



Streaming Player for OTT

User Guide

15 June 2026
1.3.0
Confidential information

Notices

Copyright

© 2026 Dolby Laboratories. All rights reserved.

Dolby Laboratories, Inc.

1275 Market Street

San Francisco, CA 94103-1410 USA

Telephone 415-558-0200

Fax 415-645-4000

<https://www.dolby.com>

Trademarks

Dolby and the double-D symbol are registered trademarks of Dolby Laboratories.

The following are trademarks of Dolby Laboratories:

Dialogue Intelligence™

Dolby®

Dolby Advanced Audio™

Dolby Atmos®

Dolby Audio™

Dolby Cinema®

Dolby Digital Plus™

Dolby Digital Plus Advanced Audio™

Dolby Digital Plus Home Theater™

Dolby Home Theater®

Dolby Theatre®

Dolby Vision®

Dolby Vision IQ™

Dolby Voice®

Feel Every Dimension™

Feel Every Dimension in Dolby™

Feel Every Dimension in Dolby Atmos™

MLP Lossless™

Pro Logic®

Surround EX™

All other trademarks remain the property of their respective owners.

Confidential information

Confidential information for Dolby Laboratories Licensees only. Unauthorized use, sale, or duplication is prohibited.

Artificial Intelligence (AI) Usage

Unauthorized use, reformatting, or ingestion for AI/ML training or reverse engineering is prohibited.

Contents

1	Browsing content.....	4
1.1	Content selection screen.....	5
1.2	Asset selection screen.....	6
2	Playback.....	8
2.1	Presentation selection.....	10
2.2	Playback information.....	10
2.3	Directplay.....	11
2.4	ADB Control.....	12
3	Debug console.....	14
3.1	Sharing logs.....	15
4	Info screen.....	16
4.1	App Information tab.....	17
4.2	Device Information tab.....	17
4.3	Audio and Video Device Capabilities tabs.....	18
4.4	Open source licenses tab.....	20
4.5	Other.....	20
4.5.1	URL Parameters docs.....	20
5	Settings.....	22
5.1	General settings.....	23
5.1.1	Web specific general settings.....	24
5.1.2	Android specific general settings.....	24
5.2	DRM configuration.....	24
5.3	SPOTT Web specific settings tabs.....	25
5.3.1	Player selection.....	25
5.3.2	Dash.JS configuration.....	25
5.3.3	Shaka Player configuration.....	26
6	Remote Control.....	27
6.1	Enable Remote Control.....	28

1

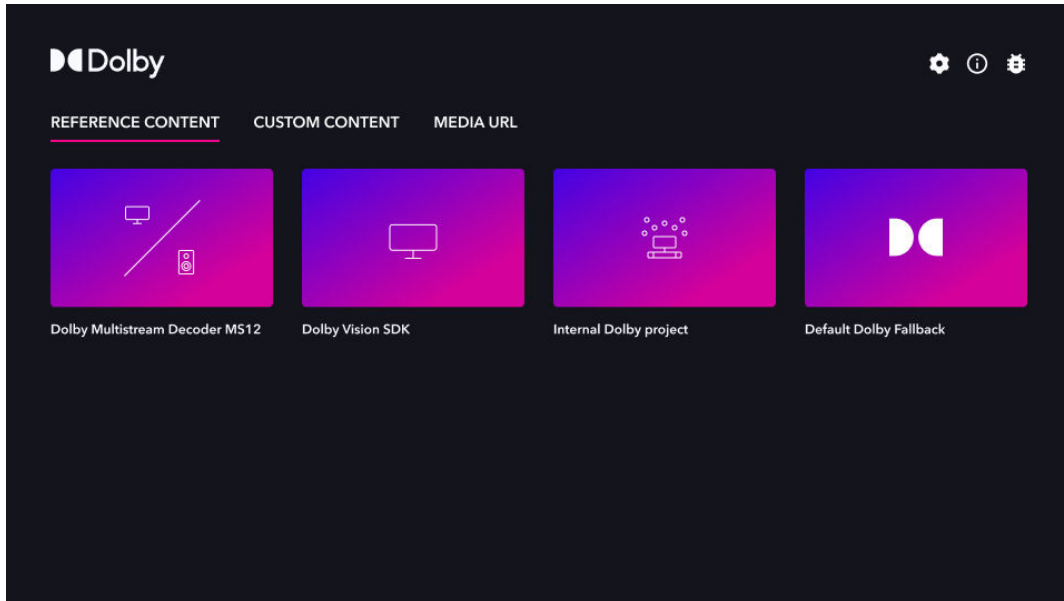
Browsing content

This section describes options to browse content in the SPOTT application.

