para conectar alimentação externa:

1. Primeiro, retire a tampa antipoeira de borracha e guarde-a em um local seguro. Conecte o adaptador AC para DC de 24 V na tomada usando um cabo de alimentação IEC C13 padrão.
2. Encaixe o conector de 8 pinos do adaptador 24 V DC ao conector de alimentação no painel traseiro da câmera.

Se a alimentação externa e a bateria estiverem conectadas, a câmera usará a fonte externa. Remover a alimentação externa enquanto uma bateria carregada estiver conectada fará a câmera alternar para a bateria sem interrupção.



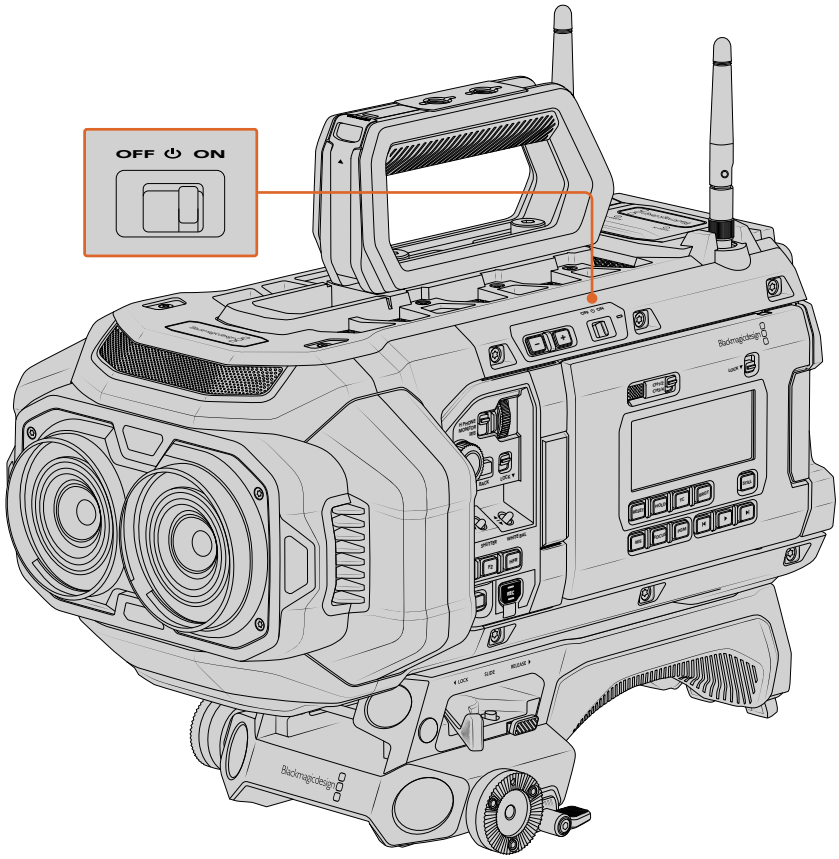
Para conectar a alimentação, alinhe os pontos vermelhos e empurre suavemente o conector até sentir que ele travou no lugar. Para remover, segure a capa metálica do conector e puxe-a delicadamente para fora da câmera. Isso fará a capa recuar, liberando o conector da tomada.

A placa de bateria incluída na câmera é uma placa de montagem B, projetada para baterias B-mount de 24 volts. Operar na faixa de 24 a 34 volts é ideal para a URSA Cine Immersive, pois a voltagem mais alta reduz perdas térmicas, permitindo gravações em altas taxas de quadros e o uso de múltiplos acessórios a partir da câmera.

Placas opcionais V-Lock e Gold Mount, compatíveis com operação em 24 V, além de placas Gold Mount Plus para baterias de terceiros de 26 V, também estão disponíveis.

OBSERVAÇÃO Ao operar a URSA Cine Immersive com fontes de 12 a 18 volts, não é possível alimentar acessórios via conector RS de 3 pinos ou conector estilo Lemo de 2 pinos.

Após instalar uma bateria ou conectar a alimentação externa, ligue a URSA Cine Immersive. Mova o interruptor de energia na parte superior da câmera para a posição “On”.



Mova o interruptor de energia para “On” para ligar a câmera.

Ajustar as Configurações de Gravação

Esta seção traz informações sobre configurações específicas da câmera que são exclusivas ou importantes ao filmar conteúdo imersivo. Para abrir o menu de configurações da câmera, toque no ícone de configurações no canto superior direito da tela retrátil ou da tela auxiliar.



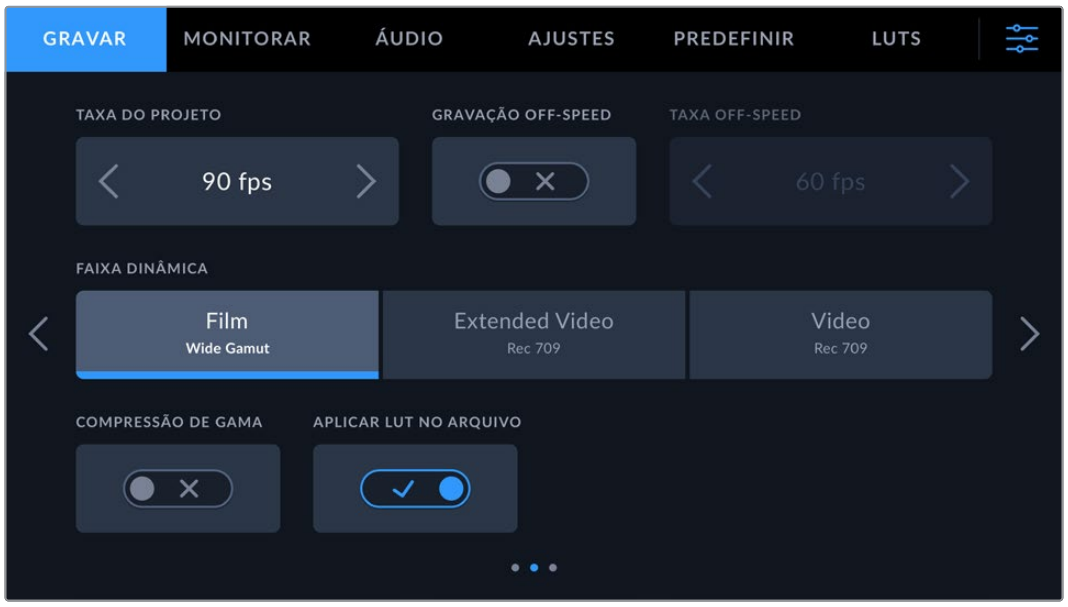
Configurações de Gravação 1

As opções de “Codec e Qualidade” nesta página permitem escolher a configuração do codec Blackmagic RAW que melhor se adapta ao seu projeto. As opções são divididas entre quatro configurações de “taxa de bits constante” e “qualidade constante”. É altamente recomendável utilizar as configurações de compressão 12:1 ou Q3 ao gravar conteúdo em 90 fps para o Apple Vision Pro. Essas configurações oferecem o equilíbrio ideal para manter taxas de dados e tamanhos de arquivo administráveis com o hardware atual, preservando ao mesmo tempo a altíssima qualidade da imagem.



Configurações de Gravação 2

A segunda página das configurações de gravação da câmera inclui os ajustes de taxa de quadros.



A taxa de quadros do projeto deve ser definida em 90 fps, em conformidade com a taxa de exibição do Apple Vision Pro.

Se estiver gravando para outros sistemas, taxas mais baixas, como 60 fps ou 50 fps, podem ser suficientes. Taxas de captura mais baixas no estilo de cinema tradicional, como 24 fps ou 25 fps, podem ser apropriadas se estiver gravando plates de efeitos visuais para conteúdo cinematográfico em 24p.

Esses são todos os passos necessários para começar a usar a URSA Cine Immersive. Com o módulo de mídia já formatado na câmera, é possível iniciar as gravações imediatamente.

Monitorar sua Imagem

A URSA Cine Immersive inclui várias opções para monitorar a imagem durante a gravação.

A tela retrátil é destinada ao uso do operador da câmera. Ela pode ser invertida para reduzir o perfil físico da câmera, mantendo a tela visível, ou virada para frente, permitindo o automonitoramento ao filmar a si mesmo.

A tela auxiliar possibilita que outro membro da equipe acompanhe a imagem sem obstruir a visão do operador. Devido ao campo de visão extremamente amplo da câmera, é necessário ter cautela ao usar a tela auxiliar.





Opções de Monitoramento e Sobreposições

A câmera exibe o espaço da lente, com apenas um olho visível por vez. As telas retrátil e auxiliar são mascaradas para mostrar uma visão de 180 graus. As saídas SDI exibem a imagem completa de 210 graus captada por cada sensor.



Todas as saídas de monitoramento exibem o mesmo olho. É possível alternar entre olho esquerdo e direito tocando no botão “L R” em qualquer uma das telas sensíveis ao toque.

Uma sobreposição opcional pode ser ativada nas telas de toque pressionando o botão de sobreposição. Essa sobreposição inclui:

- Um limite externo de 180 graus.
- Uma retícula central marcando o centro do quadro.
- Um guia interno representando o campo de visão aproximado em um Apple Vision Pro quando se olha para frente.

Essas ferramentas foram projetadas para auxiliar no enquadramento e posicionamento precisos durante a captura imersiva.

Enquadrar seu Primeiro Plano

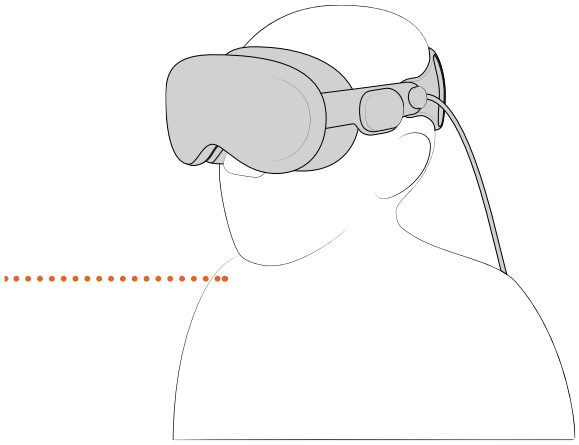
O conteúdo imersivo e o conteúdo 2D tradicional têm muito em comum, mas também apresentam diferenças importantes.

Por exemplo, já sentiu enjoo em um carro porque estava concentrado demais no celular? Isso é causado por um conflito sensorial, quando o cérebro recebe sinais contraditórios. O ouvido interno informa ao corpo que ele está em movimento, mas os olhos indicam que está parado.

É fácil reproduzir uma experiência semelhante com conteúdo imersivo. Por exemplo, se a câmera fizer uma panorâmica, os olhos do público dirão que estão se movendo, mas o pescoço e o ouvido interno indicarão que estão imóveis. Por isso, é recomendável dominar planos estacionários antes de aprender a tornar os movimentos de câmera confortáveis para o público.

Altura da Câmera

No cinema 2D tradicional, a câmera pode ser posicionada alta ou baixa para indicar o equilíbrio de poder em uma cena. Ao filmar em imersivo, considere o impacto de posicionar a câmera no chão. Como isso pode afetar o espectador? Pode transmitir a sensação de estar enterrado até o pescoço.



A altura percebida do piso pode influenciar a experiência de visualização do público. Uma câmera baixa pode gerar uma percepção de altura do chão que não faz sentido.



Ao filmar em imersivo, ângulos baixos podem causar desconforto ao público.

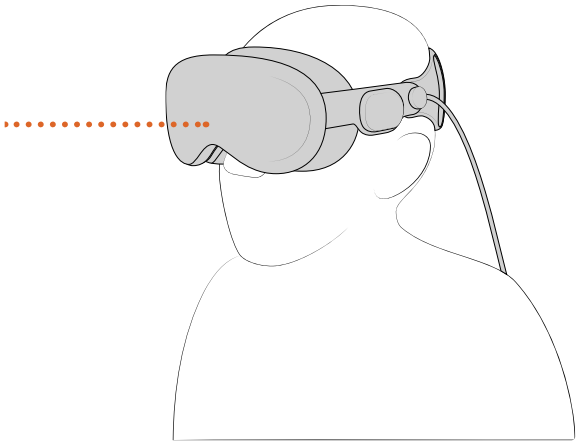
Para ajudar o espectador a se sentir fisicamente presente, é geralmente melhor escolher uma altura e posição de câmera que transmitam a sensação de estar em pé ou sentado na cena. Ainda há espaço para expressar a intenção criativa. Se houver pessoas na cena, uma câmera ligeiramente mais alta pode transmitir poder ao espectador, permitindo que ele olhe de cima para baixo. Por outro lado, uma câmera ligeiramente mais baixa força o espectador a olhar para cima.

Isso não quer dizer que não seja possível manter o conforto do público com posicionamentos mais ousados. Câmeras voadoras, vistas de cima para baixo, ângulos baixos e muitas outras técnicas já estão sendo aplicadas com sucesso em produções imersivas. Quando combinadas com edição, movimento e contexto adequados, podem entreter o público.

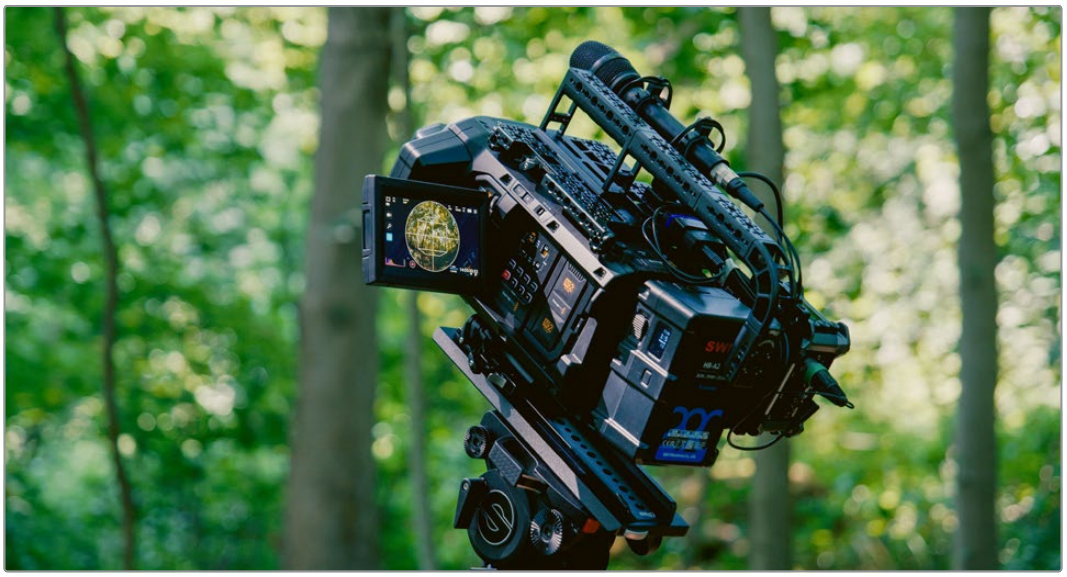
Inclinação da Câmera

Ao filmar conteúdo 2D tradicional e posicionar a câmera alta ou baixa, o diretor de fotografia normalmente usa a inclinação da câmera para manter a composição. Considere como isso afetaria uma imagem imersiva.

Se a câmera for inclinada para baixo, para que o público veja o horizonte, será necessário inclinar a cabeça para cima. Isso dificilmente causará enjoo, mas pode comprometer a sensação de presença, já que na vida real o horizonte está sempre à frente.



Os planos imersivos fazem mais sentido quando o horizonte está a zero grau.



Se a URSA Cine Immersive for inclinada, haverá um descompasso entre a posição em que o público deve olhar para ver o horizonte e a posição em que normalmente olharia.

Portanto, a maioria dos planos imersivos é captada com inclinação de zero grau ou próxima disso. Como a câmera capta um campo de visão de mais de 180 graus, tanto na horizontal quanto na vertical, o público ainda pode olhar para cima e para baixo. Mas, em vez de o cineasta forçar o público a olhar em uma direção específica, o controle fica com o espectador.

Para incentivar o público a observar diferentes partes da imagem, é possível usar outras técnicas, como iluminação, enquadramento com abertura e linhas de condução.

Posicionamento

O imersivo geralmente é assistido estando sentado. Como não é um formato de realidade aumentada ou virtual, os espectadores não ficam em pé nem se movimentam. Você está sentado agora? Como experimento, vire a cabeça de um lado para o outro e mova-a para cima e para baixo. Até onde é possível virar a cabeça confortavelmente sem mover o corpo? O campo de visão do formato imersivo ultrapassa a faixa confortável de rotação da maioria das pessoas, o que é positivo pois, ao girar para a esquerda ou para a direita, o formato ainda oferece visão periférica. Esse experimento ajuda a entender onde a ação principal deve ser posicionada dentro do campo de visão.

Faixa típica de movimento da cabeça:

Rotação: 120 a 160 graus
Flexão e extensão: 110 a 130 graus

Quando a rotação da cabeça é combinada com o movimento dos olhos, é possível olhar por cima do próprio ombro, mas manter essa posição é desconfortável para a maioria das pessoas.

Para auxiliar na composição, ative a sobreposição da câmera. Manter a ação dentro das marcas internas que mostram o campo de visão típico ajudará na visualização do conteúdo.

O movimento dos pontos de interesse dentro do quadro também deve ser considerado. Acompanhar uma pessoa caminhando da extrema esquerda até a extrema direita exige movimento extremo da cabeça. Da mesma forma, cortar de um plano em que o ponto de interesse está em uma borda da imagem para outro em que está na borda oposta também exige movimento extremo da cabeça. Considerar cuidadosamente a composição pode ajudar a manter o conforto do público sem perder a expressividade criativa.

Autonomia do Público

Ao filmar conteúdo 2D tradicional, o enquadramento, a escolha da lente e as configurações de foco podem ser usados para direcionar a atenção do espectador. Ao filmar em imersivo, o campo de visão de 180 graus dificulta excluir informações e direcionar a atenção do público. O imersivo dá ao público muito mais liberdade para explorar as imagens. Esse é um dos fatores que torna o conteúdo imersivo tão envolvente. Ainda assim, é possível conduzir a atenção do público com uso criterioso de som direcional, iluminação, posicionamento e composição.

Movimento da Câmera

Uma câmera imersiva não precisa permanecer estática, mas é preciso ter cuidado, pois alguns movimentos podem parecer artificiais ou desconfortáveis. O carro é uma boa analogia para entender como tornar o movimento de câmera imersiva confortável. Um veículo grande tem massa e inércia que resistem a mudanças bruscas de velocidade ou direção. Assim, ele acelera e desacelera gradualmente e se move de maneira previsível e controlada. Após dominar os planos estáticos e passar para planos em movimento, considerar esses fatores ajudará a criar um movimento agradável:

- **Duração:** dar tempo ao público para se ajustar ao movimento da câmera e apreciá-lo. Movimentos breves têm mais chance de causar desorientação.
- **Intensidade:** movimentos de baixa intensidade são mais fáceis de integrar em uma edição imersiva. Movimentos intensos são possíveis, mas um aquecimento e uma desaceleração podem ajudar no conforto do público.
- **Âncoras visuais:** são partes da cena que não se movem. Um plano sem âncoras visuais pode ser desorientador. Elementos de primeiro plano, como o interior de um carro, ou objetos distantes, como o horizonte, ajudam a dar estabilidade aos planos em movimento.

Tabelas de Taxas de Dados

As tabelas abaixo contêm taxas de dados para conteúdo imersivo 8K x 7K a 90 quadros por segundo e 24 quadros por segundo.

Taxas Máximas de Dados em 90 fps com Blackmagic RAW Constant Bitrate

8K x 7K Imersivo			
Codec	**Dimensões dos Pixels**	**Taxa de Quadros**	**Taxa de Dados**
Blackmagic RAW 5:1	8.160 x 7.200 Estéreo	90	3.203 MB/s
Blackmagic RAW 8:1	8.160 x 7.200 Estéreo	90	2.002 MB/s
Blackmagic RAW 12:1	8.160 x 7.200 Estéreo	90	1.335 MB/s
Blackmagic RAW 18:1	8.160 x 7.200 Estéreo	90	889 MB/s

Taxas Máximas de Dados em 24 fps com Blackmagic RAW Constant Bitrate

8K x 7K Imersivo			
Codec	**Dimensões dos Pixels**	**Taxa de Quadros**	**Taxa de Dados**
Blackmagic RAW 5:1	8.160 x 7.200 Estéreo	24	854 MB/s
Blackmagic RAW 8:1	8.160 x 7.200 Estéreo	24	534 MB/s
Blackmagic RAW 12:1	8.160 x 7.200 Estéreo	24	356 MB/s
Blackmagic RAW 18:1	8.160 x 7.200 Estéreo	24	237 MB/s

Áudio e Código de Tempo

O áudio é uma parte essencial da experiência imersiva. É importante usar sua URSA Cine Immersive em conjunto com equipamentos dedicados de captura de áudio.

Formato de Áudio Espacial da Apple

O Apple Spatial Audio Format, ou ASAF, é usado para fornecer áudio 3D ao assistir a conteúdos Apple Immersive Video. O ASAF faz parte da Apple Positional Audio Framework e pode armazenar três diferentes tipos de gravação de áudio.

- **Baseado em canais:** é possível incorporar uma mixagem surround tradicional em seu arquivo ASAF. Isso é ideal para elementos como narrações fixas na cabeça e música.
- **Baseado em objetos:** os sons podem ser tratados como objetos individuais com metadados que descrevem sua posição no espaço 3D, independentemente dos canais de alto-falantes. Normalmente, são gravações mono. Suas coordenadas 3D podem ser animadas para acompanhar objetos que se movem dentro da cena. Isso é ideal para áudio diegético, que se refere a objetos visíveis na cena.
- **Ambissônico:** um formato de esfera completa que pode reproduzir áudio em todas as direções usando harmônicos esféricos. É ideal para sons ambientais.

O restante desta seção fornece uma visão geral dos equipamentos necessários para criar um arquivo ASAF eficiente.

Ambissônico

Um microfone ambissônico contém quatro ou mais cápsulas dispostas em um padrão geométrico preciso. Cada cápsula captura os mesmos sons com diferentes tempos de chegada e intensidades, dependendo da direção da origem sonora. Essas cápsulas são gravadas em canais discretos para posterior processamento por computador. Comparando essas diferenças, o software pode estimar a direção e a natureza dos sons gravados.



Um microfone ambissônico de primeira ordem.

Microfones Ambissônicos

Os microfones ambissônicos são classificados pelo número de cápsulas que contêm. Mais cápsulas significam maior precisão na reprodução dos detalhes espaciais.

Primeira ordem: 4 canais

Segunda ordem: 9 canais

Terceira ordem: 16 canais

Quarta ordem: 25 canais ou mais

Seja montado na câmera ou ao lado dela, é fundamental que os microfones ambissônicos estejam corretamente alinhados com a direção de filmagem da câmera. Além disso, como os microfones ambissônicos podem ser usados em diferentes configurações, como end-fire, broad-side, voltados para cima ou para baixo, é importante garantir que o técnico de áudio saiba qual configuração foi utilizada. Essas informações podem ser transmitidas via metadados no gravador de áudio ou por meio de anotações.

Gravações ambissônicas podem ser usadas como base de ambiente. Outros efeitos sonoros baseados em canais ou objetos podem ser sobrepostos. Se corretamente alinhadas com a câmera, as gravações ambissônicas também podem servir de referência para o engenheiro de áudio ao adicionar mais elementos de design sonoro.

Microfones de Lapela e Miniatura

Ao filmar com uma câmera de 180 graus, é impossível usar uma vara boom, pois ela ficará visível na cena. Isso significa que será necessário usar microfones ocultos e de lapela para gravar áudio diegético e diálogos.

Por exemplo, é possível:

- Conectar essas fontes de áudio diretamente às duas entradas XLR da URSA Cine Immersive usando transmissores e receptores sem fio.
- Conectá-las a um gravador de áudio externo. Esse gravador pode ser um equipamento multicanal de grande porte ou um dispositivo miniatura de uso corporal.

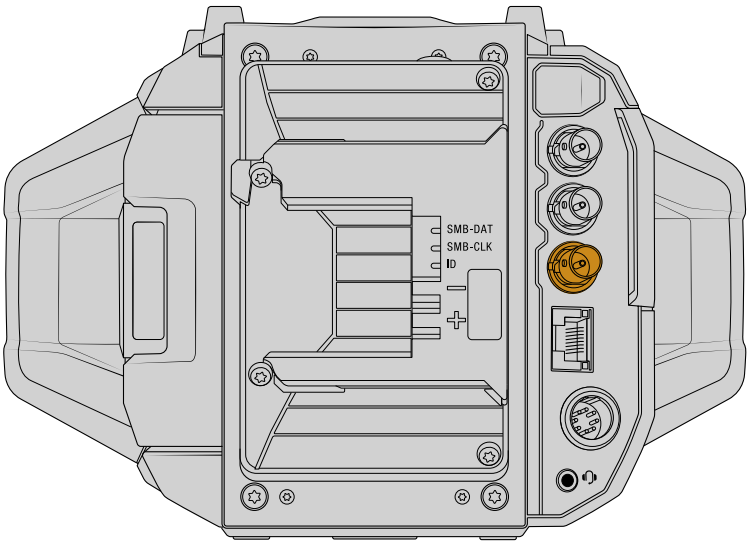
Essas gravações podem ser sincronizadas e mixadas como sons baseados em canais ou em objetos.

Microfones Portáteis

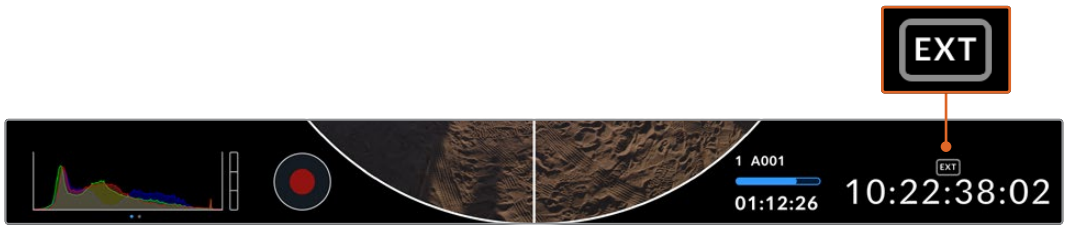
Gravar efeitos sonoros adicionais no local após a câmera parar de gravar não apenas economiza tempo na pós-produção, mas também resulta em um som mais autêntico. Levar um microfone hipercardioide portátil ou similar durante a produção é muito benéfico, permitindo repetir ações com o microfone posicionado o mais próximo possível. Essas gravações podem então ser editadas na mixagem final como sons baseados em canais ou em objetos.

Conectar o Código de Tempo

Para conectar o código de tempo externo à URSA Cine Immersive, use um cabo BNC para ligar um gerador de código de tempo à entrada de código de tempo no painel traseiro da câmera. A conexão a um código de tempo sincronizado ao filmar em sistema duplo garante que áudio e imagem possam ser facilmente sincronizados na pós-produção.



Quando um código de tempo externo com taxa de quadros correspondente é conectado a essa entrada, a câmera se ajustará automaticamente ao código recebido. O logotipo “EXT” aparecerá ao lado do contador de código de tempo nos LCDs da URSA Cine Immersive quando o código externo estiver bloqueado. Se o cabo for desconectado, o código de tempo permanecerá sincronizado e o logotipo “EXT” mudará para “INT”, indicando que agora está sendo executado a partir do relógio do código de tempo da câmera.



O código de tempo da câmera mostrará “EXT” quando conectado a um gerador de código de tempo.

OBSERVAÇÃO Existem padrões SMPTE de código de tempo para taxas de quadros típicas, como 23.976, 24, 30, etc. Não existe um padrão SMPTE de código de tempo para 90 quadros por segundo. Assim, quando a taxa de quadros do projeto da URSA Cine Immersive é definida em 90 quadros por segundo, a câmera usa na prática o código de tempo de 30 fps. As saídas SDI também operam em 30p. Qualquer acessório de código de tempo ou gravador de áudio externo deve ser configurado para 30 fps.

Filmagem

Agora que já há uma compreensão básica de como operar a câmera Blackmagic URSA Cine Immersive e aproveitar ao máximo o formato imersivo, este capítulo explorará mais detalhadamente a operação da câmera.

Requisitos de Exposição

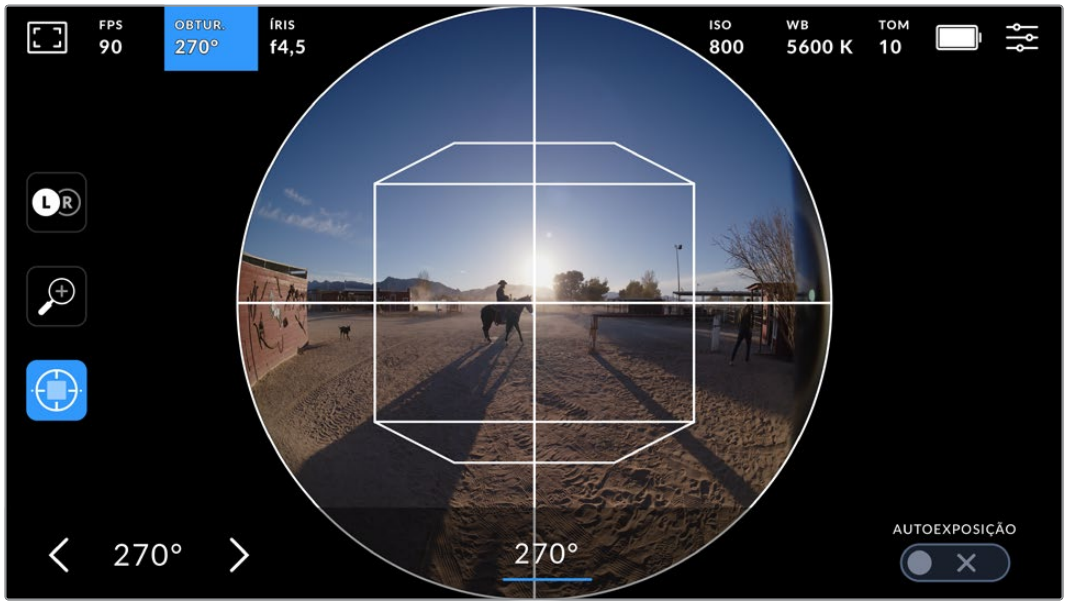
O Apple Immersive Video exige mais exposição do que a captura tradicional em 24 fps e pode exigir repensar a forma de filmar e iluminar. Mantendo um obturador de 180 graus, a captura em 90 fps exige quase quatro vezes mais luz para expor a mesma quantidade. Além disso, a íris fixa das lentes, em torno de f4.5, é até três stops mais escura que as lentes super rápidas típicas. Isso significa que pode ser necessário até sete stops de luz a mais para obter exposição adequada. No entanto, ao filmar em 90 fps, usar um ângulo de obturador amplo entre 270° e 360° pode ajudar a compensar a perda de luz causada pela alta taxa de quadros, mantendo uma quantidade agradável de desfoque de movimento.

Velocidade do Obturador

O indicador “Obturador” exibe o ângulo ou a velocidade do obturador. A configuração “Medida do Obturador” no menu “Ajustes” pode ser usada para selecionar se as informações devem ser exibidas como “Ângulo” ou “Velocidade”.

Para alterar a velocidade do obturador:

1. Toque no ícone “Obturador” na barra de ferramentas superior.
2. Use as setas esquerda ou direita para alternar entre predefinições comuns de velocidade do obturador. Para selecionar um valor específico, basta tocar duas vezes no indicador de obturador no canto inferior esquerdo da tela. Isso abrirá um teclado numérico que permite inserir qualquer valor entre 11,2 e 360 graus, com até duas casas decimais.



Taxas de quadros personalizadas podem ser inseridas usando o teclado na tela.

A velocidade do obturador tem menos impacto na aparência do conteúdo imersivo de alta taxa de quadros. Embora o obturador de 180 graus ainda seja preferido por muitos ao filmar em 24 ou 30p, é mais comum usar a velocidade do obturador para auxiliar na exposição em filmagens imersivas.

Ao filmar em pouca luz, o obturador pode ser ampliado para 270 graus ou, em casos excepcionais, para 360 graus.

Em condições de muita luz, a velocidade do obturador pode ser reduzida até 11,2 graus. No entanto, em condições de muita luz, é mais comum manter a velocidade do obturador no ajuste preferido para o movimento e usar filtros ND para controlar a exposição.

Vale destacar que a alta taxa de quadros do Apple Immersive Video, de 90 quadros por segundo, pode causar problemas ao filmar com luzes que piscam. Um obturador de 270 graus é sugerido ao trabalhar em ambientes de 60 Hz e um obturador de 324 graus é recomendado em ambientes de 50 Hz.

ISO e Faixa Dinâmica

O indicador “ISO” exibe a configuração atual de ISO ou de sensibilidade à luz da câmera. Tocar nesse indicador permite ajustar o ISO de acordo com diferentes condições de iluminação. A faixa de ISO da URSA Cine Immersive vai de 200 a 3200. O ideal é 800.



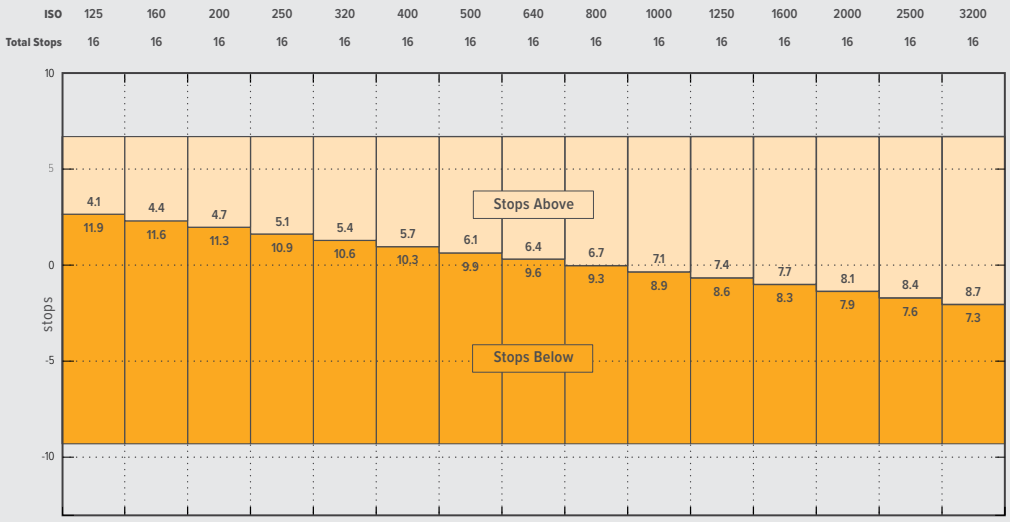
O ISO pode ser configurado tocando no indicador de ISO.

Predefinições numéricas permitem selecionar uma configuração de ISO em incrementos de stop completos tocando no número. O controle deslizante abaixo das predefinições permite ajustar finamente o ISO entre essas predefinições em incrementos de 1/3 de stop.

Dependendo da situação, pode ser escolhido um ISO mais baixo ou mais alto. Por exemplo, em condições de pouca luz, ISO 1600 pode ser adequado, mas pode introduzir algum ruído visível. Em condições de luz intensa, o ISO 200 pode proporcionar cores mais vibrantes.

As câmeras Blackmagic conseguem captar em ampla faixa dinâmica, oferecendo bastante flexibilidade tanto na captura quanto na gradação de cores no DaVinci Resolve. Confira nesta tabela a faixa dinâmica disponível na URSA Cine Immersive.

Faixa Dinâmica da URSA Cine Immersive



ND

Os filtros ND da câmera são normalmente a forma mais comum de controlar a exposição, permitindo reduzir a quantidade de luz que atinge o sensor por um número predefinido de stops de exposição.



Os filtros ND da URSA Cine Immersive são ajustados usando os botões no lado esquerdo, acima da tela de toque retrátil. Junto com um filtro claro, as opções disponíveis são de “2”, “4”, “6” e “8” stops.

A configuração atual de ND é exibida no indicador no canto superior esquerdo da tela. Quando o ND está ativo, esse indicador permanece exibido permanentemente. Quando o ND é definido como claro, o indicador desaparece após 4 segundos.



OBSERVAÇÃO É possível alterar a terminologia de ND para refletir as convenções preferidas. Use o menu “Ajustes” para selecionar entre número de ND, stops ou frações.

Gravação Off-Speed

A taxa de quadros do projeto no menu “Ajustes” define a taxa de quadros de gravação da URSA Cine Immersive e determina a velocidade de reprodução do conteúdo. Normalmente, essa configuração é ajustada para coincidir com a velocidade de reprodução e a sincronização de áudio usadas no fluxo de pós-produção e nos requisitos de entrega.



Toque no indicador de quadros por segundo da URSA Cine Immersive para acessar as configurações de taxa de quadros. A URSA Cine Immersive tem oito configurações de taxa de quadros de projeto: 23.98, 24, 25, 29.97, 30, 50, 59.94, 60 e 90 quadros por segundo. Ao filmar em formato imersivo, a taxa de quadros do projeto deve ser configurada em 90 quadros por segundo.

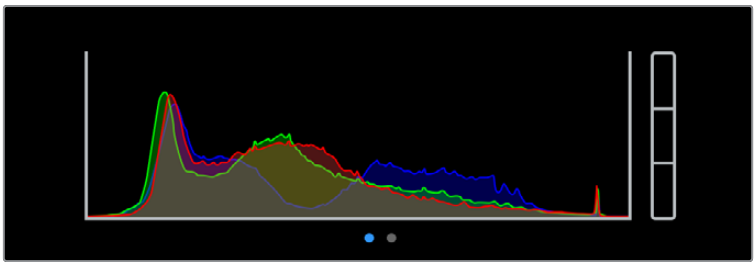
A taxa de quadros do sensor permite definir a velocidade na qual o sensor captura uma imagem. Por padrão, as taxas de quadro do projeto e do sensor da URSA Cine Immersive são iguais, para reprodução natural. No entanto, ao tocar no ícone de ativação de “Taxa Off-Speed”, no canto inferior direito do menu “FPS” da câmera, é possível configurar a taxa de quadros do sensor de forma independente.

Quando a opção “Taxa Off-Speed” está ativada, a taxa de quadros do sensor pode ser ajustada manualmente. Vale destacar que 90 quadros por segundo é a taxa máxima da câmera e corresponde à taxa de quadros do projeto quando se filma para o Apple Vision Pro. Entretanto, selecionar uma taxa de projeto mais baixa, como 30 ou 60 quadros, e filmar em off-speed a 90 quadros por segundo, permite criar belíssimos vídeos em câmera lenta.

Enquanto a opção “Taxa Off-Speed” está ativada, o indicador de taxa de quadros no canto superior esquerdo da tela exibirá a taxa de quadros do sensor seguida pela taxa de projeto.

Histograma

No canto inferior esquerdo da tela da URSA Cine Immersive, aparece o histograma. Um histograma é um gráfico que mostra a distribuição de brilho em uma imagem. O lado esquerdo representa as sombras, o meio mostra os tons médios e o lado direito indica os realces. A altura do gráfico mostra quanto da imagem está em cada nível de brilho.



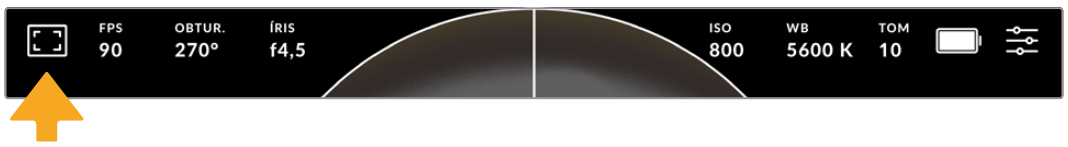
O histograma oferece uma indicação rápida da distribuição tonal do clipe. Uma imagem com o gráfico à esquerda provavelmente está subexposta. Da mesma forma, uma imagem com o gráfico à direita provavelmente está superexposta. O histograma deve sempre ser interpretado em contexto com a imagem, mas, em geral, uma imagem bem exposta apresenta um gráfico equilibrado no centro. Aumentar a exposição desloca o histograma para a direita, enquanto reduzir a exposição o desloca para a esquerda.

À direita do histograma há um indicador que mostra quando canais de cor individuais estão estourando. Normalmente, deve-se evitar cortar canais. Entretanto, existem situações em que é aceitável cortar a imagem, se isso ajudar a preservar os meios-tons e sombras.

“Expor à Direita” é uma técnica em que a imagem é deliberadamente superexposta, empurrando o histograma o máximo possível para a direita, sem cortar os canais de cor. Isso maximiza o sinal e minimiza o ruído. Desde que a exposição seja reduzida na pós-produção, essa é uma excelente forma de produzir imagens limpas.

Opções de Monitoramento LCD

Toque no ícone de monitor no canto superior esquerdo da tela LCD da URSA Cine Immersive para acessar as configurações do monitor LCD. Este menu contém opções úteis, como brilho do monitor, bem como ferramentas avançadas de exposição.



Zebra

A configuração “Zebra” ativa ou desativa a exibição de zebra na tela, além de definir o nível de zebra. A exibição de zebra exibe linhas diagonais sobre áreas da imagem que ultrapassam um nível de exposição definido. A configuração em porcentagem é usada para mostrar partes da imagem superexpostas. É comum definir esse valor em torno de 95%, em vez de 100%, pois isso permite identificar áreas da imagem em que a superexposição ainda pode ser corrigida. A segunda configuração de zebra permite ativar ou desativar zonas de zebra em cinza médio ou cinza médio mais 1 stop.

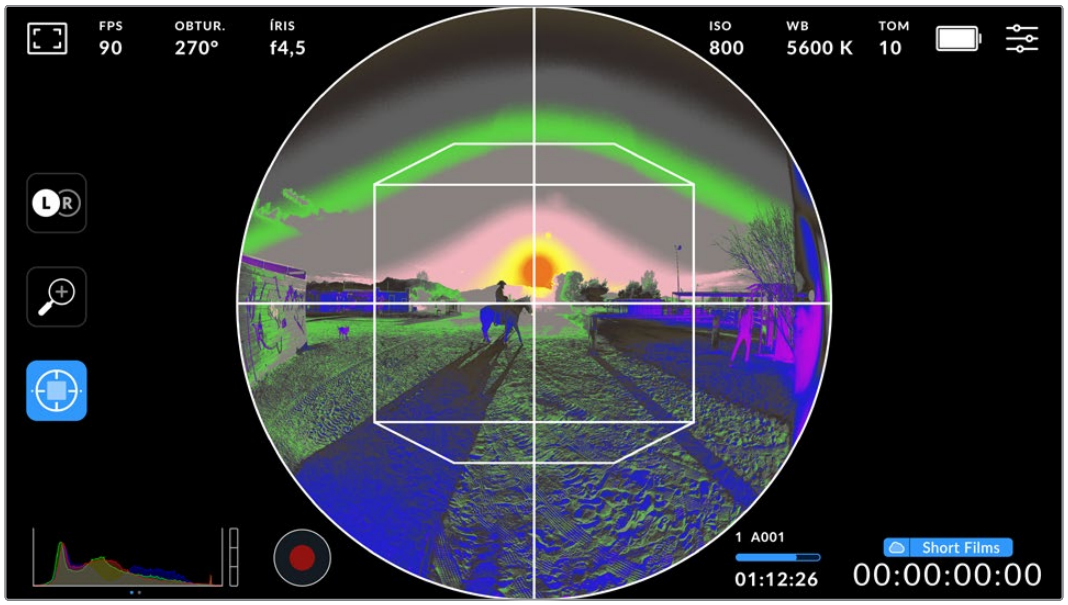
Grades

Na opção “Grades”, é possível ativar ou desativar a exibição de uma grade de regra dos terços, indicador de horizonte, retícula ou ponto central na tela LCD, além de definir a sobreposição visível em todas as saídas da URSA Cine Immersive. Ao filmar conteúdo imersivo, o indicador de horizonte é muito útil. Ele exibe uma retícula dupla no centro da imagem que permanece alinhada enquanto a câmera estiver completamente nivelada, mas se desvia quando a câmera não está.

Cor Falsa

O recurso “Cor Falsa” sobrepõe diferentes cores na imagem, representando valores de exposição para diferentes elementos. Por exemplo, rosa representa exposição ideal para tons de pele mais claros, enquanto verde é adequado para tons de pele mais escuros. Ao monitorar a cor falsa rosa ou verde durante gravações de pessoas, é possível manter uma exposição consistente para seus tons de pele.

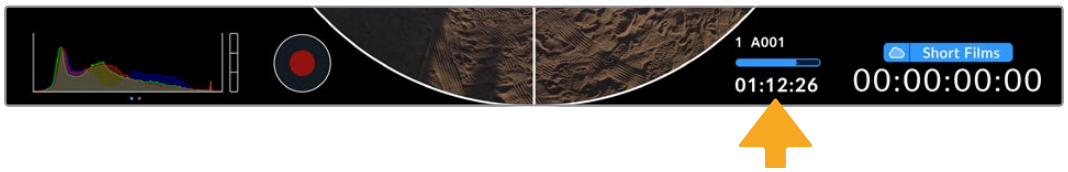
Da mesma forma, quando elementos mudam de amarelo para vermelho, significa que estão superexpostos. Assim como o histograma, essa ferramenta fornece uma compreensão rápida e intuitiva da distribuição tonal da imagem.



Gravação

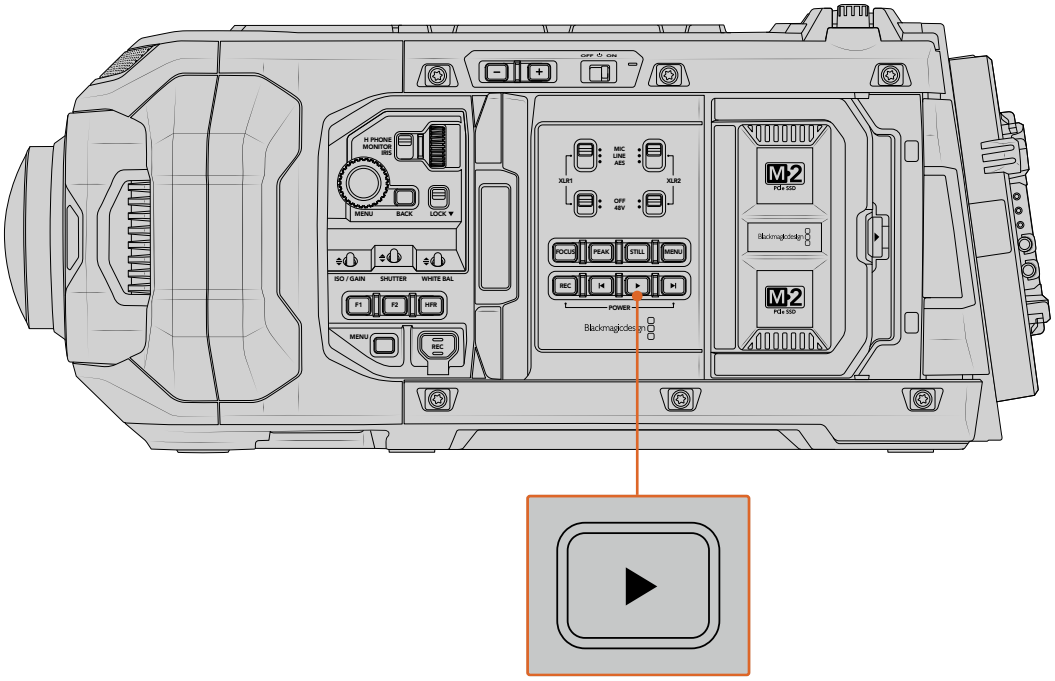
A gravação pode ser iniciada pressionando qualquer um dos botões vermelhos de gravação na câmera. Os botões de gravação estão localizados no painel de controle frontal, no painel interno atrás da tela retrátil, no painel da estação assistente e em ambas as telas LCD. Pressione novamente o botão de gravação para parar de gravar.

Os indicadores de armazenamento mostram quanto tempo de gravação resta. O tempo é exibido em horas:minutos:segundos, de acordo com a taxa de quadros escolhida. O ícone da barra de mídia acima do tempo de gravação pode estar azul, branco ou vermelho, dependendo do status atual, e exibe o espaço usado no armazenamento.



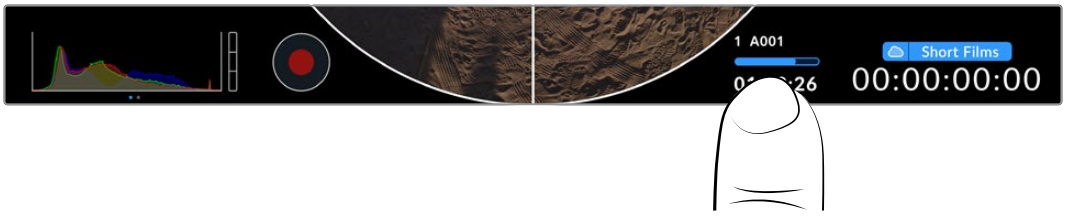
Reprodução

Depois de gravar os clipes, é possível usar os botões de transporte para reproduzi-los. Para reproduzir o último clipe gravado, pressione o botão de reprodução no lado esquerdo da câmera, localizado atrás da tela retrátil. Os botões de anterior e próximo podem ser usados para navegar entre outros clipes. Para sair do modo de reprodução e preparar a câmera para continuar gravando, pressione o botão de gravação uma vez.

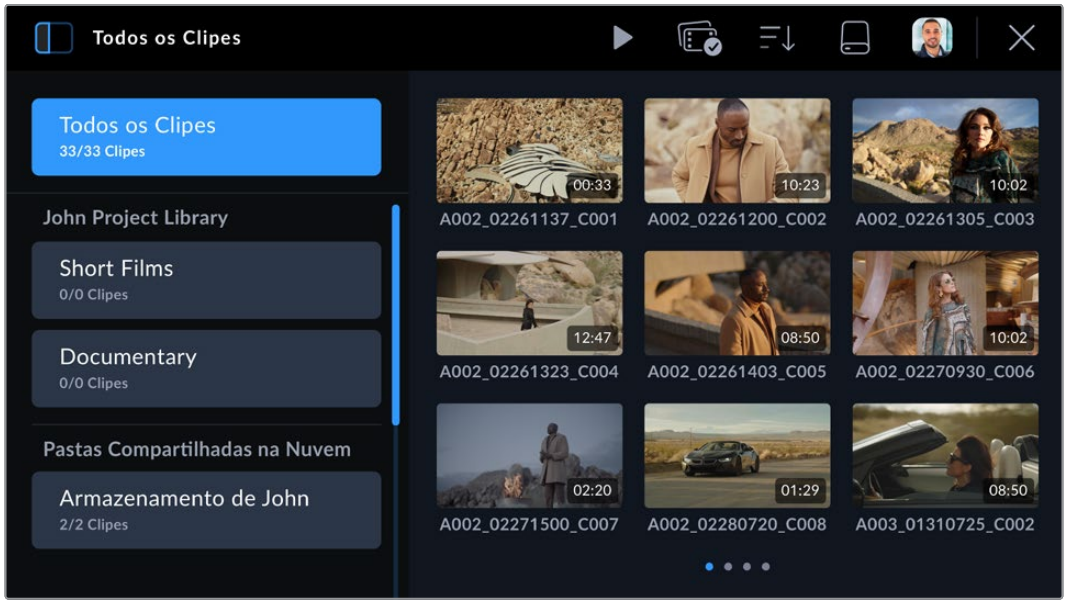


Pressione os botões de avanço e retrocesso para pular para o início ou fim dos clipes. Pressione o botão de retrocesso uma vez para ir ao início do clipe atual ou duas vezes para voltar ao início do clipe anterior. Mantenha pressionado o botão de avanço ou retrocesso para reproduzir ou voltar em velocidade 2x. Após avançar ou retroceder, pressione os botões de avanço rápido ou retrocesso duas vezes para 4x, três vezes para 8x e quatro vezes para 16x.

Se for necessária uma revisão mais extensa do material filmado, provavelmente será preferível usar o pool de mídia. Para abrir o pool de mídia, toque em um dos indicadores de armazenamento na parte inferior da tela de toque da câmera.



O navegador da página principal exibe miniaturas de todos os clipes gravados em todas as mídias conectadas à câmera. Há três linhas de miniaturas e, conforme mais clipes são adicionados, é possível deslizar para a próxima página de miniaturas usando a tela de toque. Indicadores de página na parte inferior da tela mostram quantas páginas de clipes estão disponíveis.



Conecte a URSA Cine Immersive a uma conta Blackmagic Cloud e envie material diretamente para um projeto DaVinci Resolve.

Para sair do pool de mídia a qualquer momento, toque no X no canto superior direito ou pressione qualquer um dos botões de gravação da câmera uma vez.

Transferir Clipes

Existem várias maneiras de transferir clipes do armazenamento da câmera. Por exemplo, é possível conectar a URSA Cine Immersive ao computador ou a uma rede via cabo Ethernet de 10G e copiar os clipes do armazenamento da câmera via Ethernet. Também é possível inserir o módulo de mídia em uma Blackmagic Media Dock conectada à rede ou ao computador, permitindo trabalhar com até três módulos de mídia simultaneamente.

Ao conectar ao computador via Ethernet, este reconhecerá o Blackmagic Media Module da mesma forma que ao conectar um Blackmagic Cloud Store.

A câmera vem configurada para usar DHCP por padrão, portanto o computador ou a rede reconhecerá a câmera e atribuirá um endereço IP automaticamente.

Habilitar o Compartilhamento de Arquivos

Ao configurar a câmera pela primeira vez e conectá-la ao computador ou rede via Ethernet, o compartilhamento de arquivos SMB estará desabilitado por padrão, porque, ao contrário de um Blackmagic Cloud Store, em que geralmente se deseja compartilhar arquivos imediatamente com outros usuários da rede, pode não ser interessante permitir o mesmo nível de acesso aos arquivos principais da câmera logo após um dia importante de filmagem. Por esse motivo, foi decidido que o compartilhamento de arquivos SMB deve ser habilitado manualmente.

1. Para habilitar o compartilhamento de arquivos, será necessário baixar e instalar o Blackmagic Camera Setup no computador. Esse utilitário de configuração é um software administrativo que permite atualizar a câmera e alterar as configurações de administração. O Blackmagic Camera Setup pode ser baixado em nosso site: https://www.blackmagicdesign.com/br/support/family/professional-cameras
2. Após baixar e instalar o utilitário, conecte a câmera ao computador via USB e inicie o Blackmagic Camera Setup. Conecte usando a porta USB superior, localizada na parte traseira da câmera.

3 Para abrir as configurações, clique na imagem da URSA Cine Immersive ou no ícone de configurações correspondente.

4 Role para baixo até as configurações de “Network Access” e habilite “file sharing (SMB)”. Em seguida, clique em “Save” no canto inferior direito para aplicar as alterações.

5 Agora, acesse a configuração de URL logo abaixo e clique no ícone de URL para preencher automaticamente o endereço. O endereço de URL exibirá o nome da câmera como uma localização de rede local.

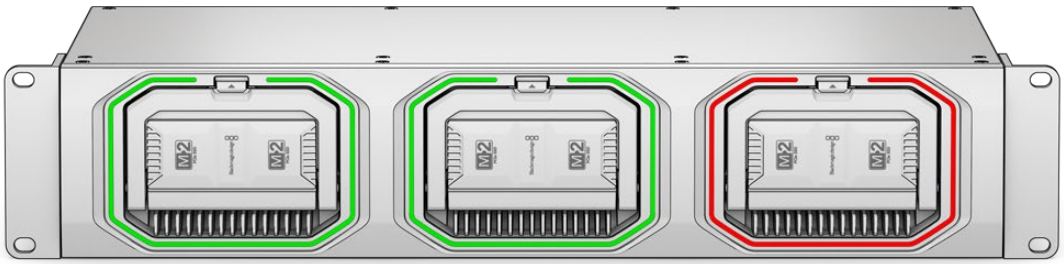
6 Clique em “Save”.

Para acessar diretamente os clipes, clique no nome de URL da câmera. Uma janela será aberta pedindo senha ou conectar-se como convidado. Clique na opção “Guest” e clique em “Connect”.

7 O computador navegará automaticamente até o módulo de mídia no Finder (Mac) ou no Explorador de Arquivos (Windows). O utilitário lembrará a configuração Ethernet, portanto, isso só precisa ser feito uma vez durante a configuração inicial. Da próxima vez que a câmera for conectada via Ethernet, o módulo de mídia poderá ser acessado imediatamente, sem necessidade de alterar essas configurações.

Usar a Blackmagic Media Dock

O Blackmagic Media Module foi projetado para ser rapidamente inserido e removido quando for necessário transferir clipes para armazenamento externo ou trocar por um módulo vazio para continuar gravando. A Blackmagic Media Dock pode ser usada para acessar o material de um Media Module sem a câmera.



Resumo

O Apple Immersive Video traz uma nova abordagem à narrativa visual e oferece uma alternativa ao cinema 2D tradicional. O intuito não é substituí-lo, mas sim a servir como um formato adicional a ser escolhido de acordo com as necessidades do projeto. Assim como em qualquer formato, as escolhas criativas variam de acordo com a história a ser contada.

Embora novos formatos geralmente envolvam fluxos de trabalho complexos, a URSA Cine Immersive foi projetada para simplificar o processo de captura do Apple Immersive Video. Ela oferece um caminho de produção direto para conteúdo imersivo e também pode ser usada para efeitos visuais e outras aplicações.

Este guia descreve os principais recursos e fluxos de trabalho da URSA Cine Immersive. É um ponto de partida. Assim como em qualquer sistema de câmera, recomenda-se a exploração e os testes adicionais para alcançar os melhores resultados de acordo com as necessidades específicas da produção.

DaVinci Resolve Studio



O DaVinci Resolve Studio é um software de pós-produção completo que integra perfeitamente edição, correção de cores, efeitos visuais, gráficos em movimento e pós-produção de áudio. Com suporte ao formato Blackmagic RAW Immersive, o software disponibiliza fluxos de trabalho adaptados às características únicas do formato, como vídeo estereoscópico em resolução ultra-alta, taxas de quadros de 90 fps e mixagem de Áudio Espacial. O DaVinci Resolve Studio tem todos os recursos necessários para produção, monitoramento e entrega no formato Apple Immersive Video, o que o torna uma ferramenta poderosa para criadores de conteúdo imersivo.

Recomendações de Estação de Trabalho

Recomendações de Hardware

- Desktop Mac Studio M3 Ultra
- Monitor com alta taxa de atualização (mínimo de 60 Hz)

Requisitos Mínimos do Sistema

- macOS 15 ou posterior
- 64 GB de memória do sistema
- Blackmagic Design Desktop Video 12.9 ou posterior (para monitoramento)
- Computador com Apple Silicon ou GPU compatível com Metal

Requisitos de Monitoramento Imersivo

- Apple Vision Pro
- Rede sem fio de alta velocidade (mínimo Wi-Fi 5; preferencialmente Wi-Fi 6)

Ajuste de Projeto

Antes de importar mídias RAW da Blackmagic URSA Cine Immersive, é recomendável conferir a seção Definições de Projeto do DaVinci Resolve para garantir que a mídia seja corretamente decodificada e processada para obter o melhor resultado visual e facilitar a edição.

Definições Máster

É fundamental indicar que o conteúdo será criado para o Apple Vision Pro ao configurar um projeto, de modo a garantir que todas as configurações corretas e opções de fluxo de trabalho estejam visíveis ao preparar, montar e entregar o projeto.

1. Inicie o DaVinci Resolve Studio.
2. No canto inferior direito da janela Gerente de Projeto, clique em “Novo Projeto”.
3. Digite um nome para o projeto, defina a localização da mídia (se necessário) e clique em “Criar”.
4. Clique no ícone de engrenagem no canto inferior direito do DaVinci Resolve Studio para abrir o menu Definições de Projeto.

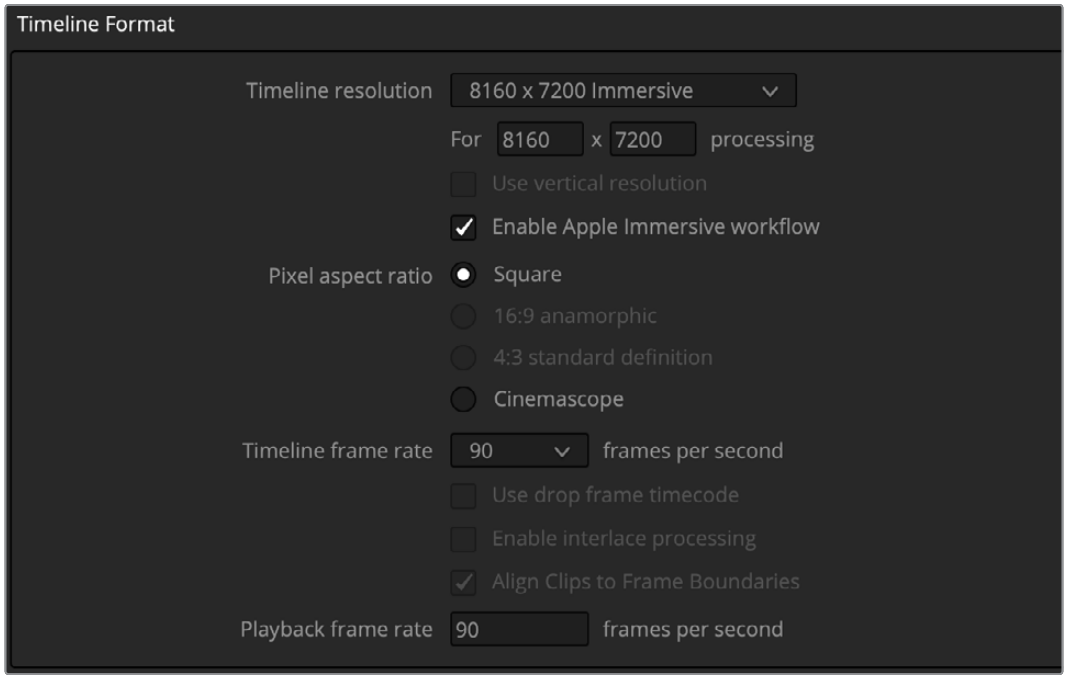
   Primeiro, é preciso definir a resolução e a taxa de quadros do projeto.
5. Selecione “Definições Máster” na barra lateral.
6. Na seção Formato da Linha de Tempo, selecione “Ativar fluxo de trabalho Apple Immersive”.



   O DaVinci Resolve será ajustado automaticamente em vários parâmetros para se adequar de forma ideal aos fluxos de trabalho imersivos. Ao criar um projeto imersivo, certifique-se de selecionar sempre esta configuração primeiro.
7. Em “Resolução da Linha de Tempo”, selecione “8160 x 7200 Immersive” para processar os clipes na resolução nativa do sensor.

   Se estiver trabalhando com recursos de processamento limitados, escolha a opção “4080 x 3600 Immersive” para reduzir a resolução pela metade e obter uma reprodução mais rápida.
8. Defina “Taxa de quadro da linha de tempo” para 90 quadros por segundo. Esse parâmetro é configurado automaticamente ao habilitar fluxos de trabalho Apple Immersive.
9. Defina a “Taxa de quadro de reprodução” para 90 fps a fim de garantir a reprodução mais precisa do vídeo imersivo.

   Se estiver trabalhando com recursos de processamento limitados (ou com um monitor com taxa de atualização mais baixa), reduza a taxa de quadros para uma reprodução mais estável. É recomendável escolher uma taxa que seja uma fração da taxa nativa, como 30 fps ou 60 fps.



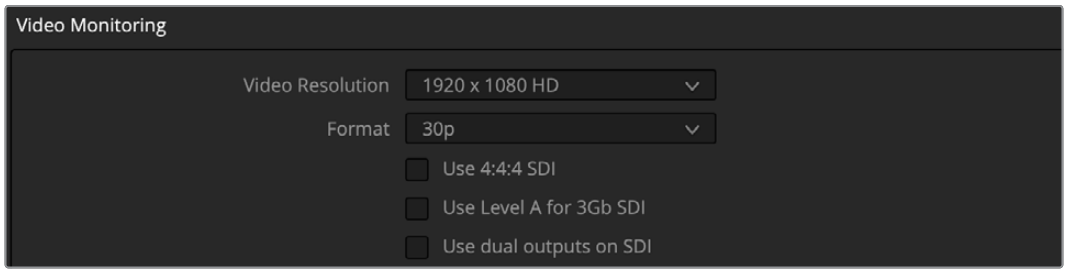
Monitoramento de Vídeo

Se estiver usando um monitor externo (um dispositivo de captura/reprodução como DeckLink ou UltraStudio), ajuste as configurações de monitoramento de vídeo de acordo com as especificações do seu monitor.

1. Em “Monitoramento de Vídeo”, defina “Resolução de Vídeo” para corresponder à resolução do monitor. Ex.: 1920 x 1080 HD ou 3840 x 2160 Ultra HD.

   Embora a taxa de quadros nativa do vídeo imersivo seja 90 fps, ela não é suportada pela maioria dos monitores profissionais.

2. Escolha o formato adequado para a taxa de quadros do seu monitor. Ex.: 30p.



Tamanho de Imagem

As configurações da seção Tamanho de Imagem garantem que a resolução e a proporção da mídia imersiva sejam corretamente exibidas no visualizador. Esses parâmetros já terão sido configurados automaticamente após habilitar a opção fluxo de trabalho Apple Immersive em Definições Máster, mas ainda assim é importante verificar se estão corretos.

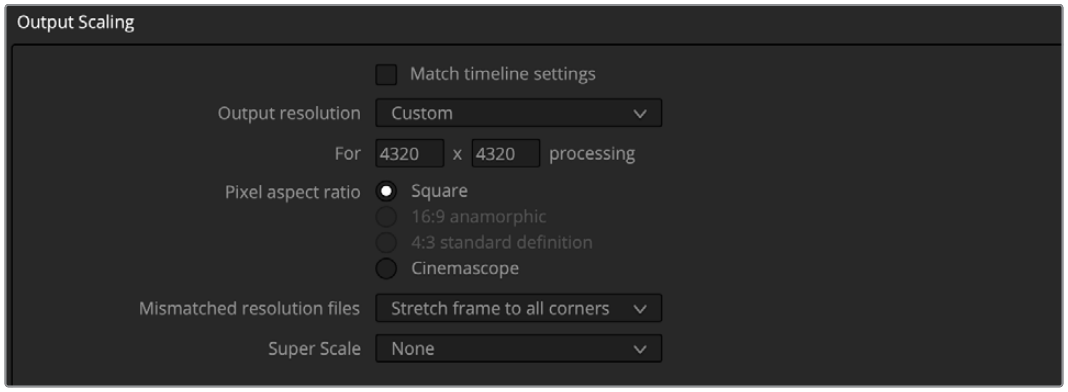
1. Selecione “Tamanho de Imagem” na barra lateral de Definições de Projeto.
2. Na seção “Tamanho de Saída”, verifique se a opção “Corresponder ajustes da linha de tempo” está desmarcada.

3 Verifique se a seção “Resolução de Saída” está definida como “Personalizada” e configurada para **4320 x 4320**.

Reduza a resolução pela metade (**2160 x 2160**) para telas menores e/ou reprodução mais rápida.

Essas resoluções em proporção quadrada são necessárias para a reprodução de vídeo no formato Apple Immersive no DaVinci Resolve Studio e para transmissão ao Apple Vision Pro (via Wi-Fi ou Developer Strap).

4 Defina “Arquivos com resoluções incompatíveis” como “Esticar todas as extremidades do quadro”.



Gerenciamento de Cor

Devido à necessidade de controle personalizado da luminância máxima em nits da saída, recomenda-se o uso dos nós Color Space Transform (CST) na página Cor para o gerenciamento de cor. O objetivo das configurações de gerenciamento de cor em Definições de Projeto é especificar o espaço de cor da linha de tempo (ou “working”) na página Cor e definir os metadados que serão associados ao arquivo renderizado.

1 Selecione “Gerenciamento de Cor” na barra lateral de Definições de Projeto.

2 Deixe “Ciência de cores” como “DaVinci YRGB”.

3 Defina “Espaço de cor da linha de tempo” como “DaVinciWG/Intermediate”.

Esse é um espaço de cor Wide Gamut à prova do futuro que oferece suporte a fluxos de trabalho HDR, tornando-o ideal para mídia da Blackmagic URSA Cine Immersive e entregas para o Apple Vision Pro.

4 Defina “Espaço de cor de saída” como “P3-D65 ST2084 1000 nit”.

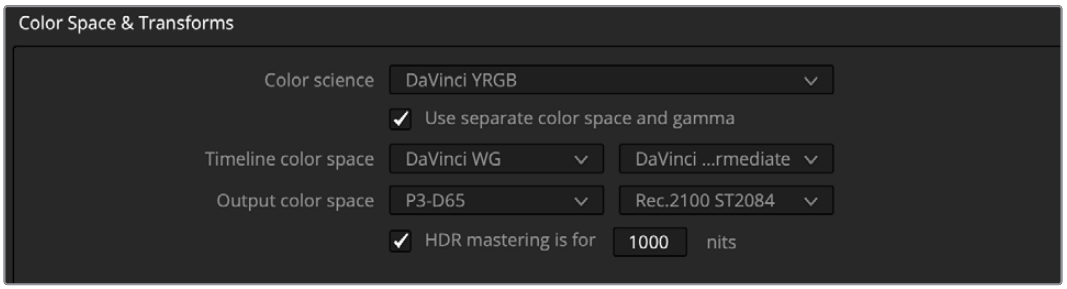
Esse espaço de cor apresenta o gamut apropriado para monitoramento nos visualizadores do DaVinci Resolve em um Mac Studio Desktop, que geralmente utiliza Display P3. Ao utilizar um monitor de gradação ou um perfil ICC diferente da Apple, será necessário alterar o espaço de cor de saída para corresponder ao padrão de cor da tela.

No entanto, será necessário alterar a gama (função de transferência) para mapear corretamente para o perfil ICC.

5 Clique em “Usar espaço de cor e gama separados” acima das configurações de espaço de cor para dividir os campos suspensos de espaço de cor e gama.

6 Defina a gama de saída (na coluna da direita) como “Rec.2100 ST2084”.

7 Selecione “A masterização HDR é para 1000 nits”.



Para economizar tempo na configuração do próximo projeto Apple Immersive, as configurações atuais podem ser salvas como uma predefinição.

8 Acesse o menu de opções de três pontos no canto superior direito da janela Definições de Projeto e escolha “Salvar Definições Atuais como Predefinição”.

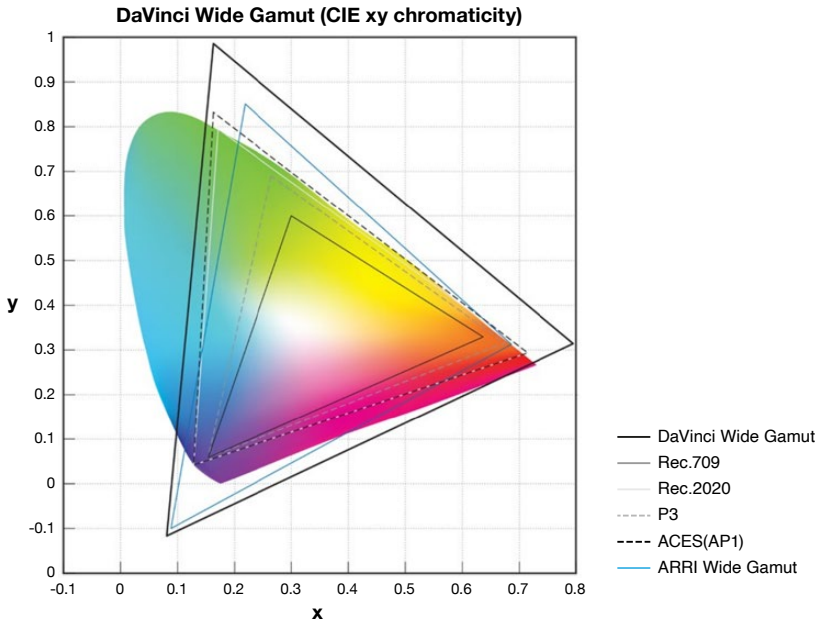
9 Nomeie a predefinição e clique em “Ok”.

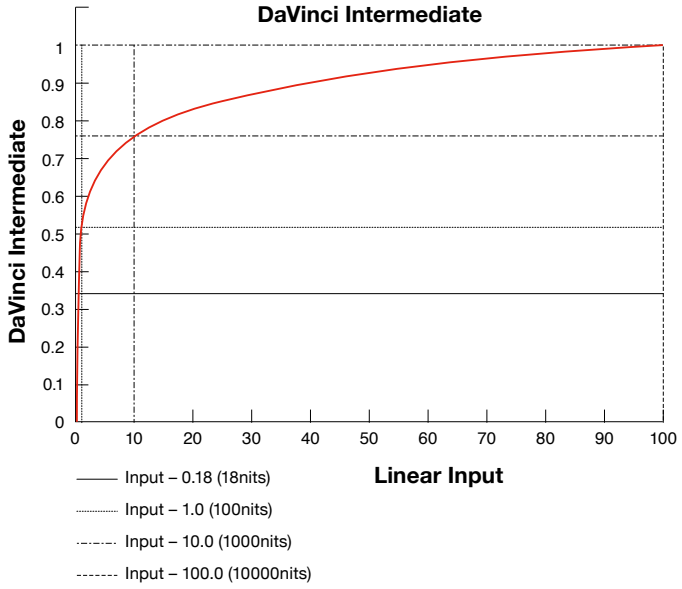
Para configurar rapidamente seu projeto ao trabalhar novamente com mídias imersivas, abra a janela Definições de Projeto, clique no menu de opções e escolha a predefinição listada.

10 Clique em “Salvar” para fechar o menu Definições de Projeto.

Sobre o DaVinci Wide Gamut

O DaVinci Wide Gamut é o espaço de cor interno avançado da Blackmagic Design desenvolvido para o DaVinci Resolve, que oferece um ambiente excepcionalmente amplo e preciso para gradação de cores e processamento de imagem. Combinado com a gama DaVinci Intermediate, forma um espaço de trabalho unificado que simplifica o gerenciamento de cores em diferentes formatos de câmera, garantindo mapeamento consistente de tons e saturação, da entrada à saída. Ideal para trabalhar com mídias HDR e imersivas, esse Wide Gamut permite que coloristas façam ajustes sutis sem risco de faixas ou cortes tonais ao masterizar para formatos de entrega de alta fidelidade.





Processamento em Apple Immersive

Na barra lateral de Definições de Projeto, a seção Apple Immersive permite configurar um ILPD (Immersive Lens Profile Data) personalizado para modificar como o conteúdo imersivo é calibrado e projetado.

1. Selecione “Apple Immersive” na barra lateral de Definições de Projeto
2. Clique em “Abrir Pasta de Calibração”.
3. Adicione ou atualize um arquivo ILPD personalizado.
4. Clique em “Atualizar Listas” para recarregar o conteúdo ILPD acessível no Pool de Mídia.
5. Clique em “Salvar” para fechar o menu Definições de Projeto.
6. No Pool de Mídia, clique com o botão direito em um clipe imersivo.
7. No menu contextual, acesse “Calibração Apple Immersive” e escolha entre o ILPD de origem (“Nenhum Arquivo de Calibração Apple Immersive Selecionado”), o ILPD personalizado ou o ILPD gap.

Por padrão, a pasta “Calibração ILPD Personalizada” está localizada na pasta ProIM, ao lado da pasta padrão de LUTs.

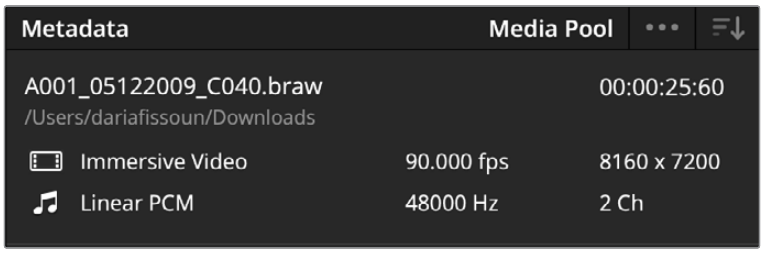
Ingestão

Ao ingerir conteúdo imersivo RAW, ambos os olhos e todos os metadados imersivos associados acompanham o clipe .braw importado.

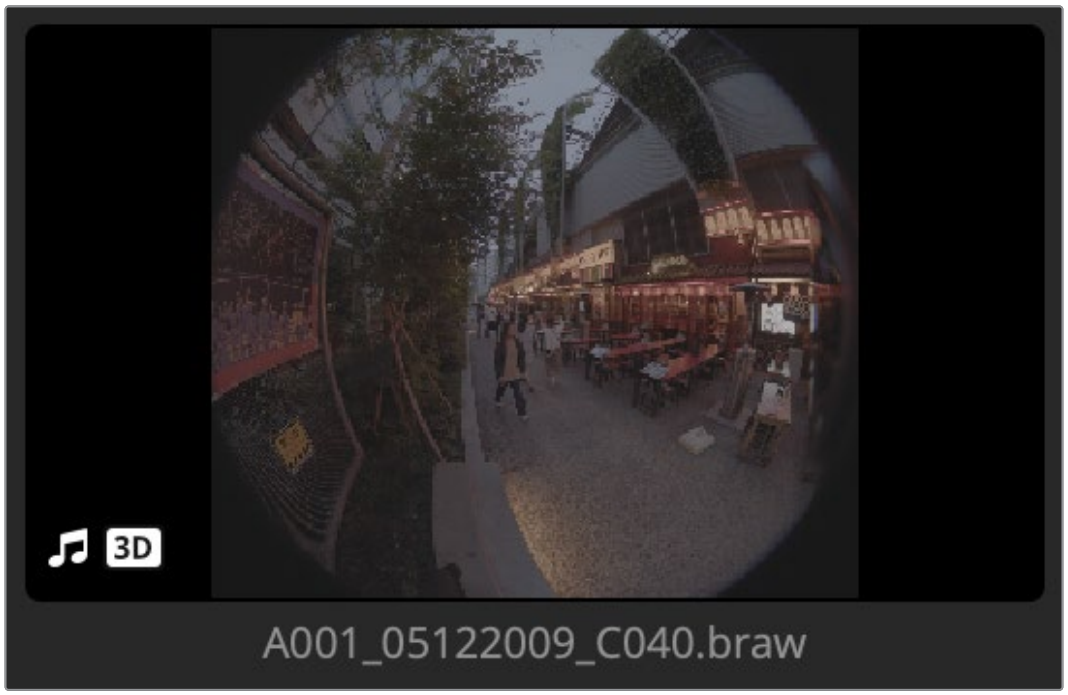
1. Acesse a página Mídias.
2. No painel Armazenamento de Mídia, navegue até a localização do conteúdo de vídeo imersivo.
3. Arraste e solte a mídia no Pool de Mídia abaixo.

DICA Para manter a estrutura original da pasta, arraste-a do painel Armazenamento de Mídia até a barra lateral do Pool de Mídia. Quando arrastadas para a repartição Mestre (ou para qualquer repartição que tenha sido criada), pastas e subpastas são mantidas.

Quando um clipe de vídeo Apple Immersive é selecionado, o parâmetro codec no cabeçalho do painel de Metadados indicará que se trata de um Vídeo Imersivo, junto com a taxa de quadros e a resolução nativas.



Além disso, um ícone 3D no canto inferior esquerdo da miniatura do vídeo confirma que se trata de um clipe 3D.



OBSERVAÇÃO Ao arrastar e soltar mídia imersiva no Pool de Mídia pela primeira vez, uma janela de diálogo pode solicitar escolher entre “Manter Configurações” ou “Alterar Configurações”. Ao escolher “Alterar Configurações”, as Definições de Projeto adotarão automaticamente a taxa de quadros e a resolução da mídia de origem.

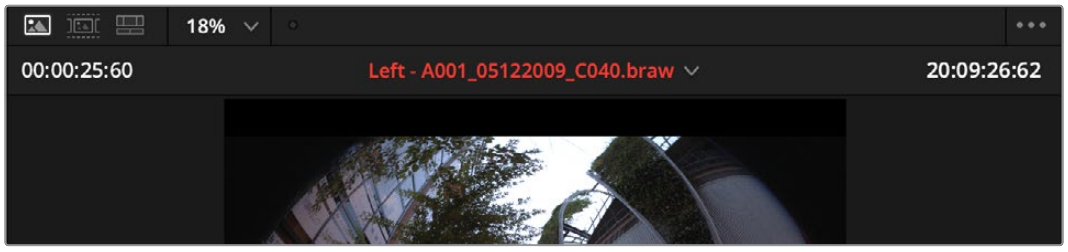
Edição

Ao trabalhar com conteúdo imersivo, as ferramentas e ações de edição permanecem praticamente as mesmas usadas em formatos de vídeo 2D convencionais. As instruções a seguir descrevem recursos e ações exclusivas dos fluxos de trabalho de vídeo imersivo.

Montagem

Ao pré-visualizar clipes durante a montagem da linha de tempo, é possível escolher qual olho (esquerdo ou direito) é exibido no visualizador de origem. Fora isso, o processo de montagem de um corte na linha de tempo permanece inalterado em relação aos fluxos de trabalho NLE padrão.

1. Acesse a página Edição.
2. Clique duas vezes em um clipe para carregá-lo no visualizador de origem, à esquerda. O rótulo na parte superior do visualizador indica qual olho está sendo usado.



3. Para alterar o olho no visualizador, clique com o botão direito no clipe de origem no Pool de Mídia e escolha Estereoscópico 3D > Modo Estereoscópico 3D > Olho Esquerdo/Olho Direito/Ambos os Olhos.

> **DICA** É possível também passar o mouse sobre as miniaturas no Pool de Mídia para percorrer os clipes no visualizador de origem, caso a opção Prévia de Mídias Ao Vivo esteja habilitada.

4. Use os atalhos de teclado I e O para definir um intervalo no visualizador de origem.
5. Pressione “Inserir Clipe” na barra de ferramentas (ou use o atalho de teclado F9) para inserir clipes na linha de tempo.

> **DICA** Você também pode arrastar e soltar clipes da Pool de Mídia no painel da linha de tempo.

6. Use o mouse (no Modo de Seleção ou Modo de Edição Aparar) para ajustar clipes na linha de tempo.

Fundos

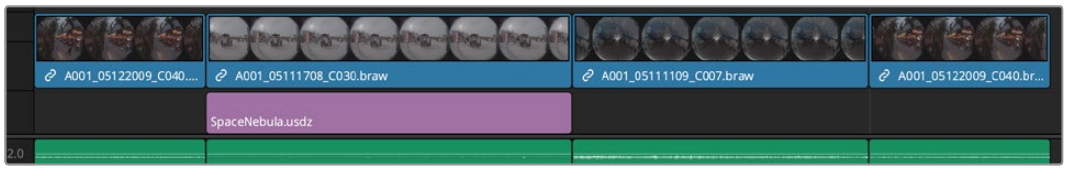
Ao selecionar “Ativar fluxo de trabalho Apple Immersive” nas Definições de Projeto, uma trilha de Fundos (“BD”) aparece automaticamente abaixo da trilha Vídeo 1.

Fundos referem-se à área que o observador verá ao olhar além do perímetro do conteúdo de vídeo imersivo, que normalmente é limitado a um campo de visão de 180 a 210 graus.

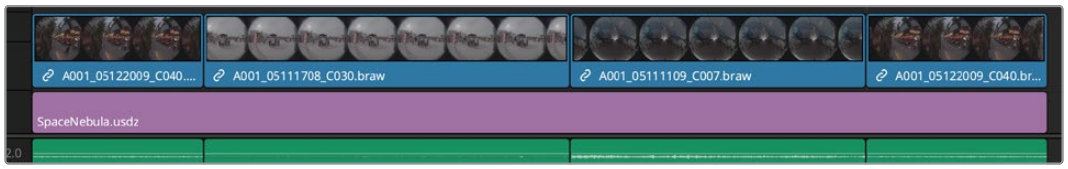
Por padrão, essa área é preta, mas é possível inserir imagens de fundo personalizadas (no formato .usdz) usando a trilha de fundos.

1 Arraste um arquivo de fundo .usdz para a trilha de Fundos para adicioná-lo à linha de tempo.

A duração das imagens de fundo deve estar alinhada aos limites dos clipes nas trilhas de vídeo. Um único arquivo de fundo pode abranger um clipe, vários clipes ou toda a linha de tempo, mas não pode cortar no meio de um clipe .braw imersivo (também chamado de “L-cut”).



Um clipe – correto ✓

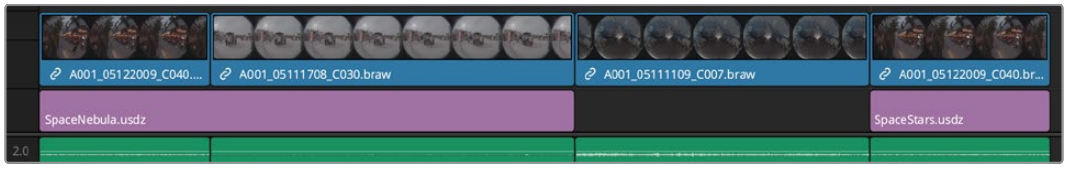


Vários clipes – correto ✓



L-cut – incorreto ✗

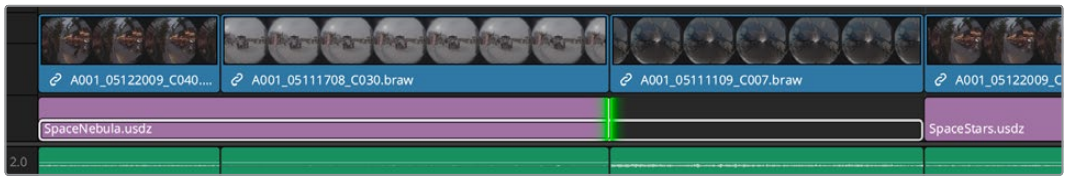
É possível optar por ter vários arquivos de fundo em uma única linha de tempo ou não adicionar nenhuma imagem, mantendo o fundo preto padrão.



Imagens diferentes e intervalos – correto ✓

2 Use os controles padrão de aparo para ajustar a duração do arquivo de fundo.

Arraste a borda do arquivo .usdz (no Modo de Seleção ou Modo de Edição Aparar) para ajustar a duração.



É possível também usar os comandos , (vírgula) e . (ponto) para avançar a borda de aparo um quadro por vez.

DICA Para garantir que os fundos estejam sempre alinhados às edições do vídeo imersivo, ative o modo Encaixar na barra de ferramentas da página Edição.

Usar o Visualizador

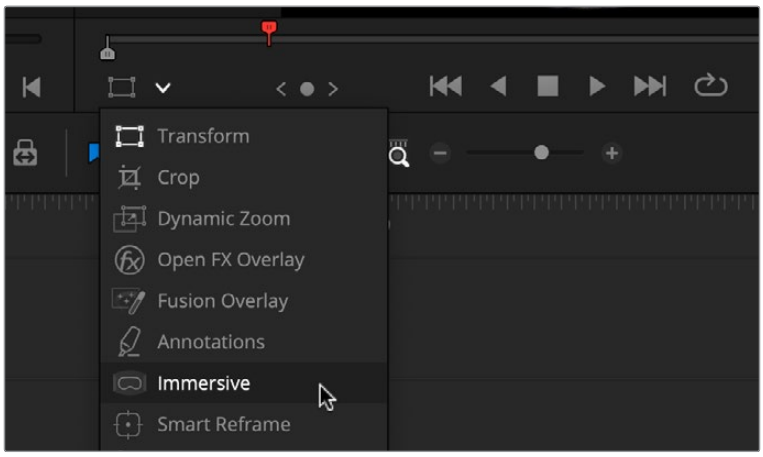
Por padrão, clipes imersivos são exibidos nos visualizadores da página do DaVinci Resolve no formato circular de espaço de lente. A calibração da lente usada para projetar a imagem tem origem na câmera e é exclusiva de cada clipe.



Há também a opção de alterar o modo de visualização para “LatLong” ou “Área de Visualização”.

1 No canto inferior esquerdo do visualizador da linha de tempo, clique na seta suspensa ao lado do botões de controle na tela.

2 Selecione “Imersivo”.



Uma nova barra de ferramentas aparecerá acima do visualizador com opções adicionais de visualização.

3 Clique em “LatLong” para ver uma projeção equirretangular da imagem. Esse modo distorce a imagem para se assemelhar mais às proporções de mídia 2D tradicionais, o que facilita certos processos de pós-produção, como rastreamento e composição.



4 Clique em “Área de Visualização” para ver um recorte central da cena imersiva em um quadro quadrado. Esse modo ajuda a visualizar claramente os detalhes da cena com proporções corretas em um monitor padrão, embora seja preciso cuidado para não perder ações fora do quadro.

É possível usar os controles de transformação na barra de ferramentas imersivas superior para inclinar e girar a imagem estereoscópica no quadro, bem como ajustar os campos de visão horizontal e vertical.



Transições

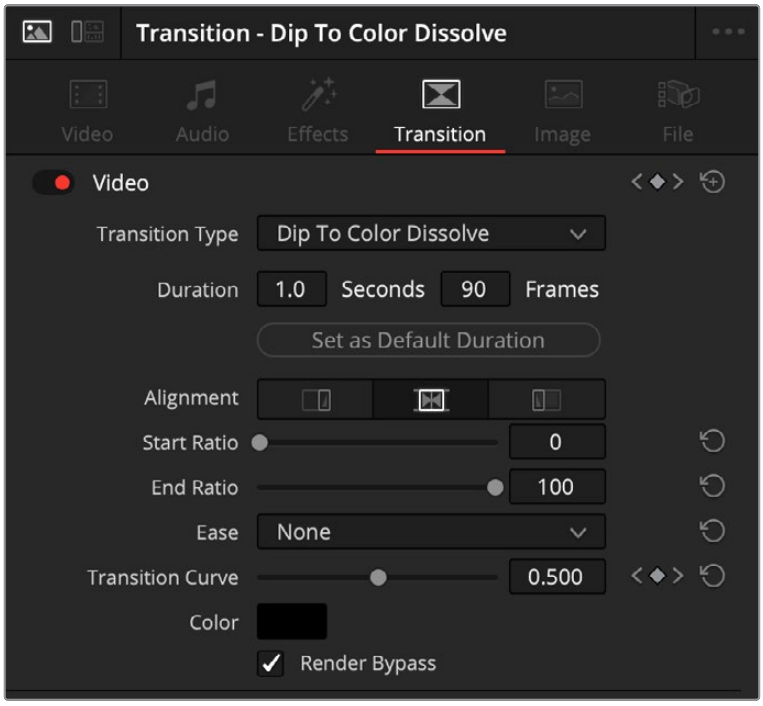
Para realizar a transição entre dois clipes de vídeo imersivo, é possível fazer um corte direto, usar as alças em cada lado do clipe para aplicar Fade In ou Fade Out, ou usar o efeito Dissolução de Imersão Colorida. As transições não devem ser incorporadas ao material, mas sim renderizadas pelo próprio Apple Vision Pro durante a reprodução.

1. Abra o painel Efeitos.
2. Na barra lateral, navegue até Caixa de Ferramentas > Transições de Vídeo.
3. Arraste Dissolução de Imersão Colorida para um ponto de corte entre dois clipes.



OBSERVAÇÃO É necessário ter dados de aparo suficientes além dos pontos de corte de cada clipe para aplicar uma transição. Se não houver dados de aparo suficientes (“alças”), a transição não será aplicada.

4. No painel “Inspetor”, acesse a guia “Transição” e selecione “Ignorar Renderização”, de modo a garantir que a transição seja renderizada corretamente pelo Apple Vision Pro.
5. Você pode então usar os parâmetros restantes no Inspetor de Dissolução de Imersão Colorida para modificar a duração, o alinhamento, a velocidade da animação e a cor da transição.



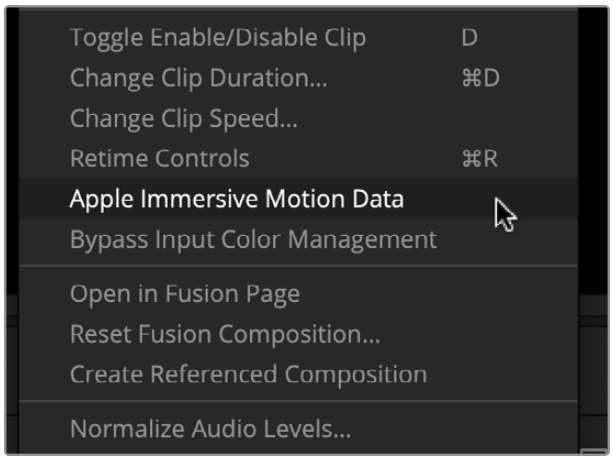
Apple Immersive Motion Data

Ao trabalhar com conteúdos imersivos, movimentos de câmera rápidos ou bruscos podem desorientar o espectador. Para garantir o conforto do público, evite tomadas com panorâmicas/inclinações rápidas, movimentos de rotação ou horizontes irregulares.

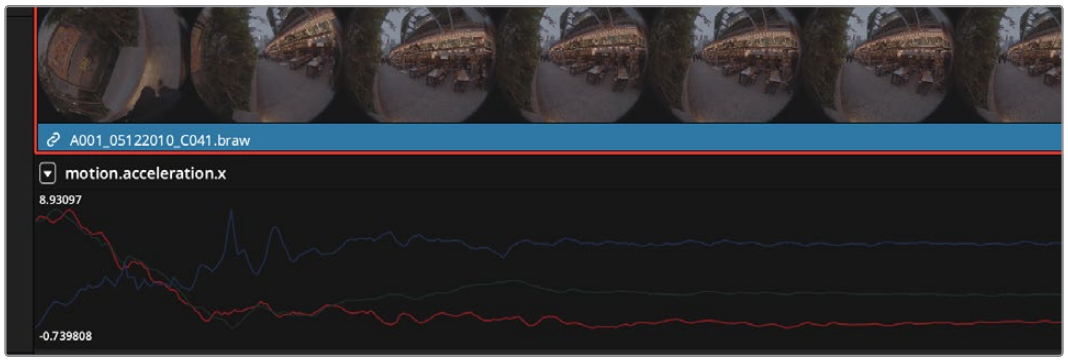
O recurso Apple Immersive Motion Data gera gráficos com os valores de aceleração x, y e z e da posição do giroscópio para demonstrar objetivamente em que ponto o movimento da câmera pode estar excessivo. Os editores podem usar essas informações para evitar trechos de filmagem que provoquem desorientação.

1 Clique com o botão direito em um clipe imersivo na linha de tempo.

2 Escolha “Apple Immersive Motion Data”.

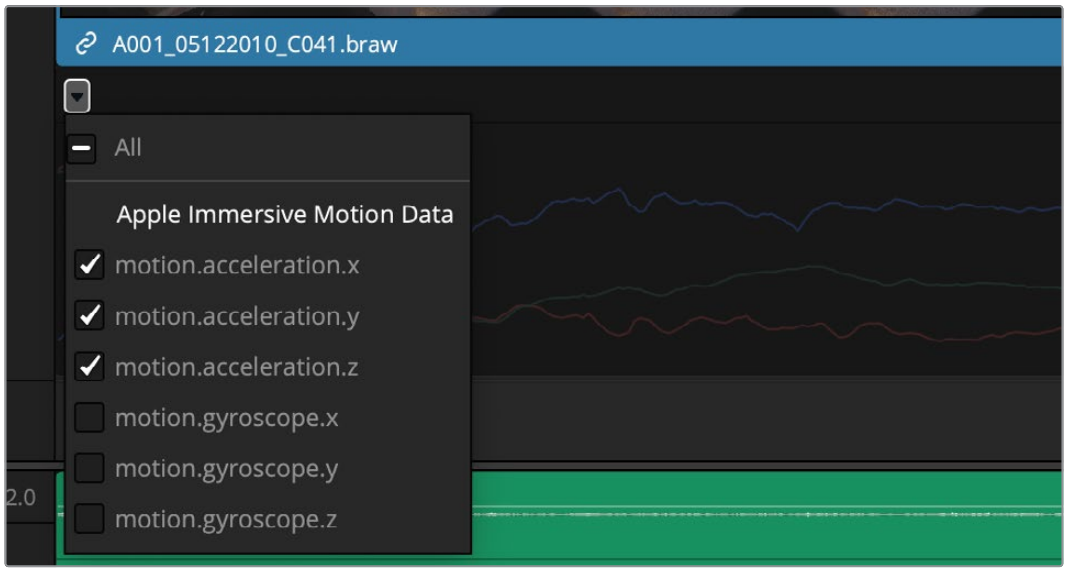


Um gráfico aparece na trilha de vídeo mostrando as taxas de aceleração x, y e z.



Neste exemplo, a câmera estava gravando enquanto o operador a configurava. Os picos no gráfico indicam movimentos bruscos quando a câmera foi erguida, girada e posicionada no ombro. O ideal é usar a parte final do vídeo, com leituras mais estáveis, após a câmera ter sido estabilizada.

3 No canto superior esquerdo do gráfico, clique na seta de deslocamento.



É possível ativar ou desativar as leituras com base no conteúdo do plano e na natureza do movimento da câmera.

Títulos

Na página Edição, é possível adicionar texto a uma cena imersiva usando qualquer ferramenta de título do painel Efeitos – Texto, Texto+, MultiText, Rolar e quaisquer variações de terço inferior.

Quando posicionado acima de um clipe imersivo, o gerador de títulos usará o ILPD (Immersive Lens Profile Data) do clipe para distorcer o texto de acordo com a distorção da lente da cena.

Se nenhum ILPD estiver disponível (por exemplo, ao adicionar texto sobre um fundo preto ou uma cena CGI imersiva), será aplicada a distorção de lente padrão para garantir que o texto continue aparecendo corretamente distorcido ao ser exibido no Apple Vision Pro.

1 No painel Efeitos, acesse Caixa de Ferramentas > Títulos.

2 Arraste o gerador de texto para a linha de tempo. Posicione-o em uma trilha de vídeo acima de um clipe imersivo para incorporá-lo à cena.



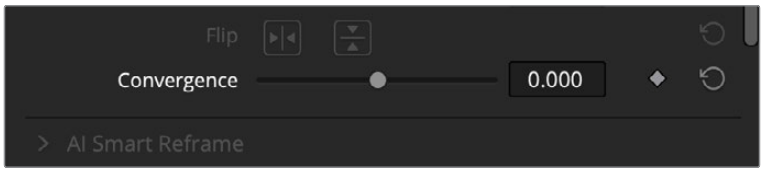
Assim como os fundos, os geradores de texto não podem ultrapassar os limites dos clipes. Como cada clipe é renderizado em um espaço de lente separado, qualquer título sobreposto corre o risco de ser distorcido entre os planos.

3 Apare o título até o(s) ponto(s) de edição do vídeo.

4 Use o Inspetor para escrever e formatar o texto do título.

5 Acesse a guia de ajustes e use o controle deslizante Convergência para posicionar o texto no espaço 3D.

Mover o controle deslizante para a esquerda empurra o título para trás no espaço, enquanto movê-lo para a direita o aproxima.



A convergência é melhor observada ao monitorar no Apple Vision Pro. Ao utilizar outros meios de monitoramento, é possível alterar como a convergência do texto aparecerá no visualizador mudando o modo Vision da camada de vídeo subjacente na página Cor.

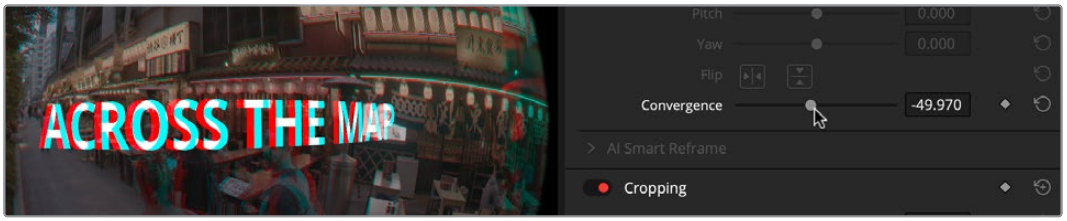
6 Acesse a página Cor.

7 Abra a paleta 3D.

A coluna Vision no centro da paleta permite alterar o padrão estereoscópico usado para projetar a mídia imersiva.

8 Selecione a guia Estéreo.

9 Altere o modo de saída para “Anáglifo (Colorido)”.



Ajustar o parâmetro “Convergência” do gerador de texto no painel Inspetor da página Edição será mais intuitivo quando visualizado com óculos anáglifos (ou Anachrome).

Considerações Criativas

Nestes primeiros dias de produção com a câmera Blackmagic URSA Cine Immersive e o Apple Vision Pro, os cineastas são incentivados a experimentar o formato para identificar as melhores práticas, estabelecer novas regras e descobrir fluxos de trabalho até então inimagináveis no cinema tradicional.

Com isso em mente, alguns aspectos da produção imersiva já foram destacados para atenção especial, a fim de garantir maior imersão do público e conforto visual.

1 A edição deve começar na pré-produção. Devido às considerações específicas sobre a duração dos planos e o conforto do público em relação aos movimentos, os cineastas devem captar as cenas já com a edição final em mente.

2 Evite ou limite o uso de clipes com movimentos rápidos de câmera, pois eles podem desorientar o espectador. Em vez disso, prefira planos fixos, panorâmicas lentas ou movimentos suaves com filmagens na dolly.

3 Ângulos de câmera extremos ou desalinhados também podem desorientar. Prefira tomadas niveladas ao sujeito e a uma distância confortável (não muito próxima).

4 Novos ambientes levam mais tempo para serem assimilados em mídias imersivas do que em uma cena 2D. Os cortes precisam ser mais lentos para dar tempo ao público de se adaptar aos novos cenários.

5 Ao cortar entre clipes, considere onde o espectador estava olhando nas tomadas anteriores. Uma ação importante não deve ocorrer no lado esquerdo do quadro se a atenção do público estava voltada anteriormente para a direita.

6 O Áudio Espacial conduz grande parte da narrativa e deve ser tratado como mais um personagem nas produções imersivas.

Composições VFX

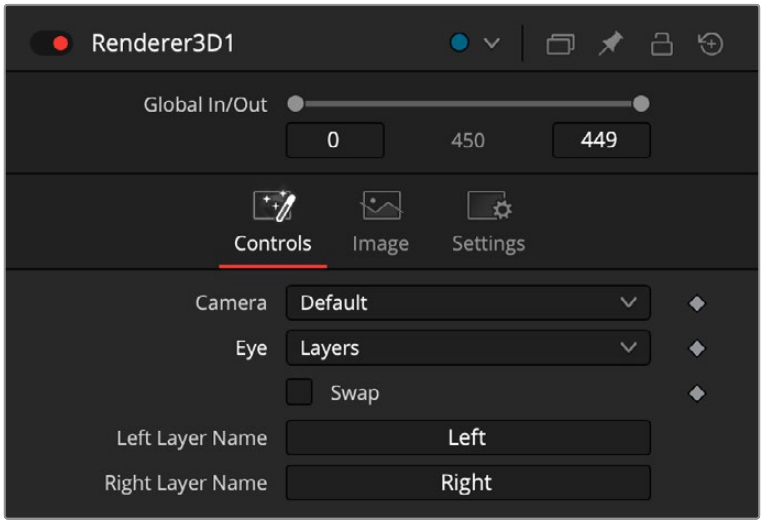
Quando a opção “Apple Immersive workflow” está ativada nas Definições do Projeto, parâmetros específicos do Fusion permitem fluxos estereoscópicos contínuos e composições compatíveis com mídias imersivas. Os nós MediaOut incluem opções de saída para os olhos esquerdo e direito, e a ferramenta Immersive Patcher permite projeções e composições equirretangulares.

Nós MediaOut

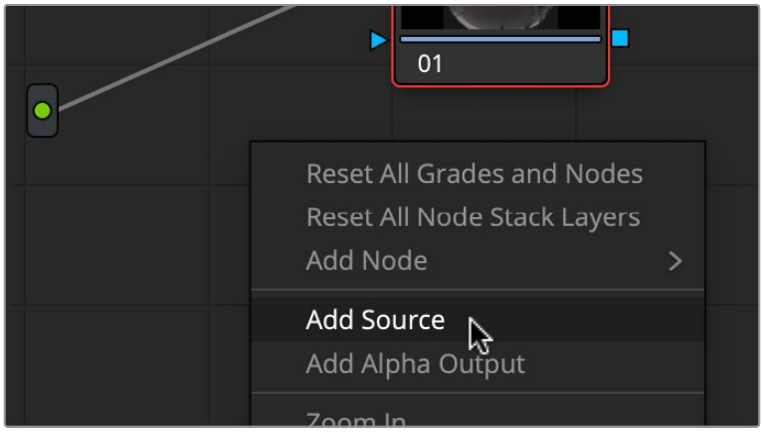
Ao acessar a página Fusion pela primeira vez em um clipe imersivo, o gráfico de nós apresenta dois nós MediaIn conectados a dois nós MediaOut correspondentes.



No Inspetor do MediaOut, o parâmetro Olho Estéreo pode ser usado para rotear a saída do sinal do olho.



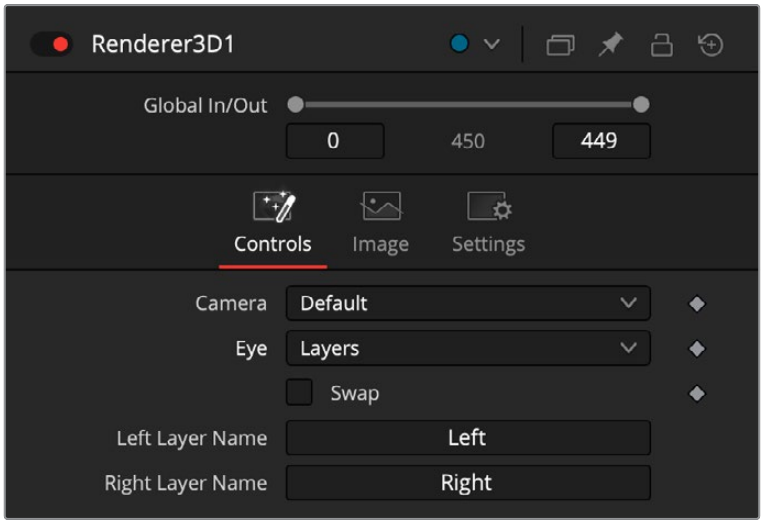
Por padrão, o MediaOut1 é definido para o olho esquerdo e o MediaOut2 para o direito. Essa configuração permite iniciar imediatamente a criação de composições estereoscópicas e sobreposições gráficas. Os dois nós MediaOut também geram duas variantes de sinal enviadas ao editor de nós da página Cor para aperfeiçoamento adicional da imagem. O sinal do MediaOut2 pode ser acessado na página Cor clicando com o botão direito no editor de nós e escolhendo “Adicionar Fonte”.



Quando o parâmetro Olho Estéreo está definido como Auto, o nó MediaOut combina os sinais e os atribui aos olhos corretos na linha de tempo, permitindo usar o fluxo de trabalho multicamadas do Fusion para enviar sinais esquerdo e direito a um único nó MediaOut.

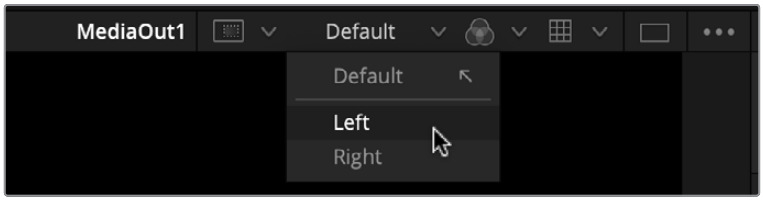
Renderizador 3D

Ao trabalhar com gráficos 3D ou CGI em uma linha de tempo imersiva, o nó Renderizador 3D inclui configurações para múltiplos olhos e a capacidade de processar as camadas esquerda e direita,



possibilitando criar títulos 3D complexos ou cenas completas em CGI e enviá-los diretamente para uma linha de tempo estereoscópica em projetos imersivos.

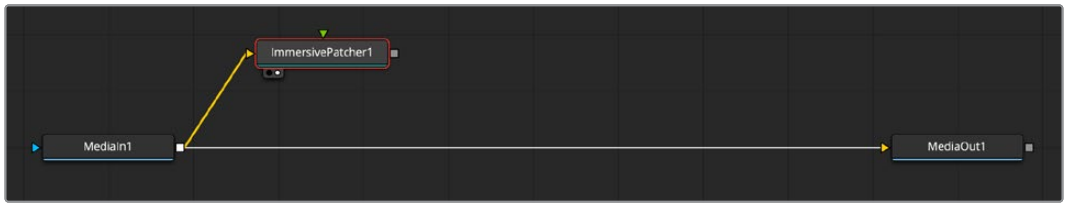
Nesse tipo de configuração, é possível usar o menu suspenso no canto superior direito dos visualizadores do Fusion para alternar entre a visualização do olho esquerdo e do direito da composição.



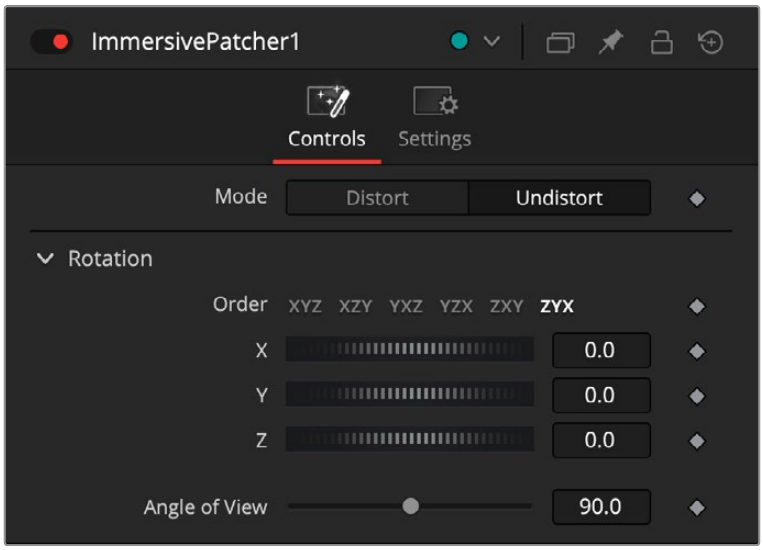
Composições com o Immersive Patcher

A ferramenta Immersive Patcher permite remover temporariamente a distorção de uma seção de um clipe imersivo para facilitar o rastreamento, a pintura e a composição em um plano 2D. Após finalizar, é possível usar um segundo Immersive Patcher para redistorcer o sinal imersivo e integrar os novos elementos compostos à cena.

1 Adicione o Immersive Patcher à pipeline de nós.



2 No Inspetor, clique em “Sem Distorção”.



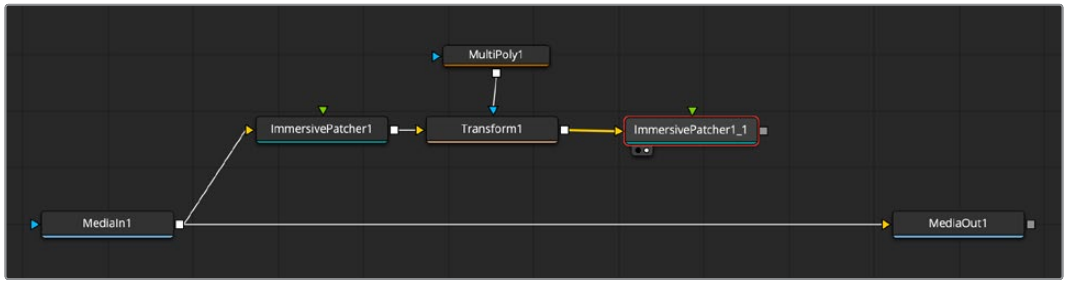
3 Use os parâmetros de rotação X, Y e Z para escolher uma seção da imagem imersiva a ser achatada.

4 Prossiga com a composição como em um sinal de vídeo 2D convencional. Isso inclui (mas não se limita a) usar métodos de pintura para remover objetos indesejados, amostrar trechos do quadro com transformações e polígonos para extensão de cenário ou aplicar máscaras, textos e camadas externas de imagem/vídeo para substituir placas ou telas.

   Ao concluir, será necessário redistorcer a imagem plana para corresponder à distorção imersiva original.

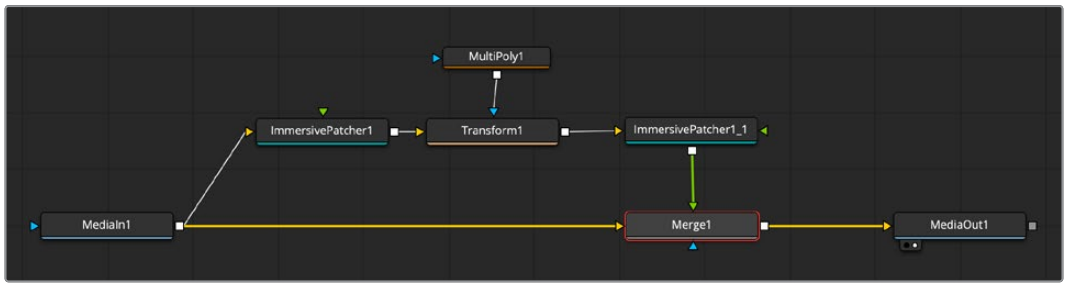
5 Copie o Immersive Patcher para preservar a rotação e as informações originais de ângulo de visão.

6 Cole o Immersive Patcher após os nós de composição e conecte ao final da pipeline.



7 No Inspetor, clique em “Distorcer”.

8 Una o patch composto ao clipe original para aplicar o efeito à cena imersiva.



Em trabalhos colaborativos, é possível sair da página Fusion e continuar a edição, mixagem e gradação de cores na linha de tempo imersiva no DaVinci Resolve.

Ao realizar turnovers imersivos, renderize cada olho em EXRs no ACES AP0 para preservar ao máximo as propriedades da entrega imersiva.

Tratamento de Cores

Esta seção aborda as ferramentas ópticas, de cor e 3D que podem ser usadas ao trabalhar com mídias imersivas na página Cor do DaVinci Resolve.

Devido à natureza única dos conteúdos imersivos, nem todas as ferramentas estarão disponíveis para uso na pós-produção. Outras ferramentas ainda não estão disponíveis, mas estão em processo de adaptação para suportar conteúdos imersivos.

Redução de Ruído

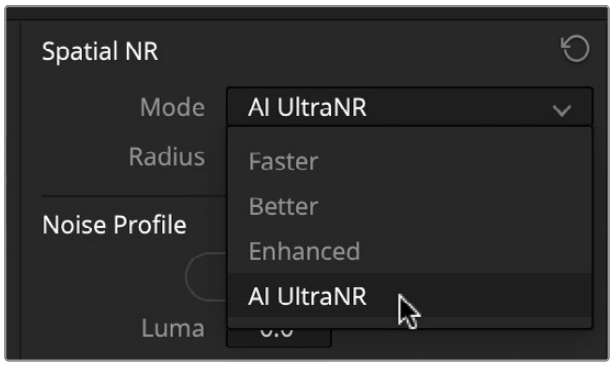
É possível aplicar redução de ruído ao trabalhar com vídeos imersivos. Na paleta “Efeitos de Movimento”, o recurso “RR Espacial” analisa o padrão de ruído da mídia de origem para reduzi-lo ao máximo, preservando os detalhes originais da imagem.

1 Na barra de ferramentas da página Cor, abra a paleta “Efeitos de Movimento”.

Na coluna “RR Espacial”, selecione o modo de redução de ruído desejado.

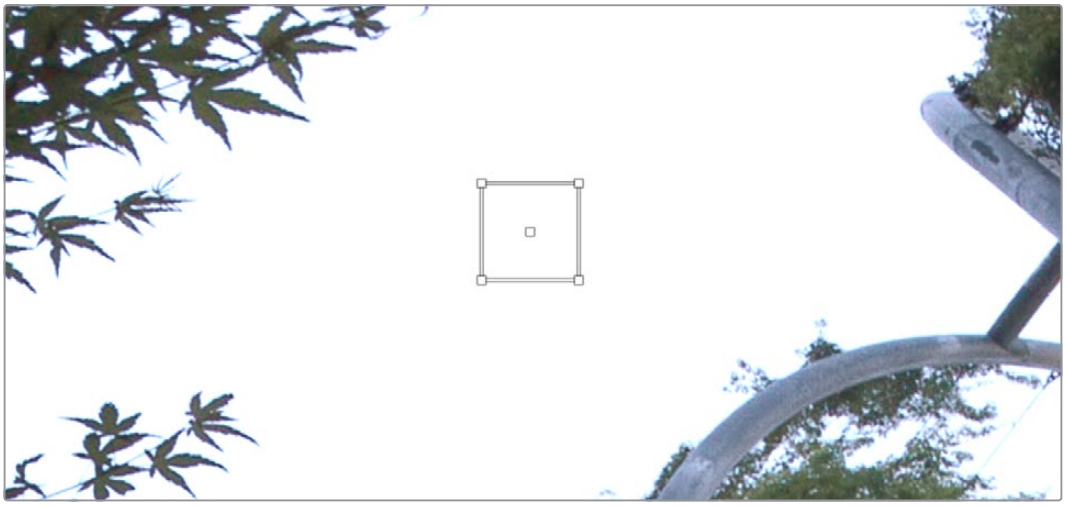
- *Os modos* Acelerada, *Superior* e *Avançada* analisam os padrões dentro de um raio definido e configuram a saída com base no limite estabelecido. Eles estão listados em ordem de complexidade de processamento.
- *O AI UltraNR* é um identificador de padrões de ruído baseado em IA que extrai o padrão do sinal localizando e analisando a seção “mais uniforme” do quadro para isolar o ruído puro.

2 Defina o modo de redução de ruído espacial para AI UltraNR.



3 Clique em “Analisar”.

Uma caixa delimitadora aparecerá no visualizador, identificando a área “mais vazia” de dados de pixels no quadro. Sem conteúdo visual tangível, os dados dentro da caixa são ideais para extrair o ruído digital produzido pelo sensor da câmera.



É possível reposicionar e redimensionar a caixa delimitadora no visualizador, se desejado, mas isso geralmente não é recomendado, pois o AI UltraNR é mais eficiente em identificar as regiões ideais de análise do que nossos olhos.

Também é possível ajustar manualmente os limites de Luma e Croma no Perfil de Ruído, mas, em geral, o AI UltraNR atinge automaticamente a combinação ideal de parâmetros para uma redução de ruído eficaz.

Gradação Primária

A gradação primária de cores funciona da mesma forma que em imagens de vídeo convencionais. Use as mesmas considerações para normalização, balanceamento e correspondência ao utilizar os discos Primárias e Log, a paleta HDR e as Curvas HSL. As paletas ColorSlice e Distorcedor de Cores também funcionam conforme esperado.

Secundárias

Os métodos de seleção baseados em pixel, como o Qualificador, funcionam normalmente com mídias imersivas.

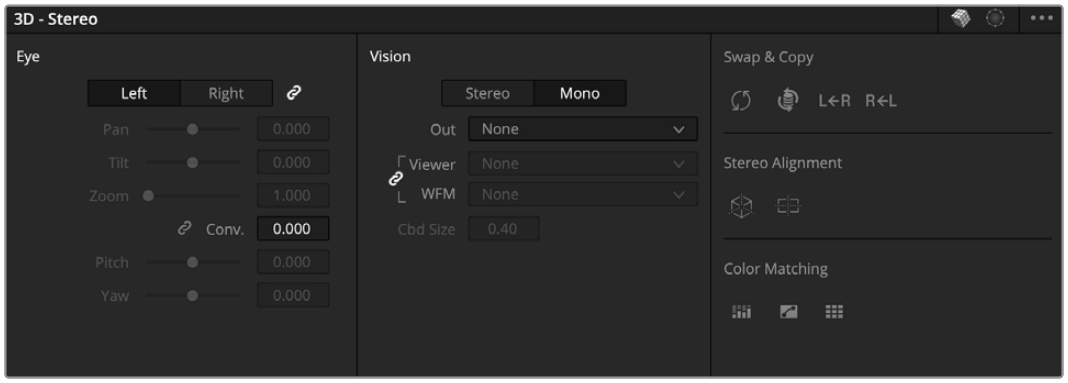
A seleção baseada em objeto ou janela só funciona com precisão em um ambiente equirretangular (ou de área de visualização), o que significa que a maioria dos métodos secundários que exige precisão na forma de qualificação está indisponível no momento. Isso inclui as Power Windows (e, por extensão, a paleta Rastreador) e a ferramenta Magic Mask.

Tamanho

Não há controles de tamanho (como zoom e panorâmica) disponíveis para conteúdos imersivos.

Paleta 3D

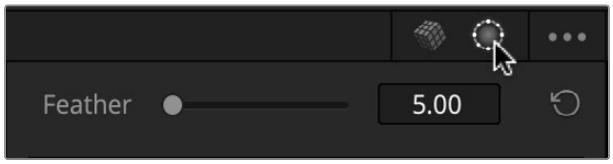
A paleta 3D da página Cor inclui controles para gerenciar diversos formatos de câmeras e gráficos 3D. Ao contrário de outros formatos estereoscópicos, não é necessário ajustar o alinhamento dos olhos, pois as lentes, a distância focal e as íris da câmera Blackmagic URSA Cine Immersive são fixos.



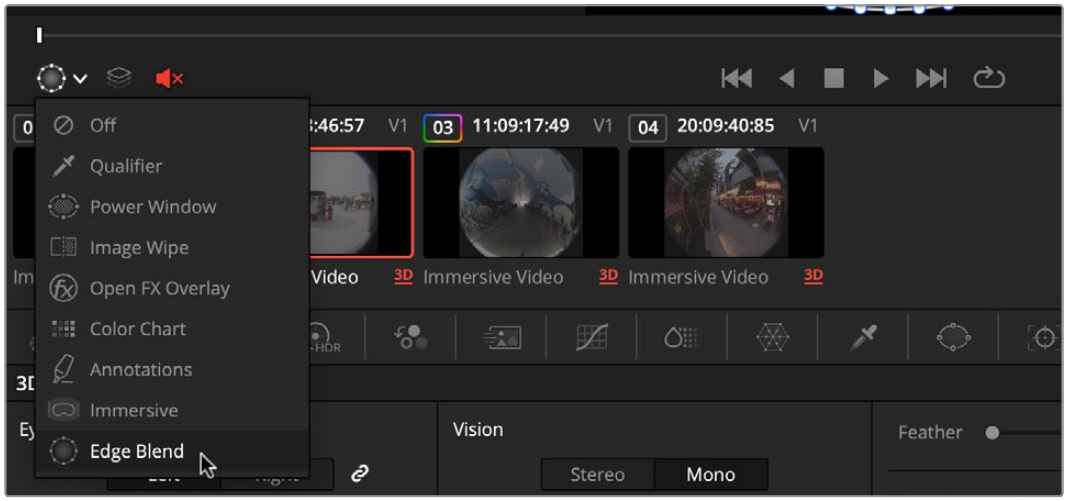
O modo Mistura de Borda é exclusivo para conteúdos imersivos e permite ajustar e suavizar o perímetro do quadro de vídeo imersivo. É possível usá-lo para ocultar equipamentos de filmagem visíveis e limitar o campo de visão do observador. Além dessa borda, o observador verá o fundo.

1 Abra a paleta 3D.

2 No canto superior direito, altere o modo para Mistura de Borda.



3 Verifique se os controles na tela do visualizador estão definidos como Mistura de Borda.



Deve aparecer um círculo de pontos de controle ao redor do perímetro do quadro do vídeo, com um ponto de ancoragem no centro.

4 Arraste um ponto de controle para alterar a posição.



Ou clique com o botão direito para excluir um ponto.

5 Na coluna direita da paleta 3D, use o parâmetro Difusão para suavizar a borda do quadro.

6 Selecione “visionOS Effect” para garantir que os novos dados de borda sejam corretamente processados durante a reprodução no Apple Vision Pro. Se não for selecionado, a borda será incorporada ao arquivo de vídeo renderizado.

7 Na coluna esquerda, use as guias Esquerdo e Direito para alternar entre os olhos no visualizador.

Observe que muitos dos controles da paleta 3D ficam esmaecidos. Parâmetros em todo o DaVinci Resolve aparecerão esmaecidos quando não forem aplicáveis à mídia em edição, que neste caso é o conteúdo imersivo.

8 Na coluna central, use as guias Estéreo e Mono para alternar entre os visualizadores de lente única e dupla do clipe de vídeo imersivo.

Gerenciamento de Cores CST

O uso do efeito Color Space Transform (CST) em nós no editor de nós da página Cor permite um mapeamento de cor preciso e um controle personalizado sobre a luminância máxima em nits da linha de tempo.

O fluxo de trabalho recomendado é composto por quatro nós CST:

1. INPUT CST – da câmera para o espaço de cor de trabalho.

   A correção de cores ocorre entre esses dois nós.

2. OUTPUT CST – do espaço de cor de trabalho para o monitor HDR do Mac.
3. MONITOR CST – de 1000 nits para 250 nits para monitoramento HDR.
4. AVP CST – de 250 nits para 108 nits para monitoramento e entrega no Apple Vision Pro.



Mapeamento de Linha de Tempo CST

Primeiro, será necessário mapear o espaço de cor de entrada da câmera para o espaço de cor de trabalho da linha de tempo. A câmera Blackmagic URSA Cine Immersive opera no color gamut Blackmagic Design Wide Gamut Gen 4/5 com a curva log DaVinci Intermediate.

1. Abra o painel Efeitos.
2. Arraste o Color Space Transform para um nó vazio (01) no editor de nós.

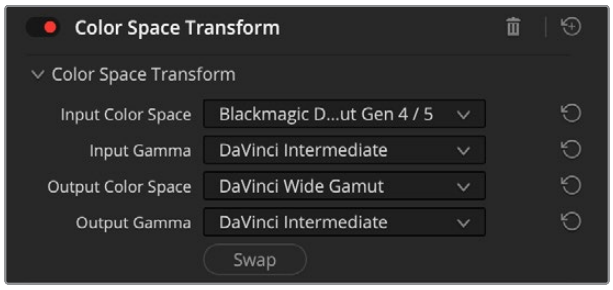
   Ao mapear do espaço de cor de entrada para o espaço de cor da linha de tempo, este nó deve estar entre os primeiros na árvore de nós, antes de qualquer operação de gradação de cores.

   Nos casos em que o espaço de cor de entrada estiver sendo mapeado diretamente para o espaço de cor de monitoramento ou entrega, este nó deve estar entre os últimos na árvore de nós, após as operações de gradação.

3. Rotule o nó 01 como **INPUT CST**.

> **OBSERVAÇÃO** Os rótulos de nó usados nestas instruções servem para facilitar a referência à documentação e capturas de tela. Você pode usar a convenção de nomenclatura que preferir ou ignorar a rotulagem por completo.

4. Defina “Espaço de Cor de Entrada” como “Blackmagic Design Wide Gamut Gen 4/5”.
5. Defina “Gama de Entrada” como “DaVinci Intermediate”.
6. Defina “Espaço de Saída” como “DaVinci Wide Gamut”.
7. Defina “Gama de Saída” como “DaVinci Intermediate”.



8 Crie um novo nó serial (02).

9 Copie e cole o primeiro nó (01) no novo nó (02).

10 Rotule o nó 02 como **OUTPUT CST**.

Será necessário definir o espaço de cor de saída e a gama como entrada neste novo nó serial para garantir a consistência do espaço de cor da linha de tempo da árvore de nós.

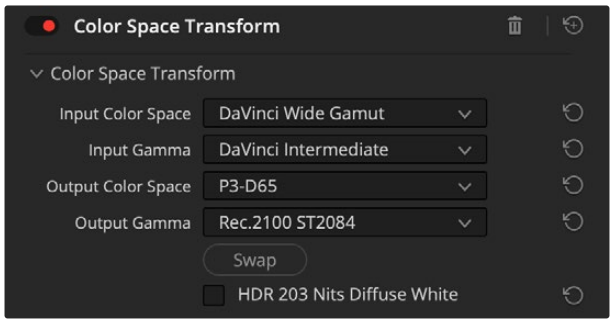
11 Pressione o botão “Trocar” para alternar os parâmetros de entrada e saída.

Agora, o espaço de cor e a gama de entrada são apropriados para este nó CST.

12 Defina “Espaço de Saída” como “P3-D65”. Este é o gamut padrão atual das telas do Mac.

13 Defina “Gama de Saída” como “Rec.2100 ST2084”, que é o padrão HDR para um sinal de 1000 nits.

Todas as operações de gradação ocorrerão entre esses dois nós. Mas, ao trabalhar com gerenciamento de cor baseado em nós, é possível também realizar operações fora dos limites do gerenciamento de cor da linha de tempo com recurso que muitos coloristas aproveitam para aplicar gradações específicas, efeitos ou elementos de terceiros.



Perfis ICC

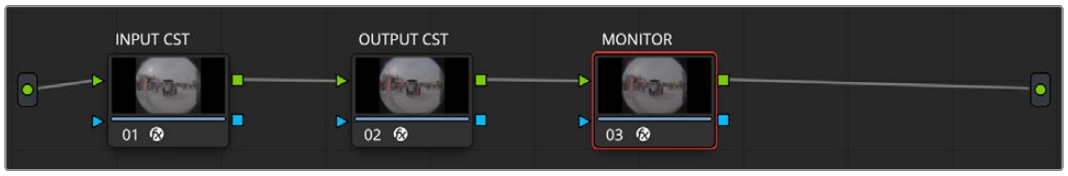
Para garantir ainda mais que o sinal da Blackmagic URSA Cine Immersive esteja mapeado corretamente para a tela do Mac, acesse Ajustes do Sistema > Telas e verifique se o perfil de cor está definido como Apple XDR Display (P3-1600 nits).

Mapeamento de Monitor de Gradação CST

É possível usar CSTs adicionais para limitar temporariamente os valores máximos de saída em nits durante a gradação.

1 Crie um novo nó serial (03) após o nó de gerenciamento de cor OUTPUT CST (02).

2 Rotule o nó 03 como **MONITOR**.



3 Adicione o efeito Color Space Transform ao nó MONITOR (03).

Os valores de entrada devem corresponder ao sinal de saída do CST anterior.

4 Defina “Espaço de Cor de Entrada” como “P3-D65”.

5 Defina “Gama de Entrada” como “Rec.2100 ST2084”.

A finalidade deste nó CST é apenas mapear a luminância da imagem para um valor de nits menor, o que significa que os parâmetros de entrada e saída podem permanecer inalterados.

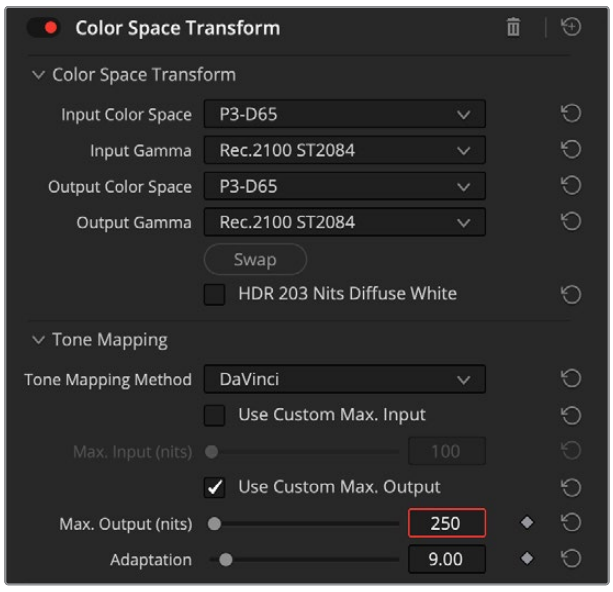
6 Defina “Espaço de Saída” como “P3-D65”.

7 Defina “Gama de Saída” como “Rec.2100 ST2084”.

OBSERVAÇÃO Em algumas circunstâncias, não é necessário definir o espaço de cor ou a gama de um CST se a entrada e a saída forem idênticas (sem mapeamento). Neste caso, é necessário porque a aplicação do mapeamento personalizado de nits é afetada pelo valor e pelo formato da curva log do sinal.

8 Selecione “Usar Saída Máx. Personalizada”.

9 Configure a asídaMáx. (nits) para **250**.



O sinal de 1000 nits será agora mapeado (usando o Método de Mapeamento do DaVinci) para 250 nits,

resultando em uma imagem muito próxima ao sinal Apple Vision Pro renderizado em um monitor HDR de gradação (como Flanders XMP310, Sony BVM-HX310, etc.).

No entanto, para monitorar seu projeto em um Apple Vision Pro, será preciso realizar mais uma operação de mapeamento de tom.

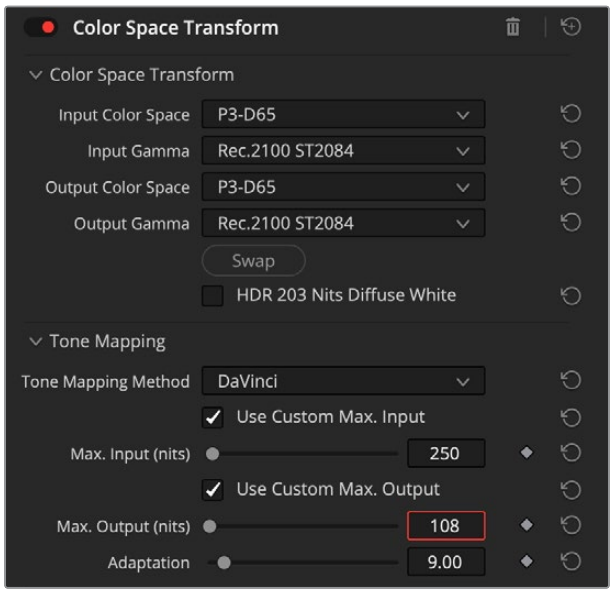
Monitoramento Apple Vision Pro CST

Um nó CST final pode ser usado para permitir monitoramento ao vivo, pré-visualização e entrega final em dispositivos Apple Vision Pro.

1. Crie um novo nó serial (04) após o nó MONITOR (03).
2. Copie e cole o nó MONITOR (03) sobre o nó 04. Serão usados os mesmos parâmetros de entrada e saída, alterando apenas o valor de nits.
3. Rotule o nó 04 como **AVP**.



4. Selecione “Usar Entrada Máx. Personalizada”.
5. Configure a Entrada Máx. (nits) para **250** para interpretar corretamente o sinal mapeado do nó MONITOR.
6. Configure a Saída Máx. (nits) para **108**, que é o valor máximo em nits para monitoramento e entrega ao Apple Vision Pro.



Agora é possível prosseguir com a gradação, desabilitando os nós CST quando (e se) necessário, conforme seu fluxo de trabalho. Por exemplo, durante a primeira passada de gradação, em que o monitor HDR externo será usado, será necessário desabilitar o nó AVP (04) para garantir que esteja vendo corretamente o sinal de 250 nits no monitor HDR. Ao alternar para a pré-visualização no Apple Vision Pro ou para a entrega final, reabilite o nó AVP (04).

Os CSTs podem ser colocados no editor de nós da linha de tempo?

Não. No momento, o mapeamento tonal de clipe depende da aplicação correta do parâmetro Color Space Transform Adaptation, que deve ocorrer clipe por clipe.

Áudio

O Apple Vision Pro e o DaVinci Resolve suportam áudio espacial e ambissônico, áudio AVP (Apple Vision Pro) e formatos AIA (Apple Immersive Audio).

Muitos aspectos da pós-produção de áudio permanecem inalterados ao se trabalhar com áudio espacial. Uma das principais diferenças é que, em vez de monitorar com alto-falantes de alta qualidade, recomenda-se o uso de fones de ouvido ou AirPods por parte dos engenheiros de som para proporcionar uma imersão espacial real e uma melhor reprodução do ambiente auditivo do público.

Enquanto nas mixagens 2D a maioria dos diálogos é mixada para os canais centrais, no áudio espacial o som precisa ser mixado na área de onde se origina. O ducking, a equalização e o balanceamento devem ser usados para mixar as trilhas de áudio de modo a não sobrecarregar ou confundir o ouvinte.

Elementos espaciais imersivos podem ser combinados com uma trilha estéreo normal, permitindo que certos sons permaneçam constantes independentemente do movimento da cabeça. Isso é útil para manter o som não diegético, como trilhas musicais e narrações, claro e de fácil interpretação.

Sons de ambiente tendem a ser reproduzidos de forma mais discreta no áudio espacial, exigindo mixagem mais intensa para que sejam percebidos. No entanto, isso não deve ser feito a ponto de desorientar o ouvinte.

A reverberação contribui muito para a imersão, pois é algo que os ouvintes experimentam naturalmente todos os dias — cada som produzido em um ambiente reflete nas superfícies ao redor, resultando em uma reverberação expansiva. Por isso, recomenda-se o uso de reverberação no design de áudio espacial para garantir uma imersão total.

O formato Apple Immersive Audio (AIA) funciona como um contêiner para diversos formatos:

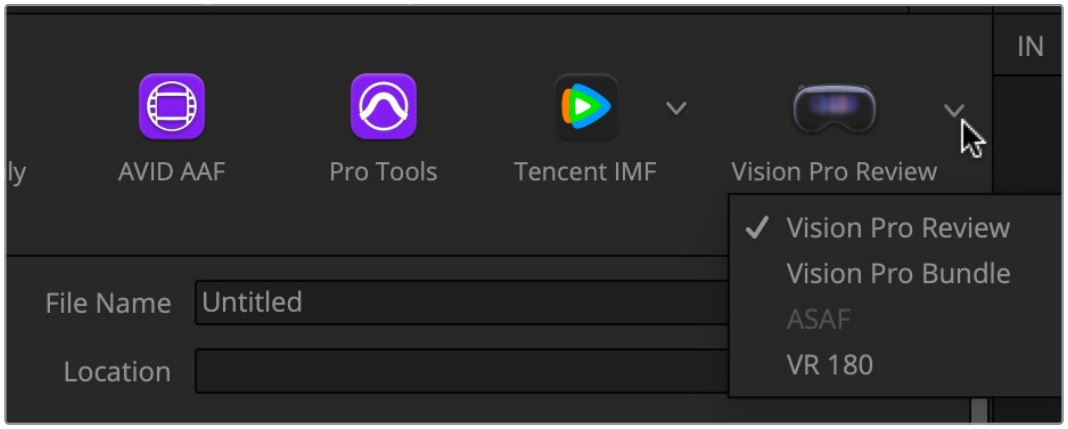
- **Apple Spatial Audio Format (ASAF)** é o formato bruto de trabalho de áudio.
- **Apple Positional Audio Codec (APAC)** é o formato .mp4 codificado, reproduzível em dispositivos compatíveis com APAC, como AirPods, ou a partir de um arquivo AIVU no Apple Vision Pro.

Apple Spatial Audio Format (ASAF)

O Apple Spatial Audio Format (ASAF) é um formato de áudio imersivo desenvolvido principalmente para aplicações de VR e AR no Apple Vision Pro. Exclusivo do macOS, o ASAF é compatível com o ProTools (via plugins) e com o DaVinci Resolve (suporte nativo). O ASAF oferece recursos técnicos avançados, incluindo suporte integrado a fontes Ambisonics (denominadas Scenes) e roteamento por meio de atribuições de barramento. A mixagem é virtual, sem barramentos físicos, e os usuários podem escolher entre vários formatos de monitoramento, incluindo binaural com rastreamento de cabeça por meio de integração direta com os Apple AirPods. A panorâmica é controlada pelas ferramentas já existentes no DaVinci Resolve, como o diálogo de panorâmica, automação, SpaceView e ferramentas de panorâmica no visualizador, com controles secundários no Inspetor. Esses recursos permitem o posicionamento espacial preciso das fontes, incluindo rastreamento automático em relação ao vídeo lat-long. As mixagens ASAF podem ser exportadas nos formatos WAV ou MP4 (com metadados correspondentes), por meio da opção Immersive Audio > Exportar Arquivo Máster. No entanto, atualmente não podem ser importadas nem reproduzidas no DaVinci Resolve.

Entrega

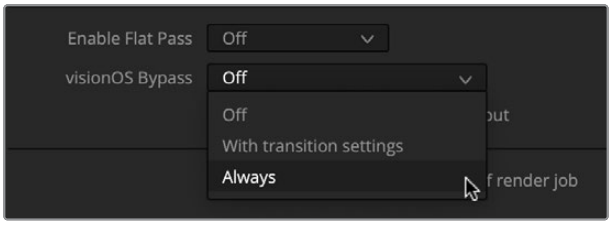
Na página Entrega, há duas opções de renderização que permitem gerar arquivos Apple Vision Pro para revisão e entrega final.



- **Vision Pro Review** – cria um pequeno pacote MV-HEVC para fins de pré-visualização. Compartilhe o arquivo resultante com o Apple Vision Pro via Wi-Fi e assista no aplicativo Apple Immersive Video Utility (AIVU), disponível na App Store.
- **Vision Pro Bundle** – cria um pacote grande de arquivos ProRes, que pode ser arquivado ou processado no Apple Compressor para preparação para reprodução no Apple TV.
- **ASAF** – renderiza uma mixagem de áudio ASAF em formatos WAV ou MP4.
- **VR180** – gera um arquivo de vídeo estereoscópico compatível com VR180.

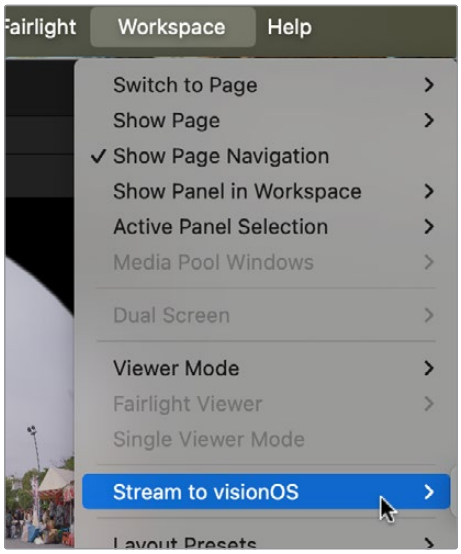
Ignorar Transições

Se uma transição for adicionada a uma linha de tempo imersiva, ela será pré-renderizada e transformada em uma única trilha de reprodução. O XML gerado para o conteúdo imersivo descreve o processo de alternância entre esses fluxos renderizados (imagem, depois transição pré-renderizada e novamente imagem). Nas Configurações Avançadas do painel Configurações de Renderização, é possível ignorar transições do visionOS (Sempre), aplicá-las de acordo com seu status na linha de tempo (Com configurações de transição) ou incluí-las como aparecem na linha de tempo (Desativar).



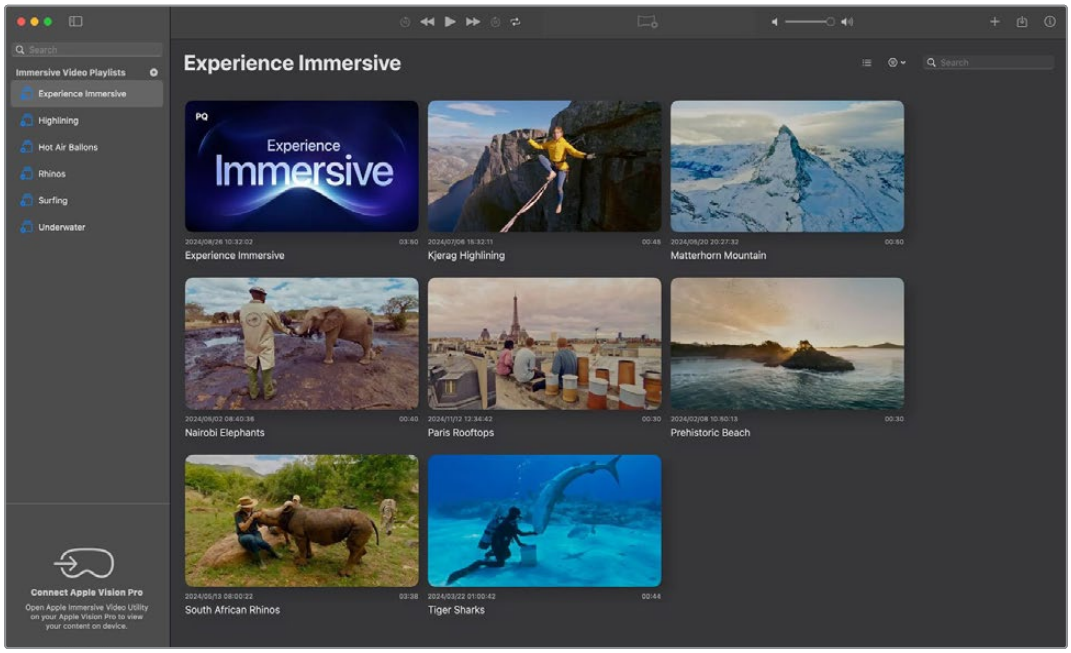
Transmitir para o Apple Vision Pro

É possível transmitir para o Apple Vision Pro a partir das páginas Edição, Cor e Entrega no DaVinci Resolve, acessando o menu Área de Trabalho, escolhendo “Stream to visionOS” e selecionando o dispositivo AVP. Uma rede sem fio rápida (Wi-Fi 5 no mínimo; Wi-Fi 6 preferencialmente) é recomendada para garantir uma transmissão fluida.



Sobre o Apple Immersive Video Utility

O Apple Immersive Video Utility é um aplicativo especializado para macOS e visionOS, projetado para simplificar a criação, o gerenciamento e a reprodução de conteúdo Apple Immersive Video. Voltado para cineastas profissionais e criadores de mídias imersivas, o utilitário permite importar, organizar, empacotar e revisar arquivos de vídeo estereoscópico 8K com campo de visão de 180 graus e Áudio Espacial. Há suporte para a criação de arquivos AIVU (Apple Immersive Video Universal), permitindo a inspeção e a modificação de metadados em fluxos de pós-produção. É possível criar playlists, classificar e pesquisar bibliotecas e até mesmo transmitir conteúdo diretamente para um ou mais dispositivos Apple Vision Pro.



Developer Information

Camera Control REST API

If you are a software or hardware developer you can build custom applications or leverage ready to use tools such as REST client or Postman to seamlessly control and interact with your compatible Blackmagic camera using Camera Control REST API. This API enables you to perform a wide range of operations, such as starting or stopping recordings, accessing disk information and much more. Whether you're developing a custom application tailored to your specific needs or utilizing existing tools, this API empowers you to unlock the full potential of your Blackmagic camera with ease. We look forward to seeing what you come up with!

NOTE It's important to mention that controlling Blackmagic cameras via REST API relies on the web manager being enabled on each compatible Blackmagic camera. Enable the web media manager in the Blackmagic Camera Setup 'network access' settings for each camera you are controlling.

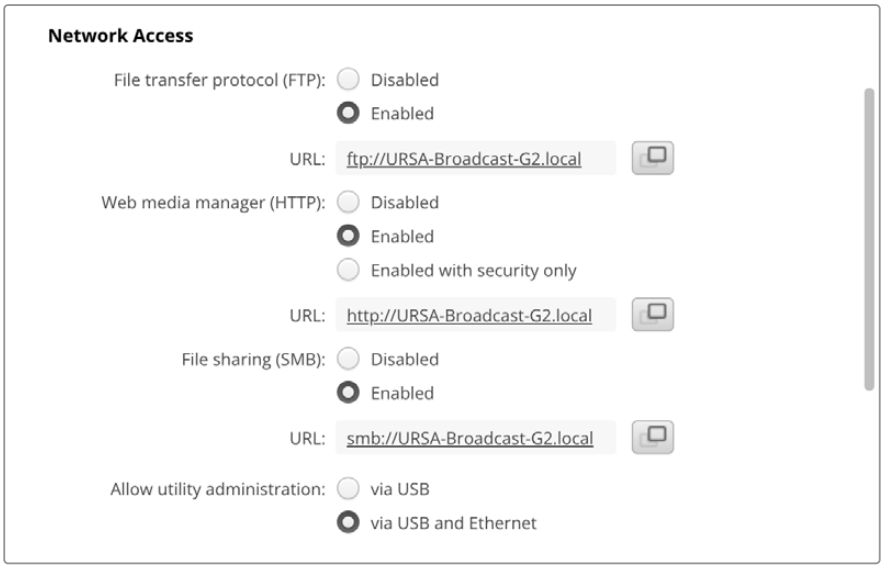
The following Blackmagic cameras are compatible with Camera Control REST API:

- Blackmagic URSA Cine 12K LF
- Blackmagic URSA Cine 17K 65
- Blackmagic URSA Cine Immersive
- Blackmagic PYXIS 6K
- Blackmagic Cinema Camera 6K
- Blackmagic URSA Broadcast G2
- Blackmagic Micro Studio Camera 4K G2
- Blackmagic Studio Camera 4K Plus
- Blackmagic Studio Camera 4K Pro
- Blackmagic Studio Camera 6K Pro
- Blackmagic Studio Camera 4K Plus G2
- Blackmagic Studio Camera 4K Pro G2

Sending API Commands

To send an API command to your camera from a third party application such as Postman, add /control/api/v1/ to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/api/v1/

You can find the Web media manager URL and IP address information in Blackmagic Camera Setup.



The Web media manager URL in Blackmagic Camera Setup

Downloading API's from your Camera

You can download REST API YAML documentation from your camera by adding /control/documentation.html to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/documentation.html

NOTE It's worth noting that changing the camera name in Blackmagic Camera Setup will also change the camera's Web media manager URL.

Livestream Control API

API for controlling Livestreams on Blackmagic Design products.

GET /livestreams/0

Get the livestream's current status.

Response

200 - Livestream's current status.

The response is JSON.

Name	Type	Description
status (required)	string	Possible values are: Idle, Connecting, Streaming, Flushing, Interrupted.
bitrate (required)	integer	Current bitrate (bps).
effectiveVideoFormat (required)	string	Effective video format for the livestream, serialised as a string.
duration	integer	Current stream duration in seconds. Absent if livestream is idle.
cache	integer	Current stream cache usage percentage.

GET /livestreams/0/start

Determine if the livestream is active.

Response

200 - Livestream active status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is active.

PUT /livestreams/0/start

Start the livestream.

Response

204 - Livestream started.

GET /livestreams/0/stop

Determine if the livestream is inactive.

Response

200 - Livestream inactive status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is inactive.

PUT /livestreams/0/stop

Stop the livestream.

Response

204 - Livestream stopped.

GET /livestreams/0/activePlatform

Get the currently selected platform configuration for the livestream.

Response

200 - Livestream active platform configuration.

The response is JSON.

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

PUT /livestreams/0/activePlatform

Set the currently selected platform configuration for the livestream.

Parameters

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

Response

204 - Livestream active platform configuration updated.

400 - Bad Request

GET /livestreams/platforms

Get the list of available platforms.

Response

200 - List of available platforms.

The response is JSON.

Name	Type	Description
	array	List of available platforms names.
[i]	string	Platform name.

GET /livestreams/platforms/{platformName}

Get the service configuration for a platform.

Parameters

Name	Type	Description
{platformName} (required)	string	Name of the platform.

Response

200 - Service configuration for specified platform.

The response is JSON.

Livestream platform service configuration.

Name	Type	Description
platform (required)	string	Corresponding platform name.
key	string	Default stream key.
servers (required)	array	List of server configurations.
servers[i]	object	Server configuration.
servers[i].server (required)	string	Server name.
servers[i].url (required)	string	Livestream destination.
servers[i].srtExtensions	array	Miscellaneous tags used for SRT livestreams.
servers[i].srtExtensions[i]	object	Dictionary object mapping SRT tag strings to values.
servers[i].srtExtensions[i][{key}]	string	SRT tag value.
servers[i].group	string	Logical grouping of the server.
profiles (required)	array	List of profile configurations.
profiles[i]	object	Quality configuration.
profiles[i].profile (required)	string	Quality level name.
profiles[i].configs (required)	array	List of video format configurations.
profiles[i].configs[i]	object	Video format configuration for profiles.
profiles[i].configs[i].resolution (required)	string	Video format serialised as a string.
profiles[i].configs[i].fps (required)	string	Frames per second.
profiles[i].configs[i].bitrate (required)	integer	Pixel bitrate (bps).
profiles[i].configs[i].audioBitrate	integer	Audio bitrate (bps).
profiles[i].configs[i].keyFrameInterval	integer	How often a key frame is sent, in seconds.
profiles[i].configs[i].videoCodecs	array	Supported video encoding algorithm/s.

Name	Type	Description
profiles[i].configs[i].videoCodecs[i]	string	Video encoding algorithm. Possible values are: H264, H265.
profiles[i].lowLatency (required)	boolean	If true, fewer frames will be buffered in the livestream.
defaultProfile	string	Quality level name.
credentials	object	Credientials used for RTMP streams.
credentials.username (required)	string	The username part of the creditials. Only used for RTMP streams.
credentials.password (required)	string	Used for RTMP streams, also used as Passphrase for SRT streams.
customizableUrlEnabled	boolean	True when the server URL is customizable.

400 - Bad Request

GET /livestreams/customPlatforms

Get a list of custom platform files.

Response

200 - List of custom platform files.

The response is JSON.

Name	Type	Description
	array	List of custom platform file names.
[i]	string	Custom platform file name.

DELETE /livestreams/customPlatforms

Remove all custom configuration files.

Response

204 - All custom configuration files removed.

GET /livestreams/customPlatforms/{filename}

Get a custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to get.

Response

200 - Custom platform file.

The response is XML.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

404 - Not Found

PUT /livestreams/customPlatforms/{filename}

Update a custom platform file if it exists, if not, create a new file with the given file name.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to update/create.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

Response

204 - Custom platform file created or updated.

400 - Bad Request

DELETE /livestreams/customPlatforms/{filename}

Remove the given custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to be removed.

Response

204 - Custom platform file removed.

404 - Not Found

Clips Control API

API for listing clips on disk.

GET /clips

Get the list of clips on the active disk.

Response

200 - List of clips on the active disk.

The response is JSON.

List of media clips.

Name	Type	Description
clips (required)	array	
clips[i]	object	Media clip.
clips[i].clipUniqueId (required)	integer	Unique ID used to identify this clip.
clips[i].filePath	string	Path to the file relative to the root of a mount.
clips[i].fileSize	integer	Size of file on disk in bytes.
clips[i].codecFormat	object	
clips[i].codecFormat.codec	string	Currently selected codec.

Name	Type	Description
clips[i].codecFormat.container	string	Multimedia container format.
clips[i].videoFormat	object	Video format configuration.
clips[i].videoFormat.name (required)	string	Video format serialised as a string.
clips[i].videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
clips[i].videoFormat.height	number	Height dimension of video format.
clips[i].videoFormat.width	number	Width dimension of video format.
clips[i].videoFormat.interlaced	boolean	Is the display format interlaced?
clips[i].startTimecode	string	Start timecode of the clip serialised as string.
clips[i].durationTimecode	string	Duration of the clip in timecode format serialised as string.
clips[i].frameCount	integer	Number of frames in clip; duration of the clip in frames.

404 - There is no active disk.

Media Pool Control API

API to manage media pool and handle uploads and project data.

GET /cloud/projects

List all projects within the media pool.

Response

200 - Successfully retrieved the list of all projects.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].libraryID	string	
[i].name	string	
[i].private	boolean	
[i].shared	boolean	
[i].clips	array	List of clips associated with the project.
[i].clips[i]	string	
[i].status	object	
[i].status.numClipsRequested	integer	
[i].status.numClipsComplete	integer	
[i].status.uploadPercent	integer	Percentage of upload completion.
[i].status.numClipsPaused	integer	
[i].status.outOfSpace	boolean	
[i].status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
[i].status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/active

Retrieve data of the actively uploading project.

Response

200 - Successfully retrieved the active project's data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/{projectID}

Retrieve specific project data by project ID.

Parameters

Name	Type	Description
{projectID} (required)	integer	Unique identifier of the project.

Response

200 - Successfully retrieved the project data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	

Name	Type	Description
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

404 - Project not found.

GET /cloud/clips

List all clips within the media pool.

Response

200 - Successfully retrieved the list of all clips.

The response is JSON.

Name	Type	Description
	array	
[i]	string	

GET /cloud/clips/activeUploading

Retrieve data of actively uploading clips.

Response

200 - Successfully retrieved the list of actively uploading clips.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].path	string	
[i].projectID	integer	
[i].status	object	
[i].status.projectID	integer	
[i].status.outOfSpace	boolean	
[i].status.proxyExtension	string	
[i].status.growingFile	boolean	
[i].status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.originalClipTotalSize	integer	
[i].status.proxyClipTotalSize	integer	
[i].status.originalClipCompletedSize	integer	
[i].status.proxyClipCompletedSize	integer	
[i].status.secsRemaining	integer	

GET /cloud/clips/{deviceName}/{path}

Retrieve specific clip data by device and path.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored.
{path} (required)	string	Path to the clip.

Response

200 - Successfully retrieved the clip data.

The response is JSON.

Name	Type	Description
path	string	
projectID	integer	
status	object	
status.projectID	integer	
status.outOfSpace	boolean	
status.proxyExtension	string	
status.growingFile	boolean	
status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.originalClipTotalSize	integer	
status.proxyClipTotalSize	integer	
status.originalClipCompletedSize	integer	
status.proxyClipCompletedSize	integer	
status.secsRemaining	integer	

404 - Clip not found.

Monitoring Control API

API for monitoring and controlling display settings in video equipment.

GET /monitoring/display

Retrieve a list of all display names.

Response

200 - Returns a list of display names.

The response is JSON.

Name	Type	Description
displays	array	List of display names available.
displays[i]	string	

GET /monitoring/{displayName}/cleanFeed

Get the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/cleanFeed

Set the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Clean feed enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/displayLUT

Get the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

400 - Invalid display name.

404 - Display name not found.

PUT /monitoring/{displayName}/displayLUT

Set the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Display LUT enabled/disabled successfully.

400 - Invalid input.

422 - Unprocessable Entity - Unable to process the contained instructions.

GET /monitoring/{displayName}/zebra

Get the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/zebra

Set the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Zebra enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/focusAssist

Get the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/focusAssist

Set the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/focusAssist

Get the focus assist settings.

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

404 - Display name not found.

PUT /monitoring/focusAssist

Set the focus assist settings.

Parameters

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/frameGuide

Get the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame guide enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGuide

Set the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame guide state updated successfully.

422 - Unable to update the frame guide state.

GET /monitoring/frameGuideRatio

Get the current frame guide ratio.

Response

200 - Returns the current frame guide ratio.

The response is JSON.

Name	Type	Description
ratio	string	The frame guide ratio.

PUT /monitoring/frameGuideRatio

Set the frame guide ratio.

Parameters

Name	Type	Description
ratio	string	The frame guide ratio.

Response

204 - Frame guide ratio updated successfully.

422 - Unable to update the frame guide ratio.

GET /monitoring/frameGuideRatio/presets

Get the presets for frame guide ratios.

Response

200 - Returns a list of preset frame guide ratios.

The response is JSON.

Name	Type	Description
presets	array	
presets[i]	string	A frame guide ratio.

GET /monitoring/{displayName}/frameGrids

Get the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame grids enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGrids

Set the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame grids state updated successfully.

422 - Unable to update the frame grids state.

GET /monitoring/frameGrids

Get the global frame grids settings.

Response

200 - Returns the current frame grids settings.

The response is JSON.

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

PUT /monitoring/frameGrids

Set the global frame grids settings.

Parameters

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

Response

204 - Frame grids settings updated successfully.

400 - Invalid input, check the number of frame grids or values.

422 - Unable to update the frame grids settings.

GET /monitoring/{displayName}/safeArea

Get the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the safe area enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/safeArea

Set the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Safe area state updated successfully.

422 - Unable to update the safe area state.

GET /monitoring/safeAreaPercent

Get the current safe area percentage.

Response

200 - Returns the current safe area percentage.

The response is JSON.

Name	Type	Description
percent	integer	Safe area coverage percentage.

PUT /monitoring/safeAreaPercent

Set the safe area percentage.

Parameters

Name	Type	Description
percent	integer	Safe area coverage percentage to set.

Response

204 - Safe area percentage updated successfully.

400 - Invalid percentage value.

422 - Unable to update the safe area percentage.

GET /monitoring/{displayName}/falseColor

Get the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the false color enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/falseColor

Set the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - False color state updated successfully.

422 - Unable to update the false color state.

Event Control API

API For working with built-in websocket.

GET /event/list

Get the list of events that can be subscribed to using the websocket API.

Response

200 - Websocket API events list.

The response is JSON.

Name	Type	Description
events	array	List of events that can be subscribed to using the websocket API.
events[i]	string	

System Control API

API for controlling the System Modes on Blackmagic Design products.

GET /system

Get device system information.

Response

200 - System summary.

The response is JSON.

The properties will be populated only with the values that are supported/implemented by the device in use.

Name	Type	Description
codecFormat	object	Codec format configuration.
codecFormat.codec	string	Codec serialised as string.
codecFormat.container	string	Multimedia container format.
videoFormat	object	Video format configuration.
videoFormat.name (required)	string	Video format serialised as a string.
videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
videoFormat.height	number	Height dimension of video format.
videoFormat.width	number	Width dimension of video format.
videoFormat.interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/product

Get device product information.

Response

200 - Device product information.

The response is JSON.

Product information.

Name	Type	Description
deviceName	string	Name of device as displayed in Setup.
productName	string	Device's product name.
softwareVersion	string	Software version running on device.

501 - This functionality is not implemented for the device in use.

GET /system/supportedCodecFormats

Get the list of supported codecs.

Response

200 - List of supported codec formats.

The response is JSON.

Name	Type	Description
codecs	array	
codecs[i]	object	Codec format configuration.
codecs[i].codec	string	Codec serialised as string.
codecs[i].container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

GET /system/codecFormat

Get the currently selected codec.

Response

200 - Current codec format.

The response is JSON.

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

PUT /system/codecFormat

Update the system codec.

Parameters

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

Response

204 - The codec updated successfully.

400 - The specified codec format is unsupported.

501 - This functionality is not implemented for the device in use.

GET /system/videoFormat

Get the currently selected video format.

Response

200 - Current system video format.

The response is JSON.

Video format configuration.

Name	Type	Description
name (required)	string	Video format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height	number	Height dimension of video format.
width	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

PUT /system/videoFormat

Set the system video format.

Parameters

This parameter can be one of the following types:

Name	Type	Description
name (required)	string	Video format serialised as a string.

Name	Type	Description
frameRate (required)	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height (required)	number	Height dimension of video format.
width (required)	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

Response

204 - The video format updated successfully.

400 - Invalid request.

409 - Operation unsupported in the current state.

501 - This functionality is not implemented for the device in use.

GET /system/supportedVideoFormats

Get the list of supported video formats for the current system state.

Response

200 - List of supported video formats.

The response is JSON.

List of supported video formats.

Name	Type	Description
formats	array	List of video formats.
formats[i]	object	Video format configuration.
formats[i].name (required)	string	Video format serialised as a string.
formats[i].frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
formats[i].height	number	Height dimension of video format.
formats[i].width	number	Width dimension of video format.
formats[i].interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/supportedFormats

Get supported formats.

Response

200 - List of supported formats.

The response is JSON.

Name	Type	Description
supportedFormats	array	
supportedFormats[i]	object	
supportedFormats[i].codecs	array	
supportedFormats[i].codecs[i]	string	
supportedFormats[i].frameRates	array	
supportedFormats[i].frameRates[i]	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
supportedFormats[i]. maxOffSpeedFrameRate	number	
supportedFormats[i]. minOffSpeedFrameRate	number	
supportedFormats[i]. recordResolution	object	
supportedFormats[i]. recordResolution.height	number	Height of the resolution.
supportedFormats[i]. recordResolution.width	number	Width of the resolution.
supportedFormats[i]. sensorResolution	object	
supportedFormats[i]. sensorResolution.height	number	Height of the resolution.
supportedFormats[i]. sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

GET /system/format

Get current format.

Response

200 - Current format.

The response is JSON.

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

PUT /system/format

Set the format.

Parameters

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

Response

204 - System format updated.

501 - This functionality is not implemented for the device in use.

Transport Control API

API for controlling Transport on Blackmagic Design products.

GET /transports/0

Get device's basic transport status.

Response

200 - Transport status.

The response is JSON.

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

PUT /transports/0

Set device's basic transport status.

Parameters

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, Output.

Response

204 - Transport mode was set.

400 - Failed to set transport mode.

GET /transports/0/stop

Determine if transport is stopped.

Response

200 - Transport stop response.

The response is JSON.

Name	Type	Description
	boolean	True when transport mode is InputPreview or when in Output mode and speed is 0.

PUT /transports/0/stop

Stop transport. Deprecated, use POST /transports/0/stop instead.

Response

204 - Transport stopped.

POST /transports/0/stop

Stop transport.

Response

204 - Transport stopped.

GET /transports/0/play

Determine if transport is playing.

Response

200 - Transport play response.

The response is JSON.

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

PUT /transports/0/play

Start playing on transport. Deprecated, use POST /transports/0/play instead.

Response

204 - Transport playing.

400 - Failed to set transport to play.

POST /transports/0/play

Start playing on transport.

Response

204 - Transport playing.

400 - Failed to set transport to play.

GET /transports/0/playback

Get playback state.

Response

200 - Transport playback state.

The response is JSON.

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

PUT /transports/0/playback

Set playback state.

Parameters

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

Response

204 - Updated transport playback state.

400 - Failed to set transport playback state.

GET /transports/0/record

Get record state.

Response

200 - Recording state.

The response is JSON.

Name	Type	Description
recording	boolean	If true, transport is in InputRecord mode.

PUT /transports/0/record

Set record state. Deprecated, use POST /transports/0/record instead.

Parameters

Name	Type	Description
recording (required)	boolean	If true, starts a recording, otherwise stops.
clipName	string	Optional, sets the requested clip name to record to, when “recording” attribute is set to true.

Response

204 - Recording state updated.

400 - Failed to update recording state.

POST /transports/0/record

Start recording.

Parameters

Name	Type	Description
clipName	string	Optional, provides a specific name of clip to record to.

Response

204 - Recording started.

400 - Failed to start recording.

GET /transports/0/clipIndex

Get the clip index of the currently playing clip on the timeline.

Response

200 - Clip index response.

The response is JSON.

Name	Type	Description
clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

GET /transports/0/timecode

Get device timecode.

Response

200 - Timecode response.

The response is JSON.

Name	Type	Description
display	string	The display timecode serialised as a string.
timeline	string	The timeline timecode serialised as a string.

GET /transports/0/timecode/source

Get timecode source selected on device.

Response

200 - Timecode source response.

The response is JSON.

Name	Type	Description
timecode	string	Possible values are: Timeline, Clip.

Timeline Control API

API for controlling playback timeline.

GET /timelines/0

Get the playback timeline.

Response

200 - Playback timeline.

The response is JSON.

Name	Type	Description
clips	array	
clips[i]	object	Timeline clip.
clips[i].clipUniqueId (required)	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).
clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

404 - No timeline / disk available.

DELETE /timelines/0

Clear the current playback timeline. Deprecated, prefer to use POST /timelines/0/clear

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

POST /timelines/0

Add a clip to the timeline.

Parameters

This parameter can be one of the following types:

Add multiple media clips to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips[i].clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips[i].clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips[i].frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add multiple media clips to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add a single clip to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips.clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips.clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips.frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add a single clip to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the timeline as specified.

501 - The operation is not supported on this device.

POST /timelines/0/add

Add a clip to the end of the timeline. Deprecated, use POST /timelines/0 to add clips within the timeline.

Parameters

This parameter can be one of the following types:

Add one clip to the end of the timeline.

Name	Type	Description
clips	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add many clips to the end of the timeline.

Name	Type	Description
clips	array	List of clipUniqueIds of clips to add to end of timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the end of the timeline.

501 - The operation is not supported on this device.

POST /timelines/0/clear

Clear the playback timeline.

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

DELETE /timelines/0/clips/{timelineClipIndex}

Remove the specified clip from the timeline.

Parameters

Name	Type	Description
{timelineClipIndex} (required)	integer	The (0-based) timeline clip index of the clip to remove from the timeline.

Response

204 - The specified clip was removed from the timeline.

501 - The operation is not supported on this device.

Media Control API

API for controlling media devices in Blackmagic Design products.

GET /media/workingset

Get the list of media devices currently in the working set.

Response

200 - The list of media devices in the working set.

The response is JSON.

Name	Type	Description
size	integer	The fixed size of this device's working set.
workingset	array	The device's working set.
workingset[i]	object | null	Device within the working set. null if no device is present within the given working set slot.
workingset[i].volume	string	Volume name.
workingset[i].deviceName	string	Internal device name of this media device.
workingset[i].remainingRecordTime	integer	Remaining record time using current codec and video format in seconds.
workingset[i].totalSpace	integer	Total space on media device in bytes.
workingset[i].remainingSpace	integer	Remaining space on media device in bytes.
workingset[i].clipCount	integer	Number of clips currently on the device.

GET /media/active

Get the currently active media device.

Response

200 - The current active media device.

The response is JSON.

The active media device, or null if there is no active media.

Name	Type	Description
workingsetIndex	integer	Working set index of the active media device.
deviceName	string	Device name of media device.

204 - No media is currently active.

PUT /media/active

Set the currently active media device.

Parameters

Name	Type	Description
workingsetIndex	integer	Working set index of the media to make active.

Response

204 - The active media device was set successfully.

400 - Setting the currently active media device is not possible in the current state.

GET /media/devices/doformatSupportedFilesystems

Get the list of filesystems available to format a media device.

Response

200 - The list of filesystems permitted for formatting.

The response is JSON.

Name	Type	Description
	array	List of filesystems permitted for formatting media.
[i]	string	Filesystem serialised as string.

GET /media/devices/{deviceName}

Get information about a requested device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Information about the requested device.

The response is JSON.

Media device state.

Name	Type	Description
state	string	The current state of the media device. Possible values are: None, Scanning, Mounted, Uninitialised, Formatting, RaidComponent.

400 - Invalid device name.

404 - Device not found.

GET /media/devices/{deviceName}/doformat

Get a format key, used to format the device with a PUT request.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Format prepared.

The response is JSON.

Name	Type	Description
deviceName	string	Device name of media device to format.
key	string	The key required to format this device, provide to PUT /media/devices/{deviceName}/doformat to perform format of media device.

400 - Cannot format the device.

404 - Device not found.

PUT /media/devices/{deviceName}/doformat

Perform a format of the specified media device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by "deviceName" member of GET /media/workingset or GET /media/active.

Name	Type	Description
key	string	The key used to format this device, retrieved from prepare format media request GET /media/devices/{deviceName}/doformat. Format key provided cannot be reused after successful format.
filesystem	string	Filesystem to format to. Supported filesystems can be retrieved with GET /media/devices/doFormatSupportedFilesystems.
volume	string	Volume name to set for the disk after format.

Response

204 - Format successful.

400 - Cannot format the device, invalid filesystem or key.

404 - Device not found.

Slate Control API

API to manage digital slate data.

GET /slates/nextClip

Retrieve the digital slate for the next clip.

Response

200 - Returns the slate data for the next clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	

Name	Type	Description
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Slate data is not available.

PUT /slates/nextClip

Update the slate data for the next clip.

Parameters

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

POST /slates/nextClip/resetProjectData

Reset the project data for the next clip's slate.

Response

200 - Project data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetProjectData

Reset the project data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Project data reset successfully.

POST /slates/nextClip/resetLensData

Reset the lens data for the next clip's slate.

Response

200 - Lens data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetLensData

Reset the lens data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Lens data reset successfully.

GET /slates/clips/{deviceName}/{path}

Retrieve slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Returns the slate data for the specified clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

404 - Clip not found.

PUT /slates/clips/{deviceName}/{path}

Update the slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.

Name	Type	Description
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

Preset Control API

API For controlling the presets on Blackmagic Design products

GET /presets

Get the list of the presets on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presets	array	List of the presets on the camera (.cset files)
presets[i]	string	

POST /presets

Send a preset file to the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presetAdded	string	Name of the preset uploaded (without .cset extension)

400 - Bad request - missing Content-Disposition header or filename

GET /presets/active

Get the currently active preset on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

PUT /presets/active

Set the active preset on the camera

Parameters

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

Response

204 - No Content

404 - Preset file not found

PARAMETERS /presets/{presetName}

GET /presets/{presetName}

Download the preset file

Response

200 - OK

The response is a binary file.

404 - File does not exist

PUT /presets/{presetName}

Save current camera state as a preset

Response

204 - No Content

DELETE /presets/{presetName}

Delete a preset from the camera

Response

204 - No Content

404 - Preset file not found

Audio Control API

API For controlling audio on Blackmagic Design Cameras

GET /audio/channels

Get the total number of audio channels available

Response

200 - Returns the total number of channels

The response is JSON.

Name	Type	Description
channels	integer	Total number of audio channels available

GET /audio/supportedInputs

Get the list of supported audio inputs

Response

200 - List of all supported audio inputs

The response is JSON.

Name	Type	Description
	array	
[i]	string	A supported audio input

GET /audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently selected input

The response is JSON.

Name	Type	Description
input	string	Audio input source and type

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/input

Set the audio input for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
input	string	Audio input source and type

Response

204 - No Content

400 - Invalid audio input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/input/description

Get the description of the current input of the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Description of the current input of the selected channel

The response is JSON.

Name	Type	Description
description	object	
description.gainRange	object	
description.gainRange.Min	number	The minimum gain value in dB
description.gainRange.Max	number	The maximum gain value in dB
description.capabilities	object	
description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter
description.capabilities.Padding	object	
description.capabilities.Padding. available	boolean	Input supports setting of padding
description.capabilities.Padding. forced	boolean	Padding is forced to be set for the input
description.capabilities.Padding. value	number	Value of the padding in dB

404 - Channel does not exist

GET /audio/channel/{channelIndex}/supportedInputs

Get the list of supported inputs and their availability to switch to for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its supported inputs are being queried. (Channels index from 0)

Response

200 - The list of supported inputs

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

404 - Channel does not exist

GET /audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set level for the selected channel

The response is JSON.

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/level

Set the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

Response

204 - No Content

400 - Invalid input or value out of range

404 - Channel does not exist

GET /audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set phantom power for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Phantom power enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/phantomPower

Set the audio phantom power for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Phantom power enabled state

Response

204 - No Content

400 - Phantom power is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set padding for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Padding enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/padding

Set the audio input padding for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Padding enabled state

Response

204 - No Content

400 - Padding is not supported or is forced for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set low cut filter for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Low cut filter enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/lowCutFilter

Set the audio input low cut filter for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Low cut filter enabled state

Response

204 - No Content

400 - Low cut filter is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set availability for the selected channel

The response is JSON.

Name	Type	Description
available	boolean	Whether the input is currently available

404 - Channel does not exist

Lens Control API

API For controlling the lens on Blackmagic Design products

GET /lens/iris

Get lens' aperture

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number

PUT /lens/iris

Set lens' aperture

Parameters

Name	Type	Description
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number
adjustmentStep	integer	Signed value for relative aperture adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens iris is not controllable or is controlled by auto exposure

GET /lens/zoom

Get lens' zoom

Response

200 - OK

The response is JSON.

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value

PUT /lens/zoom

Set lens' zoom

Parameters

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value
adjustmentFocalLength	integer	Signed value for relative focal length adjustment
adjustmentNormalised	number	Signed normalized value for relative zoom adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens zoom is not controllable

GET /lens/focus

Get lens' focus

Response

200 - OK

The response is JSON.

Name	Type	Description
normalised	number	Normalised value

PUT /lens/focus

Set lens' focus

Parameters

Name	Type	Description
normalised	number	Normalised value
focusDistance	integer	Focus distance value

Response

204 - No Content

400 - Bad Request if out of range value is provided

PUT /lens/focus/doAutoFocus

Perform auto focus

Parameters

Name	Type	Description
position	object	
position.x	number	Normalized x coordinate for autofocus ROI
position.y	number	Normalized y coordinate for autofocus ROI

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens focus is not controllable

GET /lens/opticalImageStabilization

Get optical image stabilization status

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Whether optical image stabilization is enabled

501 - Not Implemented if optical image stabilization is not supported on this product

PUT /lens/opticalImageStabilization

Enable or disable optical image stabilization

Parameters

Name	Type	Description
enabled	boolean	Enable or disable optical image stabilization

Response

204 - No Content

501 - Not Implemented if optical image stabilization is not supported on this product

GET /lens/iris/description

Get detailed description of lens' iris capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the iris can be controlled
apertureStop	object	
apertureStop.min	number	Minimum aperture stop
apertureStop.max	number	Maximum aperture stop

GET /lens/zoom/description

Get detailed description of lens' zoom capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the zoom can be controlled
focalLength	object	
focalLength.adjustable	boolean	If focal length is adjustable
focalLength.min	integer	Minimum focal length
focalLength.max	integer	Maximum focal length

GET /lens/focus/description

Get detailed description of lens' focus capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the focus can be controlled
focusDistance	object	
focusDistance.adjustable	boolean	If focus distance is adjustable
focusDistance.min	number	Minimum focus distance
focusDistance.max	number	Maximum focus distance

Video Control API

API For controlling the video on Blackmagic Design products

GET /video/iso

Get current ISO

Response

200 - OK

The response is JSON.

Name	Type	Description
iso	integer	Current ISO value

PUT /video/iso

Set current ISO

Parameters

Name	Type	Description
iso	integer	ISO value to set

Response

204 - No Content

403 - ISO cannot be changed in the current state

GET /video/supportedISOs

Get the list of supported ISO settings

Response

200 - List of supported ISO values

The response is JSON.

Name	Type	Description
supportedISOs	array	Array of supported ISO values
supportedISOs[i]	integer	

GET /video/gain

Get current gain value in decibels

Response

200 - OK

The response is JSON.

Name	Type	Description
gain	integer	Current gain value in decibels

PUT /video/gain

Set current gain value

Parameters

Name	Type	Description
gain	integer	Gain value in decibels to set

Response

204 - No Content

403 - Gain cannot be changed in the current state

GET /video/supportedGains

Get the list of supported gain settings in decibels

Response

200 - List of supported gain values in decibels

The response is JSON.

Name	Type	Description
supportedGains	array	Array of supported gain values in decibels
supportedGains[i]	integer	

GET /video/whiteBalance

Get current white balance

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	integer	Current white balance

PUT /video/whiteBalance

Set current white balance

Parameters

Name	Type	Description
whiteBalance	integer	White balance to set

Response

204 - No Content

400 - Invalid white balance temperature

GET /video/whiteBalance/description

Get white balance range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	object	
whiteBalance.min	integer	Minimum color temperature
whiteBalance.max	integer	Maximum color temperature

PUT /video/whiteBalance/doAuto

Set current white balance automatically

Response

204 - No Content

GET /video/whiteBalanceTint

Get white balance tint

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	integer	Current white balance tint

PUT /video/whiteBalanceTint

Set white balance tint

Parameters

Name	Type	Description
whiteBalanceTint	integer	White balance tint to set

Response

204 - No Content

400 - Invalid white balance tint

GET /video/whiteBalanceTint/description

Get white balance tint range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	object	
whiteBalanceTint.min	integer	Minimum white balance tint
whiteBalanceTint.max	integer	Maximum white balance tint

GET /video/ndFilter

Get ND filter stop

Response

200 - OK

The response is JSON.

Name	Type	Description
stop	number	Current filter power (fStop)

501 - Not implemented for this device

PUT /video/ndFilter

Set ND filter stop

Parameters

Name	Type	Description
stop	number	Filter power (fStop) to set

Response

204 - No Content

400 - Invalid ND filter stop

501 - Not implemented for this device

GET /video/supportedNDFilters

Get the list of available ND filter stops

Response

200 - List of available ND filter stops

The response is JSON.

Name	Type	Description
supportedStops	array	Array of available ND filter stops
supportedStops[i]	number	

501 - Not implemented for this device

GET /video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

Response

200 - List of supported display modes

The response is JSON.

Name	Type	Description
supportedDisplayModes	array	Array of supported display modes
supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

GET /video/ndFilter/displayMode

Get ND filter display mode on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

PUT /video/ndFilter/displayMode

Set ND filter display mode on the camera

Parameters

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

Response

204 - No Content

400 - Invalid display mode for ND filter

501 - Not implemented for this device

GET /video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

Response

200 - Indicates if ND filter is selectable

The response is JSON.

Name	Type	Description
selectable	boolean	True if ND filter adjustments are selectable via a slider

501 - Not implemented for this device

GET /video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

PUT /video/shutter

Set current shutter

Parameters

Name	Type	Description
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

Response

204 - No Content

GET /video/shutter/measurement

Get the current shutter measurement mode

Response

200 - OK

The response is JSON.

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

PUT /video/shutter/measurement

Set the shutter measurement mode

Parameters

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

Response

204 - No Content

400 - Invalid shutter measurement

GET /video/supportedShutters

Get supported shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of supported shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of supported shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of flicker-free shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of flicker-free shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/autoExposure

Get current auto exposure mode

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

PUT /video/autoExposure

Set auto exposure

Parameters

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

Response

204 - No Content

400 - Failed to set auto exposure mode

GET /video/detailSharpening

Get the current state of detail sharpening

Response

200 - Current detail sharpening state

The response is JSON.

Name	Type	Description
enabled	boolean	Whether detail sharpening is enabled

501 - Not implemented for this device

PUT /video/detailSharpening

Enable or disable detail sharpening

Parameters

Name	Type	Description
enabled	boolean	Enable or disable detail sharpening

Response

204 - Detail sharpening state updated

501 - Not implemented for this device

GET /video/detailSharpeningLevel

Get the current detail sharpening level

Response

200 - Current detail sharpening level

The response is JSON.

Name	Type	Description
level	string	Current detail sharpening level Possible values are: Low, Medium, High.

501 - Not implemented for this device

PUT /video/detailSharpeningLevel

Set the detail sharpening level

Parameters

Name	Type	Description
level	string	Desired level of detail sharpening Possible values are: Low, Medium, High.

Response

204 - Detail sharpening level updated

400 - Invalid detail sharpening level

501 - Not implemented for this device

Camera Control API

API For controlling the Camera specific features on Blackmagic Design products

GET /camera/colorBars

Get the status of color bars display

Response

200 - Returns the current status of color bars

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the color bars are currently enabled

PUT /camera/colorBars

Set the status of color bars display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the color bars

Response

204 - Color bars status updated successfully

GET /camera/programFeedDisplay

Get the status of program feed display

Response

200 - Returns the current status of program feed display

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the program feed display is currently enabled

PUT /camera/programFeedDisplay

Set the status of program feed display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the program feed display

Response

204 - Program feed display status updated successfully

GET /camera/tallyStatus

Get the tally status of the camera

Response

200 - Returns the current tally status of the camera

The response is JSON.

Name	Type	Description
status	string	Current tally status of the camera Possible values are: None, Preview, Program.

GET /camera/power

Get the power status of the camera

Response

200 - Returns the current power status

The response is JSON.

Name	Type	Description
source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
batteries	array	List of batteries currently connected to the camera
batteries[i]	object	
batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
batteries[i].chargeRemainingPercent	integer	Remaining battery charge percentage
batteries[i].statusFlags	array	List of battery status flags
batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

GET /camera/power/displayMode

Get the power display mode of the camera

Response

200 - Returns the current power display mode

The response is JSON.

Name	Type	Description
mode	string	Current power display mode Possible values are: Percentage, Voltage.

PUT /camera/power/displayMode

Set the power display mode of the camera

Parameters

Name	Type	Description
mode	string	Power display mode to set Possible values are: Percentage, Voltage.

Response

204 - Power display mode updated successfully

400 - Invalid power display mode

GET /camera/timingReferenceLock

Get the timing reference lock status

Response

200 - Returns the timing reference lock status

The response is JSON.

Name	Type	Description
locked	boolean	Indicates if timing reference is locked

Immersive Control API

API for controlling immersive camera settings on Blackmagic Design cameras

GET /immersive/display/{displayName}/eye

Get the current eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to query (from /monitoring/display endpoint)

Response

200 - OK

The response is JSON.

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

400 - Invalid display name format

404 - Display not found

422 - Failed to get eye view

PUT /immersive/display/{displayName}/eye

Set the eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to control (from /monitoring/display endpoint)

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

Response

204 - No Content

400 - Invalid input or display name format

404 - Display not found

422 - Failed to set eye view

Color Correction Control API

API For controlling the color correction on Blackmagic Design products based on DaVinci Resolve Color Corrector

GET /colorCorrection/lift

Get color correction lift

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

PUT /colorCorrection/lift

Set color correction lift

Parameters

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gamma

Get color correction gamma

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

PUT /colorCorrection/gamma

Set color correction gamma

Parameters

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gain

Get color correction gain

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

PUT /colorCorrection/gain

Set color correction gain

Parameters

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/offset

Get color correction offset

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

PUT /colorCorrection/offset

Set color correction offset

Parameters

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/contrast

Get color correction contrast

Response

200 - OK

The response is JSON.

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/contrast

Set color correction contrast

Parameters

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/color

Get color correction color properties

Response

200 - OK

The response is JSON.

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/color

Set color correction color properties

Parameters

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/lumaContribution

Get color correction luma contribution

Response

200 - OK

The response is JSON.

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

PUT /colorCorrection/lumaContribution

Set color correction luma contribution

Parameters

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

Response

204 - No Content

Notification websocket - 1.0.0

Service that notifies subscribers of device state changes.

messages

Subscribe (The messages from the server/device)

Websocket Opened Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: websocketOpened.
.type	string	Possible values are: event.

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*

Name	Type	Description
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Event Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: propertyValueChanged.

Name	Type	Description
.data.property	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.value	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.type	string	Possible values are: event.

Publish (The messages that user can send to the server/device)

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	

Name	Type	Description
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	

Name	Type	Description
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Device Properties

/media/workingset

The value JSON returned via the eventResponse when the `/media/workingset` property changes on the device:

Name	Type	Description
.size	integer	The fixed size of this device's working set.
.workingset	array	Array of devices within the working set. null if no device is present within the given working set slot.
.workingset[i]		

/media/active

The value JSON returned via the eventResponse when the `/media/active` property changes on the device:

Name	Type	Description
.workingsetIndex	integer	Working set index of the active media device.
.deviceName	string	Internal device name of this media device.

/system

The value JSON returned via the eventResponse when the `/system` property changes on the device:

Name	Type	Description
.codecFormat	object	Codec format configuration.
.codecFormat.codec	string	Codec format serialised as a string.
.codecFormat.container	string	Multimedia container format.
.videoFormat	object	Currently selected video format.
.videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.videoFormat.height	number	Height dimension of video format.
.videoFormat.width	number	Width dimension of video format.
.videoFormat.interlaced	boolean	Is the display format interlaced?.
.videoFormat.name	string	Video format serialised as a string.

/system/codecFormat

Codec format configuration.

The value JSON returned via the eventResponse when the `/system/codecFormat` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.container	string	Multimedia container format.

/system/videoFormat

Currently selected video format.

The value JSON returned via the eventResponse when the `/system/videoFormat` property changes on the device:

Name	Type	Description
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.height	number	Height dimension of video format.
.width	number	Width dimension of video format.
.interlaced	boolean	Is the display format interlaced?.
.name	string	Video format serialised as a string.

/system/format

The value JSON returned via the eventResponse when the `/system/format` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.maxOffSpeedFrameRate	number	
.minOffSpeedFrameRate	number	
.offSpeedEnabled	boolean	
.offSpeedFrameRate	number	
.recordResolution	object	
.recordResolution.height	number	Height of the resolution.
.recordResolution.width	number	Width of the resolution.
.sensorResolution	object	
.sensorResolution.height	number	Height of the resolution.
.sensorResolution.width	number	Width of the resolution.

/system/supportedFormats

The value JSON returned via the eventResponse when the `/system/supportedFormats` property changes on the device:

Name	Type	Description
.supportedFormats	array	
.supportedFormats[i]	object	
.supportedFormats[i].codecs	array	
.supportedFormats[i].codecs[i]	string	
.supportedFormats[i].frameRates	array	
.supportedFormats[i].frameRates[i]	string	Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.supportedFormats[i].maxOffSpeedFrameRate	number	
.supportedFormats[i].minOffSpeedFrameRate	number	
.supportedFormats[i].recordResolution	object	
.supportedFormats[i].recordResolution.height	number	Height of the resolution.
.supportedFormats[i].recordResolution.width	number	Width of the resolution.
.supportedFormats[i].sensorResolution	object	
.supportedFormats[i].sensorResolution.height	number	Height of the resolution.
.supportedFormats[i].sensorResolution.width	number	Width of the resolution.

/timelines/0

The value JSON returned via the eventResponse when the `/timelines/0` property changes on the device:

Name	Type	Description
.clips	array	
.clips[i]	object	Timeline clip.
.clips[i].clipUniqueId	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
.clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
.clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
.clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
.clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).

Name	Type	Description
.clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
.clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

/transports/0

The value JSON returned via the eventResponse when the `/transports/0` property changes on the device:

Name	Type	Description
.mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

/transports/0/stop

true when transport mode is InputPreview or when in Output mode and speed is 0.

The value JSON returned via the eventResponse when the `/transports/0/stop` property changes on the device:

Name	Type	Description
	boolean	true when transport mode is InputPreview or when in Output mode and speed is 0.

/transports/0/play

True when transport is in Output mode and speed is non-zero.

The value JSON returned via the eventResponse when the /transports/0/play property changes on the device:

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

/transports/0/playback

The value JSON returned via the eventResponse when the `/transports/0/playback` property changes on the device:

Name	Type	Description
.type	string	Possible values are: Play, Jog, Shuttle, Var.
.loop	boolean	When true playback loops from the end of the timeline to the beginning of the timeline.
.singleClip	boolean	When true playback loops from the end of the current clip to the beginning of the current clip.
.speed	number	Playback speed, 1.0 for normal forward playback
.position	integer	Playback position on the timeline in units of video frames.

/transports/0/record

The value JSON returned via the eventResponse when the `/transports/0/record` property changes on the device:

Name	Type	Description
.recording	boolean	Is transport in Input Record mode.

/transports/0/timecode

The value JSON returned via the eventResponse when the `/transports/0/timecode` property changes on the device:

Name	Type	Description
.display	string	The display timecode serialised as a string.
.timeline	string	The timeline timecode serialised as a string.

/transports/0/timecode/source

The value JSON returned via the eventResponse when the `/transports/0/timecode/source` property changes on the device:

Name	Type	Description
.timecode	string	Possible values are: Timeline, Clip.

/transports/0/clipIndex

The value JSON returned via the eventResponse when the `/transports/0/clipIndex` property changes on the device:

Name	Type	Description
.clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

/slates/nextClip

The value JSON returned via the eventResponse when the `/slates/nextClip` property changes on the device:

Name	Type	Description
.clip	object	
.clip.clipName	string	
.clip.reel	integer	
.clip.scene	string	
.clip.sceneLocation	string	Possible values are: Interior, Exterior.
.clip.sceneTime	string	Possible values are: Day, Night.
.clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
.clip.take	integer	
.clip.takeType	string	Possible values are: None, PU, VFX, SER.
.clip.goodTake	boolean	
.lens	object	
.lens.lensType	string	
.lens.iris	string	

Name	Type	Description
.lens.focalLength	string	
.lens.distance	string	
.lens.filter	string	
.project	object	
.project.projectName	string	
.project.director	string	
.project.camera	string	
.project.cameraOperator	string	

/monitoring/{displayName}/cleanFeed

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/cleanFeed` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/displayLUT

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/displayLUT` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/zebra

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/zebra` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/focusAssist` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGuide

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGuide` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGrids` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/safeArea

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/safeArea` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/falseColor

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/falseColor` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/focusAssist` property changes on the device:

Name	Type	Description
.mode	string	Possible values are: Peak, ColoredLines.
.color	string	Possible values are: Red, Green, Blue, White, Black.
.intensity	integer	

/monitoring/frameGuideRatio

The value JSON returned via the eventResponse when the `/monitoring/frameGuideRatio` property changes on the device:

Name	Type	Description
.ratio	string	

/monitoring/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/frameGrids` property changes on the device:

Name	Type	Description
.frameGrids	array	
.frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

/monitoring/safeAreaPercent

The value JSON returned via the eventResponse when the `/monitoring/safeAreaPercent` property changes on the device:

Name	Type	Description
.percent	integer	Safe area coverage percentage.

/audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input` property changes on the device:

Name	Type	Description
.input	string	Audio input source and type

/audio/channel/{channelIndex}/supportedInputs

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/supportedInputs` property changes on the device:

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

/audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/level` property changes on the device:

Name	Type	Description
.gain	number	Gain value in dB
.normalised	number	Normalised level value between 0.0 and 1.0

/audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/phantomPower` property changes on the device:

Name	Type	Description
.enabled	boolean	Phantom power enabled state

/audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/padding` property changes on the device:

Name	Type	Description
.enabled	boolean	Padding enabled state

/audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/lowCutFilter` property changes on the device:

Name	Type	Description
.enabled	boolean	Low cut filter enabled state

/audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/available` property changes on the device:

Name	Type	Description
.available	boolean	Whether the input is currently available

/audio/channel/{channelIndex}/input/description

Description of the current input of the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input/description` property changes on the device:

Name	Type	Description
.description	object	
.description.gainRange	object	
.description.gainRange.Min	number	The minimum gain value in dB
.description.gainRange.Max	number	The maximum gain value in dB
.description.capabilities	object	
.description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
.description.capabilities. LowCutFilter	boolean	Input supports setting of low cut filter

Name	Type	Description
.description.capabilities.Padding	object	
.description.capabilities.Padding.available	boolean	Input supports setting of padding
.description.capabilities.Padding.forced	boolean	Padding is forced to be set for the input
.description.capabilities.Padding.value	number	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.

/colorCorrection/lift

Get color correction lift

The value JSON returned via the eventResponse when the `/colorCorrection/lift` property changes on the device:

Name	Type	Description
.red	number	Red lift component
.green	number	Green lift component
.blue	number	Blue lift component
.luma	number	Luma lift component

/colorCorrection/gamma

Get color correction gamma

The value JSON returned via the eventResponse when the `/colorCorrection/gamma` property changes on the device:

Name	Type	Description
.red	number	Red gamma component
.green	number	Green gamma component
.blue	number	Blue gamma component
.luma	number	Luma gamma component

/colorCorrection/gain

Get color correction gain

The value JSON returned via the eventResponse when the `/colorCorrection/gain` property changes on the device:

Name	Type	Description
.red	number	Red gain component
.green	number	Green gain component
.blue	number	Blue gain component
.luma	number	Luma gain component

/colorCorrection/offset

Get color correction offset

The value JSON returned via the eventResponse when the `/colorCorrection/offset` property changes on the device:

Name	Type	Description
.red	number	Red offset component
.green	number	Green offset component
.blue	number	Blue offset component
.luma	number	Luma offset component

/colorCorrection/contrast

Get color correction contrast

The value JSON returned via the eventResponse when the `/colorCorrection/contrast` property changes on the device:

Name	Type	Description
.pivot	number	Contrast pivot point
.adjust	number	Contrast adjustment

/colorCorrection/color

Get color correction color properties

The value JSON returned via the eventResponse when the `/colorCorrection/color` property changes on the device:

Name	Type	Description
.hue	number	Color hue adjustment
.saturation	number	Color saturation adjustment

/colorCorrection/lumaContribution

Get color correction luma contribution

The value JSON returned via the eventResponse when the `/colorCorrection/lumaContribution` property changes on the device:

Name	Type	Description
.lumaContribution	number	Luma contribution value

/lens/iris

Get lens' aperture

The value JSON returned via the eventResponse when the `/lens/iris` property changes on the device:

Name	Type	Description
.continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
.apertureStop	number	Aperture stop value
.normalised	number	Normalised value
.apertureNumber	integer	Aperture number

/lens/iris/description

Get detailed description of lens' iris capabilities

The value JSON returned via the eventResponse when the `/lens/iris/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the iris can be controlled
.apertureStop	object	
.apertureStop.min	number	Minimum aperture stop
.apertureStop.max	number	Maximum aperture stop

/lens/focus

Get lens' focus

The value JSON returned via the eventResponse when the `/lens/focus` property changes on the device:

Name	Type	Description
.normalised	number	Normalised value

/lens/focus/description

Get detailed description of lens' focus capabilities

The value JSON returned via the eventResponse when the `/lens/focus/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the focus can be controlled
.focusDistance	object	
.focusDistance.adjustable	boolean	If focus distance is adjustable
.focusDistance.min	number	Minimum focus distance
.focusDistance.max	number	Maximum focus distance

/lens/zoom

Get lens' zoom

The value JSON returned via the eventResponse when the `/lens/zoom` property changes on the device:

Name	Type	Description
.focalLength	integer	Focal length in mm
.normalised	number	Normalised value

/lens/zoom/description

Get detailed description of lens' zoom capabilities

The value JSON returned via the eventResponse when the `/lens/zoom/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the zoom can be controlled
.focalLength	object	
.focalLength.adjustable	boolean	If focal length is adjustable
.focalLength.min	integer	Minimum focal length
.focalLength.max	integer	Maximum focal length

/presets

Get the list of the presets on the camera

The value JSON returned via the eventResponse when the `/presets` property changes on the device:

Name	Type	Description
.presets	array	List of the presets on the camera (.cset files)
.presets[i]	string	

/presets/active

Get the currently active preset on the camera

The value JSON returned via the eventResponse when the `/presets/active` property changes on the device:

Name	Type	Description
.preset	string	Name of the active preset (with .cset extension, or 'default')

/camera/colorBars

Get the status of color bars display

The value JSON returned via the eventResponse when the `/camera/colorBars` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the color bars are currently enabled

/camera/programFeedDisplay

Get the status of program feed display

The value JSON returned via the eventResponse when the `/camera/programFeedDisplay` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the program feed display is currently enabled

/camera/tallyStatus

Get the tally status of the camera

The value JSON returned via the eventResponse when the `/camera/tallyStatus` property changes on the device:

Name	Type	Description
.status	string	Current tally status of the camera Possible values are: None, Preview, Program.

/camera/power

Get the power status of the camera

The value JSON returned via the eventResponse when the `/camera/power` property changes on the device:

Name	Type	Description
.source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
.milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
.batteries	array	
.batteries[i]	object	
.batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
.batteries[i]. chargeRemainingPercent	integer	Remaining battery charge percentage
.batteries[i].statusFlags	array	List of battery status flags
.batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

/camera/power/displayMode

Get the power display mode of the camera

The value JSON returned via the eventResponse when the `/camera/power/displayMode` property changes on the device:

Name	Type	Description
.mode	string	Current power display mode Possible values are: Percentage, Voltage.

/camera/timingReferenceLock

Get the timing reference lock status

The value JSON returned via the eventResponse when the `/camera/timingReferenceLock` property changes on the device:

Name	Type	Description
.locked	boolean	Indicates if timing reference is locked

/video/iso

Get current ISO

The value JSON returned via the eventResponse when the `/video/iso` property changes on the device:

Name	Type	Description
.iso	integer	Current ISO value

/video/supportedISOs

Get the list of supported ISO settings

The value JSON returned via the eventResponse when the `/video/supportedISOs` property changes on the device:

Name	Type	Description
.supportedISOs	array	Array of supported ISO values
.supportedISOs[i]	integer	

/video/gain

Get current gain value in decibels

The value JSON returned via the eventResponse when the `/video/gain` property changes on the device:

Name	Type	Description
.gain	integer	Current gain value in decibels

/video/supportedGains

Get the list of supported gain settings in decibels

The value JSON returned via the eventResponse when the `/video/supportedGains` property changes on the device:

Name	Type	Description
.supportedGains	array	Array of supported gain values in decibels
.supportedGains[i]	integer	

/video/whiteBalance

Get current white balance

The value JSON returned via the eventResponse when the `/video/whiteBalance` property changes on the device:

Name	Type	Description
.whiteBalance	integer	Current white balance

/video/whiteBalance/description

Get white balance range

The value JSON returned via the eventResponse when the `/video/whiteBalance/description` property changes on the device:

Name	Type	Description
.whiteBalance	object	
.whiteBalance.min	integer	Minimum color temperature
.whiteBalance.max	integer	Maximum color temperature

/video/whiteBalanceTint

Get white balance tint

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint` property changes on the device:

Name	Type	Description
.whiteBalanceTint	integer	Current white balance tint

/video/whiteBalanceTint/description

Get white balance tint range

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint/description` property changes on the device:

Name	Type	Description
.whiteBalanceTint	object	
.whiteBalanceTint.min	integer	Minimum white balance tint
.whiteBalanceTint.max	integer	Maximum white balance tint

/video/ndFilter

Get ND filter stop

The value JSON returned via the eventResponse when the `/video/ndFilter` property changes on the device:

Name	Type	Description
.stop	number	Current filter power (fStop)

/video/supportedNDFilters

Get the list of available ND filter stops

The value JSON returned via the eventResponse when the `/video/supportedNDFilters` property changes on the device:

Name	Type	Description
.supportedStops	array	Array of available ND filter stops
.supportedStops[i]	number	

/video/ndFilter/displayMode

Get ND filter display mode on the camera

The value JSON returned via the eventResponse when the `/video/ndFilter/displayMode` property changes on the device:

Name	Type	Description
.displayMode	string	ND filter display mode Possible values are: Stop, Number, Fraction.

/video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

The value JSON returned via the eventResponse when the `/video/supportedNDFilterDisplayModes` property changes on the device:

Name	Type	Description
.supportedDisplayModes	array	Array of supported display modes
.supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

/video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

The value JSON returned via the eventResponse when the `/video/ndFilterSelectable` property changes on the device:

Name	Type	Description
.selectable	boolean	True if ND filter adjustments are selectable via a slider

/video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

The value JSON returned via the eventResponse when the `/video/shutter` property changes on the device:

Name	Type	Description
.continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
.shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
.shutterAngle	number	Shutter angle

/video/shutter/measurement

Get the current shutter measurement mode

The value JSON returned via the eventResponse when the `/video/shutter/measurement` property changes on the device:

Name	Type	Description
.measurement	string	Current shutter measurement mode Possible values are: ShutterAngle, ShutterSpeed.

/video/supportedShutters

Get supported shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/supportedShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of supported shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array of flicker-free shutter speeds
.shutterSpeeds[i]	integer	

/video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/flickerFreeShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of flicker-free shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array
.shutterSpeeds[i]	integer	

/video/autoExposure

Get current auto exposure mode

The value JSON returned via the eventResponse when the `/video/autoExposure` property changes on the device:

Name	Type	Description
.mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
.type	string	Comma-separated list of device types in the auto exposure stack

/video/detailSharpening

Get the current state of detail sharpening

The value JSON returned via the eventResponse when the `/video/detailSharpening` property changes on the device:

Name	Type	Description
.enabled	boolean	Whether detail sharpening is enabled

/video/detailSharpeningLevel

Get the current detail sharpening level

The value JSON returned via the eventResponse when the `/video/detailSharpeningLevel` property changes on the device:

Name	Type	Description
.level	string	Current detail sharpening level of supported shutter speeds Possible values are: Low, Medium, High.

Ajuda

A forma mais rápida de obter ajuda é acessar as páginas de suporte online da Blackmagic Design e consultar o material mais recente disponível para a sua câmera.

Páginas de Suporte Online da Blackmagic Design

O manual, o software e as notas de suporte mais recentes podem ser encontrados na Central de Suporte Técnico da Blackmagic Design em www.blackmagicdesign.com/br/support.

Entrar em Contato com o Suporte Técnico Blackmagic Design

Caso não encontre a ajuda que precisa no nosso material de suporte, utilize o botão “Enviar email” na página de suporte para nos encaminhar uma solicitação de suporte. Como alternativa, clique no botão “Encontre sua equipe de suporte local” na página de suporte e ligue para a assistência técnica da Blackmagic Design mais próxima.

Verificar a Versão de Software Instalada

Para verificar qual versão do utilitário de software Blackmagic Camera Setup está instalada no seu computador, abra a janela “Sobre o Blackmagic Camera Setup”.

- No macOS, abra o Blackmagic Camera Setup na pasta “Aplicativos”. Selecione “Sobre o Blackmagic Camera Setup” no menu do aplicativo para exibir o número da versão.
- No Windows, abra o Blackmagic Camera Setup no menu “Iniciar” ou na tela inicial. Clique no menu “Ajuda” e selecione “Sobre o Blackmagic Camera Setup” para exibir o número da versão.

Como Obter as Atualizações de Software Mais Recentes

Depois que verificar a versão do software Blackmagic Camera Utility instalada no seu computador, visite a Central de Suporte Técnico da Blackmagic Design em www.blackmagicdesign.com/br/support para conferir as últimas atualizações. Embora seja aconselhável executar as atualizações mais recentes, evite atualizar o software se estiver no meio de um projeto importante.

Informações Regulatórias

Descarte de Resíduos de Equipamentos Elétricos e Eletrônicos na União Europeia

O símbolo no produto indica que este equipamento não pode ser eliminado com outros materiais residuais. Para descartar seus resíduos de equipamento, ele deve ser entregue a um ponto de coleta designado para reciclagem. A coleta e reciclagem separadas desses resíduos, no momento do descarte, ajudam a conservar recursos naturais e garantem que o material seja reciclado de forma a proteger a saúde humana e o meio ambiente. Para mais informações sobre onde descartar seu equipamento para reciclagem, entre em contato com o serviço de reciclagem da sua cidade ou com o revendedor onde o produto foi adquirido.

Este equipamento foi testado e mostrou estar em conformidade com os limites para dispositivos digitais da Classe B, conforme a Parte 15 das normas da FCC. Esses limites foram estabelecidos para fornecer proteção razoável contra interferências nocivas quando o equipamento é instalado em um ambiente residencial. Este equipamento gera, usa e pode irradiar energia de radiofrequência e, se não for instalado ou utilizado de acordo com as instruções, poderá causar interferências nocivas às comunicações de rádio. A operação deste produto em uma área residencial pode causar interferência nociva, nesse caso o usuário será solicitado a corrigir a interferência às suas próprias custas.

A operação deste equipamento está sujeita às duas condições a seguir:

1. Este equipamento ou dispositivo não poderá causar interferência nociva.
2. Este equipamento ou dispositivo deve aceitar qualquer interferência recebida, incluindo interferência que possa causar uma operação indesejada.

R-R-BMD-20250213001

ICES-3 (A)
NMB-3 (A)

Declaração do ISED Canadá

Este dispositivo está em conformidade com as normas canadenses aplicáveis a aparelhos digitais de Classe A.

Qualquer modificação ou uso indevido do dispositivo poderá invalidar a conformidade com tais normas.

A conexão com interfaces HDMI devem ser feitas com cabos HDMI blindados de alta qualidade.

Este equipamento foi testado para conformidade com o uso pretendido em um ambiente comercial. Se o equipamento for usado em um ambiente doméstico, ele poderá causar interferência radioelétrica.

Informações de Segurança

A fonte de alimentação AC para DC 24 V incluída com a URSA Cine deve ser conectada a uma tomada com conexão de aterramento de proteção.

Não há partes reparáveis pelo operador no interior da Blackmagic URSA Cine. Solicite o serviço de manutenção à assistência técnica local da Blackmagic Design.

Ao plugar o conector de entrada de alimentação a uma fonte de bateria DC externa, é necessário usar um cabo de alimentação com limitação de corrente ou proteção por fusível.

O cabo utilizado deve ser marcado VW-1 ou cumprir as normas IEC 60332 ou IEC 60695.

Recomendamos que a fonte DC para esta câmera seja de 24 V. Se a fonte de alimentação DC for de 12 V, o cabo deve ser adequado para uma corrente de 14 A.

Uma placa de bateria B-Mount é fornecida com este produto. Recomendamos o uso de uma bateria de 24 V (28,8 V) ou uma bateria de dupla voltagem.

Para saber a corrente máxima que sua bateria de 12 V (14,4 V) pode fornecer continuamente, consulte o manual ou as marcações da bateria. Sugerimos uma classificação mínima de 12 A.

Declaração do Estado da Califórnia

Este produto pode expô-lo a produtos químicos, tais como vestígios de bifenilos polibromados dentro de peças de plástico, reconhecidos pelo estado da Califórnia como causadores de câncer, defeitos congênitos ou outros danos reprodutivos.

Para mais informações, visite www.P65Warnings.ca.gov.

Escritório na Europa

Blackmagic Design Europe B.V.
Rijnlanderweg 766, Unit D
2132 NM Hoofddorp
NL

Blackmagic URSA Cine

Sob condições de muita luz solar, considere proteger a câmera sob sombras para prevenir a exposição da câmera ou da bateria de lítio à luz direta por longos períodos de tempo. Mantenha as baterias de lítio afastadas de todas as fontes de calor.

O conector de saída “RS” de 24 V é adequado para fornecer alimentação a acessórios de terceiros. Assegure-se de que o consumo de energia seja inferior a 48 W no total.

Garantia

Garantia Limitada

A Blackmagic Design garante que este produto estará livre de defeitos de materiais e fabricação por um período de 12 meses a partir da data de compra. Se o produto apresentar defeito durante este período de garantia, a Blackmagic Design, a seu critério, consertará o produto com defeito sem cobrança pelos componentes e mão de obra, ou providenciará a substituição do produto defeituoso.

Para obter o serviço sob esta garantia você, o Consumidor, deve notificar a Blackmagic Design do defeito antes da expiração do período de garantia e tomar as providências necessárias para a execução do serviço. O Consumidor é responsável pelo empacotamento e envio do produto defeituoso para um centro de assistência designado pela Blackmagic Design com os custos de envio pré-pagos. O Consumidor é responsável pelo pagamento de todos os custos de envio, seguro, taxas, impostos e quaisquer outros custos para os produtos que nos forem devolvidos por qualquer razão.

Esta garantia não se aplica a defeitos, falhas ou danos causados por uso inadequado ou manutenção e cuidado inadequado ou impróprio. A Blackmagic Design não é obrigada a fornecer serviços sob esta garantia: a) para consertar danos causados por tentativas de instalar, consertar ou fornecer assistência técnica ao produto por pessoas que não sejam representantes da Blackmagic Design, b) para consertar danos causados por uso ou conexão imprópria a equipamentos não compatíveis, c) para consertar danos ou falhas causadas pelo uso de componentes ou materiais que não são da Blackmagic Design, d) para fornecer assistência técnica de um produto que foi modificado ou integrado a outros produtos quando o efeito de tal modificação ou integração aumenta o tempo ou a dificuldade da assistência técnica do serviço.

A exposição do URSA Viewfinder à luz solar direta pode danificar o visor do viewfinder, pois a lente atua como uma lente de aumento. A exibição de imagens estáticas ou de alto contraste nos painéis OLED por longos períodos de tempo podem resultar em retenção ou queima de imagem. Para evitar isto, certifique-se de que o sensor IR para a detecção de rostos não esteja coberto intencionalmente e desconecte o visor quando não estiver em uso por longos períodos de tempo. A retenção de imagem não está coberta por esta garantia de produto.

ESTA GARANTIA É FORNECIDA PELA BLACKMAGIC DESIGN NO LUGAR DE QUAISQUER OUTRAS GARANTIAS, EXPLÍCITAS OU IMPLÍCITAS. A BLACKMAGIC DESIGN E SEUS FORNECEDORES ISENTAM-SE DE QUAISQUER GARANTIAS IMPLÍCITAS DE COMERCIABILIDADE OU ADEQUAÇÃO A UMA FINALIDADE ESPECÍFICA. A RESPONSABILIDADE DA BLACKMAGIC DESIGN DE CONSERTAR OU SUBSTITUIR PRODUTOS DEFEITUOSOS DURANTE O PERÍODO DE GARANTIA É A ÚNICA E EXCLUSIVA REPARAÇÃO FORNECIDA AO CONSUMIDOR. A BLACKMAGIC DESIGN NÃO É RESPONSÁVEL POR QUAISQUER DANOS INDIRETOS, ESPECIAIS, ACIDENTAIS OU CONSEQUENTES, INDEPENDENTEMENTE DE A BLACKMAGIC DESIGN OU O FORNECEDOR TEREM SIDO AVISADOS PREVIAMENTE SOBRE A POSSIBILIDADE DE TAIS DANOS. A BLACKMAGIC DESIGN NÃO É RESPONSÁVEL POR QUAISQUER USOS ILEGAIS DO EQUIPAMENTO PELO CONSUMIDOR. A BLACKMAGIC NÃO É RESPONSÁVEL POR QUAISQUER DANOS CAUSADOS PELO USO DESTE PRODUTO. O USUÁRIO DEVE OPERAR ESTE PRODUTO POR CONTA E RISCO PRÓPRIOS.

© Direitos autorais 2025 Blackmagic Design. Todos os direitos reservados. “Blackmagic Design”, “URSA”, “DeckLink”, “HDLink”, “Workgroup Videohub”, “Multibridge Pro”, “Multibridge Extreme”, “Intensity” e “Leading the creative video revolution” são marcas comerciais registradas nos Estados Unidos e em outros países. Todos os outros nomes de empresas e produtos podem ser marcas comerciais de suas respectivas empresas com as quais estão associados.

A marca nominativa e os logotipos Bluetooth são marcas registradas e de propriedade da Bluetooth SIG, Inc., e qualquer uso dessas marcas pela Blackmagic Design é feito sob licença. Outros nomes e marcas comerciais pertencem aos seus respectivos proprietários.

Ekim 2025

Kamera Kılavuzu

Blackmagicdesign

Blackmagic URSA Cine Immersive



İçindekiler

Blackmagic URSA Cine Immersive 1525
“Immersive” nedir? 1525
Apple Immersive Video 1525
URSA Cine Immersive 1526
Başlarken 1527
Kameraya Güç Sağlanması 1527
Kayıt Ayarlarının Yapılması 1528
Kaydın Görüntülenmesi 1530
Görüntüleme Seçenekleri ve Katmanlar 1530
İlk Çekiminizi Kadrajlarken 1531
Kamera Yüksekliği 1531
Kamera Eğimi 1532
Konumlandırma 1533
İzleyici Etkeni 1534
Kamera Hareketi 1534
Veri Hızı Çizelgeleri 1535
Ses ve Zaman Kodu 1536
Apple Spatial Audio Formatı 1536
Ambisonik 1536
Yaka Mikrofonları ve Minyatür Mikrofonlar 1537
El Mikrofonları 1537
Zaman Kodunun Bağlanması 1537
Çekim Yaparken 1539
Pozlama Gereksinimleri 1539
Örtücü Hızı 1539
ISO ve Dinamik Aralık 1540
ND 1541
Farklı Hızda Kayıt 1542
Histogram 1543
LCD Ekran Seçenekleri 1543
Kayıt İşlemi 1544
Oynatım 1545
Kliplerin Aktarılması 1546
Özet 1547
DaVinci Resolve Studio 1548
İş Bilgisayarı Önerileri 1548
Proje Kurulumu 1549
Ana Ayarlar 1549
Video Görüntüleme 1550
Görüntü Ölçeklendirme 1550
Renk Yönetimi 1551
DaVinci Geniş Gam Hakkında 1552
Apple Immersive İşleme 1553
Dosyaların Yüklenmesi 1553
Kurgu 1555
Yerleştirme 1555
Fonlar 1555
Görüntüleyicinin Kullanımı 1557
Geçişler 1559
Apple Immersive Motion Data 1560
Yazılar 1561
Yaratıcı Hususlar 1562
VFX Görüntü Birleştirme 1563
MediaOut düğümleri 1563
Renderer3D Düğümü 1564
Immersive Patcher ile Görüntü Birleştirme 1565
Renk Derecelendirme 1566
Gürültü Azaltma 1566
Birincil Renk Derecelendirme 1567
İkincil Renk Derecelendirme 1567
Boyutlandırma 1568
3D Palet 1568
CST Renk Yönetimi 1570
CST Zaman Çizelgesi Yönlendirme 1570
CST Renk Derecelendirme Monitörü Eşleştirme 1571
CST Apple Vision Pro’da Görüntüleme 1573
Ses 1574
Apple Spatial Audio Format (ASAF) 1574
Teslimat 1575
Geçişleri Atlama 1575
Apple Vision Pro'da İnternet Yayını 1576
Apple Immersive Video Utility Yardımcı Yazılım Hakkında 1576
Developer Information 1577
Camera Control REST API 1577
Livestream Control API 1579
Clips Control API 1583
Media Pool Control API 1584
Monitoring Control API 1587
Event Control API 1596

System Control API 1596
Transport Control API 1602
Timeline Control API 1606
Media Control API 1609
Slate Control API 1612
Preset Control API 1618
Audio Control API 1620
Lens Control API 1625
Video Control API 1629
Camera Control API 1637
Immersive Control API 1640
Color Correction Control API 1641
Notification websocket - 1.0.0 1646
messages 1646
Yardım 1670
Mevzuata İlişkin Bildirimler 1671
Güvenlik Bilgileri 1672
Garanti 1673

Blackmagic URSA Cine Immersive

Blackmagic URSA Cine Immersive kamera satın aldığınız veya kiraladığınız için sizi kutluyoruz.

Blackmagic URSA Cine Immersive, genellikle “Immersive” yani üç boyutlu video olarak bilinen 180 derecelik stereoskopik VR çekimleri yapmak için tasarlanmıştır. Özellikle, şu anda mevcut olan en yüksek kalite üç boyutlu deneyim olan Apple Vision Pro için Apple Immersive Video çekimleri için optimize edilmiştir.

Bu kullanım kılavuzu, Blackmagic URSA Cine Immersive ve bu kamerayla çekim yapma hakkında bir giriş niteliğinde olup, kameranın temel kullanımına ve Apple Immersive Video teslim etmek üzere tasarlanmış benzersiz formatına ağırlık vermektedir. Bu formatı kavramak, kamerayı daha iyi anlamanıza yardımcı olacaktır.

İlerleyen sayfalarda, kamerayı kullanmanın yanı sıra üç boyutlu formatın avantajlarını anlamanıza yardımcı olacak teknik ve yaratıcı tavsiyeler bulacaksınız.

“Immersive” nedir?

“Immersive”, üç boyutlu uzamsal bir video formatıdır. Benzer teknolojilerin bir süredir var olduğunu muhtemelen biliyorsunuzdur. Immersive formatının farkını anlamanız için, diğer teknolojilere genel bir bakış aşağıdadır:

- **Geleneksel 3D Sinemada**, her sahne iki kamera kullanılarak çekilir ve sağ ile sol kamera lensi arasındaki yatay mesafe insan gözleri arasındaki mesafeyi taklit eder, böylece görüntü karesi izleyicide bir derinlik algısı oluşturur. Stereoskopik görüntüler izleyiciye derinlik hissi verse de izleyicinin bakış açısı, görüntü yönetmeninin kompozisyonu tarafından kontrol edilir.
- **Küresel Video**; 3 boyutlu ve 360 derece kameraların ilkelerini birleştirerek, izleyicilerin derinlik hissi yaşarken her yöne bakabilmelerini sağlar. 360 derece görüntüler, görüntü kapsamında izleyicinin baktığı küçük kısmın yeterli netlikte olmasını sağlamak için çok yüksek çözünürlük gerektirir. İnsan fizyolojisi, başımızı sadece omuzdan omuza çevirmemize izin verdiğinden, arkaya bakabilmek için izleyiciler genellikle ayakta durur veya bir döner sandalyede oturur.
- **Hacimsel Video**, izleyicilere sadece derinlik hissi ve etrafı görebilme imkanı sunmakla kalmaz, aynı zamanda hareket etme ve bakış açısını değiştirme imkanı da sunar. Kamera düzenleri hacimsel videoyu kaydedebilirken, çoğu sanal gerçeklik veya artırılmış gerçeklik, dinamik olarak işlenen 3D varlıkları kullanır ve katılımcılar için güvenli bir ortam gerektirir.

Immersive yani üç boyutlu video, geleneksel sinemanın en iyi özelliklerini ve 3D interaktif formatları bir araya getirir. En basit haliyle, yüksek kare hızına ve yüksek çözünürlüğe sahip 180 derecelik stereoskopik bir formattır.

Apple Immersive Video

Blackmagic URSA Cine Immersive, Apple Immersive Video çekmek için özel olarak tasarlanmıştır. Bu format, Apple Vision Pro ile kullanılır ve yüksek çözünürlüklü, stereoskopik 3 boyutlu videoyu 180 derecelik görüş alanında destekler.

Apple Immersive Video aşağıdaki özelliklere sahiptir:

- Saniyede 90 kare oynatım.
- Her bir göz için 4320 x 4320 çözünürlüğe kadar dağıtımı yapılabilir.
- Orijinal kamera görüntüsünü 180 derecelik ortama yerleştirmek için benzersiz bir metaveri sistemi.
- 3 boyutlu ses sağlamak için Apple Spatial Audio Formatı.

Apple Immersive Video, yüksek çözünürlüklü stereoskopik 180 derece video oynatabilen donanım gerektirir. Apple Vision Pro, şu anda bu formatı destekleyen tek cihazdır.



Apple Vision Pro

Bu sanal gerçeklik gözlüğü, 7,5 mikron piksel aralığı olan, çift 3660 x 3200 piksel mikro OLED ekrana sahiptir. Görüntüleri baş ve göz hareketleriyle uyumlu hale getirmek için 13 adet takip kamerası, bir LiDAR tarayıcı, dahili ölçüm cihazları ve göz takip sensörlerine sahiptir.

8 çekirdekli CPU ve 10 çekirdekli GPU'ya sahip bir M2 işlemci, saniyede 100 kareye kadar oynatım hızı sağlar.

URSA Cine Immersive

Blackmagic URSA Cine Immersive, Apple Vision Pro için Apple Immersive Video çekmek üzere tasarlandı.



Blackmagic URSA Cine Immersive

URSA Cine Immersive, iki adet eşleşen 58 megapiksel RGBW sensör kullanır. Bu sensörler hem yatay hem de dikey olarak 180 derecelik tam görüş alanını yakalamak için neredeyse kare şeklindedir. Her bir sensör, 210 derecelik balık gözü lens ile eşleştirilmiştir. Lens merkezleri arasındaki mesafe 64 mm'dir.

Dahili ND filtreler, 8 durak ışık kontrolü sağlar. Lensin diyafram açıklığı, yaklaşık f4.5'e sabitlenmiştir.

Kamera, saniyede 90 kare hızında iki adet 8K görüntü akışı kaydeder. Böylelikle, standart 4K 24p kayda kıyasla, önemli ölçüde daha yüksek veri hızları sağlanır.

Başlarken

Blackmagic URSA Cine'yi kullanmaya başlamak için kameranıza güç sağlayıp çalıştırmanız yeterlidir. Kameranın modül yuvasına formatlanmış bir Blackmagic Media Module 8TB takılıdır ve hemen klip kaydetmeye hazırdır!

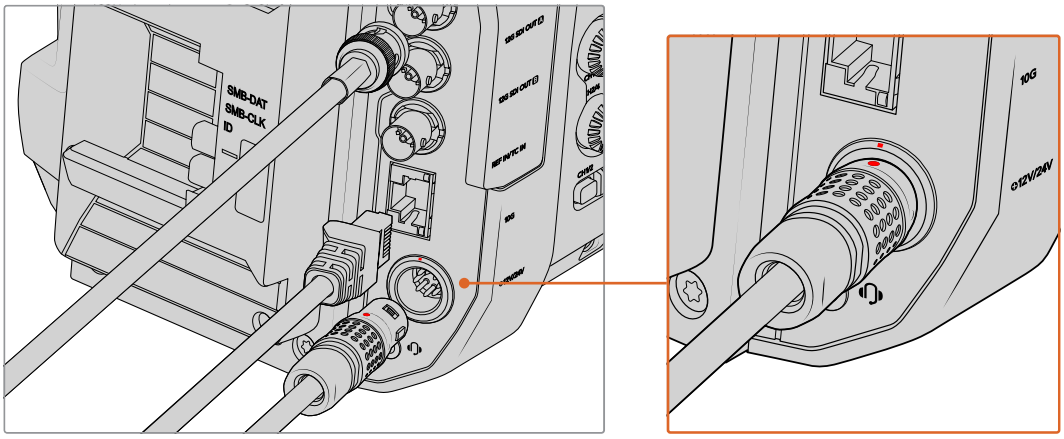
Kameraya Güç Sağlanması

Kameranıza güç sağlamanın en hızlı yolu, size sağlanan güç adaptörünü kullanarak, harici bir güç kaynağına bağlamaktır.

Harici güç kaynağı bağlamak için:

1. Önce lastik koruyucu toz kapağını çıkarın ve güvenli bir yere koyun. AC - 24V DC adaptörün fişini standart bir IEC C13 şebeke güç kablosu kullanarak elektrik prizine bağlayın.
2. AC to 12V DC adaptörünün 8 pimli konektörünü, kameranın arka panelindeki 24V DC güç girişine takın.

Hem harici güç hem de bataryayı bağladığınızda, kamera harici gücü kullanır. Şarjı dolu bir batarya takılıyken harici güçten fişin çıkarılması, kameranızı kesintisiz bir şekilde batarya gücüne geçirir.



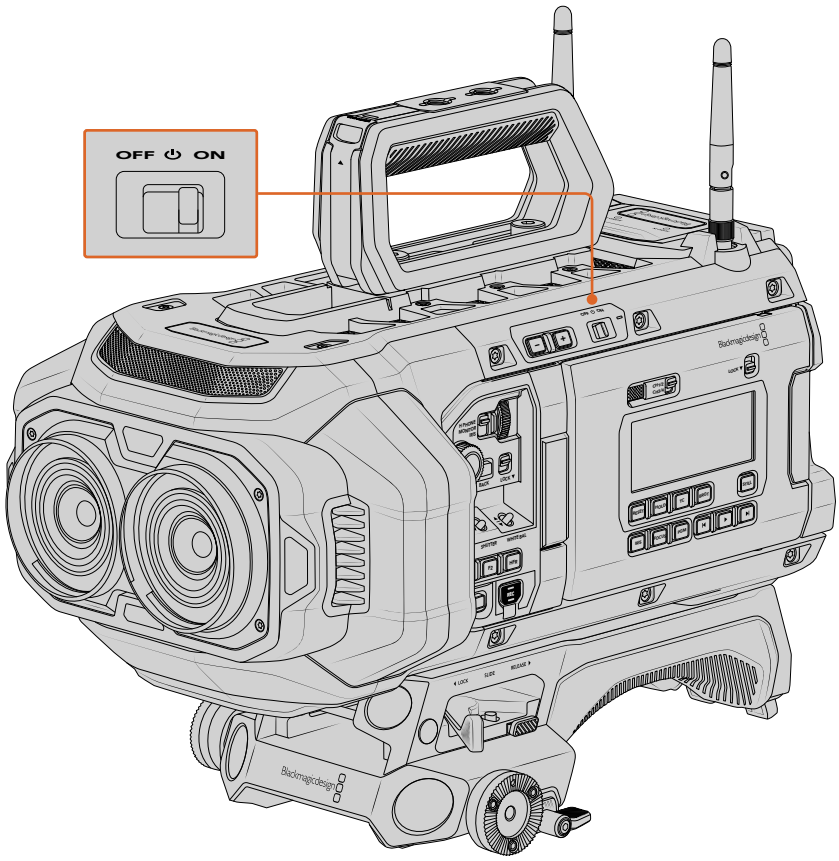
Gücü takmak için kırmızı noktaları hizalayın ve yerine kilitlendiğini hissedene kadar konektörü yavaşça içeri itin. Çıkarmak için konektörün metal kılıfını tutun ve kameradan dışa doğru yavaşça çekin. Bu işlem kılıfı geri çekerek konektörü soketten ayırır.

Kameranıza dahil edilen batarya plakası, 24 volt B yuvalı bataryalar için tasarlanmış bir B yuva plakadır. 24 ila 34 voltaj aralığında çalışmak URSA Cine Immersive ile kullanım için idealdir, çünkü daha yüksek voltaj termal kayıpları azaltarak, yüksek kare hızında kayıt yapılmasını ve kameranızdan birden fazla aksesuara güç verilmesini sağlar.

24V çalışma ile uyumlu isteğe bağlı VLock ve Gold Mount plakalar ve 26V üçüncü parti bataryalar için Gold Mount Plus plakaları da mevcuttur.

NOT URSA Cine Immersive'i 12 ila 18 voltluk güç kaynakları veya bataryalarla çalıştırırken, 3 pimli RS konektör veya 2 pimli lemo tipi konektör üzerinden aksesuarlara güç sağlamak mümkün değildir.

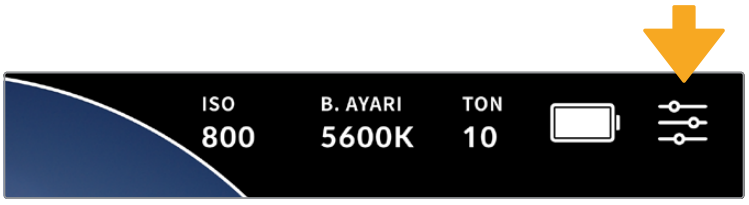
Bir batarya taktıktan veya harici gücü bağladıktan sonra URSA Cine Immersive'i çalıştırın. Kameranın üstündeki açma/kapatma düğmesini, “on” konumuna getirin.



Kamerayı çalıştırmak için, güç düğmesini “on” konumuna getirin.

Kayıt Ayarlarının Yapılması

Bu bölüm, üç boyutlu içerik çekerken benzersiz veya önemli olan belirli kamera ayarları hakkında bilgi sağlar. Kameranın ayarlar menüsünü açmak için, katlanır ekran veya dokunmatik asistan ekranının sağ üst köşesindeki menü ayarları simgesine dokunun.



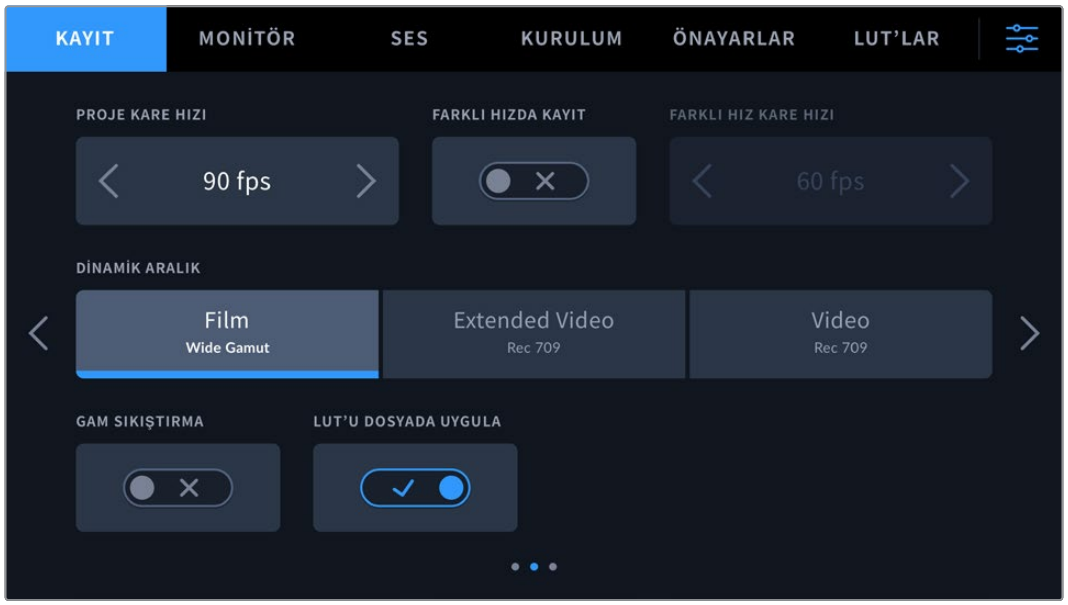
Kayıt Ayarları 1

Bu sayfadaki “kodek ve kalite” seçenekleri, üzerinde çalıştığınız projeye uygun Blackmagic RAW kodek ayarını seçmenizi sağlar. Blackmagic RAW seçenekleri, dört adet “sabit bit oranı” ve “sabit kalite” ayarı arasında bölünmüştür. Apple Vision Pro için 90 fps içerik kaydederken, 12:1 veya Q3 sıkıştırma ayarlarını kullanmanızı şiddetle tavsiye ederiz. Bu ayarlar, mevcut bilgisayar donanımının kaldırabileceği veri hızı ve dosya boyutlarını korurken, aynı zamanda son derece yüksek görüntü kalitesini de muhafaza etmek için ideal bir denge sağlar.



Kayıt Ayarları 2

Kameranızın “kayıt” ayarlarının ikinci sayfasında kare hızı ayarları bulunur.



Kameranızın proje kare hızı, Apple Vision Pro'nun ekran kare hızına uyacak şekilde saniyede 90 kareye ayarlanması gerekir.

Başka sistemler için çekim yapıyorsanız, 60 fps veya 50 fps gibi daha düşük kare hızları yeterli olabilir. 24 fps veya 25 fps gibi daha düşük geleneksel film tarzı çekim kare hızları, 24p sinema filmi içeriği için görsel efekt şablonları çektiğinizde uygun olabilir.

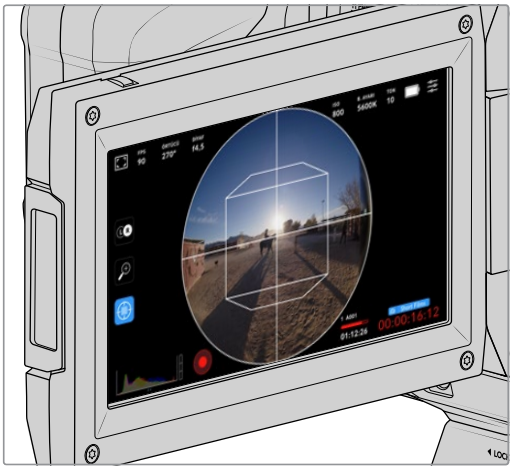
URSA Cine Immersive'inizi kullanmaya başlamak için yapmanız gerekenler bu kadar! Kameranızdaki önceden formatlanmış olan medya modülü sayesinde, hemen kayda başlayabilirsiniz.

Kaydın Görüntülenmesi

URSA Cine Immersive, çekim sırasında görüntüyü denetlemeniz için çeşitli seçenekler sunar.

Katlanır dokunmatik ekran, kameraman tarafından kullanılmak üzere tasarlanmıştır. Bu ekran, kameranın fiziksel boyutunu küçültmek için ters çevirilse bile görünür kalır, veya kendinizi çekerken çekimi görebilmeniz için öne bakacak şekilde konumlandırılabilir.

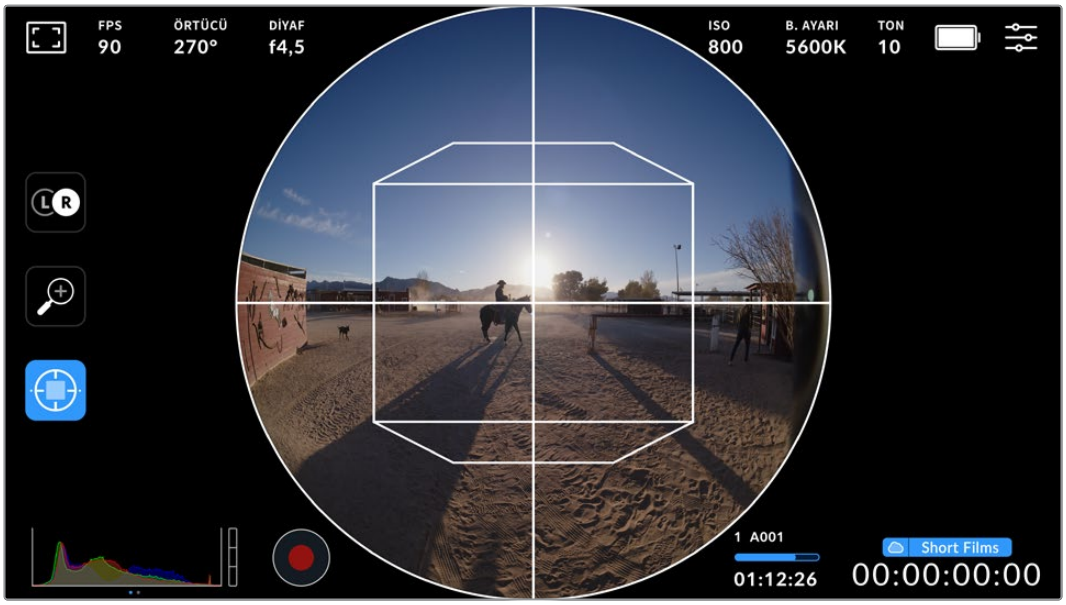
Dokunmatik asistan ekranı, kameramanın görüşünü engellemeden başka bir ekip üyesinin görüntüyü denetlemesini sağlar. Kameranın görüş alanının son derece geniş olması nedeniyle, dokunmatik asistan ekranını kullanırken dikkatli olunmalıdır.





Görüntüleme Seçenekleri ve Katmanlar

Kamera, lens alanını gösterir ve her bir göz teker teker görüntülenebilir. Katlanır ve dokunmatik asistan ekranları, 180 derecelik bir görüş açısı sunmak için maskelenmiştir. SDI çıkışları, her sensör tarafından yakalanan 210 derecelik görüntünün tamamını gösterir.



Tüm monitör çıkışları aynı gözü gösterir. Her iki dokunmatik ekranda bulunan “L R” etiketli göz değiştirme butonuna dokunarak sol ve sağ göz arasında geçiş yapabilirsiniz.

Dokunmatik ekranlarda, katman butonuna basarak isteğe bağlı bir katman etkinleştirilebilir. Bu katman aşağıdakileri içerir:

- 180 derecelik bir dış sınır
- Kadrajın merkezini gösteren bir artı işareti
- Apple Vision Pro'da düz ileri baktığınızda görülen tahmini görüş alanını temsil eden çember içindeki bir kılavuz.

Bu araçlar, üç boyutlu çekim sırasında doğru kadrajlama ve konumlandırma işlemlerine yardımcı olmak için tasarlanmıştır.

İlk Çekiminizi Kadrajlarken

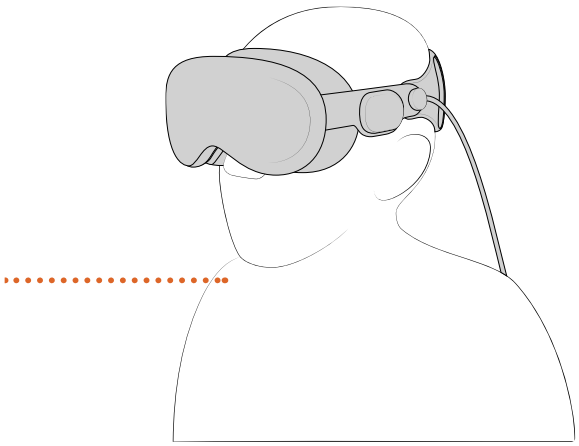
Üç boyutlu ve geleneksel 2 boyutlu (2B) içerikler birçok ortak özelliğe sahip olsa da aralarında belirgin farklılıklar da vardır.

Örneğin, arabada seyahat ederken telefonunuza çok fazla baktığınızdan dolayı araba sizi hiç tuttu mu? Bu, beyniniz çelişkili sinyaller aldığında ortaya çıkan duyusal uyumsuzluktan kaynaklanır. İç kulağınız beyninize vücudunuzun hareket ettiğini söyleyen bir sinyal gönderirken, gözleriniz ise beyne sabit olduğunuzu söyleyen bir sinyal gönderir.

Üç boyutlu format ile benzer bir deneyim yaratmak çok kolaydır. Örneğin, kamerayı döndürdüğünüzde, izleyiciler gözleriyle hareket ettiklerini algılar, ancak boyunları ve iç kulaklarıyla da sabit durduklarını algılarlar. Sonuç olarak, izleyicileriniz için kamera hareketlerini akıcı hale getirmeyi öğrenmeden önce, sabit kurulum çekimlerinde ustalaşmak en iyisidir.

Kamera Yüksekliği

Geleneksel 2B sinemada, bir sahnedeki güç dengesini göstermek için kamerayı yüksek veya alçak bir konuma yerleştirebilirsiniz. Üç boyutlu çekim yaparken, kamerayı yere yerleştirmenin etkisini değerlendirin. Bu, izleyiciyi nasıl etkileyebilir? İzleyiciye boğazına kadar gömülmüş hissi verebilir.



Zeminin algılanan yüksekliğinin, izleyicilerin izleme deneyimi üzerinde bir etkisi olabilir. Alçaktaki bir kamera, izleyici için bir anlam ifade etmeyen bir zemin yüksekliği algısı verir!



Üç boyutlu çekimler yaparken, alçak açılı çekimler izleyicinin rahatsız hissetmesine neden olabilir.