- [Content selection screen](#)
- [Asset selection screen](#)

1.1 Content selection screen

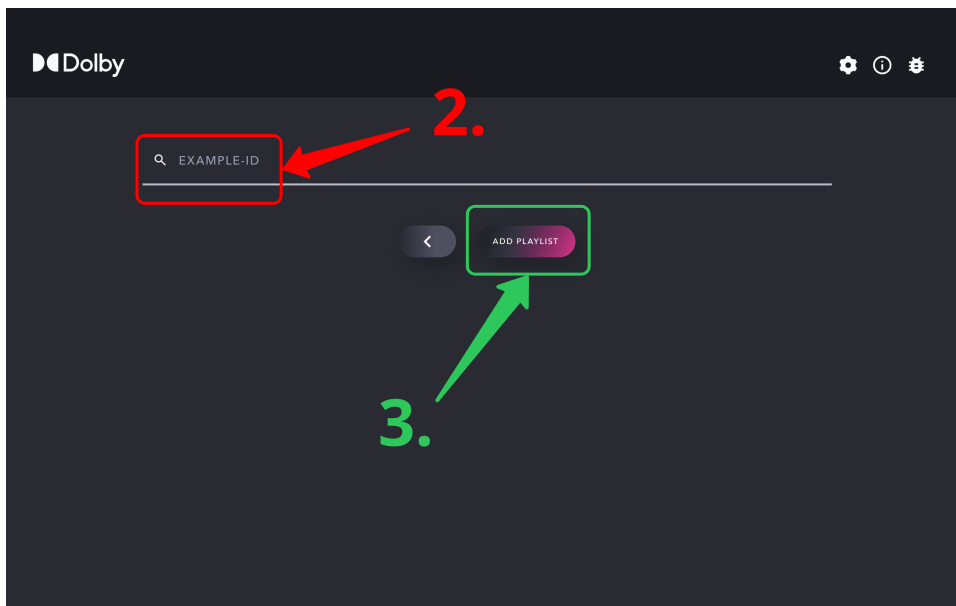
After you launch the SPOTT application for the first time, the content selection screen appears. In this screen you can browse and select content groups that are either pre-defined ("REFERENCE CONTENT" tab) or added by an application user ("CUSTOM CONTENT" tab). You can also play a stream directly from a given URL ("MEDIA URL" tab).



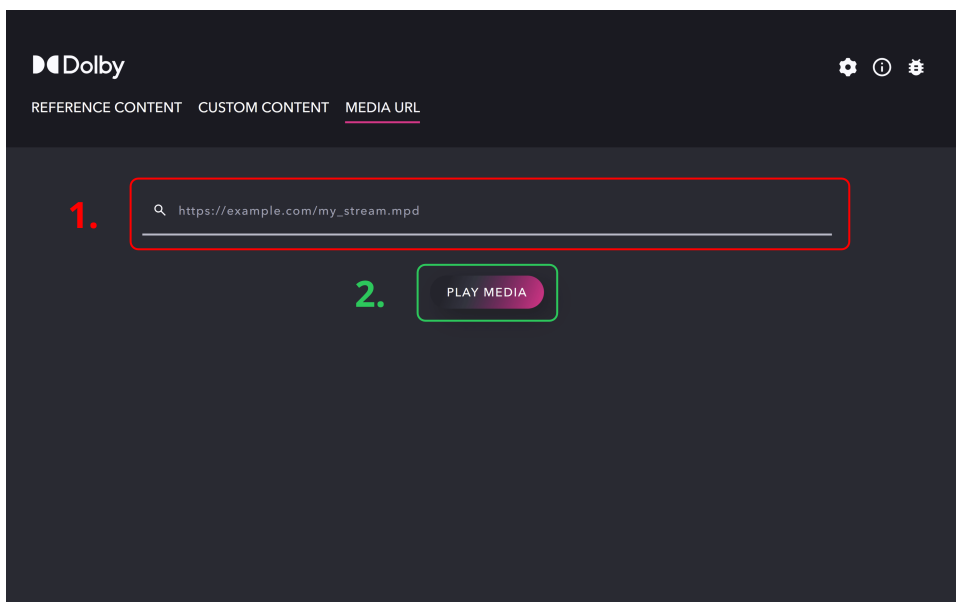
In the "REFERENCE CONTENT" tab, select content groups by navigating through content tiles.