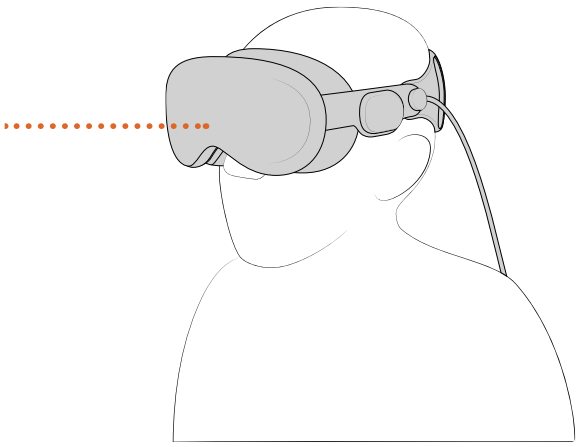
İzleyicinin fiziksel olarak görüntünün içinde olduğunu hissetmesini sağlamak için, kamera yüksekliğini ve konumunu, izleyicinin sahnede ayakta veya oturuyormuş gibi hissedeceği bir şekilde seçmek, genellikle en iyisidir. Yaratıcılığınızı ifade etmeniz için hala yer vardır. Çekilen sahnede insanlar olduğunda, biraz daha yüksekteki bir kamera, başkalarına yukarıdan bakma hissi yaratarak izleyicinin güçlü hissetmesini sağlayabilir. Bunun tersi olararak, daha alçaktaki bir kamera, diğer insanlara bakmak için izleyiciyi başını kaldırmaya zorlar.

Bu, izleyicileri rahatsız etmeden daha yaratıcı şekilde kamera yerleştirmenin imkansız olduğu anlamına gelmez. Uçan kameralar, üstten aşağıya bakışlar, alçak açılı çekimler ve diğer birçok teknik, üç boyutlu yapımlarda başarıyla kullanılmaktadır. Doğru kurgu, hareket ve bağlamla birleştirildiğinde, bu teknikler izleyicilerinizi eğlendirmek için kullanılabilir.

Kamera Eğimi

Geleneksel 2B içerik çekerken ve kamerayı yüksek veya alçak bir konuma yerleştirirken, görüntü yönetmeni kompozisyonu korumak için genellikle kamera eğimini kullanır. Bunun, üç boyutlu görüntüyü nasıl etkileyeceğini düşünün.

Kamerayı aşağı doğru eğerseniz, izleyicileriniz ufku görebilmek için başlarını yukarı doğru eğmeleri gerekir. Bu durumun, harekete bağlı bulantıya sebebiyet verme olasılığı düşüktür, ancak gerçek hayatta ufuk her zaman tam önümüzde olduğundan, izleyicideki görüntünün içinde olma hissini bozabilir.



Üç boyutlu çekimler, ufuk çizgisi sıfır derecede, yani tam karşıda olduğunda en mantıklı şekilde görünür.



URSA Cine Immersive'yi eğdiğinizde, izleyicilerin ufku görmek için bakmaları gereken yer ile normalde ufku görmek için baktıkları yer arasında bir kopukluk oluşur.

Bu nedenle, çoğu üç boyutlu görüntü sıfır derece veya sıfıra yakın bir eğimle çekilir. Kamera yatay ve dikey olarak 180 dereceden fazla görüş açısını yakaladığından, izleyiciler yukarı ve aşağı bakmaya devam edebilir. Film yapımcısının izleyiciyi belirli bir yöne bakmaya zorlaması yerine, kontrol izleyicidedir.

İzleyicilerin görüntünüzün farklı kısımlarına bakmasını sağlamak için ışıklandırma, diyafram açıklığı ile kadrajlama ve yönlendirici çizgiler gibi diğer teknikleri kullanabilirsiniz.

Konumlandırma

Üç boyutlu çekimler genellikle otururken izlenir. Bu, artırılmış gerçeklik veya sanal gerçeklik formatı olmadığı için izleyiciler ayakta durmaz veya hareket etmezler. Şu anda oturuyor musunuz? Deney olarak, başınızı bir yandan diğer yana çevirin ve yukarı aşağı hareket ettirin. Vücudunuzu hareket ettirmeden, başınızı rahatça ne kadar çevirebilirsiniz? Üç boyutlu formatın görüş alanı, çoğu insanın kendini rahatlıkla döndürebildiği aralığı aşar. Sola veya sağa döndüğünüzde bu format hala çevresel görüş sağladığından, bu iyi bir şeydir. Bu deney, görüş alanı içinde ana aksiyonun nerede konumlandırılması gerektiğini anlamaya başlamanıza yardımcı olur.

Genellikle baş hareket aralığı şöyledir:

Dönüş 120 ila 160 derece Eğme ve Uzatma: 110 ila 130 derece

Başın dönüşü göz hareketiyle birleştirildiğinde, kendi omzunuzun üzerinden bakmak mümkün olsa da çoğu insan için bu pozisyonda uzun süre durmak rahatsız edicidir.

Kompozisyonu kolaylaştırmak için kameranızın görüntü bindirme özelliğini etkinleştirin. Aksiyonu genel görüş alanını gösteren içteki işaretlerin dahilinde tutmak, içeriğinizin görünürlüğünü artırır.

Kadrajdaki ilgi alanlarının hareketi de dikkate alınmalıdır. Bir kişiyi en soldan en sağa doğru yürürken takip etmek, aşırı baş hareketi gerektirir. Benzer şekilde, ilgi noktasının görüntünün kenarında olduğu bir çekimden, ilgi noktasının karşı kenarda olduğu bir çekime geçmek için başın aşırı hareket ettirilmesi gerekir. Kompozisyonun dikkatlice düşünülmesi, yaratıcılığınızı ifade etmenizi sağlarken izleyicilerin rahat izlemesini de mümkün kılar.

İzleyici Etkeni

Geleneksel 2B içerik çekerken; kadraj, lens seçimi ve netlik ayarlarını kullanarak izleyicinin dikkatini yönlendirebilirsiniz. Üç boyutlu çekim yaparken, 180 derecelik görüş açısı, görüntüden bazı bilgileri çıkararak izleyicinin dikkatini yönlendirmeyi zorlaştırır. Üç boyutlu çekim, izleyicilere görüntülerinizi keşfetmeleri için çok daha fazla seçim hakkı sunar. Üç boyutlu içeriği bu kadar ilgi çekici kılan faktörlerden biri de budur. Yönsel ses, ışıklandırma, bloklama ve kompozisyonun dikkatli kullanımıyla, izleyicilerin dikkatini yönlendirmek yine de mümkündür.

Kamera Hareketi

Üç boyutlu bir kameranın sabit kalması zorunlu değildir, ancak bazı kamera hareketleri doğal olmayan veya rahatsız edici bir his verebileceğinden, dikkatli olunmalıdır. Üç boyutlu kamera hareketini nasıl rahat hale getirebileceğimizi anlamaya yardımcı olmak için, araba iyi bir benzetmedir. Büyük bir aracın, hız veya yön değişikliklerine direnç gösteren kütlesi ve ataleti vardır. Bu nedenle, bu tür bir araç zamanla hızlanıp yavaşlar ve öngörülebilir ve kontrollü bir şekilde hareket eder. Sabit kurulumlu çekimleri öğrendikten sonra üç boyutlu hareketli çekimlere geçerseniz, aşağıdaki unsurları göz önünde bulundurarak keyifle izlenen hareketler yaratabilirsiniz:

- **Süre:** İzleyicilerin kameranın hareketine alışması ve keyfini çıkarması için zaman tanıyın. Kısa süreli hareketlerin, izleyicide yön duygusunu kaybetmeye neden olma olasılığı daha yüksektir.
- **Yoğunluk:** Yavaş tempolu hareketlerin, üç boyutlu bir kurguya entegre edilmesi daha kolaydır. Yüksek tempolu hareketlerin olması mümkündür, bu hareketlerden önce ve sonra bir alışma süresi eklemek, izleyicilerin rahat etmesine yardımcı olabilir.
- **Görsel Referans Noktaları:** Bunlar, sahnedeki hareketsiz kısımlardır. Görsel referans noktaları olmayan bir çekim, izleyicinin kafasını karıştırabilir. Arabanın içi gibi ön plandaki ögeler veya ufuk çizgisi gibi uzaktaki nesneler, hareketli çekimlerinizi sabitlemeye yardımcı olur.

Veri Hızı Çizelgeleri

Aşağıdaki çizelgeler, saniyede 90 kare ve saniyede 24 kare hızında 8K x 7K üç boyutlu içerik için veri hızlarını gösterir.

Blackmagic RAW Sabit Bit Oranı ile 90fps'de Azami Veri Hızları

8K x 7K Immersive			
Kodek	**Piksel Ebatları**	**Kare Hızı**	**Veri Hızı**
Blackmagic RAW 5:1	8,160 x 7,200 Stereo	90	3,203 MB/sn
Blackmagic RAW 8:1	8,160 x 7,200 Stereo	90	2,002 MB/sn
Blackmagic RAW 12:1	8,160 x 7,200 Stereo	90	1,335 MB/sn
Blackmagic RAW 18:1	8,160 x 7,200 Stereo	90	889 MB/sn

Blackmagic RAW Sabit Bit Oranı ile 24fps'de Azami Veri Hızları

8K x 7K Immersive			
Kodek	**Piksel Ebatları**	**Kare Hızı**	**Veri Hızı**
Blackmagic RAW 5:1	8,160 x 7,200 Stereo	24	854 MB/sn
Blackmagic RAW 8:1	8,160 x 7,200 Stereo	24	534 MB/sn
Blackmagic RAW 12:1	8,160 x 7,200 Stereo	24	356 MB/sn
Blackmagic RAW 18:1	8,160 x 7,200 Stereo	24	237 MB/sn

Ses ve Zaman Kodu

Ses, üç boyutlu deneyimin önemli bir parçasıdır. URSA Cine Immersive'i özel ses kayıt ekipmanıyla birlikte kullanmanız önemlidir.

Apple Spatial Audio Formatı

Apple Spatial Audio Format (ASAF), Apple Immersive Video izlerken 3 boyutlu ses sağlamak için kullanılır. ASAF, daha geniş Apple Positional Audio Framework'ün bir parçasıdır ve üç farklı türde ses kaydı içerebilir.

- **Kanal tabanlı:** ASAF dosyanıza geleneksel surround miksaj ekleyebilirsiniz. Bu, izleyicilerin başlarını sabit tuttuğu anlatımlar, müzik ve benzeri durumlar için idealdir.
- **Nesne tabanlı:** Sesler, ayrı hoparlör kanallarından bağımsız olarak 3D alandaki konumlarını tanımlayan metaverilerle nesneler olarak tek tek ele alınabilir. Bunlar genellikle mono kayıtlardır. 3D koordinatları, sahne içinde hareket eden nesneleri takip edecek şekilde hareketlendirilebilir. Bu, sahnede görünen nesnelere bağlı olan diegetik sesler için idealdir.
- **Ambisonik:** Küresel harmonikler kullanarak sesi tüm yönlere aktarabilen tam küresel surround ses formatı. Çevresel sesler için idealdir.

Bu bölümün geri kalanında, etkili bir ASAF dosyası oluşturmak için ihtiyacınız olan ekipman hakkında genel bilgiler verilmektedir.

Ambisonik

Bir Ambisonik mikrofon, hassas doğrulukta bir geometrik düzen içinde yerleştirilmiş dört veya daha fazla kapsül içerir. Her bir kapsül, sesin geldiği yöne bağlı olarak farklı varış zamanları ve yoğunluklarla aynı sesleri yakalar. Bu kapsüllerin yakaladığı sesler, daha sonra bilgisayar tarafından işlenmek üzere ayrı kanallara kaydedilir. Yazılım, seslerdeki farklılıkları karşılaştırarak kaydedilen seslerin yönünü ve niteliğini tahmin edebilir.



Birinci derece bir ambisonik mikrofon

Ambisonik Mikrofonlar

Ambisonik mikrofonlar, içerdikleri kapsül sayısına göre sınıflandırılır. Daha fazla kapsül, mikrofonun uzamsal ayrıntıları daha doğru bir şekilde yeniden üretebilir demektir.

Birinci derece:	4 kanal
İkinci derece:	9 kanal
Üçüncü derece:	16 kanal
Dördüncü derece:	25 kanal ve üzeri

Mikrofon kameranın üzerinde veya yanında olmasına bakmaksızın, ambisonik mikrofonların kameranın çekim yönüyle doğru şekilde hizalanması çok önemlidir. Ayrıca, ambisonik mikrofonlar; ters dönük, öne dönük, yukarı dönük veya aşağı dönük gibi farklı düzenlerde kullanılabildiğinden, ses teknisyenine kullandığınız düzeni söylediğinizden emin olmanız önemlidir. Bu bilgileri, ses kayıt cihazındaki metaveriler aracılığıyla veya notlarınızı paylaşarak iletilebilirsiniz.

Ambisonik kayıtlar, ortam sesi katmanı olarak kullanılabilir. Diğer kanal ve nesne tabanlı ses efektleri, bu katmanın üstüne yerleştirilebilir. Kamera ile doğru şekilde hizalandığında, ambisonik kayıtlar ses mühendisleri için ses tasarımı eklerken referans görevi de görebilir.

Yaka Mikrofonları ve Minyatür Mikrofonlar

180 derecelik bir kamera ile çekim yaparken, sahnede görüneceği için bir boom sopası kullanmak imkansızdır! Bu nedenle, diegetik sesleri ve diyalogları kaydetmek için gizli mikrofonlar ve yaka mikrofonları kullanmanız gerekir.

Örneğin, şunları yapabilirsiniz:

- Bu ses kaynaklarını kablosuz vericiler ve alıcılar kullanarak URSA Cine Immersive'in iki XLR girişine doğrudan bağlayabirsiniz.
- Ses kaynaklarını, harici bir ses kaydediciye bağlayabilirsiniz. Bu kaydedici, büyük çok kanallı bir cihaz veya vücuda takılan minyatür bir cihaz olabilir.

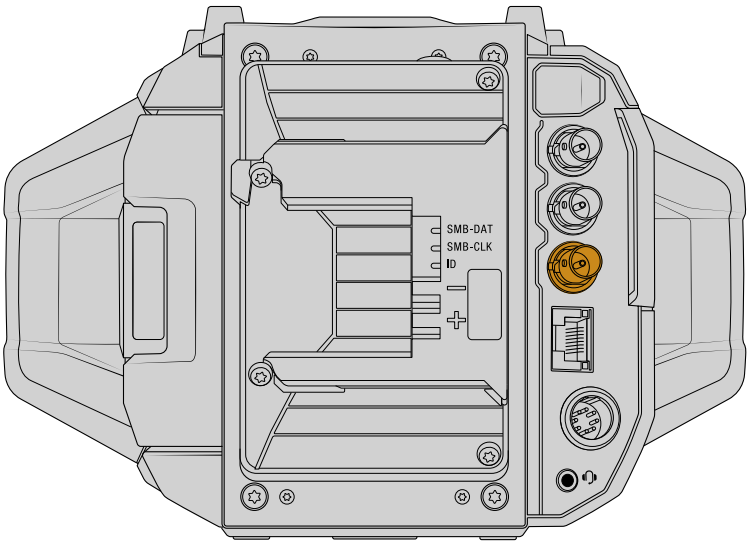
Bu kayıtlar daha sonra senkronize edilip, kanal veya nesne tabanlı sesler olarak miksajı yapılabilir.

El Mikrofonları

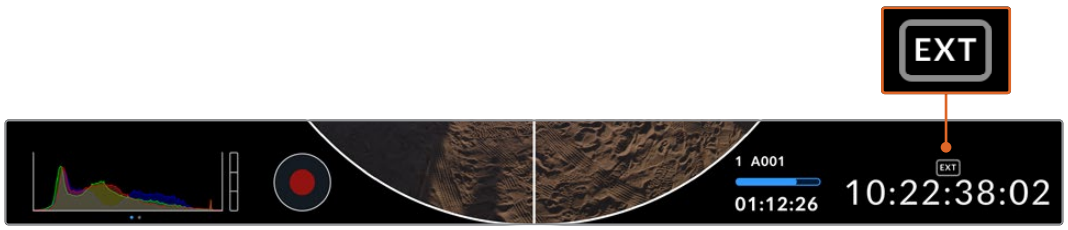
Kamera çekimi bittikten sonra çekim yerinde ek ses efektleri kaydetmek, post prodüksiyon sürecinde zaman kazanmanın yanı sıra daha gerçekçi bir ses elde etmenizi de sağlayabilir. Prodüksiyon sırasında el tipi hiperkardiyot veya benzeri bir mikrofon taşımak çok faydalıdır, çünkü aksiyon tekrarlanırken mikrofonun mümkün olduğunca yakın bir konuma yerleştirilmesini sağlar. Bu kayıtlar daha sonra düzenlenip, kanal veya nesne tabanlı sesler olarak son miksaja eklenebilir.

Zaman Kodunun Bağlanması

URSA Cine Immersive'inize harici zaman kodu bağlamak için, bir BNC kablosu kullanarak zaman kodu üretecini kameranın arka panelindeki zaman kodu girişine bağlayın. Çift sistemle çekim yaparken senkronize zaman koduna bağlanmak, post prodüksiyon sırasında ses ve görüntünün kolayca senkronize edilmesini sağlar.



Bu konektöre aynı kare hızına sahip harici zaman kodu takıldığında, kamera otomatik olarak gelen zaman koduna kilitlenir. Harici zaman kodu kilitli olduğunda URSA Cine Immersive'in LCD ekranındaki zaman kodu sayacının yanında, "EXT" ibaresi görünür. Kabloyu çıkardığınızda, zaman kodu takılmış kalır ve zaman kodunun artık kameranızın dahili zaman kodu saatinden çalıştığını göstermek için, "EXT" ibaresi "INT" olarak değişir.



Kamera zaman kodu, bir zaman kodu üretecine bağlandığında, "ext" ibaresini gösterir.

NOT SMPTE zaman kodu standartları, 23,976, 24, 30 vb. gibi tipik kare hızları için mevcuttur. Saniyede 90 kare için SMPTE zaman kodu standardı yoktur. Bu nedenle, URSA Cine Immersive'in Proje Kare Hızı saniyede 90 kareye ayarlandığında, kamera aslında 30 fps zaman kodunu kullanır. SDI çıkışları da 30p'de çalışır. Tüm zaman kodu aksesuarları veya harici ses kayıt cihazları 30 fps olarak yapılandırılmalıdır.

Çekim Yaparken

Blackmagic URSA Cine Immersive kameranızı nasıl kullanacağınız ve Immersive formatından en iyi şekilde nasıl yararlanacağınız konusunda temel bilgileri edindiğinize göre, bu bölümde kameranın çalışması daha ayrıntılı olarak ele alınacaktır.

Pozlama Gereksinimleri

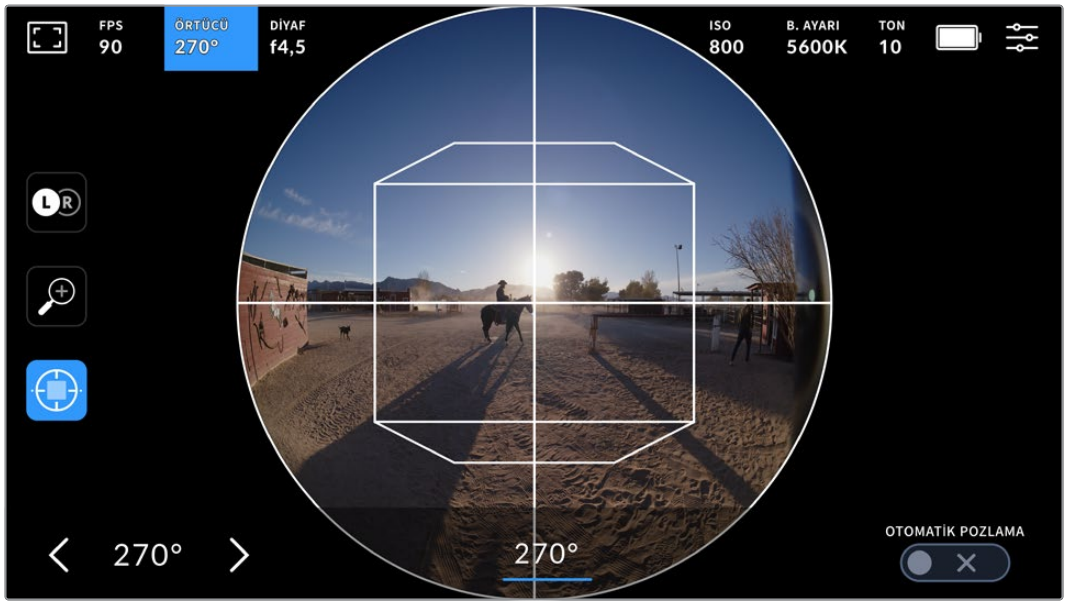
Apple Immersive Video, geleneksel 24 fps çekimden daha yüksek pozlama gereksinimlerine sahiptir ve çekim ve aydınlatma yaklaşımınızı yeniden düşünmeniz gerekebilir. 180 derecelik bir örtücü kullanıyorsanız, 90 fps çekim için aynı miktarda pozlama elde etmek için neredeyse dört kat daha fazla ışık gerekir. Ayrıca, f4.5 civarında olan lenslerin sabit diyaframı, tipik süper hızlı lenslerden üç durak daha karanlıktır. Yani, yeterli pozlama elde etmek için yedi durak daha fazla ışığa ihtiyacınız olabilir. Ancak, 90 fps'de çekim yapıyorsanız, 270° ile 360° arasında geniş bir örtücü açısı kullanmak, yüksek kare hızından kaynaklanan ışık kaybını telafi etmeye yardımcı olurken, aynı zamanda hoş bir hareket bulanıklığı elde etmenizi sağlar.

Örtücü Hızı

"Örtücü" göstergesi, örtücü açısını veya örtücü hızını görüntüler. Kurulum menüsündeki örtücü ölçümü ayarı; örtücü bilgilerinin "örtücü açısı" veya "örtücü hızı" olarak görüntülenmesini sağlayacak şekilde ayarlanabilir.

Örtücü hızını değiştirmek için:

1. Üst araç çubuğundaki "örtücü" simgesine dokunun.
2. Sol veya sağ ok tuşlarını kullanarak yaygın örtücü hızı ön ayarları arasında gezinin. Ekranın sol alt köşesindeki mevcut örtücü göstergesini çift tıklayarak, belli bir örtücü değerini seçebilirsiniz. Bu, 11.2 ile 360 derece arasında iki ondalık basamağa kadar herhangi bir örtücü değerini girmenize olanak tanıyan bir klavye açar.



Ekran klavyesi kullanılarak özel kare hızları girilebilir.

Yüksek kare hızına sahip üç boyutlu içeriğin görünümünde, örtücü hızının etkisi daha azdır. 24 veya 30p çekimlerde hala birçok kişi tarafından 180 derecelik örtücü tercih edilse de üç boyutlu çekimlerde pozlamayı desteklemek için örtücü hızının kullanılması daha yaygındır.

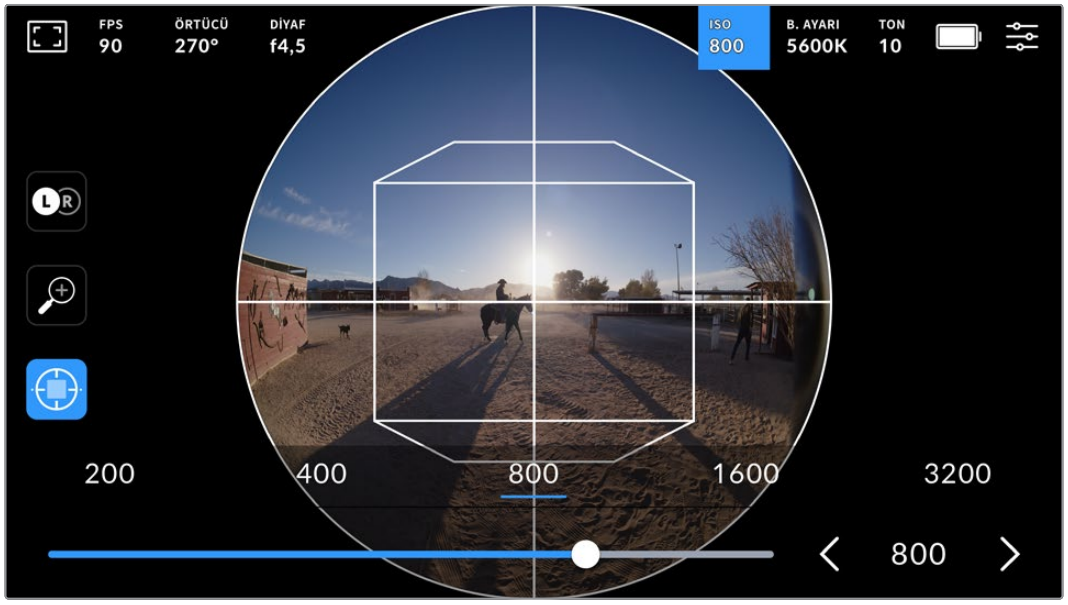
Düşük ışıkta çekim yaparken, örtücü 270 dereceye veya bazı istisnai durumlarda 360 dereceye kadar artırılabilir.

Parlak ışık koşullarında çekim yaparken, örtücü hızı 11,2 dereceye kadar düşürülebilir. Ancak, parlak ışık koşullarında, hareket için örtücü hızını tercih ettiğiniz ayarda bırakıp, ND filtreleri kullanarak pozlamayı kontrol etmek daha yaygındır.

Apple Immersive Video'nun saniyede 90 kare olan yüksek kare hızının, kırpışan ışıklar altında çekim yaparken sorunlara neden olabileceğini belirtmekte fayda var. 60 Hz ortamlarda çalışırken 270 derecelik bir örtücü, 50 Hz ortamlarda çalışırken ise 324 derecelik bir örtücü önerilir.

ISO ve Dinamik Aralık

“ISO” göstergesi, kameranızın geçerli ISO ayarını veya ışık hassasiyetini görüntüler. Bu göstergeyi tıklamak, değişen ışıklandırma ortamlarına uyacak şekilde ISO ayarını yapmanızı sağlar. URSA Cine Immersive'deki ISO aralığı ISO 200 ila 3200’dür. İdeal ISO değeri 800'dür.



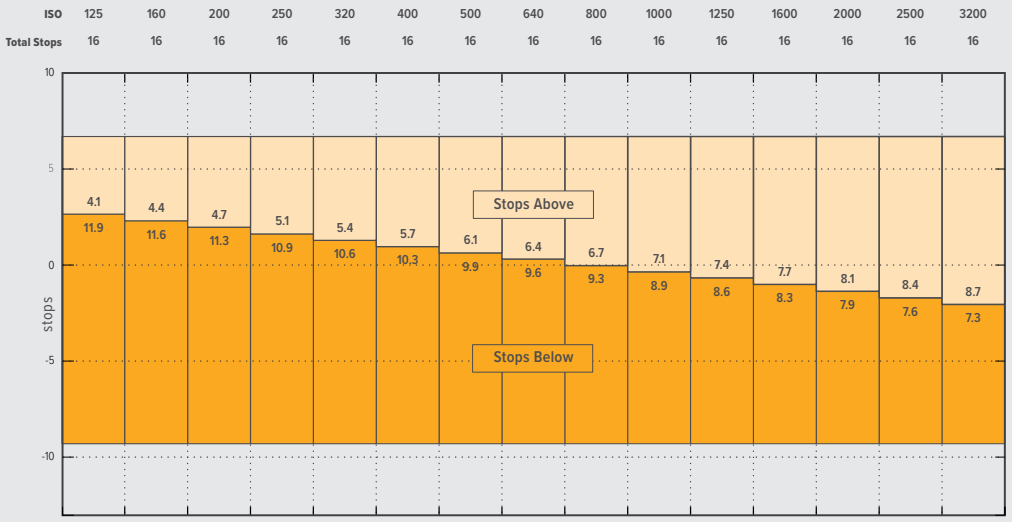
ISO göstergesine dokunarak ISO ayarı yapılabilir.

Sayısal önayarlar, sayının üzerine tıklayarak tam duraklı artışlarla bir ISO ayarı seçimi sağlar. Önayarların altındaki sürgü, bu önayarlar arasında, ISO ayarınızı 1/3 duraklık kademelerle ayarlamanızı sağlar.

Duruma bağlı olarak daha düşük veya daha yüksek bir ISO ayarı seçebilirsiniz. Örneğin; loş ışıklı ortamlarda çekim yaparken, ISO 1600 uygun olabilir fakat belirgin bir gürültü oluşturabilir. Aydınlık ortamlarda, ISO 200 daha canlı renkler sağlayabilir.

Blackmagic kameralar geniş dinamik aralıkta kayıt yapabilir, böylece görüntü kaydederken ve DaVinci Resolve'de renk derecelendirme yaparken çok fazla esneklik sağlar. Aşağıdaki tablo, URSA Cine Immersive'deki kullanılabilir dinamik aralığı gösterir.

URSA Cine Immersive Dinamik Aralık



ND

Bu kameranın ND filtreleri, genellikle pozlamayı kontrol etmenin en yaygın yoludur, kameranızın sensörüne ulaşan ışık miktarını, önayarlı bir pozlama “durağı” sayısı kadar azaltmanızı sağlar.



URSA Cine Immersive'nin ND filtreleri, katlanabilir dokunmatik ekranın sol tarafındaki butonlarla ayarlanır. Saydam filtre ile birlikte mevcut ayarlar; “2”, “4”, “6” ve “8” duraktır.

Mevcut ND ayarı, dokunmatik ekranın sol üst köşesindeki göstergede görüntülenir. ND devredeyken, bu gösterge sürekli açık kalır. ND saydam olarak ayarlandığında, gösterge 4 saniye sonra kaybolur.



NOT ND filtre göstergesi terminolojisini, alıştığınız düzenleri yansıtacak şekilde değiştirebilirsiniz. ND sayısı, duraklar ve kesirler arasında seçim yapmak için kurulum menüsünü kullanın.

Farklı Hızda Kayıt

Ayar menüsündeki proje kare hızı, URSA Cine Immersive'nin kayıt formatı kare hızıdır ve çekimlerinizin oynatım hızını belirler. Bu, genel olarak teslimat gereksinimlerinde ve post prodüksiyon iş akışınızda kullanılan oynatım hızı ve ses senkronizasyonuna eşleşecek şekilde ayarlıdır.



Kare hızı ayarlarına erişmek için URSA Cine Immersive’in saniyedeki kare sayısı FPS göstergesini tıklayın. URSA Cine Immersive’in, saniyede 23.98, 24, 25, 29.97, 30, 50, 59.94, 60 ve 90 kare olmak üzere, 9 adet proje kare hızı ayarı vardır. Üç boyutlu çekim yaparken, proje kare hızı saniyede 90 kareye ayarlanmalıdır.

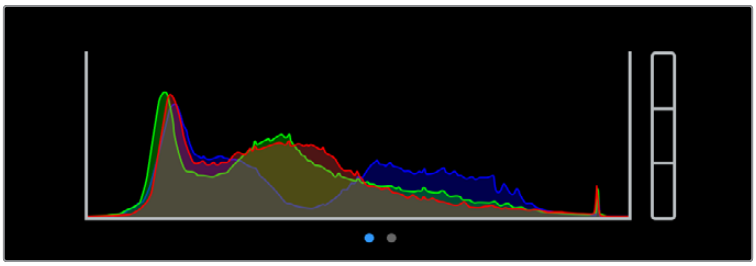
Sensör kare hızı, sensörünüzün bir görüntüyü yakalama hızını belirlemenizi sağlar. Varsayılan olarak, URSA Cine Immersive’inizin proje ve sensör kare hızları doğal bir oynatım hızı için eşleştirilmiştir. Fakat, kameranızın “FPS” menüsünün sağ alt köşesindeki “farklı hız kare hızı” düğmesi simgesini tıklayarak, sensör kare hızını bağımsız olarak ayarlayabilirsiniz.

Düşük hız kare hızı düğmesi etkinleştirildiğinde, sensör kare hızını manuel olarak ayarlayabilirsiniz. Saniyede 90 kare, kameranın en yüksek kare hızıdır ve Apple Vision Pro için çekim yaparken 90 fps'lik proje kare hızıyla eşleşir. Ancak, 30 veya 60 kare gibi daha düşük bir proje kare hızı seçip ardından saniyede 90 kare hızında farklı hızda kayıt yaparsanız, güzel ağır çekim görüntüler elde edebilirsiniz.

Farklı Hız Kare Hızı düğmesi etkinleştirildiğinde, ekranın sol üst köşesindeki kare hızı göstergesi sensör kare hızını ve ardından proje kare hızını gösterir.

Histogram

URSA Cine Immersive'in dokunmatik ekranının sol alt köşesinde, histogramı göreceksiniz. Histogram, bir görüntüdeki parlaklık dağılımını gösteren bir grafiktir. Sol tarafı gölgeleri, ortası orta tonları ve sağ tarafı ise parlaklıkları gösterir. Grafiğin yüksekliği, görüntünün o parlaklık seviyesindeki alanını gösterir.



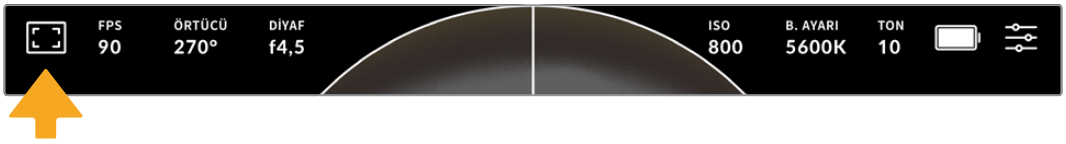
Histogram, bir bakışta klibinizdeki ton dağılımını gösterir. Grafiğin sol tarafta olduğu bir görüntüde, muhtemelen pozlama yetersizdir. Benzer şekilde, grafiğin sağ tarafta olduğu bir görüntü de aşırı pozlanmış olabilir. Histogram daima görüntüye göre yorumlanmalıdır, ancak genel olarak iyi pozlanmış bir görüntü, ortada dengelenmiş bir grafiğe sahiptir. Pozlamayı artırmak histogramı sağa kaydırır, pozlamayı azaltmak ise sola kaydırır.

Histogramın sağında, tek tek renk kanallarının ne zaman kırpıldığını gösteren bir gösterge bulunur. Genellikle, herhangi bir kanalı kırpmaktan kaçınmanız gerekir. Bununla birlikte, görüntünün orta tonlarını ve gölgelerini korumak için görüntüyü kırpmanın kabul edilebileceği bazı durumlar vardır.

Sağa Pozlama tekniği, renk kanallarını kırpmadan histogramı mümkün olduğunca sağa doğru iterek görüntüyü kasıtlı olarak aşırı pozlama tekniğidir. Bu, sinyali en üst düzeye çıkarır ve gürültüyü en aza indirir. Post prodüksiyon aşamasında pozlama değeri düşürüldüğü sürece, temiz görüntüler elde etmek için bu, mükemmel bir yöntemdir!

LCD Ekran Seçenekleri

LCD ekran ayarlarına erişmek için URSA Cine Immersive'in dokunmatik LCD ekranının sol üst köşesindeki "monitör" simgesini tıklayın. Bu menüde, ekran parlaklığı gibi kullanışlı seçenekler bulunur. Ayrıca etkili pozlama araçları da bulunur.



Zebra

"Zebra" ayarı, dokunmatik ekrandaki zebranın görünümünü değiştirmekle birlikte zebra seviyesini de ayarlar. Zebra, görüntünüzün belirli bir pozlama seviyesini aşan alanlarının üzerine çapraz çizgiler yerleştirir. Yüzde ayarı, görüntünüzün aşırı pozlanmış kısımlarını göstermek için kullanılır. Bu değeri %100 yerine %95 gibi bir değere ayarlamak yaygındır, çünkü bu şekilde, görüntünün aşırı pozlanmaya maruz kalan bölümlerini belirleyebilirsiniz. İkinci zebra ayarı; orta gri zebra bölgelerini ya da orta gri +1 ayarını açmanızı veya kapatmanızı sağlar.

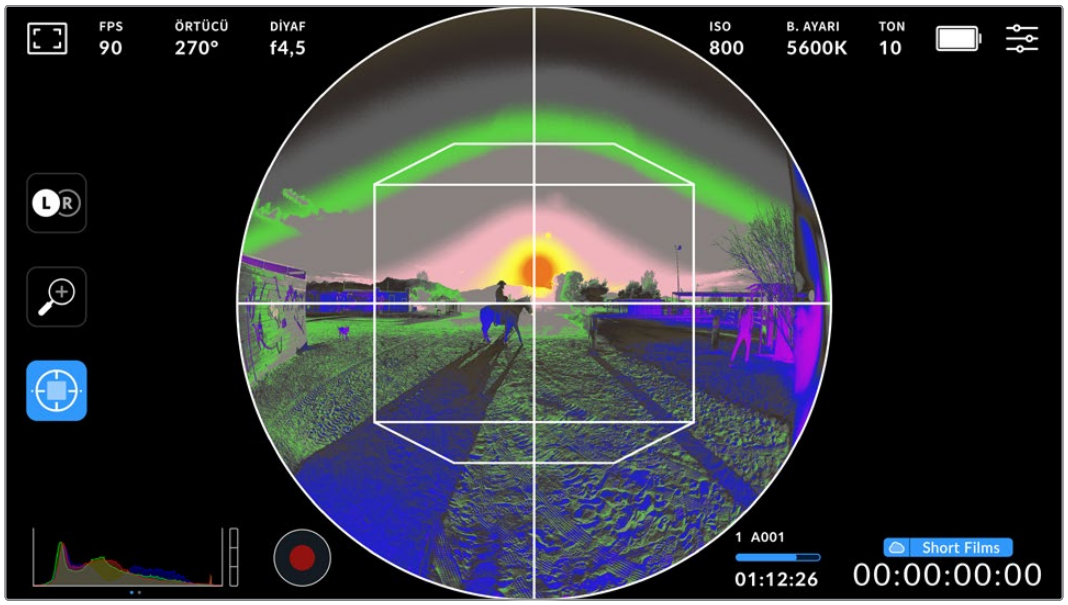
Kılavuzlar

"Kılavuzlar" ayarı, dokunmatik LCD ekranda bir üçler kuralı kılavuzu, denge göstergesi, hedef veya orta nokta görünümünü değiştirmenin yanı sıra URSA Cine Immersive'in tüm çıkışlarında görünür olacak katmanı belirler. Üç boyutlu içerik çekerken denge göstergesi çok kullanışlıdır. Kamera tamamen düz konumda olduğu sürece görüntünün ortasında çift artı işareti görüntülenir, ancak kamera düz konumda olmadığında, bu işaretler birbirinden uzaklaşır.

Sahte Renk

Sahte Renk özelliği; görüntünüzdeki farklı ögeler için pozlama değerlerini temsil eden farklı renk katmanlarını, görüntünüzün üzerine bindirir. Örneğin; pembe renk, açık ten tonları için ideal pozlamayı temsil ederken, yeşil renk daha koyu olan ten tonları için uygundur. İnsanları kaydederken, pembe veya yeşil sahte renklerinin izlenmesi, cilt tonları için gerekli olan tutarlı pozlamanın muhafaza edilmesini sağlar.

Benzer şekilde, görüntünüzdeki ögeler sarıdan kırmızıya dönüştüğünde, bu aşırı pozlamaya maruz kaldıklarını gösterir. Histograma benzer şekilde, bu araç görüntünün ton dağılımını çok hızlı ve sezgisel bir şekilde anlamanıza yardımcı olur.



Kayıt İşlemi

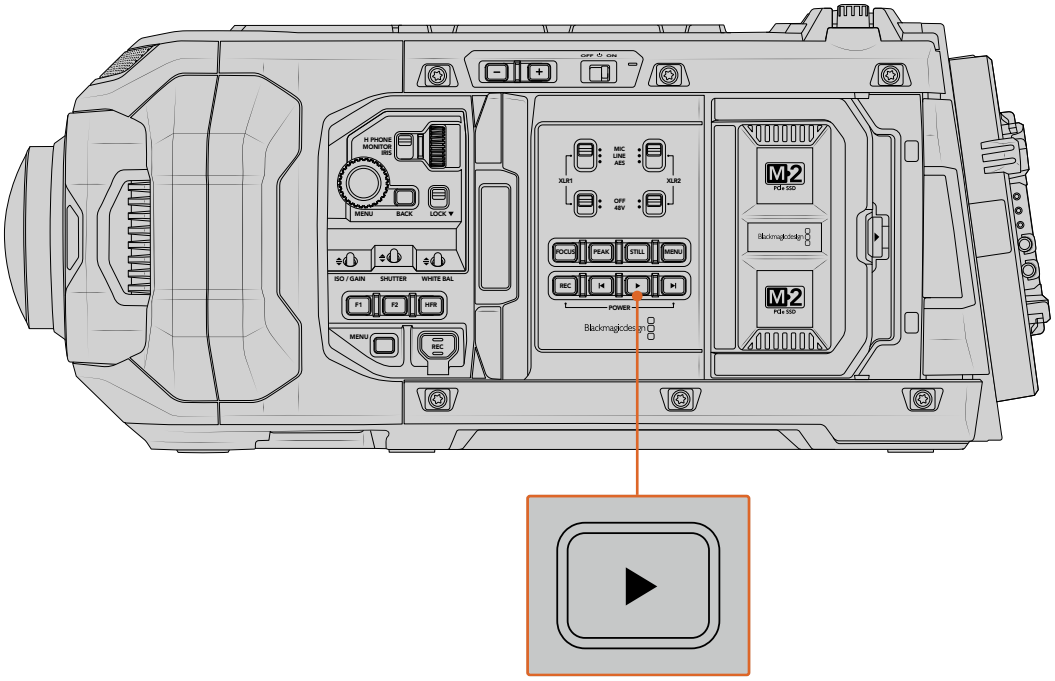
Kaydı başlatmak için, kameranızdaki kırmızı kayıt butonlarından herhangi birine basın. Kayıt butonları; ön kontrol panelinde, katlanabilir dokunmatik ekranın arkasında bulunan iç kısımdaki kontrol panelinde, asistan paneli kontrol panelinde ve her iki dokunmatik LCD ekranda bulunur. Kaydı durdurmak için kayıt butonuna tekrar basın.

Depolama göstergeleri, ne kadar kayıt süresinin kaldığını gösterir. Bu süre, seçtiğiniz kare hızına göre saat:dakika:saniye olarak görüntülenir. Kayıt süresinin üzerindeki ortam çubuğu simgesi, mevcut durumuna bağlı olarak mavi, beyaz veya kırmızı renkte olur ve depolamada kullanılan hacmi gösterir.



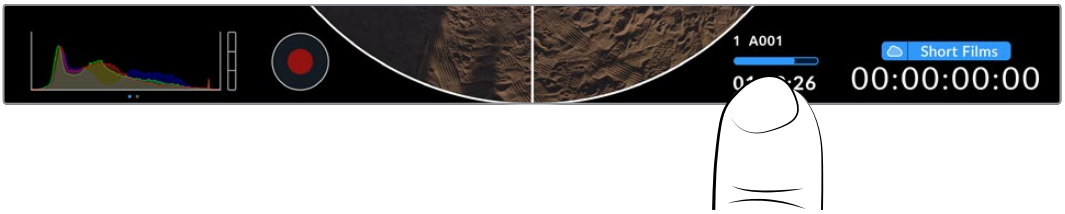
Oynatım

Videonuzu kaydettikten sonra, kliplerinizin oynatımı için aktarım kontrol butonlarını kullanabilirsiniz. Kaydettiğiniz son klibi oynatmak için, kameranın sol tarafında, katlanabilir dokunmatik ekranın arkasında bulunan fiziksel oynatım butonuna basın. Geri ve ileri butonları başka kliplere geçmek için kullanılabilir. Oynatım modundan çıkmak ve kamerayı kayda devam etmeye hazır hale getirmek için kayıt butonuna bir kez basın.

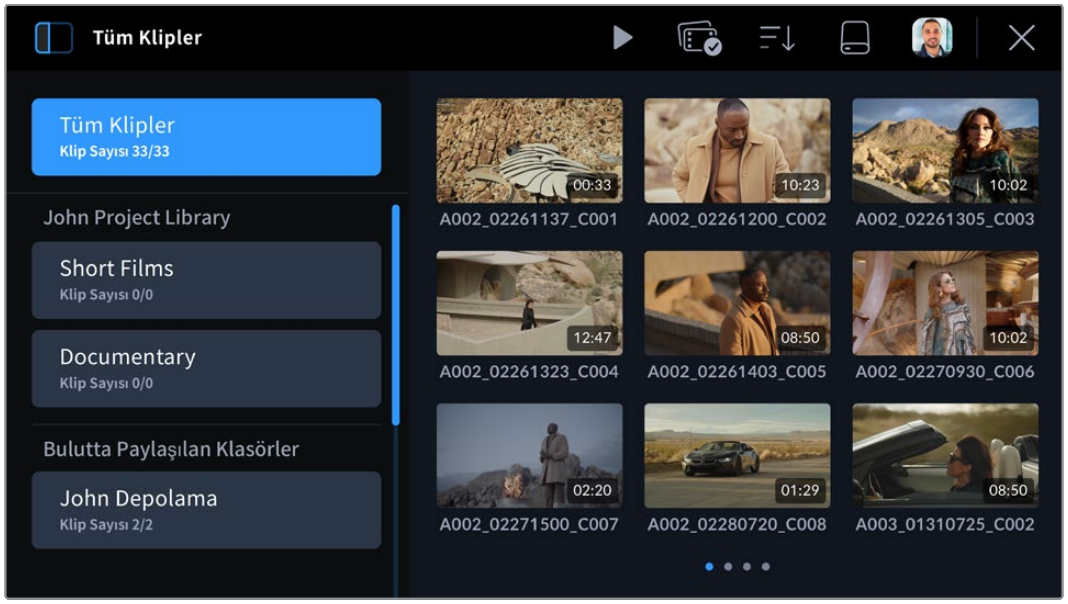


Kliplerin başına veya sonuna atlamak için, “ileri” veya “geri” butonlarına basın. Mevcut klibin başlangıcına gitmek için, “geri atlat” butonuna bir kez basın veya bir önceki klibin başlangıcına gitmek için de iki kez basın. 2 kat hızda ileri veya geri yönde oynatım veya sarım için “ileri atlat” butonunu ya da “geri atlat” butonunu basılı tutun. İleri veya geri hızlı oynatım başladıktan sonra, “ileri atlat” veya “geri atlat” butonlarına; 4x sarım için iki kez, x8 için üç kez ve x16 için dört kez basın.

Çektiğiniz görüntülerin daha kapsamlı bir incelemesi gerektiğinde, Medya Havuzunu kullanmayı tercih etmeniz muhtemeldir. Medya havuzunu açmak için, kameranızın dokunmatik ekranının alt kısmındaki depolama göstergelerinden birine dokunun.



Ana sayfa tarayıcısı, kameranıza bağlı tüm ortamlardaki kayıtlı kliplerin küçük resimlerini görüntüler. Üç sıra küçük resim bulunur ve daha klip eklendikçe, dokunmatik ekranı kaydırarak küçük resimlerin bir sonraki sayfasına ulaşabilirsiniz. Ekranın en altındaki sayfa belirteçleri, kaç sayfa klip bulunduğunu gösterir.



URSA Cine Immersive'i bir Blackmagic Cloud hesabına bağlayın ve çekimleri doğrudan DaVinci Resolve projesine gönderin.

Medya havuzundan çıkmak istediğinizde, sağ üst köşedeki X işaretine dokunun veya kameranın kayıt butonlarından herhangi birine bir kez basın.

Kliplerin Aktarılması

Klipleri kameranızın depolama alanından aktarmanın birçok yolu vardır. Örneğin: URSA Cine Immersive'i bir 10G Ethernet kablosu ile bilgisayarınıza veya bir ağa bağlayabilir ve kameranızın depolamasındaki klipleri Ethernet üzerinden kopyalayabilirsiniz. Hatta medya modülünü ağınıza veya bilgisayarınıza bağlı bir Blackmagic Media Dock'a takarak, aynı anda 3 adede kadar medya modülüyle çalışabilirsiniz.

Bilgisayarınıza Ethernet üzerinden bağlanırken, tıpkı bir Blackmagic Cloud Store'u takarken olduğu gibi bilgisayarınız Blackmagic Media Module'yi tanır.

Kameranız varsayılan olarak DHCP'ye ayarlıdır, yani bilgisayarınız veya ağınız kamerayı görür ve kameranıza otomatik olarak bir IP adresi atar.

Dosya Paylaşımının Etkinleştirilmesi

Kameranızın ilk kez kurulumunu yaparken ve bilgisayarınıza veya ağınıza Ethernet üzerinden bağlandığınızda, SMB dosya paylaşımı varsayılan olarak devre dışı bırakılır. Bunun nedeni, dosyaları ağdaki diğer kullanıcılarla hemen paylaşmak isteyeceğiniz Blackmagic Cloud Store'un aksine, önemli bir günün çekimlerinden sonra, muhtemelen ana kamera dosyalarına aynı düzeyde erişime hemen izin vermek istemeyeceğinizdir. Bu nedenle, SMB dosya paylaşımını etkinleştirme seçeneğini size bıraktık.

1. Dosya paylaşımını etkinleştirmek için Blackmagic Camera Setup yardımcı yazılımını indirip, bilgisayarınıza yüklemeniz gerekir. Bu kurulum yardımcı yazılımı, kameranızı güncellemenizi ve yönetici ayarlarını değiştirmenizi sağlayan bir yönetim yazılımı uygulamasıdır. Blackmagic Camera Setup yardımcı yazılımını, https://www.blackmagicdesign.com/tr/support/family/professional-cameras adresinden indirebilirsiniz.
2. Kurulum yardımcı programını indirip kurduktan sonra, kameranızı USB ile bilgisayarınıza bağlayın ve Blackmagic Camera Setup yardımcı yazılımı başlatın. Kameranızın üst panelinin arka tarafındaki USB portu üzerinden bağlayın.

3 Ayarları açmak için URSA Cine Immersive'inizin resmini veya ayarlar simgesini tıklayın.

4 "Ağ erişimi" ayarlarına gidin ve "dosya paylaşımı (SMB)" seçeneğini etkinleştirin. Ardından, bu ayarların etkili olması için sağ alt köşedeki "Kaydet" butonunu tıklayın.

5 Şimdi altındaki URL ayarına gidin ve otomatik olarak adrese gitmek için URL adresini tıklayın. URL adresi, kameranızın adını yerel ağ konumu olarak gösterir.

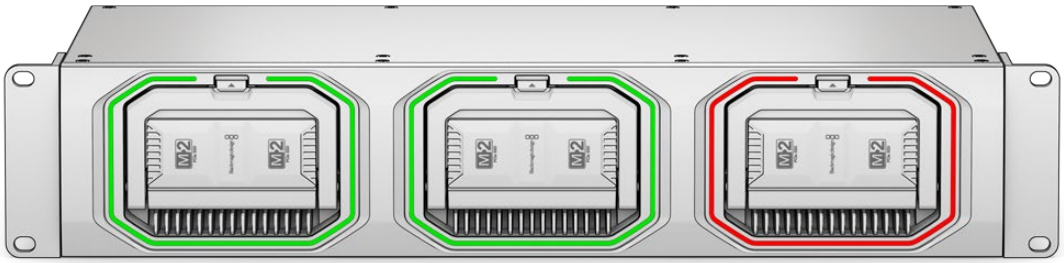
6 "Kaydet" butonunu tıklayın.

Doğrudan kliplerinize gitmek için kameranızın URL ismini tıklayın. Şifre girmenizi veya misafir olarak bağlanmanızı isteyen bir pencere açılır. "Misafir" seçim butonunu tıklayın ve "bağlan" ibaresini tıklayın.

7 Mac bilgisayarlar Finder uygulamasını kullanarak ve Windows bilgisayarlar da Windows Explorer uygulamasını kullanarak, otomatik olarak medya modülünüzü gösterir. Yardımcı yazılım, Ethernet kurulumunuzu hatırlayacağından, ilk kez kurulum yaparken bunu yalnızca bir kez yapmanız gerekir ve kameranızı bir daha Ethernet aracılığıyla taktığınızda, bu ayarların hiçbirine dokunmadan medya modülüne hemen erişebilirsiniz.

Blackmagic Media Dock Kullanımı

Blackmagic Medya Module, klipleri harici bir depolamaya aktarmanız veya kayda devam edebilmek için boş bir modülle değiştirmeniz gerektiğinde, hızla takılıp çıkarılacak şekilde tasarlanmıştır. Blackmagic Media Dock, kamera olmadan bir Medya Module'deki çekimlere erişmek için kullanılabilir.



Özet

Apple Immersive Video, görsel hikaye anlatımına yeni bir yaklaşım getiriyor ve geleneksel 2B sinemaya alternatif sunuyor. Bu, geleneksel sinemanın yerini alması için değil, projenin ihtiyaçlarına göre seçilebilecek ek bir format olarak tasarlanmıştır. Her formatta olduğu gibi, yaratıcı seçimler de anlatılan hikayeye göre değişiklik gösterecektir.

Yeni formatlar genellikle karmaşık iş akışları gerektirirken, URSA Cine Immersive, Apple Immersive Video'yu çekme sürecini basitleştirmek için tasarlanmıştır. Üç boyutlu içerik için kolay anlaşılır bir yapım yöntemi sağlar ve görsel efektler ve başka uygulamalar için de kullanılabilir.

Bu kılavuz, URSA Cine Immersive'in temel özelliklerini ve iş akışlarını özetlemektedir. Bir başlangıç noktası olarak tasarlanmıştır. Herhangi bir kamera sisteminde olduğu gibi, yapımınızdaki özel ihtiyaçlarınız için en iyi sonuçları elde etmek amacıyla daha fazla araştırma ve test yapmanız önerilir.

DaVinci Resolve Studio



Kurgu, renk düzeltme, görsel efekt, hareketli grafik ve ses post prodüksiyonunu akıcı şekilde birleştiren DaVinci Resolve Studio, kapsamlı bir post prodüksiyon yazılımıdır. Ultra yüksek çözünürlüklü stereoskopik video, 90 fps kare hızı ve Spatial Audio miksajı da dahil, Blackmagic RAW Immersive formatının eşsiz niteliklerine uygun hale getirilmiş iş akışlarını destekler. Apple Immersive Video'nun prodüksiyonu, görüntülenmesi ve dağıtımı için eksiksiz donatılmış olması, DaVinci Resolve Studio'yu üç boyutlu içerik yapımcıları için güçlü bir araç haline getirir.

İş Bilgisayarı Önerileri

Donanım Önerileri

- M3 Ultra işlemcili Mac Studio Masaüstü
- Yüksek yenileme hızına sahip monitör (asgari 60Hz)

Asgari sistem gereksinimleri

- MacOS 15 veya sonraki sürüm
- 64 GB sistem belleği
- Görüntüleme için Blackmagic Design Desktop Video 12.9 veya daha yeni sürümler
- Apple Silicon tabanlı bilgisayar veya Metal destekli GPU

Üç boyutlu görüntüleme gereksinimleri

- Apple Vision Pro
- Yüksek hızlı kablosuz ağ (asgari Wi-Fi 5; önerilen Wi-Fi 6)

Proje Kurulumu

Blackmagic URSA Cine Immersive ham medyayı yüklemeden önce, DaVinci Resolve'un proje ayarlarını gözden geçirerek medyanın doğru “demosaic” işlemine tabi tutulduğundan ve en iyi görsel sonuç ve kurgu kolaylığı için doğru şekilde işlendiğinden emin olmakta fayda vardır.

Ana Ayarlar

Bir proje oluştururken, içeriği Apple Vision Pro için oluşturduğunuzu belirtmeniz çok önemlidir. Bu, projenizi hazırlarken, birleştirirken ve çıktı alırken, tüm doğru ayarları ve iş akışı seçeneklerini görmenizi sağlar.

1. DaVinci Resolve Studio’yu başlatın.
2. “Project Manager” penceresinin sağ alt köşesindeki “New Project” ibareli yeni proje seçeneğini tıklayın.
3. Bir proje adı girin, gerekiyorsa “Media Location” ibareli medya konumunu belirleyin ve “Create” ibaresini tıklayın.
4. DaVinci Resolve Studio'nun sağ alt köşesindeki çark simgesini tıklayarak “Project Settings” yani proje ayarlarını açın.

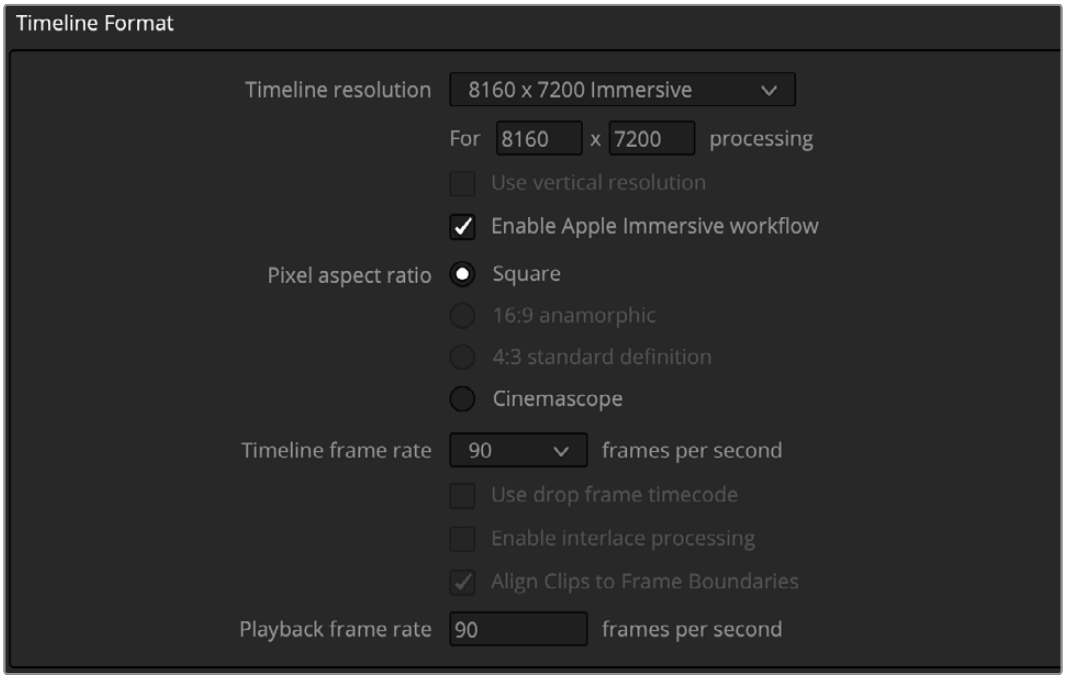
   İlk olarak, projenin çözünürlüğünü ve kare hızını ayarlayın.
5. Kenar çubuğundaki “Master Settings” ibareli ana ayarları seçin.
6. “Timeline Format”, yani zaman çizelgesi formatı bölümünde, “Enable Apple Immersive workflow” ibareli Apple Immersive iş akışını etkinleştirmeyi seçin.



   Bu, üç boyutlu iş akışlarına en iyi şekilde uyumu sağlamak için birden fazla DaVinci Resolve ayarını yapılandırır. Üç boyutlu bir proje oluştururken, her zaman ilk bu ayarı seçtiğinizden emin olun.
7. Klipleri orijinal sensör çözünürlüğünde işlemek için “Timeline resolution” ayarını, “8160 x 7200 Immersive” olarak değiştirin.

   İşlem gücü sınırlı bir bilgisayarla çalışırken, hızlı oynatım için çözünürlüğü yarıya düşürmek için “4080 x 3600 Immersive” seçeneğini belirleyin.
8. “Timeline frame rate” yani zaman çizelgesi kare hızı ayarı, saniyede 90 kare olmalıdır. Apple Immersive iş akışlarını etkinleştirildiğinde, bu parametre otomatik olarak yapılandırılır.
9. Üç boyutlu video içeriğinizin en doğru şekilde oluşturulması için “Playback frame rate” ibareli oynatım kare hızını, saniyede 90 kareye ayarlayın.

   İşlem gücü sınırlı bir bilgisayarla veya yenileme hızı düşük bir monitörle çalışırken, daha istikrarlı bir oynatım için kare hızını düşürün. 30 fps veya 60 fps gibi, yerel kare hızının kesiri olan bir kare hızı seçmeniz önerilir.



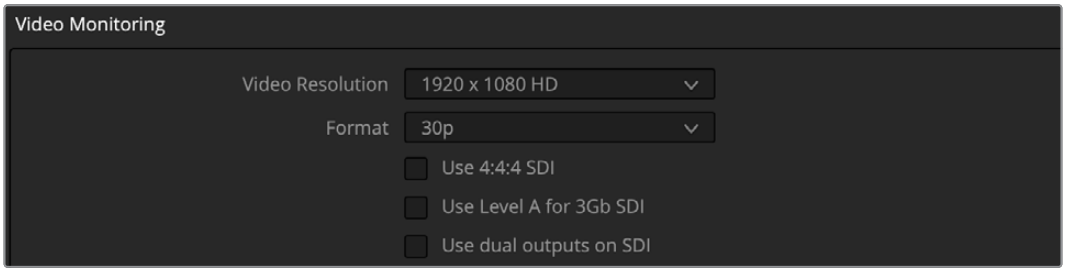
Video Görüntüleme

Decklink veya UltraStudio gibi bir kayıt/oynatım cihazı aracılığıyla harici bir monitör kullanıyorsanız “Video Monitoring” ibareli görüntüleme ayarlarını, monitörünüzün özelliklerine göre yapılandırın.

1. “Video Monitoring” bölümündeki “Video Resolution” seçeneğini, monitörünüzün çözünürlüğüyle eşleşecek şekilde ayarlayın. Örneğin; 1920 x 1080 HD veya 3840 x 2160 Ultra HD.

   Üç boyutlu videonun yerel kare hızı 90 fps olsa da bu kare hızı, çoğu yayın monitörü tarafından desteklenmez.

2. “Format” seçeneğini, monitörünüzün kare hızıyla eşleşecek şekilde ayarlayın. Örneğin; 30p.



Görüntü Ölçeklendirme

“Image Scaling” ibareli görüntü ölçeklendirme ayarları, üç boyutlu medya çözünürlüğünün ve görüntü oranının görüntüleyicide doğru görüntülenmesini sağlar. “Master Settings”de “enabled the Apple Immersive workflow” ibaresi seçildikten sonra, bu parametreler otomatik yapılandırılsa da doğru olduklarınından emin olmanız gerekir.

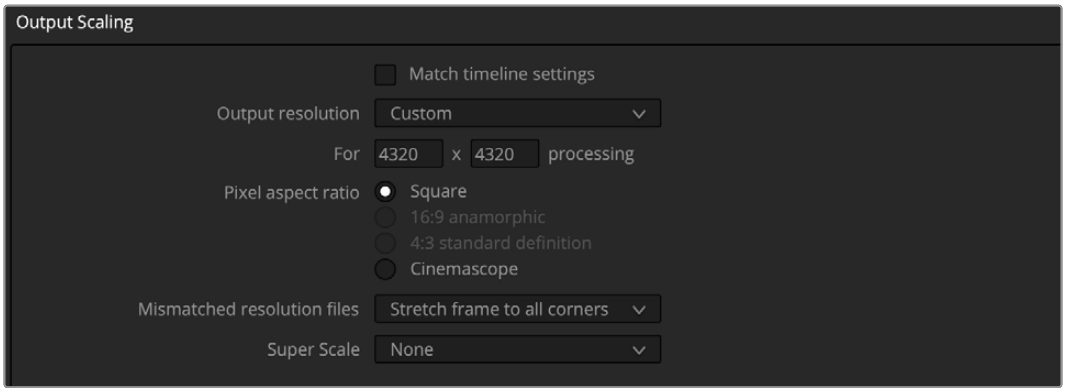
1. "Project Settings" ibareli kenar çubuğunda, "Image Scaling", yani görüntü ölçeklendirmeyi seçin.
2. "Output Scaling" ibareli bölümde, "Match timeline settings" seçeneğinin işaretlenmediğinden emin olun.

3 "Output resolution" ayarının "Custom", yani özele ve **4320 x 4320**'ye ayarlandığından emin olun.

Küçük ekranlar veya hızlı oynatım için yarı çözünürlük olan **2160 x 2160**'a ayarlayın.

Kare görüntü oranlı bu çözünürlükler, DaVinci Resolve Studio dahilinde Apple Immersive video içeriği oynatmak ve (Wi-Fi veya Developer Strap aracılığıyla) Apple Vision Proya internet yayını yapmak için gereklidir.

4 "Mismatched resolution files" ibareli çözünürlüğü uyumsuz dosyalar ayarında, "Stretch frame to all corners", yani çerçeyi tüm köşelere genişlet seçeneğinin belirlendiğinden emin olun.



Renk Yönetimi

Azami çıkış parlaklığı nit değeri üzerinde özel kontrol gerekliliğinden, Color sayfasında Color Space Transform (CST) düğümleri kullanarak renk yönetimi önerilir. "Project Settings" yani proje ayarlarındaki renk yönetimi ayarlarının amacı, Color sayfasının zaman çizelgesi veya çalışma renk alanını belirlemek ve işlenen dosya ile bağlantılı renk alanı metaverisini ayarlamaktır.

1 "Project Settings" kenar çubuğunda "Color Management", yani renk yönetimini seçin.

2 "Color science", yani renk bilimi ayarını "DaVinci YRGB" olarak bırakın.

3 "Timeline color space" ibareli zaman çizelgesi renk alanı ayarını, "DaVinciWG/Intermediate" olarak belirleyin.

Bu, HDR iş akışlarını destekleyen, geleceğe hazır bir geniş gam renk alanıdır ve Blackmagic URSA Cine Immersive medya ile Apple Vision Pro çıktıları için idealdir.

4 "Output color space" ibareli çıkış renk alanı ayarını, "P3-D65 ST2084 1000 nit" olarak belirleyin.

Bu renk alanı, Mac Studio masaüstü bilgisayarda DaVinci Resolve görüntüleyicilerini denetlemek için uygun renk gamını sunar. Bu, genellikle Display P3'dir. Bir renk derecelendirme monitörü veya farklı bir Apple ICC profili kullanıyorsanız çıkış renk alanı, ekranınızın renk standardına uyacak şekilde değiştirilmelidir.

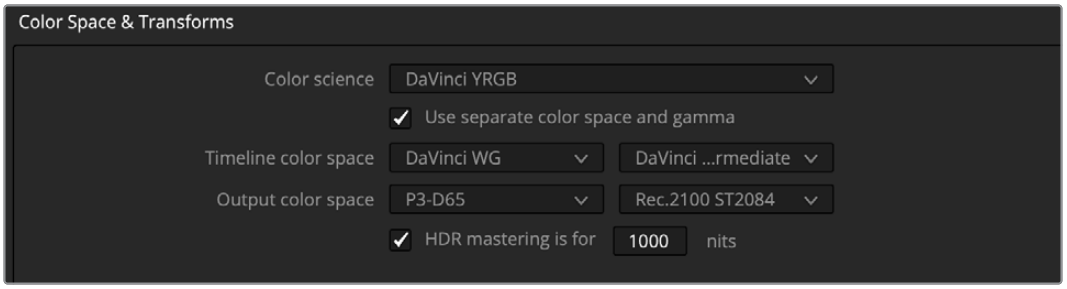
Ancak, ICC profili ile doğru eşleştirme için gama (transfer fonksiyonu) değerini değiştirmeniz gerekir.

5 Renk alanı ayarlarının üzerindeki "Use separate color space and gamma" seçeneğini tıklayın.

Bu, açılır menüdeki renk alanı ve gama alanlarını ayırır.

6 Sag sütundaki çıkış gamasını, "Rec.2100 ST2084" olarak ayarlayın.

7 "HDR mastering is for 1000 nits.", yani 1000 nit için HDR ana kopya oluşturma seçeneğini belirleyin.



Bir sonraki Apple Immersive projenizi daha kısa sürede ayarlamak için, mevcut ayarları bir önayar olarak kaydedebilirsiniz.

8 "Project Settings" penceresinin sağ üst köşesindeki üç noktalı seçenekler menüsüne gidin ve "Save Current Settings as Preset" seçeneğini belirleyerek, mevcut ayarları önayar olarak kaydedin.

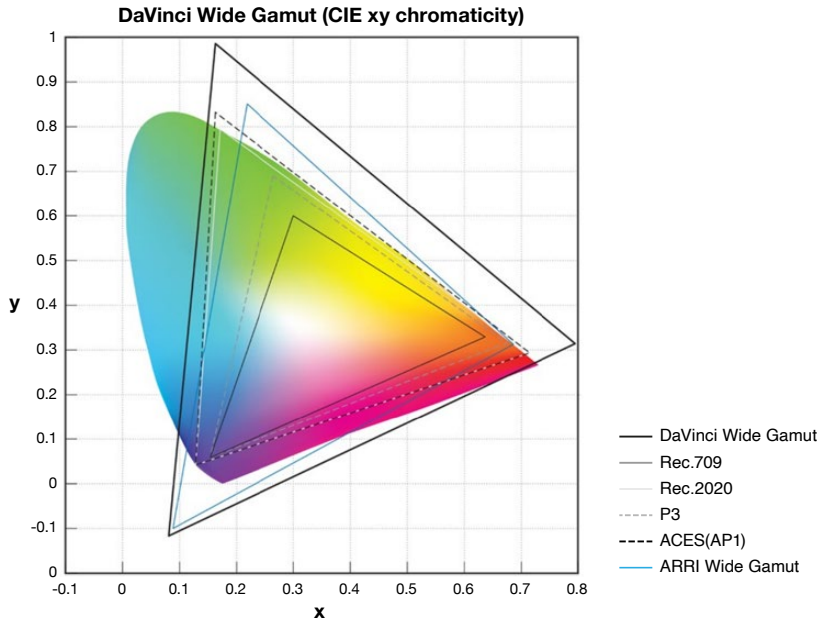
9 Bu önayara bir isim verin "Ok" ibaresini tıklayın.

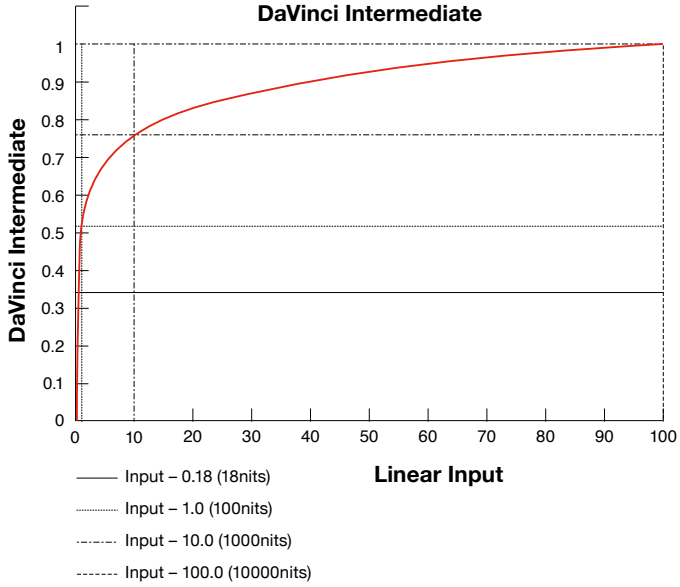
Bir sonraki 3 boyutlu medya çalışmanızda proje ayarlarını hızla yapılandırmak için, "Project Settings" penceresini açın, seçenekler menüsünü tıklayın ve listedeki bu önayarı seçin.

10 “Project Settings”i kapatmak için "Save" ibaresini tıklayın.

DaVinci Geniş Gam Hakkında

Blackmagic Design'in DaVinci Resolve için geliştirdiği gelişmiş dahili renk alanı "DaVinci Wide Gamut" yani DaVinci Geniş Gam, renk derecelendirme ile görüntü işleme için olağanüstü genişlikte ve hassas bir çalışma ortamı sunar. "DaVinci Intermediate" gama ile birlikte kullanıldığında, farklı kamera formatları üzerinde renk yönetimini basitleştiren ve sinyal girişinden çıkışına kadar tutarlı renk tonu ve doygunluk eşleşmesi sağlayan, bütünleştirilmiş bir çalışma alanı oluşturur. HDR ve 3 boyutlu medya ile çalışmak için ideal olan bu geniş renk gamı, aslına bağlı teslimat formatları için ana kopya oluştururken, renk uzmanlarının kuşaklanma veya ton kırpma riski olmadan ince ayarlamalar yapmalarını sağlar.





Apple Immersive İşleme

"Project Settings" kenar çubuğundaki "Apple Immersive" bölümü, üç boyutlu içeriğin kalibre edilme ve gösterilme şeklini değiştirmek üzere özel bir ILPD (Üç Boyutlu Lens Profil Verisi) ayarlamanıza olanak tanır.

1. "Project Settings" kenar çubuğundaki "Apple Immersive" ibaresini seçin.
2. "Open Calibration Folder" seçeneğini tıklayarak, kalibrasyon klasörünü açın.
3. Özel bir ILPD dosyası ekleyin veya dosyayı güncelleyin.
4. "Media Pool"da bulunan ILPD listesini yenilemek için "Update Lists" seçeneğini tıklayın.
5. "Project Settings"i kapatmak için "Save" ibaresini tıklayın.
6. "Media Pool"da, bir üç boyutlu klibi sağ tıklayın.
7. İçerik menüsünde, "Apple Immersive Calibration" seçeneğine gidin ve kaynak ILPD için "No Apple Immersive Calibration File Selected" ibaresini, bir özel ILPD dosyasını veya bir aralık ILPD dosyasını seçin.

Varsayılan olarak, "Custom ILPD Calibration" klasörü, varsayılan LUT klasörüyle birlikte ProIM klasörünün içindedir.

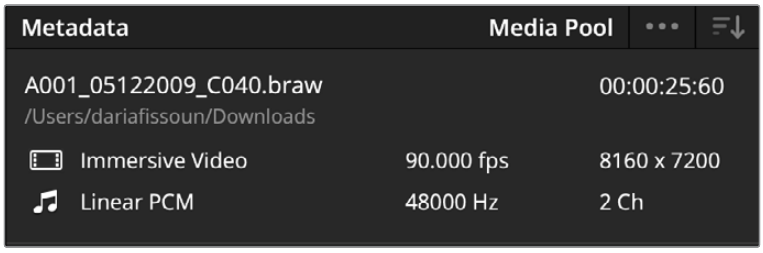
Dosyaların Yüklenmesi

Ham üç boyutlu içeriği yazılıma yüklerken, her iki göz ve tüm bağlantılı üç boyutlu metaveri de yüklenen .braw kliple birlikte taşınır.

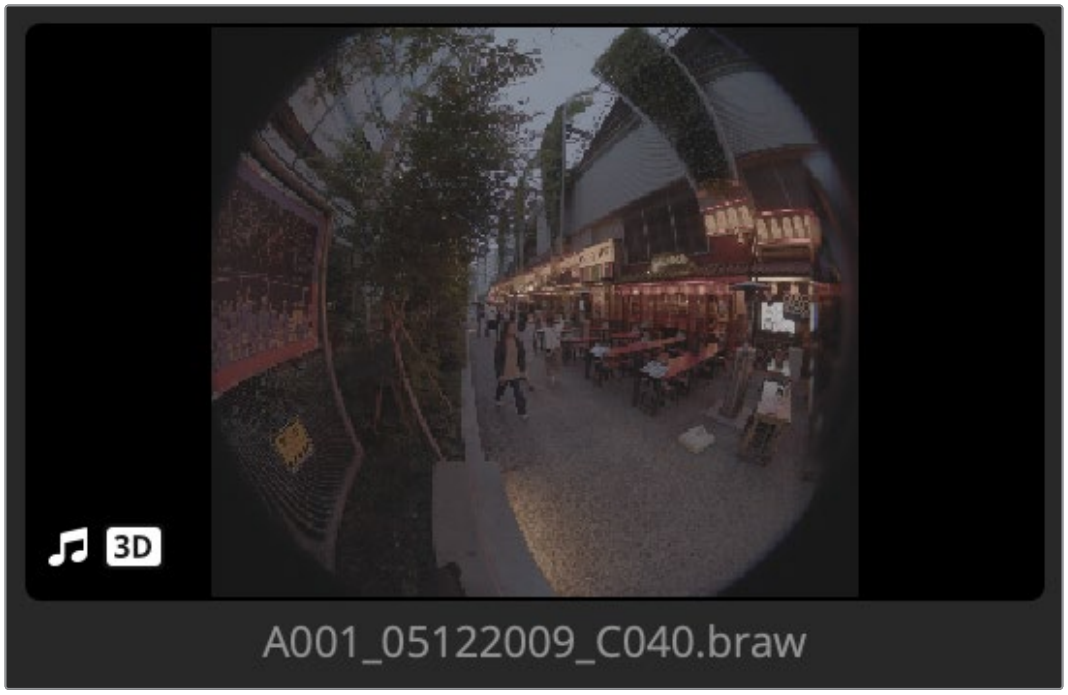
1. "Media" sayfasına gidin.
2. “Media Storage” ibareli medya depolama panelinde, üç boyutlu video içeriğinin bulunduğu yere gidin.
3. Medyayı, sürükleyerek alttaki "Media Pool"a bırakın.

BİLGİ: Orijinal klasör yapısını korumak için klasörleri, “Media Storage” panelinden "Media Pool" kenar çubuğuna sürükleyin. “Master” ibareli ana medya kutusuna veya oluşturduğunuz herhangi bir medya kutusuna sürüklendiğinde, klasörler ve alt klasörler korunur.

Bir Apple Immersive video klibi seçildiğinde, “Metadata” ibareli metaveri panelinin başlığındaki kodek parametresi, bu klibin bir Immersive Video olduğunun yanında, yerel kare hızını ve çözünürlüğünü de gösterir.



Ayrıca, video küçük resminin sol alt köşesindeki bir 3D simgesi, videonun 3D bir klip olduğunu doğrular.



NOT: “Media Pool”a ilk kez üç boyutlu medya sürükleyip bıraktığınızda açılan bir iletişim penceresi, “Keep Settings” (Ayarları Koru) veya “Change Settings” (Ayarları Değiştir) seçeneklerinden birini seçmenizi isteyebilir. “Change Settings” seçeneği, “Project Settings” yani proje ayarlarını kaynak medya kare hızını ve çözünürlüğünü otomatik seçmeye ayarlar.

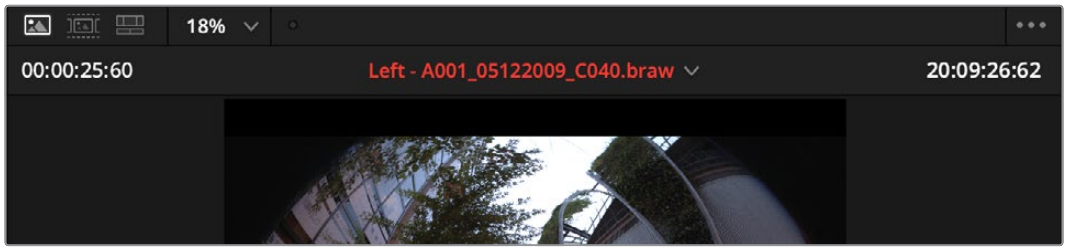
Kurgu

Üç boyutlu içerik üzerinde çalışırken, kurgu araçları ve uygulamalar, çoğunlukla alışıldık 2 boyutlu video formatlarıyla çalışırken olduğu gibi kalır. Aşağıdaki yönerge dizisi, üç boyutlu video iş akışlarına özgü kurgu özelliklerini veya uygulamalarını açıklar.

Yerleştirme

Zaman çizelgesi yerleştirme sırasında klipleri önizlerken, kaynak görüntüleyicide hangi gözü (sol veya sağ) göreceğinizi seçebilirsiniz. Aksi takdirde, zaman çizelgesinde bir kesit yerleştirme işlemi, standart NLE iş akışları ile aynı kalır.

1. Edit sayfasına gidin.
2. Soldaki kaynak görüntüleyiciye yüklemek üzere bir klibi çift tıklayın. Görüntüleyicinin üst kısmındaki etiket, hangi gözü görüntülediğinizi bildirir.



3. Görüntüleyicideki gözü değiştirmek için, “Media Pool” ibareli medya havuzunda kaynak klibi sağ tıklayın ve Stereoscopic 3D > Stereoscopic 3D Mode > Left Eye/Right Eye/Both Eyes ibarelerini seçin.

BİLGİ: Kaynak görüntüleyici seçeneklerinde “Live Media Preview” özelliğini etkinleştirdiyseniz fareyi “Media Pool”daki küçük resimlerin üzerinde gezdirerek de kaynak görüntüleyicide kliplere göz atabilirsiniz.

4. Kaynak görüntüleyicide bir aralık belirlemek için I ve O klavye kısayollarını kullanın.
5. Zaman çizelgesine klip eklemek için araç çubuğundaki “Insert Clip” butonuna basın veya F9 klavye kısayolunu kullanın.

BİLGİ: Ayrıca, “Media Pool”daki klipleri sürükleyerek, zaman çizelgesi paneline bırakabilirsiniz.

6. Zaman çizelgesindeki klipleri kırpmak için (Selection veya Trim Edit modunda) farenizi kullanın.

Fonlar

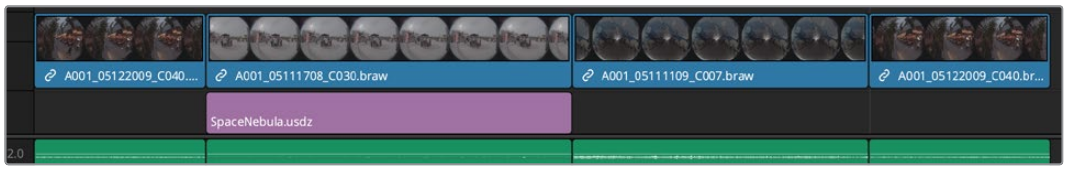
“Project Settings”te “Enable Apple Immersive workflow” ibaresini seçtiğinizde, Video 1 kanalının altında otomatik olarak bir Backdrop (“BD”), yani fon belirir.

Fonlar, izleyici üç boyutlu video içerik çeperinin ötesine baktığında göreceği alan demektir ve genellikle 180 ila 210 derecelik bir görüş alanıyla sınırlıdır.

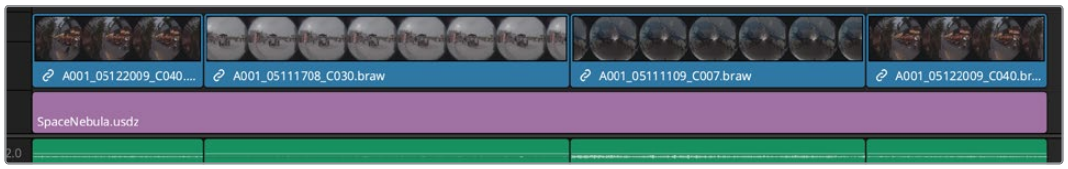
Bu alan, varsayılan olarak siyah olsa da fon kanalını kullanarak, .usdz formatında özel arka plan görüntüleri ekleyebilirsiniz.

1 Bir .usdz fon dosyasını Fon (BD) kanalına sürükleyerek zaman çizelgesine ekleyin.

Fon görüntülerinin süresi, video kanalının klip sınırlarıyla uyumlu olmalıdır. Tek arka plan dosyası, bir veya birden fazla klip ya da tüm zaman çizelgesi boyunca sürebilir, ancak üç boyutlu bir .braw klibin ortasında (diğer adıyla "L-geçiş") kesilemez.



Bir klip – doğru ✓

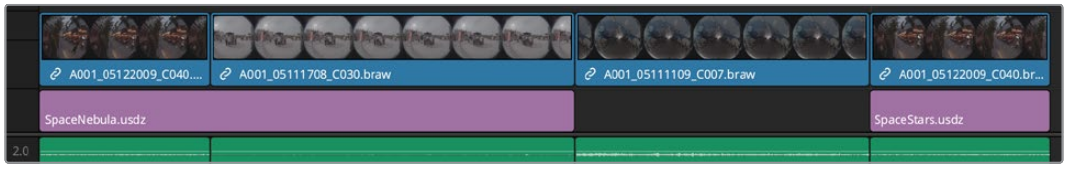


Birden fazla klip – doğru ✓



L geçiş – yanlış ✗

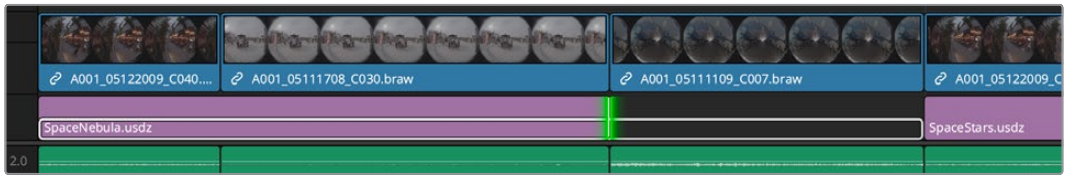
Tek zaman çizelgesinde birden fazla fon dosyası kullanmayı veya varsayılan siyah fonu kullanmak için görüntü eklememeyi seçebilirsiniz.



Farklı görüntüler ve aralıklar – doğru ✓

2 Fon dosyasının uzunluğunu ayarlamak için standart kırpma kontrollerini kullanın.

Uzunluğu ayarlamak için .usdz dosyasının kenarını (Selection veya Trim Edit modunda) sürükleyin.



Ayrıca , (virgül) ve . (nokta) komutlarını kullanarak, kırpılan kenarı kare kare ilerletebilirsiniz.

BİLGİ: Fonların daima Immersive Video düzenlemeleri ile hizallandığından emin olmak için, Edit sayfası araç çubuğunda Snap, yani kapma modunu etkinleştirin.

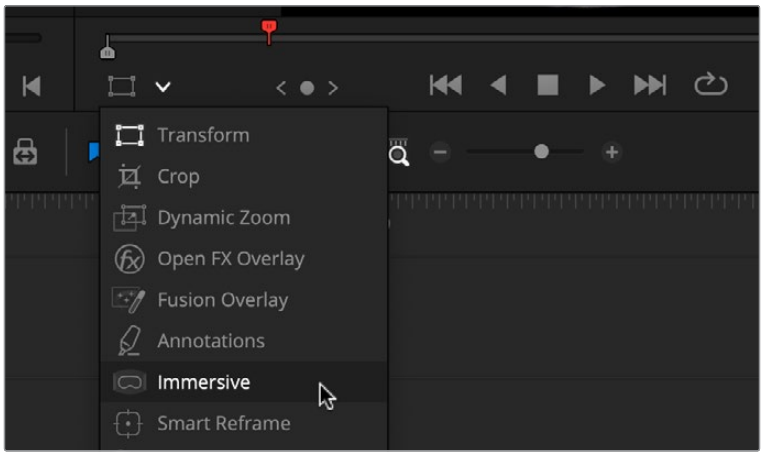
Görüntüleyicinin Kullanımı

Üç boyutlu klipler, varsayılan olarak DaVinci Resolve sayfa görüntüleyicilerinde yuvarlak “lens alanı”nda görüntülenir. Görüntüyü oluşturmak için kullanılan görsel lens kalibrasyonu, kameradan gelir ve klibe özeldir.



Ayrıca, görüntüleme modunu LatLong (Koordinat) veya Viewport (Görünüm Alanı) olarak değiştirmeyi seçebilirsiniz.

1. Zaman çizelgesi görüntüleyicinin sol alt köşesinde, ekran üstü kontrol butonlarının yanında bulunan açılır ok menüsünü tıklayın.
2. “Immersive” ibaresini seçin.



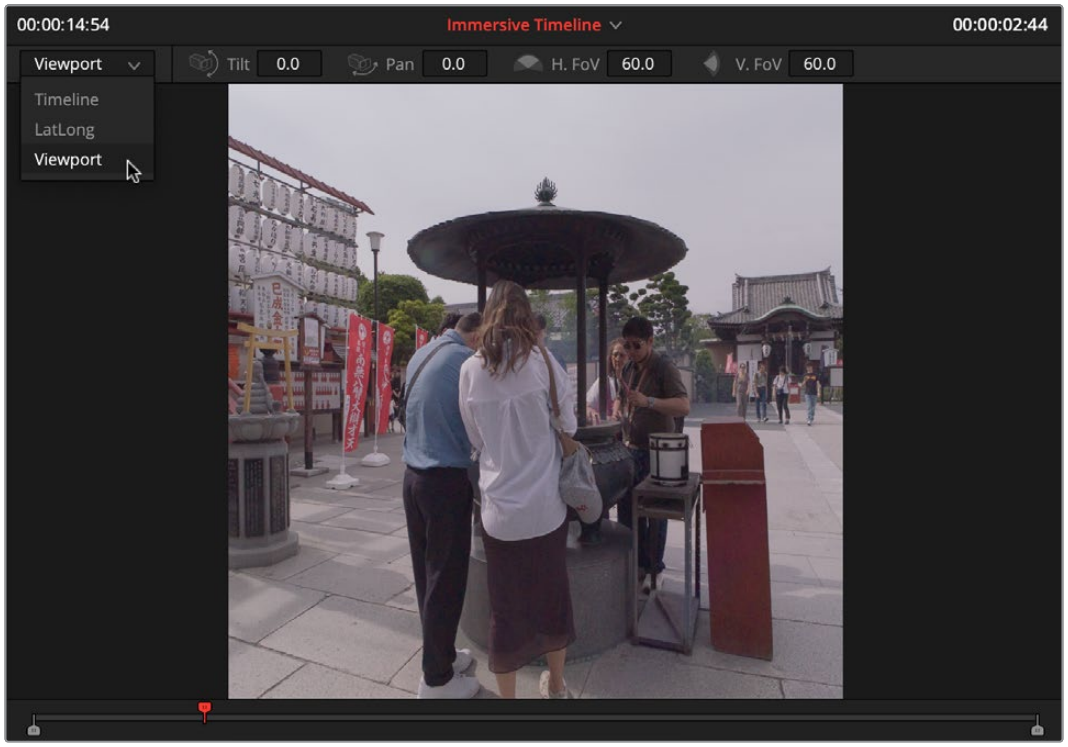
Görüntüleyicinin üzerinde, ek görüntüleme seçenekleri olan yeni bir araç çubuğu belirir.

3. Görüntünün eşkenar dörtgen iz düşümünü görmek için LatLong'a tıklayın. Bu mod, görüntüyü geleneksel 2 boyutlu medya oranlarına daha çok benzemesi için büker ve bu, görüntü takibi ve kurgu gibi belirli post prodüksiyon işlemlerini kolaylaştırır.



4 Üç boyutlu sahnenin orta kesitini kare bir kadraj içinde görmek için, “Viewport” seçeneğini tıklayın. Bu mod, sahnenin doğru orantılı ayrıntılarını standart bir monitörde net bir şekilde görmenize yardım eder, yine de kadraj dışındaki hareketleri kaçırmamaya dikkat etmeniz gerekir.

Üst “Immersive” araç çubuğundaki dönüştürme kontrollerini kullanarak, stereoskopik görüntüyü eğme ve panlamanın yanı sıra, yatay ve dikey görüş alanlarını ayarlayabilirsiniz.



Geçişler

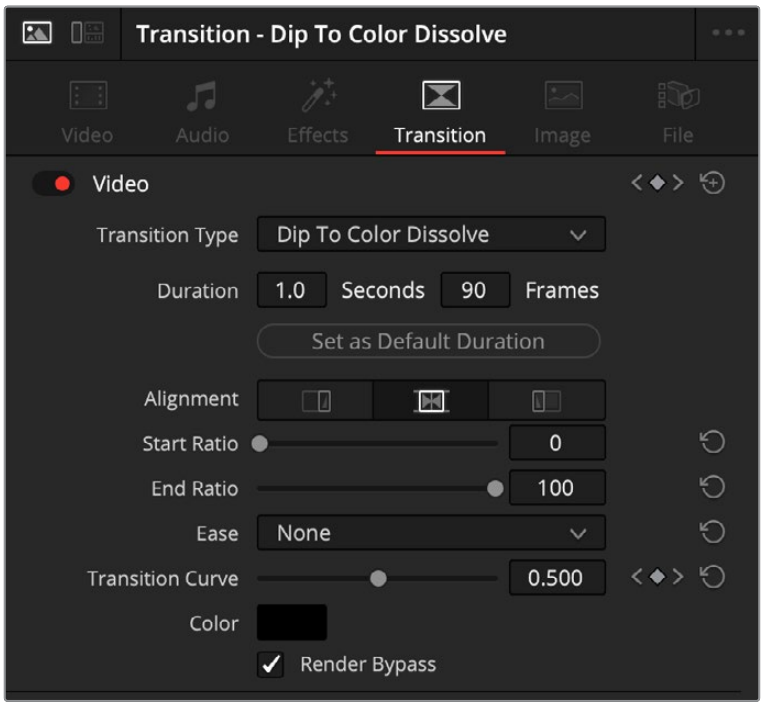
İki adet üç boyutlu video klip arasında geçiş yapmak için; doğrudan kesim, klibin her iki ucundaki tutamakları kullanarak yavaşça görünmesini veya kararmasını sağlayabilir ya da “Dip To Color Dissolve” efektini kullanabilirsiniz. Geçişler, görüntüye kalıcı olarak kaydedilmemeli ve oynatım sırasında Apple Vision Pro tarafından işlenmelidir.

1. “Effekts” panelini açın.
2. Kenar çubuğunda, Toolbox > Video Transitions seçeneklerine gidin.
3. “Dip To Color Dissolve” efektini, iki klip arasındaki kesim noktasına sürükleyin.



NOT: Bir geçiş uygulamak için her iki klibin kesim noktalarının ilerisine uzanan, yeterli miktarda kırpma verisine sahip olmanız gerekir. Kırpma verisi, yani tutamaklar yetersiz ise geçiş uygulanmaz.

4. Denetleyici panelinde, “Transition” ibareli geçiş sekmesine gidin ve “Render Bypass” ibaresini tıklayarak işlemeyi atlayın. Bu, Apple Vision Pro’nun çözülme efektini doğru işlemesini sağlar.
5. Ardından, denetleyici panelindeki kalan “Dip To Color Dissolve” parametrelerini kullanarak geçişin süresini, hizasını, animasyon hızını ve rengini değiştirebilirsiniz.



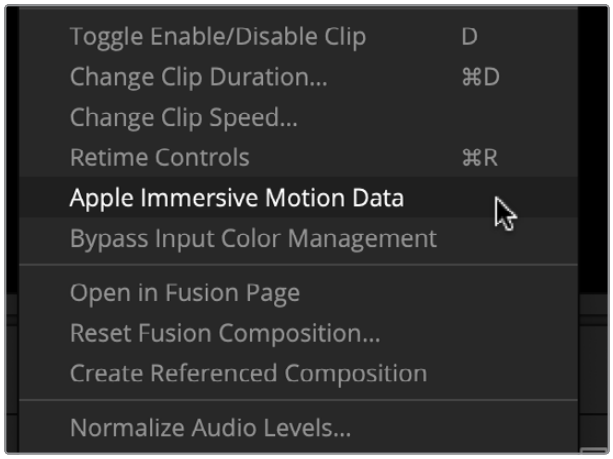
Apple Immersive Motion Data

Üç boyutlu içerikle çalışırken, hızlı veya ani kamera hareketleri izleyicinin yönünü şaşırmasına neden olabilir. İzleyicinin rahat izlemesi için hızlı çevirme/eğme, dönme hareketi ya da dengelenmemiş ufuk çizgisi içeren çekimleri kullanmaktan kaçının.

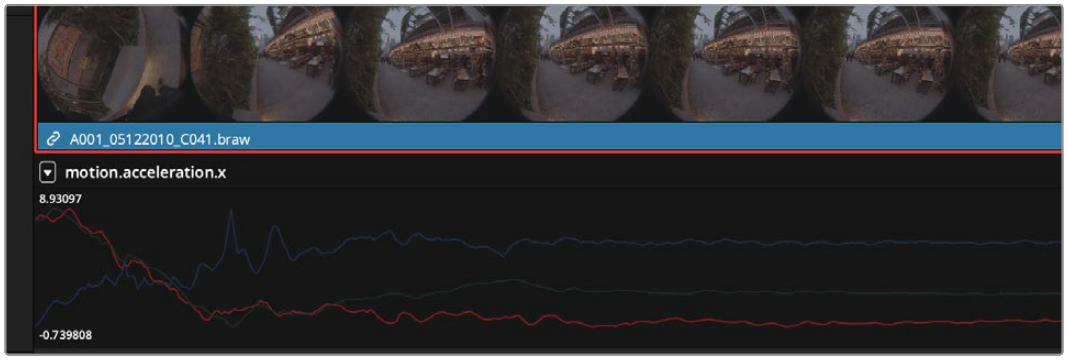
Apple Immersive Motion Data özelliği, kamera titreşiminin ve hareketinin yoğun olabileceği bölümleri ölçülebilir bir şekilde göstermek için x, y ve z ivme ve jiroskop konumu değerlerini bir grafikte gösterir. Görüntülerin kafa karışaştıran bölümlerini kullanmaktan kaçınmak için kurgu uzmanları bu bilgileri kullanabilir.

1 Zaman çizelgesindeki üç boyutlu bir klibi sağ tıklayın.

2 Apple Immersive Motion Data'yı seçin.

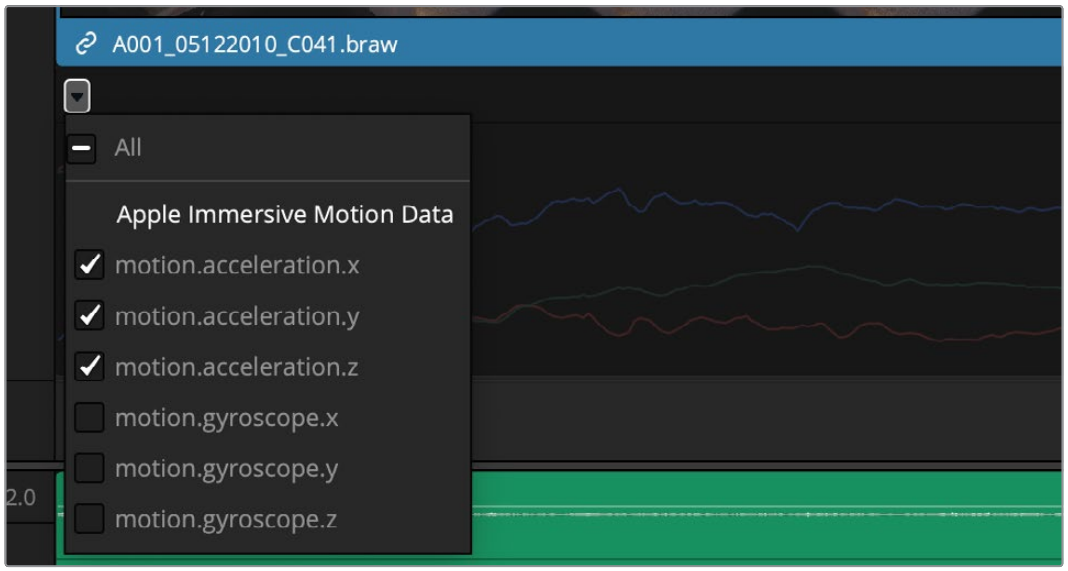


Video kanalında x, y, z ivme oranlarını gösteren bir grafik belirir.



Bu örnekte, kameraman kameranın kurulumunu yaparken kamera kayıttaydı. Grafikteki yükselme noktaları, kameraman kamerayı kaldırıp, döndürüp, omzuna aldığında gerçekleşen ani hareket olduğunu göstermektedir. İdeal olarak, videonun daha sonraki bölümleri, yani kamera sabitlendikten sonra daha düz değerlere sahip olan kısımları kullanılmalıdır.

3 Grafiğin sol üst köşesindeki yer değiştirme okunu tıklayın.



Çekimin içeriğine ve kamera hareketinin türüne bağlı olarak değer okumayı etkinleştirebilir veya devre dışı bırakabilirsiniz.

Yazılar

Edit sayfasının efektler panelindeki Text, Text+, MultiText, Scroll ve ekran altı yazı türleri gibi, yazı araçlarından herhangi birini kullanarak üç boyutlu bir sahneye yazı ekleyebilirsiniz.

Üç boyutlu bir klibin üzerine yerleştirildiğinde KJ, sahnenin lens eğriliğine uyacak şekilde yazıyı doğru bir oranda eğmek için klibin ILPD (Üç Boyutlu Lens Profili Verileri) bilgilerini kullanır.

Kullanılabilir ILPD bilgisi yoksa (örneğin, siyah bir arka plan veya üç boyutlu bir CGI sahnesi üzerine yazı eklerken), Apple Vision Pro'da yazı görüntülendiğinde, uygun miktarda eğilmiş görünmesini sağlamak için varsayılan lens bozulması uygulanır.

1 “Effects” panelinde Toolbox > Titles yani yazılara gidin.

2 Text, yani yazı oluşturucuyu zaman çizelgesine sürükleyin. Sahneye dahil etmek için, üç boyutlu bir klibin üzerindeki bir video kanalına yerleştirin.



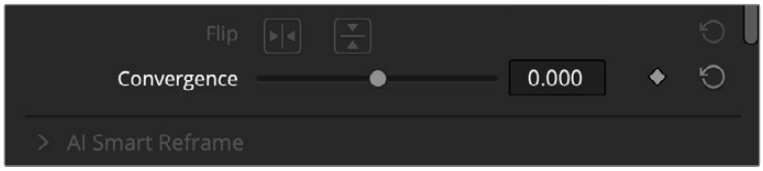
Arka planlarda olduğu gibi, yazı oluşturucular klip sınırlarını geçemez. Her klip ayrı bir lens alanıyla işlendiğinden, üst üste binen yazıların farklı görüntülerde bozulmuş olarak görünme riski bulunur.

3 Yazıyı, video düzenleme noktasına/noktalarına göre kırpın.

4 Yazı metnini yazmak ve biçimlendirmek için denetleyiciyi kullanın.

5 “Settings” yani ayarlar sekmesine gidin ve “Converge” yani yakınsama sürgüsünü kullanarak yazıyı 3D alanda konumlandırın.

Sürgüyü sola hareket ettirmek, yazıyı üç boyutlu alanda arkaya doğru iter, sağa hareket ettirmek de öne doğru getirir.



Yakınsama en iyi Apple Vision Pro'da izlendiğinde görülür. Farklı görüntüleme araçlarını kullanıyorsanız, Color sayfasında alttaki video katmanının “Vision” yani görme modunu değiştirerek, yazı yakınsamasının görüntüleyicide nasıl görüneceğini değiştirebilirsiniz.

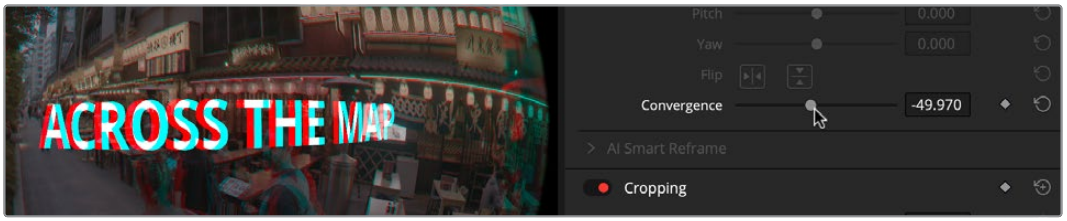
6 Color sayfasına git.

7 3D paleti aç.

Paletin ortasındaki “Vision” sütunu, üç boyutlu medyayı yansıtmak için kullanılan stereoskopik standardı değiştirmenize olanak tanır.

8 “Stereo” sekmesini seçin.

9 Çıkış modunu “Anaglyph” (anaglif) olarak değiştirin.



Edit sayfasının “Inspector” yani denetleyici panelindeki yazı üretecinin “Convergence” yani yakınsama parametresinin değiştirilmesi, anaglif (veya anakrom) gözlüklerle bakıldığında artıkvdaha sezgisel görünecektir.

Yaratıcı Hususlar

Blackmagic URSA Cine Immersive kamera ve Apple Vision Pro kullanarak gerçekleştirilen prodüksiyonların bu erken döneminde; en iyi uygulamaları belirlemek, yeni kurallar oluşturmak ve geleneksel film yapımcılığının daha önce hayal bile edemediği iş akışlarını keşfetmek için, film yapımcıları bu format ile denemeler yapmaya teşvik edilmektedir.

Bundan dolayı, izleyicinin azami düzeyde keyif almasını ve görsel konforu sağlamak için üç boyutlu prodüksiyonun bazı yönleri, dikkatle ele alınmaya başladı.

1 Kurgu süreci, yapım öncesi aşamada başlamalıdır. Çekim uzunluğu ve izleyicinin hareket açısından konforu gibi özel konular nedeniyle, film yapımcıları son kurguyu düşünerek sahnelerin çekimini yapmalıdır.

2 Hızlı kamera hareketleri içeren klipleri kullanmaktan kaçının veya en aza indirin, çünkü bunlar izleyicinin yönünü şaşırtmasına neden olabilir. Bunun yerine, sabit çekimleri veya yavaş hareketleri ve dolly kullanmayı tercih edin.

3 Dengesiz veya aşırı kamera açıları da yön şaşırtabilir, bu yüzden konuyla aynı yükseklikte ve (çok yakın olmayan) rahat bir mesafeden çekimler yapmayı tercih edin.

4 Yeni ortamlar, iki boyutlu bir sahneyi izlerken olduğundan daha uzun bir sürede üç boyutlu medyada algılanır. İzleyicinin yeni ortama uyum sağlaması için kesme geçişlerinin daha yavaş uygulanması gerekir.

5 Klipler arasında kesme geçişi yaparken, görüntüleyicinin daha önceki görüntülerde nereye baktığını aklınızda bulundurun. Daha önceki görüntülerde izleyicinin dikkati ekran karesinin sağ tarafındaysa karenin sol tarafında önemli bir aksiyonun yer almaması gerekir.

6 Spatial Audio, yani uzamsal ses, hikayenin anlatımında büyük bir rol oynar, bu nedenle üç boyutlu yapımlarda onu bir karakter gibi ele alın.

VFX Görüntü Birleştirme

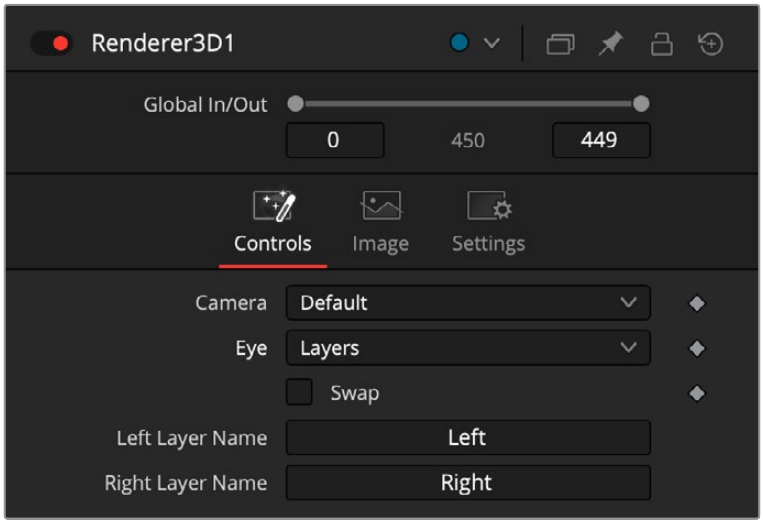
“Project Settings” ibareli proje ayarlarında “Apple Immersive Workflow” etkinleştirildiğinde, özel Fusion parametreleri kusursuz stereoskopik iş akışlarına ve üç boyutlu uyumlu görüntü birleştirmeye olanak tanır. MediaOut düğümleri, sol ve sağ göz çıkışı seçeneklerine sahiptir ve “Immersive Patcher” aracı, eşkenar dörtgen projeksiyona ve kompozisyona olanak tanır.

MediaOut düğümleri

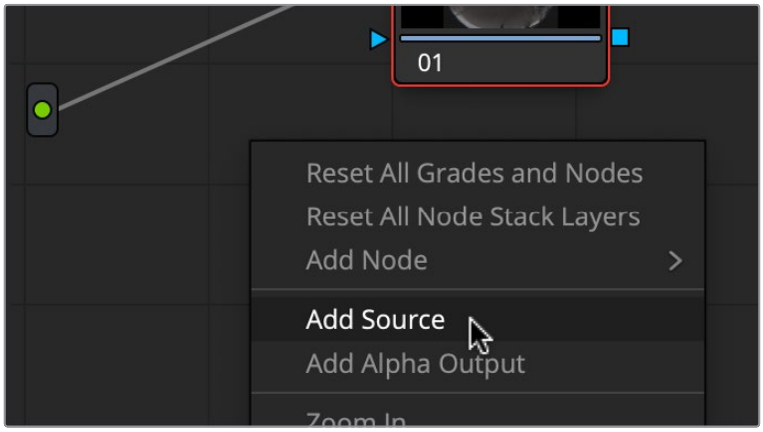
Üç boyutlu bir klip üzerinde çalışırken Fusion sayfasına ilk girdiğinizde, “Nodes” etiketli düğüm grafiği, iki MediaIn düğümüne bağlı iki MediaOut düğümü içerir.



MediaOut Inspector isimli denetleyicide, Stereo Eye parametresi, göz sinyali çıkışını yönlendirmek için kullanılabilir.



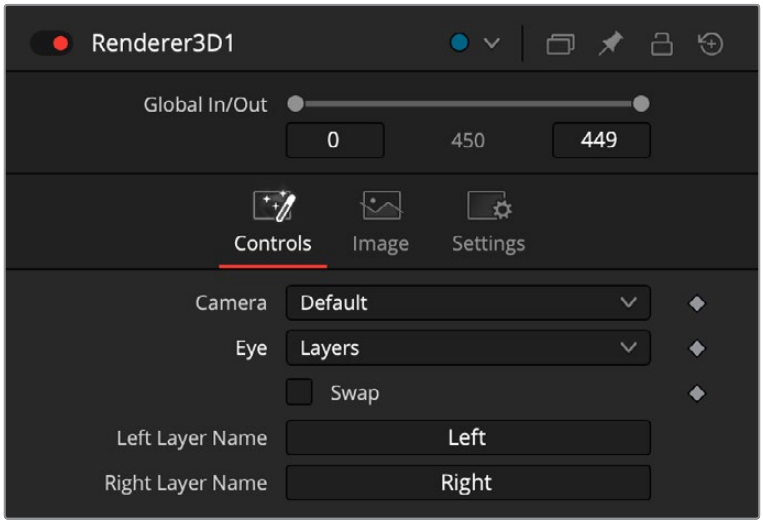
Varsayılan olarak, MediaOut1 Sol göze, MediaOut2 ise Sağ göze ayarlıdır. Bu yapılandırma, stereo uyumlu birleşik görüntüleri ve grafik katmanları hemen oluşturmaya başlamanıza olanak tanır. Çift MediaOut düğümleri ayrıca, görüntüyü daha da geliştirebilmek için color sayfasının düğüm düzenleyicisine sinyalin iki versiyonunun gönderilmesini de sağlar. Color sayfasındaki MediaOut2 sinyaline, düğüm düzenleyicide sağ tıklayıp “Add Source” ibareli kaynak ekleme işlevini seçerek erişebilirsiniz.



“Stereo Eye” ibareli stereo göz parametresi “Auto”, yani otomatik olarak ayarlandığında, MediaOut düğümü sinyalleri birleştirir ve bunları zaman çizelgesindeki uygun gözlere atar. Bu işlem, Fusion'un çok katmanlı iş akışını kullanarak sol ve sağ sinyalleri tek MediaOut düğümüne göndermenizi sağlar.

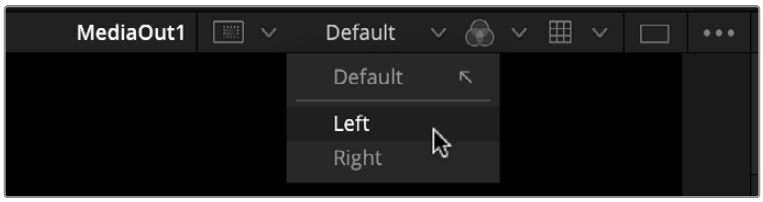
Renderer3D Düğümü

Üç boyutlu bir zaman çizelgesinde 3D grafikler veya bilgisayar tabanlı görüntüler (CGI) ile çalışırken, Renderer3D düğümü, birden fazla göz düzenine ve sol ile sağ katmanları işleme özelliğine sahiptir.



Bu sayede, karmaşık 3D yazılar veya bir CGI sahnesinin tamamını oluşturabilir ve bunları, üç boyutlu projelerde doğrudan stereoskopik bir zaman çizelgesine gönderebilirsiniz.

Bu kurulumla çalışırken, kompozisyonunuzu sol ve sağ göz olarak izlemek arasında geçiş yapmak için Fusion görüntüleyicilerin sağ üst köşesindeki açılır menüyü kullanabilirsiniz.



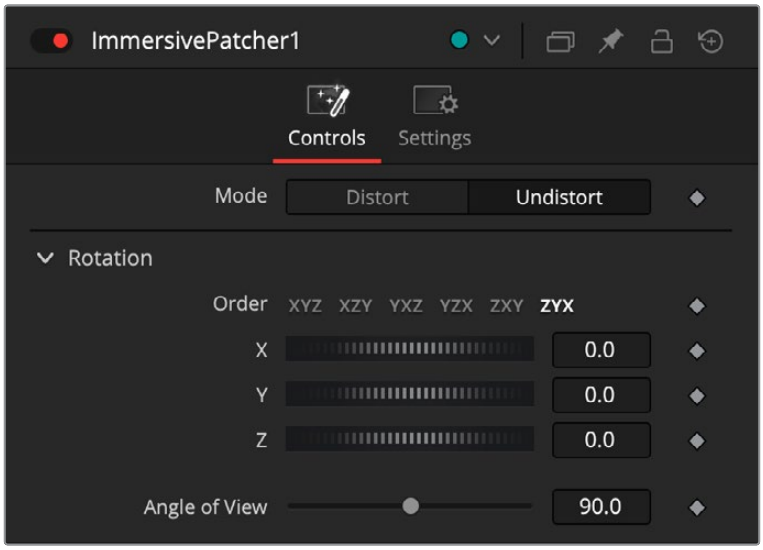
Immersive Patcher ile Görüntü Birleştirme

“Immersive Patcher” isimli üç boyutlu yama aracı, düz bir düzlemde daha kolay takip, boyama ve görüntü birleştirme yapabilmek için, üç boyutlu bir klibin bir bölümünün görüntü bozukluğunu geçici olarak gidermenizi sağlar. İşlem tamamlandığında, üç boyutlu sinyaldeki görüntüyü yeniden çarpıtmak ve yeni birleştirilen ögeleri sahneyle bütünleştirmek için, ikinci bir üç boyutlu yama aracı kullanabilirsiniz.

1 “Immersive Patcher” aracını düğüm iş akışına ekle.



2 Denetleyicide “undistort”, yani görüntü bozukluğunu gider ibaresini tıkla.



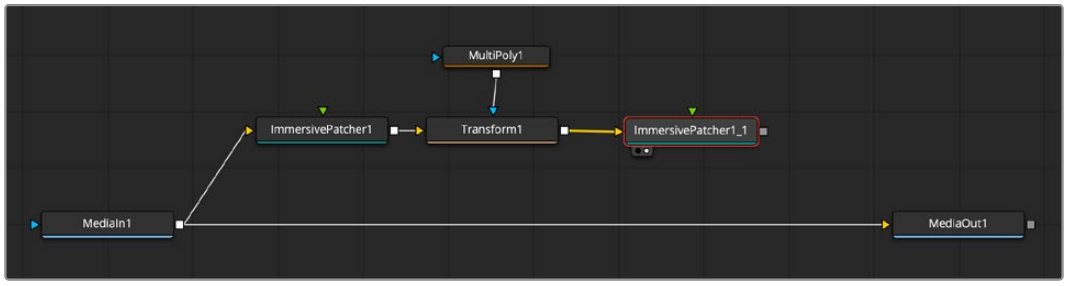
3 Üç boyutlu görüntünün düzleştirmek istediğiniz bir bölümünü seçmek için X, Y ve Z döndürme parametrelerini kullan.

4 Standart bir 2D video sinyalinde olduğu gibi, birleştirme işlemine devam et. Bu, istenmeyen nesneleri çıkarmak için standart boyama yöntemlerini kullanmayı, seti genişletmek için dönüşüm araçlarıyla ve çokgenlerle görüntü karesinin belirli bölümlerini örneklemeyi veya tabelaların ya da yeşil perdelerin yerine farklı bir görüntü yerleştirmek için mat, metin ve harici görüntü/video katmanları uygulamayı içerir (fakat bunlarla sınırlı değildir).

   Bu işlemler tamamlandığında, orijinal çarpıtılmış üç boyutlu görüntüyle eşleşecek şekilde, düz görüntüyü yeniden çarpıtmanız gerekir.

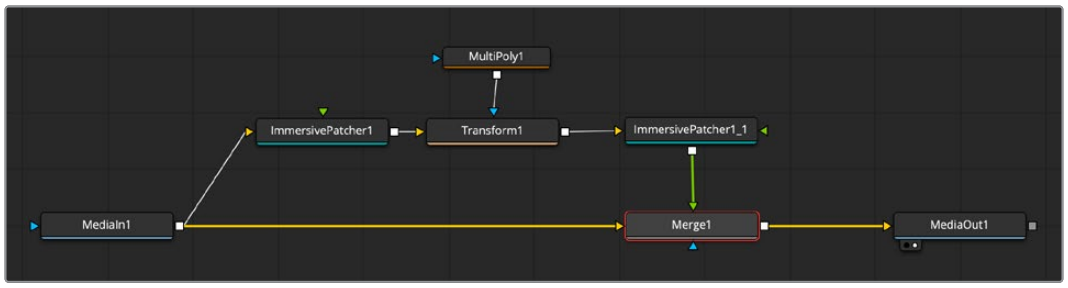
5 Orijinal dönüş ve görüş açısı bilgilerini korumak için “Immersive Patcher” aracını kopyalayın.

6 Birleşik görüntü düğümlerinden sonra “Immersive Patcher”ı yapıştırın ve kompozisyon işlemlerinin sonuna bağlayın.



7 Denetleyicide, “Distort”u tıklayın.

8 Efekti üç boyutlu sahneye entegre etmek için, bindirilmiş yamayı orijinal kliple birleştirin.



Ortak çalışıyorsanız Fusion sayfasından çıkıp, DaVinci Resolve'de üç boyutlu zaman çizelgesini kurgulamaya, miksajını yapmaya ve derecelendirmeye devam edebilirsiniz.

Üç boyutlu çıktılarla çalışıyorsanız, üç boyutlu teslimat için azami özellikleri korumak üzere, her bir gözü ACES AP0'da EXR dosyaları olarak işleyin.

Renk Derecelendirme

Bu bölüm, DaVinci Resolve'un color sayfasında üç boyutlu medya ile çalışırken kullanabileceğiniz optik, renk ve 3D araçları anlatır.

Üç boyutlu içeriğin benzersiz doğası nedeniyle, post prodüksiyon sırasında tüm araçları kullanmak mümkün olmayacaktır. Diğer araçlar şu anda kullanıma hazır değil, fakat üç boyutlu içeriği desteklemek üzere uyarlanmaya başlanmıştır.

Gürültü Azaltma

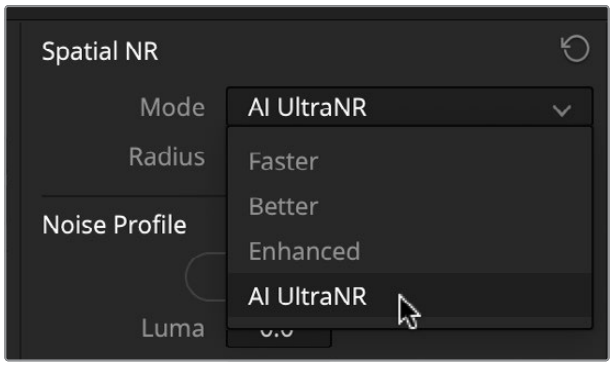
Üç boyutlu videolarla çalışırken gürültü azaltma işlemi mümkündür. “Motion Effects” paletindeki “Spatial NR” aracı, orijinal görüntü detaylarını korurken, gürültüyü en aza indirmek için kaynak medyadaki gürültü düzenini analiz eder.

1 Color sayfası araç çubuğunda, “Motion Effects” etiketli hareket efektleri paletini açın.

“Spatial NR” sütununda, kullanmak istediğiniz gürültü azaltma modunu seçebilirsiniz.

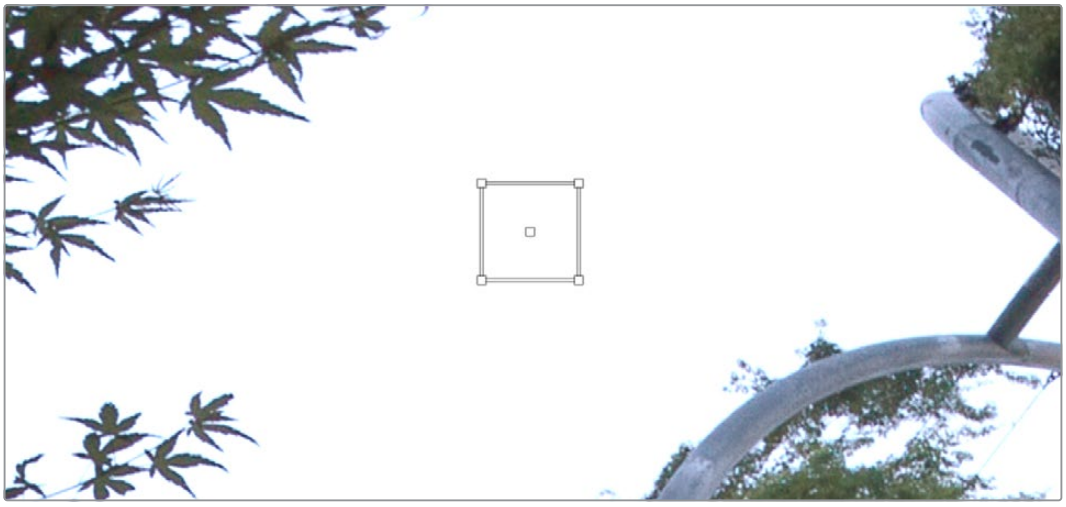
- “Faster” yani daha hızlı, “better” yani daha iyi ve “enhanced” yani gelişmiş modların tümü, belirli bir yarıçap boyutu içerisinde düzen analizini uygular ve çıktılarını, ayarlanan eşiğe göre yapılandırır. İşleme zorluğuna göre listelenmişlerdir.
- *AI UltraNR,* yapay zeka destekli bir gürültü düzeni belirleyicidir ve saf gürültü açısından, görüntü karesindeki en düz bölümü bulup analiz ederek, sinyalden gürültü düzenini çıkarır.

2 Uzaysal gürültü azaltma modunu “AI UltraNR” olarak ayarlayın.



3 “Analyze” ibaresini tıklayın.

İlgili karedeki piksel verisinin "en boş" alanını tespit eden bir sınırlayıcı kutu, görüntüleyicide belirir. Önemli bir görsel içerik olmadığından, kare içindeki veriler kamera sensörünün ürettiği dijital gürültüyü çıkarmak için idealdir.



Dilerseniz, görüntüleyicideki sınırlayıcı kutuyu yeniden konumlandırabilir ve boyutlandırabilirsiniz, fakat bu genellikle tavsiye edilmez. AI UltraNR, en uygun analiz bölgelerini tespit etmede gözlerimizden daha iyidir.

“Noise Profile” etiketli gürültü profili sütununda “Luma” ve Chroma” eşiği ayarlarını manuel olarak da ayarlayabilirsiniz, fakat yine AI UltraNR aracı, temiz bir gürültü azaltma işlemi için ideal eşik karışımını otomatik olarak elde eder.

Birincil Renk Derecelendirme

Birincil renk derecelendirme işlemi, normal video görüntülerinde olduğu gibidir. Ana renk ve log tekerlerini, HDR paleti ve HSL Eğrilerini kullanırken; normalleştirme, dengeleme ve eşleştirme için aynı değerlendirmeleri yapın. “Color Slice” ve “Color Warper” paletleri de normal videoda olduğu gibidir.

İkincil Renk Derecelendirme

Niteleyici gibi piksel tabanlı seçim yöntemleri, üç boyutlu medya ile beklendiği gibi çalışır.

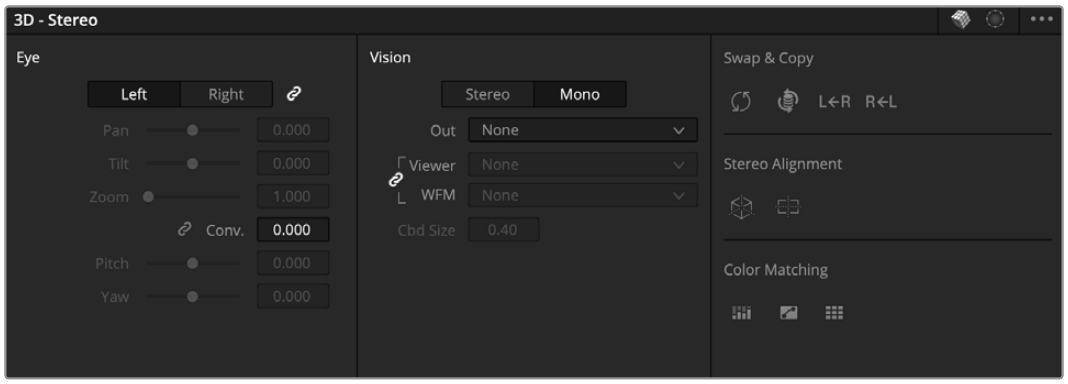
Nesne veya pencere tabanlı seçimler, yalnızca düz eşkenar dörtgen (veya görüntü alanı) ortamında doğru bir şekilde çalışabilir, yani şeklin niteliği açısından hassasiyet gerektiren ikincil yöntemlerin çoğu şu anda kullanılmaya hazır değildir. Bunlara; Power Windows, yani görünmez pencereler (ve dolayısıyla Tracker, yani takipçi paleti) ve Magic Mask isimli sihirli maske aracı da dahildir.

Boyutlandırma

Üç boyutlu içerik için boyutlandırma kontrolleri (yani yakınlaştırma ve panlama) mevcut değildir.

3D Palet

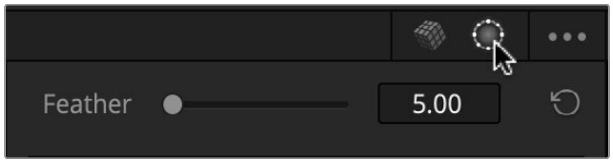
Color sayfasının 3D paletinde, çeşitli 3D kamera ve grafik formatlarını yönetmeye yönelik kontroller bulunur. Diğer stereoskopik formatların aksine, gözlerin hizalanmasını ayarlamanıza gerek yoktur, çünkü Blackmagic URSA Cine Immersive kamera lensleri, odak mesafesi ve diyaframlar sabittir.



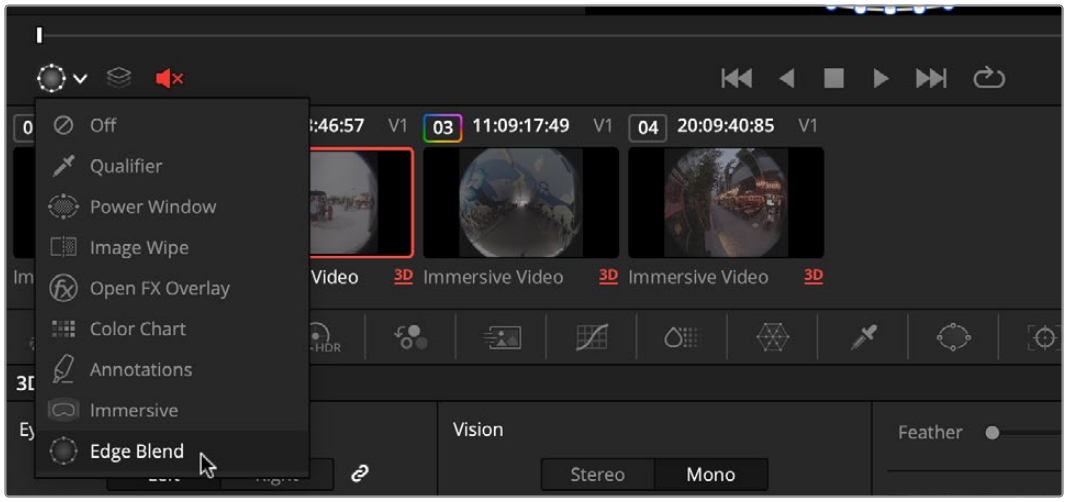
“Edge Blend” isimli kenar harmanlama modu, yalnızca üç boyutlu içeriklere özeldir ve üç boyutlu video karesinin çevresini ayarlamanıza ve yumuşatmanıza olanak tanır. Bu, çekimlerde görünen film ekipmanlarını gizlemek ve izleyicinin görüş alanını sınırlamak için kullanılabilir. Bu kenarın ötesinde, izleyici fonu görecektir.

1 3D paleti aç.

2 Sağ üst köşede modu “Edge Blend”, yani kenar harmanlama olarak değiştir.



3 Görüntüleyicinin ekran kontrollerini Edge Blend olarak ayarlayın.



Video karesinin çevresinde, ortasında bir sabitleme kolu bulunan bir kontrol noktaları çemberi görmelisiniz.

4 Bir kontrol noktasının konumunu değiştirmek için onu sürükleyin.



Veya bir noktayı silmek için sağ tıklayın.

5 3D paletin sağ sütununda, karenin kenarlarını yumuşatmak için “Feather” etiketli tüy parametresini kullanın.

6 Apple Vision Pro'da oynatım esnasında yeni kenar verilerinin doğru şekilde işlenmesini sağlamak için “visionOS Effect”i seçin. Seçilmezse bu kenar, oluşturulan video dosyasına kalıcı olarak işlenir.

7 Sol sütunda, görüntüleyicide gözler arasında geçiş yapmak için “Left” yani sol ve “Right” yani sağ sekmelerini kullanın.

3D paletindeki kontrollerin çoğunun gri renkte ve kullanılamaz olduğunu unutmayın. DaVinci Resolve'deki parametreler, aktif olarak üzerinde çalıştığınız medyaya, yani bu durumda üç boyutlu içeriğe uygulanabilir olmadığında, gri görünür.

8 Orta sütunda bulunan Stereo ve Mono sekmelerini kullanarak, üç boyutlu video klibin tek ve çift lensli görüntüleyicileri arasında geçiş yapabilirsiniz.

CST Renk Yönetimi

Color sayfasının düğüm düzenleyicisindeki düğümlerde Color Space Transform (CST) ibareli renk alanını dönüştürme efektinin kullanılması, doğru renk eşleştirmeye ve zaman çizelgesinin azami çıkış nit parlaklığı üzerinde özel kontrole olanak tanır.

Önerilen iş akışı, dört CST düğümünü içerir:

1. INPUT (GİRİŞ) CST – kameradan çalışma renk alanına.

   Renk düzeltme işlemi, bu iki düğüm arasında gerçekleşir.

2. OUTPUT (ÇIKIŞ) CST – çalışma renk alanından HDR Mac ekranına.
3. MONITOR (MONİTÖR) CST – HDR görüntüleme için 1000 nit ila 250 nit.
4. AVP CST – Apple Vision Pro görüntüleme ve teslimat için 250 nit ila 108 nit.



CST Zaman Çizelgesi Yönlendirme

Önce, kamera kaynağı girişi renk alanından “çalışılan” zaman çizelgesi renk alanına yönlendirme yapacaksınız. Blackmagic URSA Cine Immersive kamera, DaVinci Intermediate log eğrisi ile Blackmagic Design Wide Gamut Gen 4/5 renk gamında çalışır.

1. Efektler panelini açın.
2. Color Space Transform efektini düğüm düzenleyicideki boş bir düğüme (01) sürükleyin.

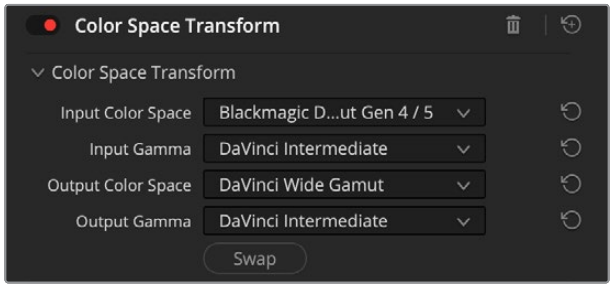
   Giriş renk alanından zaman çizelgesi renk alanına yönlendirme yaparken, herhangi bir derecelendirme işleminden önce bu düğüm, düğüm ağacındaki ilk düğümler arasında olmalıdır.

   Giriş renk uzayından doğrudan izleme veya teslim renk uzayına eşleştirme yaptığınız durumlarda, bu düğüm, derecelendirme işlemlerinden sonra düğüm ağacındaki son düğümler arasında olmalıdır.

3. Düğüm 01’i **INPUT CST** olarak etiketleyin.

> **NOT**: Bu talimatlarda kullanılan düğüm etiketleri, belgelere ve ekran görüntülerine kolayca başvurulabilmesi için belirlenmiştir. Sizin için en uygun adlandırma kuralını kullanabilir veya etiketlemeyi tamamen atlayabilirsiniz.

4. "Input Color Space”yi, Blackmagic Design Wide Gamut Gen 4/5 olarak ayarlayın.
5. “Input Gamma”yı DaVinci Intermediate olarak ayarlayın.
6. “Output Color Space”yi DaVinci Wide Gamut olarak ayarlayın.
7. “Output Gamma”yı DaVinci Intermediate olarak ayarlayın.



8 Yeni bir seri düğüm (02) oluşturun.

9 İlk düğümü (01) yeni düğüm (02) üzerine kopyalayıp yapıştırın.

10 Düğüm 02'yi **OUTPUT CST olarak etiketleyin**.

Düğüm ağacının, zaman çizelgesi renk alanının tutarlı kalması için, çıkış renk alanı ve gamanın bu yeni seri düğümde girişte olması gerekir.

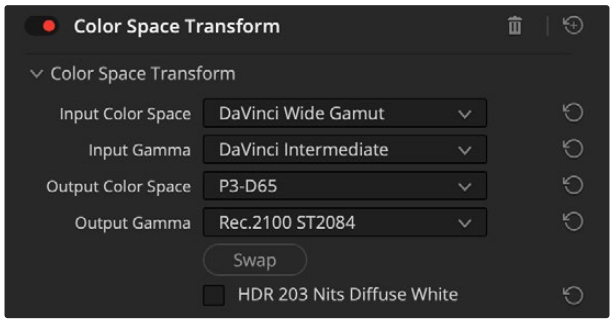
11 Giriş ve çıkış parametrelerini değiştirmek için "Swap" etiketli butona basın.

Giriş Renk Alanı ve Gama artık bu CST düğümü için uygundur.

12 "Output Color Space"yi P3-D65 olarak ayarlayın. Bu, Mac ekranlarının şu anki varsayılan gamıdır.

13 Output Gamma'yı Rec.2100 ST2084 olarak ayarlayın çünkü bu, 1000 nit bir sinyal için HDR standardıdır.

Tüm derecelendirme işlemleri, bu iki düğüm arasında gerçekleşecektir. Fakat düğüm tabanlı renk yönetimiyle çalıştığınız için zaman çizelgesi renk yönetiminin sınırları dışındaki işlemleri gerçekleştirmek için de benzersiz bir imkana sahipsiniz ve birçok renk uzmanı, belirli derecelendirme işlemlerini, efektleri veya üçüncü parti ögeleri uygulamak için bundan yararlanmaktadır.



ICC Profilleri

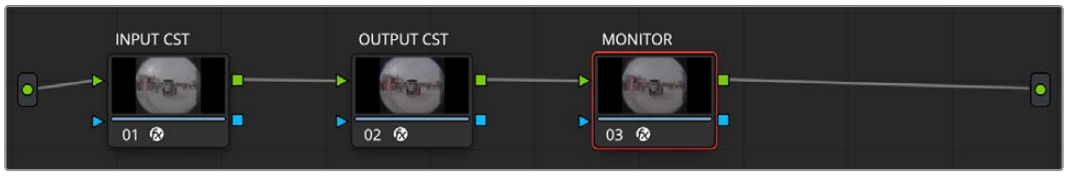
Blackmagic URSA Cine Immersive sinyalinin Mac ekranınızla doğru şekilde eşleştirildiğinden emin olmak için, Sistem Ayarları > Ekranlar bölümüne gidin ve renk profili Ön Ayarının Apple XDR Display (P3-1600 nit) olarak ayarlandığını doğrulayın.

CST Renk Derecelendirme Monitörü Eşleştirme

Renk derecelendirme esnasında görüntüleme için maksimum nit çıkış değerlerini geçici olarak sınırlamak üzere ek CST'ler kullanabilirsiniz.

1 Renk yönetimi OUTPUT CST (02)'den sonra yeni bir seri düğüm (03) oluşturun.

2 Düğüm 03'ü **MONITOR** olarak etiketleyin.



3 “Color Space Transform” efektini MONITOR düğüm (03)’e ekleyin.

Giriş değerlerinin bir önceki CST'den gelen çıkış sinyaline uyması gerekir.

4 “Input Color Space”yi P3-D65 olarak ayarlayın.

5 “Input Gamma” değerini Rec.2100 ST2084 olarak ayarlayın.

Bu CST düğümünün amacı sadece görüntünün parlaklığını daha küçük bir nit değeriyle eşleştirmektir, böylece giriş ve çıkış parametreleri aynı kalabilir.

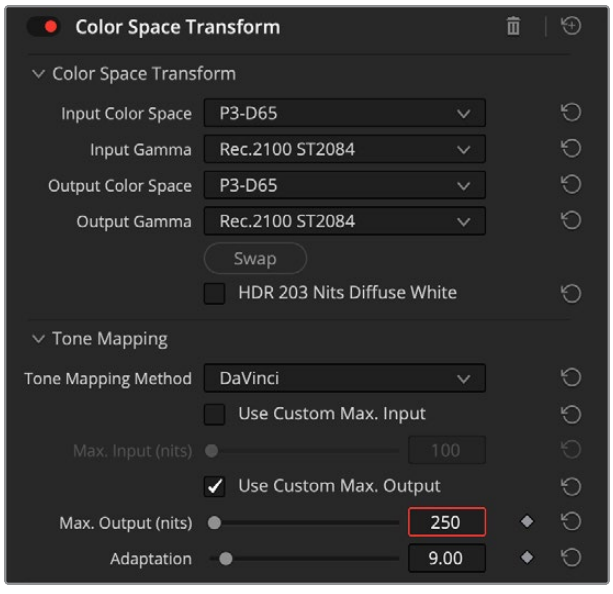
6 “Output Color Space”yi P3-D65 olarak ayarlayın.

7 “Output Gamma” değerini Rec.2100 ST2084 olarak ayarlayın.

NOT: Bazı durumlarda, giriş ve çıkış aynı ise (eşleştirme işlemi gerekmeyeceğinden) bir CST'nin renk alanını veya gama değerini ayarlamak gerekli değildir. Bu durumda bu zorunludur, çünkü özel nit eşleştirme işlemi, sinyalin log eğrisinin değerinden ve şeklinden etkilenir.

8 “Use Custom Max. Output” ibareli özel maksimum çıkış değerini kullan seçeneğini belirleyin.

9 “Max Ouput” (nits) alanını **250**’ye ayarlayın.



1000 nit sinyal artık (DaVinci Tone Mapping Method, yani ton eşleştirme yöntemi kullanılarak) 250 nit’e eşleştirilecektir.

Bu, bir HDR renk derecelendirme monitöründe (örneğin Flanders XMP310, Sony BVM-HX310, vb.) işlenen Apple Vision Pro sinyaline yakın bir görüntü sağlar.

Ancak, projenizi Apple Vision Pro'da görüntülemek için bir ton eşleştirme işlemi daha uygulamanız gerekecektir.

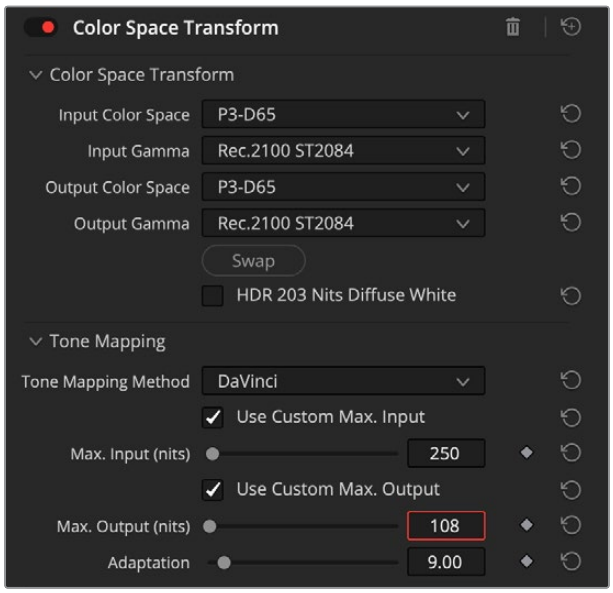
CST Apple Vision Pro'da Görüntüleme

Canlı görüntüleme, önizleme ve Apple Vision Pro cihazlarına son teslimat için son bir CST düğümü kullanılabilir.

1. MONITOR düğüm (03)'ten sonra yeni bir seri düğüm (04) oluşturun.
2. MONITOR düğüm (03)'ü kopyalayıp 04 düğümü üzerine yapıştırın. Aynı giriş ve çıkış parametrelerini kullanacaksınız ve sadece nit sayısını değiştireceksiniz.
3. Düğüm 04'ü **AVP olarak etiketleyin**.



4. "Use Custom Max. Input" ibareli özel maksimum giriş değerini kullan seçeneğini belirleyin.
5. MONITOR düğümünden gelen eşleştirilmiş sinyali doğru bir şekilde yorumlamak için, "Max. Input" (nits) alanını **250** olarak ayarlayın.
6. "Max. Output" (nits) alanını **108**'e ayarlayın, bu, Apple Vision Pro ile görüntüleme ve teslimat için maksimum nit değeridir.



Artık renk derecelendirme işlemine devam edebilir ve iş akışınıza bağlı olarak, gerekirse CST düğümlerini devre dışı bırakabilirsiniz. Örneğin, harici renk derecelendirme monitörünüzü kullanacağınız ilk derecelendirme çıktınız esnasında, HDR renk derecelendirme monitörünüzde 250 nit sinyali doğru şekilde gördüğünüzden emin olmak için AVP düğüm (04)'ü devre dışı bırakmalısınız. Apple Vision Pro'da önizlemeye geçerken veya son teslimat için, AVP düğüm (04)'ü yeniden etkinleştirin.

CST'ler, zaman çizelgesi düğüm düzenleyiciye yerleştirilebilir mi?

Hayır. Şu anda, klip ton eşleştirme işlemi, "Color Space Adaptation" ibareli renk alanı dönüştürme adaptasyonu parametresinin doğru uygulanmasına dayalıdır ve bunun klip bazında gerçekleşmesi gerekir.

Ses

Apple Vision Pro ve DaVinci Resolve; uzamsal ve ambisonik ses, AVP (Apple Vision Pro) ses ve AIA (Apple Immersive Audio) formatını destekler.

Uzamsal sesle çalışırken, ses post prodüksiyonunun birçok yönü değiştirilmemiştir. En büyük farklardan biri ise, uzamsal seste gerçek bir üç boyutlu deneyim ve izleyicinin dinleme ortamını daha iyi yansıtmak için, ses mühendislerinin üst düzey hoparlörlerle dinlemek yerine, kulaklık veya Airpods kullanmalarının gerektiğidir.

Oysa 2D miksajlarda, diyalogların büyük bir bölümü merkez kanallara miksajı yapılırken, uzamsal ses ise sesin kaynaklandığı alana yerleştirilmelidir. Ses kanal gruplarının miksajında; ses kısma, eşitleme ve dengeleme işlemleri dinleyiciyi rahatsız etmeyecek veya kafasını karıştırmayacak şekilde uygulanmalıdır.

Üç boyutlu uzamsal ögeler, normal bir stereo ses kanal grubuyla birleştirilebilir, böylece oynatım çubuğu nereye hareket ettirilirse ettirilsin belirli seslerin sabit kalması sağlanır. Bu, müzik notaları ve seslendirmeler gibi (diegetik olmayan) anlatı seslerin, net ve kolay yorumlanabilir olmasında faydalıdır.

Uzamsal ses bağlamında, çevre sesleri daha kısık sesli yansır, bu nedenle duyulabilmeleri için miksajının genellikle daha baskın bir şekilde yapılması gerekir. Ancak bu, dinleyicinin kafasını karıştırma pahasına yapılmamalıdır.

Yankılama, dinleyicilerin her gün doğal olarak deneyimlediği bir efekt olduğundan, üç boyutlu sese büyük katkı sağlar. Belirli bir ortamda yer alan her ses, dinleyiciyi çevreleyen yüzeylerden yansır ve "yükselerek ilerleyen" bir yankı kalitesi elde edilir. Dolayısıyla, tam üç boyutlu bir ses elde etmek amacıyla uzamsal ses tasarımında yankılanma teşvik edilir.

Apple Immersive Audio (AIA) formatı, çeşitli formatları kapsar:

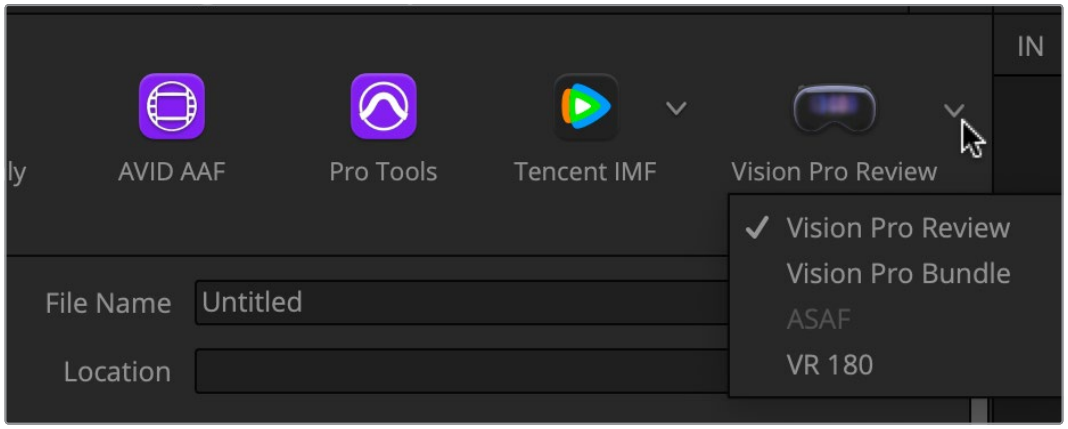
- **Apple Spatial Audio Formatı (ASAF),** bir ham ses formatıdır.
- **Apple Positional Audio Codec (APAC),** Airpods gibi APAC destekli cihazlarda veya Apple Vision Pro'daki bir AIVU dosyasında oynatılabilen kodlanmış .mp4 formatıdır.

Apple Spatial Audio Format (ASAF)

Apple Spatial Audio Format (ASAF), Apple Vision Pro'da esasen VR ve AR uygulamaları için geliştirilen üç boyutlu bir ses formatıdır. Sadece macOS'a özel olan ASAF, (eklentiler aracılığıyla) ProTools ve (yerel olarak) DaVinci Resolve'de desteklenir. ASAF, (Scenes olarak adlandırılan) Ambisonik kaynaklar için yerleşik destek ve bus kanal atamaları aracılığıyla yönlendirme dahil olmak üzere gelişmiş teknik kapasiteye sahiptir. Miksaj, fiziksel bir bus kanalı olmadan sanal olarak gerçekleşir ve kullanıcılar doğrudan Apple AirPod entegrasyonu aracılığıyla baş takipli stereofonik ses dahil olmak üzere, çeşitli ses dinleme formatları arasından seçim yapabilir. Ses yerleştirme işlemleri, DaVinci Resolve'un mevcut araçları olan diyalog yerleştirme, otomasyon, SpaceView ve görüntüleyicideki ses yerleştirme araçları ile denetleyicideki ikincil kontroller kullanılarak yapılır. Enlem-boylam video bilgilerine dayalı otomatik takip de dahil olmak üzere bu araçlar, kaynakların hassas uzamsal yerleşimine olanak tanır. “Immersive Audio > Export Master File” seçeneğini kullanarak, ASAF miksajlarının WAV veya MP4 formatlarında (ilgili metaverileriyle birlikte) çıktısı alınabilir, fakat bu formatlar şu anda DaVinci Resolve'a yüklenememekte veya oynatılamamaktadır.

Teslimat

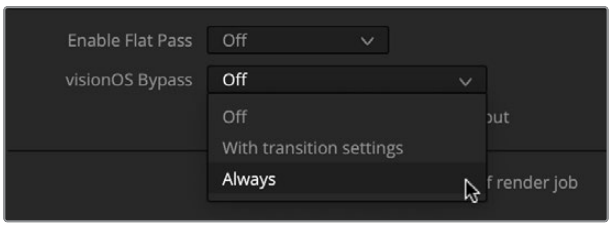
Teslimat sayfasındaki iki çıktı ayarı, inceleme ve son teslimat için Apple Vision Pro dosyalarının çıktısını almayı mümkün kılar.



- **Vision Pro Review**, önizleme amaçlı küçük bir MV-HEVC paketi oluşturur. Apple, oluşturulan dosyayı Wi-Fi aracılığıyla Apple Vision Pro ile “paylaşır” ve bu dosya, App Store'dan indirilebilen Apple Immersive Video Utility (AIVU) uygulamasında izlenebilir.
- **Vision Pro Bundle** – arşivlenebilen veya Apple TV'de oynatılmaya hazır hale getirmek üzere Apple Compressor'da işlenebilen büyük bir ProRes dosya paketi oluşturur.
- **ASAF** – WAV veya MP4 formatında ASAF ses miksajı oluşturur.
- **VR180** – VR180 uyumlu stereoskopik video dosyası oluşturur.

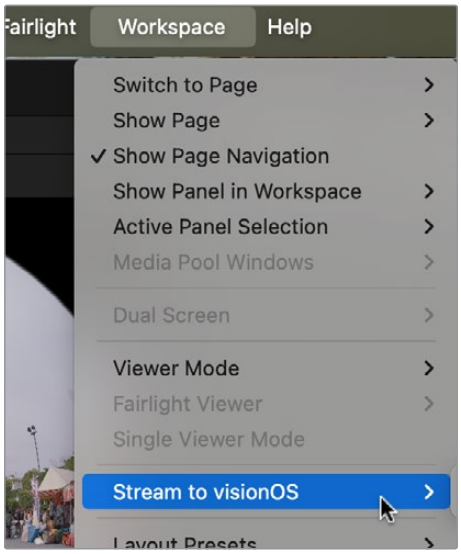
Geçişleri Atlama

Üç boyutlu bir zaman çizelgesine bir geçiş eklenirse, bu geçiş önceden işlenir ve tek oynatım kanalına dönüştürülür. Üç boyutlu içerik için oluşturulan XML dosyası, bu işlenmiş sinyal akışları (önce görüntü, sonra önceden işlenmiş geçiş, sonra tekrar görüntü) arasında geçiş yapma sürecini tanımlar. “Render Settings” ibareli çıktı ayarları panelinin “Advanced Settings” ibareli gelişmiş ayarlar bölümünde; visionOS geçişlerini atlat (“Always”, yani her zaman), bunları zaman çizelgesindeki durumlarına göre uygula (“With transition settings”, yani geçiş ayarları ile) veya zaman çizelgesinde göründükleri gibi dahil et (“Off”, yani kapalı) seçenekleri bulunur.



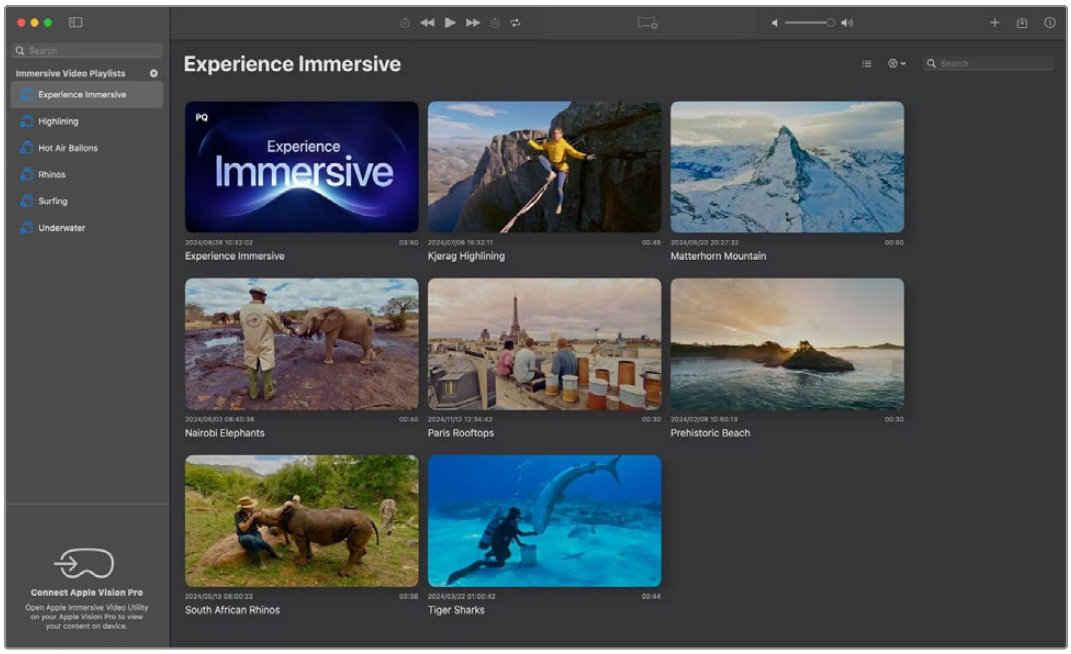
Apple Vision Pro'da İnternet Yayını

DaVinci Resolve yazılımında “edit”, “color” ve “deliver” sayfalarından Apple Vision Pro'ya internetten yayın yapmak için, “Workspace” menüsüne gidip “Stream to visionOS” ibareli visonOS’e yayın yapma seçeneğini belirleyip, AVP cihazınızı seçmeniz yeterlidir. Sorunsuz internet yayını için hızlı bir kablosuz ağ (en az Wi-Fi 5; Wi-Fi 6 tercih edilir) önerilir.



Apple Immersive Video Utility Yardımcı Yazılım Hakkında

Apple Immersive Video Utility, Apple Immersive Video içeriğinin oluşturulmasını, yönetilmesini ve oynatılmasını kolaylaştırmak için tasarlanmış özel bir macOS ve visionOS uygulamasıdır. Profesyonel film yapımcıları ve üç boyutlu medya oluşturucuları için özel olarak geliştirilen bu uygulama, kullanıcıların 180 derecelik görüş alanı ve “Spatial Audio” olarak bilinen uzamsal ses ile 8K stereoskopik video dosyalarını yüklemelerine, düzenlemelerine, paketlemelerine ve incelemelerine olanak tanır. AIVU (Apple Immersive Video Universal) dosyalarının oluşturulmasını destekleyerek, post prodüksiyon iş akışları için metaverilerin incelenmesini ve değiştirilmesini sağlar. Kullanıcılar oynatım listeleri oluşturabilir, belgelikler kapsamında sıralama ve arama yapabilir ve hatta, içeriği doğrudan bir veya daha fazla Apple Vision Pro cihazında yayınlayabilirler.



Developer Information

Camera Control REST API

If you are a software or hardware developer you can build custom applications or leverage ready to use tools such as REST client or Postman to seamlessly control and interact with your compatible Blackmagic camera using Camera Control REST API. This API enables you to perform a wide range of operations, such as starting or stopping recordings, accessing disk information and much more. Whether you're developing a custom application tailored to your specific needs or utilizing existing tools, this API empowers you to unlock the full potential of your Blackmagic camera with ease. We look forward to seeing what you come up with!

NOTE It's important to mention that controlling Blackmagic cameras via REST API relies on the web manager being enabled on each compatible Blackmagic camera. Enable the web media manager in the Blackmagic Camera Setup 'network access' settings for each camera you are controlling.

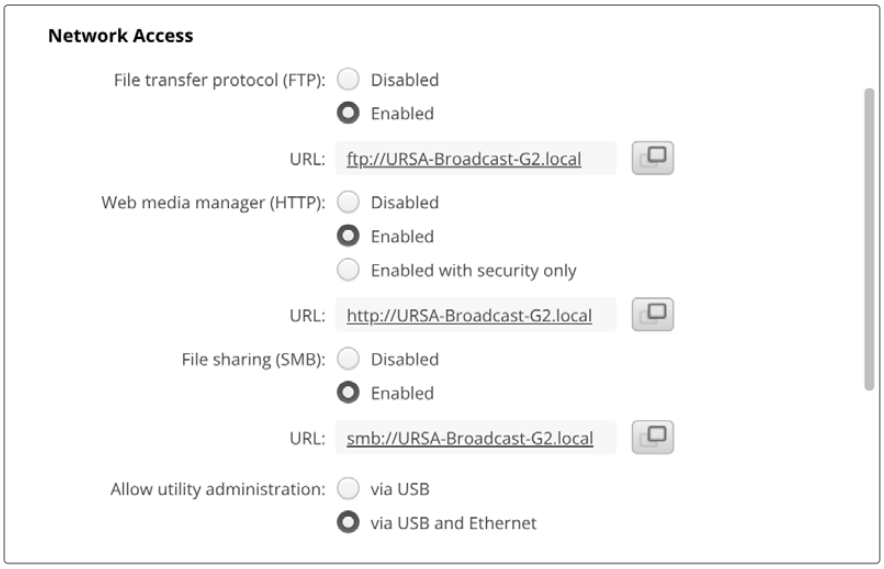
The following Blackmagic cameras are compatible with Camera Control REST API:

- Blackmagic URSA Cine 12K LF
- Blackmagic URSA Cine 17K 65
- Blackmagic URSA Cine Immersive
- Blackmagic PYXIS 6K
- Blackmagic Cinema Camera 6K
- Blackmagic URSA Broadcast G2
- Blackmagic Micro Studio Camera 4K G2
- Blackmagic Studio Camera 4K Plus
- Blackmagic Studio Camera 4K Pro
- Blackmagic Studio Camera 6K Pro
- Blackmagic Studio Camera 4K Plus G2
- Blackmagic Studio Camera 4K Pro G2

Sending API Commands

To send an API command to your camera from a third party application such as Postman, add /control/api/v1/ to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/api/v1/

You can find the Web media manager URL and IP address information in Blackmagic Camera Setup.



The Web media manager URL in Blackmagic Camera Setup

Downloading API's from your Camera

You can download REST API YAML documentation from your camera by adding /control/documentation.html to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/documentation.html

NOTE It's worth noting that changing the camera name in Blackmagic Camera Setup will also change the camera's Web media manager URL.

Livestream Control API

API for controlling Livestreams on Blackmagic Design products.

GET /livestreams/0

Get the livestream's current status.

Response

200 - Livestream's current status.

The response is JSON.

Name	Type	Description
status (required)	string	Possible values are: Idle, Connecting, Streaming, Flushing, Interrupted.
bitrate (required)	integer	Current bitrate (bps).
effectiveVideoFormat (required)	string	Effective video format for the livestream, serialised as a string.
duration	integer	Current stream duration in seconds. Absent if livestream is idle.
cache	integer	Current stream cache usage percentage.

GET /livestreams/0/start

Determine if the livestream is active.

Response

200 - Livestream active status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is active.

PUT /livestreams/0/start

Start the livestream.

Response

204 - Livestream started.

GET /livestreams/0/stop

Determine if the livestream is inactive.

Response

200 - Livestream inactive status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is inactive.

PUT /livestreams/0/stop

Stop the livestream.

Response

204 - Livestream stopped.

GET /livestreams/0/activePlatform

Get the currently selected platform configuration for the livestream.

Response

200 - Livestream active platform configuration.

The response is JSON.

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

PUT /livestreams/0/activePlatform

Set the currently selected platform configuration for the livestream.

Parameters

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

Response

204 - Livestream active platform configuration updated.

400 - Bad Request

GET /livestreams/platforms

Get the list of available platforms.

Response

200 - List of available platforms.

The response is JSON.

Name	Type	Description
	array	List of available platforms names.
[i]	string	Platform name.

GET /livestreams/platforms/{platformName}

Get the service configuration for a platform.

Parameters

Name	Type	Description
{platformName} (required)	string	Name of the platform.

Response

200 - Service configuration for specified platform.

The response is JSON.

Livestream platform service configuration.

Name	Type	Description
platform (required)	string	Corresponding platform name.
key	string	Default stream key.
servers (required)	array	List of server configurations.
servers[i]	object	Server configuration.
servers[i].server (required)	string	Server name.
servers[i].url (required)	string	Livestream destination.
servers[i].srtExtensions	array	Miscellaneous tags used for SRT livestreams.
servers[i].srtExtensions[i]	object	Dictionary object mapping SRT tag strings to values.
servers[i].srtExtensions[i][{key}]	string	SRT tag value.
servers[i].group	string	Logical grouping of the server.
profiles (required)	array	List of profile configurations.
profiles[i]	object	Quality configuration.
profiles[i].profile (required)	string	Quality level name.
profiles[i].configs (required)	array	List of video format configurations.
profiles[i].configs[i]	object	Video format configuration for profiles.
profiles[i].configs[i].resolution (required)	string	Video format serialised as a string.
profiles[i].configs[i].fps (required)	string	Frames per second.
profiles[i].configs[i].bitrate (required)	integer	Pixel bitrate (bps).
profiles[i].configs[i].audioBitrate	integer	Audio bitrate (bps).
profiles[i].configs[i].keyFrameInterval	integer	How often a key frame is sent, in seconds.
profiles[i].configs[i].videoCodecs	array	Supported video encoding algorithm/s.

Name	Type	Description
profiles[i].configs[i].videoCodecs[i]	string	Video encoding algorithm. Possible values are: H264, H265.
profiles[i].lowLatency (required)	boolean	If true, fewer frames will be buffered in the livestream.
defaultProfile	string	Quality level name.
credentials	object	Credientials used for RTMP streams.
credentials.username (required)	string	The username part of the creditials. Only used for RTMP streams.
credentials.password (required)	string	Used for RTMP streams, also used as Passphrase for SRT streams.
customizableUrlEnabled	boolean	True when the server URL is customizable.

400 - Bad Request

GET /livestreams/customPlatforms

Get a list of custom platform files.

Response

200 - List of custom platform files.

The response is JSON.

Name	Type	Description
	array	List of custom platform file names.
[i]	string	Custom platform file name.

DELETE /livestreams/customPlatforms

Remove all custom configuration files.

Response

204 - All custom configuration files removed.

GET /livestreams/customPlatforms/{filename}

Get a custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to get.

Response

200 - Custom platform file.

The response is XML.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

404 - Not Found

PUT /livestreams/customPlatforms/{filename}

Update a custom platform file if it exists, if not, create a new file with the given file name.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to update/create.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

Response

204 - Custom platform file created or updated.

400 - Bad Request

DELETE /livestreams/customPlatforms/{filename}

Remove the given custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to be removed.

Response

204 - Custom platform file removed.

404 - Not Found

Clips Control API

API for listing clips on disk.

GET /clips

Get the list of clips on the active disk.

Response

200 - List of clips on the active disk.

The response is JSON.

List of media clips.

Name	Type	Description
clips (required)	array	
clips[i]	object	Media clip.
clips[i].clipUniqueId (required)	integer	Unique ID used to identify this clip.
clips[i].filePath	string	Path to the file relative to the root of a mount.
clips[i].fileSize	integer	Size of file on disk in bytes.
clips[i].codecFormat	object	
clips[i].codecFormat.codec	string	Currently selected codec.

Name	Type	Description
clips[i].codecFormat.container	string	Multimedia container format.
clips[i].videoFormat	object	Video format configuration.
clips[i].videoFormat.name (required)	string	Video format serialised as a string.
clips[i].videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
clips[i].videoFormat.height	number	Height dimension of video format.
clips[i].videoFormat.width	number	Width dimension of video format.
clips[i].videoFormat.interlaced	boolean	Is the display format interlaced?
clips[i].startTimecode	string	Start timecode of the clip serialised as string.
clips[i].durationTimecode	string	Duration of the clip in timecode format serialised as string.
clips[i].frameCount	integer	Number of frames in clip; duration of the clip in frames.

404 - There is no active disk.

Media Pool Control API

API to manage media pool and handle uploads and project data.

GET /cloud/projects

List all projects within the media pool.

Response

200 - Successfully retrieved the list of all projects.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].libraryID	string	
[i].name	string	
[i].private	boolean	
[i].shared	boolean	
[i].clips	array	List of clips associated with the project.
[i].clips[i]	string	
[i].status	object	
[i].status.numClipsRequested	integer	
[i].status.numClipsComplete	integer	
[i].status.uploadPercent	integer	Percentage of upload completion.
[i].status.numClipsPaused	integer	
[i].status.outOfSpace	boolean	
[i].status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
[i].status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/active

Retrieve data of the actively uploading project.

Response

200 - Successfully retrieved the active project's data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/{projectID}

Retrieve specific project data by project ID.

Parameters

Name	Type	Description
{projectID} (required)	integer	Unique identifier of the project.

Response

200 - Successfully retrieved the project data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	

Name	Type	Description
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

404 - Project not found.

GET /cloud/clips

List all clips within the media pool.

Response

200 - Successfully retrieved the list of all clips.

The response is JSON.

Name	Type	Description
	array	
[i]	string	

GET /cloud/clips/activeUploading

Retrieve data of actively uploading clips.

Response

200 - Successfully retrieved the list of actively uploading clips.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].path	string	
[i].projectID	integer	
[i].status	object	
[i].status.projectID	integer	
[i].status.outOfSpace	boolean	
[i].status.proxyExtension	string	
[i].status.growingFile	boolean	
[i].status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.originalClipTotalSize	integer	
[i].status.proxyClipTotalSize	integer	
[i].status.originalClipCompletedSize	integer	
[i].status.proxyClipCompletedSize	integer	
[i].status.secsRemaining	integer	

GET /cloud/clips/{deviceName}/{path}

Retrieve specific clip data by device and path.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored.
{path} (required)	string	Path to the clip.

Response

200 - Successfully retrieved the clip data.

The response is JSON.

Name	Type	Description
path	string	
projectID	integer	
status	object	
status.projectID	integer	
status.outOfSpace	boolean	
status.proxyExtension	string	
status.growingFile	boolean	
status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.originalClipTotalSize	integer	
status.proxyClipTotalSize	integer	
status.originalClipCompletedSize	integer	
status.proxyClipCompletedSize	integer	
status.secsRemaining	integer	

404 - Clip not found.

Monitoring Control API

API for monitoring and controlling display settings in video equipment.

GET /monitoring/display

Retrieve a list of all display names.

Response

200 - Returns a list of display names.

The response is JSON.

Name	Type	Description
displays	array	List of display names available.
displays[i]	string	

GET /monitoring/{displayName}/cleanFeed

Get the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/cleanFeed

Set the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Clean feed enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/displayLUT

Get the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

400 - Invalid display name.

404 - Display name not found.

PUT /monitoring/{displayName}/displayLUT

Set the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Display LUT enabled/disabled successfully.

400 - Invalid input.

422 - Unprocessable Entity - Unable to process the contained instructions.

GET /monitoring/{displayName}/zebra

Get the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/zebra

Set the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Zebra enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/focusAssist

Get the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/focusAssist

Set the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/focusAssist

Get the focus assist settings.

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

404 - Display name not found.

PUT /monitoring/focusAssist

Set the focus assist settings.

Parameters

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/frameGuide

Get the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame guide enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGuide

Set the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame guide state updated successfully.

422 - Unable to update the frame guide state.

GET /monitoring/frameGuideRatio

Get the current frame guide ratio.

Response

200 - Returns the current frame guide ratio.

The response is JSON.

Name	Type	Description
ratio	string	The frame guide ratio.

PUT /monitoring/frameGuideRatio

Set the frame guide ratio.

Parameters

Name	Type	Description
ratio	string	The frame guide ratio.

Response

204 - Frame guide ratio updated successfully.

422 - Unable to update the frame guide ratio.

GET /monitoring/frameGuideRatio/presets

Get the presets for frame guide ratios.

Response

200 - Returns a list of preset frame guide ratios.

The response is JSON.

Name	Type	Description
presets	array	
presets[i]	string	A frame guide ratio.

GET /monitoring/{displayName}/frameGrids

Get the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame grids enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGrids

Set the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame grids state updated successfully.

422 - Unable to update the frame grids state.

GET /monitoring/frameGrids

Get the global frame grids settings.

Response

200 - Returns the current frame grids settings.

The response is JSON.

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

PUT /monitoring/frameGrids

Set the global frame grids settings.

Parameters

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

Response

204 - Frame grids settings updated successfully.

400 - Invalid input, check the number of frame grids or values.

422 - Unable to update the frame grids settings.

GET /monitoring/{displayName}/safeArea

Get the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the safe area enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/safeArea

Set the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Safe area state updated successfully.

422 - Unable to update the safe area state.

GET /monitoring/safeAreaPercent

Get the current safe area percentage.

Response

200 - Returns the current safe area percentage.

The response is JSON.

Name	Type	Description
percent	integer	Safe area coverage percentage.

PUT /monitoring/safeAreaPercent

Set the safe area percentage.

Parameters

Name	Type	Description
percent	integer	Safe area coverage percentage to set.

Response

204 - Safe area percentage updated successfully.

400 - Invalid percentage value.

422 - Unable to update the safe area percentage.

GET /monitoring/{displayName}/falseColor

Get the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the false color enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/falseColor

Set the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - False color state updated successfully.

422 - Unable to update the false color state.

Event Control API

API For working with built-in websocket.

GET /event/list

Get the list of events that can be subscribed to using the websocket API.

Response

200 - Websocket API events list.

The response is JSON.

Name	Type	Description
events	array	List of events that can be subscribed to using the websocket API.
events[i]	string	

System Control API

API for controlling the System Modes on Blackmagic Design products.

GET /system

Get device system information.

Response

200 - System summary.

The response is JSON.

The properties will be populated only with the values that are supported/implemented by the device in use.

Name	Type	Description
codecFormat	object	Codec format configuration.
codecFormat.codec	string	Codec serialised as string.
codecFormat.container	string	Multimedia container format.
videoFormat	object	Video format configuration.
videoFormat.name (required)	string	Video format serialised as a string.
videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
videoFormat.height	number	Height dimension of video format.
videoFormat.width	number	Width dimension of video format.
videoFormat.interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/product

Get device product information.

Response

200 - Device product information.

The response is JSON.

Product information.

Name	Type	Description
deviceName	string	Name of device as displayed in Setup.
productName	string	Device's product name.
softwareVersion	string	Software version running on device.

501 - This functionality is not implemented for the device in use.

GET /system/supportedCodecFormats

Get the list of supported codecs.

Response

200 - List of supported codec formats.

The response is JSON.

Name	Type	Description
codecs	array	
codecs[i]	object	Codec format configuration.
codecs[i].codec	string	Codec serialised as string.
codecs[i].container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

GET /system/codecFormat

Get the currently selected codec.

Response

200 - Current codec format.

The response is JSON.

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

PUT /system/codecFormat

Update the system codec.

Parameters

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

Response

204 - The codec updated successfully.

400 - The specified codec format is unsupported.

501 - This functionality is not implemented for the device in use.

GET /system/videoFormat

Get the currently selected video format.

Response

200 - Current system video format.

The response is JSON.

Video format configuration.

Name	Type	Description
name (required)	string	Video format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height	number	Height dimension of video format.
width	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

PUT /system/videoFormat

Set the system video format.

Parameters

This parameter can be one of the following types:

Name	Type	Description
name (required)	string	Video format serialised as a string.

Name	Type	Description
frameRate (required)	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height (required)	number	Height dimension of video format.
width (required)	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

Response

204 - The video format updated successfully.

400 - Invalid request.

409 - Operation unsupported in the current state.

501 - This functionality is not implemented for the device in use.

GET /system/supportedVideoFormats

Get the list of supported video formats for the current system state.

Response

200 - List of supported video formats.

The response is JSON.

List of supported video formats.

Name	Type	Description
formats	array	List of video formats.
formats[i]	object	Video format configuration.
formats[i].name (required)	string	Video format serialised as a string.
formats[i].frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
formats[i].height	number	Height dimension of video format.
formats[i].width	number	Width dimension of video format.
formats[i].interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/supportedFormats

Get supported formats.

Response

200 - List of supported formats.

The response is JSON.

Name	Type	Description
supportedFormats	array	
supportedFormats[i]	object	
supportedFormats[i].codecs	array	
supportedFormats[i].codecs[i]	string	
supportedFormats[i].frameRates	array	
supportedFormats[i].frameRates[i]	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
supportedFormats[i]. maxOffSpeedFrameRate	number	
supportedFormats[i]. minOffSpeedFrameRate	number	
supportedFormats[i]. recordResolution	object	
supportedFormats[i]. recordResolution.height	number	Height of the resolution.
supportedFormats[i]. recordResolution.width	number	Width of the resolution.
supportedFormats[i]. sensorResolution	object	
supportedFormats[i]. sensorResolution.height	number	Height of the resolution.
supportedFormats[i]. sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

GET /system/format

Get current format.

Response

200 - Current format.

The response is JSON.

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

PUT /system/format

Set the format.

Parameters

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

Response

204 - System format updated.

501 - This functionality is not implemented for the device in use.

Transport Control API

API for controlling Transport on Blackmagic Design products.

GET /transports/0

Get device's basic transport status.

Response

200 - Transport status.

The response is JSON.

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

PUT /transports/0

Set device's basic transport status.

Parameters

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, Output.

Response

204 - Transport mode was set.

400 - Failed to set transport mode.

GET /transports/0/stop

Determine if transport is stopped.

Response

200 - Transport stop response.

The response is JSON.

Name	Type	Description
	boolean	True when transport mode is InputPreview or when in Output mode and speed is 0.

PUT /transports/0/stop

Stop transport. Deprecated, use POST /transports/0/stop instead.

Response

204 - Transport stopped.

POST /transports/0/stop

Stop transport.

Response

204 - Transport stopped.

GET /transports/0/play

Determine if transport is playing.

Response

200 - Transport play response.

The response is JSON.

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

PUT /transports/0/play

Start playing on transport. Deprecated, use POST /transports/0/play instead.

Response

204 - Transport playing.

400 - Failed to set transport to play.

POST /transports/0/play

Start playing on transport.

Response

204 - Transport playing.

400 - Failed to set transport to play.

GET /transports/0/playback

Get playback state.

Response

200 - Transport playback state.

The response is JSON.

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

PUT /transports/0/playback

Set playback state.

Parameters

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

Response

204 - Updated transport playback state.

400 - Failed to set transport playback state.

GET /transports/0/record

Get record state.

Response

200 - Recording state.

The response is JSON.

Name	Type	Description
recording	boolean	If true, transport is in InputRecord mode.

PUT /transports/0/record

Set record state. Deprecated, use POST /transports/0/record instead.

Parameters

Name	Type	Description
recording (required)	boolean	If true, starts a recording, otherwise stops.
clipName	string	Optional, sets the requested clip name to record to, when “recording” attribute is set to true.

Response

204 - Recording state updated.

400 - Failed to update recording state.

POST /transports/0/record

Start recording.

Parameters

Name	Type	Description
clipName	string	Optional, provides a specific name of clip to record to.

Response

204 - Recording started.

400 - Failed to start recording.

GET /transports/0/clipIndex

Get the clip index of the currently playing clip on the timeline.

Response

200 - Clip index response.

The response is JSON.

Name	Type	Description
clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

GET /transports/0/timecode

Get device timecode.

Response

200 - Timecode response.

The response is JSON.

Name	Type	Description
display	string	The display timecode serialised as a string.
timeline	string	The timeline timecode serialised as a string.

GET /transports/0/timecode/source

Get timecode source selected on device.

Response

200 - Timecode source response.

The response is JSON.

Name	Type	Description
timecode	string	Possible values are: Timeline, Clip.

Timeline Control API

API for controlling playback timeline.

GET /timelines/0

Get the playback timeline.

Response

200 - Playback timeline.

The response is JSON.

Name	Type	Description
clips	array	
clips[i]	object	Timeline clip.
clips[i].clipUniqueId (required)	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).
clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

404 - No timeline / disk available.

DELETE /timelines/0

Clear the current playback timeline. Deprecated, prefer to use POST /timelines/0/clear

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

POST /timelines/0

Add a clip to the timeline.

Parameters

This parameter can be one of the following types:

Add multiple media clips to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips[i].clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips[i].clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips[i].frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add multiple media clips to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add a single clip to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips.clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips.clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips.frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add a single clip to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the timeline as specified.

501 - The operation is not supported on this device.

POST /timelines/0/add

Add a clip to the end of the timeline. Deprecated, use POST /timelines/0 to add clips within the timeline.

Parameters

This parameter can be one of the following types:

Add one clip to the end of the timeline.

Name	Type	Description
clips	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add many clips to the end of the timeline.

Name	Type	Description
clips	array	List of clipUniqueIds of clips to add to end of timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the end of the timeline.

501 - The operation is not supported on this device.

POST /timelines/0/clear

Clear the playback timeline.

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

DELETE /timelines/0/clips/{timelineClipIndex}

Remove the specified clip from the timeline.

Parameters

Name	Type	Description
{timelineClipIndex} (required)	integer	The (0-based) timeline clip index of the clip to remove from the timeline.

Response

204 - The specified clip was removed from the timeline.

501 - The operation is not supported on this device.

Media Control API

API for controlling media devices in Blackmagic Design products.

GET /media/workingset

Get the list of media devices currently in the working set.

Response

200 - The list of media devices in the working set.

The response is JSON.

Name	Type	Description
size	integer	The fixed size of this device's working set.
workingset	array	The device's working set.
workingset[i]	object | null	Device within the working set. null if no device is present within the given working set slot.
workingset[i].volume	string	Volume name.
workingset[i].deviceName	string	Internal device name of this media device.
workingset[i].remainingRecordTime	integer	Remaining record time using current codec and video format in seconds.
workingset[i].totalSpace	integer	Total space on media device in bytes.
workingset[i].remainingSpace	integer	Remaining space on media device in bytes.
workingset[i].clipCount	integer	Number of clips currently on the device.

GET /media/active

Get the currently active media device.

Response

200 - The current active media device.

The response is JSON.

The active media device, or null if there is no active media.

Name	Type	Description
workingsetIndex	integer	Working set index of the active media device.
deviceName	string	Device name of media device.

204 - No media is currently active.

PUT /media/active

Set the currently active media device.

Parameters

Name	Type	Description
workingsetIndex	integer	Working set index of the media to make active.

Response

204 - The active media device was set successfully.

400 - Setting the currently active media device is not possible in the current state.

GET /media/devices/doformatSupportedFilesystems

Get the list of filesystems available to format a media device.

Response

200 - The list of filesystems permitted for formatting.

The response is JSON.

Name	Type	Description
	array	List of filesystems permitted for formatting media.
[i]	string	Filesystem serialised as string.

GET /media/devices/{deviceName}

Get information about a requested device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Information about the requested device.

The response is JSON.

Media device state.

Name	Type	Description
state	string	The current state of the media device. Possible values are: None, Scanning, Mounted, Uninitialised, Formatting, RaidComponent.

400 - Invalid device name.

404 - Device not found.

GET /media/devices/{deviceName}/doformat

Get a format key, used to format the device with a PUT request.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Format prepared.

The response is JSON.

Name	Type	Description
deviceName	string	Device name of media device to format.
key	string	The key required to format this device, provide to PUT /media/devices/{deviceName}/doformat to perform format of media device.

400 - Cannot format the device.

404 - Device not found.

PUT /media/devices/{deviceName}/doformat

Perform a format of the specified media device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by "deviceName" member of GET /media/workingset or GET /media/active.

Name	Type	Description
key	string	The key used to format this device, retrieved from prepare format media request GET /media/devices/{deviceName}/doformat. Format key provided cannot be reused after successful format.
filesystem	string	Filesystem to format to. Supported filesystems can be retrieved with GET /media/devices/doFormatSupportedFilesystems.
volume	string	Volume name to set for the disk after format.

Response

204 - Format successful.

400 - Cannot format the device, invalid filesystem or key.

404 - Device not found.

Slate Control API

API to manage digital slate data.

GET /slates/nextClip

Retrieve the digital slate for the next clip.

Response

200 - Returns the slate data for the next clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	

Name	Type	Description
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Slate data is not available.

PUT /slates/nextClip

Update the slate data for the next clip.

Parameters

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

POST /slates/nextClip/resetProjectData

Reset the project data for the next clip's slate.

Response

200 - Project data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetProjectData

Reset the project data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Project data reset successfully.

POST /slates/nextClip/resetLensData

Reset the lens data for the next clip's slate.

Response

200 - Lens data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetLensData

Reset the lens data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Lens data reset successfully.

GET /slates/clips/{deviceName}/{path}

Retrieve slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Returns the slate data for the specified clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

404 - Clip not found.

PUT /slates/clips/{deviceName}/{path}

Update the slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.

Name	Type	Description
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

Preset Control API

API For controlling the presets on Blackmagic Design products

GET /presets

Get the list of the presets on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presets	array	List of the presets on the camera (.cset files)
presets[i]	string	

POST /presets

Send a preset file to the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presetAdded	string	Name of the preset uploaded (without .cset extension)

400 - Bad request - missing Content-Disposition header or filename

GET /presets/active

Get the currently active preset on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

PUT /presets/active

Set the active preset on the camera

Parameters

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

Response

204 - No Content

404 - Preset file not found

PARAMETERS /presets/{presetName}

GET /presets/{presetName}

Download the preset file

Response

200 - OK

The response is a binary file.

404 - File does not exist

PUT /presets/{presetName}

Save current camera state as a preset

Response

204 - No Content

DELETE /presets/{presetName}

Delete a preset from the camera

Response

204 - No Content

404 - Preset file not found

Audio Control API

API For controlling audio on Blackmagic Design Cameras

GET /audio/channels

Get the total number of audio channels available

Response

200 - Returns the total number of channels

The response is JSON.

Name	Type	Description
channels	integer	Total number of audio channels available

GET /audio/supportedInputs

Get the list of supported audio inputs

Response

200 - List of all supported audio inputs

The response is JSON.

Name	Type	Description
	array	
[i]	string	A supported audio input

GET /audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently selected input

The response is JSON.

Name	Type	Description
input	string	Audio input source and type

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/input

Set the audio input for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
input	string	Audio input source and type

Response

204 - No Content

400 - Invalid audio input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/input/description

Get the description of the current input of the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Description of the current input of the selected channel

The response is JSON.

Name	Type	Description
description	object	
description.gainRange	object	
description.gainRange.Min	number	The minimum gain value in dB
description.gainRange.Max	number	The maximum gain value in dB
description.capabilities	object	
description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter
description.capabilities.Padding	object	
description.capabilities.Padding. available	boolean	Input supports setting of padding
description.capabilities.Padding. forced	boolean	Padding is forced to be set for the input
description.capabilities.Padding. value	number	Value of the padding in dB

404 - Channel does not exist

GET /audio/channel/{channelIndex}/supportedInputs

Get the list of supported inputs and their availability to switch to for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its supported inputs are being queried. (Channels index from 0)

Response

200 - The list of supported inputs

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

404 - Channel does not exist

GET /audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set level for the selected channel

The response is JSON.

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/level

Set the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

Response

204 - No Content

400 - Invalid input or value out of range

404 - Channel does not exist

GET /audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set phantom power for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Phantom power enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/phantomPower

Set the audio phantom power for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Phantom power enabled state

Response

204 - No Content

400 - Phantom power is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set padding for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Padding enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/padding

Set the audio input padding for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Padding enabled state

Response

204 - No Content

400 - Padding is not supported or is forced for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set low cut filter for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Low cut filter enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/lowCutFilter

Set the audio input low cut filter for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Low cut filter enabled state

Response

204 - No Content

400 - Low cut filter is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set availability for the selected channel

The response is JSON.

Name	Type	Description
available	boolean	Whether the input is currently available

404 - Channel does not exist

Lens Control API

API For controlling the lens on Blackmagic Design products

GET /lens/iris

Get lens' aperture

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number

PUT /lens/iris

Set lens' aperture

Parameters

Name	Type	Description
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number
adjustmentStep	integer	Signed value for relative aperture adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens iris is not controllable or is controlled by auto exposure

GET /lens/zoom

Get lens' zoom

Response

200 - OK

The response is JSON.

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value

PUT /lens/zoom

Set lens' zoom

Parameters

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value
adjustmentFocalLength	integer	Signed value for relative focal length adjustment
adjustmentNormalised	number	Signed normalized value for relative zoom adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens zoom is not controllable

GET /lens/focus

Get lens' focus

Response

200 - OK

The response is JSON.

Name	Type	Description
normalised	number	Normalised value

PUT /lens/focus

Set lens' focus

Parameters

Name	Type	Description
normalised	number	Normalised value
focusDistance	integer	Focus distance value

Response

204 - No Content

400 - Bad Request if out of range value is provided

PUT /lens/focus/doAutoFocus

Perform auto focus

Parameters

Name	Type	Description
position	object	
position.x	number	Normalized x coordinate for autofocus ROI
position.y	number	Normalized y coordinate for autofocus ROI

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens focus is not controllable

GET /lens/opticalImageStabilization

Get optical image stabilization status

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Whether optical image stabilization is enabled

501 - Not Implemented if optical image stabilization is not supported on this product

PUT /lens/opticalImageStabilization

Enable or disable optical image stabilization

Parameters

Name	Type	Description
enabled	boolean	Enable or disable optical image stabilization

Response

204 - No Content

501 - Not Implemented if optical image stabilization is not supported on this product

GET /lens/iris/description

Get detailed description of lens' iris capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the iris can be controlled
apertureStop	object	
apertureStop.min	number	Minimum aperture stop
apertureStop.max	number	Maximum aperture stop

GET /lens/zoom/description

Get detailed description of lens' zoom capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the zoom can be controlled
focalLength	object	
focalLength.adjustable	boolean	If focal length is adjustable
focalLength.min	integer	Minimum focal length
focalLength.max	integer	Maximum focal length

GET /lens/focus/description

Get detailed description of lens' focus capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the focus can be controlled
focusDistance	object	
focusDistance.adjustable	boolean	If focus distance is adjustable
focusDistance.min	number	Minimum focus distance
focusDistance.max	number	Maximum focus distance

Video Control API

API For controlling the video on Blackmagic Design products

GET /video/iso

Get current ISO

Response

200 - OK

The response is JSON.

Name	Type	Description
iso	integer	Current ISO value

PUT /video/iso

Set current ISO

Parameters

Name	Type	Description
iso	integer	ISO value to set

Response

204 - No Content

403 - ISO cannot be changed in the current state

GET /video/supportedISOs

Get the list of supported ISO settings

Response

200 - List of supported ISO values

The response is JSON.

Name	Type	Description
supportedISOs	array	Array of supported ISO values
supportedISOs[i]	integer	

GET /video/gain

Get current gain value in decibels

Response

200 - OK

The response is JSON.

Name	Type	Description
gain	integer	Current gain value in decibels

PUT /video/gain

Set current gain value

Parameters

Name	Type	Description
gain	integer	Gain value in decibels to set

Response

204 - No Content

403 - Gain cannot be changed in the current state

GET /video/supportedGains

Get the list of supported gain settings in decibels

Response

200 - List of supported gain values in decibels

The response is JSON.

Name	Type	Description
supportedGains	array	Array of supported gain values in decibels
supportedGains[i]	integer	

GET /video/whiteBalance

Get current white balance

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	integer	Current white balance

PUT /video/whiteBalance

Set current white balance

Parameters

Name	Type	Description
whiteBalance	integer	White balance to set

Response

204 - No Content

400 - Invalid white balance temperature

GET /video/whiteBalance/description

Get white balance range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	object	
whiteBalance.min	integer	Minimum color temperature
whiteBalance.max	integer	Maximum color temperature

PUT /video/whiteBalance/doAuto

Set current white balance automatically

Response

204 - No Content

GET /video/whiteBalanceTint

Get white balance tint

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	integer	Current white balance tint

PUT /video/whiteBalanceTint

Set white balance tint

Parameters

Name	Type	Description
whiteBalanceTint	integer	White balance tint to set

Response

204 - No Content

400 - Invalid white balance tint

GET /video/whiteBalanceTint/description

Get white balance tint range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	object	
whiteBalanceTint.min	integer	Minimum white balance tint
whiteBalanceTint.max	integer	Maximum white balance tint

GET /video/ndFilter

Get ND filter stop

Response

200 - OK

The response is JSON.

Name	Type	Description
stop	number	Current filter power (fStop)

501 - Not implemented for this device

PUT /video/ndFilter

Set ND filter stop

Parameters

Name	Type	Description
stop	number	Filter power (fStop) to set

Response

204 - No Content

400 - Invalid ND filter stop

501 - Not implemented for this device

GET /video/supportedNDFilters

Get the list of available ND filter stops

Response

200 - List of available ND filter stops

The response is JSON.

Name	Type	Description
supportedStops	array	Array of available ND filter stops
supportedStops[i]	number	

501 - Not implemented for this device

GET /video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

Response

200 - List of supported display modes

The response is JSON.

Name	Type	Description
supportedDisplayModes	array	Array of supported display modes
supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

GET /video/ndFilter/displayMode

Get ND filter display mode on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

PUT /video/ndFilter/displayMode

Set ND filter display mode on the camera

Parameters

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

Response

204 - No Content

400 - Invalid display mode for ND filter

501 - Not implemented for this device

GET /video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

Response

200 - Indicates if ND filter is selectable

The response is JSON.

Name	Type	Description
selectable	boolean	True if ND filter adjustments are selectable via a slider

501 - Not implemented for this device

GET /video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

PUT /video/shutter

Set current shutter

Parameters

Name	Type	Description
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

Response

204 - No Content

GET /video/shutter/measurement

Get the current shutter measurement mode

Response

200 - OK

The response is JSON.

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

PUT /video/shutter/measurement

Set the shutter measurement mode

Parameters

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

Response

204 - No Content

400 - Invalid shutter measurement

GET /video/supportedShutters

Get supported shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of supported shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of supported shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of flicker-free shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of flicker-free shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/autoExposure

Get current auto exposure mode

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

PUT /video/autoExposure

Set auto exposure

Parameters

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

Response

204 - No Content

400 - Failed to set auto exposure mode

GET /video/detailSharpening

Get the current state of detail sharpening

Response

200 - Current detail sharpening state

The response is JSON.

Name	Type	Description
enabled	boolean	Whether detail sharpening is enabled

501 - Not implemented for this device

PUT /video/detailSharpening

Enable or disable detail sharpening

Parameters

Name	Type	Description
enabled	boolean	Enable or disable detail sharpening

Response

204 - Detail sharpening state updated

501 - Not implemented for this device

GET /video/detailSharpeningLevel

Get the current detail sharpening level

Response

200 - Current detail sharpening level

The response is JSON.

Name	Type	Description
level	string	Current detail sharpening level Possible values are: Low, Medium, High.

501 - Not implemented for this device

PUT /video/detailSharpeningLevel

Set the detail sharpening level

Parameters

Name	Type	Description
level	string	Desired level of detail sharpening Possible values are: Low, Medium, High.

Response

204 - Detail sharpening level updated

400 - Invalid detail sharpening level

501 - Not implemented for this device

Camera Control API

API For controlling the Camera specific features on Blackmagic Design products

GET /camera/colorBars

Get the status of color bars display

Response

200 - Returns the current status of color bars

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the color bars are currently enabled

PUT /camera/colorBars

Set the status of color bars display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the color bars

Response

204 - Color bars status updated successfully

GET /camera/programFeedDisplay

Get the status of program feed display

Response

200 - Returns the current status of program feed display

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the program feed display is currently enabled

PUT /camera/programFeedDisplay

Set the status of program feed display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the program feed display

Response

204 - Program feed display status updated successfully

GET /camera/tallyStatus

Get the tally status of the camera

Response

200 - Returns the current tally status of the camera

The response is JSON.

Name	Type	Description
status	string	Current tally status of the camera Possible values are: None, Preview, Program.

GET /camera/power

Get the power status of the camera

Response

200 - Returns the current power status

The response is JSON.

Name	Type	Description
source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
batteries	array	List of batteries currently connected to the camera
batteries[i]	object	
batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
batteries[i].chargeRemainingPercent	integer	Remaining battery charge percentage
batteries[i].statusFlags	array	List of battery status flags
batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

GET /camera/power/displayMode

Get the power display mode of the camera

Response

200 - Returns the current power display mode

The response is JSON.

Name	Type	Description
mode	string	Current power display mode Possible values are: Percentage, Voltage.

PUT /camera/power/displayMode

Set the power display mode of the camera

Parameters

Name	Type	Description
mode	string	Power display mode to set Possible values are: Percentage, Voltage.

Response

204 - Power display mode updated successfully

400 - Invalid power display mode

GET /camera/timingReferenceLock

Get the timing reference lock status

Response

200 - Returns the timing reference lock status

The response is JSON.

Name	Type	Description
locked	boolean	Indicates if timing reference is locked

Immersive Control API

API for controlling immersive camera settings on Blackmagic Design cameras

GET /immersive/display/{displayName}/eye

Get the current eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to query (from /monitoring/display endpoint)

Response

200 - OK

The response is JSON.

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

400 - Invalid display name format

404 - Display not found

422 - Failed to get eye view

PUT /immersive/display/{displayName}/eye

Set the eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to control (from /monitoring/display endpoint)

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

Response

204 - No Content

400 - Invalid input or display name format

404 - Display not found

422 - Failed to set eye view

Color Correction Control API

API For controlling the color correction on Blackmagic Design products based on DaVinci Resolve Color Corrector

GET /colorCorrection/lift

Get color correction lift

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

PUT /colorCorrection/lift

Set color correction lift

Parameters

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gamma

Get color correction gamma

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

PUT /colorCorrection/gamma

Set color correction gamma

Parameters

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gain

Get color correction gain

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

PUT /colorCorrection/gain

Set color correction gain

Parameters

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/offset

Get color correction offset

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

PUT /colorCorrection/offset

Set color correction offset

Parameters

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/contrast

Get color correction contrast

Response

200 - OK

The response is JSON.

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/contrast

Set color correction contrast

Parameters

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/color

Get color correction color properties

Response

200 - OK

The response is JSON.

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/color

Set color correction color properties

Parameters

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/lumaContribution

Get color correction luma contribution

Response

200 - OK

The response is JSON.

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

PUT /colorCorrection/lumaContribution

Set color correction luma contribution

Parameters

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

Response

204 - No Content

Notification websocket - 1.0.0

Service that notifies subscribers of device state changes.

messages

Subscribe (The messages from the server/device)

Websocket Opened Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: websocketOpened.
.type	string	Possible values are: event.

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*

Name	Type	Description
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Event Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: propertyValueChanged.

Name	Type	Description
.data.property	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.value	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.type	string	Possible values are: event.

Publish (The messages that user can send to the server/device)

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	

Name	Type	Description
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	

Name	Type	Description
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Device Properties

/media/workingset

The value JSON returned via the eventResponse when the `/media/workingset` property changes on the device:

Name	Type	Description
.size	integer	The fixed size of this device's working set.
.workingset	array	Array of devices within the working set. null if no device is present within the given working set slot.
.workingset[i]		

/media/active

The value JSON returned via the eventResponse when the `/media/active` property changes on the device:

Name	Type	Description
.workingsetIndex	integer	Working set index of the active media device.
.deviceName	string	Internal device name of this media device.

/system

The value JSON returned via the eventResponse when the `/system` property changes on the device:

Name	Type	Description
.codecFormat	object	Codec format configuration.
.codecFormat.codec	string	Codec format serialised as a string.
.codecFormat.container	string	Multimedia container format.
.videoFormat	object	Currently selected video format.
.videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.videoFormat.height	number	Height dimension of video format.
.videoFormat.width	number	Width dimension of video format.
.videoFormat.interlaced	boolean	Is the display format interlaced?.
.videoFormat.name	string	Video format serialised as a string.

/system/codecFormat

Codec format configuration.

The value JSON returned via the eventResponse when the `/system/codecFormat` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.container	string	Multimedia container format.

/system/videoFormat

Currently selected video format.

The value JSON returned via the eventResponse when the `/system/videoFormat` property changes on the device:

Name	Type	Description
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.height	number	Height dimension of video format.
.width	number	Width dimension of video format.
.interlaced	boolean	Is the display format interlaced?.
.name	string	Video format serialised as a string.

/system/format

The value JSON returned via the eventResponse when the `/system/format` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.maxOffSpeedFrameRate	number	
.minOffSpeedFrameRate	number	
.offSpeedEnabled	boolean	
.offSpeedFrameRate	number	
.recordResolution	object	
.recordResolution.height	number	Height of the resolution.
.recordResolution.width	number	Width of the resolution.
.sensorResolution	object	
.sensorResolution.height	number	Height of the resolution.
.sensorResolution.width	number	Width of the resolution.

/system/supportedFormats

The value JSON returned via the eventResponse when the `/system/supportedFormats` property changes on the device:

Name	Type	Description
.supportedFormats	array	
.supportedFormats[i]	object	
.supportedFormats[i].codecs	array	
.supportedFormats[i].codecs[i]	string	
.supportedFormats[i].frameRates	array	
.supportedFormats[i].frameRates[i]	string	Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.supportedFormats[i].maxOffSpeedFrameRate	number	
.supportedFormats[i].minOffSpeedFrameRate	number	
.supportedFormats[i].recordResolution	object	
.supportedFormats[i].recordResolution.height	number	Height of the resolution.
.supportedFormats[i].recordResolution.width	number	Width of the resolution.
.supportedFormats[i].sensorResolution	object	
.supportedFormats[i].sensorResolution.height	number	Height of the resolution.
.supportedFormats[i].sensorResolution.width	number	Width of the resolution.

/timelines/0

The value JSON returned via the eventResponse when the `/timelines/0` property changes on the device:

Name	Type	Description
.clips	array	
.clips[i]	object	Timeline clip.
.clips[i].clipUniqueId	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
.clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
.clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
.clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
.clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).

Name	Type	Description
.clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
.clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

/transports/0

The value JSON returned via the eventResponse when the `/transports/0` property changes on the device:

Name	Type	Description
.mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

/transports/0/stop

true when transport mode is InputPreview or when in Output mode and speed is 0.

The value JSON returned via the eventResponse when the `/transports/0/stop` property changes on the device:

Name	Type	Description
	boolean	true when transport mode is InputPreview or when in Output mode and speed is 0.

/transports/0/play

True when transport is in Output mode and speed is non-zero.

The value JSON returned via the eventResponse when the /transports/0/play property changes on the device:

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

/transports/0/playback

The value JSON returned via the eventResponse when the `/transports/0/playback` property changes on the device:

Name	Type	Description
.type	string	Possible values are: Play, Jog, Shuttle, Var.
.loop	boolean	When true playback loops from the end of the timeline to the beginning of the timeline.
.singleClip	boolean	When true playback loops from the end of the current clip to the beginning of the current clip.
.speed	number	Playback speed, 1.0 for normal forward playback
.position	integer	Playback position on the timeline in units of video frames.

/transports/0/record

The value JSON returned via the eventResponse when the `/transports/0/record` property changes on the device:

Name	Type	Description
.recording	boolean	Is transport in Input Record mode.

/transports/0/timecode

The value JSON returned via the eventResponse when the `/transports/0/timecode` property changes on the device:

Name	Type	Description
.display	string	The display timecode serialised as a string.
.timeline	string	The timeline timecode serialised as a string.

/transports/0/timecode/source

The value JSON returned via the eventResponse when the `/transports/0/timecode/source` property changes on the device:

Name	Type	Description
.timecode	string	Possible values are: Timeline, Clip.

/transports/0/clipIndex

The value JSON returned via the eventResponse when the `/transports/0/clipIndex` property changes on the device:

Name	Type	Description
.clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

/slates/nextClip

The value JSON returned via the eventResponse when the `/slates/nextClip` property changes on the device:

Name	Type	Description
.clip	object	
.clip.clipName	string	
.clip.reel	integer	
.clip.scene	string	
.clip.sceneLocation	string	Possible values are: Interior, Exterior.
.clip.sceneTime	string	Possible values are: Day, Night.
.clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
.clip.take	integer	
.clip.takeType	string	Possible values are: None, PU, VFX, SER.
.clip.goodTake	boolean	
.lens	object	
.lens.lensType	string	
.lens.iris	string	

Name	Type	Description
.lens.focalLength	string	
.lens.distance	string	
.lens.filter	string	
.project	object	
.project.projectName	string	
.project.director	string	
.project.camera	string	
.project.cameraOperator	string	

/monitoring/{displayName}/cleanFeed

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/cleanFeed` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/displayLUT

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/displayLUT` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/zebra

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/zebra` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/focusAssist` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGuide

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGuide` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGrids` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/safeArea

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/safeArea` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/falseColor

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/falseColor` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/focusAssist` property changes on the device:

Name	Type	Description
.mode	string	Possible values are: Peak, ColoredLines.
.color	string	Possible values are: Red, Green, Blue, White, Black.
.intensity	integer	

/monitoring/frameGuideRatio

The value JSON returned via the eventResponse when the `/monitoring/frameGuideRatio` property changes on the device:

Name	Type	Description
.ratio	string	

/monitoring/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/frameGrids` property changes on the device:

Name	Type	Description
.frameGrids	array	
.frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

/monitoring/safeAreaPercent

The value JSON returned via the eventResponse when the `/monitoring/safeAreaPercent` property changes on the device:

Name	Type	Description
.percent	integer	Safe area coverage percentage.

/audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input` property changes on the device:

Name	Type	Description
.input	string	Audio input source and type

/audio/channel/{channelIndex}/supportedInputs

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/supportedInputs` property changes on the device:

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

/audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/level` property changes on the device:

Name	Type	Description
.gain	number	Gain value in dB
.normalised	number	Normalised level value between 0.0 and 1.0

/audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/phantomPower` property changes on the device:

Name	Type	Description
.enabled	boolean	Phantom power enabled state

/audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/padding` property changes on the device:

Name	Type	Description
.enabled	boolean	Padding enabled state

/audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/lowCutFilter` property changes on the device:

Name	Type	Description
.enabled	boolean	Low cut filter enabled state

/audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/available` property changes on the device:

Name	Type	Description
.available	boolean	Whether the input is currently available

/audio/channel/{channelIndex}/input/description

Description of the current input of the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input/description` property changes on the device:

Name	Type	Description
.description	object	
.description.gainRange	object	
.description.gainRange.Min	number	The minimum gain value in dB
.description.gainRange.Max	number	The maximum gain value in dB
.description.capabilities	object	
.description.capabilities.PhantomPower	boolean	Input supports setting of phantom power
.description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter

Name	Type	Description
.description.capabilities.Padding	object	
.description.capabilities.Padding.available	boolean	Input supports setting of padding
.description.capabilities.Padding.forced	boolean	Padding is forced to be set for the input
.description.capabilities.Padding.value	number	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.

/colorCorrection/lift

Get color correction lift

The value JSON returned via the eventResponse when the `/colorCorrection/lift` property changes on the device:

Name	Type	Description
.red	number	Red lift component
.green	number	Green lift component
.blue	number	Blue lift component
.luma	number	Luma lift component

/colorCorrection/gamma

Get color correction gamma

The value JSON returned via the eventResponse when the `/colorCorrection/gamma` property changes on the device:

Name	Type	Description
.red	number	Red gamma component
.green	number	Green gamma component
.blue	number	Blue gamma component
.luma	number	Luma gamma component

/colorCorrection/gain

Get color correction gain

The value JSON returned via the eventResponse when the `/colorCorrection/gain` property changes on the device:

Name	Type	Description
.red	number	Red gain component
.green	number	Green gain component
.blue	number	Blue gain component
.luma	number	Luma gain component

/colorCorrection/offset

Get color correction offset

The value JSON returned via the eventResponse when the `/colorCorrection/offset` property changes on the device:

Name	Type	Description
.red	number	Red offset component
.green	number	Green offset component
.blue	number	Blue offset component
.luma	number	Luma offset component

/colorCorrection/contrast

Get color correction contrast

The value JSON returned via the eventResponse when the `/colorCorrection/contrast` property changes on the device:

Name	Type	Description
.pivot	number	Contrast pivot point
.adjust	number	Contrast adjustment

/colorCorrection/color

Get color correction color properties

The value JSON returned via the eventResponse when the `/colorCorrection/color` property changes on the device:

Name	Type	Description
.hue	number	Color hue adjustment
.saturation	number	Color saturation adjustment

/colorCorrection/lumaContribution

Get color correction luma contribution

The value JSON returned via the eventResponse when the `/colorCorrection/lumaContribution` property changes on the device:

Name	Type	Description
.lumaContribution	number	Luma contribution value

/lens/iris

Get lens' aperture

The value JSON returned via the eventResponse when the `/lens/iris` property changes on the device:

Name	Type	Description
.continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
.apertureStop	number	Aperture stop value
.normalised	number	Normalised value
.apertureNumber	integer	Aperture number

/lens/iris/description

Get detailed description of lens' iris capabilities

The value JSON returned via the eventResponse when the `/lens/iris/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the iris can be controlled
.apertureStop	object	
.apertureStop.min	number	Minimum aperture stop
.apertureStop.max	number	Maximum aperture stop

/lens/focus

Get lens' focus

The value JSON returned via the eventResponse when the `/lens/focus` property changes on the device:

Name	Type	Description
.normalised	number	Normalised value

/lens/focus/description

Get detailed description of lens' focus capabilities

The value JSON returned via the eventResponse when the `/lens/focus/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the focus can be controlled
.focusDistance	object	
.focusDistance.adjustable	boolean	If focus distance is adjustable
.focusDistance.min	number	Minimum focus distance
.focusDistance.max	number	Maximum focus distance

/lens/zoom

Get lens' zoom

The value JSON returned via the eventResponse when the `/lens/zoom` property changes on the device:

Name	Type	Description
.focalLength	integer	Focal length in mm
.normalised	number	Normalised value

/lens/zoom/description

Get detailed description of lens' zoom capabilities

The value JSON returned via the eventResponse when the `/lens/zoom/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the zoom can be controlled
.focalLength	object	
.focalLength.adjustable	boolean	If focal length is adjustable
.focalLength.min	integer	Minimum focal length
.focalLength.max	integer	Maximum focal length

/presets

Get the list of the presets on the camera

The value JSON returned via the eventResponse when the `/presets` property changes on the device:

Name	Type	Description
.presets	array	List of the presets on the camera (.cset files)
.presets[i]	string	

/presets/active

Get the currently active preset on the camera

The value JSON returned via the eventResponse when the `/presets/active` property changes on the device:

Name	Type	Description
.preset	string	Name of the active preset (with .cset extension, or 'default')

/camera/colorBars

Get the status of color bars display

The value JSON returned via the eventResponse when the `/camera/colorBars` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the color bars are currently enabled

/camera/programFeedDisplay

Get the status of program feed display

The value JSON returned via the eventResponse when the `/camera/programFeedDisplay` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the program feed display is currently enabled

/camera/tallyStatus

Get the tally status of the camera

The value JSON returned via the eventResponse when the `/camera/tallyStatus` property changes on the device:

Name	Type	Description
.status	string	Current tally status of the camera Possible values are: None, Preview, Program.

/camera/power

Get the power status of the camera

The value JSON returned via the eventResponse when the `/camera/power` property changes on the device:

Name	Type	Description
.source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
.milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
.batteries	array	
.batteries[i]	object	
.batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
.batteries[i]. chargeRemainingPercent	integer	Remaining battery charge percentage
.batteries[i].statusFlags	array	List of battery status flags
.batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

/camera/power/displayMode

Get the power display mode of the camera

The value JSON returned via the eventResponse when the `/camera/power/displayMode` property changes on the device:

Name	Type	Description
.mode	string	Current power display mode Possible values are: Percentage, Voltage.

/camera/timingReferenceLock

Get the timing reference lock status

The value JSON returned via the eventResponse when the `/camera/timingReferenceLock` property changes on the device:

Name	Type	Description
.locked	boolean	Indicates if timing reference is locked

/video/iso

Get current ISO

The value JSON returned via the eventResponse when the `/video/iso` property changes on the device:

Name	Type	Description
.iso	integer	Current ISO value

/video/supportedISOs

Get the list of supported ISO settings

The value JSON returned via the eventResponse when the `/video/supportedISOs` property changes on the device:

Name	Type	Description
.supportedISOs	array	Array of supported ISO values
.supportedISOs[i]	integer	

/video/gain

Get current gain value in decibels

The value JSON returned via the eventResponse when the `/video/gain` property changes on the device:

Name	Type	Description
.gain	integer	Current gain value in decibels

/video/supportedGains

Get the list of supported gain settings in decibels

The value JSON returned via the eventResponse when the `/video/supportedGains` property changes on the device:

Name	Type	Description
.supportedGains	array	Array of supported gain values in decibels
.supportedGains[i]	integer	

/video/whiteBalance

Get current white balance

The value JSON returned via the eventResponse when the `/video/whiteBalance` property changes on the device:

Name	Type	Description
.whiteBalance	integer	Current white balance

/video/whiteBalance/description

Get white balance range

The value JSON returned via the eventResponse when the `/video/whiteBalance/description` property changes on the device:

Name	Type	Description
.whiteBalance	object	
.whiteBalance.min	integer	Minimum color temperature
.whiteBalance.max	integer	Maximum color temperature

/video/whiteBalanceTint

Get white balance tint

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint` property changes on the device:

Name	Type	Description
.whiteBalanceTint	integer	Current white balance tint

/video/whiteBalanceTint/description

Get white balance tint range

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint/description` property changes on the device:

Name	Type	Description
.whiteBalanceTint	object	
.whiteBalanceTint.min	integer	Minimum white balance tint
.whiteBalanceTint.max	integer	Maximum white balance tint

/video/ndFilter

Get ND filter stop

The value JSON returned via the eventResponse when the `/video/ndFilter` property changes on the device:

Name	Type	Description
.stop	number	Current filter power (fStop)

/video/supportedNDFilters

Get the list of available ND filter stops

The value JSON returned via the eventResponse when the `/video/supportedNDFilters` property changes on the device:

Name	Type	Description
.supportedStops	array	Array of available ND filter stops
.supportedStops[i]	number	

/video/ndFilter/displayMode

Get ND filter display mode on the camera

The value JSON returned via the eventResponse when the `/video/ndFilter/displayMode` property changes on the device:

Name	Type	Description
.displayMode	string	ND filter display mode Possible values are: Stop, Number, Fraction.

/video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

The value JSON returned via the eventResponse when the `/video/supportedNDFilterDisplayModes` property changes on the device:

Name	Type	Description
.supportedDisplayModes	array	Array of supported display modes
.supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

/video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

The value JSON returned via the eventResponse when the `/video/ndFilterSelectable` property changes on the device:

Name	Type	Description
.selectable	boolean	True if ND filter adjustments are selectable via a slider

/video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

The value JSON returned via the eventResponse when the `/video/shutter` property changes on the device:

Name	Type	Description
.continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
.shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
.shutterAngle	number	Shutter angle

/video/shutter/measurement

Get the current shutter measurement mode

The value JSON returned via the eventResponse when the `/video/shutter/measurement` property changes on the device:

Name	Type	Description
.measurement	string	Current shutter measurement mode Possible values are: ShutterAngle, ShutterSpeed.

/video/supportedShutters

Get supported shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/supportedShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of supported shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array of flicker-free shutter speeds
.shutterSpeeds[i]	integer	

/video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/flickerFreeShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of flicker-free shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array
.shutterSpeeds[i]	integer	

/video/autoExposure

Get current auto exposure mode

The value JSON returned via the eventResponse when the `/video/autoExposure` property changes on the device:

Name	Type	Description
.mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
.type	string	Comma-separated list of device types in the auto exposure stack

/video/detailSharpening

Get the current state of detail sharpening

The value JSON returned via the eventResponse when the `/video/detailSharpening` property changes on the device:

Name	Type	Description
.enabled	boolean	Whether detail sharpening is enabled

/video/detailSharpeningLevel

Get the current detail sharpening level

The value JSON returned via the eventResponse when the `/video/detailSharpeningLevel` property changes on the device:

Name	Type	Description
.level	string	Current detail sharpening level of supported shutter speeds Possible values are: Low, Medium, High.

Yardım

Yardım almanın en hızlı yolu, Blackmagic Design internet destek sayfalarına girip, kameranız için olan en son destek malzemesini incelemenizdir.

Blackmagic Design İnternet Destek Sayfaları

En son kılavuz, yazılım ve destek notlarına www.blackmagicdesign.com/tr/support adresindeki, Blackmagic Design destek merkezinden ulaşılabilir.

Blackmagic Design Destek Ekibiyle İrtibat

Aradığınız yardımı destek kaynaklarımızda bulamazsanız lütfen, destek sayfamıza girerek ve "Bize e-posta gönderin" butonunu tıklayarak, e-posta yoluyla destek talebinde bulunun. Alternatif olarak, destek sayfasındaki "Yerel destek ekibinizi bulun" butonunu tıklayabilir ve size en yakın Blackmagic Design destek ofisini arayabilirsiniz.

Mevcut Yazılım Sürümünün Kontrol Edilmesi

Bilgisayarınızda yüklü olan Blackmagic Camera Setup kurulum yazılımının sürümünü kontrol etmek için “Blackmagic Camera Setup Hakkında” yardımcı yazılım penceresini açın.

- Mac bilgisayarlarda, Uygulamalar dosyasından Blackmagic Camera Setup Utility uygulamasını açın. Sürüm numarasını görüntülemek için, uygulamalar menüsünden “About Blackmagic Camera Setup utility” sekmesini seçin.
- Windows bilgisayarlarında, “Blackmagic Camera Setup utility” uygulamasını, “Başlat” menüsünden ya da Başlat ekranından açın. Sürüm numarasını görüntülemek için, Yardım menüsünü tıklayın ve “Blackmagic Camera Setup utility Hakkında” sekmesini seçin.

En Son Yazılım Güncellemelerine Erişim

Bilgisayarınızda yüklü olan Blackmagic Camera Utility yazılımının sürümünü kontrol ettikten sonra, www.blackmagicdesign.com/tr/support adresindeki Blackmagic Design destek merkezine girerek en son güncellemeleri gözden geçirin. En son güncellemeleri kullanmak faydalı olsa da önemli bir projenin ortasındayken, herhangi bir yazılımı güncellemekten kaçınmakta yarar vardır.

Mevzuata İlişkin Bildirimler

Avrupa Birliği Dahilinde Elektrikli ve Elektronik Cihazlara Dair Atık Kontrol Yönetmeliği.
Ürün üzerindeki sembol, bu cihazın başka atık malzemelerle bertaraf edilmemesi şartını gösterir. Atık ekipmanlarınızı bertaraf edebilmeniz için geri dönüşümünü sağlamak üzere, belirlenmiş toplama noktasına teslim edilmeleri şarttır. Bertaraf anında atık cihazlarınızın ayrı olarak toplanması ve geri dönüşümü, doğal kaynakların korunmasına yardımcı olur ve insan sağlığını ve çevreyi koruyucu bir şekilde geri dönüşümünü sağlar. Atık cihazlarınızı geri dönüşüm için nereye teslim edebileceğiniz konusunda daha fazla bilgi için, lütfen yerel belediyenizin geri dönüşüm şubesini ya da ürünü satın aldığınız satış bayisini arayınız.

Bu cihaz, test edilmiş ve Federal İletişim Komisyonu (FCC) koşullarının 15. bölümü doğrultusunda, A Sınıfı dijital cihazların sınırlarıyla uyumlu olduğu tespit edilmiştir. İlgili sınırlar, bu cihaz ticari bir ortamda çalıştırıldığında, zararlı müdahalelere karşı makul koruma sağlaması amacıyla tasarlanmıştır. Bu cihaz; radyo frekans enerjisi üretir, kullanır ve saçabilir ve talimatlar doğrultusunda kurulmadığı ve kullanılmadığı takdirde, telsiz haberleşmelerine zararlı parazite neden olabilir. Bu ürünün bir yerleşim bölgesinde çalıştırılması, zararlı parazitlere neden olması muhtemeldir. Bu durumda, kullanıcı bu paraziti gidermek ve ilgili maliyeti karşılamakla yükümlüdür.

Bu ekipmanın çalıştırılması aşağıdaki iki şarta bağlıdır:

1. Bu ekipman veya cihaz, zararlı müdahaleye sebebiyet vermemelidir.
2. Bu ekipman veya cihaz, arzu edilmeyen bir çalışma şekline yol açacak müdahaleler de dahil olmak üzere, maruz kaldığı her türlü müdahaleyi kabul etmelidir.

R-R-BMD-20250213001

ICES-3 (A)
NMB-3 (A)

ISED Kanada Beyannamesi
Bu cihaz, A Sınıfı dijital cihazlar için Kanada standartlarıyla uyumludur.

Bu cihaza amaçlanan kulanımının haricinde herhangi bir değişiklik yapılması veya kullanılması, bu standartlarla uyumluluğunu geçersiz kılabilir.

HDMI arayüzlerine bağlantı, yüksek kaliteli korumalı HDMI kablolarıyla yapılmalıdır.

Bu cihaz, ticari ortamda kullanım amacına uygunluk için test edilmiştir. Cihaz ev ortamında kullanıldığında, radyo parazitine neden olabilir.

Güvenlik Bilgileri

Blackmagic URSA Cine ile gelen AC'den 12V DC'ye güç kaynağı, koruyucu topraklama bağlantısı olan bir şebeke prizine takılmalıdır.

Blackmagic URSA Cine'nin içinde, kullanıcı tarafından tamir edilebilecek bir parça yoktur. Bakım ve tamirat için bölgenizdeki Blackmagic Design servis merkezine başvurun.

Giriş güç konektörünü harici bir DC batarya kaynağına bağlarken, besleme kabloları elektrik akımı sınırlaması veya elektrik sigortası içermek zorundadır.

Kullanılan kablolar, VW-1 damgalı veya IEC 60332 veya IEC 60695'in ilgili bölümlerine uygun olmalıdır.

Bu kamera için DC kaynağın 24V olmasını öneririz. 12V DC bir kaynak kullanılıyorsa 14A akım için yeterli kablo kullanılmalıdır.

Bu ürünle birlikte B-Yuva bir batarya plakası verilmektedir. 24V (28.8V) veya çift voltajlı bir batarya kullanılmasını tavsiye ederiz.

12V (14,4V) bir bataryanın Maksimum Sürekli Çalışma Akımını belirlemek için bataryanın kılavuzuna veya etiketlerine bakın. Bu akımın asgari 12A gücünde olmasını tavsiye ederiz.

Kaliforniya Eyaleti Beyannamesi

Bu ürün; plastik parçaları dahilinde, eser miktarda polibromine bifenil gibi kimyasal maddelere sizi maruz bırakabilir. Kaliforniya eyaletinde, bu maddelerin kansere, doğum kusurlarına veya başka üreme bozukluklarına sebebiyet verdiği bilinmektedir.

Daha fazla bilgi için, www.P65Warnings.ca.gov adresini ziyaret ediniz.

Avrupa Ofisi

Blackmagic Design Europe B.V.
Rijnlanderweg 766, Unit D
2132 NM Hoofddorp
Hollanda

Blackmagic URSA Cine

Güneşli ortamlarda, kameranın veya Lityum bataryanın, uzun süre boyunca güneş ışığına maruz kalmasını engellemek için, kamerayı gölgede tutmaya dikkat edin. Lityum bataryaları, her türlü ısı kaynağından uzak tutun.

24V 'RS' çıkış konektörü, üçüncü parti aksesuarlara güç sağlamak için uygundur. Güç tüketiminin 48 W'tan az olduğundan emin olun.

Garanti

Sınırlı Garanti

Blackmagic Design şirketi, bu ürünün satın alındığı tarihten itibaren 12 ay boyunca malzeme ve işçilik bakımından arızasız olacağına dair garanti sunar. Bu garanti süresi içinde üründe bir arıza ve kusur olduğu ortaya çıkarsa Blackmagic Design, kendi seçimi doğrultusunda ya arızalı ürünü parça ve işçilik bedeli talep etmeden tamir eder ya da arızalı ürünü yenisiyle değiştirir.

Bu garanti kapsamındaki hizmetten yararlanmak için, kusur ve arızaya ilişkin garanti süresi sona ermeden, Müşteri olarak siz, Blackmagic Design'i bilgilendirmekle ve söz konusu hizmetin sağlanması için uygun düzenlemeleri yapmakla sorumlusunuz. Müşteri, Blackmagic Design tarafından belirlen bir yetkili servis merkezine arızalı ürünü paketleyip, nakliye ücretleri önceden ödenmiş olarak göndermekle yükümlüdür. Herhangi bir sebepten dolayı bize iade edilen ürünlerin; tüm nakliye, sigorta, gümrük vergileri, vergi ve tüm diğer masrafların ödenmesi ile Müşteri yükümlüdür.

Bu garanti; yanlış kullanım ya da yanlış veya kusurlu bakımdan kaynaklanan herhangi bir arızayı, bozukluğu ya da hasarı kapsamaz. Bu garanti kapsamında Blackmagic Design'in aşağıda açıklanan durumlarda zorunluluğu yoktur: a) Blackmagic Design temsilcileri haricindeki başka personelin ürünü kurma, tamir etme ya da bakımını yapma girişimlerinden kaynaklanan hasarın tamir edilmesi, b) uygun olmayan kullanım veya uyumlu olmayan ekipmanlara bağlama sonucu kaynaklanan hasarın tamir edilmesi, c) Blackmagic Design parçaları ya da malzemesi olmayan ürünlerin kullanımından kaynaklanan hasarın ya da arızanın tamir edilmesi, ya da d) Modifiye veya başka ürünlerle entegre edilmiş bir ürünü; söz konusu modifikasyon ya da entegrasyonun, gereken tamiratın süresini uzattığı ya da ürün bakımını zorlaştırdığı durumlarda ürünün servis edilmesi.

URSA Viewfinder'in doğrudan güneş ışığına maruz bırakılması, vizörün optikleri bir büyüteç işlevi göreceğinden, vizör ekranına zarar verebilir. Statik görüntüler veya kare kılavuzları gibi yüksek kontrastlı görüntüler uzun süre boyunca OLED panellerinde gösterildiği zaman, OLED panellerinde görüntü tutma (image retention) veya görüntü izi (burn-in) oluşabilir. Bunu engellemek için, yüz algılama IR sensörünün kasıtlı olarak üstünün örtülmediğinden emin olun ve uzun süre kullanılmadığı durumlarda vizörü çıkarın. Görüntü tutma, garanti kapsamında değildir.

BU GARANTİ, BLACKMAGIC DESIGN TARAFINDAN AÇIK YA DA ZIMNİ, HERHANGİ BİR GARANTİNİN YERİNİ TUTAR. BLACKMAGIC DESIGN VE SATICILARI, TİCARİ ELVERİŞLİLİK YA DA ÖZEL BİR AMACA UYGUNLUK KONUSUNDA HERHANGİ BİR ZIMNİ GARANTİ VERMEZ. BLACKMAGIC DESIGN'İN BU GARANTİ SÜRESİ İÇİNDE HASARLI ÜRÜNLERİ TAMİR ETME YA DA DEĞİŞTİRME SORUMLULUĞU, MÜŞTERİYE SUNACAĞI TEK VE MÜNHASIR ÇÖZÜMDÜR. BLACKMAGIC DESIGN YA DA SATICILARININ OLABİLECEK HASARLAR HAKKINDA ÖNCEDEN BİLGİSİ OLMASINI GÖZETMEKSİZİN; ÜRÜNDE DOLAYLI, ÖZEL, TESADÜFİ YA DA NETİCE OLARAK ORTAYA ÇIKAN HERHANGİ BİR HASAR İÇİN BLACKMAGIC DESIGN SORUMLU DEĞİLDİR. MÜŞTERİLER TARAFINDAN EKİPMANIN YASAL OLMAYAN HERHANGİ BİR KULLANIMINDAN, BLACKMAGIC DESIGN SORUMLU DEĞİLDİR. BLACKMAGIC DESIGN, BU ÜRÜNÜN KULLANIMINDAN KAYNAKLANAN HERHANGİ BİR HASARDAN, SORUMLU DEĞİLDİR. BU ÜRÜNÜN ÇALIŞTIRILMASINDAN DOĞAN RİSK, KULLANICININ KENDİSİNE AİTTİR.

© Telif Hakkı Saklıdır 2025 Blackmagic Design. Tüm Hakları Saklıdır. "Blackmagic Design", "URSA", "DeckLink", "HDLink", "Workgroup Videohub", "Multibridge Pro", "Multibridge Extreme", "Intensity" ve "Leading the creative video revolution" ABD ve diğer ülkelerde tescil edilmiş, ticari markalardır. Diğer tüm şirket ve ürün isimleri, bağlantılı oldukları ilgili şirketlerin ticari markaları olabilir.

Bluetooth® kelime işareti ve logosu, Bluetooth SIG, Inc.'e ait tescilli ticari markalardır ve bu tür işaretlerin Blackmagic Design tarafından herhangi bir kullanımı lisansa tabiidir. Diğer ürün isimleri ve şirket isimleri, bağlantılı oldukları ilgili şirketlerin ticari markaları olabilir.

Październik 2025

Przewodnik użytkownika kamery

Blackmagicdesign

Blackmagic URSA Cine Immersive



Spis treści

Blackmagic URSA Cine Immersive 1677
Immersyjny – co to znaczy? 1677
Apple Immersive Video 1677
URSA Cine Immersive 1678
Rozpoczęcie pracy 1679
Zasilanie kamery 1679
Ustawienia zapisu 1680
Monitorowanie obrazu 1682
Opcje monitorowania i nakładki 1682
Kadrowanie pierwszego ujęcia 1683
Wysokość kamery 1683
Przechylenie kamery 1684
Pozycja 1685
Oddziaływanie na widzów 1686
Ruch kamery 1686
Tabele prędkości transmisji danych 1687
Audio i kod czasowy 1688
Apple Spatial Audio Format 1688
Mikrofony ambisoniczne 1688
Mikrofony krawatowe i miniaturowe 1689
Mikrofony ręczne 1689
Podłączanie kodu czasowego 1689
Filmowanie 1691
Wymagania dotyczące ekspozycji 1691
Migawka 1691
ISO i zakres dynamiki 1692
ND 1693
Zapis off speed 1694
Histogram 1695
Opcje podglądu LCD 1695
Nagrywanie 1696
Odtwarzanie 1697
Transfer klipów 1698
Podsumowanie 1699
DaVinci Resolve Studio 1700
Zalecenia dotyczące stacji roboczej 1700
Ustawienia projektu 1701
Ustawienia główne 1701
Monitorowanie wideo 1702
Skalowanie obrazu 1702
Zarządzanie kolorem 1703
Informacje dotyczące DaVinci Wide Gamut 1704
Przetwarzanie Apple Immersive 1705
Pobieranie danych 1705
Montaż 1707
Zestawienie 1707
Tła 1707
Korzystanie z okna montażowego 1709
Przejścia 1711
Apple Immersive Motion Data 1712
Nagłówki 1713
Wskazówki dotyczące kreatywności 1714
Kompozytowanie VFX 1715
Węzły MediaOut 1715
Renderer3D 1716
Kompozytowanie za pomocą Immersive Patcher 1717
Korekcja kolorów 1718
Redukcja szumów 1718
Podstawowa korekcja kolorów 1719
Korekcja wtórna 1719
Zmiana rozmiaru 1720
Paleta 3D 1720
Zarządzanie kolorem CST 1722
Mapowanie osi czasu CST 1722
Mapowanie CST monitora referencyjnego 1723
Monitorowanie CST Apple Vision Pro 1725
Audio 1726
Apple Spatial Audio Format (ASAF) 1726
Dystrybucja 1727
Pomijanie przejść 1727
Streaming do Apple Vision Pro 1728
Informacje dotyczące Apple Immersive Video Utility 1728
Developer Information 1729
Camera Control REST API 1729
Livestream Control API 1731
Clips Control API 1735
Media Pool Control API 1736
Monitoring Control API 1739
Event Control API 1748

System Control API 1748
Transport Control API 1754
Timeline Control API 1758
Media Control API 1761
Slate Control API 1764
Preset Control API 1770
Audio Control API 1772
Lens Control API 1777
Video Control API 1781
Camera Control API 1789
Immersive Control API 1792
Color Correction Control API 1793
Notification websocket - 1.0.0 1798
messages 1798
Pomoc 1822
Wymogi prawne 1823
Informacje dotyczące bezpieczeństwa 1824
Gwarancja 1825

Blackmagic URSA Cine Immersive

Gratulujemy zakupu lub wypożyczenia Blackmagic URSA Cine Immersive.

Blackmagic URSA Cine Immersive zaprojektowano do filmowania stereoskopowego materiału VR 180 stopni, powszechnie nazywanego immersyjnym wideo. W szczególności kamerę zoptymalizowano do przechwytywania Apple Immersive Video dla Apple Vision Pro, najwyższej jakości doświadczenia immersyjnego, jakie jest obecnie dostępne.

Niniejszy przewodnik został stworzony jako wprowadzenie do Blackmagic URSA Cine Immersive i filmowania przy jej użyciu, skupiając się na podstawowym zakresie działania kamery i dostarczanym przez nią wyjątkowym formacie: Apple Immersive Video. Zrozumienie formatu pozwoli lepiej zrozumieć samą kamerę.

Na kolejnych stronach znajdują się techniczne i kreatywne porady, które nie tylko ułatwią korzystanie z kamery, ale też pomogą zrozumieć korzyści związane z immersyjnym formatem.

Immersyjny – co to znaczy?

Immersyjny to trójwymiarowy, przestrzenny format wideo. Najpewniej zdajesz sobie sprawę, że podobne technologie istnieją już od jakiegoś czasu. Aby zrozumieć, czym wyróżnia się format immersyjny, zamieściliśmy przegląd innych technologii:

- **Tradyjne kino 3D** pozwala widzom doświadczyć głębi w kadrze dzięki przechwytywaniu każdej sceny dwoma kamerami ustawionymi od siebie w odległości imitującej rozstaw ludzkich oczu. Choć stereoskopowe obrazy pozwalają widzom poczuć głębię, ich perspektywa jest kontrolowana przez kompozycję filmowca.
- **Film sferyczny** łączy działanie kamer 3D i 360°, umożliwiając widzom patrzenie w dowolnym kierunku przy jednoczesnym doświadczaniu głębi obrazu. Obrazy 360° wymagają bardzo wysokich rozdzielczości, aby niewielki fragment pola widzenia odbiorcy zawierał wystarczającą wierność odwzorowania. Fizjologia ludzkiego ciała pozwala obrócić głowę od ramienia do ramienia. Aby móc spojrzeć za siebie, widzowie zazwyczaj stoją lub siedzą na krześle obrotowym.
- **Film wolumetryczny** pozwala widzom nie tylko doświadczać głębi i rozglądać się wokół siebie, ale też przemieszczać i zmieniać perspektywę. Zestawy kamer mogą przechwytywać film wolumetryczny, ale większość rozwiązań do rzeczywistości wirtualnej lub rozszerzonej korzysta z dynamicznie renderowanych zasobów 3D i wymaga bezpiecznego środowiska dla uczestników.

Format immersyjny łączy to, co najlepsze w kinie tradycyjnym i interaktywnym kinie 3D. W najprostszej formie jest to format stereoskopowy 180° o wysokim klatkażu oraz wysokiej rozdzielczości.

Apple Immersive Video

Blackmagic URSA Cine Immersive zaprojektowano z myślą o przechwytywaniu Apple Immersive Video. Ten format jest używany z Apple Vision Pro i obsługuje wysokiej jakości wideo stereoskopowe 3D w polu widzenia 180°.

Charakterystyka Apple Immersive Video:

- Odtwarzanie z prędkością 90 kl./s.
- Możliwa dystrybucja do 4320 x 4320 na oko.
- Unikatowy system metadanych do mapowania oryginalnego obrazu z kamery do środowiska 180°.
- Apple Spatial Audio Format zapewnia dźwięk 3D.

Apple Immersive Video wymaga sprzętu będącego w stanie odtwarzać wysokiej jakości stereoskopowe wideo 180°. Apple Vision Pro to obecnie jedyne urządzenie obsługujące ten format.



Apple Vision Pro

Zestaw słuchawkowy posiada podwójne mikrowyświetlacze OLED 3660 x 3200 z rozstawem pikseli wynoszącym 7,5 mikrona. Są wyposażone w 13 kamer śledzących, skaner LiDAR, wewnętrzne jednostki pomiarowe, sensory śledzące ruch oka do synchronizacji obrazów z ruchem głowy oraz oczu.

Procesor M2 z 8-rdzeniowym CPU i 10-rdzeniowym GPU zapewnia odtwarzanie z prędkością do 100 kl./s.

URSA Cine Immersive

Blackmagic URSA Cine Immersive zaprojektowano do przechwytywania Apple Immersive Video dla Apple Vision Pro.



Blackmagic URSA Cine Immersive

URSA Cine Immersive wykorzystuje dwa identycznie skalibrowane sensory RGBW o rozdzielczości 58 megapikseli. Sensory są niemal kwadratowe, aby przechwycić pełne pole widzenia 180°, w pionie oraz poziomie. Każdy sensor posiada obiektyw 210° typu rybie oko. Centralne punkty obiektywów są oddalone o 64 mm.

Wbudowane filtry ND zapewniają redukcję światła do 8 stopni. Wartość przysłony wynosi f4.5.

Kamera nagrywa dwa strumienie wideo 8K z prędkością 90 kl./s. To zapewnia znacznie szybszy transfer danych niż w przypadku standardowego zapisu 4K 24p.

Rozpoczęcie pracy

Rozpoczęcie pracy z Blackmagic URSA Cine Immersive jest bardzo proste: wystarczy podłączyć zasilanie i włączyć kamerę. Blackmagic Media Module 8TB jest włożony do wnęki modułowej kamery, sformatowany i gotowy do natychmiastowego nagrywania klipów.

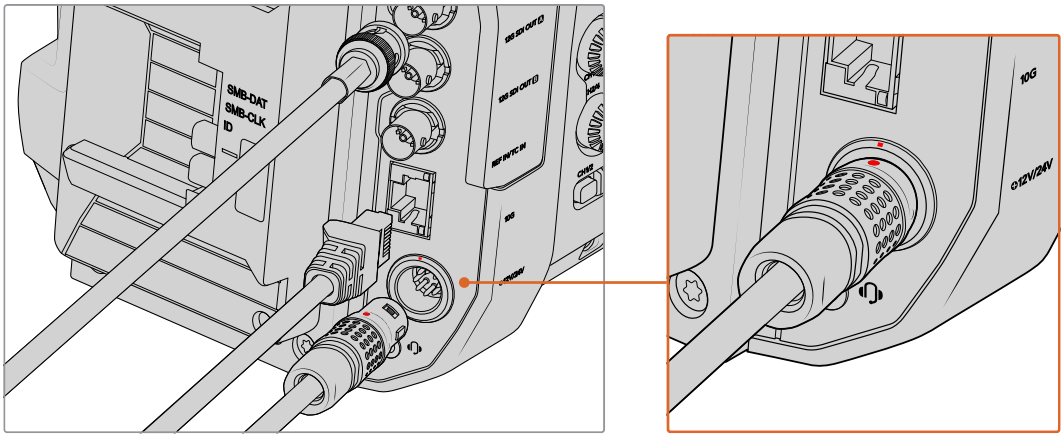
Zasilanie kamery

Najszybszym sposobem zasilania kamery jest podłączenie zasilania zewnętrznego za pomocą dostarczonego adaptera.

Aby podłączyć zasilanie zewnętrzne:

1. Najpierw zdejmij gumową zaślepkę i odłóż ją w bezpieczne miejsce. Podłącz wtyczkę adaptera AC do 24 V DC do gniazdka sieciowego przy użyciu standardowego kabla zasilającego IEC C13.
2. Podłącz 8-pinowe złącze adaptera do złącza zasilania 24 V DC na tylnym panelu kamery.

Jeśli podłączone jest zasilanie zewnętrzne i akumulatorowe, kamera będzie pobierać energię z zewnętrznego źródła zasilania. Po odłączeniu zasilania zewnętrznego, gdy podłączony jest naładowany akumulator, kamera przełączy się na zasilanie akumulatorowe bez żadnych zakłóceń.



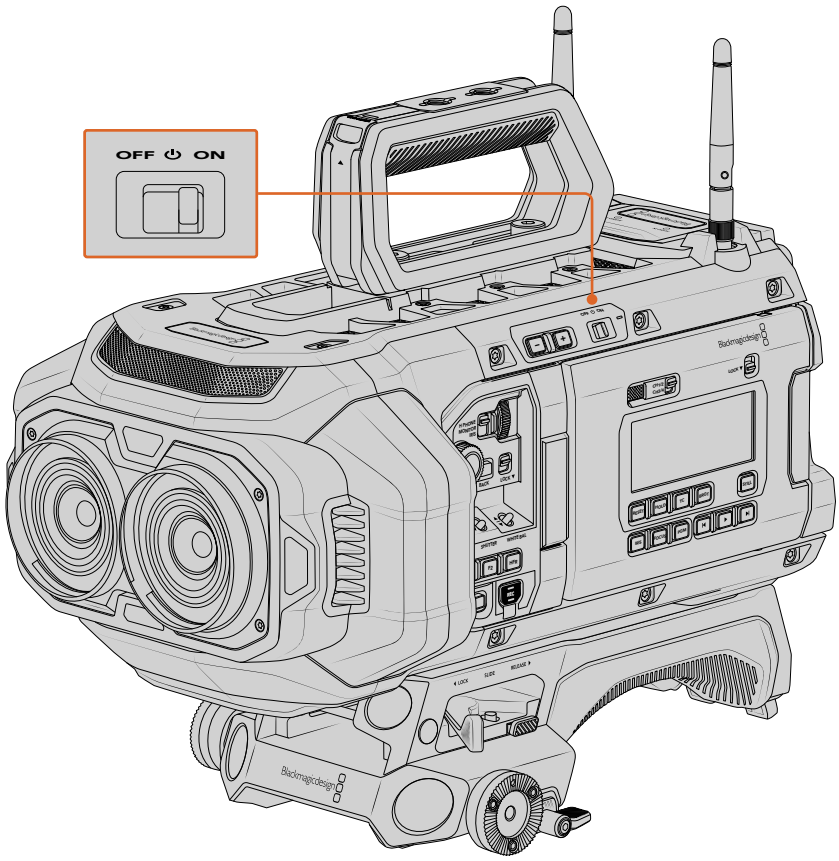
Aby podłączyć zasilanie, wyrównaj czerwone kropki i delikatnie wciśnij złącze, aż poczujesz, że jest na swoim miejscu. W celu odłączenia należy chwycić metalową tuleję złącza i delikatnie wysunąć ją z kamery. Spowoduje to cofnięcie tulei i zwolnienie złącza.

Płyta akumulatora dołączona do kamery to płyta z mocowaniem typu B przeznaczona do akumulatorów 24 V z mocowaniem tego typu. Zakres napięcia od 24 do 34 V jest idealny do stosowania z URSA Cine Immersive, ponieważ wyższe napięcie zmniejsza straty termiczne, umożliwiając nagrywanie z wysokim klatkażem i zasilanie wielu akcesoriów z kamery.

Dostępne są również opcjonalne płyty VLock i Gold Mount kompatybilne z 24 V oraz Gold Mount Plus do akumulatorów 26 V innych producentów.

UWAGA Zasilając URSA Cine Immersive zasilaczami lub akumulatorami o mocy od 12 do 18 V, zasilanie akcesoriów przy pomocy 3-pinowego złącza RS lub 2-pinowego złacza Lemo nie jest możliwe.

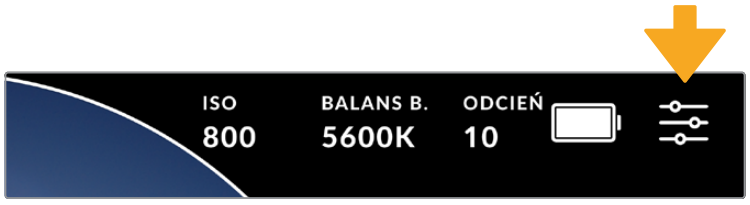
Po zamocowaniu akumulatora lub podłączeniu zewnętrznego zasilania, włącz URSA Cine Immersive. Ustaw przełącznik na górze kamery w pozycji **ON**.