In the "CUSTOM CONTENT" tab, add the content groups first, using special IDs.





In the "MEDIA URL" tab, enter a URL for a single stream.



Known limitations

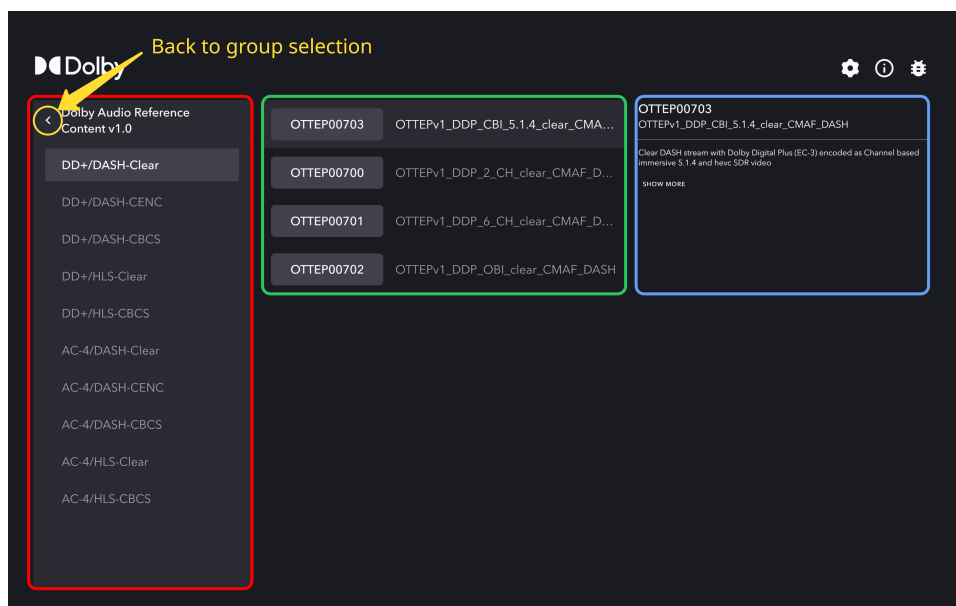
- In TV platforms, we rely on the platform-specific virtual keyboard for entering text. On some TVs (especially older models), virtual keyboard support may be limited. Currently, the only workaround is to use an external keyboard connected to the TV.

1.2 Asset selection screen

After you select a content group, the asset selection screen appears. In this screen, browse through the available test assets within the selected group and choose the one to play. The screen is divided into three main sections:

- The asset categories appear on the left side of the screen (marked with a red rectangle).
- The test assets that belong to that category appear in the middle of the screen (marked with a green rectangle).
- The test asset details appear on the right side of the screen (marked with a blue rectangle). Values such as "UNDEFINED" or "00:00" indicate that the metadata for the selected asset is incomplete or missing.

To return to the content selection screen, select the back button in the top left corner of the categories column (marked with a yellow circle).



Playback

This section describes playback in SPOTT application.

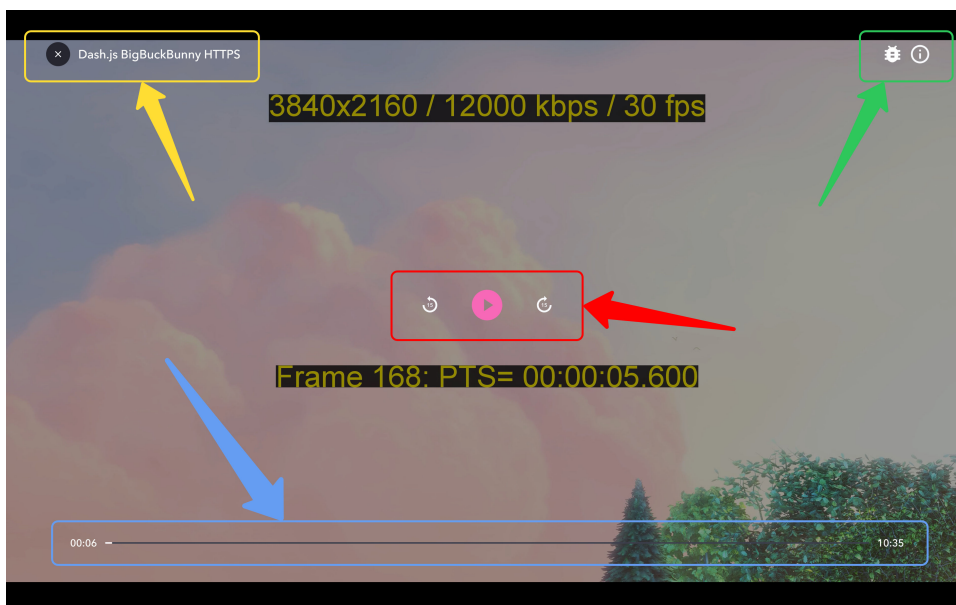
There are two ways to start playback in the SPOTT application:

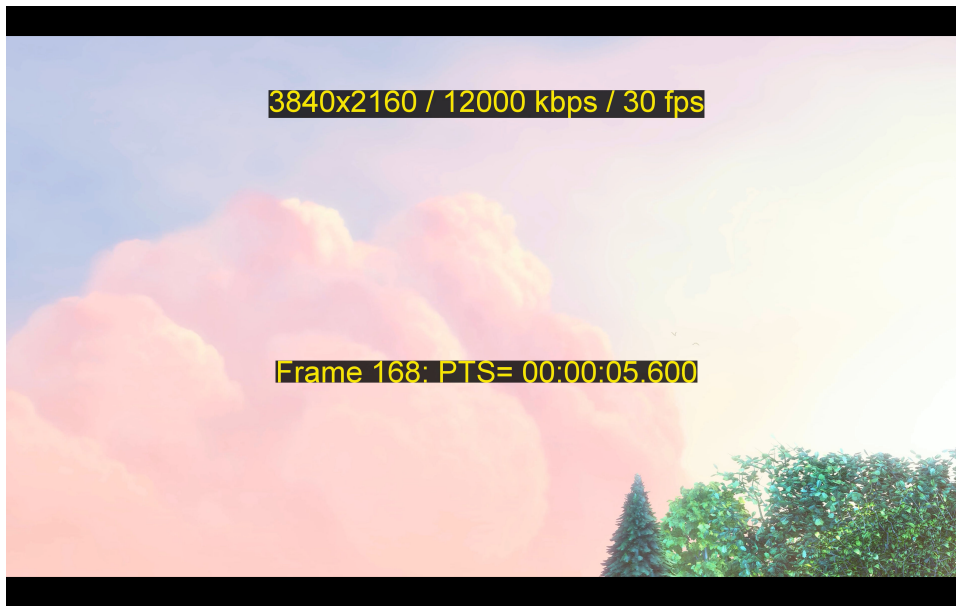
- Select an asset from the asset selection screen (see [Browsing content](#))
- Specify playback parameters using URL query parameters (see [Directplay](#) on page 11) or Android Debug Bridge CLI (see [ADB Control](#) on page 12)

After you start playback, the player screen appears. When you move the cursor or press a button, the playback controls are shown. Player controls consist of:

- play/pause and seek buttons (marked with a red square),
- a seek bar (blue square),
- exit button and title (yellow square),
- and other functional buttons such as debug console or info overlay (green square).

Player controls can also include a presentation selection button if the asset has Dolby AC-4 presentations defined and ALPS is enabled (see [Presentation selection](#) on page 10 section)





Note: In this particular example, the yellow text is part of the video signal and not displayed by the application.