Ustaw przełącznik w pozycji **ON**, aby włączyć kamerę

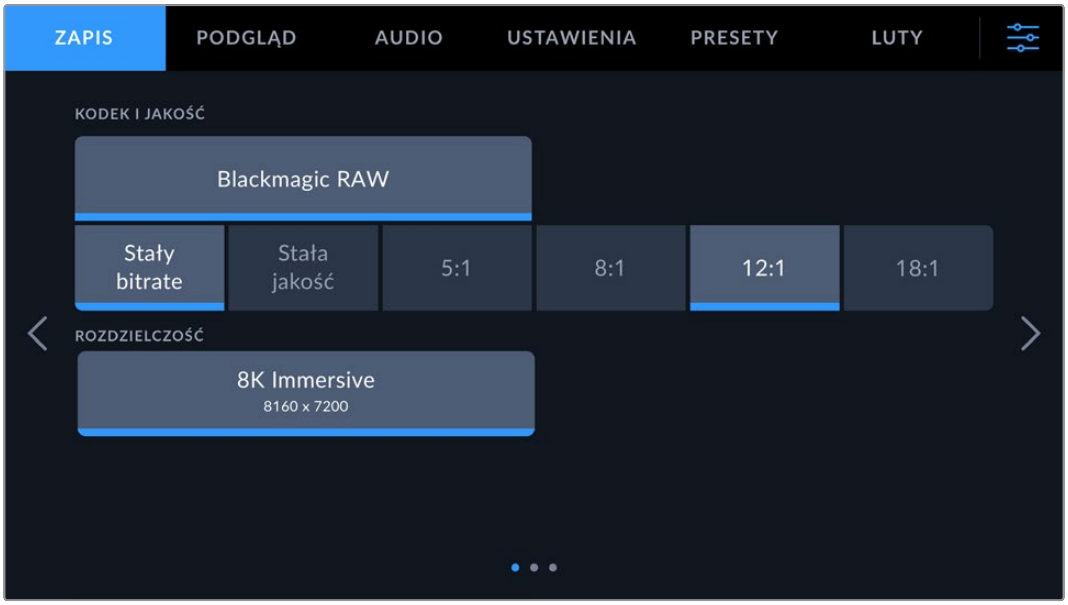
Ustawienia zapisu

Ta sekcja zawiera informacje na temat ustawień kamery, które są szczególnie ważne lub istotne podczas filmowania treści immersyjnych. Aby otworzyć menu ustawień kamery, stuknij ikonę menu ustawień w prawym górnym rogu rozkładanego ekranu lub panelu pomocniczego.



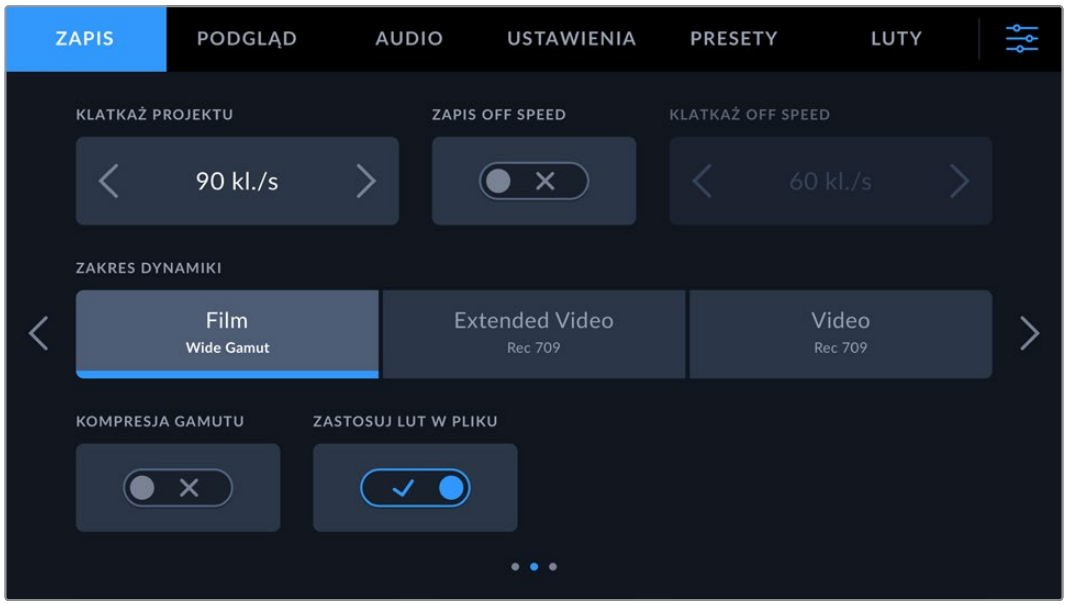
Zapis – Strona 1

Opcje ustawienia **Kodek i jakość** na tej stronie pozwalają wybrać ustawienie kodeka Blackmagic RAW, tak aby był zgodny z projektem, nad którym pracujesz. Podzielono je na 4 ustawienia **Stały bitrate** i **Stała jakość**. Zalecamy korzystanie z ustawienia 12:1 lub kompresji Q3 podczas nagrywania treści z prędkością 90 kl./s dla Apple Vision Pro. Zapewni to optymalne ustawienia w celu utrzymania łatwych do zarządzania prędkości transmisji i wielkości plików przy pomocy posiadanego komputera, zachowując jednocześnie obraz o niezwykle wysokiej jakości.



Zapis – Strona 2

Druga strona zakładki **ZAPIS** zawiera ustawienia klatkażu.



Klatkaż projektu kamery powinien być ustawiony na 90 kl./s, aby odpowiadać klatkażowi wyświetlacza Apple Vision Pro.

Jeśli nagrywasz na potrzeby innych systemów, wystarczający może być niższy klatkaż, taki jak 60 lub 50 kl./s. Niższe klatkaże w stylu klasycznych filmów, takie jak 24 lub 25 kl./s mogą być odpowiednie, jeśli nagrywasz tła efektów wizualnych na potrzeby treści do filmu 24p.

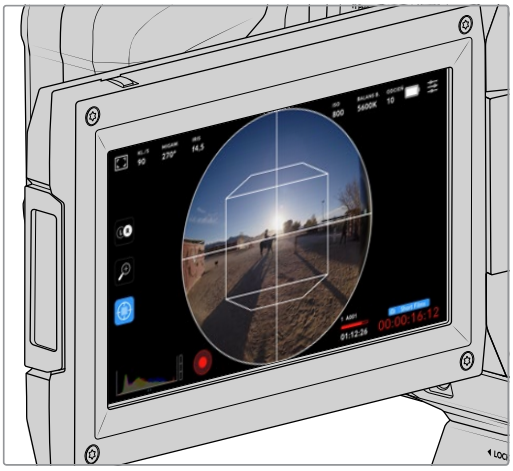
To wystarczy, aby rozpocząć pracę z kamerą URSA Cine Immersive. Moduł multimedialny jest już sformatowany w kamerze, więc możesz od razu rozpocząć nagrywanie.

Monitorowanie obrazu

URSA Cine Immersive ma kilka opcji pozwalających na monitorowanie obrazu w trakcie filmowania.

Rozkładany ekran dotykowy jest przeznaczony dla operatora. Można go odwrócić, aby pozostał widoczny, ale nie zajmował zbyt dużo przestrzeni lub ustawić go przodem, umożliwiając podgląd obrazu podczas filmowania samego siebie.

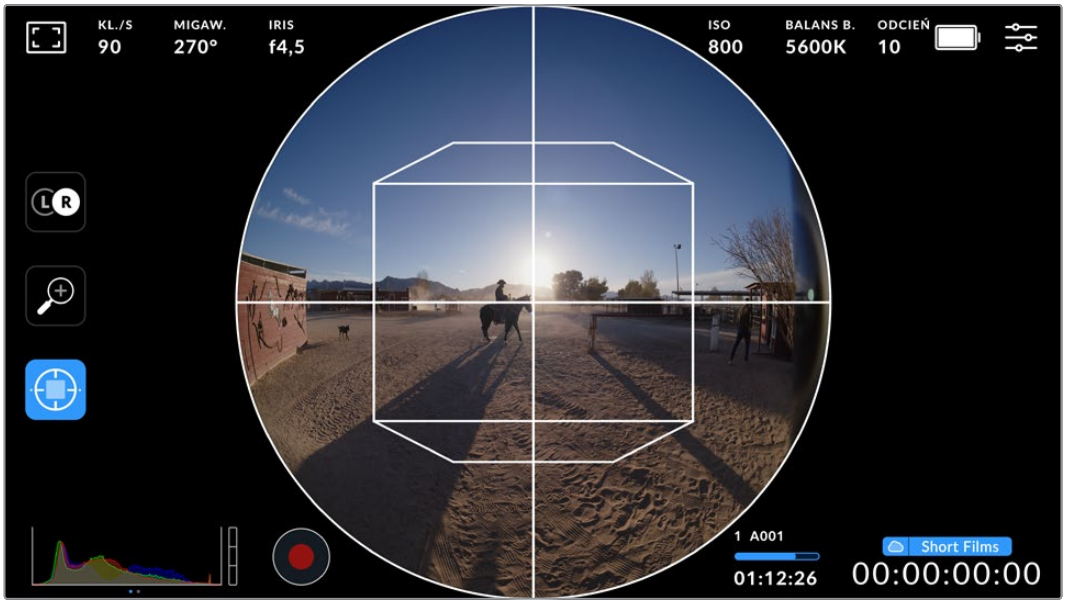
Panel pomocniczy pozwala innym członkom ekipy monitorować obraz bez przesłaniania widoku operatorowi. Ze względu na niezwykle szerokie pole widzenia kamery należy zachować szczególną ostrożność podczas korzystania z panelu pomocniczego.





Opcje monitorowania i nakładki

Kamera wyświetla przestrzeń obiektywu, pokazując widok dla jednego oka. Rozkładany ekran i panel pomocniczy są złożone, aby umożliwić widok 180 stopni. Wyjścia SDI pokazują pełny obraz 210°, przechwycony przez każdy z sensorów.



Wszystkie wyjścia monitorowania wyświetlają pogląd dla tego samego oka. Możesz przełączyć między lewym i prawym okiem, stukając przycisk **L R** na jednym z ekranów dotykowych.

Można włączyć dodatkową nakładkę na ekranach dotykowych, naciskając przycisk nakładki. Ta nakładka zawiera:

- Ograniczony obszar zewnętrzny wynoszący 180°
- Celownik wyznaczający środek kadru
- Wewnętrzny obszar reprezentujący przybliżone pole widzenia widoczne na Apple Vision Pro, gdy użytkownik patrzy przed siebie.

Te narzędzia zaprojektowano w celu ułatwienia precyzyjnego kadrowania i pozycjonowania w trakcie przechwytywania immersyjnego wideo.

Kadrowanie pierwszego ujęcia

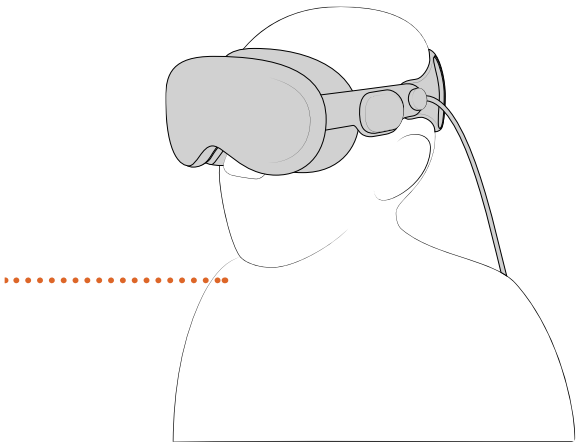
Treści immersyjne i 2D wiele łączy, ale istnieją także znaczące różnice.

Na przykład, czy kiedykolwiek dopadła cię choroba lokomocyjna w samochodzie, bo byłeś zbyt zajęty przeglądaniem telefonu? Przyczyną jest niezgodność sensoryczna, gdy mózg otrzymuje sprzeczne sygnały. Ucho wewnętrzne informuje ciało, że jesteś w ruchu, ale oczy przekazują do mózgu informacje o tym, że siedzisz nieruchomo.

Podobne wrażenie można uzyskać przy pomocy technik immersyjnych. Na przykład, filmując panoramę, oczy widzów będą ich informować o tym, że się przemieszczają, ale ich szyje oraz ucho wewnętrzne będą przekonywać, że trwają w bezruchu. Z tego powodu najlepiej opanować najpierw ujęcia statyczne, a dopiero potem przejść do nauki o tym, jak sprawić, aby ruch kamery był przyjemny dla odbiorców.

Wysokość kamery

W przypadku tradycyjnej kamery 2D możesz umieścić ją wyżej lub niżej, aby zaznaczyć przewagę sił w danej scenie. Filmując immersyjny obraz, zastanów się nad tym, jaki efekt uzyskasz, kładąc kamerę na ziemi. Jak to wpłynie na odbiorcę? Może mieć wrażenie, że jest zakopany w ziemi aż po szyję.



Postrzegana przez widza wysokość podłoża może mieć wpływ na jego doświadczenie. Nisko umieszczona kamera oferuje odbiorcom widok na takiej wysokości podłoża, która nie ma sensu.



Przechwytując immersyjne obrazy, ujęcia pod niskim kątem mogą sprawić, że widzowie poczują się niekomfortowo

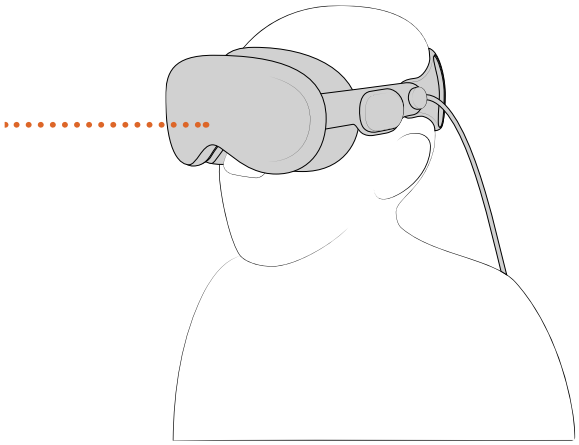
Aby mogli poczuć się fizycznie obecni, często najlepszym wyborem jest ustawienie i skierowanie kamery tak, aby widzowie mieli poczucie, że w danej scenie stoją albo siedzą. Istnieje jeszcze wiele innych sposobów, abyś mógł wyrazić swój twórczy zamysł. Jeśli w scenie pojawiają się inne osoby, kamera umieszczona nieco wyżej może napełnić widza poczuciem władzy, pozwalając mu patrzeć na pozostałych z góry. Niżej umieszczona kamera z kolei zmusi go do patrzenia w górę na innych ludzi.

To nie oznacza jednak, że zawsze odbieramy widzom komfort, gdy ustawiamy kamerę w bardziej niestandardowy sposób. Latające kamery, ujęcia z góry, niskie kąty i wiele innych technik z powodzeniem wykorzystuje się w produkcjach immersyjnych. W połączeniu z odpowiednim montażem, ruchem i kontekstem – takie obrazy mogą zapewnić rozrywkę widzom.

Przechylenie kamery

Filmując tradycyjne treści 2D lub umieszczając kamerę wyżej lub niżej, operator często przechyli kamerę, aby zachować kompozycję. Zastanów się, jak to wpłynie na immersyjny obraz.

Jeśli skierujesz kamerę do dołu pod kątem, widzowie musieliby przekrzywić głowę do przodu, aby zobaczyć horyzont. Nie wywołałoby to choroby lokomocyjnej, ale mogłoby przerwać poczucie uczestnictwa, ponieważ w rzeczywistości horyzont jest zawsze bezpośrednio przed nami.



Immersyjne ujęcia będą miały najwięcej sensu, jeśli horyzont osiąga wartość 0 stopni.



Jeśli przechylisz URSA Cine Immersive, widzowie odczują rozdźwięk między tym, gdzie muszą wypatrywać horyzontu, a gdzie normalnie powinien się znajdywać.

Z tego powodu większość ujęć immersyjnych przechwytuje się przy przechyleniu osiągającym 0 lub niemal 0 stopni. Ponieważ kamera przechwytuje pole widzenia 180 stopni, w pionie oraz w poziomie, widzowie nadal mogą patrzeć w górę oraz w dół. Jednak filmowiec nie zmusza widzów do patrzenia w konkretnym kierunku, to oni kontrolują sytuację.

Aby zachęcić widzów do zwrócenia uwagi na inne obszary obrazu, możesz użyć technik, takich jak oświetlenie, kadrowanie przysłoną lub linii naprowadzających.

Pozycja

Immersyjne wideo najczęściej ogląda się na siedząco. Nie jest to rzeczywistość rozszerzona ani wirtualna, więc widzowie nie wstają i nie przemieszczają się. Czy w tym momencie siedzisz? W ramach eksperymentu poruszaj głową na boki oraz z góry na dół. Jaki jest komfortowy zakres ruchu Twojej głowy bez poruszania resztą ciała? Obszar widzenia formatu immersyjnego przekracza komfortowy zakres ruchu głowy większości ludzi. To dobra informacja, ponieważ, jeśli obrócisz się w lewo lub w prawo, format zapewni widok peryferyjny. Ten eksperyment pomaga w zrozumieniu, gdzie powinna koncentrować się główna akcja w ramach pola widzenia.

Typowy zakres ruchu głowy:

Obrót: 120 do 160 stopni
Pochylenie i odchylenie: 110 do 130 stopni

Gdy obrót głowy jest połączony z ruchem oczu, możliwe jest zerknięcie przez własne ramię, ale dla większości osób utrzymanie takiej pozycji powoduje dyskomfort.

Aby lepiej zaplanować kompozycję, włącz nakładkę kamery. Utrzymanie akcji w ramach wewnętrznych znaczników pokazujących typowe pole widzenia wspomoże widoczność treści.

Należy także rozważyć przemieszczanie punktów zainteresowania w ramach kadru. Śledzenie osoby, która przemieszcza się z lewego krańca na prawy wymaga intensywnego ruchu głową. Podobnie, przejście z ujęcia, gdzie punkt zainteresowania znajduje się na skraju kadru, do ujęcia z punktem zainteresowania po przeciwnej stronie również wymaga skrajnego obrotu głowy. Dokładne przemyślenie kompozycji nie przekreśla kreatywnych form wyrazu, ale może zwiększyć komfort widzów.

Oddziaływanie na widzów

Filmując tradycyjne treści 2D, możesz wykorzystać kadr, obiektyw lub ustawienia ogniskowej, aby ukierunkować uwagę widzów. W przypadku filmu immersyjnego pole widzenia 180 stopni utrudnia selekcję informacji i przyciągnięcie uwagi odbiorcy. Format immersyjny umożliwia odbiorcom zgłębianie Twoich obrazów w o wiele większym stopniu. To jeden z czynników, które sprawiają, że treści immersyjne są tak wciągające. Nadal możesz wpływać na uwagę widzów poprzez przemyślane użycie dźwięku kierunkowego, oświetlenia, ustawienia aktorów lub kompozycji w kadrze.

Ruch kamery

Kamera immersyjna nie musi być nieruchoma, ale należy zachować szczególną ostrożność, bo niektóre ruchy kamery mogą sprawiać wrażenie nienaturalnych lub wywoływać dyskomfort. Samochód to świetny przykład, który pomoże zrozumieć jak sprawić, aby ruch kamery immersyjnej był przyjemny. Duży pojazd charakteryzuje masa i bezwładność, które opierają się gwałtownym zmianom prędkości oraz kierunku. W związku z tym będzie przyśpieszać oraz zwalniać w trakcie jazdy i poruszać się w przewidywany, pozostający pod kontrolą sposób. Po opanowaniu ujęć statycznych i przejściu do immersyjnych zdjęć w ruchu, wzięcie pod uwagę następujących czynników pomoże stworzyć przyjemne wrażenie ruchu:

- **Czas trwania:** Daj widzom czas, aby przyzwyczaili się do ruchu kamery i poczuli komfort. Krótki, szybki ruch częściej wywołuje dezorientację.
- **Intensywność:** Ruchy o niskiej intensywności łatwiej zintegrować z immersyjnym materiałem. Intensywne ruchy są dopuszczalne, ale brak drastycznych i nagłych zmian w dynamice obrazu może zwiększyć komfort widzów.
- **Wizualne punkty odniesienia:** To elementy sceny, które pozostają nieruchome. Ujęcie bez wizualnych punktów odniesienia może dezorientować widza. Elementy pierwszego planu (wnętrze auta) lub te w oddali (horyzont) pomogą ustabilizować ujęcia w ruchu.

Tabele prędkości transmisji danych

Tabele poniżej zawierają prędkości transmisji danych dla treści immersyjnych 8K x 7K przy 90 kl./s i 24 kl./s.

Maksymalna prędkość transmisji danych przy 90 kl./s z Blackmagic RAW Constant Bitrate

Immersyjny 8K x 7K			
Kodek	**Wymiary piksela**	**Klatkaż**	**Prędkość transmisji**
Blackmagic RAW 5:1	8160 x 7200 Stereo	90	3203 MB/s
Blackmagic RAW 8:1	8160 x 7200 Stereo	90	2002 MB/s
Blackmagic RAW 12:1	8160 x 7200 Stereo	90	1335 MB/s
Blackmagic RAW 18:1	8160 x 7200 Stereo	90	889 MB/s

Maksymalna prędkość transmisji danych przy 24 kl./s z Blackmagic RAW Constant Bitrate

Immersyjny 8K x 7K			
Kodek	**Wymiary piksela**	**Klatkaż**	**Prędkość transmisji**
Blackmagic RAW 5:1	8160 x 7200 Stereo	24	854 MB/s
Blackmagic RAW 8:1	8160 x 7200 Stereo	24	534 MB/s
Blackmagic RAW 12:1	8160 x 7200 Stereo	24	356 MB/s
Blackmagic RAW 18:1	8160 x 7200 Stereo	24	237 MB/s

Audio i kod czasowy

Dźwięk to kluczowy aspekt immersyjnego doświadczenia. To ważne, aby używać URSA Cine Immersive z dedykowanym sprzętem do przechwytywania dźwięku.

Apple Spatial Audio Format

Apple Spatial Audio Format lub ASAF odpowiada za dźwięk 3D podczas oglądania Apple Immersive Video. ASAF to część szerszego Apple Positional Audio Framework i może przechowywać trzy różne typy nagrań dźwięku.

- **Oparty na kanale:** Możesz osadzić tradycyjny miks przestrzenny w pliku ASAF. To idealne rozwiązanie w przypadku narracji powiązanej z ruchem głowy lub muzyki.
- **Oparty na obiekcie:** Dźwięk można traktować jako osobne obiekty z metadanymi, które opisują ich pozycję w przestrzeni 3D, niezależnie od poszczególnych kanałów głośnika. Są to zazwyczaj nagrania monofoniczne. Ich współrzędne 3D można animować tak, aby śledziły obiekty poruszające się w danej scenie. To idealne rozwiązanie w przypadku dźwięku diegetycznego, który jest powiązany z obiektami widocznymi w danej scenie.
- **Ambisoniczny:** W pełni sferyczny format przestrzenny, który może renderować dźwięk we wszystkich kierunkach przy pomocy harmonik sferycznych. To idealnie rozwiązanie dla dźwięków pochodzących z otoczenia.

Dalsza część tej sekcji zawiera przegląd sprzętu niezbędnego do stworzenia efektywnego pliku ASAF.

Mikrofony ambisoniczne

Mikrofon ambisoniczny zawiera cztery lub więcej kapsuł ułożonych w precyzyjny, geometryczny wzór. Każda kapsuła rejestruje te same dźwięki, które różnią się czasem dotarcia oraz natężenia, w zależności od kierunku, z którego pochodzą. Te kapsuły są nagrywane na osobne kanały w celu dalszej obróbki na komputerze. Porównując różnice, oprogramowanie jest w stanie określić kierunek i typ nagranych dźwięków.



Mikrofon ambisoniczny pierwszego rzędu

Mikrofony ambisoniczne

Mikrofony ambisoniczne klasyfikuje się ze względu na liczbę kapsuł, które zawierają. Większa ich liczba oznacza, że mikrofon może odtworzyć szczegóły przestrzenne z większą precyzją.

Mikrofon pierwszego rzędu:	4 kanały
Mikrofon drugiego rzędu:	9 kanałów
Mikrofon trzeciego rzędu:	16 kanałów
Mikrofon czwartego rzędu:	25 kanałów i więcej

Niezależnie czy znajdują się na kamerze, czy z boku, należy pamiętać, że mikrofony ambisoniczne są ustawione zgodnie z kierunkiem przechwytywania obrazu przez kamerę. Ponieważ mikrofony ambisoniczne mogą być używane w różnych konfiguracjach – na wprost, w stronę głównego źródła dźwięku, do góry lub do dołu – upewnij się, że Twój inżynier dźwięku wie, która konfiguracja została użyta. Tę informację można przekazać przy pomocy metadanych na rejestratorze dźwięku lub poprzez przekazanie dodatkowych uwag.

Nagrania ambisoniczne mogą być używane jako tło środowiskowe. Można na nie nałożyć inne efekty dźwiękowe oparte na kanałach i obiektach. Jeśli nagrania ambisoniczne są odpowiednio ustawione względem kamery, mogą służyć jako punkt odniesienia dla inżyniera dźwięku podczas dalszego projektowania dźwięku.

Mikrofony krawatowe i miniaturowe

Filmując kamerą 180 stopni, użycie mikrofonu na wysięgniku jest niemożliwe, bo byłby widoczny w scenie! To oznacza konieczność korzystania z mikrofonów ukrytych i krawatowych do rejestrowania dźwięku diegetycznego oraz dialogów.

Możesz na przykład:

- Podłączyć te źródła dźwięku bezpośrednio do dwóch wejść XLR URSA Cine Immersive przy pomocy bezprzewodowych przekaźników i odbiorników.
- Podłączyć je do zewnętrznego rejestratora dźwięku: dużego, wielokanałowego urządzenia lub miniaturowego urządzenia, dającego się umieścić na ciele.

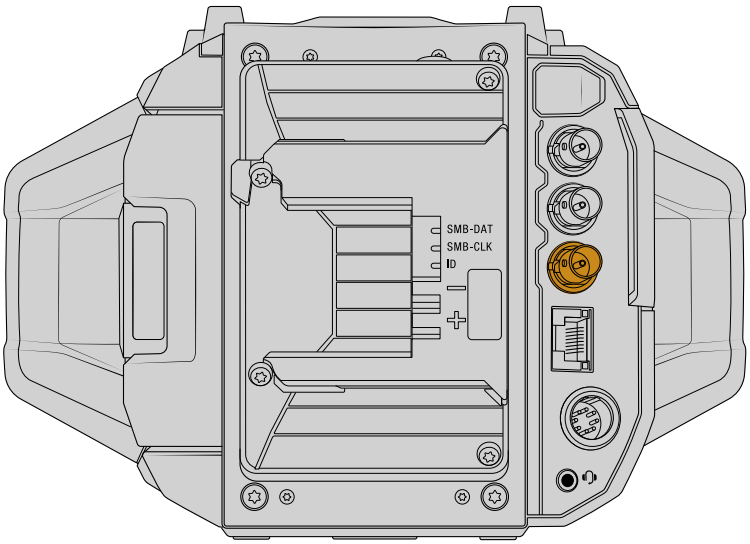
Te nagrania można następnie zsynchronizować i zmiksować jako dźwięk oparty na kanale lub obiekcie.

Mikrofony ręczne

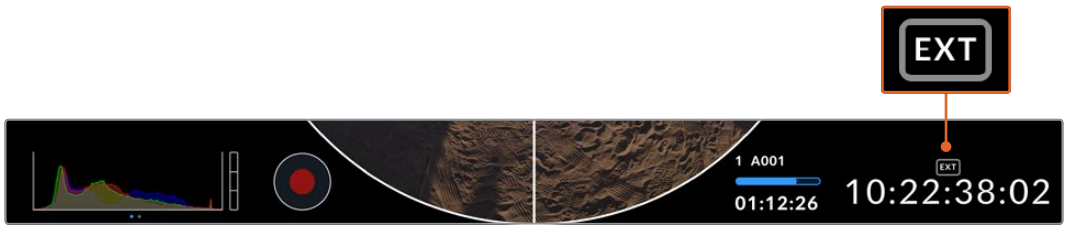
Nagrywanie dodatkowych efektów dźwiękowych na miejscu po wyłączeniu kamery pomoże nie tylko zaoszczędzić czas na etapie postprodukcji, ale również uzyskać bardziej autentyczny dźwięk. Posiadanie ręcznego mikrofonu hiperkardioidalnego lub podobnego sprzętu w trakcie produkcji jest bardzo pomocne: umożliwia powtórki akcji z mikrofonem ustawionym jak najbliżej źródła dźwięku. Takie nagrania można następnie dodać do końcowego miksu jako dźwięki oparte na kanale lub obiekcie.

Podłączanie kodu czasowego

Aby podłączyć zewnętrzny kod czasowy do URSA Cine Immersive, użyj kabla BNC do podłączenia generatora kodu czasowego do wejścia kodu czasowego na tylnym panelu kamery. Dzięki podłączeniu do zsynchronizowanego kodu czasowego podczas nagrywania audiowizualnego dźwięk i obraz można łatwo zsynchronizować podczas postprodukcji.



Gdy do tego złącza zostanie podłączony zewnętrzny kod czasowy o zgodnym klatkażu, kamera automatycznie dostosuje się do przychodzącego kodu czasowego. Na wyświetlaczu LCD URSA Cine Immersive pojawi się napis **EXT** obok licznika kodu czasowego, gdy zewnętrzny kod czasowy jest zablokowany. Jeśli odłączysz kabel, kod czasowy pozostanie zablokowany, a ikona **EXT** zmieni się na **INT**. To oznacza, że działa w oparciu o wewnętrzny zegar kodu czasowego kamery.



Kod czasowy kamery wyświetli ikonę **EXT** po podłączeniu do generatora kodu czasowego

UWAGA Dla typowych klatkaży, takich jak 23,976, 24, 30 itd., istnieją standardy kodu czasowego SMPTE. Nie istnieje standard kodu czasowego SMPTE dla 90 kl./s. Z tego powodu, gdy klatkaż projektu URSA Cine Immersive jest ustawiony na 90 kl./s, kamera faktycznie obsługuje kod czasowy 30 kl./s. Wyjścia SDI obsługują również 30p. Wszelkie akcesoria do kodu czasowego lub zewnętrzne rejestratory dźwięku powinny być skonfigurowane na 30 kl./s.

Filmowanie

Skoro posiadasz już podstawową wiedzę na temat obsługi kamery Blackmagic URSA Cine Immersive i korzystania z formatu immersyjnego, ten rozdział pozwoli Ci zgłębić kolejne aspekty działania kamery.

Wymagania dotyczące ekspozycji

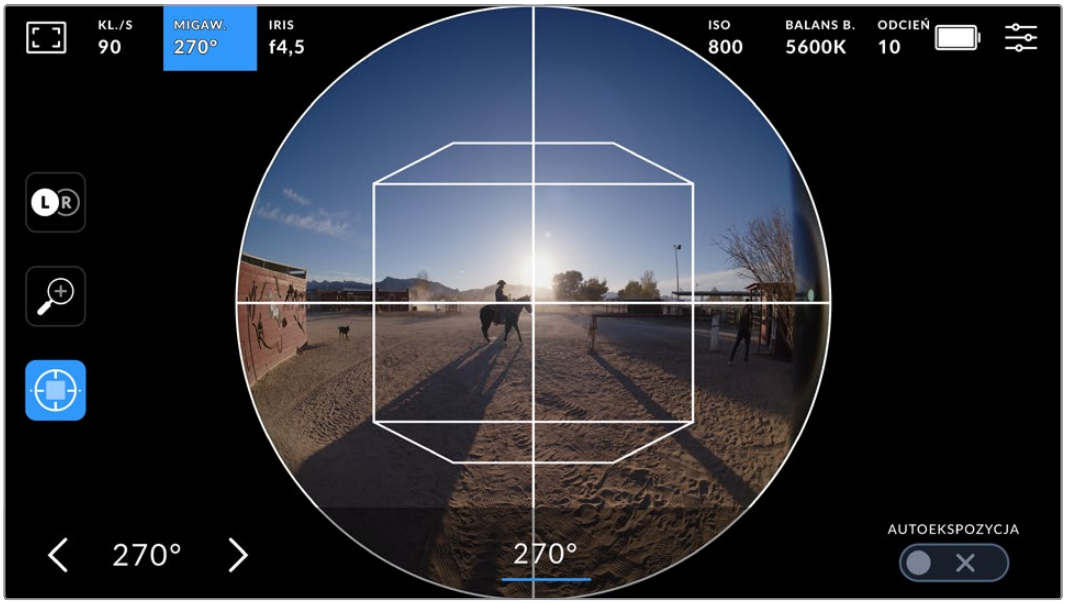
Apple Immersive Video ma większe wymagania dotyczące ekspozycji niż tradycyjne formaty przechwytywania 24 kl./s i może wymagać ponownego przemyślenia podejścia do filmowania i oświetlenia. Zachowując kąt migawki o wartości 180°, nagrywanie przy 90 kl./s wymaga niemal cztery razy więcej światła, aby uzyskać taką samą ekspozycję. Ponadto stała przysłona obiektywu (około f4.5) jest nawet o trzy stopnie ciemniejsza niż w przypadku typowych obiektywów Super Speed. To oznacza, że możesz potrzebować nawet siedmiu stopni światła więcej, aby osiągnąć odpowiednią ekspozycję. Jeśli jednak filmujesz przy 90 kl./s, użycie większego kąta migawki między 270° a 360° może zrekompensować utratę światła przy wysokim klatkażu, osiągając jednocześnie satysfakcjonującą ilość rozmycia ruchu.

Migawka

Wskaźnik **MIGAW.** wyświetla kąt lub czas otwarcia migawki. Za pomocą ustawienia pomiaru migawki w menu **USTAWIENIA** można wybrać, czy informacje o migawce mają być wyświetlane jako **Kąt migawki** czy **Migawka**.

Aby zmienić czas otwarcia migawki:

1. Stuknij ikonę **MIGAW.** na górnym pasku narzędzi.
2. Użyj strzałek w lewą i prawą stronę, aby przejrzeć powszechne presety czasu otwarcia migawki. Jeśli chcesz wybrać określoną wartość migawki, możesz to zrobić poprzez dwukrotne stuknięcie aktualnego wskaźnika migawki w lewym dolnym rogu ekranu. Otwiera to klawiaturę, która pozwala na wpisanie dowolnej wartości migawki z dokładnością do dwóch miejsc po przecinku, między 11,2 a 360°.



Niestandardowe wartości klatkażu można wprowadzić przy użyciu klawiatury ekranowej

Migawka ma mniejszy wpływ na wygląd treści immersyjnych o wysokim klatkażu. Choć wiele osób preferuje migawkę o kącie 180° do nagrywania 24 lub 30p, bardziej powszechną praktyką jest dostosowanie czasu otwarcia migawki w celu uzyskania odpowiedniej ekspozycji podczas nagrywania materiałów immersyjnych.

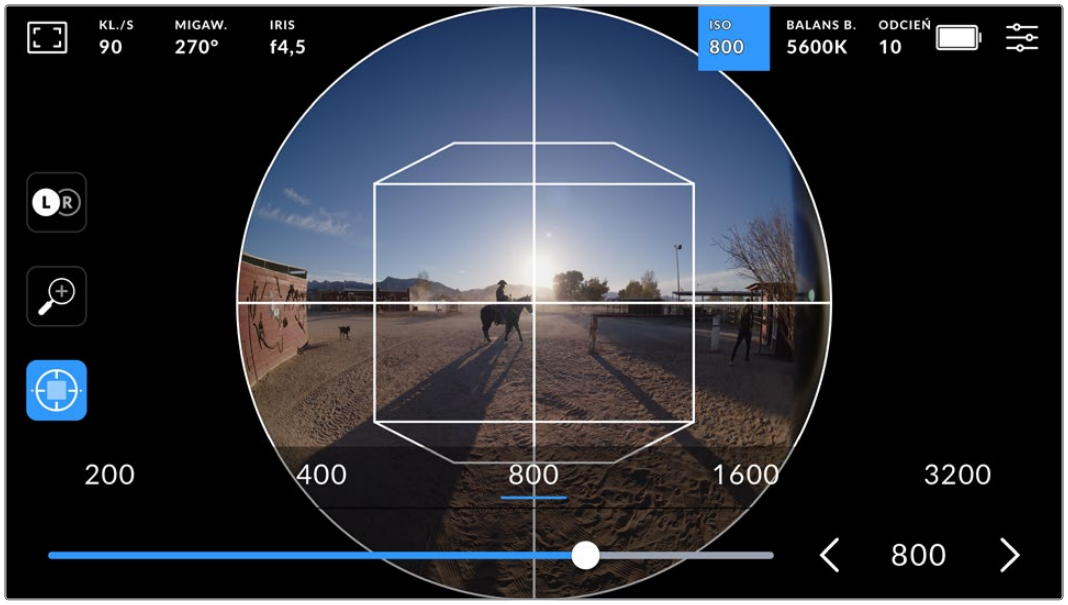
Filmując przy słabym świetle, migawkę można zwiększyć do 270° lub, w bardzo wyjątkowych okolicznościach, do 360°.

Kręcąc w jasnych warunkach, czas otwarcia migawki można obniżyć nawet do 11,2°. Jednak w jasnych warunkach bardziej powszechnym jest zachowanie preferowanych ustawień migawki i użycie filtrów ND do kontrolowania ekspozycji.

Warto zaznaczyć, że wysoki klatkaż Apple Immersive Video wynoszący 90 kl./s może być kłopotliwy podczas filmowania migoczących świateł. Zaleca się migawkę o kącie 270° w środowiskach 60 Hz oraz migawkę 324° podczas pracy przy oświetleniu 50 Hz.

ISO i zakres dynamiki

Wskaźnik **ISO** wyświetla aktualne ustawienie ISO lub inaczej światłoczułość. Stuknięcie tego wskaźnika umożliwia dostosowanie ISO do zmiennych warunków oświetleniowych. Zakres ISO w URSA Cine Immersive wynosi między 200 a 3200. Optymalna wartość ISO to 800.



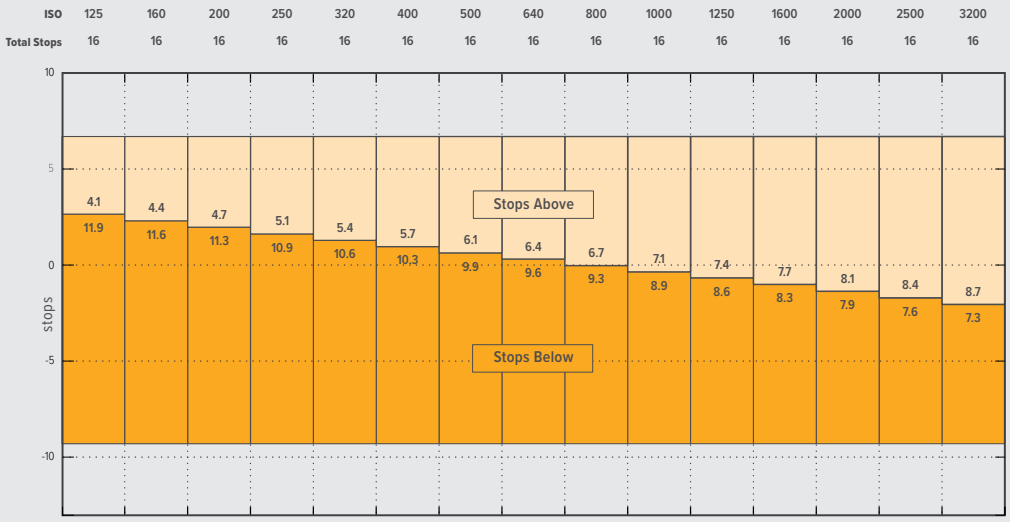
Wartość ISO można skonfigurować, stukając wskaźnik ISO

Presety numeryczne umożliwiają wybór ustawienia ISO z przyrostem co pełny stopień poprzez stuknięcie w liczbę. Suwak poniżej presetów umożliwia precyzyjną regulację ustawienia ISO pomiędzy tymi presetami z przyrostem co 1/3 stopnia.

W zależności od sytuacji można wybrać niższe lub wyższe ustawienie ISO. Na przykład w warunkach słabego oświetlenia czułość ISO 1600 może być odpowiednia, ale może powodować powstawanie widocznych szumów. W jasnych warunkach ISO 200 może zapewnić bogatszą kolorystykę.

Kamery Blackmagic są w stanie nagrywać z szerokim zakresem dynamiki, co zapewnia dużą wszechstronność podczas rejestrowania obrazów oraz w trakcie obróbki barwnej w DaVinci Resolve. Poniższa tabela zawiera dostępny zakres dynamiki URSA Cine Immersive.

Zakres dynamiki URSA Cine Immersive



ND

Filtry ND kamery to najbardziej powszechny sposób kontrolowania ekspozycji poprzez zmniejszenie ilości światła docierającego do sensora kamery o określoną liczbę stopni ekspozycji.



Filtry ND URSA Cine Immersive można dostosować za pomocą przycisków po lewej stronie nad rozkładanym ekranem dotykowym. Oprócz „czystej” opcji bez redukcji dostępne ustawienia filtru to 2, 4, 6 i 8 stopni.

Wskaźnik po lewej stronie u góry ekranu wskazuje bieżące ustawienie filtrów ND. Gdy filtry ND są włączone, wskaźnik jest stale wyświetlany. Gdy filtry ND są ustawione na „czystą” opcję, wskaźnik zniknie po 4 sekundach.



UWAGA Możesz zmienić terminologię filtrów ND, aby odpowiadały konwencjom, do których jesteś przyzwyczajony. Użyj menu **USTAWIENIA**, aby wybrać, czy filtr ND ma być wyświetlany jako numer, stopień lub ułamek.

Zapis off speed

Klatkaż projektu w menu **USTAWIENIA** to klatkaż formatu nagrywania URSA Cine Immersive i określa prędkość, z jaką Twój materiał będzie odtwarzany. Zwykle jest ustawiany tak, aby pasował do prędkości odtwarzania i synchronizacji dźwięku używanych w procesie postprodukcji oraz wymagań dotyczących dystrybucji.



Stuknij wskaźnik klatkażu URSA Cine Immersive, aby uzyskać dostęp do ustawień klatkażu. Blackmagic URSA Cine Immersive ma 9 ustawień klatkażu projektu, w tym 23,98, 24, 25, 29,97, 30, 50, 59,94, 60 i 90 klatek na sekundę. Podczas nagrywania materiału immersyjnego klatkaż projektu powinien być ustawiony na 90 kl./s.

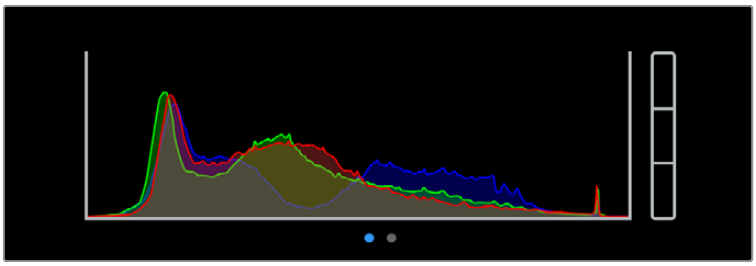
Klatkaż sensora pozwala określić prędkość, z jaką sensor przechwytuje obraz. Domyślnie klatkaże projektu i sensora URSA Cine Immersive są dopasowywane, aby uzyskać naturalne prędkości odtwarzania. Poprzez stuknięcie ikony przełącznika **KLATKAŻ OFF SPEED** w prawym dolnym rogu menu **KL./S** kamery można niezależnie ustawić klatkaż sensora.

Gdy przełącznik klatkażu off speed jest włączony, możesz ręcznie ustawić klatkaż sensora. Warto podkreślić, że 90 kl./s to najwyższy klatkaż kamery i będzie odpowiadał klatkażowi projektu (90 kl./s) podczas filmowania dla Apple Vision Pro. Jednak wybór niższego klatkażu projektu, jak 30 czy 60 klatek na sekundę, a następnie nagrywanie off speed przy 90 kl./s pozwoli uzyskać piękny materiał w zwolnionym tempie.

Gdy przełącznik **KLATKAŻ OFF SPEED** jest włączony, wskaźnik klatkażu w lewym górnym rogu ekranu będzie wyświetlał w pierwszej kolejności klatkaż sensora, a następnie klatkaż projektu.

Histogram

Na dole po lewej stronie ekranu dotykowego URSA Cine Immersive znajduje się histogram. Histogram to wykres, na którym widoczny jest rozkład jasności w obrazie. Lewa strona pokazuje cienie, środkowa półcienie, a prawa – podświetlenia. Wysokość wykresu wskazuje jaki obszar obrazu osiąga dany poziom jasności.



Histogram oferuje szybki podgląd rozkładu tonów w klipie. Obraz, którego wykres znajduje się po lewej stronie jest najpewniej niedoświetlony. Podobnie obraz, którego wykres znajduje się po prawej stronie, jest najprawdopodobniej prześwietlony. Histogram należy zawsze interpretować w kontekście z obrazem, ale ogólnie wykres obrazu z prawidłową ekspozycją będzie skupiał się w środkowej części. Zwiększenie eskpozycji przesunie histogram w prawo, a obniżenie ekspozycji – w lewo.

Po prawej stronie histogramu znajduje się wskaźnik, który informuje o przycinaniu kanałów poszczególnych kolorów. Zazwyczaj należy unikać przycinania kanałów. Jednak istnieją sytuacje, w których przycinanie obrazu jest dopuszczalne – jeśli pomaga to zachować półcienie i cienie obrazu.

„Expose to the Right" (ETTR) to technika, która celowo prześwietla obraz, przesuwając histogram jak najdalej w prawo bez przycinania kanałów koloru. To maksymalizuje sygnał i minimalizuje szum. Pod warunkiem, że ekspozycja zostanie obniżona w postprodukcji, jest to świetny sposób, aby uzyskać czyste obrazy!

Opcje podglądu LCD

Stuknij ikonę monitora w lewym górnym rogu ekranu dotykowego LCD URSA Cine Immersive, aby uzyskać dostęp do ustawień podglądu LCD. To menu zawiera przydatne opcje, jak jasność monitora, a także zaawansowane narzędzia ekspozycji.



Zebra

To ustawienie przełącza wygląd zebry na ekranie dotykowym LCD, jak również ustawienie poziomu zebry. Zebra wyświetla ukośne linie na obszarach obrazu, które przekraczają ustawiony poziom ekspozycji. Ustawienie procentowe jest używane, aby wskazać obszary obrazu, które są prześwietlone. Najczęściej ustawia się je na wartość 95%, zamiast 100%. Pozwala to zidentyfikować obszary obrazu, które mogą być prześwietlone. Drugie ustawienie zebry włącza lub wyłącza obszary środkowej szarości zebry lub środkowej szarości plus 1 stopień.

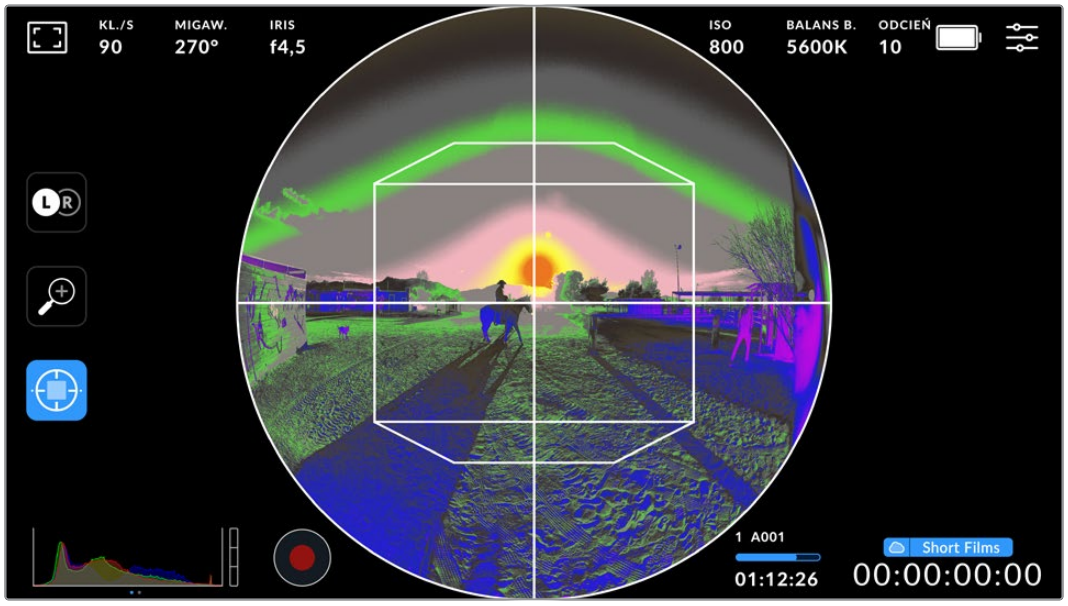
Siatki

To ustawienie przełącza wygląd siatki trójpodziału, miernika horyzontu, celownika lub punktu na ekranie dotykowym LCD, jak również ustawienie nakładki, która będzie widoczna na wyjściach URSA Cine Immersive. Podczas nagrywania treści immersyjnych miernik horyzontu jest bardzo przydatny. Wyświetla podwójny celownik pośrodku obrazu, który jest wyrównany, pod warunkiem, że kamera jest wypoziomowana – w przeciwnym wypadku będzie rozbieżny.

Fałszywy kolor

Fałszywy kolor nakłada na obraz różne kolory, które reprezentują wartości ekspozycji dla różnych elementów obrazu. Na przykład kolor różowy reprezentuje optymalną ekspozycję dla jaśniejszych odcieni skóry, podczas gdy kolor zielony dobrze pasuje do ciemniejszych odcieni skóry. Monitorując różowy lub zielony fałszywy kolor podczas nagrywania ludzi można zachować spójną ekspozycję dla ich odcieni skóry.

Podobnie zmiana koloru elementów na ujęciu z żółtego na czerwony oznacza ich prześwietlenie. Tak jak histogram, to narzędzie daje szybki i intuicyjny wgląd w rozkład tonów w obrazie.



Nagrywanie

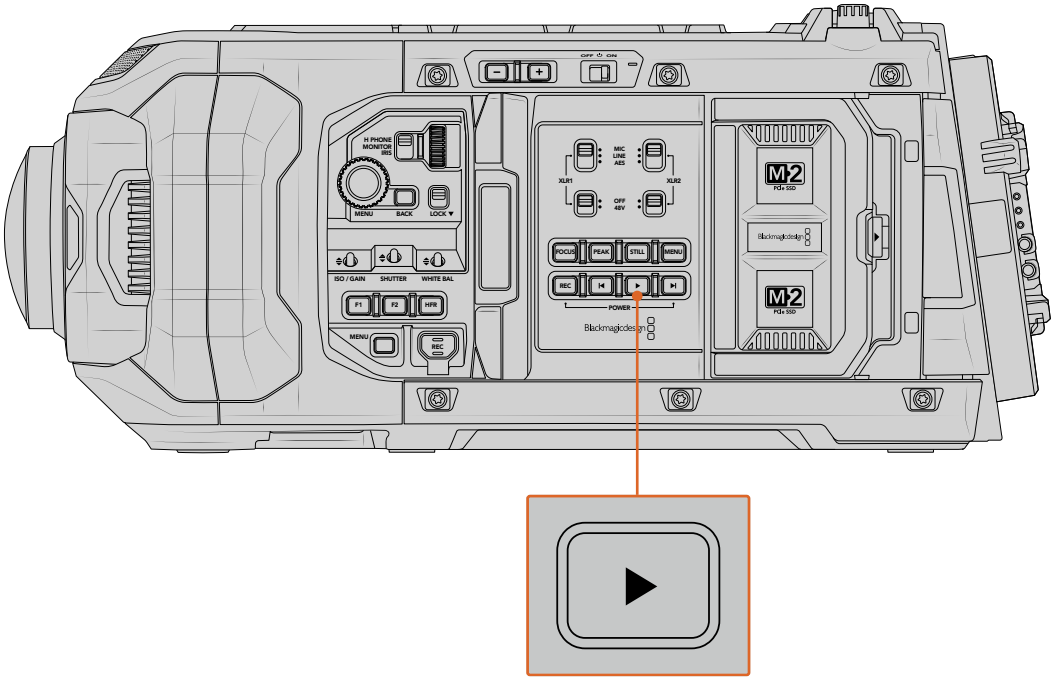
Rozpocznij nagrywanie, naciskając dowolny z czerwonych przycisków nagrywania na kamerze. Przyciski nagrywania znajdują się na przednim panelu sterowania, wewnętrznym panelu sterowania za rozkładanym ekranem dotykowym, panelu pomocniczym i obu ekranach dotykowych LCD. Naciśnij przycisk nagrywania ponownie, aby zatrzymać nagrywanie.

Wskaźniki zapisu informują, ile czasu pozostało do końca nagrywania. Czas jest wyświetlany w formacie godziny:minuty:sekundy zależnie od wybranego klatkażu. Pasek postępu nad czasem nagrywania będzie niebieski, biały lub czerwony w zależności od aktualnego stanu i wyświetli dostępną ilość miejsca na nośniku.



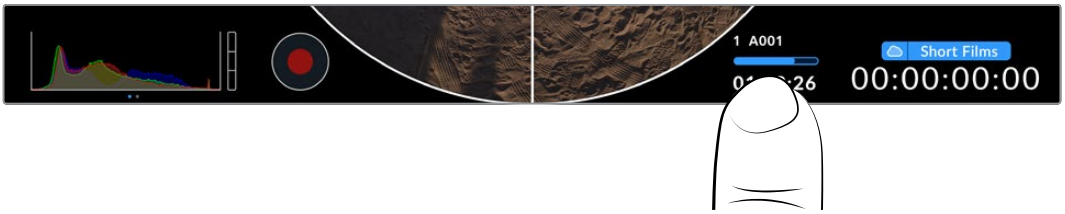
Odtwarzanie

Po nagraniu klipów można użyć przycisków sterowania transportem, aby je odtworzyć. Aby odtworzyć ostatni nagrany plik, naciśnij przycisk odtwarzania po lewej stronie kamery zlokalizowany za rozkładanym ekranem dotykowym. Za pomocą przycisków przewijania można przejść do innych klipów. Aby opuścić tryb odtwarzania i przygotować kamerę do dalszego nagrywania, naciśnij jednokrotnie przycisk nagrywania.

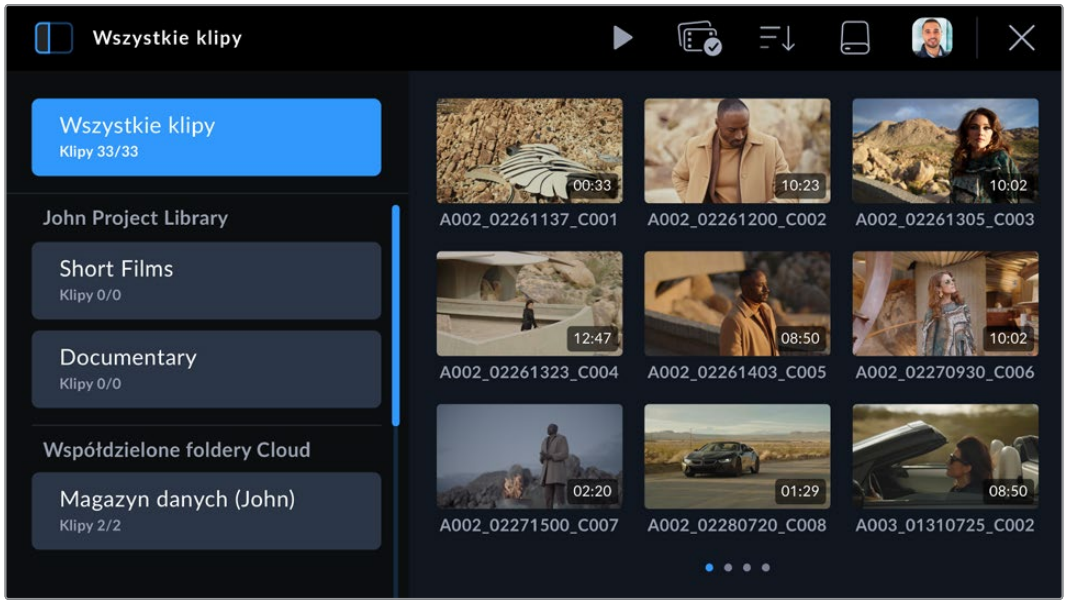


Naciskaj przyciski przewijania w przód i wstecz, aby przewijać do początku lub do końca klipów. Naciśnij przycisk przewijania jednokrotnie, aby przejść do początku bieżącego klipu lub naciśnij dwukrotnie, aby przejść do początku poprzedniego klipu. Przytrzymaj przycisk przewijania w przód lub przewijania wstecz, aby odtwarzać lub cofać klip z dwukrotnym przyspieszeniem. Po przewinięciu w przód lub wstecz naciśnij przyciski szybkiego przewijania w przód lub przewijania wstecz dwa razy dla x4, trzy razy dla x8 i cztery razy dla x16.

Jeśli wymagany jest bardziej zaawansowany podgląd nakręconego materiału, najprawdopodobniej będziesz chciał skorzystać z puli multimediów. Aby otworzyć pulę multimediów, stuknij jeden ze wskaźników zapisu na dole ekranu dotykowego kamery.



W przeglądarce strony głównej widoczne są miniatury wszystkich nagranych klipów na wszystkich nośnikach podłączonych do kamery. Miniatury są wyświetlane w trzech rzędach. Wraz z dodawaniem kolejnych klipów można przejść do następnej strony za pomocą ekranu dotykowego. Wskaźnik stron na dole wyświetlacza informuje, ile stron klipów jest dostępnych.



Podłącz URSA Cine Immersive do konta Blackmagic Cloud i prześlij materiał bezpośrednio do projektu DaVinci Resolve.

Jeśli chcesz opuścić pulę multimediów w dowolnym momencie, stuknij **X** w prawym górnym rogu lub naciśnij jednokrotnie dowolny z przycisków nagrywania kamery.

Transfer klipów

Istnieje wiele sposobów przesyłania klipów z pamięci kamery. Na przykład możesz podłączyć URSA Cine Immersive do komputera lub sieci przez Ethernet 10G i kopiować klipy z pamięci kamery przez Ethernet. Możesz nawet włożyć Media Module do Blackmagic Media Dock podłączonego do sieci lub komputera, co pozwala na pracę z maksymalnie trzema modułami jednocześnie.

Po podłączeniu do komputera przez sieć Ethernet komputer rozpozna Blackmagic Media Module tak samo, jak po podłączeniu Blackmagic Cloud Store.

Kamera jest domyślnie ustawiona na korzystanie z protokołu DHCP, więc komputer lub sieć wykryje kamerę i automatycznie przypisze jej adres IP.

Włączanie funkcji udostępniania plików

Podczas konfigurowania kamery po raz pierwszy i po podłączeniu do komputera lub sieci przez Ethernet, udostępnianie plików SMB będzie domyślnie wyłączone. Wynika to z faktu, że w przeciwieństwie do Blackmagic Cloud Store, gdzie prawdopodobnie będziesz chciał od razu udostępniać pliki innym użytkownikom w sieci, możesz nie chcieć od razu zezwalać na ten sam poziom dostępu do głównych plików kamery po ważnym dniu zdjęciowym. Z tego powodu zdecydowaliśmy się na samodzielne włączanie udostępniania plików SMB.

1. Aby włączyć udostępnianie plików, należy pobrać i zainstalować Blackmagic Camera Setup na komputerze. To narzędzie konfiguracyjne jest aplikacją administracyjną, która umożliwia aktualizację kamery i zmianę ustawień administracyjnych. Blackmagic Camera Setup można pobrać z naszej strony internetowej pod adresem: https://www.blackmagicdesign.com/pl/support/family/professional-cameras
2. Po pobraniu i zainstalowaniu narzędzia konfiguracyjnego, podłącz kamerę do komputera przez USB i uruchom Blackmagic Camera Setup. Podłącz za pomocą portu USB na górnym panelu w kierunku tylnej części kamery.

3 Aby otworzyć ustawienia, kliknij zdjęcie URSA Cine Immersive lub ikonę ustawień.

4 Przewiń w dół do ustawień **Network access** i włącz **File sharing (SMB)**. Następnie naciśnij **Save** w prawym dolnym rogu, aby aktywować ustawienia.

5 Teraz przejdź do ustawienia adresu URL pod spodem i kliknij ikonę adresu URL, aby automatycznie wypełnić adres. Adres URL wyświetli nazwę kamery jako lokalizację w sieci lokalnej.

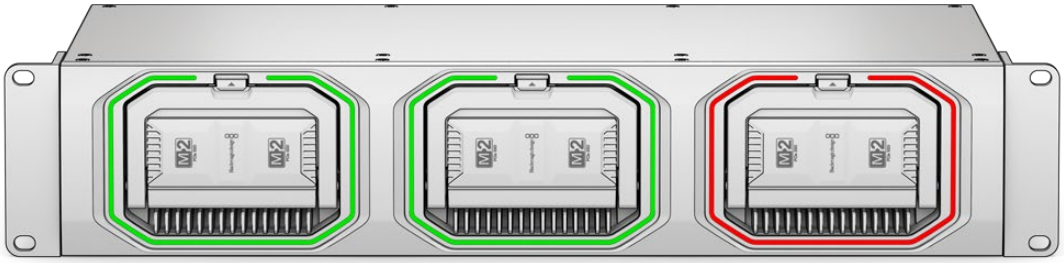
6 Kliknij **Save**.

Aby przejść bezpośrednio do klipów, kliknij nazwę adresu URL kamery. Otworzy się okno z prośbą o podanie hasła lub połączenie się jako gość. Kliknij przycisk opcji **Guest** i kliknij **Connect**.

7 Komputer będzie teraz automatycznie przełączał się do Media Module w oknie Finder na komputerach Mac i Windows Explorer w systemie Windows. Narzędzie zapamięta konfigurację sieci Ethernet, więc wystarczy to zrobić tylko raz podczas pierwszej konfiguracji, a gdy następnym razem podłączysz kamerę przez Ethernet, będziesz mógł od razu uzyskać dostęp do Media Module bez konieczności wprowadzania jakichkolwiek ustawień.

Korzystanie z Blackmagic Media Dock

Blackmagic Media Module został zaprojektowany do szybkiego wkładania i wyjmowania, gdy trzeba przenieść klipy do pamięci zewnętrznej lub zamienić na pusty moduł, aby można było kontynuować nagrywanie. Blackmagic Media Dock można wykorzystać w celu dostępu do materiału na Media Module bez kamery.



Podsumowanie

Apple Immersive Video oferuje nowe podejście do wizualnej narracji i stanowi alternatywę dla tradycyjnego kina 2D. Nie powstał jako zamiennik, lecz jako dodatkowy format, na który można się zdecydować, uwzględniając wymagania danego projektu. Jak w przypadku każdego formatu, kreatywne wybory będą się różnić w zależności od opowiadanej historii.

Nowe formaty często narzucają skomplikowane przepływy pracy, ale URSA Cine Immersive zaprojektowano tak, aby upraszczała proces przechwytywania formatu Apple Immersive Video. Zapewnia on przejrzystą ścieżkę produkcyjną dla treści immersyjnych i może być wykorzystany do efektów wizualnych oraz innych zastosowań.

Niniejsza instrukcja omawia podstawowe funkcje i przepływy pracy URSA Cine Immersive. Stanowi punkt wyjścia. Jak w przypadku wszystkich systemów kamer, warto dalej zgłębiać oraz testować kamerę, aby osiągać jak najlepsze rezultaty na potrzeby konkretnych produkcji.

DaVinci Resolve Studio



DaVinci Resolve Studio to kompleksowe oprogramowanie do postprodukcji, które płynnie integruje montaż, korekcję kolorów, efekty wizualne, animacje i postprodukcję dźwięku. Obsługuje przepływy pracy dostosowane do unikalnych cech formatu Blackmagic RAW Immersive – w tym stereoskopowe wideo w ultrawysokiej rozdzielczości, klatkaż 90 kl./s i miksowanie dźwięku przestrzennego. DaVinci Resolve Studio jest w pełni wyposażone na potrzeby produkcji, monitorowania i dystrybucji Apple Immersive Video, co czyni je zaawansowanym narzędziem dla twórców immersyjnych treści.

Zalecenia dotyczące stacji roboczej

Zalecenia sprzętowe

- Mac Studio M3 Ultra Desktop
- Monitor o wysokiej częstotliwości odświeżania (minimum 60 Hz)

Minimalne wymagania systemowe

- macOS 15 lub nowszy
- 64 GB pamięci systemowej
- Do monitorowania: Blackmagic Design Desktop Video 12.9 lub nowszy
- Komputer lub procesor graficzny oparty na architekturze Apple Silicon obsługujący technologię Metal

Wymagania dotyczące monitorowania immersyjnego

- Apple Vision Pro
- Szybka łączność bezprzewodowa (minimum Wi-Fi 5; preferowane Wi-Fi 6)

Ustawienia projektu

Przed zaimportowaniem surowych plików multimedialnych Blackmagic URSA Cine Immersive zaleca się sprawdzenie ustawień projektu w programie DaVinci Resolve, aby upewnić się, że pliki zostały poprawnie przetworzone i pozbawione efektu debayeringu, co zapewni optymalną jakość obrazu oraz ułatwi montaż.

Ustawienia główne

Podczas konfigurowania projektu należy koniecznie zaznaczyć, że tworzone treści będą przeznaczone dla Apple Vision Pro. Dzięki temu wszystkie prawidłowe ustawienia i opcje przepływu pracy będą widoczne podczas przygotowywania, montażu i dystrybucji projektu.

1. Uruchom DaVinci Resolve Studio.
2. W prawym dolnym rogu okna **Project Manager** kliknij **New Project**.
3. Wprowadź nazwę projektu, ustaw lokalizację multimediów (w razie potrzeby) i kliknij **Create**.
4. Otwórz **Project Settings**, klikając ikonę koła zębatego w prawym dolnym rogu DaVinci Resolve Studio.

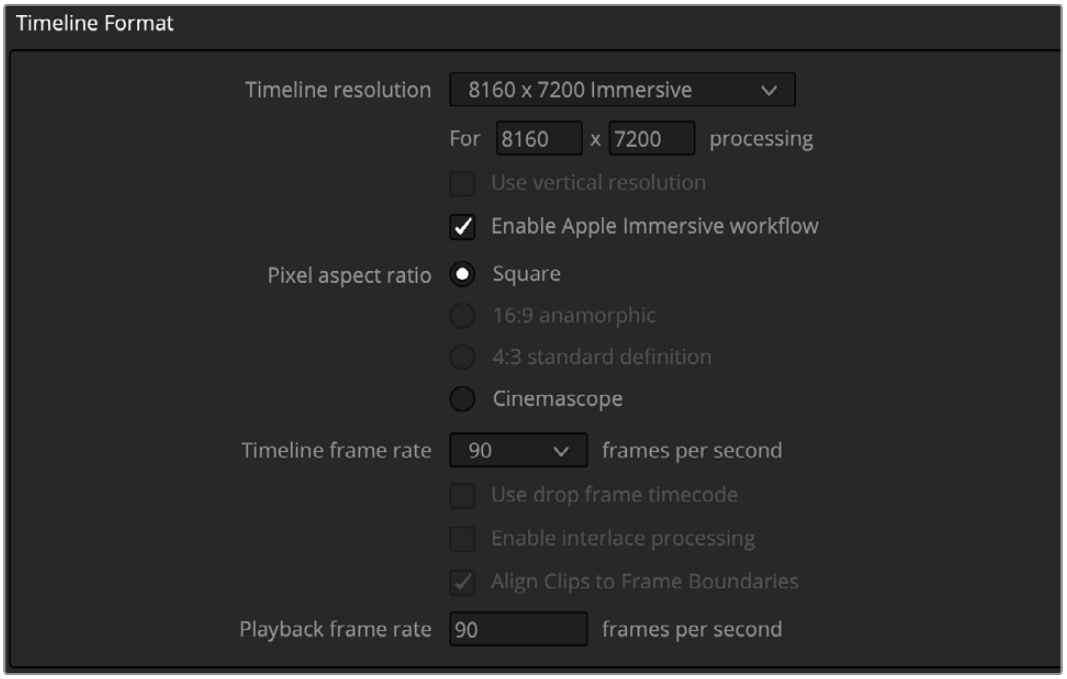
   Najpierw należy ustawić rozdzielczość i klatkaż projektu.
5. Wybierz opcję **Master Settings** na pasku bocznym.
6. W sekcji **Timeline Format** wybierz **Enable Apple Immersive workflow**.



   Spowoduje to skonfigurowanie wielu ustawień DaVinci Resolve w celu optymalnego dostosowania go do immersyjnych przepływów pracy. Podczas tworzenia projektu immersyjnego należy zawsze najpierw wybrać to ustawienie.
7. Zmień **Timeline resolution** na **8160 x 7200 Immersive**, aby przetworzyć klipy w ich natywnej rozdzielczości sensora.

   Jeśli pracujesz z ograniczoną mocą obliczeniową, wybierz opcję **4080 x 3600 Immersive**, aby zmniejszyć rozdzielczość o połowę i przyspieszyć odtwarzanie.
8. **Timeline frame rate** powinna być ustawiona na 90 klatek na sekundę. Ten parametr jest konfigurowany automatycznie po włączeniu przepływów pracy Apple Immersive.
9. Ustaw **Playback frame rate** na 90 klatek na sekundę, aby uzyskać najdokładniejsze odtwarzanie treści Immersive Video.

   Jeśli pracujesz z ograniczoną mocą obliczeniową (lub monitorem o niższej częstotliwości odświeżania), zmniejsz klatkaż, aby uzyskać bardziej stabilne odtwarzanie. Zaleca się wybranie klatkażu, który jest ułamkiem natywnego klatkażu, np. 30 klatek na sekundę lub 60 klatek na sekundę.



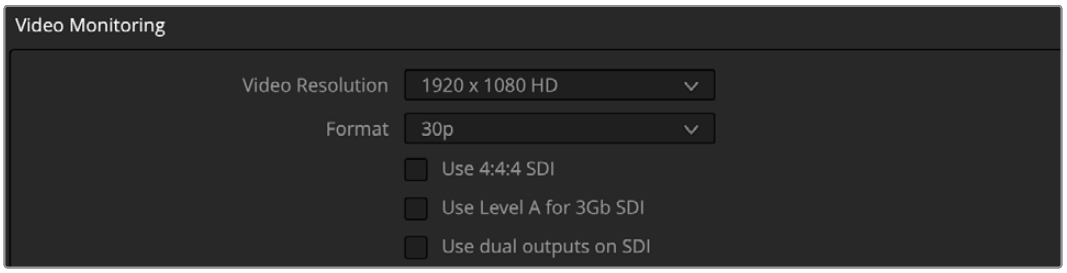
Monitorowanie wideo

Jeśli korzystasz z zewnętrznego monitora (za pośrednictwem urządzenia do przechwytywania/odtwarzania, takiego jak DeckLink lub UltraStudio), skonfiguruj ustawienia monitorowania wideo zgodnie ze specyfikacją monitora.

1. W sekcji **Video Monitoring** ustaw **Video Resolution** zgodnie z rozdzielczością monitora. Na przykład: **1920 x 1080 HD** lub **3840 x 2160 Ultra HD**.

   Chociaż natywny klatkaż immersyjnego wideo wynosi 90 kl./s, nie jest to częstotliwość obsługiwana przez większość monitorów telewizyjnych.

2. Ustaw format tak, aby odpowiadał częstotliwości odświeżania monitora. Na przykład: 30p.



Skalowanie obrazu

Ustawienia skalowania obrazu zapewniają prawidłowe wyświetlanie rozdzielczości i proporcji multimediów immersyjnych w oknie montażowym. Parametry te zostałyby automatycznie skonfigurowane po włączeniu opcji **Apple Immersive workflow** w **Master Settings**, ale należy sprawdzić, czy są one prawidłowe.

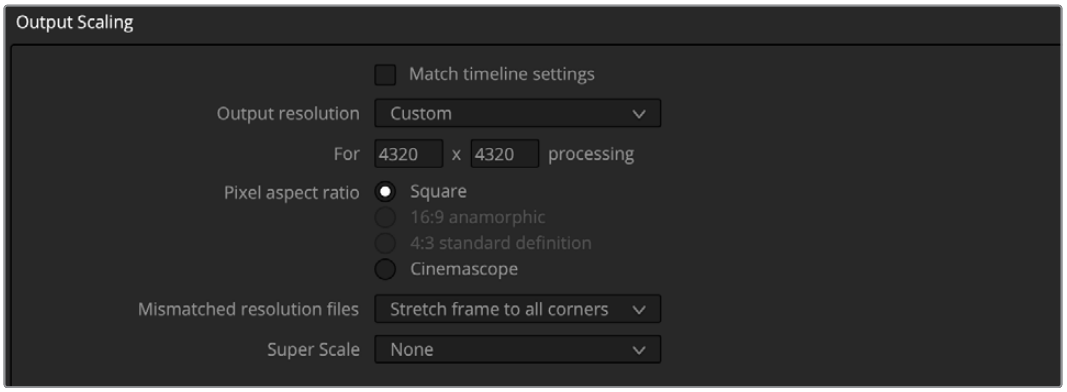
1. Wybierz **Image Scalling** na pasku bocznym **Project Settings**.
2. W sekcji **Output Scaling** upewnij się, że opcja **Match timeline settings** jest odznaczona.

3 Upewnij się, że **Output resolution** jest ustawiona na **Custom** i ustaw na **4320 x 4320**.

Połowa rozdzielczości (**2160 x 2160**) dla mniejszych wyświetlaczy i/lub szybszego odtwarzania.

Te rozdzielczości o proporcjach kwadratowych są wymagane do odtwarzania treści wideo Apple Immersive w programie DaVinci Resolve Studio oraz do przesyłania strumieniowego do Apple Vision Pro (przez Wi-Fi lub Developer Strap).

4 Upewnij się, że opcja **Mismatched resolution files** jest ustawiona na **Stretch frame to all corners**.



Zarządzanie kolorem

Ze względu na konieczność dostosowania maksymalnej luminancji nitów do indywidualnych potrzeb, zaleca się stosowanie zarządzania kolorami za pomocą Color Space Transform (CST) na stronie Koloru. Celem ustawień zarządzania kolorami w ustawieniach projektu jest określenie przestrzeni kolorów osi czasu (lub „roboczej”) strony Koloru oraz ustawienie metadanych przestrzeni kolorów, które będą powiązane z renderowanym plikiem.

1 Wybierz **Color Management** na pasku bocznym **Project Settings**.

2 W polu **Color science** pozostaw ustawienie **DaVinci YRGB**.

3 Ustaw **Timeline color space** jako **DaVinci WG/Intermediate**.

Jest to przyszłościowa przestrzeń kolorów o szerokim gamucie barw, która obsługuje przepływy pracy HDR, dzięki czemu idealnie nadaje się do multimediów Blackmagic URSA Cine Immersive i końcowych materiałów na Apple Vision Pro.

4 Ustaw **Output color space** jako **P3-D65 ST2084 1000 nits**.

Ta przestrzeń kolorów charakteryzuje się gamutem barw odpowiednim do monitorowania okien montażowych DaVinci Resolve na komputerach Mac Studio Desktop, które zazwyczaj odpowiadają standardowi Display P3. Jeśli używasz monitora referencyjnego lub innego profilu ICC Apple, należy zmienić wyjściową przestrzeń kolorów, aby dopasować ją do standardu kolorów wyświetlacza.

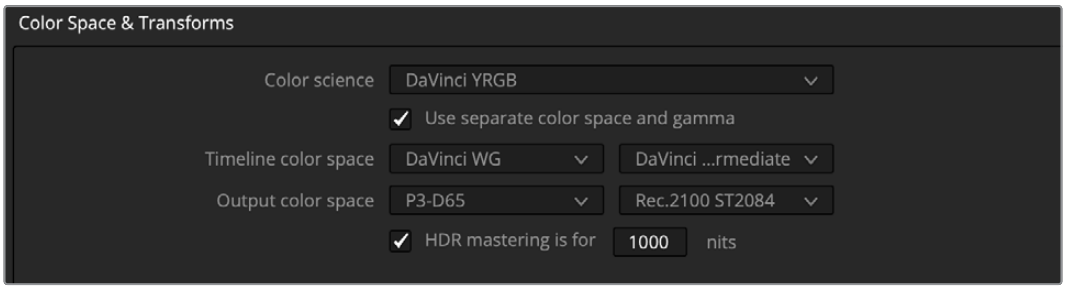
Aby jednak uzyskać prawidłowe odwzorowanie profilu ICC, konieczna będzie zmiana wartości gamma (funkcji transferu).

5 Kliknij **Use separate color space and gamma** nad ustawieniami przestrzeni kolorów.

Spowoduje to rozdzielenie pól rozwijanych przestrzeni kolorów i gamma.

6 Ustaw gamma wyjściowe (w prawej kolumnie) jako **Rec.2100 ST2084**.

7 Wybierz **HDR mastering is for 1000 nits**.



Aby skrócić czas konfiguracji kolejnego projektu immersyjnego Apple, możesz zapisać bieżące ustawienia jako preset.

8 Przejdź do menu opcji z trzema kropkami w prawym górnym rogu okna **Project Settings** i wybierz opcję **Save Current Settings as Preset**.

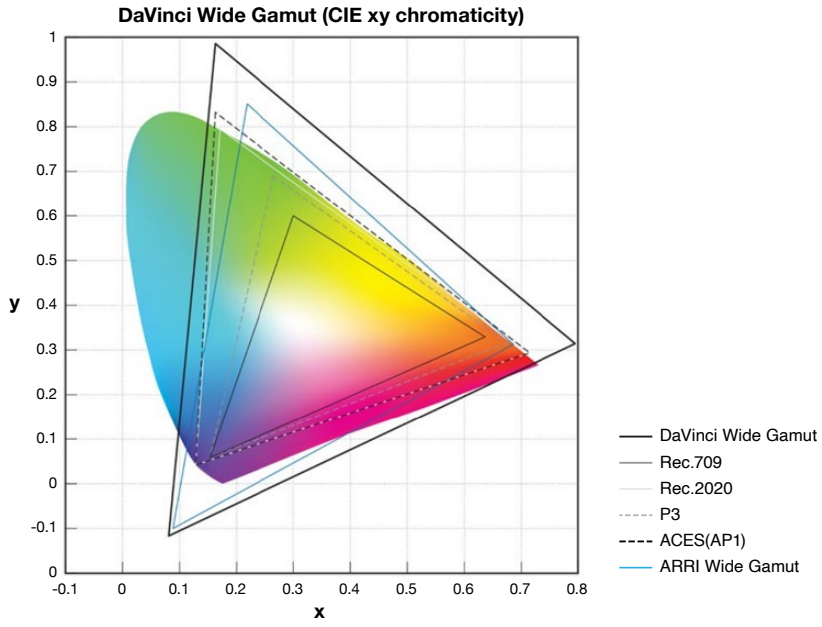
9 Nazwij preset i kliknij **OK**.

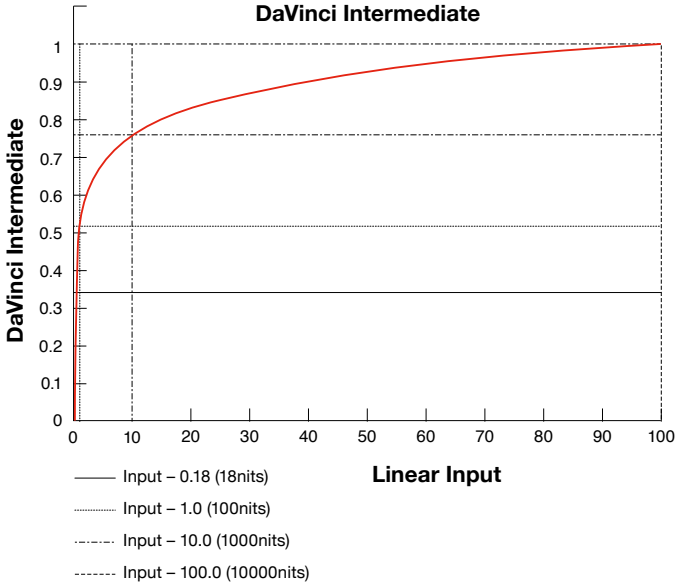
Aby szybko skonfigurować ustawienia projektu podczas następnej pracy z multimediami immersyjnymi, otwórz okno **Project Settings**, kliknij menu opcji i wybierz jeden z dostępnych presetów.

10 Kliknij **Save** i zamknij **Project Settings**.

Informacje dotyczące DaVinci Wide Gamut

DaVinci Wide Gamut to zaawansowana wewnętrzna przestrzeń kolorów opracowana przez Blackmagic Design dla programu DaVinci Resolve, oferująca wyjątkowo szerokie i precyzyjne środowisko do korekcji kolorów oraz przetwarzania obrazu. W połączeniu z profilem DaVinci Intermediate gamma tworzy jednolitą przestrzeń roboczą, która upraszcza zarządzanie kolorami w różnych formatach kamer, zapewniając spójne odwzorowanie tonów i nasycenia od wejścia do wyjścia. Idealny do pracy z HDR i multimediami immersyjnymi, szeroki gamut kolorów pozwala kolorystom na wprowadzanie subtelnych poprawek bez ryzyka powstawania pasm lub przycinania tonalnego podczas masteringu do formatów końcowych o wysokiej wierności odwzorowania.





Przetwarzanie Apple Immersive

W pasku bocznym **Project Settings** sekcja Apple Immersive umożliwia ustawienie niestandardowych danych ILPD (Immersive Lens Profile Data), aby zmodyfikować sposób kalibracji i wyświetlania treści immersyjnych.

1. Wybierz **Apple Immersive** w pasku bocznym **Project Settings**.
2. Kliknij **Open Calibration Folder**.
3. Dodaj lub zaktualizuj niestandardowy plik ILPD.
4. Kliknij opcję **Update Lists**, aby odświeżyć zawartość ILPD dostępną w puli multimediów.
5. Kliknij **Save** i zamknij **Project Settings**.
6. W puli multimediów kliknij prawym przyciskiem myszy immersyjny klip.
7. W menu kontekstowym przejdź do **Apple Immersive Calibration** i wybierz źródło ILPD (No Apple Immersive Calibration File Selected), niestandardowy ILPD lub gap.ILPD.

Domyślnie folder kalibracji ILPD znajduje się w folderze ProIM, obok domyślnego folderu LUT.

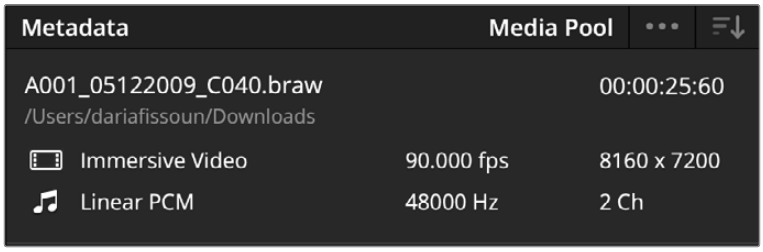
Pobieranie danych

Podczas importowania treści immersyjnych w formacie RAW, wraz z importowanym klipem .braw przesyłane są również wszystkie powiązane metadane immersyjne.

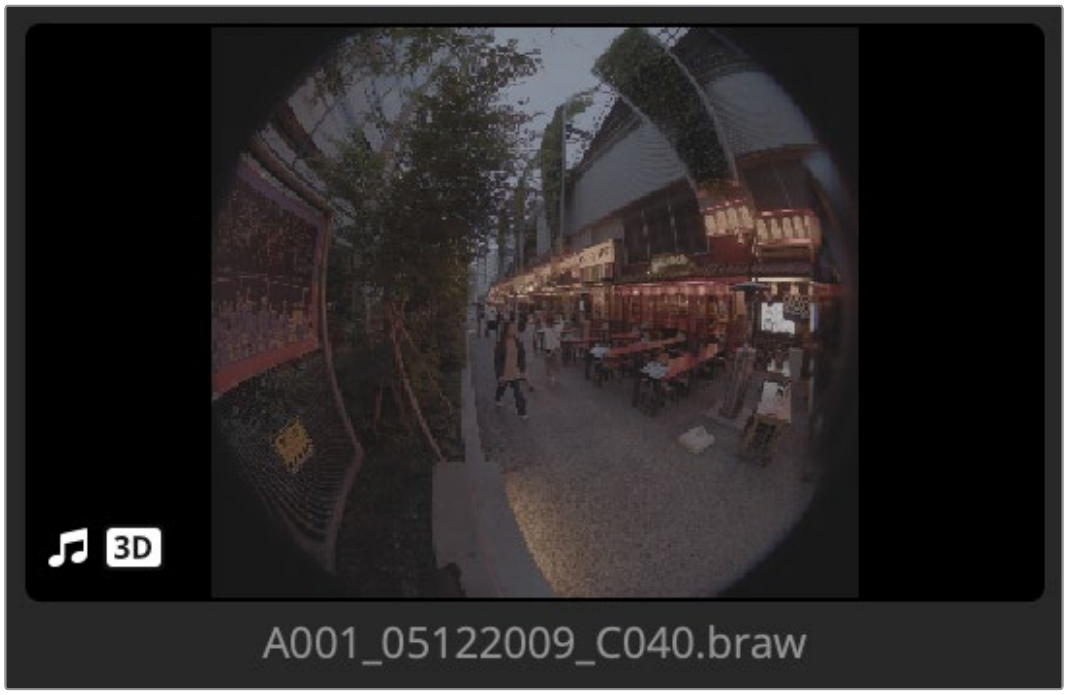
1. Przejdź do strony Multimediów.
2. W panelu **Media Storage** przejdź do lokalizacji immersyjnych treści wideo.
3. Przeciągnij i upuść multimedia do puli multimediów poniżej.

WSKAZÓWKA Aby zachować oryginalną strukturę folderów, przeciągnij foldery z panelu **Media Storage** do paska bocznego puli multimediów. Po przeciągnięciu do kosza głównego (lub dowolnego innego kosza utworzonego przez użytkownika) foldery i podfoldery zostają zachowane.

Po wybraniu klipu wideo Apple Immersive, parametr kodeka w nagłówku panelu **Metadata** wskaże, że jest to wideo immersyjne, wraz z jego natywnym klatkażem i rozdzielczością.



Dodatkowo ikona 3D w lewym dolnym rogu miniatury wideo potwierdza, że jest to klip 3D.



UWAGA Podczas przeciągania i upuszczania multimediów immersyjnych do puli multimediów po raz pierwszy może pojawić się okno dialogowe z pytaniem, czy chcesz zachować ustawienia, czy je zmienić. Wybranie opcji **Change Settings** spowoduje automatyczne dostosowanie ustawień projektu do klatkażu i rozdzielczości źródłowych multimediów.

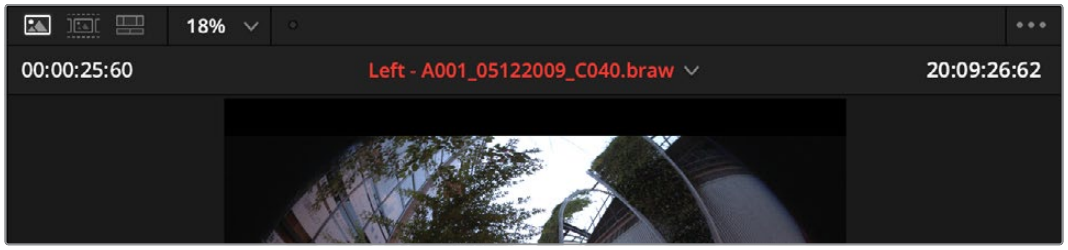
Montaż

Podczas pracy z treściami immersyjnymi narzędzia montażu i działania pozostają w większości takie same jak podczas pracy z konwencjonalnymi formatami wideo 2D. Poniższy zestaw instrukcji opisuje funkcje montażu lub działania, które są charakterystyczne dla immersyjnych procesów pracy z materiałami wideo.

Zestawienie

Podczas podglądu klipów w trakcie zestawiania osi czasu można zmienić oko (lewe lub prawe), którym ogląda się obraz w źródłowym oknie montażowym. W przeciwnym razie proces zestawiania ujęcia na osi czasu pozostaje niezmieniony w stosunku do standardowych procedur pracy w programach do montażu nieliniowego.

1. Przejdź do strony Edycji.
2. Kliknij dwukrotnie klip, aby załadować go do źródłowego okna montażowego po lewej stronie. Etykieta u góry okna montażowego wskazuje, którym okiem aktualnie oglądasz obraz.



3. Aby zmienić oko w oknie montażowym, kliknij prawym przyciskiem myszy klip źródłowy w puli multimediów i wybierz **Stereoscopic 3D** > **Stereoscopic 3D Mode** > **Left Eye/Right Eye/Both Eyes**.

> **WSKAZÓWKA** Możesz również najechać kursorem na miniatury w puli multimediów, aby przewijać klipy w źródłowym oknie montażowym, jeśli w opcjach tego okna włączono **Live Media Preview**.

4. Użyj skrótów klawiszowych I i O, aby zdefiniować zakres w źródłowym oknie montażowym.
5. Naciśnij przycisk **Insert clip** na pasku narzędzi (lub użyj skrótu klawiszowego F9), aby wstawić klipy na oś czasu.

> **WSKAZÓWKA** Możesz również przeciągać i upuszczać klipy z puli multimediów do panelu osi czasu.

6. Za pomocą myszki (w trybie **Selection** lub **Trim Edit**) możesz przycinać klipy na osi czasu.

Tła

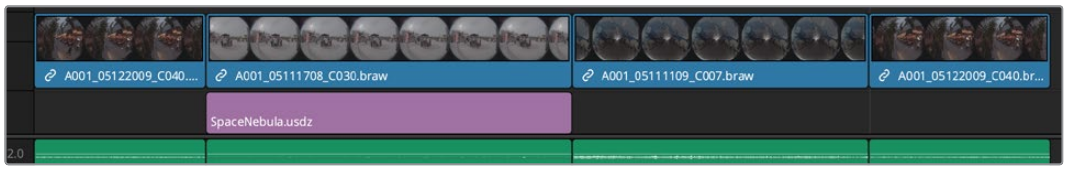
Po wybraniu opcji **Enable Apple Immersive workflow** w **Project Settings**, pod ścieżką **Video 1** automatycznie pojawi się ścieżka tła (**Backdrops**, BD).

Tło odnosi się do obszaru, który obserwator widzi, gdy patrzy poza obręb immersyjnej treści wideo, który zazwyczaj jest ograniczony do pola widzenia wynoszącego 180–210 stopni.

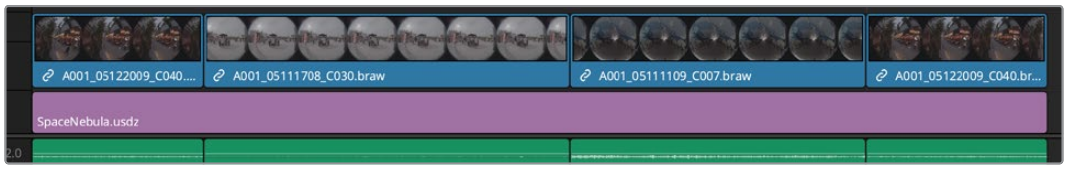
Domyślnie obszar ten jest czarny, ale można wstawić własne tło (w formacie .usdz) za pomocą ścieżki tła.

1 Przeciągnij plik tła .usdz na ścieżkę **Backdrops**, aby dodać go do osi czasu.

Czas trwania obrazów tła musi być dopasowany do krawędzi klipów na ścieżkach wideo. Pojedynczy plik tła może obejmować jeden klip, wiele klipów lub całą oś czasu, ale nie można wykonać cięcia w środku immersyjnego klipu .braw (tzw. „cięcie L”).



Pojedynczy klip – prawidłowo ✓

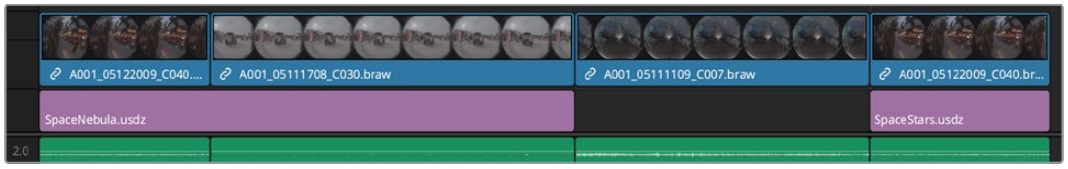


Wiele klipów – prawidłowo ✓



Cięcie L – nieprawidłowo ✗

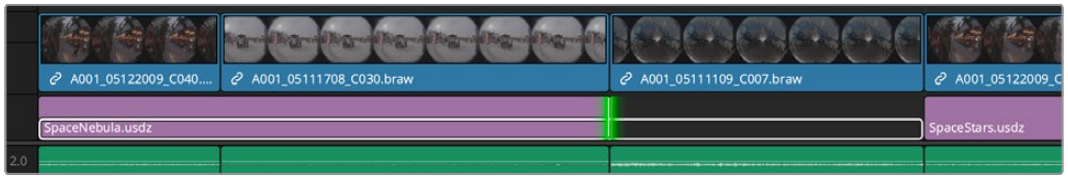
Możesz wybrać opcję umieszczenia wielu plików tła w jednej osi czasu lub nie dodawać żadnego obrazu, aby domyślnym tłem było czarne tło.



Różne obrazy i luki – prawidłowo ✓

2 Użyj standardowych elementów sterujących przycinaniem, aby dostosować długość pliku tła.

Przeciągnij krawędź pliku .usdz (w trybie **Selection** lub **Trim Edit**), aby dostosować długość.



Można również użyć poleceń **,** (przecinek) i **.** (kropka), aby przesuwać krawędź przycinania o jedną klatkę naraz.

WSKAZÓWKA Aby zapewnić stałe dopasowanie do montowanych fragmentów wideo immersyjnego, włącz tryb **Snap** na pasku narzędzi strony Edycji.

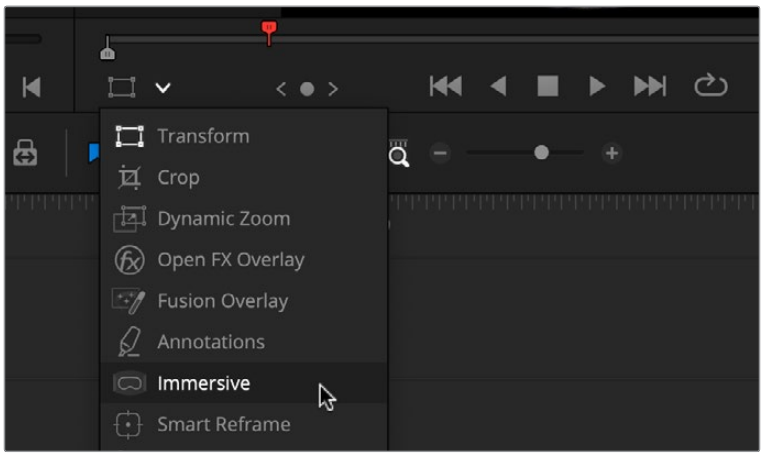
Korzystanie z okna montażowego

Domyślnie klipy immersyjne są wyświetlane w oknach montażowych stron DaVinci Resolve w okrągłej „przestrzeni obiektywu”. Kalibracja wizualna obiektywu użyta do wyświetlenia obrazu pochodzi z kamery i jest unikalna dla każdego klipu.



Możesz również zmienić tryb widoku na **LatLong** lub **Viewport**.

1. W lewym dolnym rogu okna montażowego osi czasu kliknij strzałkę menu rozwijanego obok przycisku sterowania ekranowego.
2. Wybierz **Immersive**.



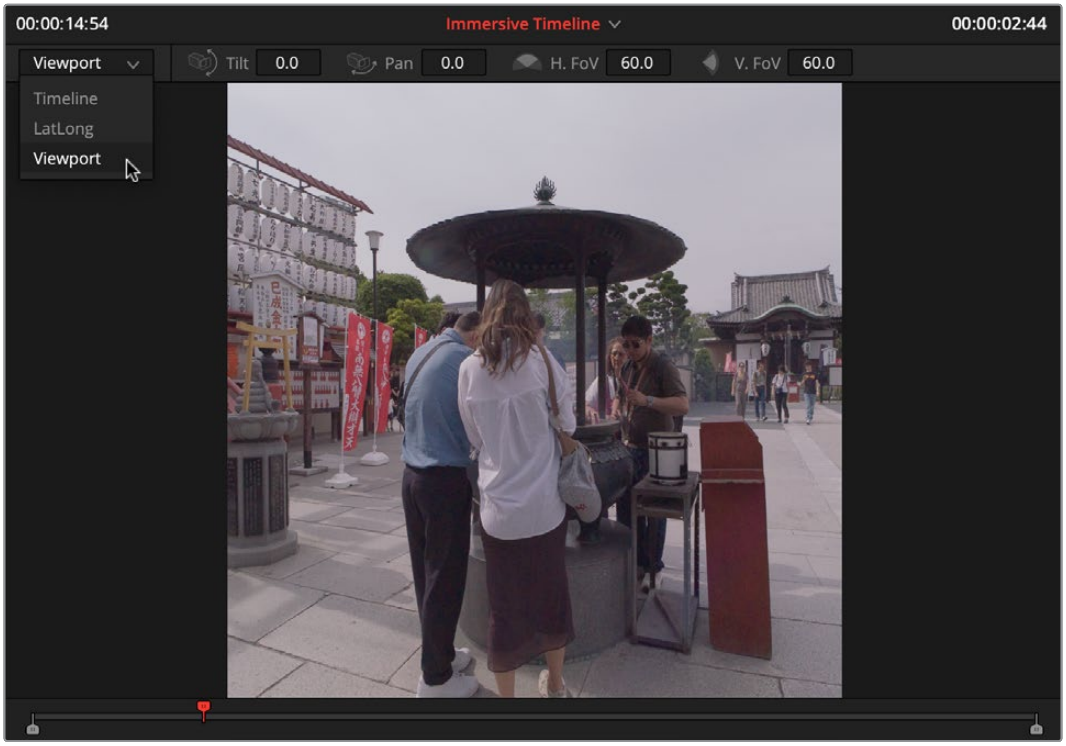
Nad oknem montażowym pojawi się nowy pasek narzędzi z dodatkowymi opcjami wyświetlania.

3. Kliknij **LatLong**, aby wyświetlić obraz w odwzorowaniu równoodległościowym. Ten tryb zniekształca obraz, aby bardziej przypominał tradycyjne proporcje multimediów 2D, co ułatwia niektóre procesy postprodukcji, takie jak śledzenie i kompozytowanie.



4 Kliknij opcję **Viewport**, aby wyświetlić wycięty środek immersyjnej sceny w kwadratowym kadrze. Ten tryb pomaga wyraźnie zobaczyć prawidłowo proporcjonalne szczegóły sceny na standardowym monitorze, ale trzeba uważać, żeby nie przeoczyć żadnego ruchu poza kadrem.

Za pomocą elementów sterujących transformacją na pasku narzędzi **Immersive** u góry można przechylać i przesuwać obraz stereoskopowy w kadrze, a także dostosowywać poziome i pionowe pola widzenia.



Przejścia

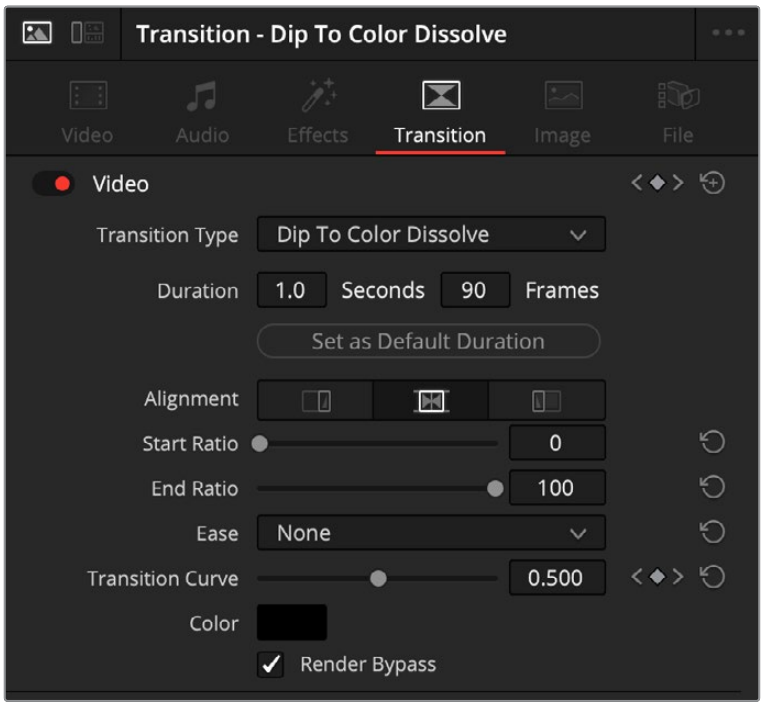
Aby wykonać przejście między dwoma immersyjnymi klipami wideo, możesz wykonać bezpośrednie cięcie, użyć uchwytów po obu stronach klipu, aby spowodować wygaszenie lub zanikanie obrazu lub użyć efektu **Dip To Color Dissolve**. Przejścia nie powinny być osadzane w materiałach filmowych, ale raczej renderowane przez sam Apple Vision Pro podczas odtwarzania.

1. Otwórz panel **Effects**.
2. Na pasku bocznym przejdź do **Toolbox** > **Video Transitions**.
3. Przeciągnij opcję **Dip To Color Dissolve** na punkt przecięcia dwóch klipów.



UWAGA Aby zastosować przejście, musisz mieć wystarczającą ilość danych poza punktami cięcia obu klipów. Jeśli dane przycinania (uchwyty) są niewystarczające, przejście nie zostanie zastosowane.

4. W panelu Inspektora przejdź do zakładki **Transition** i wybierz opcję **Render Bypass**. Dzięki temu efekt rozpuszczania zostanie poprawnie wyrenderowany przez Apple Vision Pro.
5. Następnie można użyć pozostałych parametrów **Dip To Color Dissolve** w oknie Inspektora, aby zmodyfikować czas trwania, wyrównanie, prędkość animacji i kolor przejścia.



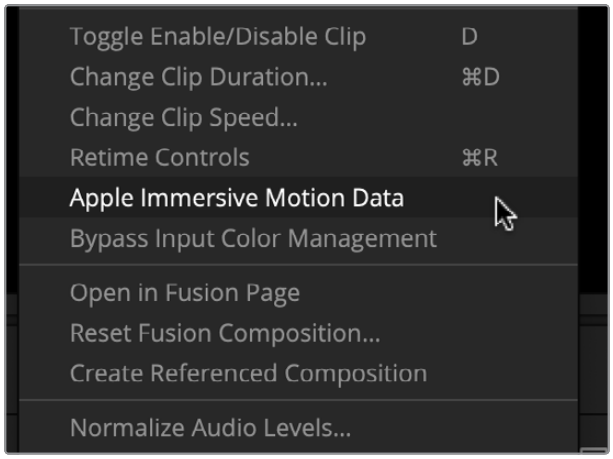
Apple Immersive Motion Data

Podczas pracy z immersyjnym materiałem szybki lub nagły ruch kamery może być dezorientujący dla widza. Aby zapewnić mu komfort, należy unikać ujęć z szybkim panoramowaniem/odchylaniem, ruchem obrotowym lub nierównym horyzontem.

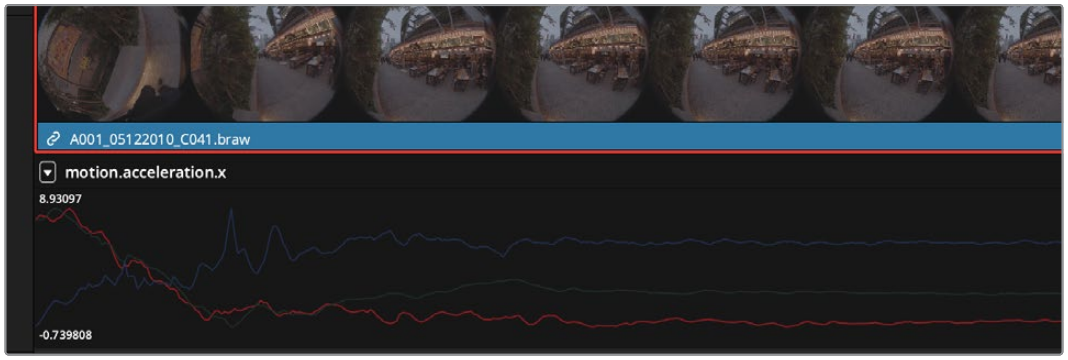
Funkcja Apple Immersive Motion Data wyświetla wykresy przyspieszenia x, y i z oraz wartości położenia żyroskopu, aby obiektywnie pokazać, gdzie ruch kamery może być zbyt intensywny. Montażyści mogą wykorzystać te informacje, aby uniknąć używania dezorientujących fragmentów materiału filmowego.

1 Kliknij prawym przyciskiem myszy immersyjny klip na osi czasu.

2 Wybierz opcję **Apple Immersive Motion Data**.

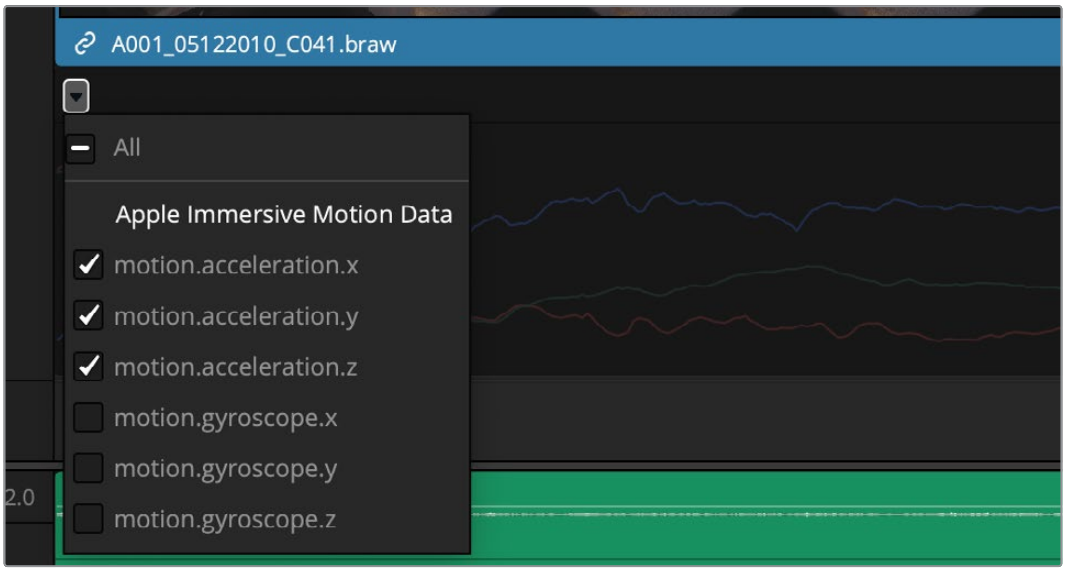


Na ścieżce wideo pojawi się wykres przedstawiający wartości przyspieszenia x, y, z.



W tym przykładzie kamera nagrywała podczas konfigurowania jej przez operatora. Skoki na wykresie wskazują na nagły ruch, gdy kamera została podniesiona, obrócona i umieszczona na jego ramieniu. Na potrzeby filmu najlepiej byłoby użyć jego ostatniej części z bardziej płaskimi odczytami będącymi rezultatem ustabilizowania kamery.

3 W lewym górnym rogu wykresu kliknij strzałkę menu rozwijanego.



Odczyty można włączać i wyłączać w zależności od zawartości ujęcia i charakteru ruchu kamery.

Nagłówki

Na stronie Edycji do immersyjnej sceny można dodać tekst za pomocą dowolnego z narzędzi do tworzenia nagłówków w panelu Effects: Text, Text+, MultiText, Scroll i dowolnej opcji belek na dole ekranu.

Po umieszczeniu nad klipem immersyjnym, generator nagłówków użyje danych ILPD (Immersive Lens Profile Data) klipu, aby zniekształcić tekst w taki sposób, aby idealnie pasował do wypaczenia obiektywu w scenie.

Jeśli dane ILPD nie są dostępne (na przykład podczas dodawania tekstu na czarnym tle lub na immersyjnej scenie CGI), zastosowane zostanie domyślne zniekształcenie obiektywu, aby tekst nadal wyglądał na odpowiednio wypaczony podczas oglądania w Apple Vision Pro.

1 W panelu Effects przejdź do opcji **Toolbox** > **Titles**.

2 Przeciągnij generator tekstu na oś czasu. Umieść go na ścieżce wideo nad klipem immersyjnym, aby wkomponować go w scenę.



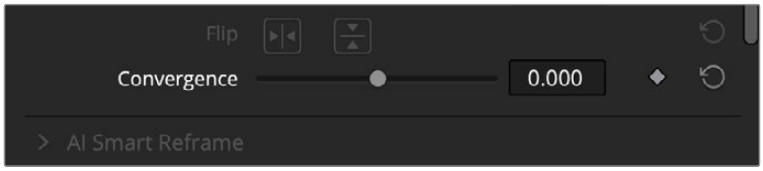
Podobnie jak w przypadku tła, generatory tekstu nie mogą wychodzić poza granice klipu. Ponieważ każdy klip jest renderowany z indywidualną przestrzenią obiektywu, wszelkie nakładające się nagłówki mogą zostać wypaczone między ujęciami.

3 Przytnij nagłówek do punktów edycji wideo.

4 Przy pomocy Inspektora napisz i sformatuj tekst nagłówka.

5 Przejdź do zakładki **Settings** i użyj suwaka **Convergence**, aby umiejscowić tekst w przestrzeni 3D.

Przesunięcie suwaka w lewo cofa nagłówek w przestrzeni, a przesunięcie go w prawo przesuwa go do przodu.



Działanie tej funkcji jest najlepiej widoczne podczas monitorowania na Apple Vision Pro. W przypadku korzystania z innych opcji monitorowania można zmienić sposób wyświetlania konwergencji tekstu w oknie montażowym, zmieniając tryb **Vision** bazowej warstwy wideo na stronie Koloru.

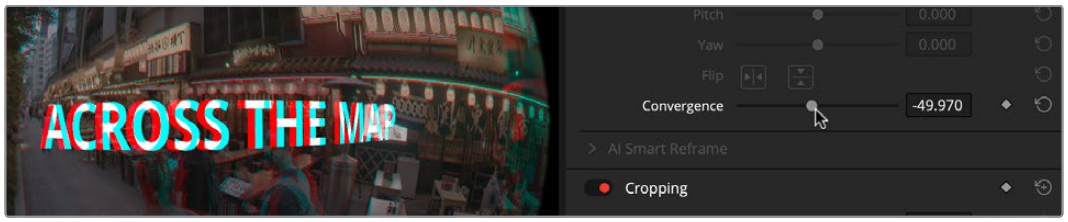
6 Przejdź do strony Koloru.

7 Otwórz paletę **3D**.

Kolumna **Vision** pośrodku palety umożliwia zmianę standardu stereoskopowego używanego do wyświetlania immersyjnych multimediów.

8 Wybierz zakładkę **Stereo**.

9 Zmień tryb **Out** na **Anaglyph (Color)**.



Zmiana parametru **Convergence** generatora tekstu w panelu Inspektora strony Edycji będzie bardziej intuicyjna, jeśli będzie się ją oglądać w okularach anaglifowych (lub anachromowych).

Wskazówki dotyczące kreatywności

W początkowej fazie produkcji z wykorzystaniem kamery Blackmagic URSA Cine Immersive i Apple Vision Pro zachęcamy filmowców do eksperymentowania z tym formatem w celu określenia najlepszych praktyk, ustalenia nowych zasad i odkrycia sposobów pracy, które nie były wcześniej możliwe w tradycyjnej kinematografii.

Mając to na uwadze, niektóre aspekty produkcji immersyjnej zostały oznaczone jako wymagające szczególnej uwagi, aby zapewnić maksymalne zaangażowanie widzów i komfort wizualny.

1 Montaż powinien rozpocząć się na etapie preprodukcji. Ze względu na szczególne względy dotyczące długości ujęć i komfortowego dla widzów ruchu kamery, filmowcy muszą nagrywać sceny z myślą o ostatecznym montażu.

2 Należy unikać lub ograniczać używanie klipów z szybkimi ruchami kamery, ponieważ mogą one dezorientować widza. Zamiast tego warto postawić na zablokowane ujęcia lub powolne panoramowanie i ruchy na wózku dolly.

3 Niezbalansowane lub ekstremalne kąty kamery również mogą być dezorientujące – należy wybierać ujęcia, które są na tym samym poziomie, co filmowany obiekt lub postać i znajdują się w wygodnej odległości (nie za blisko).

4 W przypadku immersyjnych multimediów przyswajanie nowych środowisk trwa dłużej niż w przypadku oglądania sceny 2D. Cięcia powinny być wolniejsze, aby dać widzowi czas na dostosowanie się do nowego otoczenia.

5 Podczas przejść między klipami należy mieć na uwadze to, gdzie widz patrzył podczas poprzednich ujęć. Ważna akcja nie powinna rozgrywać się po lewej stronie kadru, jeśli uwaga widzów była wcześniej skierowana w prawo.

6 Dźwięk przestrzenny ma ogromne znaczenie dla fabuły – traktuj go jako kolejną postać w immersyjnych produkcjach.

Kompozytowanie VFX

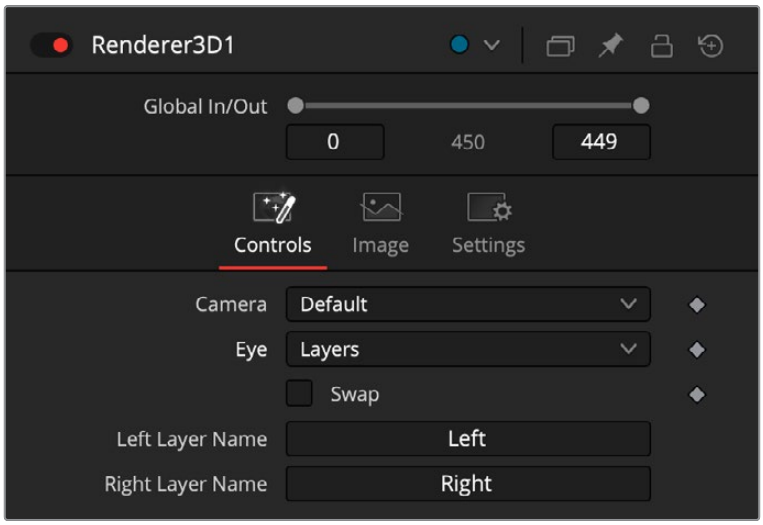
Po włączeniu opcji **Enable Apple Immersive workflow** w **Project Settings**, dedykowane parametry Fusion umożliwiają płynne stereoskopowe przepływy pracy i kompozytowanie zgodne z technologią immersyjną. Węzły MediaOut zawierają opcje wyjścia dla lewego i prawego oka, a narzędzie Immersive Patcher pozwala na projekcję i kompozycję w odwzorowaniu równoodległościowym.

Węzły MediaOut

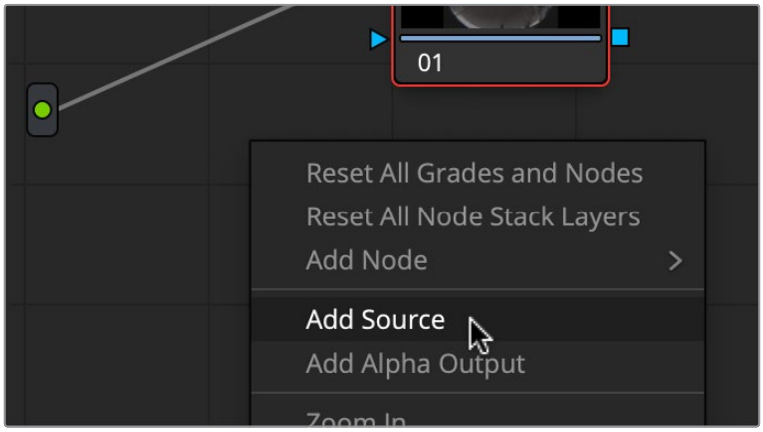
Po pierwszym wejściu na stronę Fusion podczas pracy nad klipem immersyjnym wykres węzłów zawiera dwa węzły MediaIn połączone z dwoma odpowiadającymi węzłami MediaOut.



W MediaOut w Inspektorze można użyć parametru **Stereo Eye** do kierowania sygnału wyjściowego oka.



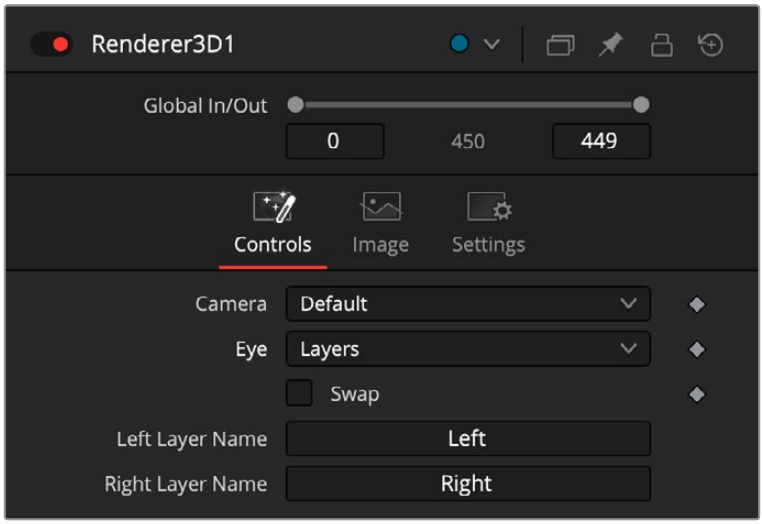
Domyślnie **MediaOut1** jest ustawiony na opcję **Left** oka, a **MediaOut2** na **Right**. Ta konfiguracja umożliwia natychmiastowe rozpoczęcie montażu kompozytów zgodnych z technologią stereo oraz nakładek graficznych. Podwójne węzły MediaOut powodują również wysłanie dwóch wariantów sygnału do edytora węzłów na stronie Koloru w celu dalszego udoskonalenia obrazu. Dostęp do sygnału **MediaOut2** na stronie Koloru można uzyskać, klikając prawym przyciskiem myszy w edytorze węzłów i wybierając opcję **Add Source**.



Gdy parametr **Stereo Eye** jest ustawiony na **Auto**, węzeł MediaOut łączy sygnały i przypisuje je do odpowiednich oczu na osi czasu. To umożliwia wykorzystanie wielowarstwowego przepływu pracy Fusion do wysyłania lewego i prawego sygnału do pojedynczego węzła MediaOut.

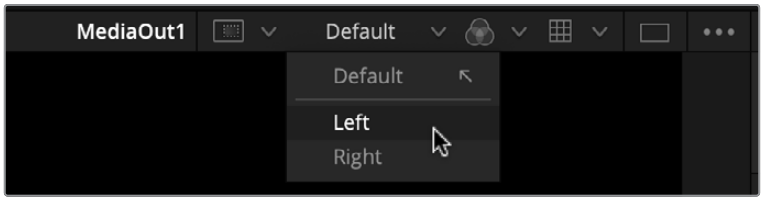
Renderer3D

Podczas pracy z grafiką 3D lub CGI na immersyjnej osi czasu, węzeł Renderer3D oferuje wiele ustawień oczu oraz możliwość przetwarzania lewej i prawej warstwy.



To pozwala tworzyć złożone nagłówki 3D lub całe sceny CGI i wysyłać je bezpośrednio do stereoskopowej osi czasu w projektach immersyjnych.

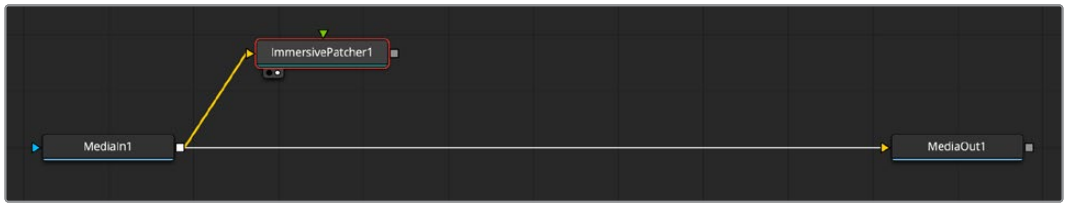
Podczas pracy z tą konfiguracją można użyć menu rozwijanego w prawym górnym rogu okien montażowych Fusion, aby przełączać się między widokiem lewego i prawego oka kompozycji.



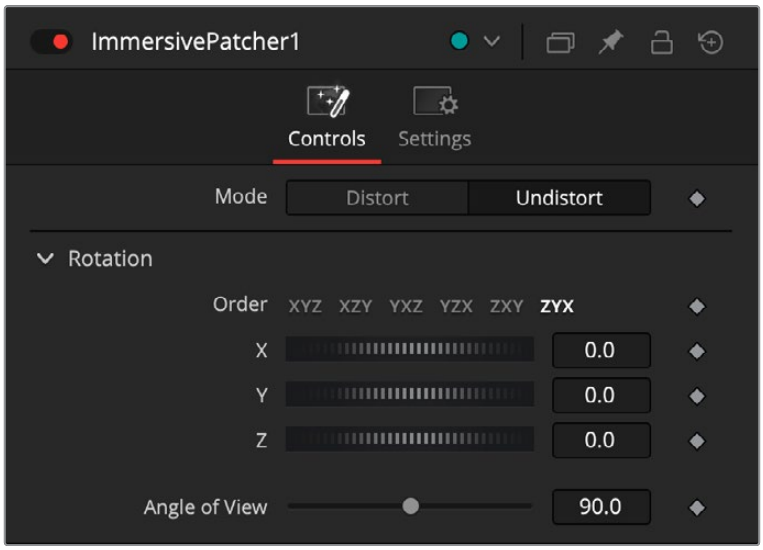
Kompozytowanie za pomocą Immersive Patcher

Narzędzie Immersive Patcher pozwala tymczasowo wyświetlić „niezniekształconą” sekcję immersyjnego klipu, aby ułatwić śledzenie, malowanie i kompozytowanie na płaskiej płaszczyźnie. Po zakończeniu można użyć kolejnego węzła Immersive Patcher, aby ponownie zniekształcić sygnał immersyjny i zintegrować nowo skompozytowane elementy ze sceną.

1 Dodaj Immersive Patcher do potoku węzłów.



2 Kliknij **Undistort** w Inspektorze.



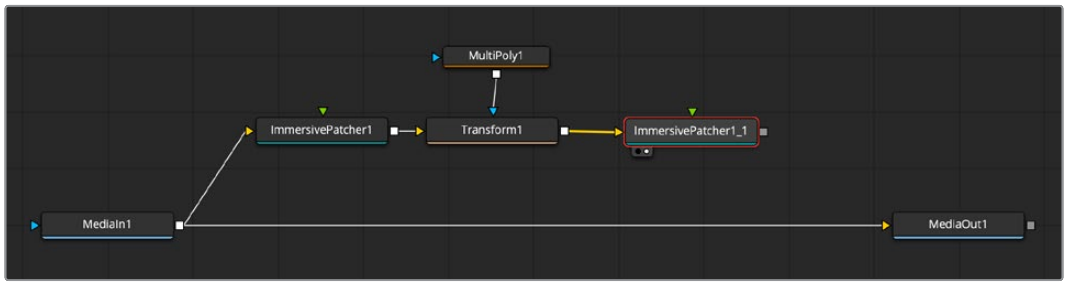
3 Użyj parametrów obrotu **X**, **Y** i **Z**, aby wybrać obszar immersyjnego obrazu do spłaszczenia.

4 Postępuj z kompozytem, tak jak w przypadku standardowego sygnału wideo 2D. Obejmuje to między innymi stosowanie standardowych metod malowania w celu usunięcia niepożądanych obiektów, pobieranie próbek fragmentów klatki za pomocą transformacji i wielokątów w celu rozszerzenia konfiguracji lub stosowanie masek, tekstu i zewnętrznych warstw obrazów/wideo w celu zastąpienia znaków lub ekranów.

   Po zakończeniu należy ponownie zniekształcić płaski obraz, aby dopasować go do oryginalnego wypaczenia immersyjnego.

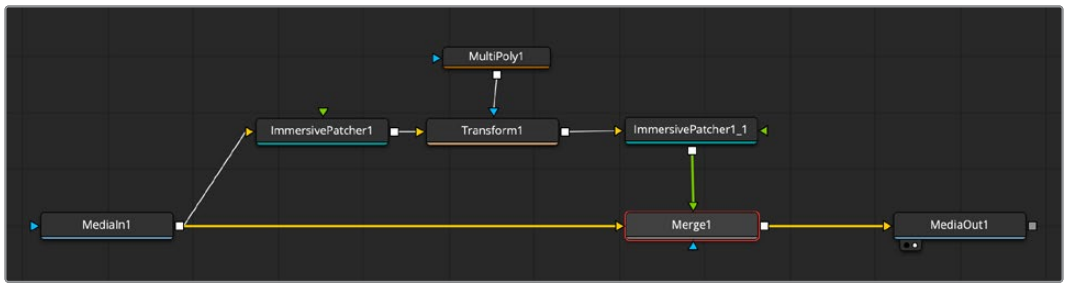
5 Skopiuj Immersive Patcher, aby zachować informacje dotyczące oryginalnego obrotu i i kąta widzenia.

6 Wklej Immersive Patcher za węzłami kompozytu i dołącz na końcu procesu kompozycji.



7 Kliknij **Distort** w Inspektorze.

8 Połącz skompozytowaną łatkę z oryginalnym klipem, aby zintegrować efekt z immersyjną sceną.



W przypadku współpracy można opuścić stronę Fusion i kontynuować montaż, miksowanie i korekcję kolorów immersyjnej osi czasu w DaVinci Resolve.

W przypadku wykonywania immersyjnych eksportów, wyrenderuj widok dla każdego oka jako EXR w ACES AP0, aby zachować maksymalne właściwości dla immersyjnej dystrybucji.

Korekcja kolorów

W tej sekcji omówiono narzędzia optyczne, kolorystyczne i 3D, których można używać podczas pracy z immersyjnymi multimediami na stronie Koloru w DaVinci Resolve.

Ze względu na unikalny charakter treści immersyjnych, nie wszystkie narzędzia będą możliwe do wykorzystania podczas postprodukcji. Inne narzędzia są obecnie niedostępne, ale są w trakcie dostosowywania do obsługi immersyjnych treści.

Redukcja szumów

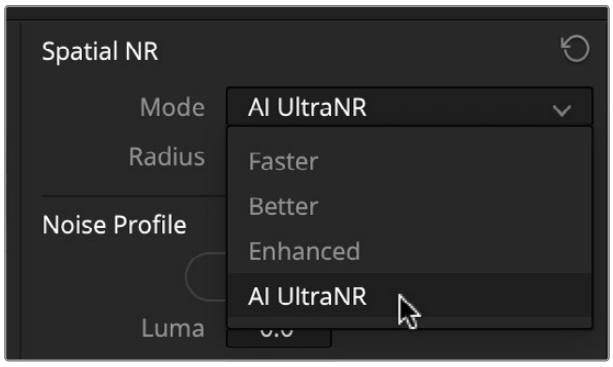
Redukcja szumów jest możliwa podczas pracy z immersyjnymi materiałami wideo. W palecie Motion Effects funkcja Spatial NR analizuje wzorzec szumu źródłowego pliku multimedialnego w celu jego maksymalnej redukcji przy jednoczesnym zachowaniu oryginalnych szczegółów obrazu.

1 Na pasku narzędzi strony Koloru otwórz paletę Motion Effects.

W kolumnie **Spatial NR** możesz wybrać pożądany tryb redukcji szumów.

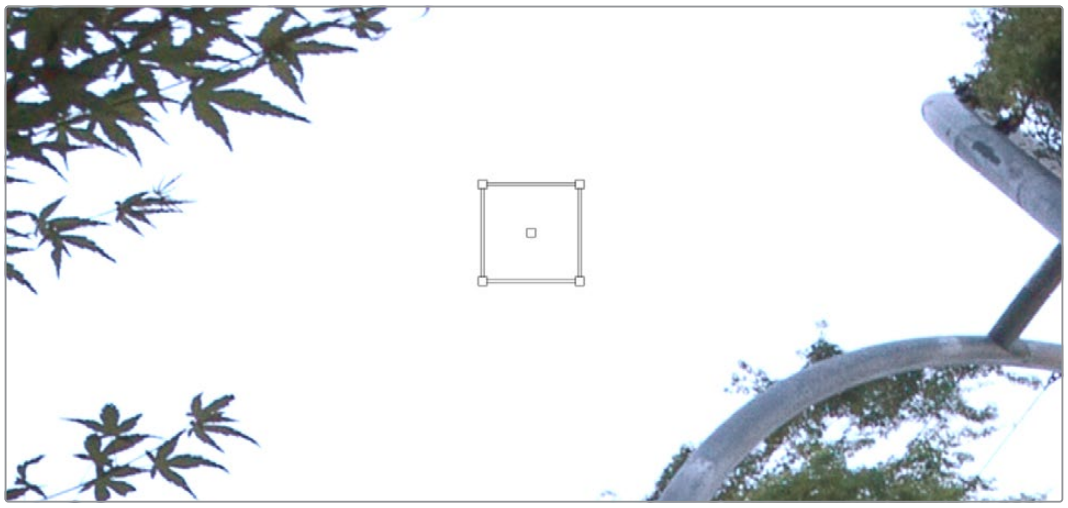
- Tryby **Faster**, **Better** i **Enhanced** przeprowadzą analizę wzorca w danym rozmiarze promienia i skonfigurują swój sygnał wyjściowy w oparciu o ustawiony próg. Są one wymienione w kolejności złożoności przetwarzania.
- **AI UltraNR** to oparty na SI identyfikator wzorca szumu, który wyodrębnia wzorzec szumu z sygnału, lokalizując i analizując „najbardziej płaską” sekcję kadru pod kątem samego szumu.

2 Ustaw tryb redukcji szumów przestrzennych na **AI UltraNR**.



3 Kliknij **Analyze**.

W oknie montażowym pojawi się pole zaznaczenia identyfikujące „najbardziej pusty” obszar danych pikseli w danym kadrze. Bez konkretnej treści wizualnej, dane wewnątrz kwadratu idealnie nadają się do wyodrębnienia cyfrowego szumu wytwarzanego przez sensor kamery.



W razie potrzeby można zmienić położenie i rozmiar pola zaznaczenia w oknie montażowym, ale generalnie nie jest to zalecane – AI UltraNR lepiej wykrywa optymalne obszary analizy niż ludzki wzrok.

Można również ręcznie dostosować ustawienia progów **Luma** i **Chroma** w **Noise Profile**, ale ponownie: AI UltraNR potrafi automatycznie osiągnąć idealną kombinację progów dla czystej redukcji szumów.

Podstawowa korekcja kolorów

Podstawowa korekcja kolorów działa tak samo, jak w przypadku zwykłego materiału wideo. Te same zasady normalizacji, balansowania i dopasowywania obowiązują podczas korzystania z kół kolorów Primaries i kół kolorów Log, palety HDR i HSL Curves. Palety ColorSlice i Color Warper również działają zgodnie z oczekiwaniami.

Korekcja wtórna

Metody selekcji oparte na pikselach, takie jak Qualifier, działają zgodnie z oczekiwaniami w przypadku immersyjnych multimediów.

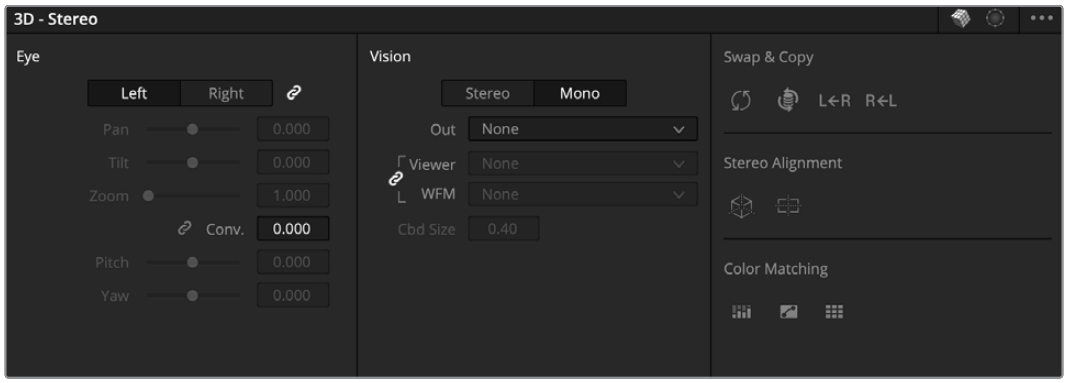
Selekcja oparta na obiektach lub oknach może działać dokładnie tylko w płaskim środowisku w odwzorowaniu równoodległościowym (lub obszarze widoku), co oznacza, że większość drugorzędnych metod wymagających precyzji pod kątem kształtu kwalifikacji, jest obecnie niedostępna. Obejmuje to Power Windows (oraz co za tym idzie, paletę Tracker) oraz narzędzie Magic Mask.

Zmiana rozmiaru

Dla treści immersyjnych nie są dostępne żadne elementy sterujące rozmiarem (tj. powiększanie i panoramowanie).

Paleta 3D

Paleta 3D strony Koloru zawiera elementy sterujące do zarządzania różnymi formatami 3D z kamer i grafiki 3D. W przeciwieństwie do innych formatów stereoskopowych nie trzeba dostosowywać ustawienia oczu, ponieważ obiektywy kamery Blackmagic URSA Cine Immersive, ogniskowa i przysłony są stałe.



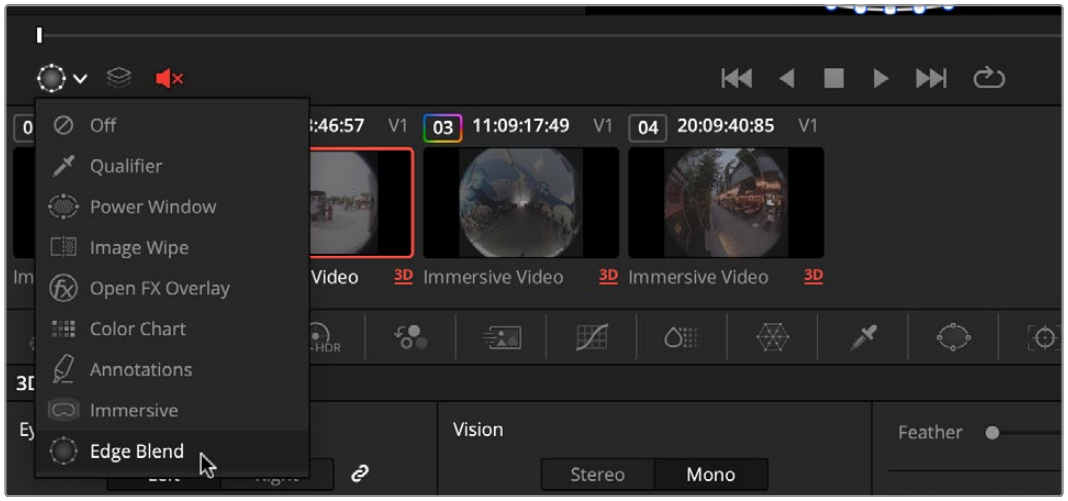
Tryb Edge Blend jest przeznaczony wyłącznie dla treści immersyjnych i umożliwia dostosowanie i zmiękczenie obwodu kadru immersyjnego wideo. Może to być wykorzystane do ukrycia widocznego sprzętu filmowego i ograniczenia pola widzenia obserwatora. Za tą krawędzią obserwator zobaczy tło.

1 Otwórz paletę 3D.

2 W prawym górnym rogu zmień tryb na **Edge Blend**.



3 Upewnij się, że ekranowe elementy sterujące okna montażowego są ustawione na opcję **Edge Blend**.



Na obwodzie kadru wideo powinien pojawić się okrąg punktów kontrolnych z uchwytem kotwiczenia pośrodku.

4 Przeciągnij punkt kontrolny, aby zmienić jego położenie



lub kliknij prawym przyciskiem myszy, aby usunąć punkt.

5 W prawej kolumnie palety 3D użyj parametru **Feather**, aby zmiękczyć krawędź kadru.

6 Wybierz opcję **visionOS Effect**, aby zapewnić prawidłowe przetwarzanie danych nowych krawędzi podczas odtwarzania na Apple Vision Pro. Jeśli opcja ta nie zostanie wybrana, krawędź zostanie zagnieżdżona w renderowanym pliku wideo.

7 W lewej kolumnie użyj zakładek **Left** i **Right**, aby przełączać się między oczami w oknie montażowym.

Należy zauważyć, że wiele elementów sterujących w palecie 3D jest wyszarzonych. Parametry w programie DaVinci Resolve będą wyszarzone, jeśli nie mają zastosowania do multimediów poddawanych aktywnej obróbce, którymi w tym przypadku są treści immersyjne.

8 W środkowej kolumnie użyj zakładek **Stereo** i **Mono**, aby przełączać się między widokiem pojedynczego obiektywu a widokiem podwójnego obiektywu immersyjnego klipu wideo.

Zarządzanie kolorem CST

Zastosowanie efektu Color Space Transform (CST) do węzłów w edytorze węzłów strony Koloru pozwoli na dokładne mapowanie kolorów i niestandardowe sterowanie maksymalną luminancją nitów na osi czasu.

Zalecany przepływ pracy obejmuje cztery węzły CST:

1. INPUT CST – źródło z kamery do roboczej przestrzeni kolorów.

   Korekcja kolorów odbywa się pomiędzy tymi dwoma węzłami.

2. OUTPUT CST – robocza przestrzeń kolorów do wyświetlacza HDR Mac.
3. MONITOR CST – 1000 nitów do 250 nitów w celu monitorowania HDR.
4. AVP CST – 250 nitów do 108 nitów w celu monitorowania i dystrybucji na Apple Vision Pro.



Mapowanie osi czasu CST

Najpierw należy zmapować wejściową przestrzeń kolorów źródła kamery na „roboczą” przestrzeń kolorów osi czasu. Kamera Blackmagic URSA Cine Immersive działa w gamie kolorów Blackmagic Design Wide Gamut Gen 4/5 z krzywą logarytmiczną DaVinci Intermediate.

1. Otwórz panel Effects.
2. Przeciągnij Color Space Transform na pusty węzeł (01) w edytorze węzłów.

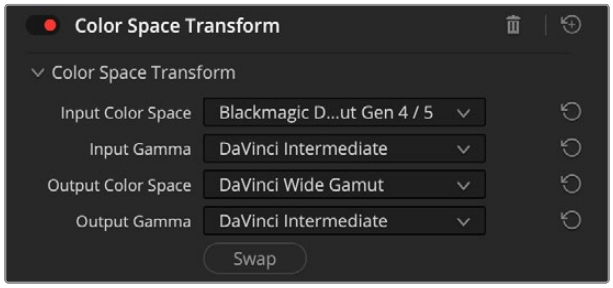
   Podczas mapowania wejściowej przestrzeni kolorów do przestrzeni kolorów osi czasu, ten węzeł powinien znajdować się wśród pierwszych węzłów w strukturze węzłów, przed jakąkolwiek korekcją kolorów.

   W przypadkach mapowania wejściowej przestrzeni kolorów bezpośrednio do przestrzeni kolorów monitorowania lub dystrybucji, ten węzeł powinien znajdować się wśród ostatnich węzłów w strukturze węzłów, po operacjach korekcji kolorów.

3. Nadaj węzłowi 01 etykietę **INPUT CST**.

> **UWAGA** Etykiety węzłów użyte w niniejszej instrukcji zostały stworzone w celu ułatwienia odniesienia do dokumentacji i zrzutów ekranu. Możesz użyć dowolnej i najwygodniejszej dla Ciebie konwencji nazewnictwa lub całkowicie z niej zrezygnować.

4. Ustaw **Input Color Space** na **Blackmagic Design Wide Gamut Gen 4/5**.
5. Ustaw **Input Gamma** na **DaVinci Intermediate**.
6. Ustaw **Output Color Space** na **DaVinci Wide Gamut**.
7. Ustaw **Output Gamma** na **DaVinci Intermediate**.



8 Utwórz nowy węzeł szeregowy (02).

9 Skopiuj i wklej pierwszy węzeł (01) do nowego węzła (02).

10 Nadaj węzłowi 02 etykietę **OUTPUT CST**.

Wyjściowa przestrzeń kolorów i gamma powinny stanowić wejście w tym nowym węźle szeregowym, aby zapewnić spójność przestrzeni kolorów osi czasu w strukturze węzłów.

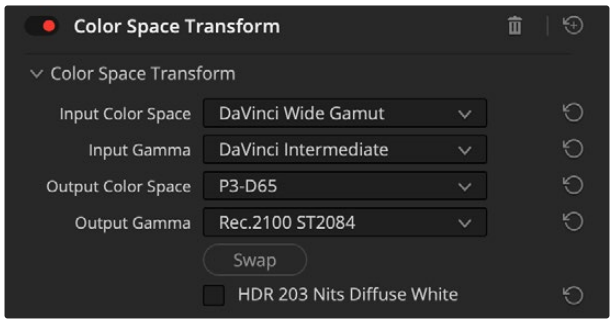
11 Naciśnij przycisk **Swap**, aby przełączyć parametry wejścia i wyjścia.

Input Color Space i **Gamma** są teraz odpowiednie dla tego węzła CST.

12 Ustaw **Output Color Space** na **P3-D65**. Jest to obecnie domyślny gamut monitorów Mac.

13 Ustaw **Output Gamma** na **Rec.2100 ST2084**, który jest standardem HDR dla sygnału o jasności 1000 nitów.

Wszystkie operacje korekcji kolorów będą wykonywane między tymi dwoma węzłami. Praca z zarządzaniem kolorami opartym na węzłach oferuje unikalną możliwość wykonywania działań poza granicami zarządzania kolorami na osi czasu, co wielu kolorystów wykorzystuje do stosowania specyficznej korekcji kolorów, efektów lub elementów innych producentów.



Profile ICC

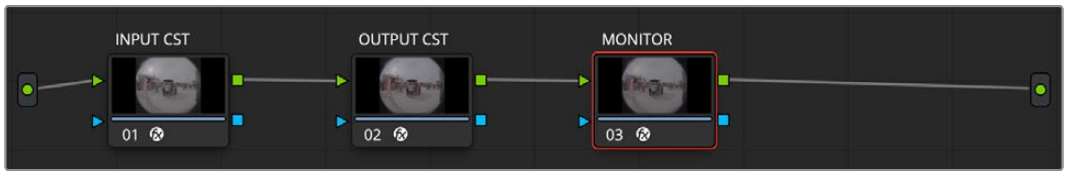
Aby upewnić się, że sygnał z Blackmagic URSA Cine Immersive jest prawidłowo mapowany na monitorze Mac, przejdź do **System Settings** > **Displays** i sprawdź, czy **Preset** w profilu kolorów jest ustawiony na **Apple XDR Display (P3-1600 nits)**.

Mapowanie CST monitora referencyjnego

Można użyć dodatkowych CST, aby tymczasowo ograniczyć maksymalne wartości sygnału wyjściowego (w nitach) do monitorowania podczas korekcji kolorów.

1 Utwórz nowy węzeł szeregowy (03) za węzłem zarządzania kolorami OUTPUT CST (02).

2 Nadaj węzłowi 03 etykietę **MONITOR**.



3 Dodaj efekt **Color Space Transform** do węzła MONITOR (03).

Wartości wejściowe muszą być zgodne z sygnałem wychodzącym z poprzedniego CST.

4 Ustaw **Input Color Space** na **P3-D65**.

5 Ustaw **Input Gamma** na **Rec.2100 ST2084**.

Celem tego węzła CST jest jedynie zmapowanie luminancji obrazu na niższą wartość (w nitach), co oznacza, że parametry sygnału wejściowego i wyjściowego mogą pozostać takie same.

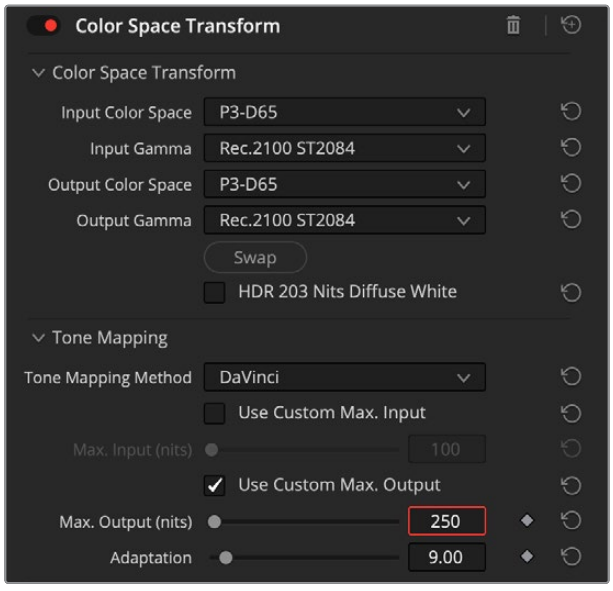
6 Ustaw **Output Color Space** na **P3-D65**.

7 Ustaw **Output Gamma** na **Rec.2100 ST2084**.

UWAGA W niektórych okolicznościach nie ma potrzeby ustawiania przestrzeni kolorów lub gammy CST, jeśli sygnał wejściowy i wyjściowy jest identyczny (ponieważ nie występuje mapowanie). W tym przypadku jest to konieczne, ponieważ zastosowanie niestandardowego mapowania nitów zależy od wartości i kształtu krzywej logarytmicznej sygnału.

8 Zaznacz opcję **Use Custom Max. Output**.

9 Ustaw **Max. Output (nits)** na **250**.



Sygnał o jasności 1000 nitów zostanie teraz zmapowany (przy użyciu **Tone Mapping Method** ustawionej na opcję **DaVinci**) do 250 nitów.

Dzięki temu uzyskamy obraz, który będzie bardzo zbliżony do renderowanego sygnału Apple Vision Pro na monitorze referencyjnym HDR (takim jak Flanders XMP310, Sony BVM-HX310 itp.).

Aby jednak monitorować projekt na Apple Vision Pro, należy wykonać jeszcze jedno działanie mapowania tonów.

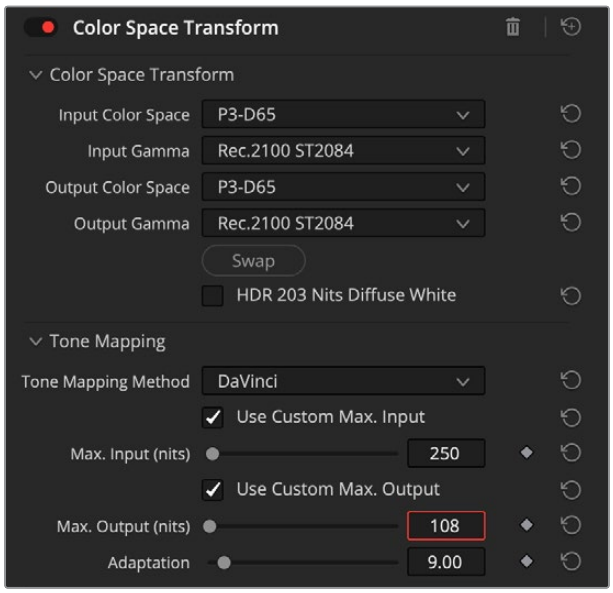
Monitorowanie CST Apple Vision Pro

Końcowego węzła CST można użyć do monitorowania na żywo, podglądu i dystrybucji gotowego materiału do urządzeń Apple Vision Pro.

1. Utwórz nowy węzeł szeregowy (04) za węzłem MONITOR (03).
2. Skopiuj i wklej węzeł MONITOR (03) do węzła 04. Będziesz używał tych samych parametrów wejściowych i wyjściowych, zmieniając jedynie liczbę nitów.
3. Nadaj węzłowi 04 etykietę **AVP**.



4. Zaznacz opcję **Use Custom Max. Input**.
5. Ustaw **Max. Input (nits)** na **250**, aby poprawnie zinterpretować zmapowany sygnał z węzła MONITOR.
6. Ustaw **Max. Output (nits)** na **108**, co jest maksymalną wartością jasności w nitach do celów monitorowania i dystrybucji do Apple Vision Pro.



Możesz teraz przejść do korekcji kolorów, wyłączając węzły CST, jeśli (i kiedy) jest to konieczne w oparciu o przepływ pracy. Na przykład podczas wstępnej korekcji kolorów, w której używasz zewnętrznego monitora referencyjnego, należy wyłączyć węzeł AVP (04), aby upewnić się, że na monitorze referencyjnym HDR prawidłowo wyświetla się sygnał o jasności 250 nitów. Po przełączeniu na podgląd w Apple Vision Pro lub w celu ostatecznej dystrybucji materiału, należy ponownie włączyć węzeł AVP (04).

Czy CST można umieścić w edytorze węzłów osi czasu?

Nie. W tej chwili mapowanie tonalne klipu zależy od dokładnego zastosowania parametru **Adaptation** w sekcji **Color Space Transform**, które musi być wykonane indywidualnie dla każdego klipu.

Audio

Apple Vision Pro i DaVinci Resolve obsługują dźwięk przestrzenny i ambisoniczny, formaty AVP (Apple Vision Pro) i AIA (Apple Immersive Audio).

Wiele aspektów postprodukcji audio wygląda tak samo podczas pracy z dźwiękiem przestrzennym. Jedną z głównych różnic jest to, że zamiast monitorować dźwięk za pomocą wysokiej klasy głośników, inżynierowie dźwięku powinni używać słuchawek nausznych lub dousznych AirPods, aby uzyskać prawdziwą immersję przestrzenną i lepsze odwzorowanie środowiska odsłuchowego publiczności.

Podczas gdy w miksach 2D większość dialogów jest miksowana do kanałów centralnych, dźwięk przestrzenny musi być miksowany do obszaru, z którego pochodzi dźwięk. Ducking, korekcja dźwięku i jego balansowanie są konieczne do miksowania ścieżek audio w sposób, który nie przytłacza ani nie dezorientuje słuchacza.

Immersyjne elementy przestrzenne można łączyć ze zwykłą ścieżką stereo, dzięki czemu niektóre dźwięki pozostają niezmienne bez względu na to, w którą stronę porusza się głowa. Jest to pomocne w utrzymaniu wyraźnego oraz łatwego do interpretacji dźwięku niediegetycznego, takiego jak ścieżki dźwiękowe i lektorskie.

Dźwięki otoczenia są zwykle interpretowane jako cichsze w dźwięku przestrzennym, więc często wymagają bardziej agresywnego miksowania, żeby były lepiej słyszalne. Nie powinno się to jednak odbywać kosztem dezorientacji słuchacza.

Pogłos w znacznym stopniu przyczynia się do immersji, ponieważ jest czymś, czego słuchacze doświadczają naturalnie każdego dnia – każdy dźwięk wytworzony w danym otoczeniu odbija się od powierzchni otaczających słuchacza, tworząc efekt rozprzestrzeniania się pogłosu. W związku z tym w projektowaniu dźwięku przestrzennego zaleca się stosowanie pogłosu, aby zapewnić pełną immersję.

Format Apple Immersive Audio (AIA) obejmuje kilka formatów:

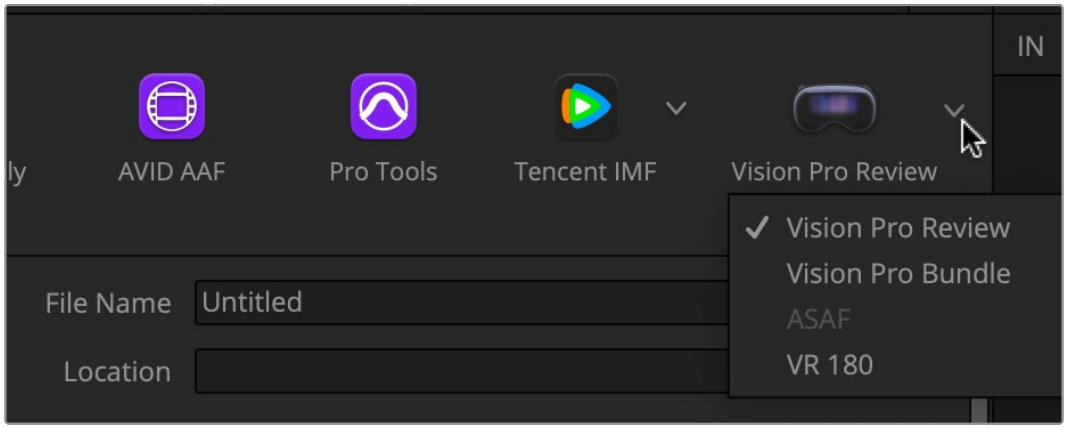
- **Apple Spatial Audio Format (ASAF)** to nieprzetworzony format roboczy audio.
- **Apple Positional Audio Codec (APAC)** to zakodowany format .mp4 odtwarzany na urządzeniach obsługujących APAC, takich jak słuchawki AirPods lub z pliku AIVU na Apple Vision Pro.

Apple Spatial Audio Format (ASAF)

Apple Spatial Audio Format (ASAF) to format dźwięku immersyjnego opracowany głównie z myślą o zastosowaniach w VR i AR w Apple Vision Pro. Wyłącznie w systemie macOS, ASAF jest obsługiwany przez ProTools (za pośrednictwem wtyczek) i DaVinci Resolve (wbudowany). ASAF oferuje zaawansowane możliwości techniczne, w tym wbudowaną obsługę źródeł dźwięku ambisonicznego (zwanych Scenes) oraz routing poprzez przypisanie magistrali. Miksowanie odbywa się wirtualnie, bez fizycznej magistrali, a użytkownicy mogą wybierać spośród różnych formatów monitorowania, w tym binauralnego (dwuusznego) z funkcją śledzenia ruchów głowy dzięki bezpośredniej integracji z Apple AirPod. Panoramowanie jest obsługiwane za pomocą istniejących narzędzi DaVinci Resolve, takich jak okno dialogowe panoramowania, automatyzacja, SpaceView i narzędzia do panoramowania w oknie montażowym, z dodatkowymi elementami sterującymi w Inspektorze. Umożliwiają one precyzyjne pozycjonowanie źródeł w przestrzeni, w tym automatyczne śledzenie względem materiału wideo w odwzorowaniu równoodległościowym (lat-long). Miksy ASAF można eksportować w formatach WAV lub MP4 (wraz z metadanymi) za pomocą opcji **Immersive Audio** > **Export Master File**, ale obecnie nie można ich importować ani odtwarzać w DaVinci Resolve.

Dystrybucja

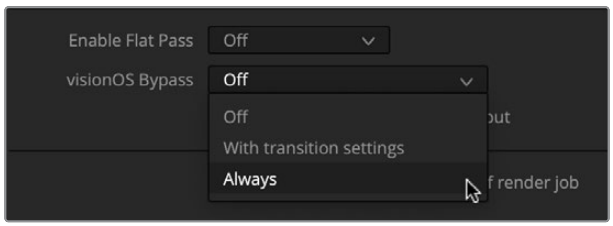
Na stronie Dystrybucji dwa ustawienia renderowania umożliwiają renderowanie plików Apple Vision Pro w celu ich sprawdzenia i ostatecznej dystrybucji.



- **Vision Pro Review** – tworzy mały pakiet MV-HEVC na potrzeby sprawdzania. Apple „udostępnia” Apple Vision Pro gotowy plik przez Wi-Fi i umożliwia oglądanie w aplikacji Apple Immersive Video Utility (AIVU), która jest dostępna w App Store.
- **Vision Pro Bundle** – tworzy duży pakiet plików ProRes, który można zarchiwizować lub przetworzyć w Apple Compressor, aby przygotować go do odtwarzania na Apple TV.
- **ASAF** – renderuje miks audio ASAF w formatach WAV lub MP4.
- **VR 180** – generuje stereoskopowy plik wideo zgodny z technologią VR180.

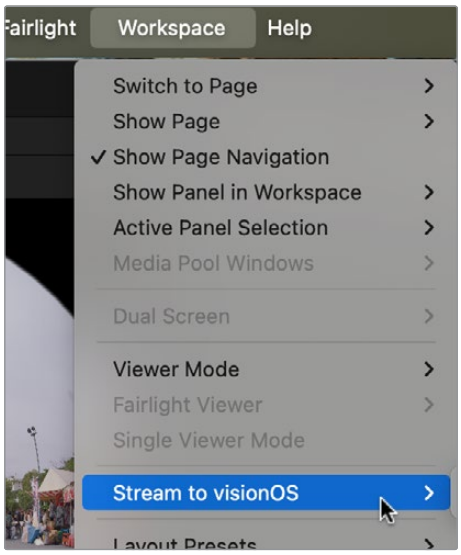
Pomijanie przejść

Jeśli do immersyjnej osi czasu zostanie dodane przejście, zostanie ono wstępnie wyrenderowane i przekształcone w pojedynczą ścieżkę odtwarzania. XML wygenerowany dla treści immersyjnych opisuje proces przełączania między tymi renderowanymi strumieniami (materiał filmowy, po którym następuje wstępnie renderowane przejście, a następnie powrót do materiału filmowego). W Advanced Settings panelu Render Settings dostępna jest opcja pomijania przejść visionOS (**Always**), stosowania ich zgodnie z ich stanem na osi czasu (**With transition settings**) lub uwzględniania ich, gdy pojawiają się na osi czasu (**Off**).



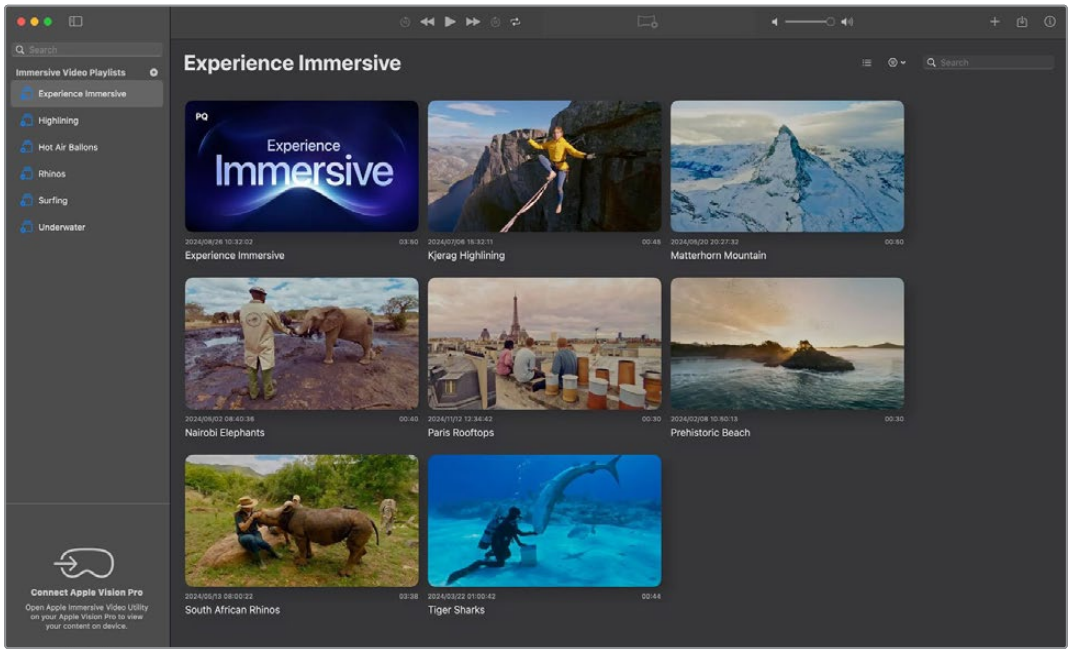
Streaming do Apple Vision Pro

Istnieje możliwość przesyłania strumieniowego do Apple Vision Pro ze stron Edycji, Koloru i Dystrybucji w DaVinci Resolve. W tym celu należy przejść do zakładki **Workspace**, wybrać opcję **Stream to visionOS**, a następnie swoje urządzenie AVP. Do płynnej transmisji strumieniowej zalecana jest szybka sieć bezprzewodowa (minimum Wi-Fi 5; preferowane Wi-Fi 6).



Informacje dotyczące Apple Immersive Video Utility

Apple Immersive Video Utility to specjalistyczna aplikacja dla systemów macOS i visionOS zaprojektowana w celu usprawnienia tworzenia, zarządzania i odtwarzania treści Apple Immersive Video. Narzędzie to, stworzone z myślą o profesjonalnych filmowcach i twórcach immersyjnych multimediów, umożliwia użytkownikom importowanie, organizowanie, pakowanie i przeglądanie stereoskopowych plików wideo 8K z polem widzenia 180 stopni i dźwiękiem przestrzennym. Obsługuje tworzenie plików AIVU (Apple Immersive Video Universal), umożliwiając sprawdzanie i modyfikowanie metadanych w ramach postprodukcyjnego przepływu pracy. Użytkownicy mogą tworzyć listy odtwarzania, sortować i przeszukiwać swoje biblioteki, a nawet przesyłać strumieniowo treści bezpośrednio do jednego lub kilku urządzeń Apple Vision Pro.



Developer Information

Camera Control REST API

If you are a software or hardware developer you can build custom applications or leverage ready to use tools such as REST client or Postman to seamlessly control and interact with your compatible Blackmagic camera using Camera Control REST API. This API enables you to perform a wide range of operations, such as starting or stopping recordings, accessing disk information and much more. Whether you're developing a custom application tailored to your specific needs or utilizing existing tools, this API empowers you to unlock the full potential of your Blackmagic camera with ease. We look forward to seeing what you come up with!

NOTE It's important to mention that controlling Blackmagic cameras via REST API relies on the web manager being enabled on each compatible Blackmagic camera. Enable the web media manager in the Blackmagic Camera Setup 'network access' settings for each camera you are controlling.

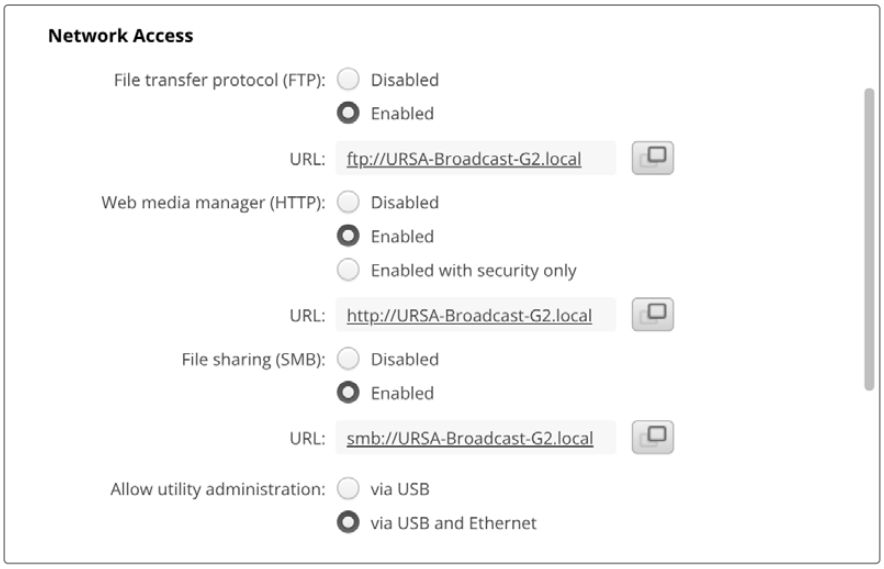
The following Blackmagic cameras are compatible with Camera Control REST API:

- Blackmagic URSA Cine 12K LF
- Blackmagic URSA Cine 17K 65
- Blackmagic URSA Cine Immersive
- Blackmagic PYXIS 6K
- Blackmagic Cinema Camera 6K
- Blackmagic URSA Broadcast G2
- Blackmagic Micro Studio Camera 4K G2
- Blackmagic Studio Camera 4K Plus
- Blackmagic Studio Camera 4K Pro
- Blackmagic Studio Camera 6K Pro
- Blackmagic Studio Camera 4K Plus G2
- Blackmagic Studio Camera 4K Pro G2

Sending API Commands

To send an API command to your camera from a third party application such as Postman, add /control/api/v1/ to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/api/v1/

You can find the Web media manager URL and IP address information in Blackmagic Camera Setup.



The Web media manager URL in Blackmagic Camera Setup

Downloading API's from your Camera

You can download REST API YAML documentation from your camera by adding /control/documentation.html to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/documentation.html

NOTE It's worth noting that changing the camera name in Blackmagic Camera Setup will also change the camera's Web media manager URL.

Livestream Control API

API for controlling Livestreams on Blackmagic Design products.

GET /livestreams/0

Get the livestream's current status.

Response

200 - Livestream's current status.

The response is JSON.

Name	Type	Description
status (required)	string	Possible values are: Idle, Connecting, Streaming, Flushing, Interrupted.
bitrate (required)	integer	Current bitrate (bps).
effectiveVideoFormat (required)	string	Effective video format for the livestream, serialised as a string.
duration	integer	Current stream duration in seconds. Absent if livestream is idle.
cache	integer	Current stream cache usage percentage.

GET /livestreams/0/start

Determine if the livestream is active.

Response

200 - Livestream active status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is active.

PUT /livestreams/0/start

Start the livestream.

Response

204 - Livestream started.

GET /livestreams/0/stop

Determine if the livestream is inactive.

Response

200 - Livestream inactive status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is inactive.

PUT /livestreams/0/stop

Stop the livestream.

Response

204 - Livestream stopped.

GET /livestreams/0/activePlatform

Get the currently selected platform configuration for the livestream.

Response

200 - Livestream active platform configuration.

The response is JSON.

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

PUT /livestreams/0/activePlatform

Set the currently selected platform configuration for the livestream.

Parameters

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

Response

204 - Livestream active platform configuration updated.

400 - Bad Request

GET /livestreams/platforms

Get the list of available platforms.

Response

200 - List of available platforms.

The response is JSON.

Name	Type	Description
	array	List of available platforms names.
[i]	string	Platform name.

GET /livestreams/platforms/platformName

Get the service configuration for a platform.

Parameters

Name	Type	Description
platformName (required)	string	Name of the platform.

Response

200 - Service configuration for specified platform.

The response is JSON.

Livestream platform service configuration.

Name	Type	Description
platform (required)	string	Corresponding platform name.
key	string	Default stream key.
servers (required)	array	List of server configurations.
servers[i]	object	Server configuration.
servers[i].server (required)	string	Server name.
servers[i].url (required)	string	Livestream destination.
servers[i].srtExtensions	array	Miscellaneous tags used for SRT livestreams.
servers[i].srtExtensions[i]	object	Dictionary object mapping SRT tag strings to values.
servers[i].srtExtensions[i][**key**]	string	SRT tag value.
servers[i].group	string	Logical grouping of the server.
profiles (required)	array	List of profile configurations.
profiles[i]	object	Quality configuration.
profiles[i].profile (required)	string	Quality level name.
profiles[i].configs (required)	array	List of video format configurations.
profiles[i].configs[i]	object	Video format configuration for profiles.
profiles[i].configs[i].resolution (required)	string	Video format serialised as a string.
profiles[i].configs[i].fps (required)	string	Frames per second.
profiles[i].configs[i].bitrate (required)	integer	Pixel bitrate (bps).
profiles[i].configs[i].audioBitrate	integer	Audio bitrate (bps).
profiles[i].configs[i].keyFrameInterval	integer	How often a key frame is sent, in seconds.
profiles[i].configs[i].videoCodecs	array	Supported video encoding algorithm/s.

Name	Type	Description
profiles[i].configs[i].videoCodecs[i]	string	Video encoding algorithm. Possible values are: H264, H265.
profiles[i].lowLatency (required)	boolean	If true, fewer frames will be buffered in the livestream.
defaultProfile	string	Quality level name.
credentials	object	Credientials used for RTMP streams.
credentials.username (required)	string	The username part of the creditials. Only used for RTMP streams.
credentials.password (required)	string	Used for RTMP streams, also used as Passphrase for SRT streams.
customizableUrlEnabled	boolean	True when the server URL is customizable.

400 - Bad Request

GET /livestreams/customPlatforms

Get a list of custom platform files.

Response

200 - List of custom platform files.

The response is JSON.

Name	Type	Description
	array	List of custom platform file names.
[i]	string	Custom platform file name.

DELETE /livestreams/customPlatforms

Remove all custom configuration files.

Response

204 - All custom configuration files removed.

GET /livestreams/customPlatforms/filename

Get a custom platform file.

Parameters

Name	Type	Description
filename (required)	string	Name of the file to get.

Response

200 - Custom platform file.

The response is XML.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

404 - Not Found

PUT /livestreams/customPlatforms/filename

Update a custom platform file if it exists, if not, create a new file with the given file name.

Parameters

Name	Type	Description
filename (required)	string	Name of the file to update/create.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

Response

204 - Custom platform file created or updated.

400 - Bad Request

DELETE /livestreams/customPlatforms/filename

Remove the given custom platform file.

Parameters

Name	Type	Description
filename (required)	string	Name of the file to be removed.

Response

204 - Custom platform file removed.

404 - Not Found

Clips Control API

API for listing clips on disk.

GET /clips

Get the list of clips on the active disk.

Response

200 - List of clips on the active disk.

The response is JSON.

List of media clips.

Name	Type	Description
clips (required)	array	
clips[i]	object	Media clip.
clips[i].clipUniqueId (required)	integer	Unique ID used to identify this clip.
clips[i].filePath	string	Path to the file relative to the root of a mount.
clips[i].fileSize	integer	Size of file on disk in bytes.
clips[i].codecFormat	object	
clips[i].codecFormat.codec	string	Currently selected codec.

Name	Type	Description
clips[i].codecFormat.container	string	Multimedia container format.
clips[i].videoFormat	object	Video format configuration.
clips[i].videoFormat.name (required)	string	Video format serialised as a string.
clips[i].videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
clips[i].videoFormat.height	number	Height dimension of video format.
clips[i].videoFormat.width	number	Width dimension of video format.
clips[i].videoFormat.interlaced	boolean	Is the display format interlaced?
clips[i].startTimecode	string	Start timecode of the clip serialised as string.
clips[i].durationTimecode	string	Duration of the clip in timecode format serialised as string.
clips[i].frameCount	integer	Number of frames in clip; duration of the clip in frames.

404 - There is no active disk.

Media Pool Control API

API to manage media pool and handle uploads and project data.

GET /cloud/projects

List all projects within the media pool.

Response

200 - Successfully retrieved the list of all projects.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].libraryID	string	
[i].name	string	
[i].private	boolean	
[i].shared	boolean	
[i].clips	array	List of clips associated with the project.
[i].clips[i]	string	
[i].status	object	
[i].status.numClipsRequested	integer	
[i].status.numClipsComplete	integer	
[i].status.uploadPercent	integer	Percentage of upload completion.
[i].status.numClipsPaused	integer	
[i].status.outOfSpace	boolean	
[i].status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
[i].status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/active

Retrieve data of the actively uploading project.

Response

200 - Successfully retrieved the active project's data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/projectID

Retrieve specific project data by project ID.

Parameters

Name	Type	Description
projectID (required)	integer	Unique identifier of the project.

Response

200 - Successfully retrieved the project data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	

Name	Type	Description
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

404 - Project not found.

GET /cloud/clips

List all clips within the media pool.

Response

200 - Successfully retrieved the list of all clips.

The response is JSON.

Name	Type	Description
	array	
[i]	string	

GET /cloud/clips/activeUploading

Retrieve data of actively uploading clips.

Response

200 - Successfully retrieved the list of actively uploading clips.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].path	string	
[i].projectID	integer	
[i].status	object	
[i].status.projectID	integer	
[i].status.outOfSpace	boolean	
[i].status.proxyExtension	string	
[i].status.growingFile	boolean	
[i].status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.originalClipTotalSize	integer	
[i].status.proxyClipTotalSize	integer	
[i].status.originalClipCompletedSize	integer	
[i].status.proxyClipCompletedSize	integer	
[i].status.secsRemaining	integer	

GET /cloud/clips/deviceName/path

Retrieve specific clip data by device and path.

Parameters

Name	Type	Description
deviceName (required)	string	Name of the device where the clip is stored.
path (required)	string	Path to the clip.

Response

200 - Successfully retrieved the clip data.

The response is JSON.

Name	Type	Description
path	string	
projectID	integer	
status	object	
status.projectID	integer	
status.outOfSpace	boolean	
status.proxyExtension	string	
status.growingFile	boolean	
status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.originalClipTotalSize	integer	
status.proxyClipTotalSize	integer	
status.originalClipCompletedSize	integer	
status.proxyClipCompletedSize	integer	
status.secsRemaining	integer	

404 - Clip not found.

Monitoring Control API

API for monitoring and controlling display settings in video equipment.

GET /monitoring/display

Retrieve a list of all display names.

Response

200 - Returns a list of display names.

The response is JSON.

Name	Type	Description
displays	array	List of display names available.
displays[i]	string	

GET /monitoring/displayName/cleanFeed

Get the clean feed enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/displayName/cleanFeed

Set the clean feed enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Clean feed enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/displayName/displayLUT

Get the display LUT enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

400 - Invalid display name.

404 - Display name not found.

PUT /monitoring/displayName/displayLUT

Set the display LUT enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Display LUT enabled/disabled successfully.

400 - Invalid input.

422 - Unprocessable Entity - Unable to process the contained instructions.

GET /monitoring/displayName/zebra

Get the zebra enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/displayName/zebra

Set the zebra enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Zebra enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/displayName/focusAssist

Get the focus assist enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/displayName/focusAssist

Set the focus assist enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/focusAssist

Get the focus assist settings.

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

404 - Display name not found.

PUT /monitoring/focusAssist

Set the focus assist settings.

Parameters

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/displayName/frameGuide

Get the frame guide enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame guide enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/displayName/frameGuide

Set the frame guide enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame guide state updated successfully.

422 - Unable to update the frame guide state.

GET /monitoring/frameGuideRatio

Get the current frame guide ratio.

Response

200 - Returns the current frame guide ratio.

The response is JSON.

Name	Type	Description
ratio	string	The frame guide ratio.

PUT /monitoring/frameGuideRatio

Set the frame guide ratio.

Parameters

Name	Type	Description
ratio	string	The frame guide ratio.

Response

204 - Frame guide ratio updated successfully.

422 - Unable to update the frame guide ratio.

GET /monitoring/frameGuideRatio/presets

Get the presets for frame guide ratios.

Response

200 - Returns a list of preset frame guide ratios.

The response is JSON.

Name	Type	Description
presets	array	
presets[i]	string	A frame guide ratio.

GET /monitoring/displayName/frameGrids

Get the frame grids enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame grids enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/displayName/frameGrids

Set the frame grids enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame grids state updated successfully.

422 - Unable to update the frame grids state.

GET /monitoring/frameGrids

Get the global frame grids settings.

Response

200 - Returns the current frame grids settings.

The response is JSON.

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

PUT /monitoring/frameGrids

Set the global frame grids settings.

Parameters

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

Response

204 - Frame grids settings updated successfully.

400 - Invalid input, check the number of frame grids or values.

422 - Unable to update the frame grids settings.

GET /monitoring/displayName/safeArea

Get the safe area enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the safe area enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/displayName/safeArea

Set the safe area enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Safe area state updated successfully.

422 - Unable to update the safe area state.

GET /monitoring/safeAreaPercent

Get the current safe area percentage.

Response

200 - Returns the current safe area percentage.

The response is JSON.

Name	Type	Description
percent	integer	Safe area coverage percentage.

PUT /monitoring/safeAreaPercent

Set the safe area percentage.

Parameters

Name	Type	Description
percent	integer	Safe area coverage percentage to set.

Response

204 - Safe area percentage updated successfully.

400 - Invalid percentage value.

422 - Unable to update the safe area percentage.

GET /monitoring/displayName/falseColor

Get the false color enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the false color enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/displayName/falseColor

Set the false color enable state for a specific display.

Parameters

Name	Type	Description
displayName (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - False color state updated successfully.

422 - Unable to update the false color state.

Event Control API

API For working with built-in websocket.

GET /event/list

Get the list of events that can be subscribed to using the websocket API.

Response

200 - Websocket API events list.

The response is JSON.

Name	Type	Description
events	array	List of events that can be subscribed to using the websocket API.
events[i]	string	

System Control API

API for controlling the System Modes on Blackmagic Design products.

GET /system

Get device system information.

Response

200 - System summary.

The response is JSON.

The properties will be populated only with the values that are supported/implemented by the device in use.

Name	Type	Description
codecFormat	object	Codec format configuration.
codecFormat.codec	string	Codec serialised as string.
codecFormat.container	string	Multimedia container format.
videoFormat	object	Video format configuration.
videoFormat.name (required)	string	Video format serialised as a string.
videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
videoFormat.height	number	Height dimension of video format.
videoFormat.width	number	Width dimension of video format.
videoFormat.interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/product

Get device product information.

Response

200 - Device product information.

The response is JSON.

Product information.

Name	Type	Description
deviceName	string	Name of device as displayed in Setup.
productName	string	Device's product name.
softwareVersion	string	Software version running on device.

501 - This functionality is not implemented for the device in use.

GET /system/supportedCodecFormats

Get the list of supported codecs.

Response

200 - List of supported codec formats.

The response is JSON.

Name	Type	Description
codecs	array	
codecs[i]	object	Codec format configuration.
codecs[i].codec	string	Codec serialised as string.
codecs[i].container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

GET /system/codecFormat

Get the currently selected codec.

Response

200 - Current codec format.

The response is JSON.

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

PUT /system/codecFormat

Update the system codec.

Parameters

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

Response

204 - The codec updated successfully.

400 - The specified codec format is unsupported.

501 - This functionality is not implemented for the device in use.

GET /system/videoFormat

Get the currently selected video format.

Response

200 - Current system video format.

The response is JSON.

Video format configuration.

Name	Type	Description
name (required)	string	Video format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height	number	Height dimension of video format.
width	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

PUT /system/videoFormat

Set the system video format.

Parameters

This parameter can be one of the following types:

Name	Type	Description
name (required)	string	Video format serialised as a string.

Name	Type	Description
frameRate (required)	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height (required)	number	Height dimension of video format.
width (required)	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

Response

204 - The video format updated successfully.

400 - Invalid request.

409 - Operation unsupported in the current state.

501 - This functionality is not implemented for the device in use.

GET /system/supportedVideoFormats

Get the list of supported video formats for the current system state.

Response

200 - List of supported video formats.

The response is JSON.

List of supported video formats.

Name	Type	Description
formats	array	List of video formats.
formats[i]	object	Video format configuration.
formats[i].name (required)	string	Video format serialised as a string.
formats[i].frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
formats[i].height	number	Height dimension of video format.
formats[i].width	number	Width dimension of video format.
formats[i].interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/supportedFormats

Get supported formats.

Response

200 - List of supported formats.

The response is JSON.

Name	Type	Description
supportedFormats	array	
supportedFormats[i]	object	
supportedFormats[i].codecs	array	
supportedFormats[i].codecs[i]	string	
supportedFormats[i].frameRates	array	
supportedFormats[i].frameRates[i]	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
supportedFormats[i]. maxOffSpeedFrameRate	number	
supportedFormats[i]. minOffSpeedFrameRate	number	
supportedFormats[i]. recordResolution	object	
supportedFormats[i]. recordResolution.height	number	Height of the resolution.
supportedFormats[i]. recordResolution.width	number	Width of the resolution.
supportedFormats[i]. sensorResolution	object	
supportedFormats[i]. sensorResolution.height	number	Height of the resolution.
supportedFormats[i]. sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

GET /system/format

Get current format.

Response

200 - Current format.

The response is JSON.

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

PUT /system/format

Set the format.

Parameters

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

Response

204 - System format updated.

501 - This functionality is not implemented for the device in use.

Transport Control API

API for controlling Transport on Blackmagic Design products.

GET /transports/0

Get device's basic transport status.

Response

200 - Transport status.

The response is JSON.

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

PUT /transports/0

Set device's basic transport status.

Parameters

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, Output.

Response

204 - Transport mode was set.

400 - Failed to set transport mode.

GET /transports/0/stop

Determine if transport is stopped.

Response

200 - Transport stop response.

The response is JSON.

Name	Type	Description
	boolean	True when transport mode is InputPreview or when in Output mode and speed is 0.

PUT /transports/0/stop

Stop transport. Deprecated, use POST /transports/0/stop instead.

Response

204 - Transport stopped.

POST /transports/0/stop

Stop transport.

Response

204 - Transport stopped.

GET /transports/0/play

Determine if transport is playing.

Response

200 - Transport play response.

The response is JSON.

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

PUT /transports/0/play

Start playing on transport. Deprecated, use POST /transports/0/play instead.

Response

204 - Transport playing.

400 - Failed to set transport to play.

POST /transports/0/play

Start playing on transport.

Response

204 - Transport playing.

400 - Failed to set transport to play.

GET /transports/0/playback

Get playback state.

Response

200 - Transport playback state.

The response is JSON.

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

PUT /transports/0/playback

Set playback state.

Parameters

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

Response

204 - Updated transport playback state.

400 - Failed to set transport playback state.

GET /transports/0/record

Get record state.

Response

200 - Recording state.

The response is JSON.

Name	Type	Description
recording	boolean	If true, transport is in InputRecord mode.

PUT /transports/0/record

Set record state. Deprecated, use POST /transports/0/record instead.

Parameters

Name	Type	Description
recording (required)	boolean	If true, starts a recording, otherwise stops.
clipName	string	Optional, sets the requested clip name to record to, when “recording” attribute is set to true.

Response

204 - Recording state updated.

400 - Failed to update recording state.

POST /transports/0/record

Start recording.

Parameters

Name	Type	Description
clipName	string	Optional, provides a specific name of clip to record to.

Response

204 - Recording started.

400 - Failed to start recording.

GET /transports/0/clipIndex

Get the clip index of the currently playing clip on the timeline.

Response

200 - Clip index response.

The response is JSON.

Name	Type	Description
clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

GET /transports/0/timecode

Get device timecode.

Response

200 - Timecode response.

The response is JSON.

Name	Type	Description
display	string	The display timecode serialised as a string.
timeline	string	The timeline timecode serialised as a string.

GET /transports/0/timecode/source

Get timecode source selected on device.

Response

200 - Timecode source response.

The response is JSON.

Name	Type	Description
timecode	string	Possible values are: Timeline, Clip.

Timeline Control API

API for controlling playback timeline.

GET /timelines/0

Get the playback timeline.

Response

200 - Playback timeline.

The response is JSON.

Name	Type	Description
clips	array	
clips[i]	object	Timeline clip.
clips[i].clipUniqueId (required)	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).
clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

404 - No timeline / disk available.

DELETE /timelines/0

Clear the current playback timeline. Deprecated, prefer to use POST /timelines/0/clear

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

POST /timelines/0

Add a clip to the timeline.

Parameters

This parameter can be one of the following types:

Add multiple media clips to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips[i].clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips[i].clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips[i].frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add multiple media clips to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add a single clip to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips.clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips.clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips.frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add a single clip to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the timeline as specified.

501 - The operation is not supported on this device.

POST /timelines/0/add

Add a clip to the end of the timeline. Deprecated, use POST /timelines/0 to add clips within the timeline.

Parameters

This parameter can be one of the following types:

Add one clip to the end of the timeline.

Name	Type	Description
clips	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add many clips to the end of the timeline.

Name	Type	Description
clips	array	List of clipUniqueIds of clips to add to end of timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the end of the timeline.

501 - The operation is not supported on this device.

POST /timelines/0/clear

Clear the playback timeline.

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

DELETE /timelines/0/clips/timelineClipIndex

Remove the specified clip from the timeline.

Parameters

Name	Type	Description
timelineClipIndex (required)	integer	The (0-based) timeline clip index of the clip to remove from the timeline.

Response

204 - The specified clip was removed from the timeline.

501 - The operation is not supported on this device.

Media Control API

API for controlling media devices in Blackmagic Design products.

GET /media/workingset

Get the list of media devices currently in the working set.

Response

200 - The list of media devices in the working set.

The response is JSON.

Name	Type	Description
size	integer	The fixed size of this device's working set.
workingset	array	The device's working set.
workingset[i]	object | null	Device within the working set. null if no device is present within the given working set slot.
workingset[i].volume	string	Volume name.
workingset[i].deviceName	string	Internal device name of this media device.
workingset[i].remainingRecordTime	integer	Remaining record time using current codec and video format in seconds.
workingset[i].totalSpace	integer	Total space on media device in bytes.
workingset[i].remainingSpace	integer	Remaining space on media device in bytes.
workingset[i].clipCount	integer	Number of clips currently on the device.

GET /media/active

Get the currently active media device.

Response

200 - The current active media device.

The response is JSON.

The active media device, or null if there is no active media.

Name	Type	Description
workingsetIndex	integer	Working set index of the active media device.
deviceName	string	Device name of media device.

204 - No media is currently active.

PUT /media/active

Set the currently active media device.

Parameters

Name	Type	Description
workingsetIndex	integer	Working set index of the media to make active.

Response

204 - The active media device was set successfully.

400 - Setting the currently active media device is not possible in the current state.

GET /media/devices/doformatSupportedFilesystems

Get the list of filesystems available to format a media device.

Response

200 - The list of filesystems permitted for formatting.

The response is JSON.

Name	Type	Description
	array	List of filesystems permitted for formatting media.
[i]	string	Filesystem serialised as string.

GET /media/devices/deviceName

Get information about a requested device.

Parameters

Name	Type	Description
deviceName (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Information about the requested device.

The response is JSON.

Media device state.

Name	Type	Description
state	string	The current state of the media device. Possible values are: None, Scanning, Mounted, Uninitialised, Formatting, RaidComponent.

400 - Invalid device name.

404 - Device not found.

GET /media/devices/deviceName/doformat

Get a format key, used to format the device with a PUT request.

Parameters

Name	Type	Description
deviceName (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Format prepared.

The response is JSON.

Name	Type	Description
deviceName	string	Device name of media device to format.
key	string	The key required to format this device, provide to PUT /media/devices/**deviceName**/doformat to perform format of media device.

400 - Cannot format the device.

404 - Device not found.

PUT /media/devices/deviceName/doformat

Perform a format of the specified media device.

Parameters

Name	Type	Description
deviceName (required)	string	Device name of the media device. Retrieved by "deviceName" member of GET /media/workingset or GET /media/active.

Name	Type	Description
key	string	The key used to format this device, retrieved from prepare format media request GET /media/devices/**deviceName**/doformat. Format key provided cannot be reused after successful format.
filesystem	string	Filesystem to format to. Supported filesystems can be retrieved with GET /media/devices/doFormatSupportedFilesystems.
volume	string	Volume name to set for the disk after format.

Response

204 - Format successful.

400 - Cannot format the device, invalid filesystem or key.

404 - Device not found.

Slate Control API

API to manage digital slate data.

GET /slates/nextClip

Retrieve the digital slate for the next clip.

Response

200 - Returns the slate data for the next clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	

Name	Type	Description
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Slate data is not available.

PUT /slates/nextClip

Update the slate data for the next clip.

Parameters

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

POST /slates/nextClip/resetProjectData

Reset the project data for the next clip's slate.

Response

200 - Project data reset successfully.

POST /slates/clips/deviceName/path/resetProjectData

Reset the project data for the next clip's slate.

Parameters

Name	Type	Description
deviceName (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
path (required)	string	Path to the clip.

Response

200 - Project data reset successfully.

POST /slates/nextClip/resetLensData

Reset the lens data for the next clip's slate.

Response

200 - Lens data reset successfully.

POST /slates/clips/deviceName/path/resetLensData

Reset the lens data for the next clip's slate.

Parameters

Name	Type	Description
deviceName (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
path (required)	string	Path to the clip.

Response

200 - Lens data reset successfully.

GET /slates/clips/deviceName/path

Retrieve slate data for a specific clip.

Parameters

Name	Type	Description
deviceName (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
path (required)	string	Path to the clip.

Response

200 - Returns the slate data for the specified clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

404 - Clip not found.

PUT /slates/clips/deviceName/path

Update the slate data for a specific clip.

Parameters

Name	Type	Description
deviceName (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
path (required)	string	Path to the clip.

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.

Name	Type	Description
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

Preset Control API

API For controlling the presets on Blackmagic Design products

GET /presets

Get the list of the presets on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presets	array	List of the presets on the camera (.cset files)
presets[i]	string	

POST /presets

Send a preset file to the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presetAdded	string	Name of the preset uploaded (without .cset extension)

400 - Bad request - missing Content-Disposition header or filename

GET /presets/active

Get the currently active preset on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

PUT /presets/active

Set the active preset on the camera

Parameters

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

Response

204 - No Content

404 - Preset file not found

PARAMETERS /presets/presetName

GET /presets/presetName

Download the preset file

Response

200 - OK

The response is a binary file.

404 - File does not exist

PUT /presets/presetName

Save current camera state as a preset

Response

204 - No Content

DELETE /presets/presetName

Delete a preset from the camera

Response

204 - No Content

404 - Preset file not found

Audio Control API

API For controlling audio on Blackmagic Design Cameras

GET /audio/channels

Get the total number of audio channels available

Response

200 - Returns the total number of channels

The response is JSON.

Name	Type	Description
channels	integer	Total number of audio channels available

GET /audio/supportedInputs

Get the list of supported audio inputs

Response

200 - List of all supported audio inputs

The response is JSON.

Name	Type	Description
	array	
[i]	string	A supported audio input

GET /audio/channel/channelIndex/input

Get the audio input (source and type) for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently selected input

The response is JSON.

Name	Type	Description
input	string	Audio input source and type

404 - Channel does not exist

PUT /audio/channel/channelIndex/input

Set the audio input for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
input	string	Audio input source and type

Response

204 - No Content

400 - Invalid audio input

404 - Channel does not exist

GET /audio/channel/channelIndex/input/description

Get the description of the current input of the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Description of the current input of the selected channel

The response is JSON.

Name	Type	Description
description	object	
description.gainRange	object	
description.gainRange.Min	number	The minimum gain value in dB
description.gainRange.Max	number	The maximum gain value in dB
description.capabilities	object	
description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter
description.capabilities.Padding	object	
description.capabilities.Padding. available	boolean	Input supports setting of padding
description.capabilities.Padding. forced	boolean	Padding is forced to be set for the input
description.capabilities.Padding. value	number	Value of the padding in dB

404 - Channel does not exist

GET /audio/channel/channelIndex/supportedInputs

Get the list of supported inputs and their availability to switch to for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its supported inputs are being queried. (Channels index from 0)

Response

200 - The list of supported inputs

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

404 - Channel does not exist

GET /audio/channel/channelIndex/level

Get the audio input level for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set level for the selected channel

The response is JSON.

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

404 - Channel does not exist

PUT /audio/channel/channelIndex/level

Set the audio input level for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

Response

204 - No Content

400 - Invalid input or value out of range

404 - Channel does not exist

GET /audio/channel/channelIndex/phantomPower

Get the audio input phantom power status for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set phantom power for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Phantom power enabled state

404 - Channel does not exist

PUT /audio/channel/channelIndex/phantomPower

Set the audio phantom power for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Phantom power enabled state

Response

204 - No Content

400 - Phantom power is not supported for this input

404 - Channel does not exist

GET /audio/channel/channelIndex/padding

Get the audio input padding status for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set padding for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Padding enabled state

404 - Channel does not exist

PUT /audio/channel/channelIndex/padding

Set the audio input padding for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Padding enabled state

Response

204 - No Content

400 - Padding is not supported or is forced for this input

404 - Channel does not exist

GET /audio/channel/channelIndex/lowCutFilter

Get the audio input low cut filter status for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set low cut filter for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Low cut filter enabled state

404 - Channel does not exist

PUT /audio/channel/channelIndex/lowCutFilter

Set the audio input low cut filter for the selected channel

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Low cut filter enabled state

Response

204 - No Content

400 - Low cut filter is not supported for this input

404 - Channel does not exist

GET /audio/channel/channelIndex/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

Parameters

Name	Type	Description
channelIndex (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set availability for the selected channel

The response is JSON.

Name	Type	Description
available	boolean	Whether the input is currently available

404 - Channel does not exist

Lens Control API

API For controlling the lens on Blackmagic Design products

GET /lens/iris

Get lens' aperture

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number

PUT /lens/iris

Set lens' aperture

Parameters

Name	Type	Description
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number
adjustmentStep	integer	Signed value for relative aperture adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens iris is not controllable or is controlled by auto exposure

GET /lens/zoom

Get lens' zoom

Response

200 - OK

The response is JSON.

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value

PUT /lens/zoom

Set lens' zoom

Parameters

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value
adjustmentFocalLength	integer	Signed value for relative focal length adjustment
adjustmentNormalised	number	Signed normalized value for relative zoom adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens zoom is not controllable

GET /lens/focus

Get lens' focus

Response

200 - OK

The response is JSON.

Name	Type	Description
normalised	number	Normalised value

PUT /lens/focus

Set lens' focus

Parameters

Name	Type	Description
normalised	number	Normalised value
focusDistance	integer	Focus distance value

Response

204 - No Content

400 - Bad Request if out of range value is provided

PUT /lens/focus/doAutoFocus

Perform auto focus

Parameters

Name	Type	Description
position	object	
position.x	number	Normalized x coordinate for autofocus ROI
position.y	number	Normalized y coordinate for autofocus ROI

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens focus is not controllable

GET /lens/opticalImageStabilization

Get optical image stabilization status

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Whether optical image stabilization is enabled

501 - Not Implemented if optical image stabilization is not supported on this product

PUT /lens/opticalImageStabilization

Enable or disable optical image stabilization

Parameters

Name	Type	Description
enabled	boolean	Enable or disable optical image stabilization

Response

204 - No Content

501 - Not Implemented if optical image stabilization is not supported on this product

GET /lens/iris/description

Get detailed description of lens' iris capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the iris can be controlled
apertureStop	object	
apertureStop.min	number	Minimum aperture stop
apertureStop.max	number	Maximum aperture stop

GET /lens/zoom/description

Get detailed description of lens' zoom capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the zoom can be controlled
focalLength	object	
focalLength.adjustable	boolean	If focal length is adjustable
focalLength.min	integer	Minimum focal length
focalLength.max	integer	Maximum focal length

GET /lens/focus/description

Get detailed description of lens' focus capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the focus can be controlled
focusDistance	object	
focusDistance.adjustable	boolean	If focus distance is adjustable
focusDistance.min	number	Minimum focus distance
focusDistance.max	number	Maximum focus distance

Video Control API

API For controlling the video on Blackmagic Design products

GET /video/iso

Get current ISO

Response

200 - OK

The response is JSON.

Name	Type	Description
iso	integer	Current ISO value

PUT /video/iso

Set current ISO

Parameters

Name	Type	Description
iso	integer	ISO value to set

Response

204 - No Content

403 - ISO cannot be changed in the current state

GET /video/supportedISOs

Get the list of supported ISO settings

Response

200 - List of supported ISO values

The response is JSON.

Name	Type	Description
supportedISOs	array	Array of supported ISO values
supportedISOs[i]	integer	

GET /video/gain

Get current gain value in decibels

Response

200 - OK

The response is JSON.

Name	Type	Description
gain	integer	Current gain value in decibels

PUT /video/gain

Set current gain value

Parameters

Name	Type	Description
gain	integer	Gain value in decibels to set

Response

204 - No Content

403 - Gain cannot be changed in the current state

GET /video/supportedGains

Get the list of supported gain settings in decibels

Response

200 - List of supported gain values in decibels

The response is JSON.

Name	Type	Description
supportedGains	array	Array of supported gain values in decibels
supportedGains[i]	integer	

GET /video/whiteBalance

Get current white balance

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	integer	Current white balance

PUT /video/whiteBalance

Set current white balance

Parameters

Name	Type	Description
whiteBalance	integer	White balance to set

Response

204 - No Content

400 - Invalid white balance temperature

GET /video/whiteBalance/description

Get white balance range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	object	
whiteBalance.min	integer	Minimum color temperature
whiteBalance.max	integer	Maximum color temperature

PUT /video/whiteBalance/doAuto

Set current white balance automatically

Response

204 - No Content

GET /video/whiteBalanceTint

Get white balance tint

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	integer	Current white balance tint

PUT /video/whiteBalanceTint

Set white balance tint

Parameters

Name	Type	Description
whiteBalanceTint	integer	White balance tint to set

Response

204 - No Content

400 - Invalid white balance tint

GET /video/whiteBalanceTint/description

Get white balance tint range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	object	
whiteBalanceTint.min	integer	Minimum white balance tint
whiteBalanceTint.max	integer	Maximum white balance tint

GET /video/ndFilter

Get ND filter stop

Response

200 - OK

The response is JSON.

Name	Type	Description
stop	number	Current filter power (fStop)

501 - Not implemented for this device

PUT /video/ndFilter

Set ND filter stop

Parameters

Name	Type	Description
stop	number	Filter power (fStop) to set

Response

204 - No Content

400 - Invalid ND filter stop

501 - Not implemented for this device

GET /video/supportedNDFilters

Get the list of available ND filter stops

Response

200 - List of available ND filter stops

The response is JSON.

Name	Type	Description
supportedStops	array	Array of available ND filter stops
supportedStops[i]	number	

501 - Not implemented for this device

GET /video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

Response

200 - List of supported display modes

The response is JSON.

Name	Type	Description
supportedDisplayModes	array	Array of supported display modes
supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

GET /video/ndFilter/displayMode

Get ND filter display mode on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

PUT /video/ndFilter/displayMode

Set ND filter display mode on the camera

Parameters

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

Response

204 - No Content

400 - Invalid display mode for ND filter

501 - Not implemented for this device

GET /video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

Response

200 - Indicates if ND filter is selectable

The response is JSON.

Name	Type	Description
selectable	boolean	True if ND filter adjustments are selectable via a slider

501 - Not implemented for this device

GET /video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

PUT /video/shutter

Set current shutter

Parameters

Name	Type	Description
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

Response

204 - No Content

GET /video/shutter/measurement

Get the current shutter measurement mode

Response

200 - OK

The response is JSON.

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

PUT /video/shutter/measurement

Set the shutter measurement mode

Parameters

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

Response

204 - No Content

400 - Invalid shutter measurement

GET /video/supportedShutters

Get supported shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of supported shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of supported shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of flicker-free shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of flicker-free shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/autoExposure

Get current auto exposure mode

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

PUT /video/autoExposure

Set auto exposure

Parameters

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

Response

204 - No Content

400 - Failed to set auto exposure mode

GET /video/detailSharpening

Get the current state of detail sharpening

Response

200 - Current detail sharpening state

The response is JSON.

Name	Type	Description
enabled	boolean	Whether detail sharpening is enabled

501 - Not implemented for this device

PUT /video/detailSharpening

Enable or disable detail sharpening

Parameters

Name	Type	Description
enabled	boolean	Enable or disable detail sharpening

Response

204 - Detail sharpening state updated

501 - Not implemented for this device

GET /video/detailSharpeningLevel

Get the current detail sharpening level

Response

200 - Current detail sharpening level

The response is JSON.

Name	Type	Description
level	string	Current detail sharpening level Possible values are: Low, Medium, High.

501 - Not implemented for this device

PUT /video/detailSharpeningLevel

Set the detail sharpening level

Parameters

Name	Type	Description
level	string	Desired level of detail sharpening Possible values are: Low, Medium, High.

Response

204 - Detail sharpening level updated

400 - Invalid detail sharpening level

501 - Not implemented for this device

Camera Control API

API For controlling the Camera specific features on Blackmagic Design products

GET /camera/colorBars

Get the status of color bars display

Response

200 - Returns the current status of color bars

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the color bars are currently enabled

PUT /camera/colorBars

Set the status of color bars display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the color bars

Response

204 - Color bars status updated successfully

GET /camera/programFeedDisplay

Get the status of program feed display

Response

200 - Returns the current status of program feed display

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the program feed display is currently enabled

PUT /camera/programFeedDisplay

Set the status of program feed display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the program feed display

Response

204 - Program feed display status updated successfully

GET /camera/tallyStatus

Get the tally status of the camera

Response

200 - Returns the current tally status of the camera

The response is JSON.

Name	Type	Description
status	string	Current tally status of the camera Possible values are: None, Preview, Program.

GET /camera/power

Get the power status of the camera

Response

200 - Returns the current power status

The response is JSON.

Name	Type	Description
source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
batteries	array	List of batteries currently connected to the camera
batteries[i]	object	
batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
batteries[i].chargeRemainingPercent	integer	Remaining battery charge percentage
batteries[i].statusFlags	array	List of battery status flags
batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

GET /camera/power/displayMode

Get the power display mode of the camera

Response

200 - Returns the current power display mode

The response is JSON.

Name	Type	Description
mode	string	Current power display mode Possible values are: Percentage, Voltage.

PUT /camera/power/displayMode

Set the power display mode of the camera

Parameters

Name	Type	Description
mode	string	Power display mode to set Possible values are: Percentage, Voltage.

Response

204 - Power display mode updated successfully

400 - Invalid power display mode

GET /camera/timingReferenceLock

Get the timing reference lock status

Response

200 - Returns the timing reference lock status

The response is JSON.

Name	Type	Description
locked	boolean	Indicates if timing reference is locked

Immersive Control API

API for controlling immersive camera settings on Blackmagic Design cameras

GET /immersive/display/displayName/eye

Get the current eye view for a specific display

Parameters

Name	Type	Description
displayName (required)	string	The display name to query (from /monitoring/display endpoint)

Response

200 - OK

The response is JSON.

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

400 - Invalid display name format

404 - Display not found

422 - Failed to get eye view

PUT /immersive/display/displayName/eye

Set the eye view for a specific display

Parameters

Name	Type	Description
displayName (required)	string	The display name to control (from /monitoring/display endpoint)

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

Response

204 - No Content

400 - Invalid input or display name format

404 - Display not found

422 - Failed to set eye view

Color Correction Control API

API For controlling the color correction on Blackmagic Design products based on DaVinci Resolve Color Corrector

GET /colorCorrection/lift

Get color correction lift

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

PUT /colorCorrection/lift

Set color correction lift

Parameters

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gamma

Get color correction gamma

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

PUT /colorCorrection/gamma

Set color correction gamma

Parameters

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gain

Get color correction gain

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

PUT /colorCorrection/gain

Set color correction gain

Parameters

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/offset

Get color correction offset

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

PUT /colorCorrection/offset

Set color correction offset

Parameters

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/contrast

Get color correction contrast

Response

200 - OK

The response is JSON.

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/contrast

Set color correction contrast

Parameters

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/color

Get color correction color properties

Response

200 - OK

The response is JSON.

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/color

Set color correction color properties

Parameters

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/lumaContribution

Get color correction luma contribution

Response

200 - OK

The response is JSON.

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

PUT /colorCorrection/lumaContribution

Set color correction luma contribution

Parameters

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

Response

204 - No Content

Notification websocket - 1.0.0

Service that notifies subscribers of device state changes.

messages

Subscribe (The messages from the server/device)

Websocket Opened Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: websocketOpened.
.type	string	Possible values are: event.

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/**displayName**/cleanFeed, /monitoring/**displayName**/displayLUT, /monitoring/**displayName**/zebra, /monitoring/**displayName**/focusAssist, /monitoring/**displayName**/frameGuide, /monitoring/**displayName**/frameGrids, /monitoring/**displayName**/safeArea, /monitoring/**displayName**/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/**channelIndex**/input, /audio/channel/**channelIndex**/supportedInputs, /audio/channel/**channelIndex**/level, /audio/channel/**channelIndex**/phantomPower, /audio/channel/**channelIndex**/padding, /audio/channel/**channelIndex**/lowCutFilter, /audio/channel/**channelIndex**/available, /audio/channel/**channelIndex**/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*

Name	Type	Description
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/**displayName**/cleanFeed, /monitoring/**displayName**/displayLUT, /monitoring/**displayName**/zebra, /monitoring/**displayName**/focusAssist, /monitoring/**displayName**/frameGuide, /monitoring/**displayName**/frameGrids, /monitoring/**displayName**/safeArea, /monitoring/**displayName**/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/**channelIndex**/input, /audio/channel/**channelIndex**/supportedInputs, /audio/channel/**channelIndex**/level, /audio/channel/**channelIndex**/phantomPower, /audio/channel/**channelIndex**/padding, /audio/channel/**channelIndex**/lowCutFilter, /audio/channel/**channelIndex**/available, /audio/channel/**channelIndex**/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Event Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: propertyValueChanged.

Name	Type	Description
.data.property	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/**displayName**/cleanFeed, /monitoring/**displayName**/displayLUT, /monitoring/**displayName**/zebra, /monitoring/**displayName**/focusAssist, /monitoring/**displayName**/frameGuide, /monitoring/**displayName**/frameGrids, /monitoring/**displayName**/safeArea, /monitoring/**displayName**/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/**channelIndex**/input, /audio/channel/**channelIndex**/supportedInputs, /audio/channel/**channelIndex**/level, /audio/channel/**channelIndex**/phantomPower, /audio/channel/**channelIndex**/padding, /audio/channel/**channelIndex**/lowCutFilter, /audio/channel/**channelIndex**/available, /audio/channel/**channelIndex**/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.value	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.type	string	Possible values are: event.

Publish (The messages that user can send to the server/device)

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	

Name	Type	Description
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/**displayName**/cleanFeed, /monitoring/**displayName**/displayLUT, /monitoring/**displayName**/zebra, /monitoring/**displayName**/focusAssist, /monitoring/**displayName**/frameGuide, /monitoring/**displayName**/frameGrids, /monitoring/**displayName**/safeArea, /monitoring/**displayName**/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/**channelIndex**/input, /audio/channel/**channelIndex**/supportedInputs, /audio/channel/**channelIndex**/level, /audio/channel/**channelIndex**/phantomPower, /audio/channel/**channelIndex**/padding, /audio/channel/**channelIndex**/lowCutFilter, /audio/channel/**channelIndex**/available, /audio/channel/**channelIndex**/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	

Name	Type	Description
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/**displayName**/cleanFeed, /monitoring/**displayName**/displayLUT, /monitoring/**displayName**/zebra, /monitoring/**displayName**/focusAssist, /monitoring/**displayName**/frameGuide, /monitoring/**displayName**/frameGrids, /monitoring/**displayName**/safeArea, /monitoring/**displayName**/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/**channelIndex**/input, /audio/channel/**channelIndex**/supportedInputs, /audio/channel/**channelIndex**/level, /audio/channel/**channelIndex**/phantomPower, /audio/channel/**channelIndex**/padding, /audio/channel/**channelIndex**/lowCutFilter, /audio/channel/**channelIndex**/available, /audio/channel/**channelIndex**/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Device Properties

/media/workingset

The value JSON returned via the eventResponse when the `/media/workingset` property changes on the device:

Name	Type	Description
.size	integer	The fixed size of this device's working set.
.workingset	array	Array of devices within the working set. null if no device is present within the given working set slot.
.workingset[i]		

/media/active

The value JSON returned via the eventResponse when the `/media/active` property changes on the device:

Name	Type	Description
.workingsetIndex	integer	Working set index of the active media device.
.deviceName	string	Internal device name of this media device.

/system

The value JSON returned via the eventResponse when the `/system` property changes on the device:

Name	Type	Description
.codecFormat	object	Codec format configuration.
.codecFormat.codec	string	Codec format serialised as a string.
.codecFormat.container	string	Multimedia container format.
.videoFormat	object	Currently selected video format.
.videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.videoFormat.height	number	Height dimension of video format.
.videoFormat.width	number	Width dimension of video format.
.videoFormat.interlaced	boolean	Is the display format interlaced?.
.videoFormat.name	string	Video format serialised as a string.

/system/codecFormat

Codec format configuration.

The value JSON returned via the eventResponse when the `/system/codecFormat` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.container	string	Multimedia container format.

/system/videoFormat

Currently selected video format.

The value JSON returned via the eventResponse when the `/system/videoFormat` property changes on the device:

Name	Type	Description
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.height	number	Height dimension of video format.
.width	number	Width dimension of video format.
.interlaced	boolean	Is the display format interlaced?.
.name	string	Video format serialised as a string.

/system/format

The value JSON returned via the eventResponse when the `/system/format` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.maxOffSpeedFrameRate	number	
.minOffSpeedFrameRate	number	
.offSpeedEnabled	boolean	
.offSpeedFrameRate	number	
.recordResolution	object	
.recordResolution.height	number	Height of the resolution.
.recordResolution.width	number	Width of the resolution.
.sensorResolution	object	
.sensorResolution.height	number	Height of the resolution.
.sensorResolution.width	number	Width of the resolution.

/system/supportedFormats

The value JSON returned via the eventResponse when the `/system/supportedFormats` property changes on the device:

Name	Type	Description
.supportedFormats	array	
.supportedFormats[i]	object	
.supportedFormats[i].codecs	array	
.supportedFormats[i].codecs[i]	string	
.supportedFormats[i].frameRates	array	
.supportedFormats[i].frameRates[i]	string	Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.supportedFormats[i].maxOffSpeedFrameRate	number	
.supportedFormats[i].minOffSpeedFrameRate	number	
.supportedFormats[i].recordResolution	object	
.supportedFormats[i].recordResolution.height	number	Height of the resolution.
.supportedFormats[i].recordResolution.width	number	Width of the resolution.
.supportedFormats[i].sensorResolution	object	
.supportedFormats[i].sensorResolution.height	number	Height of the resolution.
.supportedFormats[i].sensorResolution.width	number	Width of the resolution.

/timelines/0

The value JSON returned via the eventResponse when the `/timelines/0` property changes on the device:

Name	Type	Description
.clips	array	
.clips[i]	object	Timeline clip.
.clips[i].clipUniqueId	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
.clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
.clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
.clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
.clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).

Name	Type	Description
.clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
.clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

/transports/0

The value JSON returned via the eventResponse when the `/transports/0` property changes on the device:

Name	Type	Description
.mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

/transports/0/stop

true when transport mode is InputPreview or when in Output mode and speed is 0.

The value JSON returned via the eventResponse when the `/transports/0/stop` property changes on the device:

Name	Type	Description
	boolean	true when transport mode is InputPreview or when in Output mode and speed is 0.

/transports/0/play

True when transport is in Output mode and speed is non-zero.

The value JSON returned via the eventResponse when the /transports/0/play property changes on the device:

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

/transports/0/playback

The value JSON returned via the eventResponse when the `/transports/0/playback` property changes on the device:

Name	Type	Description
.type	string	Possible values are: Play, Jog, Shuttle, Var.
.loop	boolean	When true playback loops from the end of the timeline to the beginning of the timeline.
.singleClip	boolean	When true playback loops from the end of the current clip to the beginning of the current clip.
.speed	number	Playback speed, 1.0 for normal forward playback
.position	integer	Playback position on the timeline in units of video frames.

/transports/0/record

The value JSON returned via the eventResponse when the `/transports/0/record` property changes on the device:

Name	Type	Description
.recording	boolean	Is transport in Input Record mode.

/transports/0/timecode

The value JSON returned via the eventResponse when the `/transports/0/timecode` property changes on the device:

Name	Type	Description
.display	string	The display timecode serialised as a string.
.timeline	string	The timeline timecode serialised as a string.

/transports/0/timecode/source

The value JSON returned via the eventResponse when the `/transports/0/timecode/source` property changes on the device:

Name	Type	Description
.timecode	string	Possible values are: Timeline, Clip.

/transports/0/clipIndex

The value JSON returned via the eventResponse when the `/transports/0/clipIndex` property changes on the device:

Name	Type	Description
.clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

/slates/nextClip

The value JSON returned via the eventResponse when the `/slates/nextClip` property changes on the device:

Name	Type	Description
.clip	object	
.clip.clipName	string	
.clip.reel	integer	
.clip.scene	string	
.clip.sceneLocation	string	Possible values are: Interior, Exterior.
.clip.sceneTime	string	Possible values are: Day, Night.
.clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
.clip.take	integer	
.clip.takeType	string	Possible values are: None, PU, VFX, SER.
.clip.goodTake	boolean	
.lens	object	
.lens.lensType	string	
.lens.iris	string	

Name	Type	Description
.lens.focalLength	string	
.lens.distance	string	
.lens.filter	string	
.project	object	
.project.projectName	string	
.project.director	string	
.project.camera	string	
.project.cameraOperator	string	

/monitoring/displayName/cleanFeed

The value JSON returned via the eventResponse when the `/monitoring/`**displayName**`/cleanFeed` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/displayName/displayLUT

The value JSON returned via the eventResponse when the `/monitoring/`**displayName**`/displayLUT` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/displayName/zebra

The value JSON returned via the eventResponse when the `/monitoring/`**displayName**`/zebra` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/displayName/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/`**displayName**`/focusAssist` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/displayName/frameGuide

The value JSON returned via the eventResponse when the `/monitoring/displayName/frameGuide` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/displayName/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/displayName/frameGrids` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/displayName/safeArea

The value JSON returned via the eventResponse when the `/monitoring/displayName/safeArea` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/displayName/falseColor

The value JSON returned via the eventResponse when the `/monitoring/displayName/falseColor` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/focusAssist` property changes on the device:

Name	Type	Description
.mode	string	Possible values are: Peak, ColoredLines.
.color	string	Possible values are: Red, Green, Blue, White, Black.
.intensity	integer	

/monitoring/frameGuideRatio

The value JSON returned via the eventResponse when the `/monitoring/frameGuideRatio` property changes on the device:

Name	Type	Description
.ratio	string	

/monitoring/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/frameGrids` property changes on the device:

Name	Type	Description
.frameGrids	array	
.frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

/monitoring/safeAreaPercent

The value JSON returned via the eventResponse when the `/monitoring/safeAreaPercent` property changes on the device:

Name	Type	Description
.percent	integer	Safe area coverage percentage.

/audio/channel/channelIndex/input

Get the audio input (source and type) for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/`**channelIndex**`/input` property changes on the device:

Name	Type	Description
.input	string	Audio input source and type

/audio/channel/channelIndex/supportedInputs

The value JSON returned via the eventResponse when the `/audio/channel/`**channelIndex**`/supportedInputs` property changes on the device:

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

/audio/channel/channelIndex/level

Get the audio input level for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/`**channelIndex**`/level` property changes on the device:

Name	Type	Description
.gain	number	Gain value in dB
.normalised	number	Normalised level value between 0.0 and 1.0

/audio/channel/channelIndex/phantomPower

Get the audio input phantom power status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/`**channelIndex**`/phantomPower` property changes on the device:

Name	Type	Description
.enabled	boolean	Phantom power enabled state

/audio/channel/channelIndex/padding

Get the audio input padding status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/`**channelIndex**`/padding` property changes on the device:

Name	Type	Description
.enabled	boolean	Padding enabled state

/audio/channel/channelIndex/lowCutFilter

Get the audio input low cut filter status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/`**channelIndex**`/lowCutFilter` property changes on the device:

Name	Type	Description
.enabled	boolean	Low cut filter enabled state

/audio/channel/channelIndex/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

The value JSON returned via the eventResponse when the `/audio/channel/`**channelIndex**`/available` property changes on the device:

Name	Type	Description
.available	boolean	Whether the input is currently available

/audio/channel/channelIndex/input/description

Description of the current input of the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/`**channelIndex**`/input/description` property changes on the device:

Name	Type	Description
.description	object	
.description.gainRange	object	
.description.gainRange.Min	number	The minimum gain value in dB
.description.gainRange.Max	number	The maximum gain value in dB
.description.capabilities	object	
.description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
.description.capabilities. LowCutFilter	boolean	Input supports setting of low cut filter

Name	Type	Description
.description.capabilities.Padding	object	
.description.capabilities.Padding.available	boolean	Input supports setting of padding
.description.capabilities.Padding.forced	boolean	Padding is forced to be set for the input
.description.capabilities.Padding.value	number	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.

/colorCorrection/lift

Get color correction lift

The value JSON returned via the eventResponse when the `/colorCorrection/lift` property changes on the device:

Name	Type	Description
.red	number	Red lift component
.green	number	Green lift component
.blue	number	Blue lift component
.luma	number	Luma lift component

/colorCorrection/gamma

Get color correction gamma

The value JSON returned via the eventResponse when the `/colorCorrection/gamma` property changes on the device:

Name	Type	Description
.red	number	Red gamma component
.green	number	Green gamma component
.blue	number	Blue gamma component
.luma	number	Luma gamma component

/colorCorrection/gain

Get color correction gain

The value JSON returned via the eventResponse when the `/colorCorrection/gain` property changes on the device:

Name	Type	Description
.red	number	Red gain component
.green	number	Green gain component
.blue	number	Blue gain component
.luma	number	Luma gain component

/colorCorrection/offset

Get color correction offset

The value JSON returned via the eventResponse when the `/colorCorrection/offset` property changes on the device:

Name	Type	Description
.red	number	Red offset component
.green	number	Green offset component
.blue	number	Blue offset component
.luma	number	Luma offset component

/colorCorrection/contrast

Get color correction contrast

The value JSON returned via the eventResponse when the `/colorCorrection/contrast` property changes on the device:

Name	Type	Description
.pivot	number	Contrast pivot point
.adjust	number	Contrast adjustment

/colorCorrection/color

Get color correction color properties

The value JSON returned via the eventResponse when the `/colorCorrection/color` property changes on the device:

Name	Type	Description
.hue	number	Color hue adjustment
.saturation	number	Color saturation adjustment

/colorCorrection/lumaContribution

Get color correction luma contribution

The value JSON returned via the eventResponse when the `/colorCorrection/lumaContribution` property changes on the device:

Name	Type	Description
.lumaContribution	number	Luma contribution value

/lens/iris

Get lens' aperture

The value JSON returned via the eventResponse when the `/lens/iris` property changes on the device:

Name	Type	Description
.continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
.apertureStop	number	Aperture stop value
.normalised	number	Normalised value
.apertureNumber	integer	Aperture number

/lens/iris/description

Get detailed description of lens' iris capabilities

The value JSON returned via the eventResponse when the `/lens/iris/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the iris can be controlled
.apertureStop	object	
.apertureStop.min	number	Minimum aperture stop
.apertureStop.max	number	Maximum aperture stop

/lens/focus

Get lens' focus

The value JSON returned via the eventResponse when the `/lens/focus` property changes on the device:

Name	Type	Description
.normalised	number	Normalised value

/lens/focus/description

Get detailed description of lens' focus capabilities

The value JSON returned via the eventResponse when the `/lens/focus/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the focus can be controlled
.focusDistance	object	
.focusDistance.adjustable	boolean	If focus distance is adjustable
.focusDistance.min	number	Minimum focus distance
.focusDistance.max	number	Maximum focus distance

/lens/zoom

Get lens' zoom

The value JSON returned via the eventResponse when the `/lens/zoom` property changes on the device:

Name	Type	Description
.focalLength	integer	Focal length in mm
.normalised	number	Normalised value

/lens/zoom/description

Get detailed description of lens' zoom capabilities

The value JSON returned via the eventResponse when the `/lens/zoom/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the zoom can be controlled
.focalLength	object	
.focalLength.adjustable	boolean	If focal length is adjustable
.focalLength.min	integer	Minimum focal length
.focalLength.max	integer	Maximum focal length

/presets

Get the list of the presets on the camera

The value JSON returned via the eventResponse when the `/presets` property changes on the device:

Name	Type	Description
.presets	array	List of the presets on the camera (.cset files)
.presets[i]	string	

/presets/active

Get the currently active preset on the camera

The value JSON returned via the eventResponse when the `/presets/active` property changes on the device:

Name	Type	Description
.preset	string	Name of the active preset (with .cset extension, or 'default')

/camera/colorBars

Get the status of color bars display

The value JSON returned via the eventResponse when the `/camera/colorBars` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the color bars are currently enabled

/camera/programFeedDisplay

Get the status of program feed display

The value JSON returned via the eventResponse when the `/camera/programFeedDisplay` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the program feed display is currently enabled

/camera/tallyStatus

Get the tally status of the camera

The value JSON returned via the eventResponse when the `/camera/tallyStatus` property changes on the device:

Name	Type	Description
.status	string	Current tally status of the camera Possible values are: None, Preview, Program.

/camera/power

Get the power status of the camera

The value JSON returned via the eventResponse when the `/camera/power` property changes on the device:

Name	Type	Description
.source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
.milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
.batteries	array	
.batteries[i]	object	
.batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
.batteries[i]. chargeRemainingPercent	integer	Remaining battery charge percentage
.batteries[i].statusFlags	array	List of battery status flags
.batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

/camera/power/displayMode

Get the power display mode of the camera

The value JSON returned via the eventResponse when the `/camera/power/displayMode` property changes on the device:

Name	Type	Description
.mode	string	Current power display mode Possible values are: Percentage, Voltage.

/camera/timingReferenceLock

Get the timing reference lock status

The value JSON returned via the eventResponse when the `/camera/timingReferenceLock` property changes on the device:

Name	Type	Description
.locked	boolean	Indicates if timing reference is locked

/video/iso

Get current ISO

The value JSON returned via the eventResponse when the `/video/iso` property changes on the device:

Name	Type	Description
.iso	integer	Current ISO value

/video/supportedISOs

Get the list of supported ISO settings

The value JSON returned via the eventResponse when the `/video/supportedISOs` property changes on the device:

Name	Type	Description
.supportedISOs	array	Array of supported ISO values
.supportedISOs[i]	integer	

/video/gain

Get current gain value in decibels

The value JSON returned via the eventResponse when the `/video/gain` property changes on the device:

Name	Type	Description
.gain	integer	Current gain value in decibels

/video/supportedGains

Get the list of supported gain settings in decibels

The value JSON returned via the eventResponse when the `/video/supportedGains` property changes on the device:

Name	Type	Description
.supportedGains	array	Array of supported gain values in decibels
.supportedGains[i]	integer	

/video/whiteBalance

Get current white balance

The value JSON returned via the eventResponse when the `/video/whiteBalance` property changes on the device:

Name	Type	Description
.whiteBalance	integer	Current white balance

/video/whiteBalance/description

Get white balance range

The value JSON returned via the eventResponse when the `/video/whiteBalance/description` property changes on the device:

Name	Type	Description
.whiteBalance	object	
.whiteBalance.min	integer	Minimum color temperature
.whiteBalance.max	integer	Maximum color temperature

/video/whiteBalanceTint

Get white balance tint

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint` property changes on the device:

Name	Type	Description
.whiteBalanceTint	integer	Current white balance tint

/video/whiteBalanceTint/description

Get white balance tint range

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint/description` property changes on the device:

Name	Type	Description
.whiteBalanceTint	object	
.whiteBalanceTint.min	integer	Minimum white balance tint
.whiteBalanceTint.max	integer	Maximum white balance tint

/video/ndFilter

Get ND filter stop

The value JSON returned via the eventResponse when the `/video/ndFilter` property changes on the device:

Name	Type	Description
.stop	number	Current filter power (fStop)

/video/supportedNDFilters

Get the list of available ND filter stops

The value JSON returned via the eventResponse when the `/video/supportedNDFilters` property changes on the device:

Name	Type	Description
.supportedStops	array	Array of available ND filter stops
.supportedStops[i]	number	

/video/ndFilter/displayMode

Get ND filter display mode on the camera

The value JSON returned via the eventResponse when the `/video/ndFilter/displayMode` property changes on the device:

Name	Type	Description
.displayMode	string	ND filter display mode Possible values are: Stop, Number, Fraction.

/video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

The value JSON returned via the eventResponse when the `/video/supportedNDFilterDisplayModes` property changes on the device:

Name	Type	Description
.supportedDisplayModes	array	Array of supported display modes
.supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

/video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

The value JSON returned via the eventResponse when the `/video/ndFilterSelectable` property changes on the device:

Name	Type	Description
.selectable	boolean	True if ND filter adjustments are selectable via a slider

/video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

The value JSON returned via the eventResponse when the `/video/shutter` property changes on the device:

Name	Type	Description
.continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
.shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
.shutterAngle	number	Shutter angle

/video/shutter/measurement

Get the current shutter measurement mode

The value JSON returned via the eventResponse when the `/video/shutter/measurement` property changes on the device:

Name	Type	Description
.measurement	string	Current shutter measurement mode Possible values are: ShutterAngle, ShutterSpeed.

/video/supportedShutters

Get supported shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/supportedShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of supported shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array of flicker-free shutter speeds
.shutterSpeeds[i]	integer	

/video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/flickerFreeShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of flicker-free shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array
.shutterSpeeds[i]	integer	

/video/autoExposure

Get current auto exposure mode

The value JSON returned via the eventResponse when the `/video/autoExposure` property changes on the device:

Name	Type	Description
.mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
.type	string	Comma-separated list of device types in the auto exposure stack

/video/detailSharpening

Get the current state of detail sharpening

The value JSON returned via the eventResponse when the `/video/detailSharpening` property changes on the device:

Name	Type	Description
.enabled	boolean	Whether detail sharpening is enabled

/video/detailSharpeningLevel

Get the current detail sharpening level

The value JSON returned via the eventResponse when the `/video/detailSharpeningLevel` property changes on the device:

Name	Type	Description
.level	string	Current detail sharpening level of supported shutter speeds Possible values are: Low, Medium, High.

Pomoc

Najszybszym sposobem uzyskania pomocy jest wejście na strony wsparcia technicznego online Blackmagic Design i sprawdzenie najnowszych materiałów pomocniczych dostępnych dla Twojej kamery.

Strony wsparcia technicznego online Blackmagic Design

Najnowszą instrukcję obsługi, oprogramowanie i noty informacyjne można znaleźć w centrum wsparcia technicznego na www.blackmagicdesign.com/pl/support.

Kontakt z obsługą techniczną Blackmagic Design

Jeśli nie możesz znaleźć potrzebnej pomocy w naszych materiałach, użyj przycisku **Wyślij do nas e-mail** na stronie wsparcia technicznego dla swojej kamery, aby wysłać zgłoszenie dotyczące pomocy technicznej. Alternatywnie kliknij na stronie przycisk **Znajdź lokalny zespół pomocy technicznej** i zadzwoń do najbliższego biura pomocy technicznej Blackmagic Design.

Sprawdzanie aktualnie zainstalowanej wersji oprogramowania

Aby sprawdzić, jaka wersja narzędzia konfiguracyjnego Blackmagic Camera Setup jest zainstalowana na Twoim komputerze, otwórz okno **About Blackmagic Camera Setup**.

- W systemie Mac otwórz program narzędziowy **Blackmagic Camera Setup** w folderze **Applications**. Wybierz **About Blackmagic Camera Setup** w menu aplikacji, aby wyświetlić numer wersji.
- W systemie Windows otwórz narzędzie konfiguracyjne **Blackmagic Camera Setup** z menu **Start** lub na ekranie startowym. Kliknij na **About Blackmagic Camera Setup** w menu pomocy, aby wyświetlić numer wersji.

Jak uzyskać najnowsze aktualizacje oprogramowania

Po sprawdzeniu wersji oprogramowania Blackmagic Camera Setup zainstalowanego na komputerze odwiedź centrum wsparcia technicznego Blackmagic Design na stronie www.blackmagicdesign.com/pl/support, aby sprawdzić najnowsze aktualizacje. Zazwyczaj zaleca się instalację najnowszych aktualizacji. Rozsądne jest jednak unikanie aktualizacji oprogramowania w trakcie realizacji ważnego projektu.

Wymogi prawne

Utylizacja zużytego sprzętu elektrycznego i elektronicznego na terenie Unii Europejskiej.
Symbol na produkcie oznacza, że tego urządzenia nie wolno utylizować razem z innymi odpadami. Aby zutylizować zużyty sprzęt, należy przekazać go do wyznaczonego punktu zbiórki w celu recyklingu. Oddzielna zbiórka i recykling zużytego sprzętu w czasie utylizacji pomoże oszczędzić zasoby naturalne i zapewni, że zostanie on poddany recyklingowi w sposób chroniący zdrowie ludzi i środowisko. Więcej informacji na temat miejsc, w których można oddać zużyty sprzęt do recyklingu można uzyskać w lokalnym biurze ds. recyklingu w Twoim mieście lub u sprzedawcy, od którego zakupiłeś produkt.

Niniejsze urządzenie zostało przetestowane i uznane za zgodne z ograniczeniami dla urządzeń cyfrowych klasy A, zgodnie z częścią 15 przepisów FCC. Ograniczenia te zostały opracowane w celu zapewnienia rozsądnej ochrony przed szkodliwymi zakłóceniami podczas pracy urządzenia w środowisku komercyjnym. Tego typu sprzęt generuje, wykorzystuje i może wypromieniować energię o częstotliwości radiowej. Jeśli nie jest zainstalowany i używany zgodnie z instrukcją, może powodować szkodliwe zakłócenia w komunikacji radiowej. Użytkowanie tego produktu na terenach zamieszkałych może powodować szkodliwe zakłócenia, w którym to przypadku użytkownik będzie zobowiązany do usunięcia zakłóceń na własny koszt.

Działanie niniejszego sprzętu podlega dwóm następującym warunkom:

1. Niniejszy sprzęt lub urządzenie nie może powodować szkodliwych zakłóceń.
2. Niniejszy sprzęt lub urządzenie musi odbierać wszelkie zakłócenia zewnętrzne, w tym zakłócenia mogące powodować niepożądane funkcjonowanie.

R-R-BMD-20250213001

ICES-3 (A)
NMB-3 (A)

Certyfikacja ISED na rynek kanadyjski
Niniejsze urządzenie jest zgodne z kanadyjskimi normami dla urządzeń cyfrowych klasy A.

Wszelkie modyfikacje lub użycie tego produktu niezgodnie z jego przeznaczeniem może unieważnić zgodność z tymi normami.

Podłączenie do interfejsów HDMI musi być wykonane przy użyciu wysokiej jakości ekranowanych kabli HDMI.

Niniejsze urządzenie zostało przetestowane pod kątem zgodności z przeznaczeniem do użytku w środowisku komercyjnym. Jeśli urządzenie jest używane w środowisku domowym, może powodować zakłócenia radiowe.

Informacje dotyczące bezpieczeństwa

Dostarczony zasilacz AC do 24 V DC dla Blackmagic URSA Cine Immersive musi być podłączony do gniazda sieciowego z uziemieniem ochronnym.

Blackmagic URSA Cine Immersive nie zawiera żadnych części, które mogą być naprawiane przez użytkownika. Serwisowanie należy zlecić lokalnemu centrum serwisowemu Blackmagic Design.

W przypadku podłączania wejściowego złącza zasilania do zewnętrznego źródła akumulatora DC, okablowanie zasilające musi zawierać ogranicznik prądu lub bezpiecznik.

Zastosowane okablowanie powinno być oznaczone VW-1 lub zgodne z odpowiednimi częściami normy IEC 60332 lub IEC 60695".

Zalecamy, aby źródłem prądu stałego dla tej kamery było 24 V. Jeśli używane jest źródło prądu stałego 12 V, okablowanie powinno być wystarczające dla natężenia prądu wynoszącego 14 A.

Wraz z tym produktem dostarczana jest płyta akumulatorowa B-Mount. Zalecamy stosowanie akumulatora 24 V (28,8 V) lub akumulatora o podwójnym napięciu.

W przypadku akumulatora 12 V (14,4 V) należy zapoznać się z instrukcją obsługi akumulatora lub oznaczeniami w celu określenia maksymalnego ciągłego prądu rozładowania. Zalecamy, aby miał on minimalną wartość znamionową 12 A.

Kalifornijskie ostrzeżenie Prop 65

Niniejszy produkt może narażać użytkownika na działanie substancji chemicznych, takich jak śladowe ilości polibromowanych bifenyli w częściach plastikowych uznanych w stanie Kalifornia za powodujące raka i uszkodzenia płodu lub działające szkodliwie dla rozrodczości.

Więcej informacji na stronie www.P65Warnings.ca.gov.

Biuro Europejskie

Blackmagic Design Europe B.V.
Rijnlanderweg 766, Unit D
2132 NM Hoofddorp
NL

Blackmagic URSA Cine Immersive

W słoneczne dni należy rozważyć osłonę kamery przed słońcem, aby zapobiec wystawieniu jej lub akumulatora litowego na długotrwałe działanie promieni słonecznych. Akumulatory litowe należy trzymać z dala od wszelkich źródeł ciepła.

Złącze wyjściowe 24 V **RS** jest dostosowane do zasilania akcesoriów innych producentów. Proszę upewnić się, że pobór mocy nie przekracza 48 W.

Gwarancja

Ograniczona gwarancja

Firma Blackmagic Design gwarantuje, że niniejszy produkt będzie wolny od wad materiałowych i produkcyjnych przez okres 12 miesięcy od daty zakupu. Jeśli produkt okaże się wadliwy w okresie gwarancyjnym, Blackmagic Design, według własnego uznania, albo naprawi wadliwy produkt bez opłat za części i koszty naprawy, albo zapewni wymianę na produkt zastępczy w zamian za wadliwy produkt.

W celu uzyskania usługi w ramach niniejszej gwarancji, klient musi powiadomić Blackmagic Design o wadzie przed upływem okresu gwarancyjnego i dokonać odpowiednich przygotowań do wykonania usługi. Klient jest odpowiedzialny za zapakowanie i opłacenie wysyłki wadliwego produktu do wyznaczonego centrum serwisowego wskazanego przez Blackmagic Design. Klient jest odpowiedzialny za opłacenie wszystkich kosztów wysyłki, ubezpieczenia, ceł, podatków i innych opłat za produkty zwrócone do nas z jakiegokolwiek powodu.

Niniejsza gwarancja nie obowiązuje w przypadku wad, awarii lub uszkodzeń spowodowanych niewłaściwym użytkowaniem lub niewłaściwą lub niedostateczną konserwacją i pielęgnacją. W ramach niniejszej gwarancji firma Blackmagic Design nie jest zobowiązana do: a) naprawy uszkodzeń wynikających z prób instalacji, naprawy lub serwisowania produktu przez personel inny niż przedstawiciele Blackmagic Design, b) naprawy uszkodzeń wynikających z niewłaściwego użytkowania lub podłączenia do niekompatybilnego sprzętu, c) naprawy wszelkich uszkodzeń lub nieprawidłowego działania spowodowanego użyciem części lub materiałów eksploatacyjnych innych niż Blackmagic Design, lub d) serwisowania produktu, który został zmodyfikowany lub zintegrowany z innymi produktami, jeśli w wyniku takiej modyfikacji lub integracji zwiększa się czas lub trudność serwisowania produktu.

Wystawienie wizjera URSA Viewfinder na bezpośrednie działanie światła słonecznego może spowodować uszkodzenie wyświetlacza wizjera, ponieważ optyka wizjera działa jak szkło powiększające. Na panelach OLED może wystąpić zatrzymanie lub wypalenie obrazu, gdy statyczne lub kontrastowe obrazy, takie jak linie kadru, są wyświetlane na nich przez dłuższy czas. Aby tego uniknąć, należy upewnić się, że czujnik podczerwieni do wykrywania twarzy nie jest celowo zasłonięty i odłączyć wizjer, gdy nie jest używany przez dłuższy czas. Zatrzymanie obrazu nie jest objęte gwarancją na ten produkt.

NINIEJSZA GWARANCJA UDZIELANA PRZEZ BLACKMAGIC DESIGN ZASTĘPUJE WSZELKIE INNE GWARANCJE, WYRAŹNIE OKREŚLONE LUB DOROZUMIANE. BLACKMAGIC DESIGN I JEJ DOSTAWCY WYŁĄCZAJĄ WSZELKIE DOROZUMIANE GWARANCJE PRZYDATNOŚCI HANDLOWEJ LUB PRZYDATNOŚCI DO OKREŚLONEGO CELU. ODPOWIEDZIALNOŚĆ BLACKMAGIC DESIGN ZA NAPRAWĘ LUB WYMIANĘ USZKODZONYCH PRODUKTÓW W OKRESIE GWARANCJI JEST JEDYNYM I WYŁĄCZNYM ZADOŚĆUCZYNIENIEM DLA KLIENTA. BLACKMAGIC DESIGN NIE BĘDZIE PONOSIĆ ODPOWIEDZIALNOŚCI ZA JAKIEKOLWIEK SZKODY POŚREDNIE, SPECJALNE, PRZYPADKOWE LUB WTÓRNE, NIEZALEŻNIE OD TEGO, CZY BLACKMAGIC DESIGN LUB DOSTAWCA ZOSTALI WCZEŚNIEJ POWIADOMIENI O MOŻLIWOŚCI WYSTĄPIENIA TAKICH SZKÓD. BLACKMAGIC DESIGN NIE PONOSI ODPOWIEDZIALNOŚCI ZA JAKIEKOLWIEK NIELEGALNE UŻYCIE SPRZĘTU PRZEZ KLIENTA. FIRMA BLACKMAGIC NIE PONOSI ODPOWIEDZIALNOŚCI ZA JAKIEKOLWIEK SZKODY WYNIKAJĄCE Z UŻYTKOWANIA NINIEJSZEGO PRODUKTU. UŻYTKOWNIK OBSŁUGUJE TEN PRODUKT NA WŁASNE RYZYKO.

© Copyright 2025 Blackmagic Design. Wszelkie prawa zastrzeżone. „Blackmagic Design", „URSA", „DeckLink", „HDLink", „Workgroup Videohub", „Multibridge Pro, „Multibridge Extreme", „Intensity" i „Leading the creative video revolution" są zarejestrowanymi znakami towarowymi w Stanach Zjednoczonych i innych krajach. Wszystkie inne nazwy firm i produktów mogą być znakami towarowymi odnośnych przedsiębiorstw, z którymi są powiązane.

Słowny znak towarowy i logo Bluetooth® są zarejestrowanymi znakami towarowymi należącymi do Bluetooth SIG, Inc. i wszelkie wykorzystanie tych znaków przez Blackmagic Design odbywa się na podstawie licencji. Inne znaki towarowe i nazwy handlowe są własnością ich właścicieli.

Жовтень 2025 р.
Посібник із камери

Blackmagicdesign

Blackmagic URSA Cine Immersive



Зміст

Blackmagic URSA Cine Immersive 1829
Що таке імерсивне відео? 1829
Імерсивне відео Apple 1829
URSA Cine Immersive 1830
Підготовка до роботи 1831
Підключення живлення 1831
Установлення налаштувань запису 1832
Моніторинг зображення 1834
Опції моніторингу та виведення службових параметрів 1834
Кадрування під час першої зйомки 1835
Висота камери 1835
Нахил камери 1836
Розташування 1837
Взаємодія з глядачем 1838
Рух камери 1838
Таблиці швидкості передачі даних 1839
Аудіо й тайм-код 1840
Просторовий аудіоформат Apple 1840
Амбісонічний звук 1840
Петличні та мініатюрні мікрофони 1841
Ручні мікрофони 1841
Підключення джерела тайм-коду 1841
Зйомка 1843
Вимоги до експозиції 1843
Швидкість затвора 1843
ISO та динамічний діапазон 1844
ND-фільтри 1845
Спеціальний запис 1846
Гістограма 1847
Опції моніторингу на РК-дисплеї 1847
Запис 1848
Відтворення 1849
Передача кліпів 1850
Огляд 1851
DaVinci Resolve Studio 1852
Рекомендації щодо робочої станції 1852
Налаштування проекту 1853
Головні налаштування 1853
Моніторинг відео 1854
Масштабування зображення 1854
Керування кольором 1855
DaVinci Wide Gamut 1856
Обробка імерсивного відео Apple 1857
Імпорт 1857
Монтаж 1859
Збирання 1859
Фон 1859
Вікно перегляду 1861
Переходи 1863
Дані руху під час зйомки імерсивного відео Apple 1864
Титри 1865
Творчі аспекти 1866
VFX-композитинг 1867
Вузли MediaOut 1867
Рендеринг 3D-сцен 1868
Композитинг із застосуванням імерсивного коректора 1869
Колірний грейдинг 1870
Шумозаглушення 1870
Первинна колірна корекція 1871
Вторинна колірна корекція 1871
Параметри 1872
Панель 3D 1872
Керування кольором за допомогою ефекту CST 1874
Перетворення на колірний простір часової шкали 1874
Перетворення колірного простору для моніторингу грейдингу 1875
Перетворення колірного простору для моніторингу на пристрої Apple Vision Pro 1877
Аудіо 1878
Просторовий аудіоформат Apple (ASAF) 1878
Експорт кінцевої версії 1879
Пропуск переходів 1879
Стрімінг на пристрій Apple Vision Pro 1880
Apple Immersive Video Utility 1880
Інформація для розробників 1881
Camera Control REST API 1881
Livestream Control API 1883

Clips Control API 1887
Media Pool Control API 1888
Monitoring Control API 1891
Event Control API 1900
System Control API 1900
Transport Control API 1906
Timeline Control API 1910
Media Control API 1913
Slate Control API 1916
Preset Control API 1922
Audio Control API 1924
Lens Control API 1929
Video Control API 1933
Camera Control API 1941
Immersive Control API 1944
Color Correction Control API 1945
Notification websocket - 1.0.0 1950
messages 1950
Допомога 1974
Дотримання нормативних вимог 1975
Правила безпеки 1976
Гарантія 1977

Blackmagic URSA Cine Immersive

Вітаємо вас із придбанням або прокатом моделі Blackmagic URSA Cine Immersive.

Blackmagic URSA Cine Immersive створено для зйомки стереоскопічного VR-зображення з кутом огляду 180 градусів, яке зазвичай називають імерсивним. Камера підтримує запис імерсивного відео Apple для гарнітури Apple Vision Pro, яка забезпечує найкращий на сьогодні ефект занурення.

Цей посібник дає короткий огляд функціоналу моделі Blackmagic URSA Cine Immersive і її застосування для зйомки імерсивного відео Apple. Розуміння цього формату допоможе опанувати роботу із самою камерою.

Нижче подано низку порад щодо технічних і творчих методів, які не лише дадуть змогу вивчити функціонал цієї моделі, а й дізнатися про особливості імерсивного відео.

Що таке імерсивне відео?

Імерсивне відео — це просторовий тривимірний відеоформат. Ймовірно, ви вже мали нагоду почути про подібні технології. Щоб краще зрозуміти, у чому полягає відмінність імерсивного відео, нижче наведено детальний опис інших форматів.

- **Традиційне 3D-кіно** створює у глядача враження об'ємного зображення завдяки зйомці кожної сцени двома камерами з такою ж осьовою відстанню, що між людськими очима. Хоча стереоскопічне відео створює ілюзію занурення, воно має лише операторський ракурс.
- **Сферичне відео** має елементи тривимірної та панорамної зйомки з охопленням у 360 градусів. Це забезпечує ефект занурення, а також дозволяє глядачу керувати напрямом перегляду. Сферичне зображення потрібно записувати з дуже високою роздільною здатністю, щоб максимально точно захопити кожну частину кадру. Оскільки фізіологія людини дозволяє повертати голову лише від одного плеча до іншого, щоб побачити, що відбувається позаду, відео зазвичай переглядають стоячи чи в оборотному кріслі.
- **Об'ємне відео** не лише забезпечує ефект глибини та панорамний огляд, глядач має змогу змінювати ракурс перегляду. Для захоплення зображення використовують камерні системи, а елементи віртуальної та доданої реальності створюють за допомогою 3D-моделей із динамічним рендерингом. Перегляд такого відео потребує безпечного для аудиторії середовища.

Імерсивне відео має всі переваги традиційного кіно та тривимірних інтерактивних форматів. У найпростішій формі, це стереоскопічне панорамне 180-градусне зображення з високою кадровою частотою та роздільною здатністю.

Імерсивне відео Apple

Blackmagic URSA Cine Immersive створено для запису імерсивного відео Apple. Для його перегляду використовують пристрій Apple Vision Pro. Цей формат має високу роздільну здатність і стереоскопічне тривимірне зображення з кутом огляду 180 градусів.

Функціональні характеристики імерсивного відео Apple

- Відтворення з частотою 90 fps.
- Підтримка виведення в роздільній здатності до 4320 x 4320 для кожного ока.
- Унікальна система метаданих для проекціювання оригінального зображення камери в 180-градусне середовище.
- Підтримка просторового аудіоформату Apple для об'ємного звуку.

Для перегляду імерсивного відео Apple потрібне обладнання з підтримкою відтворення стереоскопічного 180-градусного зображення з високою роздільною здатністю. Наразі таким пристроєм є лише Apple Vision Pro.



Apple Vision Pro

Ця гарнітура має два дисплеї Micro-OLED із роздільною здатністю 3660 x 3200 пікселів і щільністю 7,5 мікрона. Крім того, є 13 камер відстеження, сканер LiDAR, інерційні вимірювальні модулі, датчики відстеження очей для узгодження зображення з рухом голови та очей.

Чип M2 з 8-ядерним центральним і 10-ядерним графічним процесором забезпечує відтворення зображення з частотою до 100 fps.

URSA Cine Immersive

Камеру Blackmagic URSA Cine Immersive створено для запису імерсивного відео Apple, призначеного для пристрою Apple Vision Pro.



Blackmagic URSA Cine Immersive

Blackmagic URSA Cine Immersive має два однакові RGBW-сенсори з роздільною здатністю 58 Мп. Вони майже квадратні, що дозволяє захоплювати повне 180-градусне поле зору, як по вертикалі, так і по горизонталі. Кожен сенсор під'єднано до об'єктива «риб'яче око» з кутом огляду 210 градусів і осьовою відстанню 64 мм.

Для контролю світла є вбудовані ND-фільтри з вісьмома ступенями. Об'єктиви мають постійну діафрагму f4.5.

Камера дозволяє захоплювати два 8K-потоки з частотою 90 fps. Через це об'єм даних значно вищий, ніж при записі звичайного відео в 4K/24p.

Підготовка до роботи

Щоб розпочати роботу з моделлю Blackmagic URSA Cine Immersive, достатньо підключити до неї живлення та ввімкнути. На камері встановлено відформатований накопичувач Blackmagic Media Module 8TB, готовий до запису матеріалу.

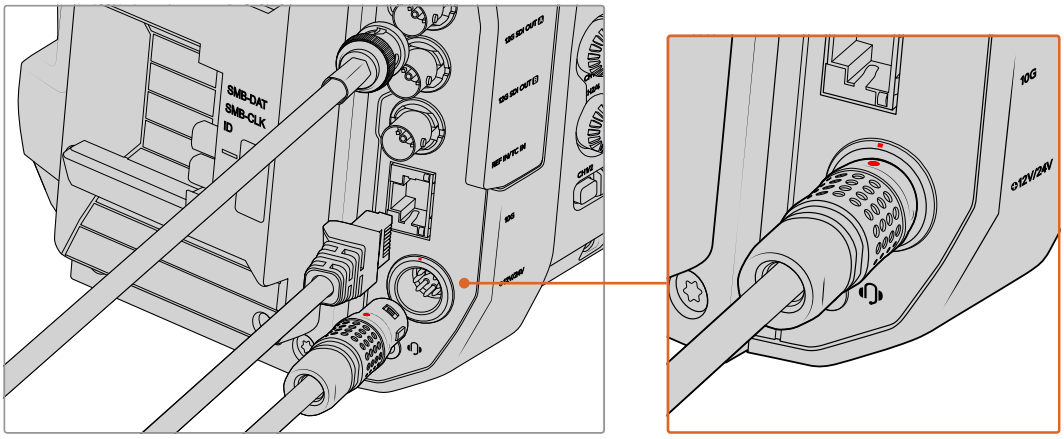
Підключення живлення

Найпростіший спосіб — використати наданий адаптер із комплекту постачання.

Порядок підключення зовнішнього джерела живлення

1. Спочатку зніміть гумову пилозахисну кришку та покладіть її в надійне місце. Вставте вилку кабелю (що на іншому кінці має гніздовий роз'єм C13 за стандартом IEC) адаптера 24 В у мережеву розетку.
2. Під'єднайте 8-контактний штекер кабелю адаптера до роз'єму 24 В DC на задній панелі камери.

При одночасному підключенні зовнішнього джерела живлення та акумуляторної батареї використовуватиметься лише зовнішнє джерело. Якщо його вимкнути, за наявності встановленої та повністю зарядженої батареї камера перейде на живлення від неї.



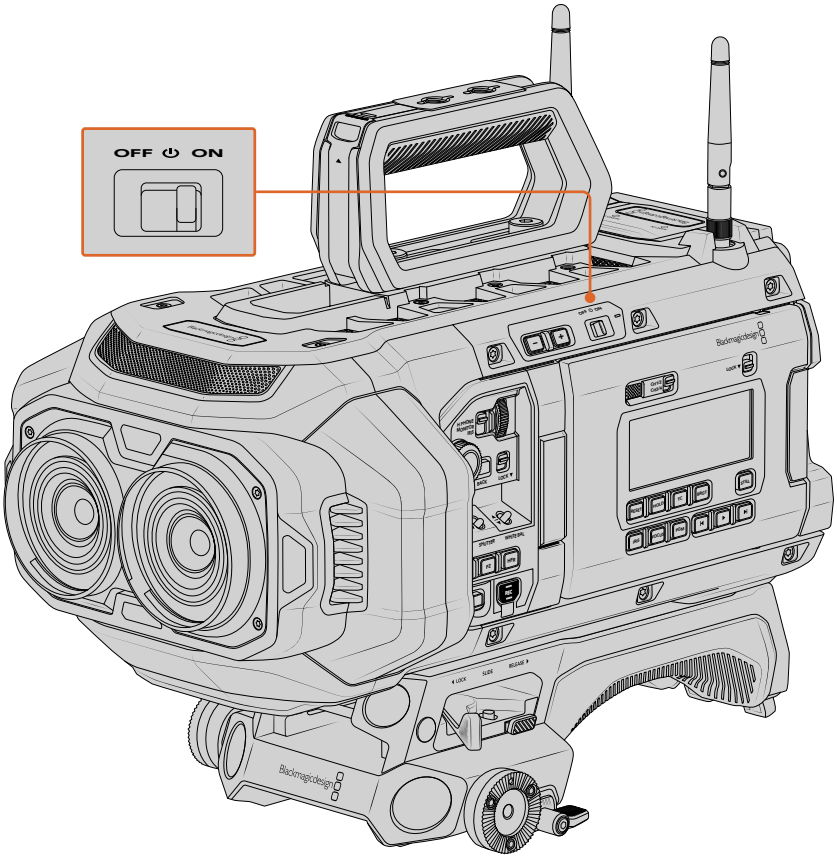
Щоб підключити живлення, сумістіть червоні крапки та обережно вставте штекер у гніздо до кінця. Щоб від'єднати кабель, візьміться за металеву трубку штекера та потягніть за неї. Штекер від'єднається від гнізда.

На камері встановлено акумуляторну пластину B-Mount, яка призначена для батарей 24 В із кріпленням B-Mount. Для роботи моделі URSA Cine Immersive живлення з напругою в діапазоні 24-34 В є ідеальним, оскільки воно знижує тепловіддачу, що дозволяє вести запис із вищою кадровою частотою та забезпечувати живленням підключені до камери аксесуари.

Також передбачено використання додаткових пластин із кріпленнями VLock і Gold Mount (24 В) та Gold Mount Plus (26 В) для застосування акумуляторів сторонніх виробників.

ПРИМІТКА. Якщо на URSA Cine Immersive надходить живлення 12-18 В, забезпечення електроенергією аксесуарів через 3-контактний роз'єм RS або 2-контактний роз'єм типу Lemo буде неможливо.

Після встановлення акумулятора або підключення до зовнішнього джерела живлення ввімкніть камеру URSA Cine Immersive. Для цього встановіть розташований зверху тумблер у положення ON.



Щоб увімкнути камеру, установіть тумблер у положення ON

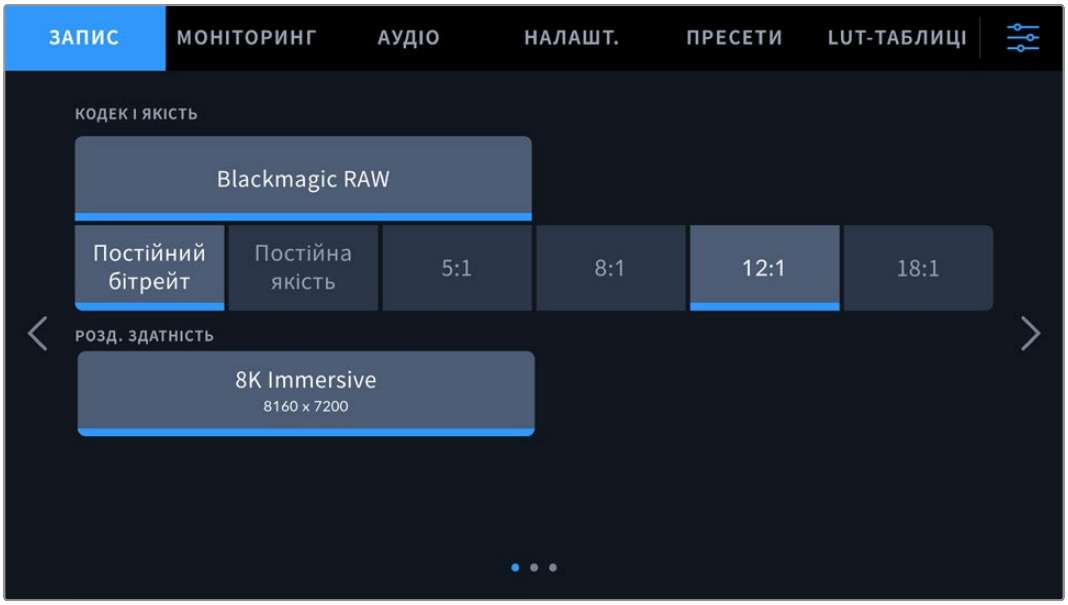
Установлення налаштувань запису

Цей розділ містить інформацію про налаштування камери, які слід використовувати під час запису імерсивного контенту. Щоб відкрити налаштування камери, торкніться піктограми меню налаштувань у верхньому правому кутку відкидного або допоміжного сенсорного екрана.



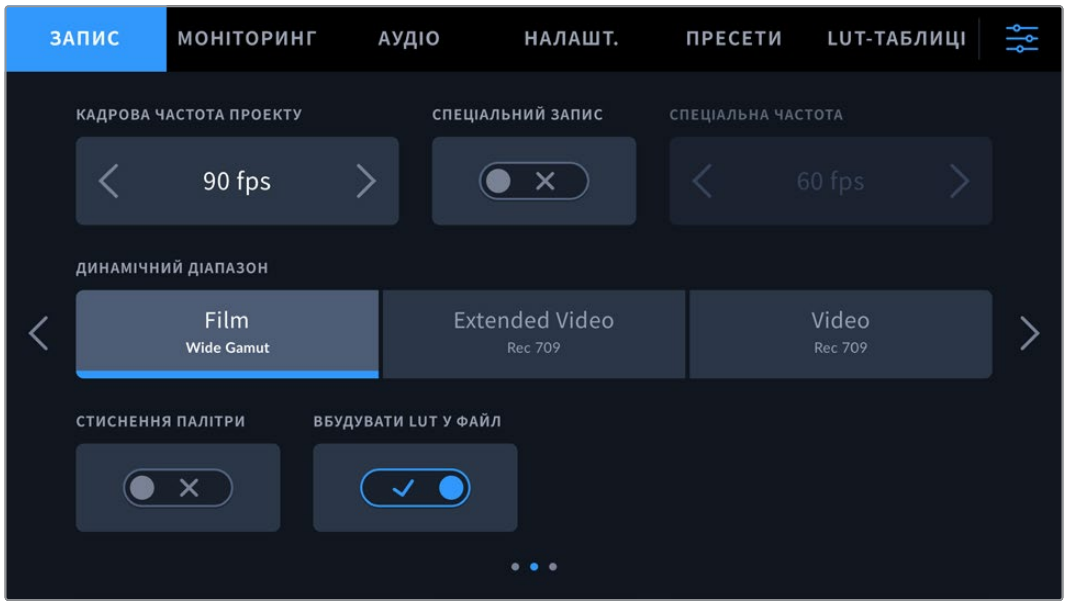
Вкладка «ЗАПИС» (стор. 1)

У меню «КОДЕК І ЯКІСТЬ» можна вибрати налаштування кодека Blackmagic RAW відповідно до вимог проекту. Передбачено по чотири параметри для запису з постійним бітрейтом і постійною якістю. Для зберігання контенту для Apple Vision Pro з кадровою частотою 90 fps радимо використовувати параметри стиснення 12:1 або Q3. Вони забезпечать оптимальну швидкість передачі даних і розмір файлів при роботі із сучасним апаратним забезпеченням, а також високу якість зображення.



Вкладка «ЗАПИС» (стор. 2)

Друга сторінка вкладки «ЗАПИС» містить налаштування кадрової частоти.



Щоб кадрова частота проекту відповідала вимогам до відео для пристрою Apple Vision Pro, потрібно використовувати значення 90 fps.

При записі для інших систем достатніми будуть нижчі значення, як-от 60 fps або 50 fps. Ще нижчу кадрову частоту, зокрема 24 fps або 25 fps, можна використовувати при зйомці візуальних ефектів для кінофільмів із налаштуванням 24 fps.

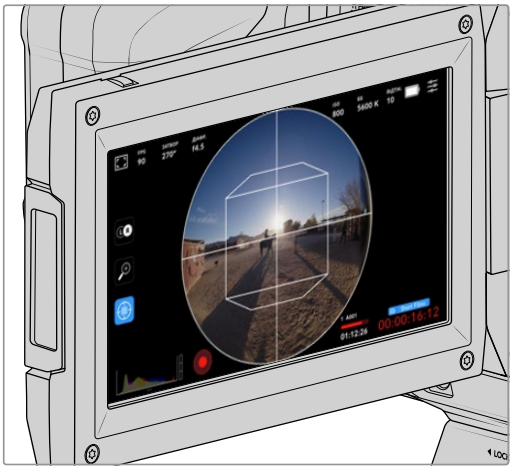
Це всі дії, які потрібно виконати для початку роботи з камерою URSA Cine Immersive. Оскільки встановлений медіамодуль уже відформатовано, на нього можна відразу зберігати матеріал.

Моніторинг зображення

URSA Cine Immersive підтримує кілька способів моніторингу зображення під час зйомки.

Для роботи оператора призначено відкидний сенсорний екран. Його можна перевернути й скласти, щоб зменшити габарити камери, але при цьому мати можливість переглядати зображення, а при самозйомці — розвернути вперед.

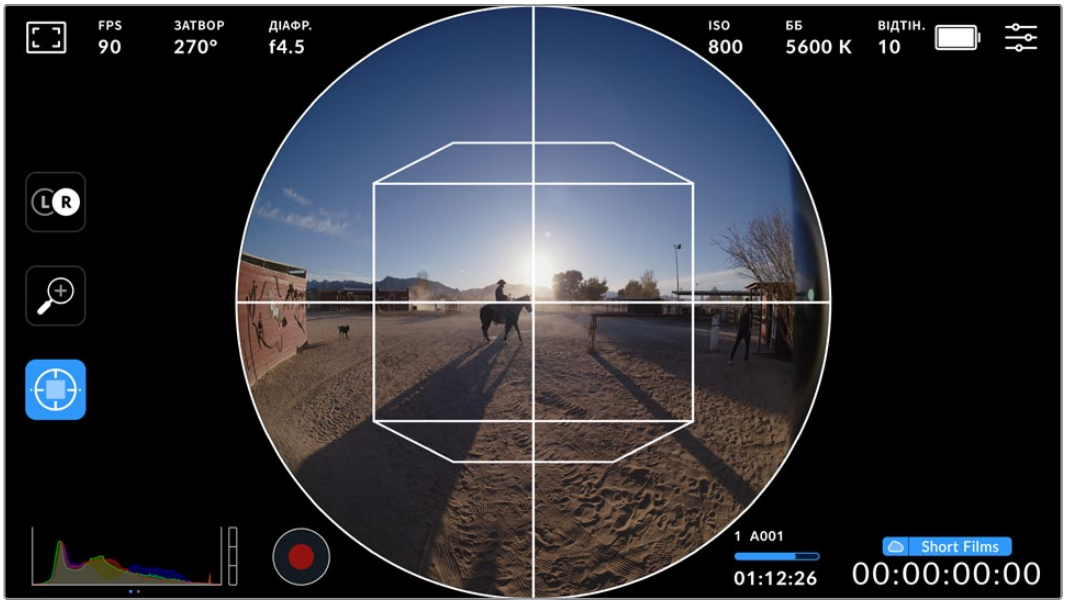
Допоміжний сенсорний екран дозволяє іншому члену знімальної групи перевіряти контент, не перешкоджаючи оператору. При моніторингу потрібно пам'ятати, що камера має надзвичайно широке поле зору.





Опції моніторингу та виведення службових параметрів

На екрани камери можна виводити зображення для одного ока за раз. На відкидному та допоміжному сенсорному екранах є маска для перегляду відео з кутом огляду 180 градусів. Через виходи SDI можна виводити зображення з кутом огляду 210 градусів, записане на кожен сенсор.



На всі виходи для моніторингу надсилається зображення для того самого ока. Для переключення між лівим і правим оком торкніться кнопки L R на будь-якому сенсорному екрані.

За допомогою відповідної кнопки можна ввімкнути виведення на сенсорний екран службових параметрів. Їхній перелік наведено нижче.

- Зовнішня межа кута огляду 180 градусів
- Перехрестя, яке позначає центр кадру
- Внутрішня рамка, яка позначає приблизне поле зору в пристрої Apple Vision Pro, якщо погляд спрямовано прямо вперед

Ці інструменти забезпечать точне кадрування та розташування елементів під час запису імерсивного контенту.

Кадрування під час першої зйомки

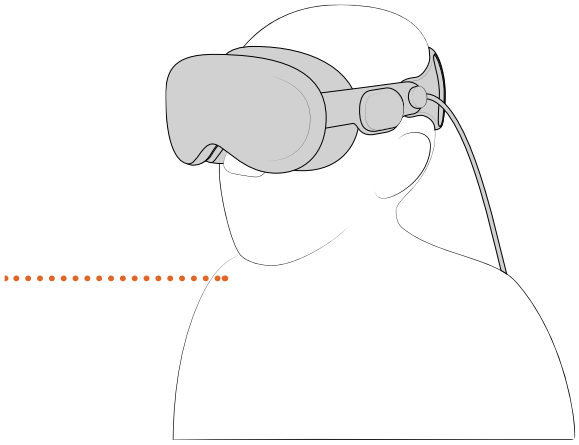
Імерсивний і традиційний двовимірний контент має багато спільного, але й чимало суттєвих відмінностей.

Наприклад, чи відчували ви захитування при використанні смартфона під час подорожі в автомобілі? Воно виникає через те, що мозок отримує суперечливі сигнали від органів відчуття. Внутрішнє вухо повідомляє про рух, а очі — що ви перебуваєте на місці.

Подібні відчуття можуть виникнути під час перегляду імерсивного зображення. Наприклад, при панорамуванні очі глядача надсилають сигнал про рух, а шия та внутрішнє вухо — про статичне положення. Тому перед тим як розпочати зйомку в русі, слід спершу добре опанувати роботу зі стаціонарною камерою.

Висота камери

У традиційному двовимірному кіно високе або низьке розміщення камери дозволяє показати співвідношення сил у сцені. Натомість імерсивне відео записне на рівні землі може сприйматись інакше. Яке саме враження воно створює в глядачів? Їм може здаватися, наче їх по шию закопано в землю.



Висота, на якій розташовано камеру під час запису, позначається на сприйманні зображення аудиторією. Якщо зйомка ведеться надто низько, можна створити небажаний ефект, ніби глядач лежить на землі.



Перегляд зображення, записаного з низького положення може викликати дискомфорт

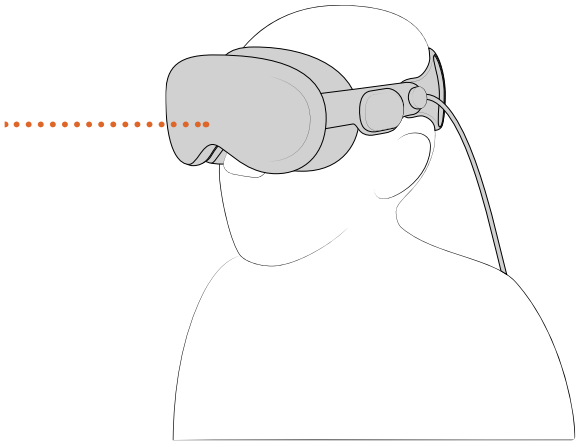
Щоб створити в глядачів відчуття присутності, потрібно вибрати такий ракурс і висоту, щоб їм здавалося, наче вони стоять або сидять у сцені. При такій зйомці зберігається можливість застосування різних художніх прийомів. У сценах із людьми, якщо камеру розмістити дещо вище, це дасть глядачам відчуття переваги, бо вони дивитимуться на інших зверху вниз. Водночас, якщо знімати з нижчої точки, аудиторія буде бачити присутніх із нижнього ракурсу.

Комфортний перегляд можна забезпечити навіть при використанні нетрадиційних способів запису. Під час виробництва імерсивного відео успішно використовують дрони, а також верхній і нижній ракурси тощо. При правильному монтажі, русі та відповідному контексті такі кадри дозволяють створити захопливий відеоряд.

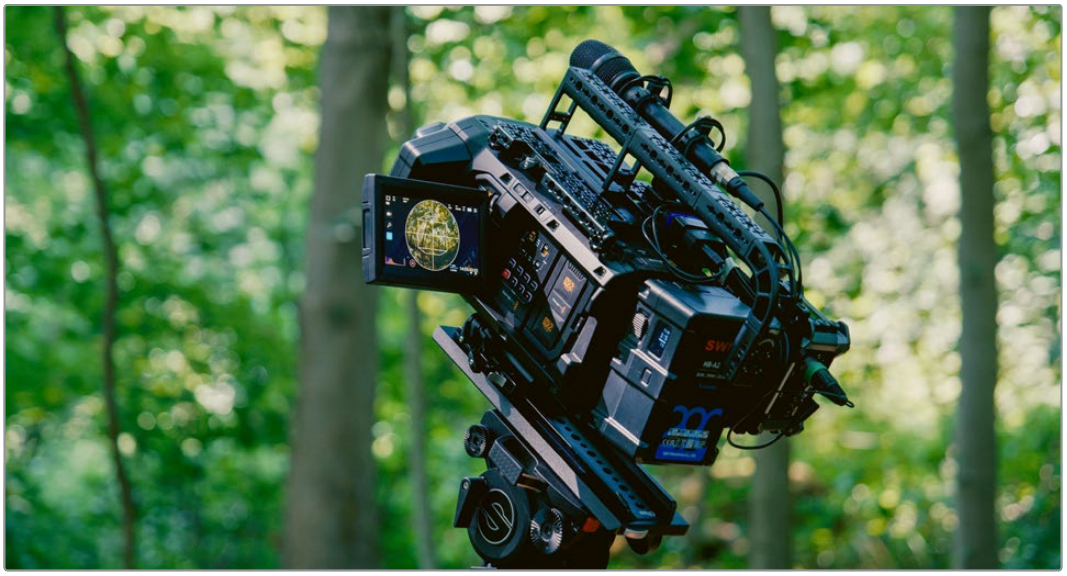
Нахил камери

Якщо під час традиційної зйомки двовимірного відео камеру встановлюють високо або низько, щоб вибрати композиційно вдалий ракурс, оператор може нахиляти камеру. Проте при записі імерсивного зображення потрібно враховувати певні особливості.

Якщо камеру нахилити вниз, щоб бачити горизонт, глядачам доводиться піднімати голову. Це не спровокує захитування, але може позначитися на ефекті присутності, оскільки в реальному житті горизонт завжди попереду.



Іммерсивне зображення найкраще
записувати при куті горизонту 0 градусів



Якщо на камеру URSA Cine Immersive знімати під нахилом, таке зображення може дезорієнтувати глядача через відмінність розташування точки горизонту

Тому зазвичай для запису імерсивного відео використовують нульовий нахил або зовсім невеликий кут. Оскільки камера має поле зору понад 180 градусів по горизонталі та вертикалі, глядачі мають змогу дивитися вгору та вниз. Проте ракурс вибирає не оператор, а аудиторія.

Щоб зосередити погляд глядача на певній частині зображення, можна використовувати різні прийоми, такі як освітлення, кадрування за допомогою зміни діафрагми або напрямних ліній.

Розташування

Імерсивний контент зазвичай дивляться сидячи. Цей формат відрізняється від доповненої чи віртуальної реальності, де глядачі стоять або рухаються. Переконайтеся, що ви правильно сидите. Для експерименту поверніть голову зі сторони в сторону та вверх і вниз. Як далеко ви можете це зробити, не обертаючи тулуб? Імерсивне зображення має більший кут огляду, ніж діапазон руху голови більшості людей. Це забезпечує периферійне бачення, навіть коли погляд спрямовано праворуч або ліворуч. Такий експеримент дозволяє краще зрозуміти, у якій частині поля зору найкраще розташовувати основний об'єкт зйомки.

Типовий діапазон руху голови

Поворот: 120-160 градусів
Нахил: 110-130 градусів

Якщо погляд спрямовано в бік повороту голови, можна побачити, що відбувається позаду. Проте для більшості людей утримати таке положення складно.

Щоб полегшити побудову композиції, увімкніть виведення рамок. Для кращого сприймання контенту об'єкт запису потрібно розмістити у внутрішніх межах, які позначають типове поле зору.

Крім того, важливо брати до уваги дію в кадрі. Під час переміщення ключового персонажа від крайньої лівої до правої точки, глядач також сильно повертатиме голову. Так само, якщо на одному кадрі об'єкт перебуває з одного боку зображення, а на іншому кадрі — з протилежного, глядачу потрібно буде різко повернути голову. Тому варто добре продумати композицію сцен, щоб забезпечити комфортний перегляд.

Взаємодія з глядачем

Під час зйомки традиційного двовимірного контенту для спрямування погляду аудиторії використовують кадрування, різні види об'єктивів і налаштування фокусної відстані. При записі імерсивного відео з кутом огляду 180 градусів доволі складно зменшити виразність одних елементів, щоб виділити інші. Імерсивний формат дає глядачу значно більшу свободу при перегляді відео. Саме тому воно настільки цікаве. Навіть за таких умов існують способи керування увагою аудиторії. Це можна зробити за допомогою спрямування звуку, освітлення, блокування та композиції.

Рух камери

Імерсивна камера не завжди має залишатися фіксованою, але потрібно зважати, що деякі її рухи можуть здаватися неприродними або створювати дискомфорт. Спосіб створення приємного для перегляду просторового зображення можна проілюструвати на прикладі з автомобілем. Велика маса та інерційна сила ускладнюють раптову зупинку чи зміну траєкторії великих транспортних засобів. Отже, їхнє прискорення й уповільнення відбуваються поступово, а рух залишається прогнозованим і контрольованим. Після того, як ви опануєте зйомку в статичному режимі, щоб створювати комфортне для перегляду імерсивне відео в динаміці, варто зважати на наведені нижче фактори.

- **Тривалість.** Глядачу потрібен час, щоб призвичаїтися до руху камери. Короткі сцени можуть дезорієнтувати аудиторію.
- **Динаміка.** Кадри з повільнішими рухами краще поєднуються під час монтажу імерсивного контенту. Щоб забезпечити комфортний перегляд, динаміку сцен потрібно нарощувати та сповільнювати поступово.
- **Візуальні прив'язки.** Це нерухомі частини сцени. Кадри без них дезорієнтують глядача. Елементи переднього плану, наприклад салон автомобіля або об'єкти на відстані, як-от лінія горизонту, допомагають стабілізувати рухомі кадри.

Таблиці швидкості передачі даних

У таблицях нижче наведено швидкість передачі даних при зйомці імерсивного контенту з роздільною здатністю 8K x 7K та кадровою частотою 90 fps і 24 fps.

Максимальна швидкість передачі даних при частоті 90 fps і налаштуваннях постійного бітрейту кодека Blackmagic RAW

Імерсивний контент 8K x 7K			
Кодек	**Розмір у пікселях**	**Кадрова частота**	**Швидкість передачі даних**
Blackmagic RAW 5:1	8160 x 7200 (стерео)	90	3203 МБ/с
Blackmagic RAW 8:1	8160 x 7200 (стерео)	90	2002 МБ/с
Blackmagic RAW 12:1	8160 x 7200 (стерео)	90	1335 МБ/с
Blackmagic RAW 18:1	8160 x 7200 (стерео)	90	889 МБ/с

Максимальна швидкість передачі даних при частоті 24 fps і налаштуваннях постійного бітрейту кодека Blackmagic RAW

Імерсивний контент 8K x 7K			
Кодек	**Розмір у пікселях**	**Кадрова частота**	**Швидкість передачі даних**
Blackmagic RAW 5:1	8160 x 7200 (стерео)	24	854 МБ/с
Blackmagic RAW 8:1	8160 x 7200 (стерео)	24	534 МБ/с
Blackmagic RAW 12:1	8160 x 7200 (стерео)	24	356 МБ/с
Blackmagic RAW 18:1	8160 x 7200 (стерео)	24	237 МБ/с

Аудіо й тайм-код

Аудіо є важливим елементом імерсивного контенту. Тому під час зйомки на камеру URSA Cine Immersive потрібно використовувати спеціальне обладнання для звукозапису.

Просторовий аудіоформат Apple

Просторовий аудіоформат Apple (ASAF) забезпечує відтворення тривимірного звуку при перегляді імерсивного відео Apple. ASAF є частиною системи позиційного аудіо Apple і підтримує три різні типи запису аудіо.

- **Канально-орієнтований.** Дозволяє вкладати традиційний об'ємний звук у файл ASAF. Ідеально підходить для запису нерухомого оповідача чи музики.
- **Об'єктно-орієнтований.** Звуки є окремими об'єктами та залежать від записаної в метаданих інформації про розташування в 3D-просторі, а не від розміщення аудіоканалів. Зазвичай це монозаписи. Їхні 3D-координати можна анімувати, щоб звуки переміщувалися разом з об'єктами сцени. Цей тип чудово підходить для дієгетичних звуків, джерелом яких є предмети в кадрі.
- **Амбісонічний.** Формат 360-градусного об'ємного аудіо, який дозволяє створювати звук у різних напрямках, використовуючи сферичні гармоніки. Цей тип ідеально підходить для запису природних шумів.

Нижче наведено опис обладнання, яке потрібне для створення якісного файлу ASAF.

Амбісонічний звук

Амбісонічні мікрофони мають чотири або більше капсул, розміщених за чіткою геометричною схемою. Кожна капсула записує ті самі звуки з різним часом захоплення та інтенсивністю, залежно від напрямку їхнього надходження. Ці капсули зберігаються в окремих каналах для пізнішої обробки на комп'ютері. Після порівняння захоплених звуків програмний додаток визначить їхні напрямок і джерело.



Мікрофон для захоплення амбісонічного звуку першого порядку

Амбісонічні мікрофони

Амбісонічні мікрофони розрізняють за кількістю капсул. Що їх більше, то точніше мікрофон відтворюватиме просторову інформацію.

Перший порядок:	4 канали
Другий порядок:	9 каналів
Третій порядок:	16 каналів
Четвертий порядок:	25 каналів і більше

Незалежно від того, де встановлено мікрофон, на камері чи біля неї, важливо, щоб його було точно вирівняно за напрямком зйомки. Оскільки амбісонічні мікрофони можна розташовувати по-різному, наприклад на одній лінії з джерелом звуку, перпендикулярно до нього, а також спрямовувати вгору або вниз, важливо повідомити звукоінженера про використовувану конфігурацію. Цю інформацію можна записати в метаданих або в примітках.

Амбісонічне аудіо може слугувати фоновим шаром для накладання інших каналів і об'єктно-орієнтованих звукових ефектів. Крім того, якщо мікрофон правильно вирівняно з камерою, запис є добрим орієнтиром для звукорежисера.

Петличні та мініатюрні мікрофони

Під час зйомки на камеру з полем зору 180 градусів неможливо використовувати мікрофонну вудку, оскільки вона потраплятиме в кадр. Тому для запису дієгетичних звуків і діалогів потрібно застосовувати приховані або петличні мікрофони.

Варіанти застосування наведено нижче.

- Під'єднайте такі джерела аудіо напряму до двох входів XLR камери URSA Cine Immersive, використовуючи бездротові передавачі та приймачі.
- Під'єднайте їх до зовнішнього пристрою для запису звуку. Це може бути як великий рекордер багатоканального аудіо, так і компактна натільна модель.

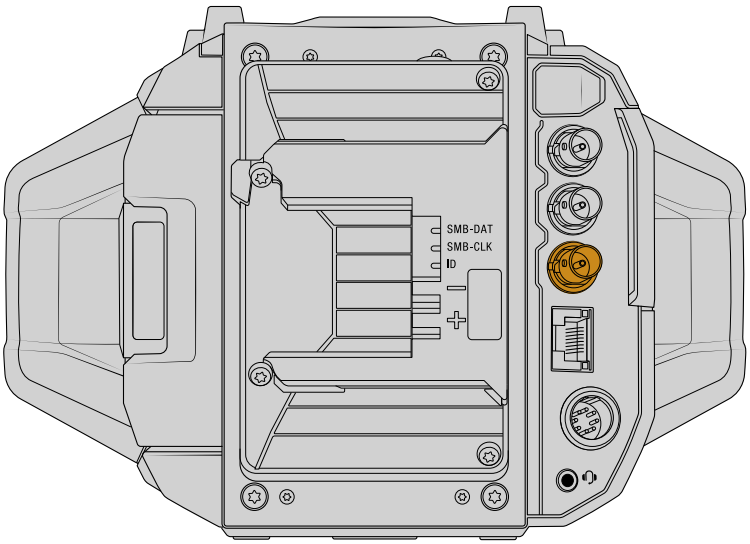
Пізніше контент можна синхронізувати для створення канально- або об'єктно-орієнтованих аудіоміксів.

Ручні мікрофони

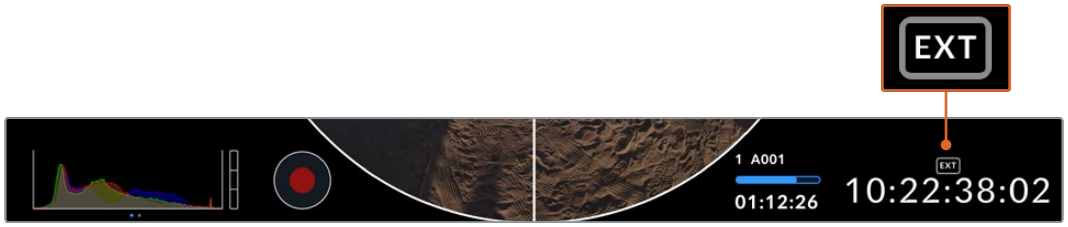
Запис додаткових звукових ефектів після зйомки відеоконтенту не лише заощадить час постобробки, а й дозволить створити природніший аудіофон. Якщо на виробничому майданчику є гіперкардіодний або подібний мікрофон, його можна встановити максимально близько до акторів, коли потрібно повторно записати сцену. Отримане аудіо можна використати для зведення фінальної доріжки в канально- або об'єктно-орієнтованому форматі.

Підключення джерела тайм-коду

Щоб підключитися до зовнішнього джерела тайм-коду, за допомогою кабелю BNC під'єднайте генератор тайм-коду до відповідного входу на задній панелі камери URSA Cine Immersive. При роботі з двома системами прив'язка до тайм-коду допомагає синхронізувати аудіо та відео на етапі поствиробництва.



Якщо кадрова частота сигналу з джерела зовнішнього тайм-коду збігається із заданими налаштуваннями, цей тайм-код буде вибрано автоматично. У цьому випадку на РК-дисплеї камери поруч із лічильником тайм-коду з'явиться позначка "EXT". Якщо від'єднати кабель, "EXT" зміниться на "INT", що вказує на використання внутрішнього тайм-коду камери.



При підключенні до генератора тайм-коду відображатиметься позначка "EXT"

ПРИМІТКА. Тайм-код за стандартом SMPTE існує лише для кадрових частот 23,976, 24, 30 тощо. Його немає для частоти 90 fps. Отже, якщо на URSA Cine Immersive вибрано кадрову частоту проекту 90 fps, використовуватиметься тайм-код для 30 fps. Виходи SDI також підтримують формат 30p. На аксесуарах для тайм-коду та зовнішніх аудіорекордерах потрібно використовувати налаштування 30 fps.

Зйомка

Базове ознайомлення з моделлю Blackmagic URSA Cine Immersive і принципами створення імерсивного контенту завершено. У цьому розділі наведено докладнішу інформацію про функціонал камери.

Вимоги до експозиції

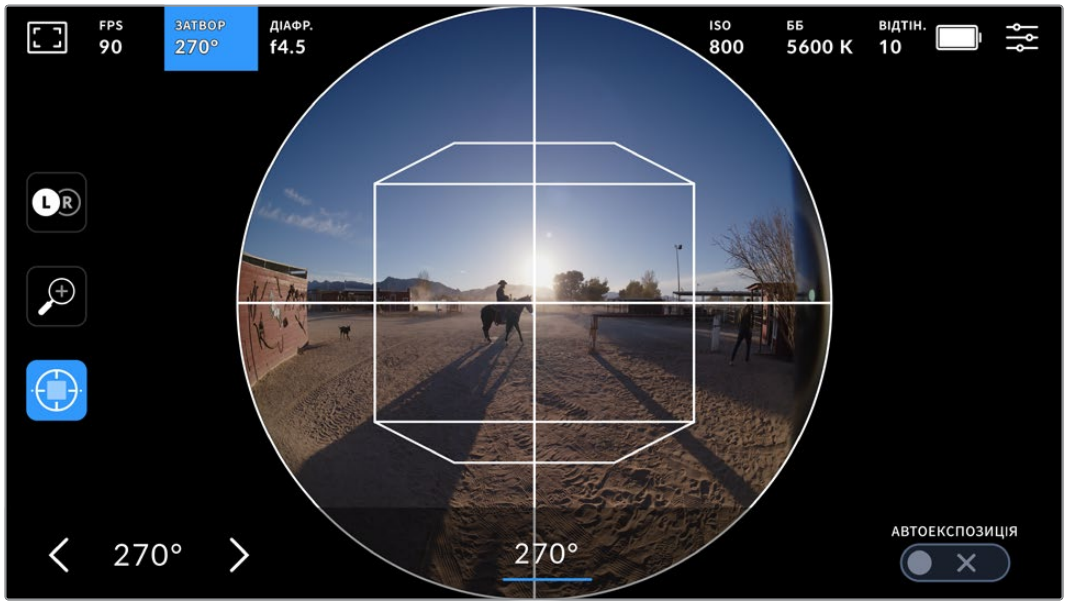
Імерсивне відео Apple має вищі вимоги до експозиції порівняно з традиційним записом із частотою 24 fps, що потребує іншого підходу до зйомки та освітлення. Для захоплення зображення при куті затвора 180 градусів із частотою 90 fps потрібно майже вчетверо більше світла. Крім того, камера має об'єктиви з постійною діафрагмою f4.5, що на три ступені менше, ніж на звичайній високошвидкісній оптиці. Таким чином, для правильної експозиції може знадобитися на сім ступенів більше освітлення. Проте, якщо при записі з частотою 90 fps використовувати значення кута затвора від 270° до 360°, це компенсує втрату світла через високу частоту кадрів і забезпечить гарний ефект розмиття.

Швидкість затвора

Індикатор «ЗАТВОР» показує кут затвора або швидкість затвора (витримку). У меню «НАЛАШТ.» камери для цього параметра передбачено два способи розрахунку та відображення.

Порядок вибору значення затвора

1. Торкніться піктограми «ЗАТВОР» на верхній панелі інструментів.
2. Для переходу між стандартними параметрами затвора використовуйте праву чи ліву стрілки. Для вибору певного налаштування двічі торкніться поточного значення в нижньому лівому кутку екрана та за допомогою клавіатури введіть необхідне число з точністю до двох десяткових знаків у діапазоні між 11,2 та 360 градусів.



Значення кадрової частоти можна ввести за допомогою екранної клавіатури

Швидкість затвора не дуже впливає на вигляд імерсивного контенту, що має високу кадрову частоту. Для запису у форматі 24 й 30p зазвичай використовують налаштування 180 градусів, проте при записі імерсивного відео експозицію регулюють саме за допомогою витримки.

Під час зйомки в умовах слабкого освітлення значення затвора можна підвищити до 270 градусів, а за виняткових обставин — до 360.

При яскравому освітленні достатньо навіть 11,2 градуса. Однак за таких умов зазвичай використовують стандартне значення затвора для правильного захоплення рухомих об'єктів, а для керування експозицією застосовують ND-фільтри.

Варто зазначити, що мерехтіння освітлення може спричиняти проблеми при записі імерсивного відео Apple із частотою 90 fps. Для зйомки в середовищі з лампами, що мають частоту 60 Гц, радимо використовувати кут затвора 270 градусів, а для 50 Гц — 324.

ISO та динамічний діапазон

Індикатор ISO показує поточне значення світлової чутливості. Торкніться індикатора, щоб змінити це налаштування. ISO на URSA Cine Immersive доступне в діапазоні від 200 до 3200, а оптимальним є значення 800.



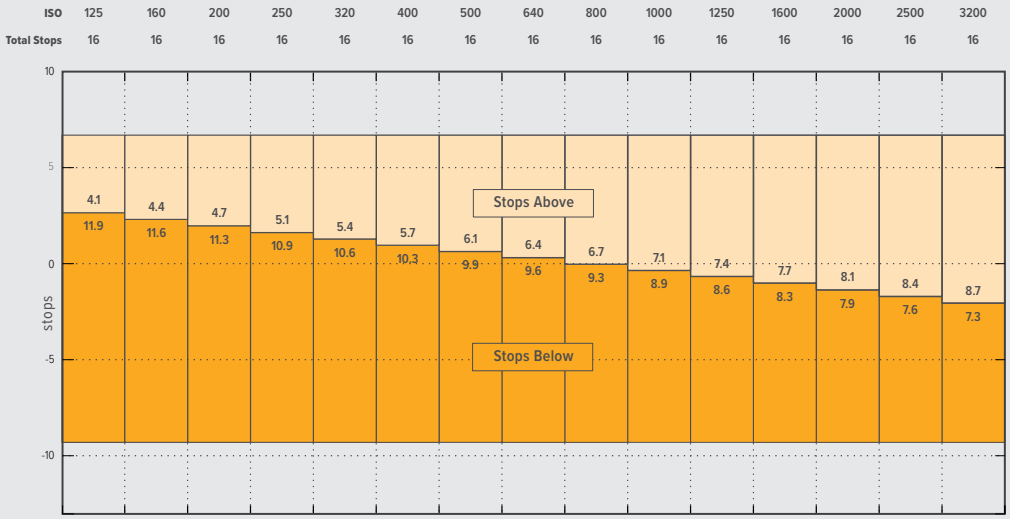
Щоб змінити значення світлової чутливості, торкніться індикатора ISO

Щоб вибрати ціле число, торкніться відповідного значення ISO. Повзунок під доступними параметрами дозволяє змінювати ISO з інтервалом 1/3 ступеня.

Залежно від конкретних умов зйомки вибирають вище або нижче значення ISO. Наприклад, для слабкого освітлення підійде налаштування ISO 1600, хоча при його використанні ймовірна поява помітного цифрового шуму. При яскравому освітленні для передачі насичених кольорів найкраще підійде ISO 200.

Камери Blackmagic дозволяють отримувати зображення із широким динамічним діапазоном, що забезпечує повну свободу при зйомці та при грейдингу в додатку DaVinci Resolve. У таблиці нижче наведено значення динамічного діапазону камери URSA Cine Immersive.

Динамічний діапазон камери URSA Cine Immersive



ND-фільтри

ND-фільтри зазвичай використовують для налаштування експозиції. Вони дозволяють зменшити кількість світла, що потрапляє на сенсор, і регулюються ступенями експозиції.



На URSA Cine Immersive фільтри ND-фільтри змінюють за допомогою кнопок на лівій панелі над відкидним сенсорним екраном. Доступні опції: Clear, 2, 4, 6 і 8 Stops.

Поточне значення ND-фільтра показує індикатор у верхньому лівому кутку сенсорного екрана. Цей індикатор відображатиметься, поки ввімкнуто ND-фільтр. При виборі опції Clear індикатор зникне за чотири секунди.



ПРИМІТКА. Для зручності позначення, що використовуються під час роботи з ND-фільтрами, можна змінити. У меню налаштувань доступні такі опції, як число, ступені та дріб.

Спеціальний запис

Це налаштування дозволяє вибрати частоту кадрів, з якою камера URSA Cine Immersive веде запис, а також відтворює матеріал. Зазвичай вибирають значення, яке відповідає швидкості відтворення та звукового супроводу в додатку постобробки.



Щоб змінити налаштування кадрової частоти, торкніться піктограми FPS. Камера URSA Cine Immersive має дев'ять налаштувань кадрової частоти проекту: 23,98; 24; 25; 29,97; 30; 50; 59,94; 60 і 90 fps. Для створення імерсивного відео потрібно використовувати значення 90 fps.

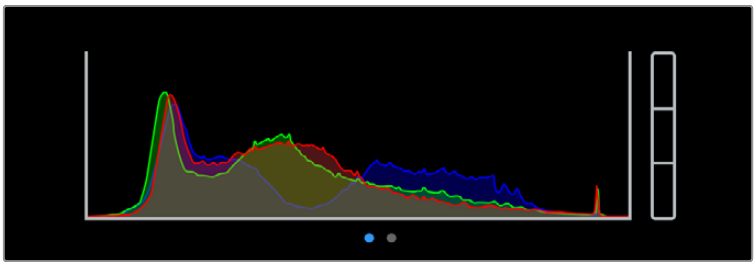
Кадрова частота сенсора дозволяє визначити швидкість, із якою він захоплюватиме зображення. За замовчуванням URSA Cine Immersive використовує кадрову частоту сенсора та проекту, яка відповідає відтворенню зі звичайною швидкістю. У разі потреби можна встановити незалежну кадрову частоту сенсора. Для цього торкніться піктограми «СПЕЦІАЛЬНА ЧАСТОТА» в нижньому правому кутку меню FPS.

При активації опції «СПЕЦІАЛЬНА ЧАСТОТА» кадрову частоту сенсора можна встановити вручну. Варто зазначити, що найвищим значенням кадрової частоти камери є 90 fps, яке відповідає аналогічній кадровій частоті проекту при зйомці для пристрою Apple Vision Pro. Якщо для проекту вибрати низьке значення кадрової частоти, наприклад 30 або 60 fps, а для запису — спеціальну частоту 90 fps, це дозволить створювати видовищний ефект уповільненого руху.

Коли опцію «СПЕЦІАЛЬНА ЧАСТОТА» активовано, індикатор у верхньому лівому кутку екрана показуватиме кадрову частоту сенсора та проекту.

Гістограма

На URSA Cine Immersive у нижньому лівому кутку сенсорного екрана відображається гістограма — графік розподілу яскравості на зображенні. Ліворуч показані темні тони, у середній частині — півтони, а праворуч — світлі. Висота графіка вказує, яке співвідношення зображення має зазначений рівень яскравості.



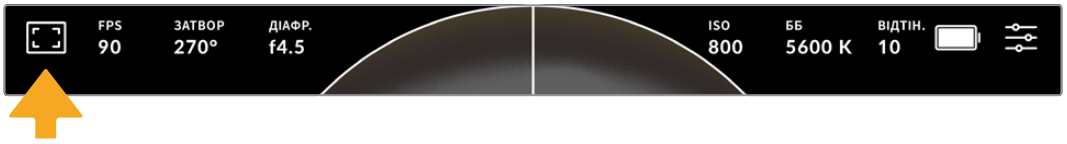
Гістограма дає змогу одразу перевірити розподіл відтінків у кліпі. Зображення, у якому графік зосереджено ліворуч, має недостатню експозицію. Відповідно, зміщений праворуч графік може свідчити про надмірну експозицію. При читанні гістограми завжди потрібно брати до уваги композицію кадру. Як правило, зображення з правильною експозицією має збалансовану, зосереджений у середній частині графік. Якщо збільшити експозицію, гістограма переміститься праворуч, а якщо зменшити — ліворуч.

З правого боку гістограми розташовано індикатор кліпінгу кожного колірного каналу. Зазвичай потрібно уникати обрізання сигналу будь-якого каналу. Однак кліпінг допускається, якщо це допоможе зберегти середні та темні тони.

Експозиція зі зміщенням праворуч — це метод, при якому навмисно використовують надмірну експозицію, щоб максимально пересунути гістограму вправо без кліпінгу колірних каналів. Це дозволяє захопити якомога більшу кількість даних при мінімальному цифровому шумі. Щоб отримати чітке зображення, достатньо на етапі постобробки зменшити рівень експозиції.

Опції моніторингу на РК-дисплеї

Щоб отримати доступ до налаштувань моніторингу, торкніться піктограми дисплея у верхньому лівому кутку сенсорного екрана. Це меню містить налаштування яскравості екрана, а також засоби, що допомагають установлювати експозицію.



Зебра

Дане налаштування дозволяє вибирати спосіб виведення на сенсорному екрані, а також встановлювати його рівень. Ця функція відображає діагональні лінії в тих областях, де експозиція перевищує встановлений рівень. Параметр відсотків використовують для відображення частин зображення з надмірною експозицією. Щоб мати можливість її виправити, встановлюють значення 95%, а не 100%. Друге налаштування дозволяє вмикати та вимикати середню зону сірого або середню зону сірого плюс один крок.

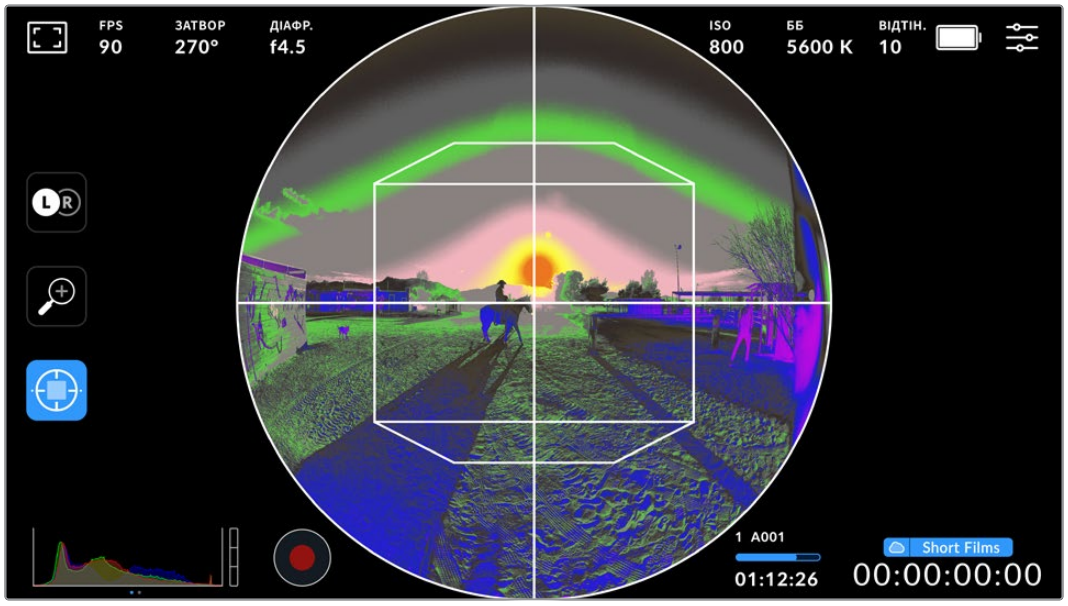
Сітка

Це налаштування дозволяє відображати один із видів сітки («Третини», «Горизонт», «Перехрестя» та «Точка») на сенсорному екрані, а також вибирати її для виведення через усі виходи камери. Під час зйомки імерсивного контенту дуже допомагає індикатор горизонту. Якщо камеру встановлено ідеально рівно, у центрі екрана відображається подвійний хрест з узгодженими лініями, тоді як при відхиленні вони розходяться.

Умовний колір

Ця функція додає до зображення різні кольори, які позначають рівень експозиції для його окремих елементів. Наприклад, рожевий є показником оптимальної експозиції для достовірної передачі відтінків світлої шкіри, зелений — темнішої. Під час зйомки людей ця функція допомагає забезпечити однорідність зображення.

Якщо його елементи змінюють колір із жовтого на червоний, це означає надмірну експозицію. Так само, як гістограма, цей інструмент дозволяє швидко перевірити розподіл відтінків на зображенні.



Запис

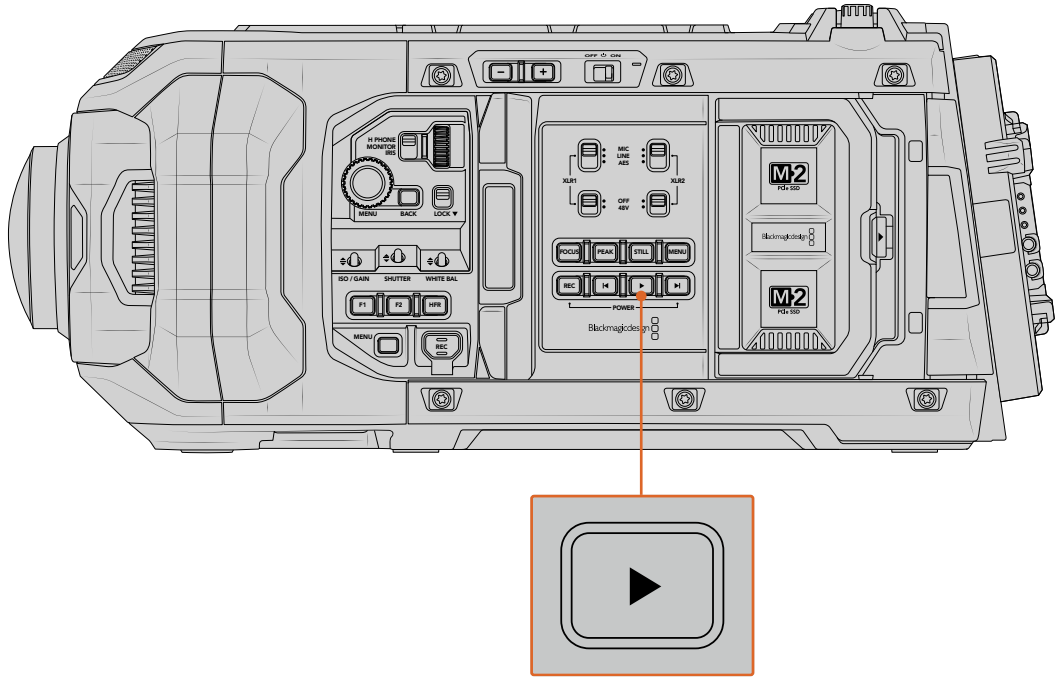
Щоб розпочати запис, натисніть одну з червоних кнопок REC. Ці кнопки розташовані в передній частині панелі керування, на внутрішній панелі за відкидним екраном, на панелі допоміжного блока та на обох сенсорних екранах. Натисніть цю кнопку ще раз, щоб зупинити запис.

Індикатори накопичувачів показують залишок часу запису. Час відображається у форматі години:хвилини:секунди з урахуванням вибраних значень кадрової частоти. Залежно від стану накопичувача смуга, розташована над значенням часу запису, може мати синій, білий або червоний колір, а ступінь заливки візуально показує рівень заповнення.



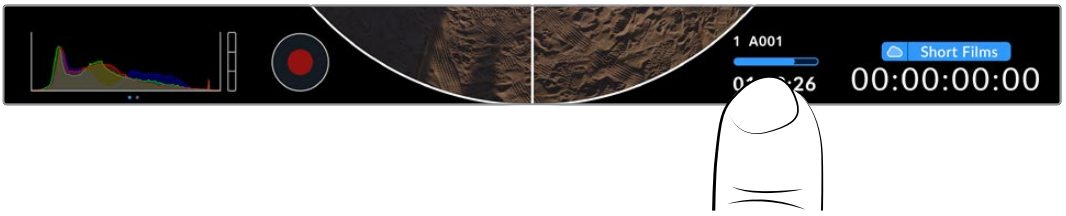
Відтворення

Після того, як кліпи записано, їх можна переглянути, використовуючи кнопки керування відтворенням. Щоб переглянути останній записаний кліп, натисніть фізичну кнопку відтворення, розташовану на лівій панелі камери поза відкидним сенсорним екраном. Для навігації між кліпам використовують кнопки зі стрілкою вліво та вправо. Щоб вийти з режиму відтворення та продовжити запис, натисніть кнопку запису один раз.

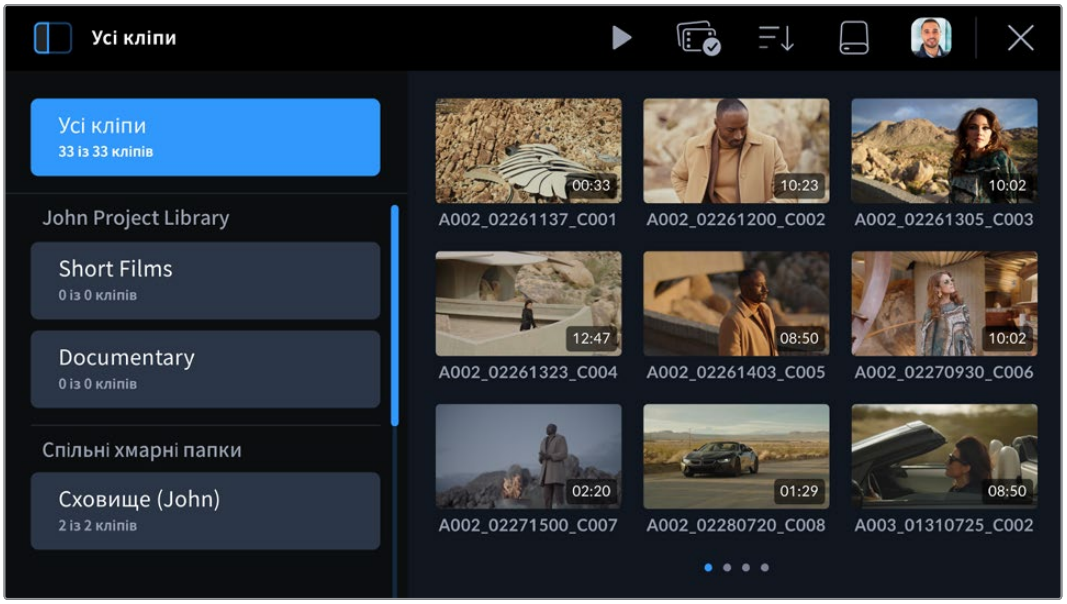


Щоб перейти до початку або закінчення кліпу, натисніть кнопку перемотки вперед або назад. Для переходу до початку поточного кліпу натисніть кнопку зворотної перемотки один раз, для переходу до початку попереднього кліпу — два рази. Щоб виконати звичайне або зворотне відтворення з подвоєною швидкістю, натисніть і утримуйте кнопку перемотки вперед або назад. Якщо в режимі перемотки натиснути одну з цих кнопок два, три чи чотири рази, швидкість відтворення збільшиться в 4, 8 і 16 разів відповідно.

Для розширеного огляду відзнятого контенту краще використовувати медіатеку. Щоб її відкрити, торкніться індикатора накопичувача внизу сенсорного екрана камери.



У браузері відображаються ескізи кліпів, збережених на носіях, що під'єднані до камери. Для ескізів передбачено три рядки, а якщо є додаткові кліпи, вони відображаються на наступній сторінці, до якої можна перейти, провівши пальцем по сенсорному екрану. Внизу екрана є індикатори, що показують кількість наявних сторінок із кліпами.



Щоб надсилати матеріал напряму в проект DaVinci Resolve, під'єднайте URSA Cine Immersive до облікового запису Blackmagic Cloud.

Для виходу з медіатеки торкніться піктограми хрестика у верхньому правому кутку або один раз натисніть будь-яку з кнопок запису.

Передача кліпів

Є різні способи копіювання кліпів із камери. Наприклад, можна під'єднати URSA Cine Immersive до комп'ютера або мережі за допомогою кабелю 10G Ethernet і скопіювати кліпи з камери через Ethernet. Можна навіть вставити медіамодуль у пристрій Blackmagic Media Dock, підключений до мережі або комп'ютера, що дозволить одночасно працювати з трьома накопичувачами.

При під'єднанні через Ethernet комп'ютер розпізнає Blackmagic Media Module так само, як і Blackmagic Cloud Store.

За замовчуванням на камері вибрано DHCP, тому комп'ютер автоматично знайде її в мережі й надасть їй IP-адресу.

Увімкнення спільного доступу до файлів

При первинному налаштуванні камери, підключеної до комп'ютера або мережі через порт Ethernet, спільний доступ до файлів за протоколом SMB за замовчуванням вимкнено. На протилежність до пристрою Blackmagic Cloud Store, призначеного для надання відповідним користувачам швидкого доступу до файлів, на медіамодулі містяться оригінальні кліпи, записані протягом усього дня, і тому вони можуть потребувати додаткового сортування. З цієї причини ми вирішили надати вам самим можливість увімкнути спільний доступ за протоколом SMB.

1. Щоб увімкнути спільний доступ до файлів, потрібно спочатку завантажити й інсталювати на комп'ютері утиліту Blackmagic Camera Setup. Вона дозволяє оновлювати програмне забезпечення камери, а також змінювати певні налаштування. Утиліту Blackmagic Camera Setup можна завантажити з нашого вебсайту за посиланням https://www.blackmagicdesign.com/ua/support/family/professional-cameras
2. Після завантаження й інсталяції утиліти підключіть комп'ютер до порту USB, розташованого в задній частині верхньої панелі камери, і запустіть Blackmagic Camera Setup.

3 Щоб відкрити налаштування, клацніть зображення камери URSA Cine Immersive або її піктограму налаштувань.

4 Прокрутіть до секції Network Access і виберіть опцію Enabled для параметра File sharing (SMB). Щоб застосувати налаштування, натисніть кнопку Save у нижньому правому кутку.

5 Тепер клацніть розташовану нижче піктограму URL, щоб автоматично ввести адресу. У ній відображатиметься ім'я камери як локальне мережеве розташування.

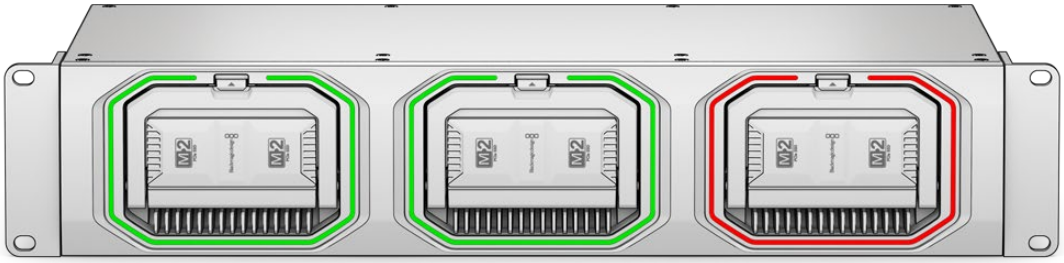
6 Клацніть Save.

Щоб перейти до кліпів, клацніть URL-адресу камери. Відкриється вікно із запитом ввести пароль або під'єднатися як гість. Виберіть опцію Guest і клацніть Connect.

7 Комп'ютер автоматично перейде до вашого медіамодуля в браузері Finder у системі Mac або у файловому провіднику у Windows. Задати налаштування потрібно лише один раз. Утиліта їх запам'ятає, і тому наступного разу при підключенні камери через порт Ethernet ви автоматично отримаєте доступ до медіамодуля.

Робота з пристроєм Blackmagic Media Dock

Blackmagic Media Module можна швидко встановити, а також вилучити, коли потрібно перенести файли на зовнішній накопичувач або замінити на незаповнений модуль для продовження запису. Пристрій Blackmagic Media Dock використовують для доступу до контенту, що міститься на вилученому з камери накопичувачі Blackmagic Media Module.



Огляд

Імерсивне відео Apple — це новітній спосіб візуальної оповіді, альтернативний класичному двовимірному кінематографу. Його створено не для того, щоб замінити традиційний формат. Цей спосіб забезпечує авторам проектів додаткові можливості. Як і з будь-яким іншим форматом, усе залежить від творчого підходу до розкриття сюжету.

Зазвичай застосування нових технологій є складним процесом, тому камеру URSA Cine Immersive розроблено так, щоб полегшити запис імерсивного відео Apple. Вона забезпечує простий підхід до виробництва такого контенту, і її також можна використовувати для створення візуальних ефектів та іншого матеріалу.

У цьому посібнику описано основні функції та принцип роботи моделі URSA Cine Immersive. Він призначений лише для ознайомлення з цим рішенням. Як і з будь-якими іншими знімальними системами, для досягнення найкращого результату потрібно експериментувати та вивчати функції камери в кожному конкретному випадку.

DaVinci Resolve Studio



DaVinci Resolve Studio — це універсальний додаток, який дозволяє в єдиному програмному середовищі виконувати монтаж і колірну корекцію, накладати візуальні ефекти, створювати анімаційну графіку та зводити звук. Він підтримує різні формати, у тому числі Blackmagic RAW Immersive — стереоскопічне відео, що має роздільну здатність Ultra HD, кадрову частоту 90 fps і просторове аудіо. DaVinci Resolve Studio містить усі необхідні засоби для виробництва, моніторингу та експорту імерсивного відео Apple, що робить його ідеальним рішенням для творців такого контенту.

Рекомендації щодо робочої станції

Рекомендації щодо апаратного забезпечення

- Mac Studio M3 Ultra Desktop
- Монітор із високою частотою оновлення (мінімум 60 Гц)

Мінімальні системні вимоги

- macOS 15 або пізніша версія
- 64 ГБ системної пам'яті
- Для моніторингу: Blackmagic Design Desktop Video 12.9 або пізніша версія
- Комп'ютер або графічний процесор на платформі Apple Silicon із підтримкою технології Metal

Вимоги для моніторингу імерсивного відео

- Apple Vision Pro
- Високошвидкісна бездротова мережа (мінімум Wi-Fi 5, бажано Wi-Fi 6)

Налаштування проекту

Перш ніж імпортувати RAW-файли, записані на Blackmagic URSA Cine Immersive, варто спочатку вибрати необхідні налаштування проекту DaVinci Resolve. Це забезпечить правильну дебайєризацію та обробку контенту, оптимізує виведення зображення та полегшить монтаж.

Головні налаштування

При налаштуванні проекту важливо вказати, що створюваний контент призначатиметься для Apple Vision Pro. Це забезпечить видимість усіх необхідних налаштувань і опцій під час підготовки, обробки та експорту матеріалу проекту.

1. Запустіть DaVinci Resolve Studio.
2. У нижньому правому кутку вікна Project Manager клацніть New Project.
3. Введіть ім'я проекту, укажіть розташування медіафайлів (якщо потрібно) і клацніть Create.
4. Відкрийте вікно Project Settings, клацнувши піктограму шестерні в нижньому правому кутку додатка DaVinci Resolve Studio.

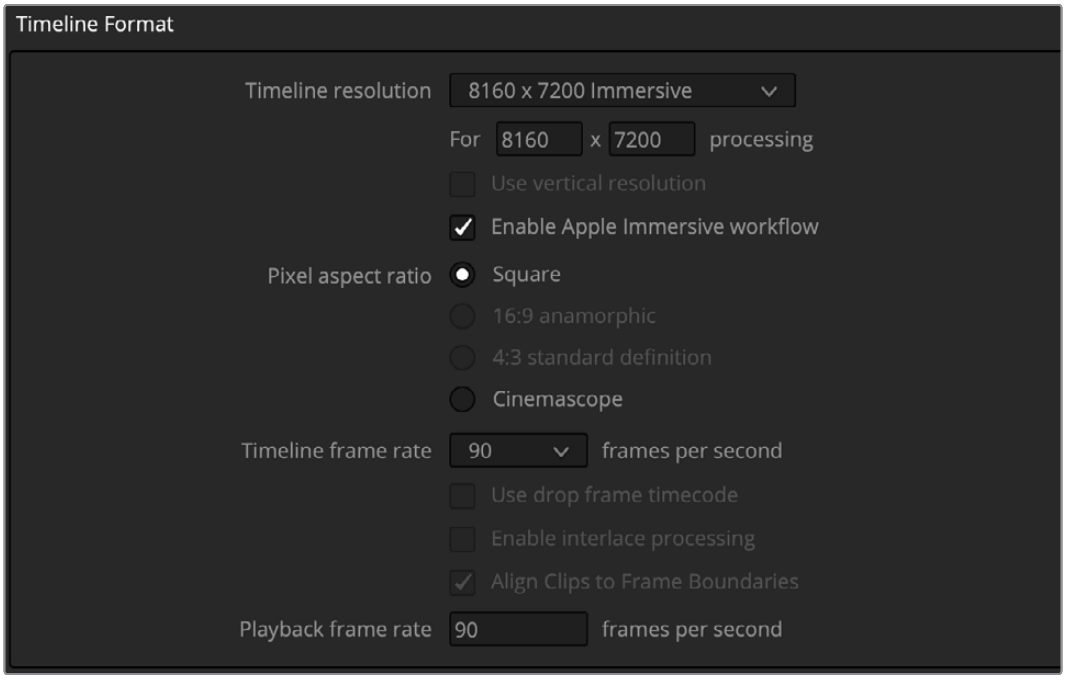
   Спочатку потрібно задати роздільну здатність і кадрову частоту проекту.
5. На бічній панелі виберіть Master Settings.
6. У секції Timeline Format установіть прапорець Enable Apple Immersive workflow.



   Це активує різні налаштування, необхідні для обробки імерсивного контенту. При створенні імерсивного проекту найперше вибирайте саме це налаштування.
7. Змініть Timeline resolution на 8160 x 7200 Immersive, щоб обробляти кліпи в оригінальній роздільній здатності сенсора.

   Якщо система недостатньо потужна, для прискорення відтворення виберіть удвічі менше значення 4080 x 3600 Immersive.
8. Для налаштування Timeline frame rate має бути задано 90 fps. Цей параметр вибирається автоматично при встановленні прапорця Enable Apple Immersive workflow.
9. Щоб імерсивне відео правильно відтворювалося, для налаштування Playback frame rate виберіть 90 fps.

   Якщо система недостатньо потужна або використовується монітор із низькою частотою оновлення, для поліпшення відтворення зменште кадрову частоту. Радимо вибирати значення, яке є дрібною частиною оригінальної кадрової частоти, наприклад 30 fps або 60 fps.



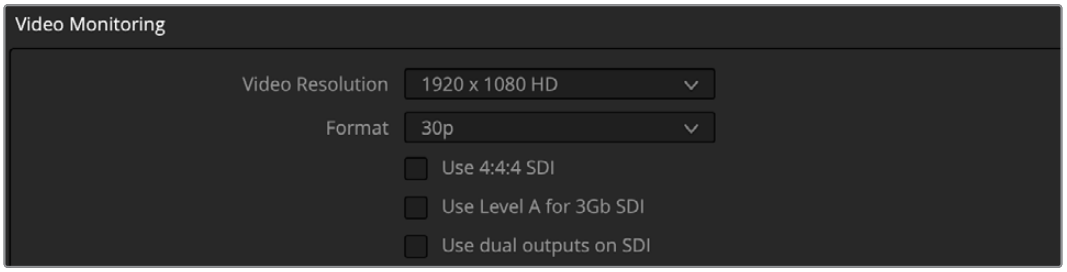
Моніторинг відео

Якщо підключено зовнішній монітор (через плату захоплення й відтворення, наприклад DeckLink або UltraStudio), виберіть налаштування моніторингу, що відповідають специфікаціям монітора.

1. У секції Video Monitoring для налаштування Video Resolution виберіть таке значення, яке відповідає роздільній здатності монітора. Наприклад, 1920 x 1080 HD або 3840 x 2160 Ultra HD.

   Оригінальна кадрова частота імерсивного відео становить 90 fps, але не всі мовні монітори її підтримують.

2. Для налаштування Format виберіть кадрову частоту монітора, наприклад 30p.



Масштабування зображення

Налаштування Image Scaling установлюють для того, щоб імерсивне зображення мало правильні пропорції та роздільну здатність у вікні перегляду. Ці параметри вибираються автоматично при встановленні прапорця Enable Apple Immersive workflow в секції Master Settings, але їх все одно потрібно перевірити.

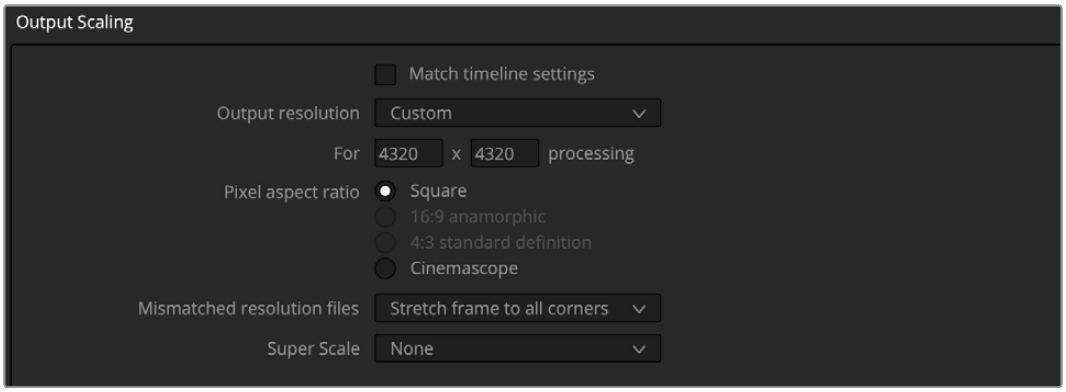
1. На бічній панелі вікна Project Settings виберіть Image Scaling.
2. У секції Output Scaling перевірте, щоб було знято прапорець Match timeline settings.

3 Переконайтеся, що для налаштування Output resolution вибрано Custom і **4320 x 4320**.

Для менших дисплеїв і (або) швидшого відтворення установіть удвічі нижчу роздільну здатність **2160 x 2160**.

Такі параметри квадратного зображення потрібні для відтворення імерсивного відео Apple у додатку DaVinci Resolve Studio та стрімінгу на Apple Vision Pro (через мережу Wi-Fi або з'єднувач Developer Strap).

4 Переконайтеся, що для налаштування Mismatched resolution files вибрано Stretch frame to all corners.



Керування кольором

Оскільки потрібно окремо налаштовувати максимальну яскравість вихідного зображення, при керуванні кольором радимо використовувати вузли Color Space Transform (CST) на сторінці Color. Налаштування Color Management вікна Project Settings застосовують для того, щоб задати робочий колірний простір часової шкали на сторінці Color і метадані колірного простору, які асоціюватимуться зі створеним файлом.

1 На бічній панелі вікна Project Settings виберіть Color Management.

2 Для налаштування Color science залиште DaVinci YRGB.

3 Для налаштування Timeline color space виберіть DaVinciWG/Intermediate.

Це колірний простір із широкою гамою, якій відповідає вимогам завтрашнього дня. Він підтримує HDR, що робить його ідеальним для редагування відео, записаного на камеру Blackmagic URSA Cine Immersive, і створення контенту для Apple Vision Pro.

4 Для налаштування Output color space виберіть P3-D65 ST2084 1000 nits.

Цей колірний простір має належну гаму для перегляду зображення у вікнах DaVinci Resolve на комп'ютері Mac Studio, у якому зазвичай застосовується Display P3. Якщо використовується спеціальний монітор для грейдингу або інший ICC-профіль Apple, для налаштування Output color space слід вибрати значення відповідно до колірного профілю дисплея.

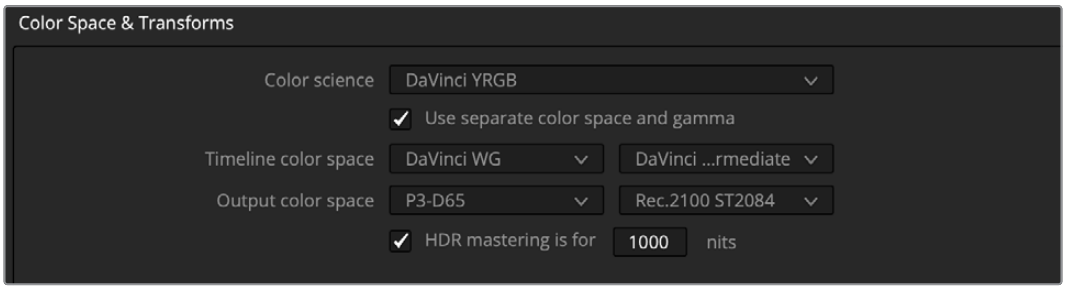
При цьому потрібно буде змінити гаму (систему передачі), щоб вона правильно відповідала ICC-профілю.

5 Установіть прапорець Use separate color space and gamma над налаштуваннями колірного простору.

Вони розділяться на окремі розкривні поля для колірного простору й гами.

6 Для вихідної гами (у правому стовпці) виберіть Rec.2100 ST2084.

7 Установіть прапорець HDR mastering is for 1000 nits.



Щоб менше витрачати часу на налаштування наступного проекту Apple Immersive, поточні параметри можна зберегти як пресет.

8 Клацніть меню опцій із трьома крапками у верхньому правому кутку вікна Project Settings і виберіть Save Current Settings as Preset.

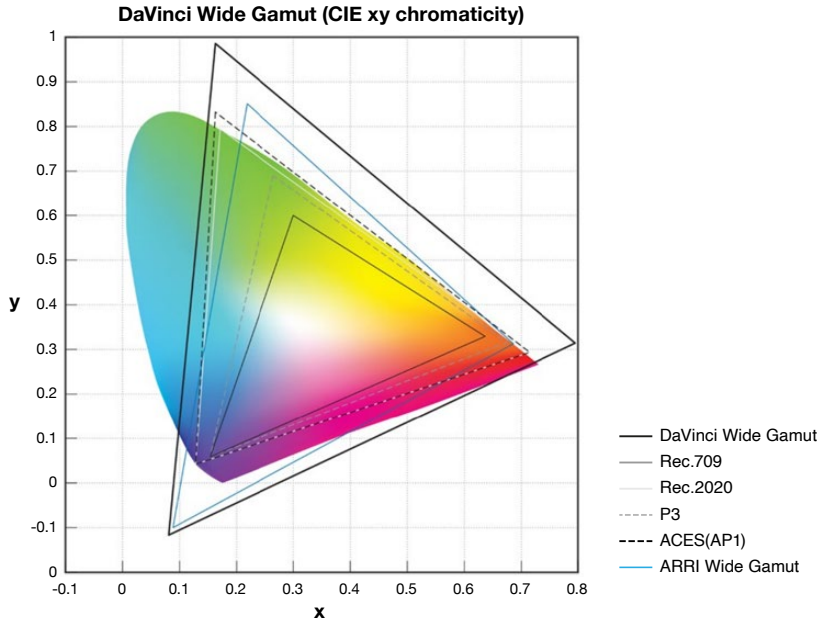
9 Введіть ім'я пресету та клацніть OK.

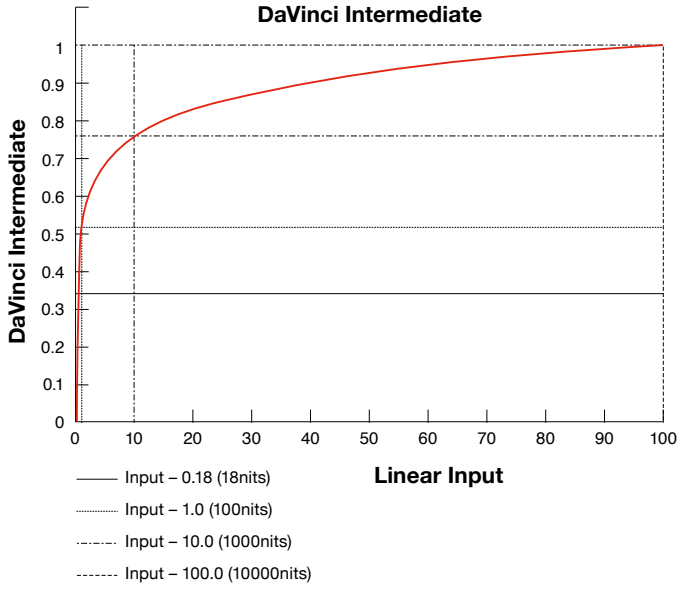
Щоб наступного разу швидко задати налаштування проекту для роботи з імерсивним контентом, відкрийте вікно Project Settings, клацніть меню опцій і виберіть потрібний пресет.

10 Натисніть кнопку Save, щоб закрити вікно Project Settings.

DaVinci Wide Gamut

DaVinci Wide Gamut — це внутрішній колірний простір Blackmagic Design, розроблений для системи DaVinci Resolve, який пропонує надзвичайно широке й точне середовище для грейдингу та обробки зображення. У комбінації з гамою DaVinci Intermediate, він створює об'єднаний робочий простір, що спрощує колірну корекцію матеріалу з різних камер, забезпечуючи одноманітність тонів і насиченості як на вході, так і на виході. Ідеально підходячи для обробки HDR- та імерсивного відео, цей простір дозволяє колористам вносити складні зміни без ризику створення колірних смуг або кліпінгу при генеруванні неймовірно чіткого кінцевого контенту.





Обробка імерсивного відео Apple

На бічній панелі Project Settings у секції Apple Immersive можна задати налаштування для даних профілю імерсивних об'єктивів (ILPD), які відповідають за калібрування та проекціювання імерсивного контенту.

1. На бічній панелі вікна Project Settings виберіть Apple Immersive.
2. Клацніть Open Calibration Folder.
3. Додайте або оновіть користувацький ILPD-файл.
4. Клацніть Update Lists, щоб оновити ILPD-контент на панелі Media Pool.
5. Натисніть кнопку Save, щоб закрити вікно Project Settings.
6. На панелі Media Pool клацніть імерсивний кліп правою кнопкою миші.
7. У контекстному меню перейдіть до Apple Immersive Calibration і виберіть первинний ILPD- (No Apple Immersive Calibration File Selected), користувацький ILPD- або проміжний ILPD-файл.

За замовчуванням папка Calibration із користувацьким ILPD-файлом розташована в папці ProIM на одному рівні зі стандартною папкою LUT.

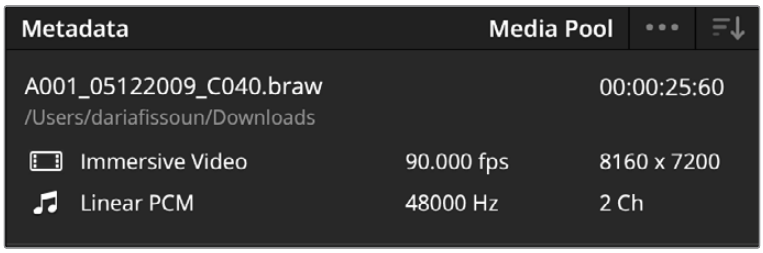
Імпорт

При імпорті імерсивного RAW-контенту зображення для обох очей і всі пов'язані з ними імерсивні метадані передаються разом з імпортованим кліпом .braw.

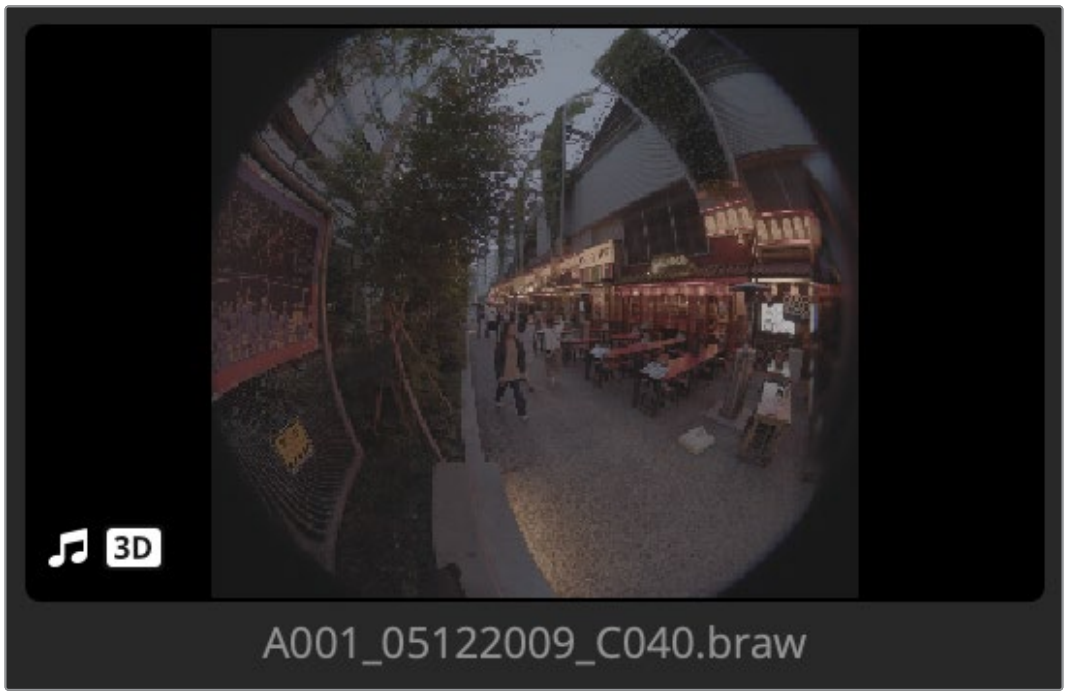
1. Перейдіть на сторінку Media.
2. На панелі Media Storage перейдіть до розташування імерсивного медіаконтенту.
3. Перетягніть його вниз на панель Media Pool.

ПОРАДА. Щоб зберегти оригінальну структуру папок, перетягніть папки з панелі Media Storage на бічну панель у Media Pool. При перетягуванні в кошик Master або будь-який інший створений кошик, порядок папок і підпапок залишається незмінним.

Коли вибрано імерсивний відеокліп Apple, на панелі Metadata в заголовку кодека відображатиметься Immersive Video, а також оригінальна кадрова частота та роздільна здатність.



Крім того, у нижньому лівому кутку відеоескізу з'явиться піктограма 3D, яка підтверджує, що кліп є тривимірним.



ПРИМІТКА. При первинному перетягуванні імерсивного контенту на панель Media Pool, може з'явитися діалогове вікно із запитом вибрати опцію Keep Settings або Change Settings. Якщо вибрати Change Settings, налаштування проекту автоматично змінюватимуться відповідно до кадрової частоти та роздільної здатності первинного медіаконтенту.

Монтаж

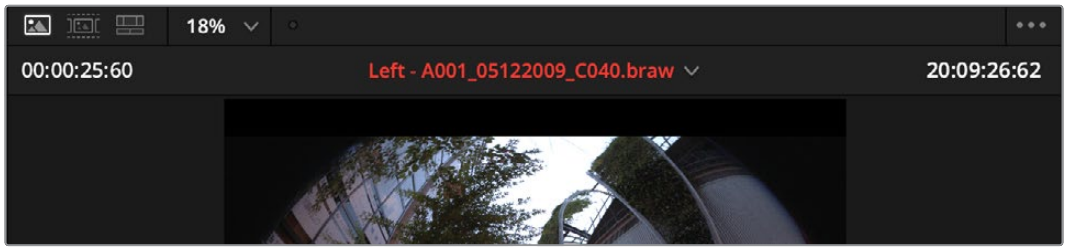
При роботі з імерсивним контентом засоби та операції монтажу переважно залишаються такими самими, як при обробці звичайного 2D-відео. Нижче наведено функції та дії, що використовуються при редагуванні саме імерсивного відео.

Збирання

При попередньому перегляді кліпів у вікні джерела під час збирання часової шкали можна вибирати зображення для лівого або правого ока. В іншому процес створення візуального ряду на часовій шкалі залишається таким самим, як при стандартному нелінійному монтажу.

1 Перейдіть на сторінку Edit.

2 Двічі клацніть кліп, щоб завантажити його в розташоване ліворуч вікно джерела. Зверху вікна виводиться ярлик, що показує, для якого ока призначено зображення.



3 Щоб змінити око у вікні перегляду, клацніть правою кнопкою миші первинний кліп на панелі Media Pool і виберіть Stereoscopic 3D > Stereoscopic 3D Mode > Left Eye/Right Eye/Both Eyes.

ПОРАДА. По ескізах на панелі Media Pool також можна провести вказівником миші, щоб прокрутити відповідні кліпи у вікні джерела, якщо в його меню опцій активовано Live Media Preview.

4 За допомогою клавіш "I" та "O" визначте фрагмент у вікні джерела.

5 Клацніть Insert Clip на панелі інструментів (або натисніть клавішу F9), щоб вставити кліп на часову шкалу.

ПОРАДА. На часову шкалу кліпи з панелі Media Pool можна також перетягнути.

6 Щоб виконати підгонку кліпів на часовій шкалі, використовуйте вказівник миші в режимі Selection або Trim Edit.

Фон

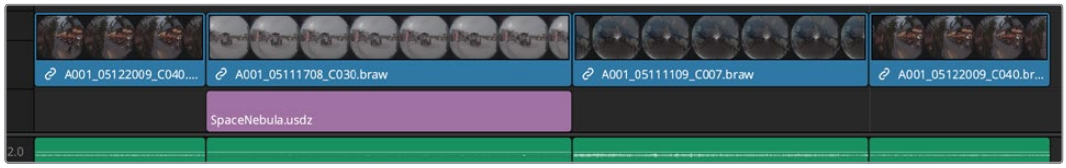
Коли в налаштуваннях проекту активовано опцію Enable Apple Immersive workflow, доріжка фону (BD) автоматично з'являється під доріжкою Video 1.

Фоном називається область, яку видно за межами периметру імерсивного відеоконтенту, що зазвичай має поле зору 180-210 градусів.

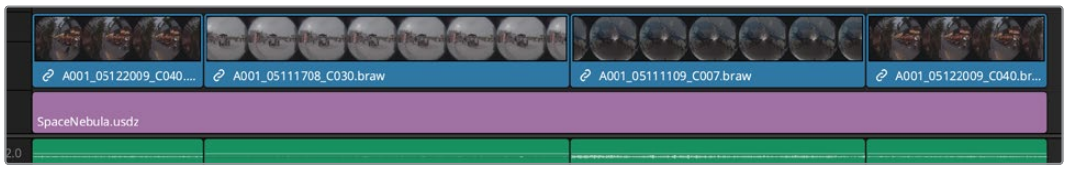
За замовчуванням ця область чорна, але можна вставити спеціально створене зображення фону (у форматі .usdz) на призначену для цього доріжку.

1 Перетягніть файл .usdz на доріжку фону, щоб додати його на часову шкалу.

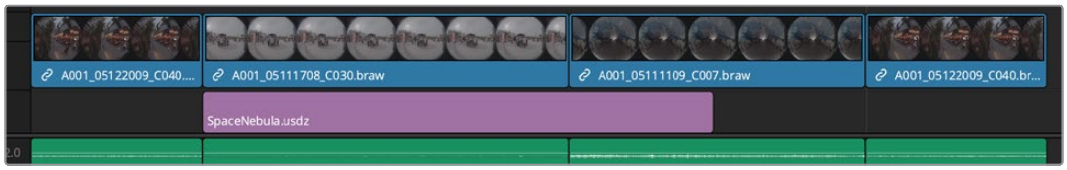
Тривалість фрагменту із зображенням фону не має виходити за межі кліпу, розташованого на відеодоріжці. Один файл фону може використовуватися для одного кліпу, кількох кліпів або всієї часової шкали, але не для частини імерсивного кліпу .braw (прийом L-cut).



Один кліп — допустимо ✓

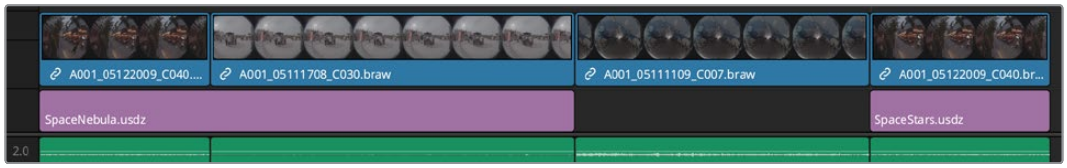


Кілька кліпів — допустимо ✓



L-cut — недопустимо ✗

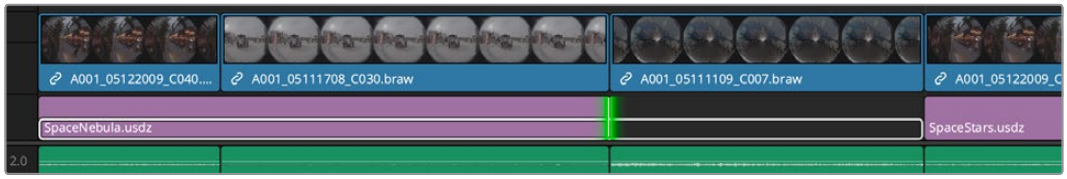
Можна використовувати кілька файлів фону на одній часовій шкалі або взагалі їх не додавати, щоб було стандартне чорне тло.



Різні зображення та проміжки — допустимо ✓

2 Довжину файлу фону регулюють за допомогою стандартних засобів підгонки.

Щоб змінити довжину файлу .usdz, перетягніть його край (у режимі Selection або Trim Edit).



Також можна застосовувати клавішу "," (кома) та "." (крапка), аби пересунути край лише на один кадр.

ПОРАДА. Щоб полегшити вирівнювання фонів за стиками кліпів імерсивного відео, увімкніть режим Snapping на панелі інструментів сторінки Edit.

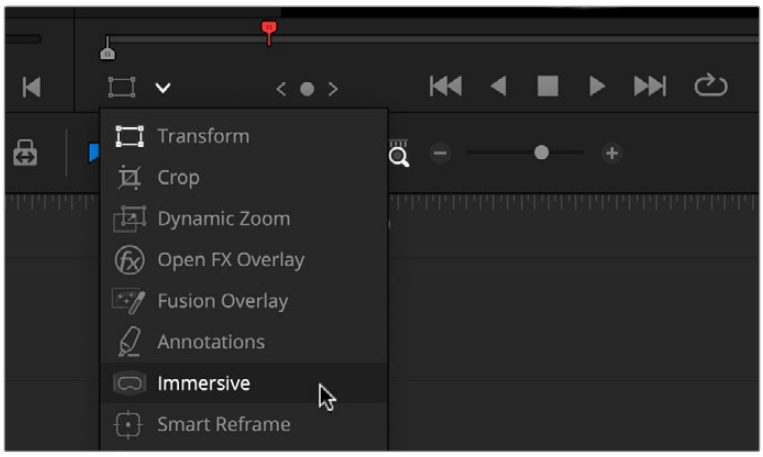
Вікно перегляду

За замовчуванням імерсивні кліпи відтворюються у вікнах додатка DaVinci Resolve у круглому «оптичному просторі». Візуальне калібрування лінзи, що використовується для проекціювання зображення, виконується на камері та є унікальним для кожного кліпу.



Також можна змінити режим перегляду на LatLong або Viewport.

1. У нижньому лівому кутку вікна часової шкали клацніть піктограму розкривного меню, розташовану біля кнопки екранних елементів керування.
2. Виберіть Immersive.



Над вікном перегляду з'явиться нова панель інструментів із додатковими опціями.

3. Клацніть LatLong, щоб побачити рівнокутну проекцію зображення. Цей режим викривляє зображення до пропорцій більш традиційного 2D-контенту, що полегшує певні процеси постобробки, наприклад відстеження та композитинг.



4 Клацніть Viewport, щоб побачити центральну ділянку імерсивної сцени у квадратному форматі. Цей режим дозволяє краще роздивитися деталі сцени з правильними пропорціями на стандартному моніторі, проте потрібно пам'ятати, що певні об'єкти можуть залишатися за кадром.

За допомогою налаштувань трансформації, розташованих на панелі інструментів над вікном перегляду, можна нахиляти та панорамувати стереоскопічне зображення, а також регулювати горизонтальне й вертикальне поле зору.



Переходи

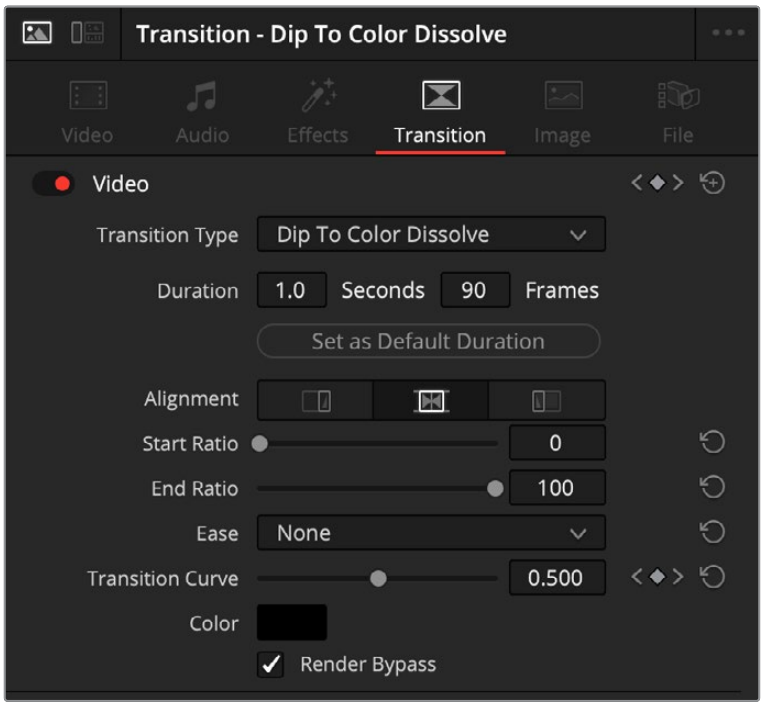
Для переходу між двома імерсивними кліпами можна застосовувати швидку зміну кадру, затемнення та появу, що визначають пересуванням буферних меж, або ефект Dip To Color Dissolve. Переходи мають додаватися пристроєм Apple Vision Pro під час відтворення, а не вбудовуватися у відео.

1. Відкрийте панель Effects.
2. На бічній панелі перейдіть до Toolbox > Video Transitions.
3. Перетягніть ефект Dip To Color Dissolve на точку стику між двома кліпами.



ПРИМІТКА. Щоб додати перехід, на кліпах має бути достатньо буферного контенту для підгонки з обох боків стику. Якщо його замало, перехід не накладеться.

4. На панелі Inspector перейдіть на вкладку Transition і встановіть прапорець Render Bypass. Це забезпечить правильне створення напливу пристроєм Apple Vision Pro.
5. На панелі Inspector також можна задати інші параметри ефекту Dip To Color Dissolve, а саме тривалість, вирівнювання, швидкість анімації та колір переходу.

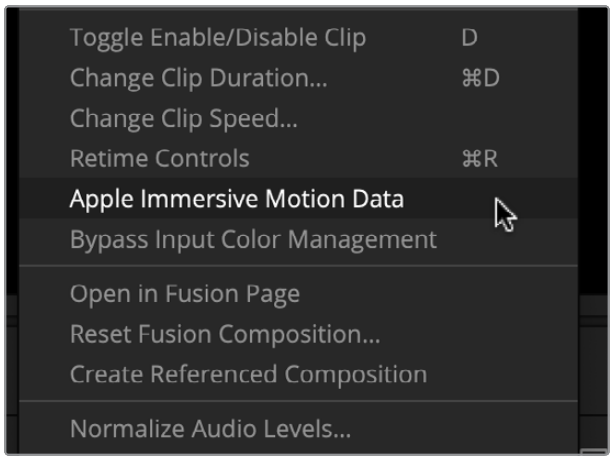


Дані руху під час зйомки імерсивного відео Apple

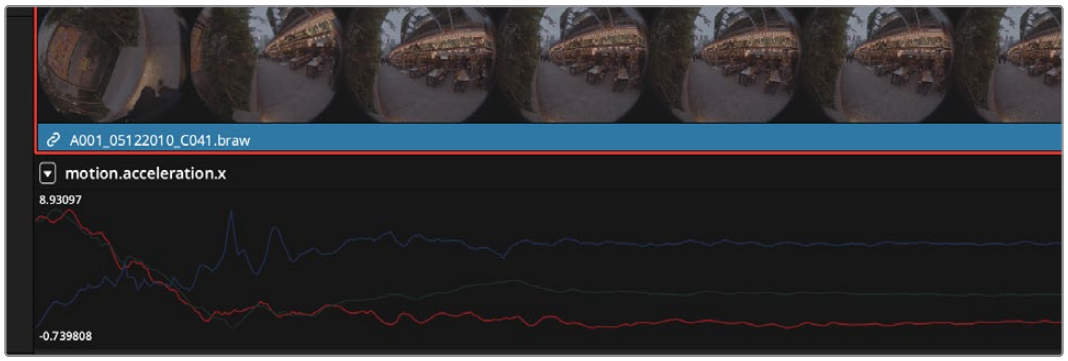
При роботі з імерсивним контентом швидкі або різкі рухи камери можуть дезорієнтувати глядача. Щоб забезпечити комфортний перегляд, уникайте використання кадрів із метушливим панорамуванням або нахилами, обертами та нерівними горизонтами.

Налаштування Apple Immersive Motion Data створює графік прискорення по осях x, y і z зі значеннями положення гіроскопа для показу ділянок із надмірними рухами камери. Монтажери можуть застосовувати цю інформацію, щоб уникати використання дезорієнтуючих фрагментів.

1. Клацніть правою кнопкою миші імерсивний кліп на часовій шкалі.
2. Виберіть Apple Immersive Motion Data.

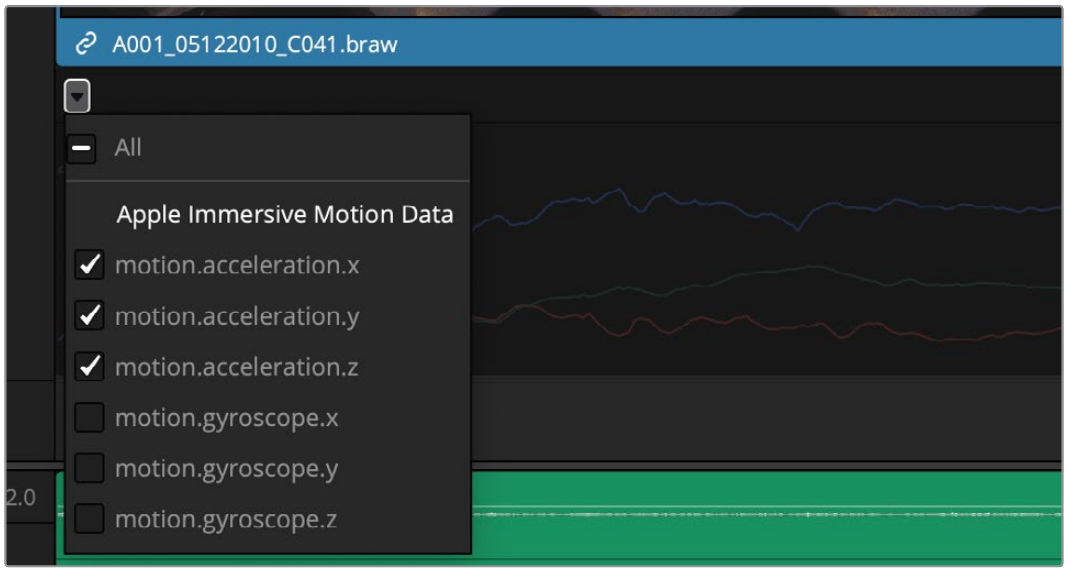


На відеодоріжці з'явиться графік прискорення по осях x, y і z.



У цьому прикладі камера вела запис, коли оператор налаштовував її. Пікові значення на графіку свідчать про раптові рухи, коли камеру піднімали, розвертали та встановлювали на плече. В ідеалі краще використовувати останню частину відео без різких змін показників, записану після стабілізації знімального обладнання.

3 У верхньому лівому кутку графіка клацніть стрілку меню переміщення.



У ньому можна ввімкнути або вимкнути відображення показників залежно від кадру та напрямку руху камери.

Титри

На сторінці Edit до імерсивних сцен можна додавати текст за допомогою розташованих на панелі Effects інструментів Titles — Text, Text+, MultiText, Scroll або одного з Lower Third (Left, Middle або Right).

Коли генератор титрів розміщено над імерсивним кліпом, він використовуватиме дані профілю імерсивних об'єктивів (ILPD) для точного викривлення тексту відповідно до оптичних спотворень на зображенні.

Якщо ці дані недоступні, наприклад при додаванні тексту на чорний фон або в імерсивну сцену з комп'ютерною графікою, то викривлення об'єктива застосовуватиметься за замовчуванням, щоб при перегляді на пристрої Apple Vision Pro текст мав відповідний вигляд.

1 На панелі Effects перейдіть до Toolbox > Titles.

2 Перетягніть генератор тексту на часову шкалу. Розмістіть його на відеодоріжці над імерсивним кліпом, щоб у ньому з'явився текст.



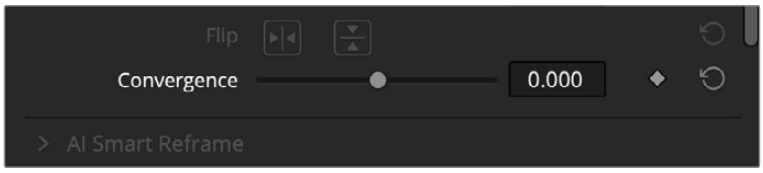
Як і фонове зображення, генератори тексту не мають виходити за межі кліпу. Оскільки кожен кліп створюється з індивідуальним оптичним простором, при перекриванні титри можуть спотворюватися між кадрами.

3 Підженіть титри відповідно до монтажних стиків відео.

4 На панелі Inspector введіть текст титрів і відформатуйте його.

5 Перейдіть на вкладку Settings і за допомогою повзунка Convergence розмістіть текст у 3D-просторі.

Перетягування повзунка ліворуч переміщує титри на задній план, а праворуч — на передній.



Злиття найкраще помітно при моніторингу на пристрої Apple Vision Pro. При використанні інших засобів моніторингу можна скоригувати злиття тексту у вікні перегляду, змінивши на сторінці Color параметри налаштування Vision для фонового відеошару.

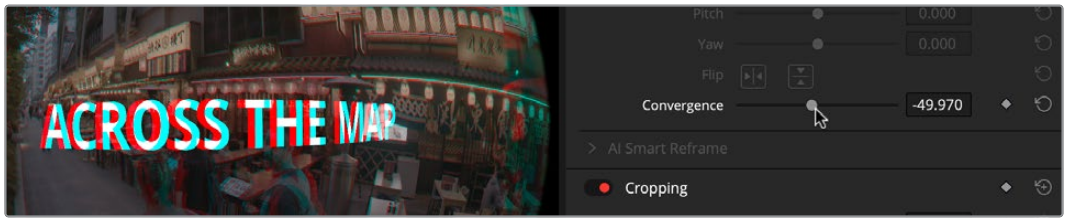
6 Перейдіть на сторінку Color.

7 Відкрийте панель 3D.

Центральний стовпець Vision дозволяє змінити стереоскопічний формат, який використовується для проекції імерсивного матеріалу.

8 Виберіть вкладку Stereo.

9 Для налаштування Out виберіть Anaglyph (Color).



На панелі Inspector сторінки Edit можна легше змінювати параметри генератора тексту за допомогою повзунка Convergence, якщо переглядати їх через анагліфічні (або анахромні) окуляри.

Творчі аспекти

На цьому ранньому етапі існування камери Blackmagic URSA Cine Immersive і виробництва контенту для пристрою Apple Vision Pro кінематографістам радять експериментувати з форматом, щоб визначити найкращі практики, установити нові правила та знайти досі невідомі традиційній індустрії кіно робочі процеси.

Зважаючи на це, уже відзначено деякі аспекти створення імерсивного контенту, які потребують особливої уваги, щоб забезпечити глядачам максимальний ефект занурення та візуальний комфорт.

1 Редагування повинно розпочинатися ще на етапі підготовки до виробництва. Через особливі вимоги до довжини кадрів і рухів, які не мають викликати дискомфорт при перегляді, кінематографістам потрібно записувати сцени з урахуванням кінцевого монтажу.

2 Уникайте або обмежте використання кліпів зі швидким пересуванням камери, оскільки вони можуть дезорієнтувати глядача. Натомість радимо вибирати матеріал, записаний фіксованою камерою, з операторських візочків або з повільним панорамуванням.

3 Незбалансовані зображення або нестандартні ракурси також можуть спричинювати дезорієнтацію. Отже, використовуйте кадри, записані на рівні об'єкта та на зручній відстані для очей (не надто близько).

4 Занурення в нові імерсивні середовища потребує більше часу, ніж перегляд 2D-матеріалу. Тому кадри повинні змінюватися повільніше, щоб глядач мав змогу пристосуватися до нового довкілля.

5 При швидкому переході між кліпами потрібно враховувати, куди дивився глядач у попередніх кадрах. Важливі події не повинні розгортатися в лівій частині кадру, якщо до цього увага глядача була спрямована на праву.

6 Просторове аудіо відіграє велику роль у розкритті сюжету, тому в імерсивному виробництві до нього варто ставитися як до ще одного персонажа.

VFX-композитинг

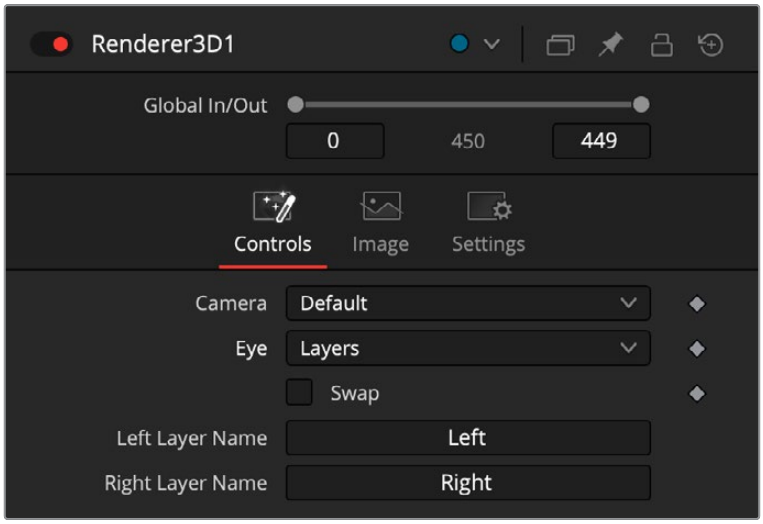
Якщо на панелі Project Settings установлено прапорець Enable Apple Immersive workflow, то спеціально призначені налаштування сторінки Fusion забезпечать бездоганну роботу зі стереоскопічним відео та композитинг відповідно до вимог імерсивного матеріалу. Вузли MediaOut дозволяють створювати кліпи для правого та лівого очей, а інструмент Immersive Patcher — рівнокутну проекцію та композицію.

Вузли MediaOut

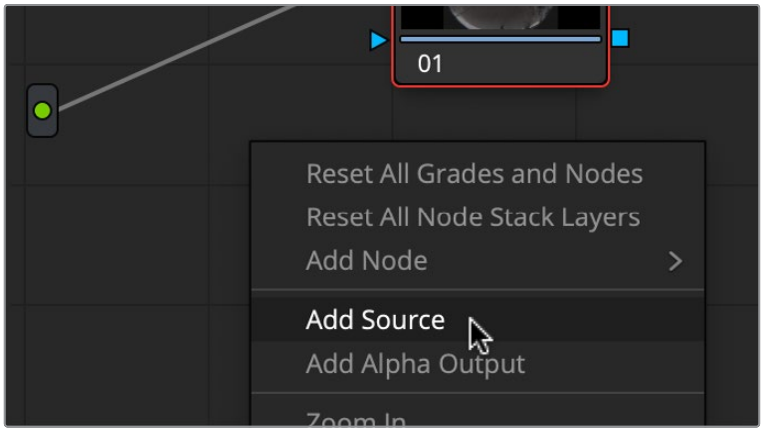
Коли при обробці імерсивного кліпу ви вперше переходите на сторінку Fusion, на графіку Nodes є по два вузли MediaIn і MediaOut.



У секції MediaOut панелі Inspector налаштування Stereo Eye можна використовувати для маршрутизації вихідного сигналу.



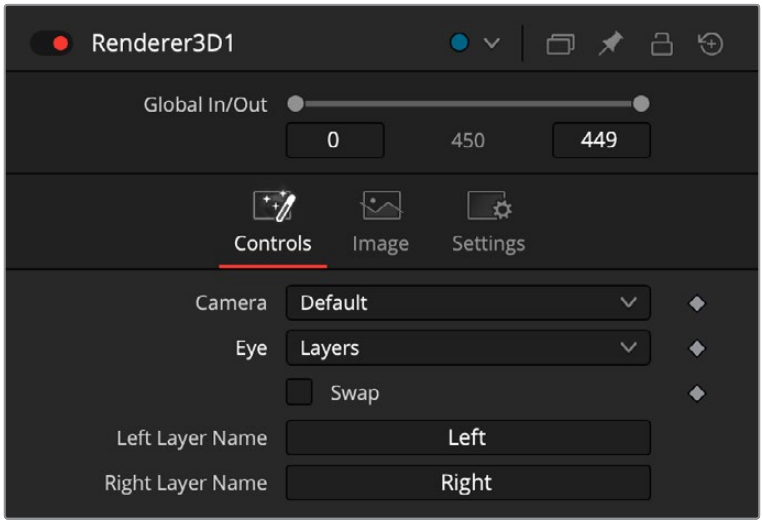
За замовчуванням MediaOut1 налаштовано на зображення для лівого ока, а MediaOut2 — для правого. Така конфігурація дозволяє відразу починати збирання композитного стереоматеріалу й накладання графіки. Завдяки двом вузлам MediaOut дві версії сигналу надходить у редактор вузлів сторінки Color для додаткового редагування зображення. Щоб отримати доступ до сигналу MediaOut2, клацніть правою кнопкою миші в редакторі вузлів і виберіть Add Source.



Коли для налаштування Stereo Eye вибрано Auto, вузол MediaOut об'єднує сигнали та призначає їх відповідному каналу на часовій шкалі. Отже, функціонал для роботи з багатошаровим матеріалом сторінки Fusion дозволяє надсилати лівий і правий канали на один вузол MediaOut.

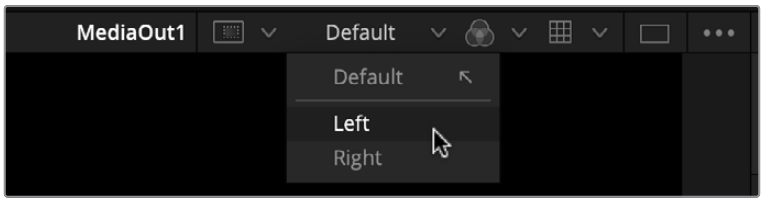
Рендеринг 3D-сцен

Працюючи з 3D- або комп'ютерною графікою на імерсивній часовій шкалі, вузол Renderer3D можна налаштувати під різні очі й обробляти за його допомогою шари лівого та правого каналів.



Це дозволяє створювати складні 3D-титри або цілі сцени з комп'ютерною графікою, які в імерсивних проектах можна відправляти безпосередньо на часову шкалу зі стереоскопічним контентом.

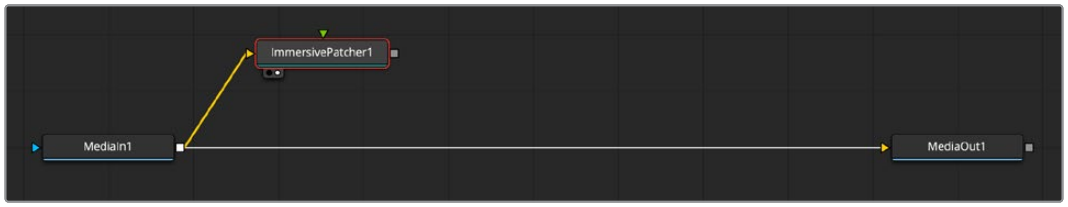
Працюючи у такий спосіб, у верхньому правому кутку вікон перегляду сторінки Fusion можна використовувати розкривне меню для почергового відображення лівого та правого каналів композиції.



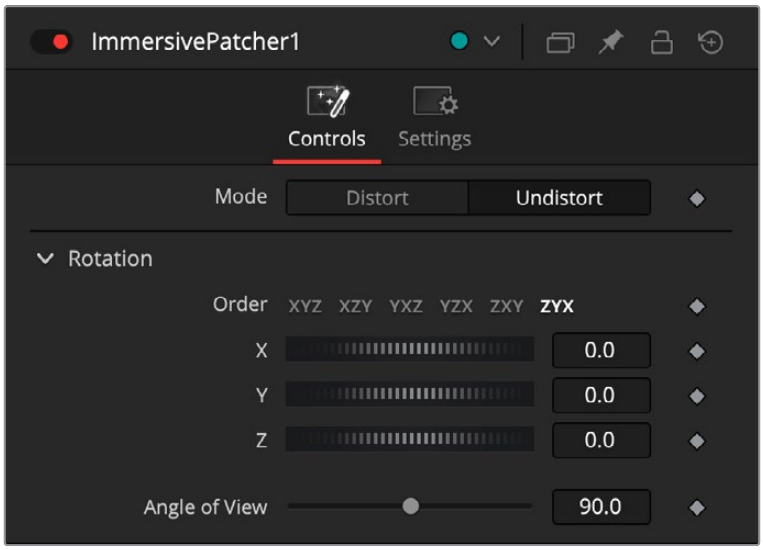
Композитинг із застосуванням імерсивного коректора

Інструмент Immersive Patcher дозволяє тимчасово прибрати спотворення в імерсивному кліпі, щоб спростити відстеження, замальовування та площинний композитинг. Після виконання дії можна скористатися другим інструментом Immersive Patcher для повторного викривлення імерсивного відео та інтеграції скомбінованих елементів у сцену.

1 Додайте вузол Immersive Patcher до вузлової структури.



2 На панелі Inspector клацніть Undistort.



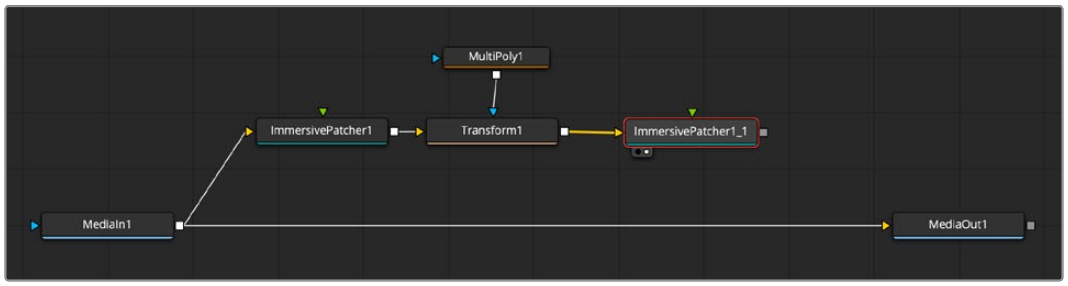
3 У секції Rotation за допомогою повзунків X, Y і Z виберіть частину імерсивного зображення, яку потрібно розпрямити.

4 Виконайте композитинг, наче ви працюєте зі стандартним 2D-відеосигналом. Під час цього процесу застосовують стандартні (й не лише) методи зафарбовування для видалення небажаних об'єктів, вибірки кадру з перетвореннями та багатокутниками для віртуальних декорацій або використання масок, тексту та додаткових зображень чи відеошарів для заміни знаків або екранів.

Після завершення дій, потрібно знову викривити плоске зображення, щоб воно відповідало оригінальній імерсивній деформації.

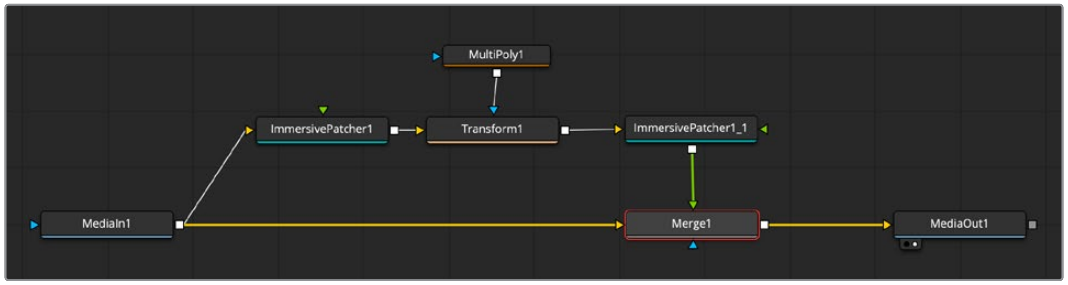
5 Скопіюйте вузол Immersive Patcher, щоб зберегти інформацію про початкове обертання та кут огляду.

6 Вставте вузол Immersive Patcher після композитних вузлів і з'єднайте його з кінцем вузлової структури.



7 На панелі Inspector клацніть Distort.

8 Об'єднайте комбіновану ділянку зображення з оригінальним кліпом для інтеграції ефекту в імерсивну сцену.



При віддаленій співпраці можна залишити сторінку Fusion і продовжити монтаж, мікшування та грейдинг імерсивної часової шкали в додатку DaVinci Resolve.

При виконанні імерсивних обертань створіть для кожного ока EXR-файл у колірному просторі ACES AP0, щоб зберегти максимальну кількість даних для генерації майстер-копії імерсивного відео.

Колірний грейдинг

У цьому розділі міститься інформація про інструменти для роботи з оптичним потоком, кольором і 3D-відео, які можна використовувати для обробки імерсивного матеріалу на сторінці Color додатка DaVinci Resolve.

Через унікальні характеристики імерсивного контенту не всі інструменти можна буде застосовувати на етапі поствиробництва. Деякі з них наразі недоступні, але фахівці вже займаються їх адаптуванням до роботи з імерсивним контентом.

Шумозаглушення

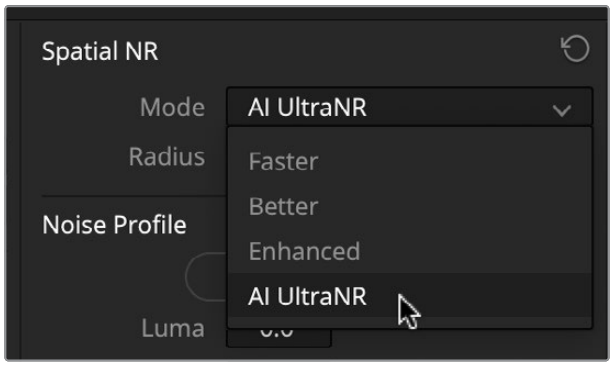
При роботі з імерсивним відео можна використовувати функцію шумозаглушення. На панелі Motion Effects інструмент Spatial NR аналізує шум первинного матеріалу, щоб максимально зменшити його, зберігаючи при цьому деталі оригінального зображення.

1 На сторінці Color відкрийте панель Motion Effects.

Стовпець Spatial NR дозволяє вибрати режим шумозаглушення.

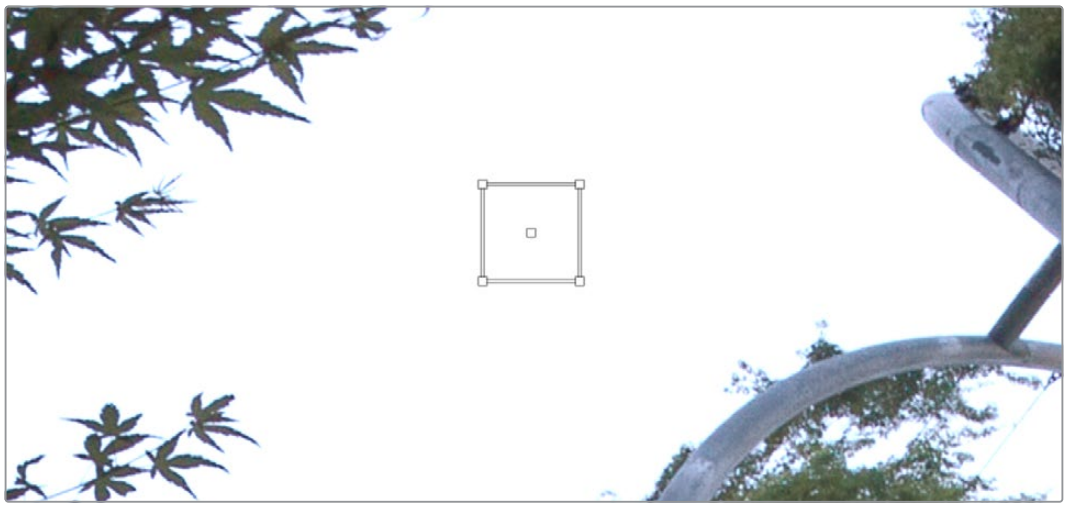
- Режими *Faster*, *Better* і *Enhanced* аналізуватимуть шум у межах заданого радіусу та змінюватимуть параметри виведення звуку залежно від граничних значень. Вони подані в порядку складності обробки.
- *AI UltraNR* — це ШІ-ідентифікатор, який вилучає зразок шуму із сигналу, виконує пошук і перевіряє «найплоскішу» частину кадру на наявність чистого шуму.

2 У секції Spatial NR виберіть режим AI UltraNR.



3 Клацніть Analyze.

У вікні перегляду з'явиться обмежувальний прямокутник, який визначає область із найменшою кількістю пікселів у даному кадрі. За відсутності візуальних орієнтирів дані в цій квадратній рамці ідеально підходять для вилучення цифрового шуму, створеного сенсором камери.



Можна переміщувати та змінювати розмір обмежувального прямокутника у вікні перегляду, проте зазвичай не радимо цього робити, тому що AI UltraNR краще виявляє оптимальні області аналізу, ніж наші очі.

Також можна вручну налаштувати шумовий профіль граничних значень яскравості та колірності, але знову ж таки, AI UltraNR автоматично створює їхній ідеальний мікс для чистого шумозаглушення.

Первинна колірна корекція

Первинна колірна корекція імерсивного матеріалу виконується за тим самим принципом, що й звичайного відеоматеріалу. При використанні колірних і логарифмічних кіл, HDR-палітри та HSL-кривих варто дотримуватися тих же правил щодо нормалізації, балансування та узгодження. Панелі ColorSlice і Color Warper також працюють належним чином.

Вторинна колірна корекція

Вибір ділянок зображення за пікселями, наприклад за допомогою інструмента Qualifier, також підходить для роботи з імерсивним матеріалом.

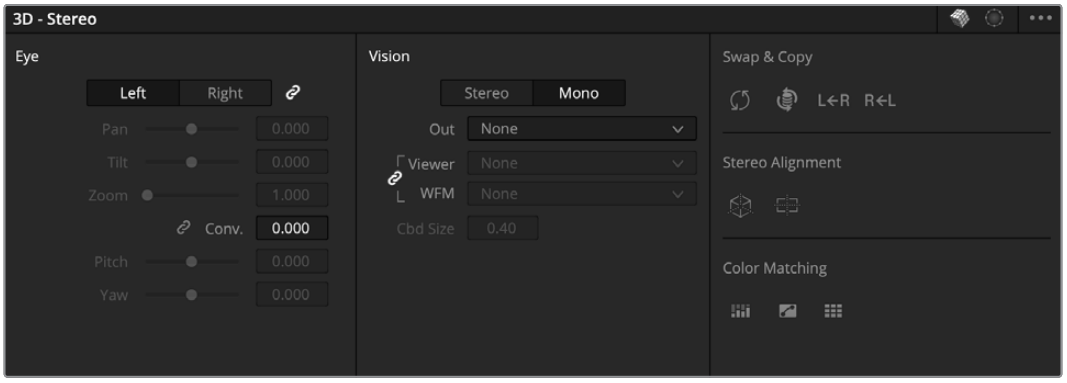
Точно вибрати об'єкт або зону можна лише в плоскому рівнопрямокутному середовищі, тому більшість необхідних для цього інструментів вторинної корекції наразі недоступні. Це також стосується засобів на панелях Window, Tracker і Magic Mask.

Параметри

Для імерсивного контенту не передбачено налаштувань для зміни параметрів (наприклад, масштабування та панорамування).

Панель 3D

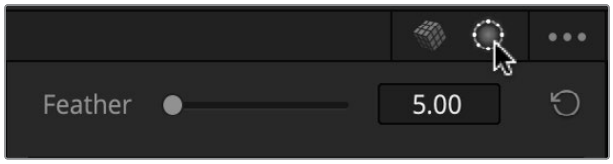
Панель 3D сторінки Color містить налаштування для керування різними 3D-форматами камери та графіки. На відміну від інших стереоскопічних форматів, не потрібно регулювати положення очей, тому що об'єктиви камери Blackmagic URSA Cine Immersive, їхні фокусна відстань і діафрагма є фіксованими.



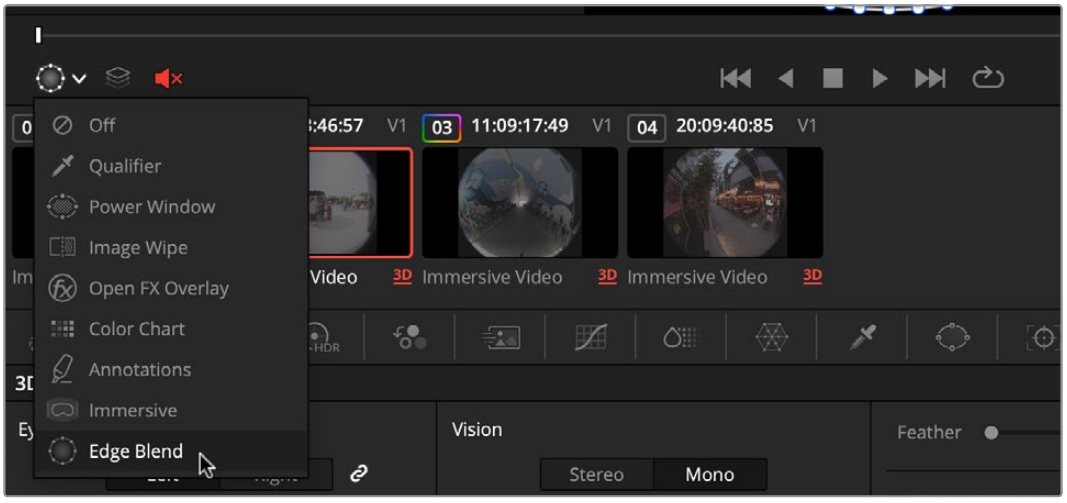
Режим Edge Blend, передбачений виключно для імерсивного контенту, дозволяє налаштовувати та згладжувати рамки імерсивного відео за периметром. За його допомогою можна приховати помітне в кадрі знімальне обладнання й обмежити поле зору глядача. За цією рамкою відображатиметься фон.

1 Відкрийте панель 3D.

2 У верхньому правому кутку змініть режим на Edge Blend.



3 Переконайтеся, що у вікні перегляду для екранних органів керування вибрано Edge Blend.



По периметру рамки зображення з'явитися коло з контрольних точок із закріпленим маркером у центрі.

4 Щоб змінити положення контрольної точки, перетягніть її.



Або клацніть правою кнопкою миші, щоб видалити точку.

5 У правому стовпці панелі 3D є фейдер Feather для згладжування зображення по краях.

6 Установіть прапорець visionOS Effect, щоб дані про нові краї зображення правильно оброблялися при відтворенні відео на пристрої Apple Vision Pro. Якщо прапорець не встановлено, ця інформація зберігатиметься у створеному відеофайлі.

7 У лівому стовпці є вкладки Left і Right для перемикання між лівим і правим оком у вікні перегляду.

Зверніть увагу, що на панелі 3D багато органів керування вимкнено. Налаштування додатку DaVinci Resolve будуть сірими, якщо їх не можна використовувати для роботи над поточним відеоматеріалом, яким у цьому разі є імерсивний контент.

8 У центральному стовпці є вкладки Stereo та Mono, які дозволяють перемикатися між виведенням імерсивного відеокліпу як зображення з одного або двох об'єктивів.

Керування кольором за допомогою ефекту CST

Використання ефекту перетворення колірного простору (CST) у редакторі вузлів сторінки Color дозволить точно призначити кольори та самостійно налаштувати максимальний вихідний рівень яскравості контенту на часовій шкалі (в нітах).

Рекомендований робочий процес із чотирма вузлами CST

1. INPUT CST — з камери на робочий колірний простір.

   Колірна корекція відбувається між цими двома вузлами.

2. OUTPUT CST — з робочого колірного простору на HDR-дисплей комп'ютера Mac.
3. MONITOR CST — з 1000 нт до 250 нт для перегляду HDR-матеріалу.
4. AVP CST — з 250 нт до 108 нт для створення кінцевої версії та моніторингу на пристрої Apple Vision Pro.



Перетворення на колірний простір часової шкали

Спершу потрібно перетворити колірний простір відеосигналу з камери на робочий колірний простір часової шкали. Камера Blackmagic URSA Cine Immersive працює в колірному просторі Blackmagic Design Wide Gamut Gen 4/5 із логарифмічною кривою DaVinci Intermediate.

1. Відкрийте панель Effects.
2. Перетягніть вузол Color Space Transform на пустий вузол (01) у редакторі вузлів.

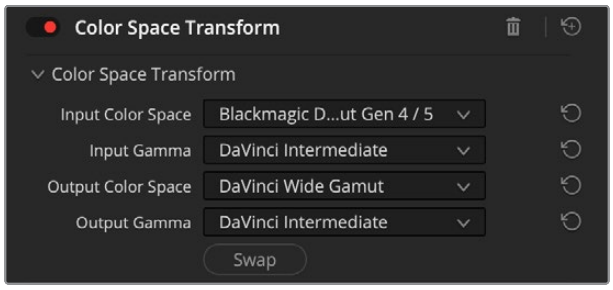
   При перетворенні вхідного колірного простору на колірний простір часової шкали цей вузол потрібно розмістити одним із перших у вузловій структурі, перш ніж розпочинати грейдинг.

   У випадках, коли вхідний колірний простір перетворюється безпосередньо на колірний простір для моніторингу або створення кінцевої версії, цей вузол потрібно розмістити одним з останніх у вузловій структурі після виконання грейдингу.

3. Призначте вузлу 01 ярлик **INPUT CST**.

> **ПРИМІТКА.** У цих інструкціях наявні ярлики вузлів створено для зручного пошуку документації та знімків екрана. Ярлики можна присвоїти будь-яким зручним для вас способом або можна обійтися й без них.

4. Для налаштування Input Color Space виберіть Blackmagic Design Wide Gamut Gen 4/5.
5. Для налаштування Input Gamma виберіть DaVinci Intermediate.
6. Для налаштування Output Color Space виберіть DaVinci Wide Gamut.
7. Для налаштування Output Gamma виберіть DaVinci Intermediate.



8 Створіть новий послідовний вузол (02).

9 Скопіюйте та вставте перший вузол (01) на новий вузол (02).

10 Призначте вузлу 02 ярлик **OUTPUT CST**.

Щоб вузлова структура на часовій шкалі мала однорідний колірний простір, вихідний колірний простір і гама мають бути вхідними на цьому новому послідовному вузлі.

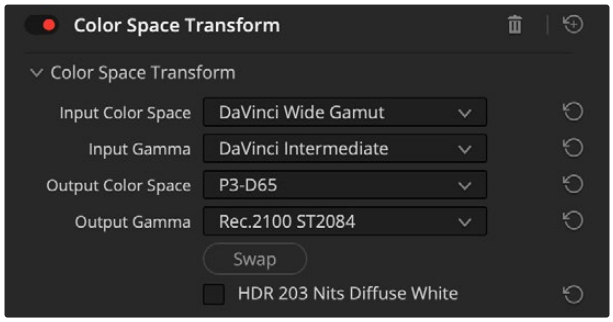
11 Для переміни вхідних і вихідних параметрів натисніть кнопку Swap.

Налаштування Input Color Space та Input Gamma тепер є правильними для цього вузла CST.

12 Для налаштування Output Color Space виберіть P3-D65. Цей колірний простір використовується за замовчуванням для дисплеїв на комп'ютерах Mac.

13 Для налаштування Output Gamma виберіть колірний простір Rec.2100 ST2084, який є стандартом для HDR-сигналу яскравістю 1000 нт.

Увесь грейдинг відбуватиметься між цими двома вузлами. Оскільки для колірної корекції використовується вузлова структура, можна також працювати за межами периметра керування кольором часової шкали. Це дозволить багатьом колористам виконувати спеціальні операції грейдингу та застосовувати ефекти або елементи третіх сторін.



Профілі ICC

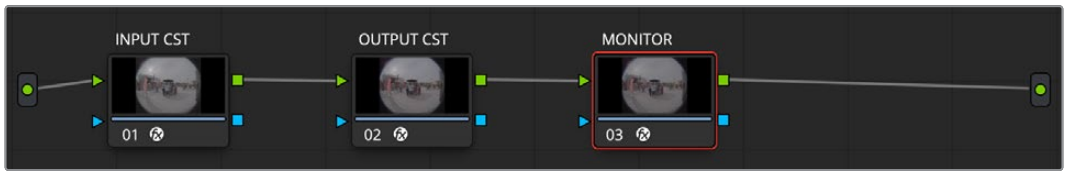
Щоб забезпечити правильне відображення сигналу з камери Blackmagic URSA Cine Immersive на дисплеї комп'ютера Mac, відкрийте «Системні параметри» > «Дисплеї» та переконайтеся, що для налаштування «Колірний профіль» вибрано Apple XDR Display (P3-1600 ніт).

Перетворення колірного простору для моніторингу грейдингу

Можна застосувати додаткові вузли CST, щоб тимчасово обмежити максимальні значення вихідного сигналу в нітах для моніторингу контенту під час грейдингу.

1 Створіть новий послідовний вузол (03) після вузла OUTPUT CST (02).

2 Призначте вузлу 03 ярлик **MONITOR**.



3 Додайте ефект Color Space Transform до вузла MONITOR (03).

Значення вхідного сигналу мають відповідати значенням вихідного сигналу попереднього вузла CST.

4 Для налаштування Input Color Space виберіть P3-D65.

5 Для налаштування Input Gamma виберіть Rec.2100 ST2084.

Цей вузол CST призначено лише для того, щоб задати зображенню нижчий рівень яскравості в нітах, тобто параметри вхідних і вихідних сигналів можуть залишатися однаковими.

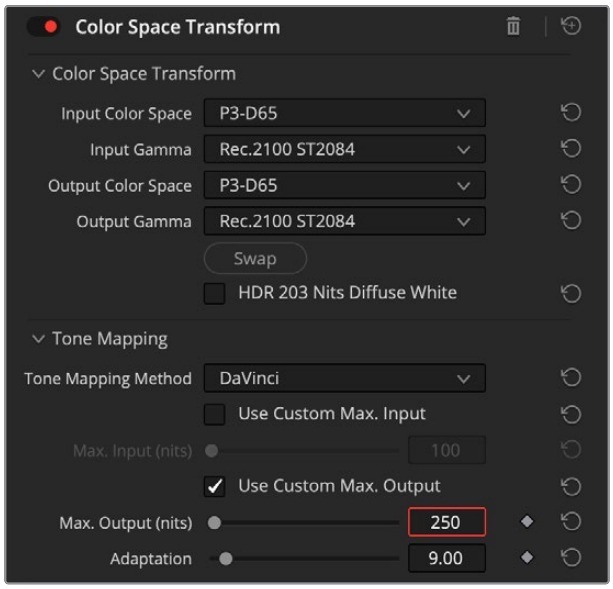
6 Для налаштування Output Color Space виберіть P3-D65.

7 Для налаштування Output Gamma виберіть Rec.2100 ST2084.

ПРИМІТКА. У деяких випадках, якщо вхідний і вихідний сигнали однакові (і призначення не відбувається), вузлу CST не потрібно задавати колірний простір або гаму. У цьому разі це необхідно зробити, тому що на встановлення настроюваної яскравості в нітах впливає значення та форма логарифмічної кривої сигналу.

8 Установіть прапорець Use Custom Max. Output.

9 Для налаштування Max. Output (nits) виберіть **250**.



Сигнал 1000 нт тепер перетворюватиметься на 250 нт згідно налаштуванню Tone Mapping Method > DaVinci.

Це дозволить отримати зображення, яке максимально відповідає сигналу, створеному пристроєм Apple Vision Pro, на HDR-моніторі для грейдингу, наприклад Flanders XMP310 або Sony BVM-HX310.

Проте для моніторингу проекту на пристрої Apple Vision Pro необхідно виконати ще один крок зіставлення тонів.

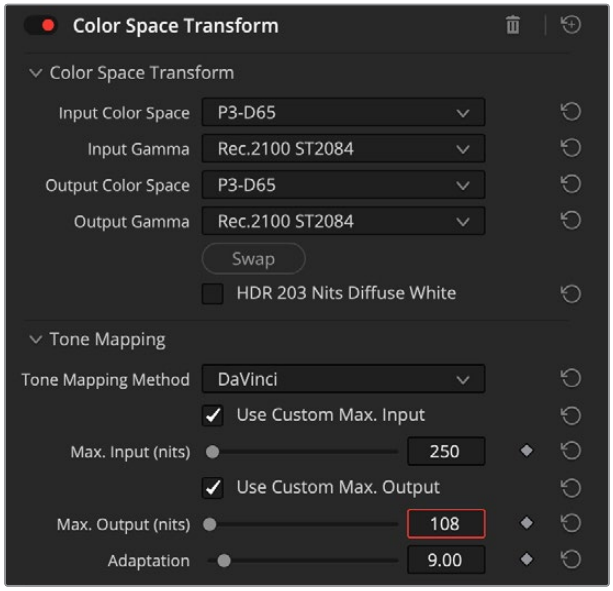
Перетворення колірного простору для моніторингу на пристрої Apple Vision Pro

Останній вузол CST може слугувати для моніторингу в реальному часі, попереднього перегляду та підготовки кінцевої версії на пристрої Apple Vision Pro.

1. Створіть новий послідовний вузол (04) після вузла MONITOR (03).
2. Скопіюйте та вставте вузол MONITOR (03) на вузол (04). Вхідні та вихідні параметри будуть однаковими, зміниться лише значення, що показує ніти.
3. Призначте вузлу 04 ярлик **AVP**.



4. Установіть прапорець Use Custom Max. Input.
5. Для налаштування Max. Input (nits) задайте число **250**, щоб правильно відобразити сигнал із вузла MONITOR.
6. Для налаштування Max. Output (nits) задайте число **108**, яке є максимальним значенням у нітах для моніторингу й експорту матеріалу на пристрій Apple Vision Pro.



Тепер можна починати грейдинг, вимикаючи деякі вузли CST відповідно до потреб робочого процесу. Наприклад, щоб під час першого етапу грейдингу на спеціальний HDR-монітор правильно виводився сигнал 250 ніт, вимкніть вузол AVP (04). Для перегляду матеріалу на пристрої Apple Vision Pro або для експорту кінцевої версії знову ввімкніть вузол AVP (04).

Чи можна розміщувати вузли CST в редакторі вузлів часової шкали?

Ні. Наразі зіставлення тонів кліпу залежить від точного застосування налаштування Color Space Transform > Adaptation, що має відбуватися окремо для кожного кліпу.

Аудіо

Пристрій Apple Vision Pro та додаток Davinci Resolve підтримують просторовий і амбісонічний звук, а також аудіоформати AVP (Apple Vision Pro) та AIA (Apple Immersive Audio).

Багато аспектів постобробки звичайного та просторового аудіо є однаковими. Одна з основних відмінностей полягає в тому, що замість моніторингу звуку через професійні динаміки звукорежисери мають використовувати навушники або AirPods, оскільки вони забезпечують справжнє просторове занурення та покращене відтворення атмосфери довкілля, у якому перебуває аудиторія.

Тоді як у 2D-міксах більша частина діалогу зводиться до центральних каналів, просторове аудіо потрібно зводити до області, з якої надходить звук. При зведенні аудіодоріжок необхідно виконувати приглушення, еквалізацію та балансування, щоб звук не перевантажував слухачів і не збивав їх із пантелику.

Імерсивні просторові елементи можна використовувати зі звичайною стереодоріжкою, що дозволяє зберегти незмінними певні звуки, незалежно від того, куди повернута голова. Це сприяє чіткій і легкій інтерпретації недієгетичного звуку, зокрема музичної партитури та закадрового голосу.

Звуки довкілля зазвичай передаються тихіше в просторовому аудіо, тому часто потребують посилення при зведенні звуку, щоб їх було чутно. Проте цього не варто досягати шляхом дезорієнтації слухача.

На створення відчуття занурення значною мірою впливає ефект реверберації, який слухачі відчувають щодня. Звуки певного довкілля, відбиваючись від поверхонь навколо слухача, призводять до поступово згасаючого відлуння. Власне тому при роботі з просторовим аудіо радимо використовувати реверберацію для отримання ефекту повного занурення.

Імерсивний аудіоформат Apple (AIA) є контейнером для наведених нижче форматів.

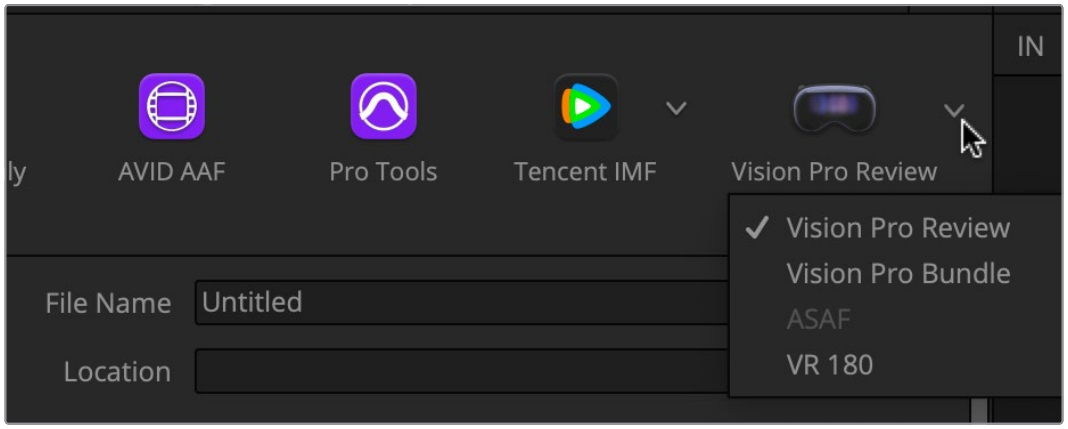
- **Просторовий аудіоформат Apple (ASAF)** — це формат необробленого звуку.
- **Позиційний аудіокодек Apple (APAC)** — це кодований формат .mp4, який можна відтворювати в навушниках Airpods або з AIVU-файлу на пристрої Apple Vision Pro.

Просторовий аудіоформат Apple (ASAF)

Просторовий аудіоформат Apple (ASAF) — це імерсивний аудіоформат, створений насамперед для використання віртуальної та доповненої реальностей у пристрої Apple Vision Pro. ASAF, призначений виключно для macOS, працює з додатками ProTools (через плагіни) та DaVinci Resolve (вбудований). Цей формат має передові технічні характеристики, зокрема підтримку джерел амбісонічного звуку, які називають сценами, і можливість маршрутизації за допомогою призначених шин. Оскільки мікшування є віртуальним, тобто без фізичної шини, користувачі можуть вибирати різні формати моніторингу, зокрема бінауральний із відстеженням руху голови в навушниках Apple AirPods. Панорамування виконується за допомогою таких наявних інструментів DaVinci Resolve, як панорамування діалогу, автоматизація, SpaceView, а також засобів панорамування у вікні перегляду та допоміжних елементів керування на панелі Inspector. Це забезпечує точне просторове розміщення джерел і автоматичне відстеження в рівнопрямокутному відео. ASAF-мікси можна експортувати у форматі WAV або MP4 (з їхніми метаданими), перейшовши до налаштування Immersive Audio > Export Master File, хоча наразі їх неможливо імпортувати або відтворювати в системі DaVinci Resolve.

Експорт кінцевої версії

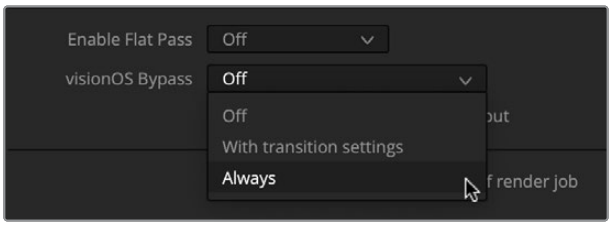
На сторінці Deliver є два налаштування рендерингу, які дозволяють отримати кінцеву версію файлів Apple Vision Pro для перегляду й експорту.



- **Vision Pro Review** забезпечує створення невеликого пакету MV-HEVC для попереднього перегляду. Доступ до готового файлу пристрій Apple Vision Pro отримує через мережу Wi-Fi, а для його перегляду потрібно завантажити з App Store додаток Apple Immersive Video Utility (AIVU).
- **Vision Pro Bundle** дозволяє створювати великий ProRes-файл, який можна заархівувати або обробити в додатку Apple Compressor, щоб підготувати до відтворення на Apple TV.
- **ASAF** забезпечує рендеринг аудіоміксу ASAF у форматі WAV або MP4.
- **VR180** дозволяє створювати стереоскопічний відеофайл, сумісний із форматом VR180.

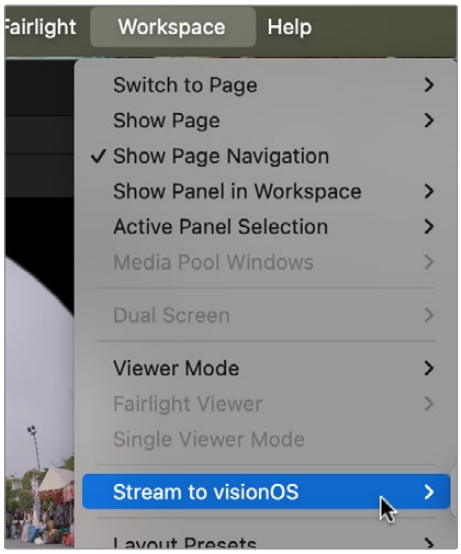
Пропуск переходів

Якщо перехід додано до імерсивної часової шкали, його буде попередньо підготовлено та перетворено на єдину доріжку для відтворення. В XML-файлі, згенерованому для імерсивного контенту, описано процес переключення між цими створеними потоками (від матеріалу до попередньо підготовленого переходу й знову назад до матеріалу). У секції Advanced Settings панелі Render Settings для налаштування visionOS Bypass виберіть Always, якщо потрібно пропустити переходи; With transition settings, щоб використовувати їх відповідно до їхнього статусу на часовій шкалі; або Off, аби застосовувати в міру їх появи на часовій шкалі.



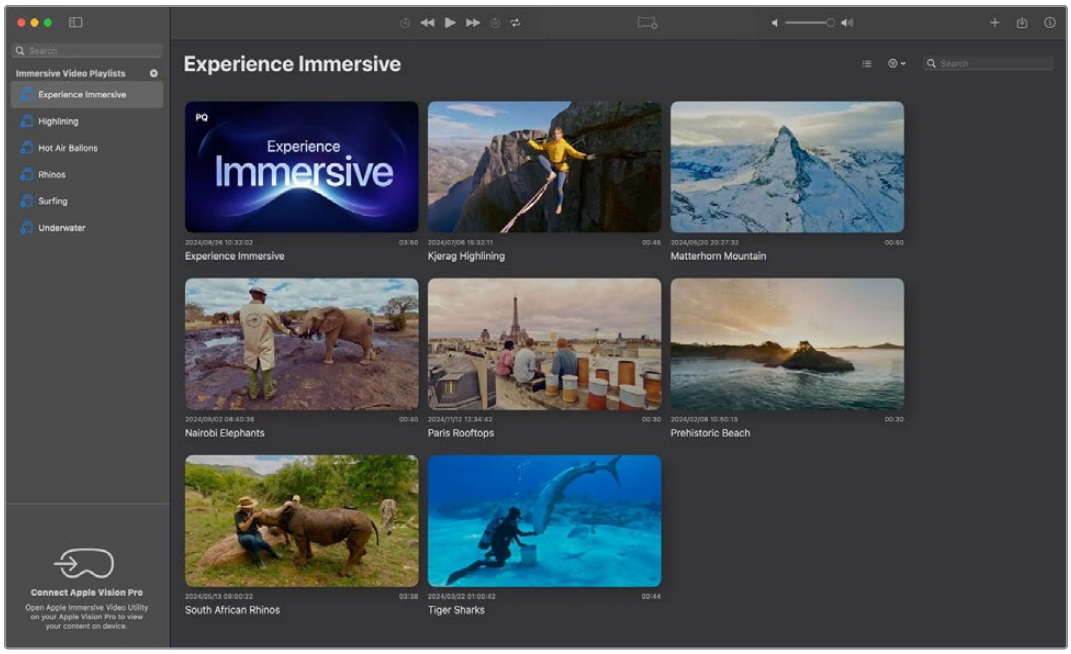
Стрімінг на пристрій Apple Vision Pro

Стрімінг на пристрій Apple Vision Pro можна виконувати зі сторінок Edit, Color і Deliver додатка DaVinci Resolve. Для цього потрібно перейти в меню Workspace і вибрати Stream to visionOS, а потім — свій AVP-пристрій. Для плавного стрімінгу радимо використовувати швидку бездротову мережу (мінімум Wi-Fi 5, бажано Wi-Fi 6).



Apple Immersive Video Utility

Apple Immersive Video Utility — це спеціальний додаток для macOS та visionOS, призначений для створення, контролю та відтворення імерсивного відео Apple. Розроблений для професійних кінематографістів і творців імерсивного медіаконтенту, цей додаток дозволяє користувачам імпортувати, організовувати, пакувати та переглядати стереоскопічне 8K-відео з полем зору 180 градусів і просторовим аудіо. Він підтримує створення AIVU-файлів (Apple Immersive Video Universal), що дає змогу перевіряти та змінювати метадані для поствиробництва. Користувачі можуть створювати плейлисти, відшукувати та сортувати записані кліпи у своїх бібліотеках і навіть стрімити контент безпосередньо на один або кілька пристроїв Apple Vision Pro.



Інформація для розробників

Camera Control REST API

If you are a software or hardware developer you can build custom applications or leverage ready to use tools such as REST client or Postman to seamlessly control and interact with your compatible Blackmagic camera using Camera Control REST API. This API enables you to perform a wide range of operations, such as starting or stopping recordings, accessing disk information and much more. Whether you're developing a custom application tailored to your specific needs or utilizing existing tools, this API empowers you to unlock the full potential of your Blackmagic camera with ease. We look forward to seeing what you come up with!

NOTE It's important to mention that controlling Blackmagic cameras via REST API relies on the web manager being enabled on each compatible Blackmagic camera. Enable the web media manager in the Blackmagic Camera Setup 'network access' settings for each camera you are controlling.

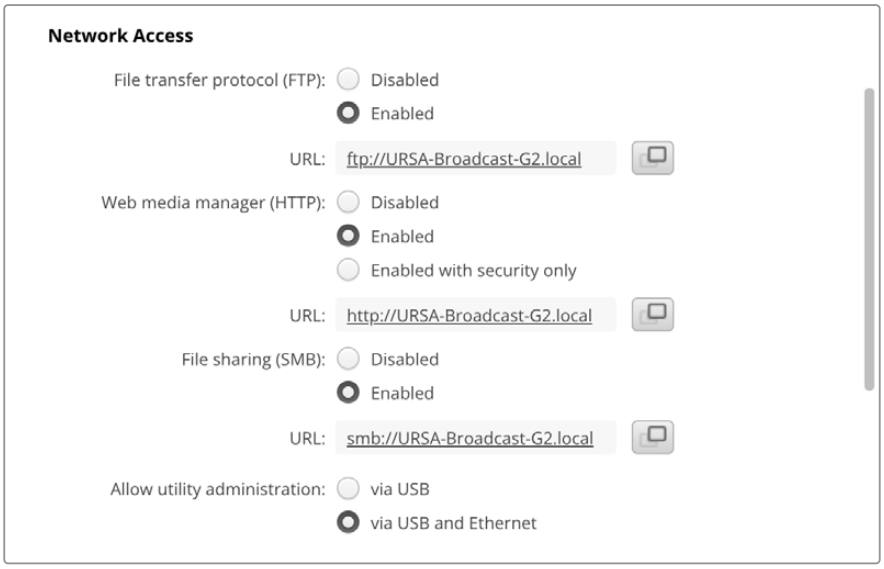
The following Blackmagic cameras are compatible with Camera Control REST API:

- Blackmagic URSA Cine 12K LF
- Blackmagic URSA Cine 17K 65
- Blackmagic URSA Cine Immersive
- Blackmagic PYXIS 6K
- Blackmagic Cinema Camera 6K
- Blackmagic URSA Broadcast G2
- Blackmagic Micro Studio Camera 4K G2
- Blackmagic Studio Camera 4K Plus
- Blackmagic Studio Camera 4K Pro
- Blackmagic Studio Camera 6K Pro
- Blackmagic Studio Camera 4K Plus G2
- Blackmagic Studio Camera 4K Pro G2

Sending API Commands

To send an API command to your camera from a third party application such as Postman, add /control/api/v1/ to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/api/v1/

You can find the Web media manager URL and IP address information in Blackmagic Camera Setup.



The Web media manager URL in Blackmagic Camera Setup

Downloading API's from your Camera

You can download REST API YAML documentation from your camera by adding /control/documentation.html to the end of the camera's Web media manager URL or IP address. For example, https://ursa-broadcast-g2.local/control/documentation.html

NOTE It's worth noting that changing the camera name in Blackmagic Camera Setup will also change the camera's Web media manager URL.

Livestream Control API

API for controlling Livestreams on Blackmagic Design products.

GET /livestreams/0

Get the livestream's current status.

Response

200 - Livestream's current status.

The response is JSON.

Name	Type	Description
status (required)	string	Possible values are: Idle, Connecting, Streaming, Flushing, Interrupted.
bitrate (required)	integer	Current bitrate (bps).
effectiveVideoFormat (required)	string	Effective video format for the livestream, serialised as a string.
duration	integer	Current stream duration in seconds. Absent if livestream is idle.
cache	integer	Current stream cache usage percentage.

GET /livestreams/0/start

Determine if the livestream is active.

Response

200 - Livestream active status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is active.

PUT /livestreams/0/start

Start the livestream.

Response

204 - Livestream started.

GET /livestreams/0/stop

Determine if the livestream is inactive.

Response

200 - Livestream inactive status.

The response is JSON.

Name	Type	Description
	boolean	True when the livestream is inactive.

PUT /livestreams/0/stop

Stop the livestream.

Response

204 - Livestream stopped.

GET /livestreams/0/activePlatform

Get the currently selected platform configuration for the livestream.

Response

200 - Livestream active platform configuration.

The response is JSON.

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

PUT /livestreams/0/activePlatform

Set the currently selected platform configuration for the livestream.

Parameters

Livestream's current active platform configuration.

Name	Type	Description
platform (required)	string	Platform name.
server (required)	string	The platform's server name, or "Custom" when the URL is customizable.
key	string	Stream key. Assumed to be empty if missing.
passphrase	string	Passphrase. Only included for SRT streams.
quality (required)	string	Quality level name.
url	string	Livestream destination. Only included when URL is customizable.

Response

204 - Livestream active platform configuration updated.

400 - Bad Request

GET /livestreams/platforms

Get the list of available platforms.

Response

200 - List of available platforms.

The response is JSON.

Name	Type	Description
	array	List of available platforms names.
[i]	string	Platform name.

GET /livestreams/platforms/{platformName}

Get the service configuration for a platform.

Parameters

Name	Type	Description
{platformName} (required)	string	Name of the platform.

Response

200 - Service configuration for specified platform.

The response is JSON.

Livestream platform service configuration.

Name	Type	Description
platform (required)	string	Corresponding platform name.
key	string	Default stream key.
servers (required)	array	List of server configurations.
servers[i]	object	Server configuration.
servers[i].server (required)	string	Server name.
servers[i].url (required)	string	Livestream destination.
servers[i].srtExtensions	array	Miscellaneous tags used for SRT livestreams.
servers[i].srtExtensions[i]	object	Dictionary object mapping SRT tag strings to values.
servers[i].srtExtensions[i][{key}]	string	SRT tag value.
servers[i].group	string	Logical grouping of the server.
profiles (required)	array	List of profile configurations.
profiles[i]	object	Quality configuration.
profiles[i].profile (required)	string	Quality level name.
profiles[i].configs (required)	array	List of video format configurations.
profiles[i].configs[i]	object	Video format configuration for profiles.
profiles[i].configs[i].resolution (required)	string	Video format serialised as a string.
profiles[i].configs[i].fps (required)	string	Frames per second.
profiles[i].configs[i].bitrate (required)	integer	Pixel bitrate (bps).
profiles[i].configs[i].audioBitrate	integer	Audio bitrate (bps).
profiles[i].configs[i].keyFrameInterval	integer	How often a key frame is sent, in seconds.
profiles[i].configs[i].videoCodecs	array	Supported video encoding algorithm/s.

Name	Type	Description
profiles[i].configs[i].videoCodecs[i]	string	Video encoding algorithm. Possible values are: H264, H265.
profiles[i].lowLatency (required)	boolean	If true, fewer frames will be buffered in the livestream.
defaultProfile	string	Quality level name.
credentials	object	Credientials used for RTMP streams.
credentials.username (required)	string	The username part of the creditials. Only used for RTMP streams.
credentials.password (required)	string	Used for RTMP streams, also used as Passphrase for SRT streams.
customizableUrlEnabled	boolean	True when the server URL is customizable.

400 - Bad Request

GET /livestreams/customPlatforms

Get a list of custom platform files.

Response

200 - List of custom platform files.

The response is JSON.

Name	Type	Description
	array	List of custom platform file names.
[i]	string	Custom platform file name.

DELETE /livestreams/customPlatforms

Remove all custom configuration files.

Response

204 - All custom configuration files removed.

GET /livestreams/customPlatforms/{filename}

Get a custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to get.

Response

200 - Custom platform file.

The response is XML.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

404 - Not Found

PUT /livestreams/customPlatforms/{filename}

Update a custom platform file if it exists, if not, create a new file with the given file name.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to update/create.

Blackmagic streaming XML file format.

Name	Type	Description
	object	Blackmagic streaming XML file format.

Response

204 - Custom platform file created or updated.

400 - Bad Request

DELETE /livestreams/customPlatforms/{filename}

Remove the given custom platform file.

Parameters

Name	Type	Description
{filename} (required)	string	Name of the file to be removed.

Response

204 - Custom platform file removed.

404 - Not Found

Clips Control API

API for listing clips on disk.

GET /clips

Get the list of clips on the active disk.

Response

200 - List of clips on the active disk.

The response is JSON.

List of media clips.

Name	Type	Description
clips (required)	array	
clips[i]	object	Media clip.
clips[i].clipUniqueId (required)	integer	Unique ID used to identify this clip.
clips[i].filePath	string	Path to the file relative to the root of a mount.
clips[i].fileSize	integer	Size of file on disk in bytes.
clips[i].codecFormat	object	
clips[i].codecFormat.codec	string	Currently selected codec.

Name	Type	Description
clips[i].codecFormat.container	string	Multimedia container format.
clips[i].videoFormat	object	Video format configuration.
clips[i].videoFormat.name (required)	string	Video format serialised as a string.
clips[i].videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
clips[i].videoFormat.height	number	Height dimension of video format.
clips[i].videoFormat.width	number	Width dimension of video format.
clips[i].videoFormat.interlaced	boolean	Is the display format interlaced?
clips[i].startTimecode	string	Start timecode of the clip serialised as string.
clips[i].durationTimecode	string	Duration of the clip in timecode format serialised as string.
clips[i].frameCount	integer	Number of frames in clip; duration of the clip in frames.

404 - There is no active disk.

Media Pool Control API

API to manage media pool and handle uploads and project data.

GET /cloud/projects

List all projects within the media pool.

Response

200 - Successfully retrieved the list of all projects.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].libraryID	string	
[i].name	string	
[i].private	boolean	
[i].shared	boolean	
[i].clips	array	List of clips associated with the project.
[i].clips[i]	string	
[i].status	object	
[i].status.numClipsRequested	integer	
[i].status.numClipsComplete	integer	
[i].status.uploadPercent	integer	Percentage of upload completion.
[i].status.numClipsPaused	integer	
[i].status.outOfSpace	boolean	
[i].status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
[i].status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/active

Retrieve data of the actively uploading project.

Response

200 - Successfully retrieved the active project's data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

GET /cloud/projects/{projectID}

Retrieve specific project data by project ID.

Parameters

Name	Type	Description
{projectID} (required)	integer	Unique identifier of the project.

Response

200 - Successfully retrieved the project data.

The response is JSON.

Name	Type	Description
libraryID	string	
name	string	
private	boolean	
shared	boolean	
clips	array	List of clips associated with the project.
clips[i]	string	
status	object	
status.numClipsRequested	integer	
status.numClipsComplete	integer	
status.uploadPercent	integer	Percentage of upload completion.
status.numClipsPaused	integer	

Name	Type	Description
status.outOfSpace	boolean	
status.secsRemaining	integer	Estimated seconds remaining until upload is completed.
status.currentByteRate	integer	Current byte rate of the upload process.

404 - Project not found.

GET /cloud/clips

List all clips within the media pool.

Response

200 - Successfully retrieved the list of all clips.

The response is JSON.

Name	Type	Description
	array	
[i]	string	

GET /cloud/clips/activeUploading

Retrieve data of actively uploading clips.

Response

200 - Successfully retrieved the list of actively uploading clips.

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].path	string	
[i].projectID	integer	
[i].status	object	
[i].status.projectID	integer	
[i].status.outOfSpace	boolean	
[i].status.proxyExtension	string	
[i].status.growingFile	boolean	
[i].status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
[i].status.originalClipTotalSize	integer	
[i].status.proxyClipTotalSize	integer	
[i].status.originalClipCompletedSize	integer	
[i].status.proxyClipCompletedSize	integer	
[i].status.secsRemaining	integer	

GET /cloud/clips/{deviceName}/{path}

Retrieve specific clip data by device and path.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored.
{path} (required)	string	Path to the clip.

Response

200 - Successfully retrieved the clip data.

The response is JSON.

Name	Type	Description
path	string	
projectID	integer	
status	object	
status.projectID	integer	
status.outOfSpace	boolean	
status.proxyExtension	string	
status.growingFile	boolean	
status.originalUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.proxyUploadState	string	Possible values are: Unqueued, Paused, Queued, Uploading, Uploaded, Failed.
status.originalClipTotalSize	integer	
status.proxyClipTotalSize	integer	
status.originalClipCompletedSize	integer	
status.proxyClipCompletedSize	integer	
status.secsRemaining	integer	

404 - Clip not found.

Monitoring Control API

API for monitoring and controlling display settings in video equipment.

GET /monitoring/display

Retrieve a list of all display names.

Response

200 - Returns a list of display names.

The response is JSON.

Name	Type	Description
displays	array	List of display names available.
displays[i]	string	

GET /monitoring/{displayName}/cleanFeed

Get the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/cleanFeed

Set the clean feed enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Clean feed enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/displayLUT

Get the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

400 - Invalid display name.

404 - Display name not found.

PUT /monitoring/{displayName}/displayLUT

Set the display LUT enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Display LUT enabled/disabled successfully.

400 - Invalid input.

422 - Unprocessable Entity - Unable to process the contained instructions.

GET /monitoring/{displayName}/zebra

Get the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/zebra

Set the zebra enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Zebra enabled/disabled successfully.

400 - Invalid input.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/focusAssist

Get the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display name not found.

PUT /monitoring/{displayName}/focusAssist

Set the focus assist enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
mode	string	Mode of focus assist, e.g., ‘Peak’ or ‘ColoredLines’. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/focusAssist

Get the focus assist settings.

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Mode of focus assist, e.g., ‘Peak’ or ‘ColoredLines’. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

404 - Display name not found.

PUT /monitoring/focusAssist

Set the focus assist settings.

Parameters

Name	Type	Description
mode	string	Mode of focus assist, e.g., 'Peak' or 'ColoredLines'. Possible values are: Peak, ColoredLines.
color	string	Color of the focus assist highlight. Possible values are: Red, Green, Blue, White, Black.
intensity	integer	Intensity of the focus assist highlight (0-100).

Response

204 - Focus assist settings updated successfully.

400 - Invalid input or configuration.

422 - Unable to process the contained instructions.

GET /monitoring/{displayName}/frameGuide

Get the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame guide enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGuide

Set the frame guide enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame guide state updated successfully.

422 - Unable to update the frame guide state.

GET /monitoring/frameGuideRatio

Get the current frame guide ratio.

Response

200 - Returns the current frame guide ratio.

The response is JSON.

Name	Type	Description
ratio	string	The frame guide ratio.

PUT /monitoring/frameGuideRatio

Set the frame guide ratio.

Parameters

Name	Type	Description
ratio	string	The frame guide ratio.

Response

204 - Frame guide ratio updated successfully.

422 - Unable to update the frame guide ratio.

GET /monitoring/frameGuideRatio/presets

Get the presets for frame guide ratios.

Response

200 - Returns a list of preset frame guide ratios.

The response is JSON.

Name	Type	Description
presets	array	
presets[i]	string	A frame guide ratio.

GET /monitoring/{displayName}/frameGrids

Get the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the frame grids enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/frameGrids

Set the frame grids enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Frame grids state updated successfully.

422 - Unable to update the frame grids state.

GET /monitoring/frameGrids

Get the global frame grids settings.

Response

200 - Returns the current frame grids settings.

The response is JSON.

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

PUT /monitoring/frameGrids

Set the global frame grids settings.

Parameters

Name	Type	Description
frameGrids	array	List of frame grids enabled.
frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

Response

204 - Frame grids settings updated successfully.

400 - Invalid input, check the number of frame grids or values.

422 - Unable to update the frame grids settings.

GET /monitoring/{displayName}/safeArea

Get the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the safe area enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/safeArea

Set the safe area enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - Safe area state updated successfully.

422 - Unable to update the safe area state.

GET /monitoring/safeAreaPercent

Get the current safe area percentage.

Response

200 - Returns the current safe area percentage.

The response is JSON.

Name	Type	Description
percent	integer	Safe area coverage percentage.

PUT /monitoring/safeAreaPercent

Set the safe area percentage.

Parameters

Name	Type	Description
percent	integer	Safe area coverage percentage to set.

Response

204 - Safe area percentage updated successfully.

400 - Invalid percentage value.

422 - Unable to update the safe area percentage.

GET /monitoring/{displayName}/falseColor

Get the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Response

200 - Returns the false color enable state.

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

404 - Display not found.

PUT /monitoring/{displayName}/falseColor

Set the false color enable state for a specific display.

Parameters

Name	Type	Description
{displayName} (required)	string	Name of the display. Obtainable from /monitoring/display which returns a list of displayNames.

Name	Type	Description
enabled	boolean	Indicates if the feature is enabled.

Response

204 - False color state updated successfully.

422 - Unable to update the false color state.

Event Control API

API For working with built-in websocket.

GET /event/list

Get the list of events that can be subscribed to using the websocket API.

Response

200 - Websocket API events list.

The response is JSON.

Name	Type	Description
events	array	List of events that can be subscribed to using the websocket API.
events[i]	string	

System Control API

API for controlling the System Modes on Blackmagic Design products.

GET /system

Get device system information.

Response

200 - System summary.

The response is JSON.

The properties will be populated only with the values that are supported/implemented by the device in use.

Name	Type	Description
codecFormat	object	Codec format configuration.
codecFormat.codec	string	Codec serialised as string.
codecFormat.container	string	Multimedia container format.
videoFormat	object	Video format configuration.
videoFormat.name (required)	string	Video format serialised as a string.
videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
videoFormat.height	number	Height dimension of video format.
videoFormat.width	number	Width dimension of video format.
videoFormat.interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/product

Get device product information.

Response

200 - Device product information.

The response is JSON.

Product information.

Name	Type	Description
deviceName	string	Name of device as displayed in Setup.
productName	string	Device's product name.
softwareVersion	string	Software version running on device.

501 - This functionality is not implemented for the device in use.

GET /system/supportedCodecFormats

Get the list of supported codecs.

Response

200 - List of supported codec formats.

The response is JSON.

Name	Type	Description
codecs	array	
codecs[i]	object	Codec format configuration.
codecs[i].codec	string	Codec serialised as string.
codecs[i].container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

GET /system/codecFormat

Get the currently selected codec.

Response

200 - Current codec format.

The response is JSON.

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

501 - This functionality is not implemented for the device in use.

PUT /system/codecFormat

Update the system codec.

Parameters

Codec format configuration.

Name	Type	Description
codec	string	Codec serialised as string.
container	string	Multimedia container format.

Response

204 - The codec updated successfully.

400 - The specified codec format is unsupported.

501 - This functionality is not implemented for the device in use.

GET /system/videoFormat

Get the currently selected video format.

Response

200 - Current system video format.

The response is JSON.

Video format configuration.

Name	Type	Description
name (required)	string	Video format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height	number	Height dimension of video format.
width	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

PUT /system/videoFormat

Set the system video format.

Parameters

This parameter can be one of the following types:

Name	Type	Description
name (required)	string	Video format serialised as a string.

Name	Type	Description
frameRate (required)	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
height (required)	number	Height dimension of video format.
width (required)	number	Width dimension of video format.
interlaced	boolean	Is the display format interlaced?

Response

204 - The video format updated successfully.

400 - Invalid request.

409 - Operation unsupported in the current state.

501 - This functionality is not implemented for the device in use.

GET /system/supportedVideoFormats

Get the list of supported video formats for the current system state.

Response

200 - List of supported video formats.

The response is JSON.

List of supported video formats.

Name	Type	Description
formats	array	List of video formats.
formats[i]	object	Video format configuration.
formats[i].name (required)	string	Video format serialised as a string.
formats[i].frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
formats[i].height	number	Height dimension of video format.
formats[i].width	number	Width dimension of video format.
formats[i].interlaced	boolean	Is the display format interlaced?

501 - This functionality is not implemented for the device in use.

GET /system/supportedFormats

Get supported formats.

Response

200 - List of supported formats.

The response is JSON.

Name	Type	Description
supportedFormats	array	
supportedFormats[i]	object	
supportedFormats[i].codecs	array	
supportedFormats[i].codecs[i]	string	
supportedFormats[i].frameRates	array	
supportedFormats[i].frameRates[i]	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
supportedFormats[i]. maxOffSpeedFrameRate	number	
supportedFormats[i]. minOffSpeedFrameRate	number	
supportedFormats[i]. recordResolution	object	
supportedFormats[i]. recordResolution.height	number	Height of the resolution.
supportedFormats[i]. recordResolution.width	number	Width of the resolution.
supportedFormats[i]. sensorResolution	object	
supportedFormats[i]. sensorResolution.height	number	Height of the resolution.
supportedFormats[i]. sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

GET /system/format

Get current format.

Response

200 - Current format.

The response is JSON.

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

501 - This functionality is not implemented for the device in use.

PUT /system/format

Set the format.

Parameters

Name	Type	Description
codec	string	Codec format serialised as a string.
frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
maxOffSpeedFrameRate	number	
minOffSpeedFrameRate	number	
offSpeedEnabled	boolean	
offSpeedFrameRate	number	
recordResolution	object	
recordResolution.height	number	Height of the resolution.
recordResolution.width	number	Width of the resolution.
sensorResolution	object	
sensorResolution.height	number	Height of the resolution.
sensorResolution.width	number	Width of the resolution.

Response

204 - System format updated.

501 - This functionality is not implemented for the device in use.

Transport Control API

API for controlling Transport on Blackmagic Design products.

GET /transports/0

Get device's basic transport status.

Response

200 - Transport status.

The response is JSON.

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

PUT /transports/0

Set device's basic transport status.

Parameters

Name	Type	Description
mode	string	Transport mode. Possible values are: InputPreview, Output.

Response

204 - Transport mode was set.

400 - Failed to set transport mode.

GET /transports/0/stop

Determine if transport is stopped.

Response

200 - Transport stop response.

The response is JSON.

Name	Type	Description
	boolean	True when transport mode is InputPreview or when in Output mode and speed is 0.

PUT /transports/0/stop

Stop transport. Deprecated, use POST /transports/0/stop instead.

Response

204 - Transport stopped.

POST /transports/0/stop

Stop transport.

Response

204 - Transport stopped.

GET /transports/0/play

Determine if transport is playing.

Response

200 - Transport play response.

The response is JSON.

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

PUT /transports/0/play

Start playing on transport. Deprecated, use POST /transports/0/play instead.

Response

204 - Transport playing.

400 - Failed to set transport to play.

POST /transports/0/play

Start playing on transport.

Response

204 - Transport playing.

400 - Failed to set transport to play.

GET /transports/0/playback

Get playback state.

Response

200 - Transport playback state.

The response is JSON.

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

PUT /transports/0/playback

Set playback state.

Parameters

Name	Type	Description
type	string	Possible values are: Play, Jog, Shuttle, Var.
loop	boolean	When true, playback loops from the end of the timeline to the beginning of the timeline.
singleClip	boolean	When true, playback loops from the end of the current clip to the beginning of the current clip.
speed	number	Playback speed, 1.0 for normal forward playback.
position	integer	Playback position on the timeline in units of video frames, where 0 is the first frame of the timeline.

Response

204 - Updated transport playback state.

400 - Failed to set transport playback state.

GET /transports/0/record

Get record state.

Response

200 - Recording state.

The response is JSON.

Name	Type	Description
recording	boolean	If true, transport is in InputRecord mode.

PUT /transports/0/record

Set record state. Deprecated, use POST /transports/0/record instead.

Parameters

Name	Type	Description
recording (required)	boolean	If true, starts a recording, otherwise stops.
clipName	string	Optional, sets the requested clip name to record to, when “recording” attribute is set to true.

Response

204 - Recording state updated.

400 - Failed to update recording state.

POST /transports/0/record

Start recording.

Parameters

Name	Type	Description
clipName	string	Optional, provides a specific name of clip to record to.

Response

204 - Recording started.

400 - Failed to start recording.

GET /transports/0/clipIndex

Get the clip index of the currently playing clip on the timeline.

Response

200 - Clip index response.

The response is JSON.

Name	Type	Description
clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

GET /transports/0/timecode

Get device timecode.

Response

200 - Timecode response.

The response is JSON.

Name	Type	Description
display	string	The display timecode serialised as a string.
timeline	string	The timeline timecode serialised as a string.

GET /transports/0/timecode/source

Get timecode source selected on device.

Response

200 - Timecode source response.

The response is JSON.

Name	Type	Description
timecode	string	Possible values are: Timeline, Clip.

Timeline Control API

API for controlling playback timeline.

GET /timelines/0

Get the playback timeline.

Response

200 - Playback timeline.

The response is JSON.

Name	Type	Description
clips	array	
clips[i]	object	Timeline clip.
clips[i].clipUniqueId (required)	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).
clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

404 - No timeline / disk available.

DELETE /timelines/0

Clear the current playback timeline. Deprecated, prefer to use POST /timelines/0/clear

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

POST /timelines/0

Add a clip to the timeline.

Parameters

This parameter can be one of the following types:

Add multiple media clips to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips[i].clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips[i].clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips[i].frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add multiple media clips to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	array	List of clips to add to the timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add a single clip to the timeline with optional insertion point and clip in/out points.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	object	Clip to add to the timeline, optionally cropping the clip before adding to the timeline.
clips.clipUniqueId (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.
clips.clipIn	integer	Insert this clip starting from this frame within the media clip. If omitted, starts from the beginning of the clip -- frame 0.
clips.frameCount	integer	Number of frames of this clip to add to the timeline. If omitted, use the whole clip, or the rest of the clip if clipIn was specified.

Add a single clip to the timeline with optional insertion point.

Name	Type	Description
insertBefore	integer	Clip(s) will be inserted before the clip at this timeline clip index, where 0 inserts to the beginning of the timeline. If omitted, inserts to the end of the timeline.
clips (required)	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the timeline as specified.

501 - The operation is not supported on this device.

POST /timelines/0/add

Add a clip to the end of the timeline. Deprecated, use POST /timelines/0 to add clips within the timeline.

Parameters

This parameter can be one of the following types:

Add one clip to the end of the timeline.

Name	Type	Description
clips	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Add many clips to the end of the timeline.

Name	Type	Description
clips	array	List of clipUniqueIds of clips to add to end of timeline.
clips[i]	integer	Unique ID (clipUniqueId) of the media clip to add to the timeline.

Response

204 - The clip was added to the end of the timeline.

501 - The operation is not supported on this device.

POST /timelines/0/clear

Clear the playback timeline.

Response

204 - The timeline was cleared.

501 - The operation is not supported on this device.

DELETE /timelines/0/clips/{timelineClipIndex}

Remove the specified clip from the timeline.

Parameters

Name	Type	Description
{timelineClipIndex} (required)	integer	The (0-based) timeline clip index of the clip to remove from the timeline.

Response

204 - The specified clip was removed from the timeline.

501 - The operation is not supported on this device.

Media Control API

API for controlling media devices in Blackmagic Design products.

GET /media/workingset

Get the list of media devices currently in the working set.

Response

200 - The list of media devices in the working set.

The response is JSON.

Name	Type	Description
size	integer	The fixed size of this device's working set.
workingset	array	The device's working set.
workingset[i]	object | null	Device within the working set. null if no device is present within the given working set slot.
workingset[i].volume	string	Volume name.
workingset[i].deviceName	string	Internal device name of this media device.
workingset[i].remainingRecordTime	integer	Remaining record time using current codec and video format in seconds.
workingset[i].totalSpace	integer	Total space on media device in bytes.
workingset[i].remainingSpace	integer	Remaining space on media device in bytes.
workingset[i].clipCount	integer	Number of clips currently on the device.

GET /media/active

Get the currently active media device.

Response

200 - The current active media device.

The response is JSON.

The active media device, or null if there is no active media.

Name	Type	Description
workingsetIndex	integer	Working set index of the active media device.
deviceName	string	Device name of media device.

204 - No media is currently active.

PUT /media/active

Set the currently active media device.

Parameters

Name	Type	Description
workingsetIndex	integer	Working set index of the media to make active.

Response

204 - The active media device was set successfully.

400 - Setting the currently active media device is not possible in the current state.

GET /media/devices/doformatSupportedFilesystems

Get the list of filesystems available to format a media device.

Response

200 - The list of filesystems permitted for formatting.

The response is JSON.

Name	Type	Description
	array	List of filesystems permitted for formatting media.
[i]	string	Filesystem serialised as string.

GET /media/devices/{deviceName}

Get information about a requested device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Information about the requested device.

The response is JSON.

Media device state.

Name	Type	Description
state	string	The current state of the media device. Possible values are: None, Scanning, Mounted, Uninitialised, Formatting, RaidComponent.

400 - Invalid device name.

404 - Device not found.

GET /media/devices/{deviceName}/doformat

Get a format key, used to format the device with a PUT request.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Response

200 - Format prepared.

The response is JSON.

Name	Type	Description
deviceName	string	Device name of media device to format.
key	string	The key required to format this device, provide to PUT /media/devices/{deviceName}/doformat to perform format of media device.

400 - Cannot format the device.

404 - Device not found.

PUT /media/devices/{deviceName}/doformat

Perform a format of the specified media device.

Parameters

Name	Type	Description
{deviceName} (required)	string	Device name of the media device. Retrieved by “deviceName” member of GET /media/workingset or GET /media/active.

Name	Type	Description
key	string	The key used to format this device, retrieved from prepare format media request GET /media/devices/{deviceName}/doformat. Format key provided cannot be reused after successful format.
filesystem	string	Filesystem to format to. Supported filesystems can be retrieved with GET /media/devices/doFormatSupportedFilesystems.
volume	string	Volume name to set for the disk after format.

Response

204 - Format successful.

400 - Cannot format the device, invalid filesystem or key.

404 - Device not found.

Slate Control API

API to manage digital slate data.

GET /slates/nextClip

Retrieve the digital slate for the next clip.

Response

200 - Returns the slate data for the next clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	

Name	Type	Description
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Slate data is not available.

PUT /slates/nextClip

Update the slate data for the next clip.

Parameters

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

POST /slates/nextClip/resetProjectData

Reset the project data for the next clip's slate.

Response

200 - Project data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetProjectData

Reset the project data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Project data reset successfully.

POST /slates/nextClip/resetLensData

Reset the lens data for the next clip's slate.

Response

200 - Lens data reset successfully.

POST /slates/clips/{deviceName}/{path}/resetLensData

Reset the lens data for the next clip's slate.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Lens data reset successfully.

GET /slates/clips/{deviceName}/{path}

Retrieve slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Response

200 - Returns the slate data for the specified clip.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

404 - Clip not found.

PUT /slates/clips/{deviceName}/{path}

Update the slate data for a specific clip.

Parameters

Name	Type	Description
{deviceName} (required)	string	Name of the device where the clip is stored. This is the same as the web browser's device name.
{path} (required)	string	Path to the clip.

Name	Type	Description
clip	object	
clip.reel	integer	
clip.scene	string	
clip.take	integer	
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

Response

200 - Successfully updated the slate data.

The response is JSON.

Name	Type	Description
clip	object	
clip.clipName	string	
clip.reel	integer	
clip.scene	string	
clip.sceneLocation	string	Possible values are: Interior, Exterior.
clip.sceneTime	string	Possible values are: Day, Night.
clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.

Name	Type	Description
clip.slateFor	string	Possible values are: Clip, Next Clip.
clip.take	integer	
clip.takeType	string	Possible values are: None, PU, VFX, SER.
clip.goodTake	boolean	
lens	object	
lens.lensType	string	
lens.iris	string	
lens.focalLength	string	
lens.distance	string	
lens.filter	string	
project	object	
project.projectName	string	
project.director	string	
project.camera	string	
project.cameraOperator	string	

409 - Partial update with errors.

The response is JSON.

Name	Type	Description
error	string	
details	array	
details[i]	object	
details[i].field	string	
details[i].message	string	

Preset Control API

API For controlling the presets on Blackmagic Design products

GET /presets

Get the list of the presets on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presets	array	List of the presets on the camera (.cset files)
presets[i]	string	

POST /presets

Send a preset file to the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
presetAdded	string	Name of the preset uploaded (without .cset extension)

400 - Bad request - missing Content-Disposition header or filename

GET /presets/active

Get the currently active preset on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

PUT /presets/active

Set the active preset on the camera

Parameters

Name	Type	Description
preset	string	Name of the active preset (with .cset extension, or 'default')

Response

204 - No Content

404 - Preset file not found

PARAMETERS /presets/{presetName}

GET /presets/{presetName}

Download the preset file

Response

200 - OK

The response is a binary file.

404 - File does not exist

PUT /presets/{presetName}

Save current camera state as a preset

Response

204 - No Content

DELETE /presets/{presetName}

Delete a preset from the camera

Response

204 - No Content

404 - Preset file not found

Audio Control API

API For controlling audio on Blackmagic Design Cameras

GET /audio/channels

Get the total number of audio channels available

Response

200 - Returns the total number of channels

The response is JSON.

Name	Type	Description
channels	integer	Total number of audio channels available

GET /audio/supportedInputs

Get the list of supported audio inputs

Response

200 - List of all supported audio inputs

The response is JSON.

Name	Type	Description
	array	
[i]	string	A supported audio input

GET /audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently selected input

The response is JSON.

Name	Type	Description
input	string	Audio input source and type

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/input

Set the audio input for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
input	string	Audio input source and type

Response

204 - No Content

400 - Invalid audio input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/input/description

Get the description of the current input of the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Description of the current input of the selected channel

The response is JSON.

Name	Type	Description
description	object	
description.gainRange	object	
description.gainRange.Min	number	The minimum gain value in dB
description.gainRange.Max	number	The maximum gain value in dB
description.capabilities	object	
description.capabilities. PhantomPower	boolean	Input supports setting of phantom power
description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter
description.capabilities.Padding	object	
description.capabilities.Padding. available	boolean	Input supports setting of padding
description.capabilities.Padding. forced	boolean	Padding is forced to be set for the input
description.capabilities.Padding. value	number	Value of the padding in dB

404 - Channel does not exist

GET /audio/channel/{channelIndex}/supportedInputs

Get the list of supported inputs and their availability to switch to for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its supported inputs are being queried. (Channels index from 0)

Response

200 - The list of supported inputs

The response is JSON.

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

404 - Channel does not exist

GET /audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set level for the selected channel

The response is JSON.

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/level

Set the audio input level for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
gain	number	Gain value in dB
normalised	number	Normalised level value between 0.0 and 1.0

Response

204 - No Content

400 - Invalid input or value out of range

404 - Channel does not exist

GET /audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set phantom power for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Phantom power enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/phantomPower

Set the audio phantom power for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Phantom power enabled state

Response

204 - No Content

400 - Phantom power is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set padding for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Padding enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/padding

Set the audio input padding for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Padding enabled state

Response

204 - No Content

400 - Padding is not supported or is forced for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set low cut filter for the selected channel

The response is JSON.

Name	Type	Description
enabled	boolean	Low cut filter enabled state

404 - Channel does not exist

PUT /audio/channel/{channelIndex}/lowCutFilter

Set the audio input low cut filter for the selected channel

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Name	Type	Description
enabled	boolean	Low cut filter enabled state

Response

204 - No Content

400 - Low cut filter is not supported for this input

404 - Channel does not exist

GET /audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

Parameters

Name	Type	Description
{channelIndex} (required)	integer	The index of the channel that its input is being controlled. (Channels index from 0)

Response

200 - Currently set availability for the selected channel

The response is JSON.

Name	Type	Description
available	boolean	Whether the input is currently available

404 - Channel does not exist

Lens Control API

API For controlling the lens on Blackmagic Design products

GET /lens/iris

Get lens' aperture

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number

PUT /lens/iris

Set lens' aperture

Parameters

Name	Type	Description
apertureStop	number	Aperture stop value
normalised	number	Normalised value
apertureNumber	integer	Aperture number
adjustmentStep	integer	Signed value for relative aperture adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens iris is not controllable or is controlled by auto exposure

GET /lens/zoom

Get lens' zoom

Response

200 - OK

The response is JSON.

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value

PUT /lens/zoom

Set lens' zoom

Parameters

Name	Type	Description
focalLength	integer	Focal length in mm
normalised	number	Normalised value
adjustmentFocalLength	integer	Signed value for relative focal length adjustment
adjustmentNormalised	number	Signed normalized value for relative zoom adjustment

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens zoom is not controllable

GET /lens/focus

Get lens' focus

Response

200 - OK

The response is JSON.

Name	Type	Description
normalised	number	Normalised value

PUT /lens/focus

Set lens' focus

Parameters

Name	Type	Description
normalised	number	Normalised value
focusDistance	integer	Focus distance value

Response

204 - No Content

400 - Bad Request if out of range value is provided

PUT /lens/focus/doAutoFocus

Perform auto focus

Parameters

Name	Type	Description
position	object	
position.x	number	Normalized x coordinate for autofocus ROI
position.y	number	Normalized y coordinate for autofocus ROI

Response

204 - No Content

400 - Bad Request if out of range value is provided

403 - Forbidden if lens focus is not controllable

GET /lens/opticalImageStabilization

Get optical image stabilization status

Response

200 - OK

The response is JSON.

Name	Type	Description
enabled	boolean	Whether optical image stabilization is enabled

501 - Not Implemented if optical image stabilization is not supported on this product

PUT /lens/opticalImageStabilization

Enable or disable optical image stabilization

Parameters

Name	Type	Description
enabled	boolean	Enable or disable optical image stabilization

Response

204 - No Content

501 - Not Implemented if optical image stabilization is not supported on this product

GET /lens/iris/description

Get detailed description of lens' iris capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the iris can be controlled
apertureStop	object	
apertureStop.min	number	Minimum aperture stop
apertureStop.max	number	Maximum aperture stop

GET /lens/zoom/description

Get detailed description of lens' zoom capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the zoom can be controlled
focalLength	object	
focalLength.adjustable	boolean	If focal length is adjustable
focalLength.min	integer	Minimum focal length
focalLength.max	integer	Maximum focal length

GET /lens/focus/description

Get detailed description of lens' focus capabilities

Response

200 - OK

The response is JSON.

Name	Type	Description
controllable	boolean	If the focus can be controlled
focusDistance	object	
focusDistance.adjustable	boolean	If focus distance is adjustable
focusDistance.min	number	Minimum focus distance
focusDistance.max	number	Maximum focus distance

Video Control API

API For controlling the video on Blackmagic Design products

GET /video/iso

Get current ISO

Response

200 - OK

The response is JSON.

Name	Type	Description
iso	integer	Current ISO value

PUT /video/iso

Set current ISO

Parameters

Name	Type	Description
iso	integer	ISO value to set

Response

204 - No Content

403 - ISO cannot be changed in the current state

GET /video/supportedISOs

Get the list of supported ISO settings

Response

200 - List of supported ISO values

The response is JSON.

Name	Type	Description
supportedISOs	array	Array of supported ISO values
supportedISOs[i]	integer	

GET /video/gain

Get current gain value in decibels

Response

200 - OK

The response is JSON.

Name	Type	Description
gain	integer	Current gain value in decibels

PUT /video/gain

Set current gain value

Parameters

Name	Type	Description
gain	integer	Gain value in decibels to set

Response

204 - No Content

403 - Gain cannot be changed in the current state

GET /video/supportedGains

Get the list of supported gain settings in decibels

Response

200 - List of supported gain values in decibels

The response is JSON.

Name	Type	Description
supportedGains	array	Array of supported gain values in decibels
supportedGains[i]	integer	

GET /video/whiteBalance

Get current white balance

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	integer	Current white balance

PUT /video/whiteBalance

Set current white balance

Parameters

Name	Type	Description
whiteBalance	integer	White balance to set

Response

204 - No Content

400 - Invalid white balance temperature

GET /video/whiteBalance/description

Get white balance range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalance	object	
whiteBalance.min	integer	Minimum color temperature
whiteBalance.max	integer	Maximum color temperature

PUT /video/whiteBalance/doAuto

Set current white balance automatically

Response

204 - No Content

GET /video/whiteBalanceTint

Get white balance tint

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	integer	Current white balance tint

PUT /video/whiteBalanceTint

Set white balance tint

Parameters

Name	Type	Description
whiteBalanceTint	integer	White balance tint to set

Response

204 - No Content

400 - Invalid white balance tint

GET /video/whiteBalanceTint/description

Get white balance tint range

Response

200 - OK

The response is JSON.

Name	Type	Description
whiteBalanceTint	object	
whiteBalanceTint.min	integer	Minimum white balance tint
whiteBalanceTint.max	integer	Maximum white balance tint

GET /video/ndFilter

Get ND filter stop

Response

200 - OK

The response is JSON.

Name	Type	Description
stop	number	Current filter power (fStop)

501 - Not implemented for this device

PUT /video/ndFilter

Set ND filter stop

Parameters

Name	Type	Description
stop	number	Filter power (fStop) to set

Response

204 - No Content

400 - Invalid ND filter stop

501 - Not implemented for this device

GET /video/supportedNDFilters

Get the list of available ND filter stops

Response

200 - List of available ND filter stops

The response is JSON.

Name	Type	Description
supportedStops	array	Array of available ND filter stops
supportedStops[i]	number	

501 - Not implemented for this device

GET /video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

Response

200 - List of supported display modes

The response is JSON.

Name	Type	Description
supportedDisplayModes	array	Array of supported display modes
supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

GET /video/ndFilter/displayMode

Get ND filter display mode on the camera

Response

200 - OK

The response is JSON.

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

501 - Not implemented for this device

PUT /video/ndFilter/displayMode

Set ND filter display mode on the camera

Parameters

Name	Type	Description
displayMode	string	Possible values are: Stop, Number, Fraction.

Response

204 - No Content

400 - Invalid display mode for ND filter

501 - Not implemented for this device

GET /video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

Response

200 - Indicates if ND filter is selectable

The response is JSON.

Name	Type	Description
selectable	boolean	True if ND filter adjustments are selectable via a slider

501 - Not implemented for this device

GET /video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

Response

200 - OK

The response is JSON.

Name	Type	Description
continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

PUT /video/shutter

Set current shutter

Parameters

Name	Type	Description
shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
shutterAngle	number	Shutter angle

Response

204 - No Content

GET /video/shutter/measurement

Get the current shutter measurement mode

Response

200 - OK

The response is JSON.

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

PUT /video/shutter/measurement

Set the shutter measurement mode

Parameters

Name	Type	Description
measurement	string	Possible values are: ShutterAngle, ShutterSpeed.

Response

204 - No Content

400 - Invalid shutter measurement

GET /video/supportedShutters

Get supported shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of supported shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of supported shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

Response

200 - OK

The response is JSON.

Name	Type	Description
shutterAngles	array	Array of flicker-free shutter angles
shutterAngles[i]	number	Shutter angle
shutterSpeeds	array	Array of flicker-free shutter speeds
shutterSpeeds[i]	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)

GET /video/autoExposure

Get current auto exposure mode

Response

200 - OK

The response is JSON.

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

PUT /video/autoExposure

Set auto exposure

Parameters

Name	Type	Description
mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
type	string	Comma-separated list of device types in the auto exposure stack

Response

204 - No Content

400 - Failed to set auto exposure mode

GET /video/detailSharpening

Get the current state of detail sharpening

Response

200 - Current detail sharpening state

The response is JSON.

Name	Type	Description
enabled	boolean	Whether detail sharpening is enabled

501 - Not implemented for this device

PUT /video/detailSharpening

Enable or disable detail sharpening

Parameters

Name	Type	Description
enabled	boolean	Enable or disable detail sharpening

Response

204 - Detail sharpening state updated

501 - Not implemented for this device

GET /video/detailSharpeningLevel

Get the current detail sharpening level

Response

200 - Current detail sharpening level

The response is JSON.

Name	Type	Description
level	string	Current detail sharpening level Possible values are: Low, Medium, High.

501 - Not implemented for this device

PUT /video/detailSharpeningLevel

Set the detail sharpening level

Parameters

Name	Type	Description
level	string	Desired level of detail sharpening Possible values are: Low, Medium, High.

Response

204 - Detail sharpening level updated

400 - Invalid detail sharpening level

501 - Not implemented for this device

Camera Control API

API For controlling the Camera specific features on Blackmagic Design products

GET /camera/colorBars

Get the status of color bars display

Response

200 - Returns the current status of color bars

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the color bars are currently enabled

PUT /camera/colorBars

Set the status of color bars display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the color bars

Response

204 - Color bars status updated successfully

GET /camera/programFeedDisplay

Get the status of program feed display

Response

200 - Returns the current status of program feed display

The response is JSON.

Name	Type	Description
enabled	boolean	Indicates if the program feed display is currently enabled

PUT /camera/programFeedDisplay

Set the status of program feed display

Parameters

Name	Type	Description
enabled	boolean	Enable or disable the program feed display

Response

204 - Program feed display status updated successfully

GET /camera/tallyStatus

Get the tally status of the camera

Response

200 - Returns the current tally status of the camera

The response is JSON.

Name	Type	Description
status	string	Current tally status of the camera Possible values are: None, Preview, Program.

GET /camera/power

Get the power status of the camera

Response

200 - Returns the current power status

The response is JSON.

Name	Type	Description
source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
batteries	array	List of batteries currently connected to the camera
batteries[i]	object	
batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
batteries[i].chargeRemainingPercent	integer	Remaining battery charge percentage
batteries[i].statusFlags	array	List of battery status flags
batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

GET /camera/power/displayMode

Get the power display mode of the camera

Response

200 - Returns the current power display mode

The response is JSON.

Name	Type	Description
mode	string	Current power display mode Possible values are: Percentage, Voltage.

PUT /camera/power/displayMode

Set the power display mode of the camera

Parameters

Name	Type	Description
mode	string	Power display mode to set Possible values are: Percentage, Voltage.

Response

204 - Power display mode updated successfully

400 - Invalid power display mode

GET /camera/timingReferenceLock

Get the timing reference lock status

Response

200 - Returns the timing reference lock status

The response is JSON.

Name	Type	Description
locked	boolean	Indicates if timing reference is locked

Immersive Control API

API for controlling immersive camera settings on Blackmagic Design cameras

GET /immersive/display/{displayName}/eye

Get the current eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to query (from /monitoring/display endpoint)

Response

200 - OK

The response is JSON.

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

400 - Invalid display name format

404 - Display not found

422 - Failed to get eye view

PUT /immersive/display/{displayName}/eye

Set the eye view for a specific display

Parameters

Name	Type	Description
{displayName} (required)	string	The display name to control (from /monitoring/display endpoint)

Name	Type	Description
eye (required)	string	The eye view to display Possible values are: Left, Right.

Response

204 - No Content

400 - Invalid input or display name format

404 - Display not found

422 - Failed to set eye view

Color Correction Control API

API For controlling the color correction on Blackmagic Design products based on DaVinci Resolve Color Corrector

GET /colorCorrection/lift

Get color correction lift

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

PUT /colorCorrection/lift

Set color correction lift

Parameters

Name	Type	Description
red	number	Red lift component. If omitted, value remains unchanged.
green	number	Green lift component. If omitted, value remains unchanged.
blue	number	Blue lift component. If omitted, value remains unchanged.
luma	number	Luma lift component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gamma

Get color correction gamma

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

PUT /colorCorrection/gamma

Set color correction gamma

Parameters

Name	Type	Description
red	number	Red gamma component. If omitted, value remains unchanged.
green	number	Green gamma component. If omitted, value remains unchanged.
blue	number	Blue gamma component. If omitted, value remains unchanged.
luma	number	Luma gamma component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/gain

Get color correction gain

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

PUT /colorCorrection/gain

Set color correction gain

Parameters

Name	Type	Description
red	number	Red gain component. If omitted, value remains unchanged.
green	number	Green gain component. If omitted, value remains unchanged.
blue	number	Blue gain component. If omitted, value remains unchanged.
luma	number	Luma gain component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/offset

Get color correction offset

Response

200 - OK

The response is JSON.

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

PUT /colorCorrection/offset

Set color correction offset

Parameters

Name	Type	Description
red	number	Red offset component. If omitted, value remains unchanged.
green	number	Green offset component. If omitted, value remains unchanged.
blue	number	Blue offset component. If omitted, value remains unchanged.
luma	number	Luma offset component. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/contrast

Get color correction contrast

Response

200 - OK

The response is JSON.

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/contrast

Set color correction contrast

Parameters

Name	Type	Description
pivot	number	Contrast pivot point. If omitted, value remains unchanged.
adjust	number	Contrast adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/color

Get color correction color properties

Response

200 - OK

The response is JSON.

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

PUT /colorCorrection/color

Set color correction color properties

Parameters

Name	Type	Description
hue	number	Color hue adjustment. If omitted, value remains unchanged.
saturation	number	Color saturation adjustment. If omitted, value remains unchanged.

Response

204 - No Content

GET /colorCorrection/lumaContribution

Get color correction luma contribution

Response

200 - OK

The response is JSON.

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

PUT /colorCorrection/lumaContribution

Set color correction luma contribution

Parameters

Name	Type	Description
lumaContribution	number	Luma contribution value. If omitted, value remains unchanged.

Response

204 - No Content

Notification websocket - 1.0.0

Service that notifies subscribers of device state changes.

messages

Subscribe (The messages from the server/device)

Websocket Opened Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: websocketOpened.
.type	string	Possible values are: event.

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*

Name	Type	Description
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Event Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: propertyValueChanged.

Name	Type	Description
.data.property	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.value	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.type	string	Possible values are: event.

Publish (The messages that user can send to the server/device)

Response Message (JSON)

Name	Type	Description
.data	object	
.data.action	string	Possible values are: subscribe, unsubscribe, listSubscriptions, listProperties, websocketOpened.
.data.properties	array	

Name	Type	Description
.data.properties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.data.values	object	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.
.data.success	boolean	
.data.deviceProperties	array	

Name	Type	Description
.data.deviceProperties[i]	string	device property that the user can subscribe to. The user can either choose a value from the predefined enum, or provide a wildcard string. Possible values are: /media/workingset, /media/active, /system, /system/codecFormat, /system/videoFormat, /system/format, /system/supportedFormats, /timelines/0, /transports/0, /transports/0/stop, /transports/0/play, /transports/0/playback, /transports/0/record, /transports/0/timecode, /transports/0/timecode/source, /transports/0/clipIndex, /slates/nextClip, /monitoring/{displayName}/cleanFeed, /monitoring/{displayName}/displayLUT, /monitoring/{displayName}/zebra, /monitoring/{displayName}/focusAssist, /monitoring/{displayName}/frameGuide, /monitoring/{displayName}/frameGrids, /monitoring/{displayName}/safeArea, /monitoring/{displayName}/falseColor, /monitoring/focusAssist, /monitoring/frameGuideRatio, /monitoring/frameGrids, /monitoring/safeAreaPercent, /audio/channel/{channelIndex}/input, /audio/channel/{channelIndex}/supportedInputs, /audio/channel/{channelIndex}/level, /audio/channel/{channelIndex}/phantomPower, /audio/channel/{channelIndex}/padding, /audio/channel/{channelIndex}/lowCutFilter, /audio/channel/{channelIndex}/available, /audio/channel/{channelIndex}/input/description, /colorCorrection/lift, /colorCorrection/gamma, /colorCorrection/gain, /colorCorrection/offset, /colorCorrection/contrast, /colorCorrection/color, /colorCorrection/lumaContribution, /lens/iris, /lens/iris/description, /lens/focus, /lens/focus/description, /lens/zoom, /lens/zoom/description, /presets, /presets/active, /camera/colorBars, /camera/programFeedDisplay, /camera/tallyStatus, /camera/power, /camera/power/displayMode, /camera/timingReferenceLock, /video/iso, /video/supportedISOs, /video/gain, /video/supportedGains, /video/whiteBalance, /video/whiteBalance/description, /video/whiteBalanceTint, /video/whiteBalanceTint/description, /video/ndFilter, /video/supportedNDFilters, /video/ndFilter/displayMode, /video/supportedNDFilterDisplayModes, /video/ndFilterSelectable, /video/shutter, /video/shutter/measurement, /video/supportedShutters, /video/flickerFreeShutters, /video/autoExposure, /video/detailSharpening, /video/detailSharpeningLevel. Must match the pattern: .*
.type	string	Possible values are: response.
.id	number	Optional parameter that repeats the id in the output for tracking messages.

Device Properties

/media/workingset

The value JSON returned via the eventResponse when the `/media/workingset` property changes on the device:

Name	Type	Description
.size	integer	The fixed size of this device's working set.
.workingset	array	Array of devices within the working set. null if no device is present within the given working set slot.
.workingset[i]		

/media/active

The value JSON returned via the eventResponse when the `/media/active` property changes on the device:

Name	Type	Description
.workingsetIndex	integer	Working set index of the active media device.
.deviceName	string	Internal device name of this media device.

/system

The value JSON returned via the eventResponse when the `/system` property changes on the device:

Name	Type	Description
.codecFormat	object	Codec format configuration.
.codecFormat.codec	string	Codec format serialised as a string.
.codecFormat.container	string	Multimedia container format.
.videoFormat	object	Currently selected video format.
.videoFormat.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.videoFormat.height	number	Height dimension of video format.
.videoFormat.width	number	Width dimension of video format.
.videoFormat.interlaced	boolean	Is the display format interlaced?.
.videoFormat.name	string	Video format serialised as a string.

/system/codecFormat

Codec format configuration.

The value JSON returned via the eventResponse when the `/system/codecFormat` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.container	string	Multimedia container format.

/system/videoFormat

Currently selected video format.

The value JSON returned via the eventResponse when the `/system/videoFormat` property changes on the device:

Name	Type	Description
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.height	number	Height dimension of video format.
.width	number	Width dimension of video format.
.interlaced	boolean	Is the display format interlaced?.
.name	string	Video format serialised as a string.

/system/format

The value JSON returned via the eventResponse when the `/system/format` property changes on the device:

Name	Type	Description
.codec	string	Codec format serialised as a string.
.frameRate	string	Frame rate. Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.maxOffSpeedFrameRate	number	
.minOffSpeedFrameRate	number	
.offSpeedEnabled	boolean	
.offSpeedFrameRate	number	
.recordResolution	object	
.recordResolution.height	number	Height of the resolution.
.recordResolution.width	number	Width of the resolution.
.sensorResolution	object	
.sensorResolution.height	number	Height of the resolution.
.sensorResolution.width	number	Width of the resolution.

/system/supportedFormats

The value JSON returned via the eventResponse when the `/system/supportedFormats` property changes on the device:

Name	Type	Description
.supportedFormats	array	
.supportedFormats[i]	object	
.supportedFormats[i].codecs	array	
.supportedFormats[i].codecs[i]	string	
.supportedFormats[i].frameRates	array	
.supportedFormats[i].frameRates[i]	string	Possible values are: 23.98, 24.00, 24, 25.00, 25, 29.97, 30.00, 30, 47.95, 48.00, 48, 50.00, 50, 59.94, 60.00, 60, 119.88, 120.00, 120.
.supportedFormats[i].maxOffSpeedFrameRate	number	
.supportedFormats[i].minOffSpeedFrameRate	number	
.supportedFormats[i].recordResolution	object	
.supportedFormats[i].recordResolution.height	number	Height of the resolution.
.supportedFormats[i].recordResolution.width	number	Width of the resolution.
.supportedFormats[i].sensorResolution	object	
.supportedFormats[i].sensorResolution.height	number	Height of the resolution.
.supportedFormats[i].sensorResolution.width	number	Width of the resolution.

/timelines/0

The value JSON returned via the eventResponse when the `/timelines/0` property changes on the device:

Name	Type	Description
.clips	array	
.clips[i]	object	Timeline clip.
.clips[i].clipUniqueId	integer	Unique identifier used to identify this media clip. If the same media clip is added to the timeline multiple times, each timeline clip has the same clipUniqueId
.clips[i].frameCount	integer	Duration of timeline clip in frames, the number of frames in this clip on the timeline.
.clips[i].durationTimecode	string	Duration of the timeline clip in timecode format serialised as string. This will differ to durationTimecode reported in /clips for this clipUniqueId if clipIn or frameCount was specified when adding this clip to the timeline.
.clips[i].clipIn	string	In frame offset for the clip on the timeline, where 0 is the first frame of the on-disk clip.
.clips[i].inTimecode	string	Clip timecode of the first frame of this timeline clip serialised as string (clip startTimecode + clipIn frames).

Name	Type	Description
.clips[i].timelineIn	string	Timeline position of the first frame of this clip, where 0 is the first frame of the timeline.
.clips[i].timelineInTimecode	string	Timeline timecode of the first frame of this timeline clip serialised as string.

/transports/0

The value JSON returned via the eventResponse when the `/transports/0` property changes on the device:

Name	Type	Description
.mode	string	Transport mode. Possible values are: InputPreview, InputRecord, Output.

/transports/0/stop

true when transport mode is InputPreview or when in Output mode and speed is 0.

The value JSON returned via the eventResponse when the `/transports/0/stop` property changes on the device:

Name	Type	Description
	boolean	true when transport mode is InputPreview or when in Output mode and speed is 0.

/transports/0/play

True when transport is in Output mode and speed is non-zero.

The value JSON returned via the eventResponse when the /transports/0/play property changes on the device:

Name	Type	Description
	boolean	True when transport is in Output mode and speed is non-zero.

/transports/0/playback

The value JSON returned via the eventResponse when the `/transports/0/playback` property changes on the device:

Name	Type	Description
.type	string	Possible values are: Play, Jog, Shuttle, Var.
.loop	boolean	When true playback loops from the end of the timeline to the beginning of the timeline.
.singleClip	boolean	When true playback loops from the end of the current clip to the beginning of the current clip.
.speed	number	Playback speed, 1.0 for normal forward playback
.position	integer	Playback position on the timeline in units of video frames.

/transports/0/record

The value JSON returned via the eventResponse when the `/transports/0/record` property changes on the device:

Name	Type	Description
.recording	boolean	Is transport in Input Record mode.

/transports/0/timecode

The value JSON returned via the eventResponse when the `/transports/0/timecode` property changes on the device:

Name	Type	Description
.display	string	The display timecode serialised as a string.
.timeline	string	The timeline timecode serialised as a string.

/transports/0/timecode/source

The value JSON returned via the eventResponse when the `/transports/0/timecode/source` property changes on the device:

Name	Type	Description
.timecode	string	Possible values are: Timeline, Clip.

/transports/0/clipIndex

The value JSON returned via the eventResponse when the `/transports/0/clipIndex` property changes on the device:

Name	Type	Description
.clipIndex	number | null	The 0-based index of the clip being played on the timeline. null when there is no timeline or an empty timeline.

/slates/nextClip

The value JSON returned via the eventResponse when the `/slates/nextClip` property changes on the device:

Name	Type	Description
.clip	object	
.clip.clipName	string	
.clip.reel	integer	
.clip.scene	string	
.clip.sceneLocation	string	Possible values are: Interior, Exterior.
.clip.sceneTime	string	Possible values are: Day, Night.
.clip.shotType	string	Possible values are: None, WS, MS, MCU, CU, BCU, ECU.
.clip.take	integer	
.clip.takeType	string	Possible values are: None, PU, VFX, SER.
.clip.goodTake	boolean	
.lens	object	
.lens.lensType	string	
.lens.iris	string	

Name	Type	Description
.lens.focalLength	string	
.lens.distance	string	
.lens.filter	string	
.project	object	
.project.projectName	string	
.project.director	string	
.project.camera	string	
.project.cameraOperator	string	

/monitoring/{displayName}/cleanFeed

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/cleanFeed` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/displayLUT

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/displayLUT` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/zebra

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/zebra` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/focusAssist` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGuide

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGuide` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/frameGrids` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/safeArea

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/safeArea` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/{displayName}/falseColor

The value JSON returned via the eventResponse when the `/monitoring/{displayName}/falseColor` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the feature is enabled.

/monitoring/focusAssist

The value JSON returned via the eventResponse when the `/monitoring/focusAssist` property changes on the device:

Name	Type	Description
.mode	string	Possible values are: Peak, ColoredLines.
.color	string	Possible values are: Red, Green, Blue, White, Black.
.intensity	integer	

/monitoring/frameGuideRatio

The value JSON returned via the eventResponse when the `/monitoring/frameGuideRatio` property changes on the device:

Name	Type	Description
.ratio	string	

/monitoring/frameGrids

The value JSON returned via the eventResponse when the `/monitoring/frameGrids` property changes on the device:

Name	Type	Description
.frameGrids	array	
.frameGrids[i]	string	Possible values are: Thirds, Crosshair, Dot, Horizon.

/monitoring/safeAreaPercent

The value JSON returned via the eventResponse when the `/monitoring/safeAreaPercent` property changes on the device:

Name	Type	Description
.percent	integer	Safe area coverage percentage.

/audio/channel/{channelIndex}/input

Get the audio input (source and type) for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input` property changes on the device:

Name	Type	Description
.input	string	Audio input source and type

/audio/channel/{channelIndex}/supportedInputs

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/supportedInputs` property changes on the device:

Name	Type	Description
	array	
[i]	object	
[i].input	string	Input name
[i].available	boolean	Is the input available to be switched into from the current input for the selected channel

/audio/channel/{channelIndex}/level

Get the audio input level for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/level` property changes on the device:

Name	Type	Description
.gain	number	Gain value in dB
.normalised	number	Normalised level value between 0.0 and 1.0

/audio/channel/{channelIndex}/phantomPower

Get the audio input phantom power status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/phantomPower` property changes on the device:

Name	Type	Description
.enabled	boolean	Phantom power enabled state

/audio/channel/{channelIndex}/padding

Get the audio input padding status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/padding` property changes on the device:

Name	Type	Description
.enabled	boolean	Padding enabled state

/audio/channel/{channelIndex}/lowCutFilter

Get the audio input low cut filter status for the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/lowCutFilter` property changes on the device:

Name	Type	Description
.enabled	boolean	Low cut filter enabled state

/audio/channel/{channelIndex}/available

Get the audio input's current availability for the selected channel. If unavailable, the source will be muted

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/available` property changes on the device:

Name	Type	Description
.available	boolean	Whether the input is currently available

/audio/channel/{channelIndex}/input/description

Description of the current input of the selected channel

The value JSON returned via the eventResponse when the `/audio/channel/{channelIndex}/input/description` property changes on the device:

Name	Type	Description
.description	object	
.description.gainRange	object	
.description.gainRange.Min	number	The minimum gain value in dB
.description.gainRange.Max	number	The maximum gain value in dB
.description.capabilities	object	
.description.capabilities.PhantomPower	boolean	Input supports setting of phantom power
.description.capabilities.LowCutFilter	boolean	Input supports setting of low cut filter

Name	Type	Description
.description.capabilities.Padding	object	
.description.capabilities.Padding.available	boolean	Input supports setting of padding
.description.capabilities.Padding.forced	boolean	Padding is forced to be set for the input
.description.capabilities.Padding.value	number	An object with property names as the key and a property value as json. Check the next section for the device properties and their return values.

/colorCorrection/lift

Get color correction lift

The value JSON returned via the eventResponse when the `/colorCorrection/lift` property changes on the device:

Name	Type	Description
.red	number	Red lift component
.green	number	Green lift component
.blue	number	Blue lift component
.luma	number	Luma lift component

/colorCorrection/gamma

Get color correction gamma

The value JSON returned via the eventResponse when the `/colorCorrection/gamma` property changes on the device:

Name	Type	Description
.red	number	Red gamma component
.green	number	Green gamma component
.blue	number	Blue gamma component
.luma	number	Luma gamma component

/colorCorrection/gain

Get color correction gain

The value JSON returned via the eventResponse when the `/colorCorrection/gain` property changes on the device:

Name	Type	Description
.red	number	Red gain component
.green	number	Green gain component
.blue	number	Blue gain component
.luma	number	Luma gain component

/colorCorrection/offset

Get color correction offset

The value JSON returned via the eventResponse when the `/colorCorrection/offset` property changes on the device:

Name	Type	Description
.red	number	Red offset component
.green	number	Green offset component
.blue	number	Blue offset component
.luma	number	Luma offset component

/colorCorrection/contrast

Get color correction contrast

The value JSON returned via the eventResponse when the `/colorCorrection/contrast` property changes on the device:

Name	Type	Description
.pivot	number	Contrast pivot point
.adjust	number	Contrast adjustment

/colorCorrection/color

Get color correction color properties

The value JSON returned via the eventResponse when the `/colorCorrection/color` property changes on the device:

Name	Type	Description
.hue	number	Color hue adjustment
.saturation	number	Color saturation adjustment

/colorCorrection/lumaContribution

Get color correction luma contribution

The value JSON returned via the eventResponse when the `/colorCorrection/lumaContribution` property changes on the device:

Name	Type	Description
.lumaContribution	number	Luma contribution value

/lens/iris

Get lens' aperture

The value JSON returned via the eventResponse when the `/lens/iris` property changes on the device:

Name	Type	Description
.continuousApertureAutoExposure	boolean	Is Aperture controlled by auto exposure
.apertureStop	number	Aperture stop value
.normalised	number	Normalised value
.apertureNumber	integer	Aperture number

/lens/iris/description

Get detailed description of lens' iris capabilities

The value JSON returned via the eventResponse when the `/lens/iris/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the iris can be controlled
.apertureStop	object	
.apertureStop.min	number	Minimum aperture stop
.apertureStop.max	number	Maximum aperture stop

/lens/focus

Get lens' focus

The value JSON returned via the eventResponse when the `/lens/focus` property changes on the device:

Name	Type	Description
.normalised	number	Normalised value

/lens/focus/description

Get detailed description of lens' focus capabilities

The value JSON returned via the eventResponse when the `/lens/focus/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the focus can be controlled
.focusDistance	object	
.focusDistance.adjustable	boolean	If focus distance is adjustable
.focusDistance.min	number	Minimum focus distance
.focusDistance.max	number	Maximum focus distance

/lens/zoom

Get lens' zoom

The value JSON returned via the eventResponse when the `/lens/zoom` property changes on the device:

Name	Type	Description
.focalLength	integer	Focal length in mm
.normalised	number	Normalised value

/lens/zoom/description

Get detailed description of lens' zoom capabilities

The value JSON returned via the eventResponse when the `/lens/zoom/description` property changes on the device:

Name	Type	Description
.controllable	boolean	If the zoom can be controlled
.focalLength	object	
.focalLength.adjustable	boolean	If focal length is adjustable
.focalLength.min	integer	Minimum focal length
.focalLength.max	integer	Maximum focal length

/presets

Get the list of the presets on the camera

The value JSON returned via the eventResponse when the `/presets` property changes on the device:

Name	Type	Description
.presets	array	List of the presets on the camera (.cset files)
.presets[i]	string	

/presets/active

Get the currently active preset on the camera

The value JSON returned via the eventResponse when the `/presets/active` property changes on the device:

Name	Type	Description
.preset	string	Name of the active preset (with .cset extension, or 'default')

/camera/colorBars

Get the status of color bars display

The value JSON returned via the eventResponse when the `/camera/colorBars` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the color bars are currently enabled

/camera/programFeedDisplay

Get the status of program feed display

The value JSON returned via the eventResponse when the `/camera/programFeedDisplay` property changes on the device:

Name	Type	Description
.enabled	boolean	Indicates if the program feed display is currently enabled

/camera/tallyStatus

Get the tally status of the camera

The value JSON returned via the eventResponse when the `/camera/tallyStatus` property changes on the device:

Name	Type	Description
.status	string	Current tally status of the camera Possible values are: None, Preview, Program.

/camera/power

Get the power status of the camera

The value JSON returned via the eventResponse when the `/camera/power` property changes on the device:

Name	Type	Description
.source	string	Current power source of the camera Possible values are: Battery, AC, Fiber, USB, POE.
.milliVolt	integer	Current voltage level in millivolts (rounded to nearest 100mV)
.batteries	array	
.batteries[i]	object	
.batteries[i].milliVolt	integer	Battery voltage in millivolts (rounded to nearest 100mV)
.batteries[i]. chargeRemainingPercent	integer	Remaining battery charge percentage
.batteries[i].statusFlags	array	List of battery status flags
.batteries[i].statusFlags[i]	string	Possible values are: Unknown Battery Status, Battery Is Present, Battery Is Charging, Battery Percentage Is Low, Battery Voltage Is Low, Battery Is Critically Low, Charge Remaining Percentage Is Estimated, Battery Communications Is Active, Battery Is Connected.

/camera/power/displayMode

Get the power display mode of the camera

The value JSON returned via the eventResponse when the `/camera/power/displayMode` property changes on the device:

Name	Type	Description
.mode	string	Current power display mode Possible values are: Percentage, Voltage.

/camera/timingReferenceLock

Get the timing reference lock status

The value JSON returned via the eventResponse when the `/camera/timingReferenceLock` property changes on the device:

Name	Type	Description
.locked	boolean	Indicates if timing reference is locked

/video/iso

Get current ISO

The value JSON returned via the eventResponse when the `/video/iso` property changes on the device:

Name	Type	Description
.iso	integer	Current ISO value

/video/supportedISOs

Get the list of supported ISO settings

The value JSON returned via the eventResponse when the `/video/supportedISOs` property changes on the device:

Name	Type	Description
.supportedISOs	array	Array of supported ISO values
.supportedISOs[i]	integer	

/video/gain

Get current gain value in decibels

The value JSON returned via the eventResponse when the `/video/gain` property changes on the device:

Name	Type	Description
.gain	integer	Current gain value in decibels

/video/supportedGains

Get the list of supported gain settings in decibels

The value JSON returned via the eventResponse when the `/video/supportedGains` property changes on the device:

Name	Type	Description
.supportedGains	array	Array of supported gain values in decibels
.supportedGains[i]	integer	

/video/whiteBalance

Get current white balance

The value JSON returned via the eventResponse when the `/video/whiteBalance` property changes on the device:

Name	Type	Description
.whiteBalance	integer	Current white balance

/video/whiteBalance/description

Get white balance range

The value JSON returned via the eventResponse when the `/video/whiteBalance/description` property changes on the device:

Name	Type	Description
.whiteBalance	object	
.whiteBalance.min	integer	Minimum color temperature
.whiteBalance.max	integer	Maximum color temperature

/video/whiteBalanceTint

Get white balance tint

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint` property changes on the device:

Name	Type	Description
.whiteBalanceTint	integer	Current white balance tint

/video/whiteBalanceTint/description

Get white balance tint range

The value JSON returned via the eventResponse when the `/video/whiteBalanceTint/description` property changes on the device:

Name	Type	Description
.whiteBalanceTint	object	
.whiteBalanceTint.min	integer	Minimum white balance tint
.whiteBalanceTint.max	integer	Maximum white balance tint

/video/ndFilter

Get ND filter stop

The value JSON returned via the eventResponse when the `/video/ndFilter` property changes on the device:

Name	Type	Description
.stop	number	Current filter power (fStop)

/video/supportedNDFilters

Get the list of available ND filter stops

The value JSON returned via the eventResponse when the `/video/supportedNDFilters` property changes on the device:

Name	Type	Description
.supportedStops	array	Array of available ND filter stops
.supportedStops[i]	number	

/video/ndFilter/displayMode

Get ND filter display mode on the camera

The value JSON returned via the eventResponse when the `/video/ndFilter/displayMode` property changes on the device:

Name	Type	Description
.displayMode	string	ND filter display mode Possible values are: Stop, Number, Fraction.

/video/supportedNDFilterDisplayModes

Get the list of supported ND filter display modes

The value JSON returned via the eventResponse when the `/video/supportedNDFilterDisplayModes` property changes on the device:

Name	Type	Description
.supportedDisplayModes	array	Array of supported display modes
.supportedDisplayModes[i]	string	Possible values are: Stop, Number, Fraction.

/video/ndFilterSelectable

Check if ND filter adjustments are selectable via a slider

The value JSON returned via the eventResponse when the `/video/ndFilterSelectable` property changes on the device:

Name	Type	Description
.selectable	boolean	True if ND filter adjustments are selectable via a slider

/video/shutter

Get current shutter. Will return either shutter speed or shutter angle depending on shutter measurement in device settings

The value JSON returned via the eventResponse when the `/video/shutter` property changes on the device:

Name	Type	Description
.continuousShutterAutoExposure	boolean	Is shutter controlled by auto exposure
.shutterSpeed	integer	Shutter speed value in fractions of a second (minimum is sensor frame rate)
.shutterAngle	number	Shutter angle

/video/shutter/measurement

Get the current shutter measurement mode

The value JSON returned via the eventResponse when the `/video/shutter/measurement` property changes on the device:

Name	Type	Description
.measurement	string	Current shutter measurement mode Possible values are: ShutterAngle, ShutterSpeed.

/video/supportedShutters

Get supported shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/supportedShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of supported shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array of flicker-free shutter speeds
.shutterSpeeds[i]	integer	

/video/flickerFreeShutters

Get flicker-free shutter settings based on current camera configuration

The value JSON returned via the eventResponse when the `/video/flickerFreeShutters` property changes on the device:

Name	Type	Description
.shutterAngles	array	Array of flicker-free shutter angles
.shutterAngles[i]	number	
.shutterSpeeds	array	Array
.shutterSpeeds[i]	integer	

/video/autoExposure

Get current auto exposure mode

The value JSON returned via the eventResponse when the `/video/autoExposure` property changes on the device:

Name	Type	Description
.mode	string	Auto exposure mode Possible values are: Off, Continuous, OneShot.
.type	string	Comma-separated list of device types in the auto exposure stack

/video/detailSharpening

Get the current state of detail sharpening

The value JSON returned via the eventResponse when the `/video/detailSharpening` property changes on the device:

Name	Type	Description
.enabled	boolean	Whether detail sharpening is enabled

/video/detailSharpeningLevel

Get the current detail sharpening level

The value JSON returned via the eventResponse when the `/video/detailSharpeningLevel` property changes on the device:

Name	Type	Description
.level	string	Current detail sharpening level of supported shutter speeds Possible values are: Low, Medium, High.

Допомога

Найшвидший спосіб отримати допомогу — перейти до сторінок підтримки на сайті Blackmagic Design і перевірити наявність нових довідкових матеріалів щодо камери.

Сторінки підтримки на сайті Blackmagic Design

Найновіші версії посібника з експлуатації, програмного забезпечення та додаткову інформацію можна знайти в центрі підтримки Blackmagic Design на сторінці www.blackmagicdesign.com/ua/support

Звернення до служби підтримки Blackmagic Design

Якщо за допомогою доступних довідкових матеріалів та форуму вирішити проблему не вдалося, скористайтесь формою «Надіслати імейл» на сторінці підтримки. Також можна зателефонувати до найближчого представництва Blackmagic Design, телефон якого ви знайдете на нашому вебсайті.

Перевірка інстальованої версії програмного забезпечення

Щоб дізнатися, яка версія утиліти Blackmagic Camera Setup інстальована на комп'ютері, відкрийте вікно About Blackmagic Camera Setup.

- На комп'ютері з операційною системою Mac відкрийте Blackmagic Camera Setup із вікна «Програми». У меню утиліти виберіть About Blackmagic Camera Setup, щоб побачити номер версії.
- На комп'ютері з операційною системою Windows відкрийте Blackmagic Camera Setup у меню «Пуск» або клацніть піктограму утиліти на початковому екрані. У меню «Довідка» виберіть About Blackmagic Camera Setup, щоб побачити номер версії.

Завантаження оновлень програмного забезпечення

Дізнавшись версію інстальованої утиліти Blackmagic Camera Setup, перейдіть до центру підтримки Blackmagic Design на сторінці www.blackmagicdesign.com/ua/support, щоб перевірити наявність оновлень. Радимо завжди використовувати актуальну версію програмного забезпечення, однак оновлення найкраще виконувати після завершення поточного проекту.

Дотримання нормативних вимог

Утилізація електрообладнання та електронної апаратури в країнах Європейського Союзу

Виріб має маркування, яке означає, що його забороняється утилізувати разом із побутовими відходами. Непридатне для експлуатації обладнання необхідно передати до пункту вторинної переробки. Роздільний збір відходів і їх повторне використання дозволяють зберігати природні ресурси, охороняти довкілля та захищати здоров'я людей. Щоб отримати докладну інформацію про порядок утилізації, зверніться до місцевих муніципальних органів або дилера, у якого ви придбали цей пристрій.

Дане обладнання протестовано за вимогами для цифрових пристроїв класу A (розділ 15 специфікацій FCC) та визнано таким, що відповідає усім критеріям. Дотримання згаданих нормативів забезпечує достатній захист від шкідливого випромінювання під час роботи обладнання в нежитлових приміщеннях. Оскільки цей виріб генерує та випромінює радіохвилі, при неправильному встановленні він може стати джерелом радіоперешкод. Якщо обладнання експлуатується в житлових приміщеннях, підвищується ймовірність виникнення перешкод, вплив яких у цьому випадку користувач повинен усунути самостійно.

До експлуатації допускається обладнання, що відповідає двом основним вимогам.

1. Обладнання або пристрій не має бути джерелом шкідливих перешкод.
2. Обладнання або пристрій має бути стійким до перешкод, включно з тими, що можуть спричинити перебій у роботі.

R-R-BMD-20250213001

ICES-3 (A)
NMB-3 (A)

Відповідність вимогам ISED (Канада)

Дане обладнання відповідає канадським стандартам для цифрових пристроїв класу A.

Будь-яка модифікація або використання виробу не за призначенням може анулювати заяву про відповідність цим стандартам.

Підключення до HDMI-інтерфейсу має виконуватися за допомогою якісного екранованого кабелю.

Це обладнання протестовано за вимогами, що висуваються до роботи пристроїв у нежитлових приміщеннях. При використанні в побутових умовах воно може стати джерелом перешкод для радіосигналу.

Правила безпеки

Блок живлення постійного струму 24 В для Blackmagic URSA Cine Immersive слід підключати до електричної розетки, яка має контакт заземлення.

Усередині камери Blackmagic URSA Cine Immersive не містяться деталі, що підлягають обслуговуванню. Для виконання ремонтних робіт зверніться до місцевого сервісного центру Blackmagic Design.

При підключенні зовнішньої батареї до силового входу слід використовувати обмежувач сили струму чи запобіжник.

Електричний кабель повинен мати маркування VW-1 або відповідати вимогам IEC 60332 або IEC 60695.

Для цієї камери радимо використовувати джерело живлення 24 В DC. Якщо використовується джерело живлення 12 В DC, кабель має підтримувати силу струму 14 А.

У комплект постачання входить акумуляторна пластина B-Mount. Радимо використовувати батарею 24 В (28,8 В) або батарею з підтримкою подвійної напруги.

Щоб дізнатися максимальний безперервний струм розряду батареї 12 В (14,4 В), перевірте її документацію чи маркування. Радимо використовувати таку, що має рейтинг мінімум 12 А.

Сповіщення для мешканців штату Каліфорнія

При роботі з цим обладнанням існує можливість контакту з мікродомішками багатобромистого біфеніла, що містяться в пластмасі. У штаті Каліфорнія цей елемент визнано канцерогеном, він збільшує ризик вроджених дефектів і пороків репродуктивної системи.

Додаткову інформацію див. на сайті www.P65Warnings.ca.gov

Представництво в Європі

Blackmagic Design Europe B.V.
Rijnlanderweg 766, Unit D
2132 NM Hoofddorp
NL

Blackmagic URSA Cine Immersive

Камера та акумулятор не мають піддаватися тривалому впливу прямих сонячних променів, а літій-іонні батареї слід зберігати подалі від джерел тепла.

Вихід RS (24 В) можна використовувати для забезпечення живленням аксесуарів сторонніх виробників. Споживана потужність має бути нижче 48 Вт.

Гарантія

Обмежена гарантія

Компанія Blackmagic Design гарантує відсутність у цьому виробі дефектів матеріалу та виробничого браку протягом 12 місяців від дати продажу. Якщо під час гарантійного терміну будуть виявлені дефекти, Blackmagic Design на власний розсуд виконає ремонт несправного виробу без оплати вартості запчастин і трудовитрат або замінить такий виріб на новий.

Щоб скористатися цією гарантією, споживач зобов'язаний повідомити компанію Blackmagic Design про дефект до закінчення гарантійного терміну та забезпечити умови для надання необхідних послуг. Споживач несе відповідальність за упаковку та доставку несправного виробу до відповідного сервісного центру Blackmagic Design, а також за оплату поштових витрат. Споживач зобов'язаний сплатити всі витрати на доставку, страхування, мита, податки та інші збори щодо повернення виробу незалежно від причини повернення.

Дана гарантія не поширюється на дефекти, відмови та пошкодження, що виникли через неналежне використання, неправильний догляд чи обслуговування. Компанія Blackmagic Design не зобов'язана за цією гарантією: а) усувати пошкодження, що виникли внаслідок дій із встановлення, ремонту або обслуговування виробу особами, які не є персоналом Blackmagic Design; б) усувати пошкодження, що виникли внаслідок неналежного використання або підключення до несумісного обладнання; в) усувати пошкодження або дефекти, спричинені використанням запчастин або матеріалів інших виробників; г) обслуговувати виріб, якщо його було модифіковано або інтегровано з іншим обладнанням, коли така модифікація або інтеграція збільшує час або підвищує складність обслуговування виробу.

Вплив прямих сонячних променів може призвести до пошкодження екрана URSA Viewfinder, оскільки його оптика має властивості збільшувального скла. При тривалому виведенні на екран статичних або висококонтрастних об'єктів (наприклад, рамок кадрування) дисплей може вигоряти або зберігати залишкове зображення. Щоб запобігти цьому, не закривайте інфрачервоний датчик. Якщо видошукач не використовується протягом тривалого часу, його слід вимкнути. Ця гарантія не поширюється на випадки появи залишкового зображення.

ДАНА ГАРАНТІЯ НАДАЄТЬСЯ КОМПАНІЄЮ BLACKMAGIC DESIGN ЗАМІСТЬ БУДЬ-ЯКИХ ІНШИХ ПРЯМИХ АБО ОПОСЕРЕДКОВАНИХ ГАРАНТІЙ. КОМПАНІЯ BLACKMAGIC DESIGN І ЇЇ ДИЛЕРИ ВІДМОВЛЯЮТЬСЯ ВІД БУДЬ-ЯКИХ ОПОСЕРЕДКОВАНИХ ГАРАНТІЙ КОМЕРЦІЙНОЇ ЦІННОСТІ АБО ПРИДАТНОСТІ ДЛЯ БУДЬ-ЯКОЇ ВИЗНАЧЕНОЇ ЦІЛІ. ВІДПОВІДАЛЬНІСТЬ BLACKMAGIC DESIGN ЗА РЕМОНТ АБО ЗАМІНУ НЕСПРАВНИХ ВИРОБІВ ПРОТЯГОМ ГАРАНТІЙНОГО ТЕРМІНУ ОБСЛУГОВУВАННЯ Є ПОВНИМ ТА ВИНЯТКОВИМ ЗАСОБОМ ВІДШКОДУВАННЯ, ЩО НАДАЄТЬСЯ СПОЖИВАЧЕВІ. BLACKMAGIC DESIGN НЕ НЕСЕ ВІДПОВІДАЛЬНОСТІ ЗА НЕПРЯМІ, ФАКТИЧНІ, СУПУТНІ АБО ПОДАЛЬШІ ЗБИТКИ, НЕЗАЛЕЖНО ВІД ТОГО, БУЛА КОМПАНІЯ BLACKMAGIC DESIGN (АБО ЇЇ ДИЛЕР) ПОПЕРЕДНЬО ПОВІДОМЛЕНА ПРО МОЖЛИВІСТЬ ТАКИХ ЗБИТКІВ. BLACKMAGIC DESIGN НЕ НЕСЕ ВІДПОВІДАЛЬНОСТІ ЗА ПРОТИПРАВНЕ ВИКОРИСТАННЯ ОБЛАДНАННЯ СПОЖИВАЧЕМ. BLACKMAGIC DESIGN НЕ НЕСЕ ВІДПОВІДАЛЬНОСТІ ЗА БУДЬ-ЯКІ ЗБИТКИ ВНАСЛІДОК ВИКОРИСТАННЯ ЦЬОГО ВИРОБУ. РИЗИКИ, ПОВ'ЯЗАНІ З ЙОГО ЕКСПЛУАТАЦІЄЮ, ПОКЛАДАЮТЬСЯ НА СПОЖИВАЧА.

© Copyright 2025 Blackmagic Design. Усі права захищені. Blackmagic Design, URSA, DeckLink, HDLink, Workgroup Videohub, Multibridge Pro, Multibridge Extreme, Intensity та "Leading the creative video revolution" зареєстровані як товарні знаки в США та інших країнах. Назви інших компаній і найменування продуктів можуть бути товарними знаками відповідних правовласників.

Позначення та логотипи Bluetooth® є зареєстрованими товарними знаками Bluetooth SIG, Inc., а їх використання з боку Blackmagic Design здійснюється за ліцензією. Інші товарні знаки та торгові марки є власністю відповідних власників.