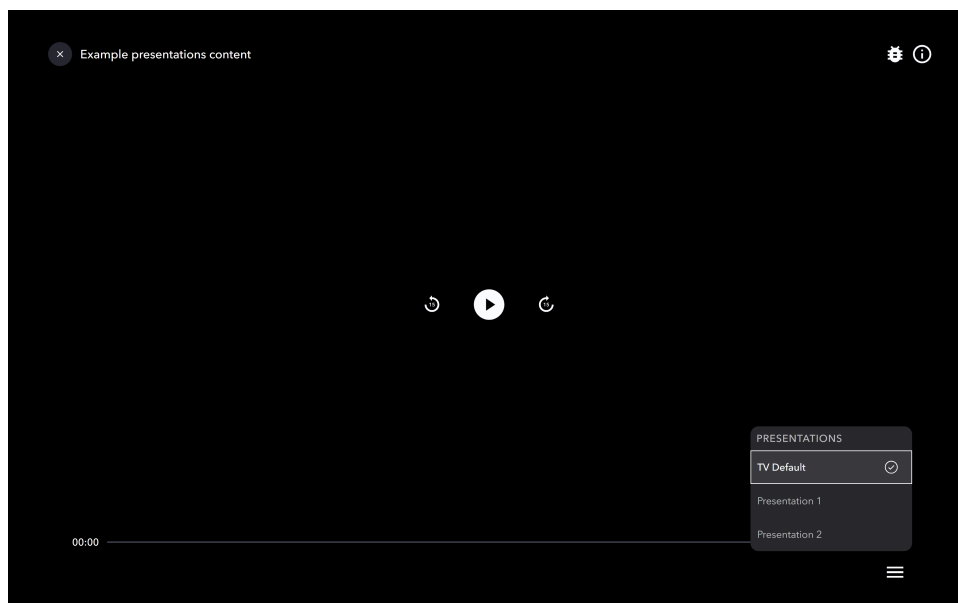
- [Presentation selection](#)
- [Playback information](#)
- [Directplay](#)
- [ADB Control](#)

2.1 Presentation selection

Dolby AC-4 content can include multiple presentations. The SPOTT application supports presentation selection through ALPS. When ALPS is enabled, a presentations list button appears under the seek bar in player controls.

Note: ALPS (Application Layer Presentation Selection) is OSS library provided by Dolby for application developers. The libraries for several platforms are available at these GitHub locations: [Android](#), [Web](#), [Native](#)

Because the application cannot determine the default presentation that is decoded by the device, the first item in the list is always "TV Default".



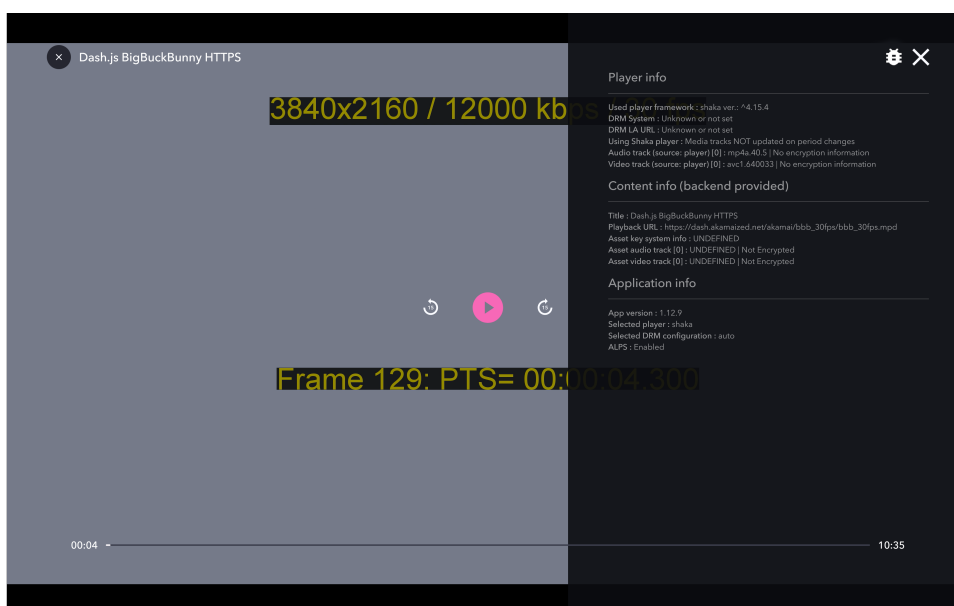
2.2 Playback information

The info button in player controls opens an overlay with playback information such as the used player framework, DRM information, title, URL, selected playback settings and more.

The "Player info" section displays information gathered by the player framework.

The "Content info (backend provided)" section displays metadata provided by the SPOTT backend for the currently playing asset.

The "Application info" section displays other info about the application state, such as the current version, the selected player, and so on.



2.3 Directplay

The directplay feature allows you to start playback of a specific asset directly, without going through the content selection and asset selection screens. You have to specify URL query parameters when you launch the application. This feature is only available in browsers.

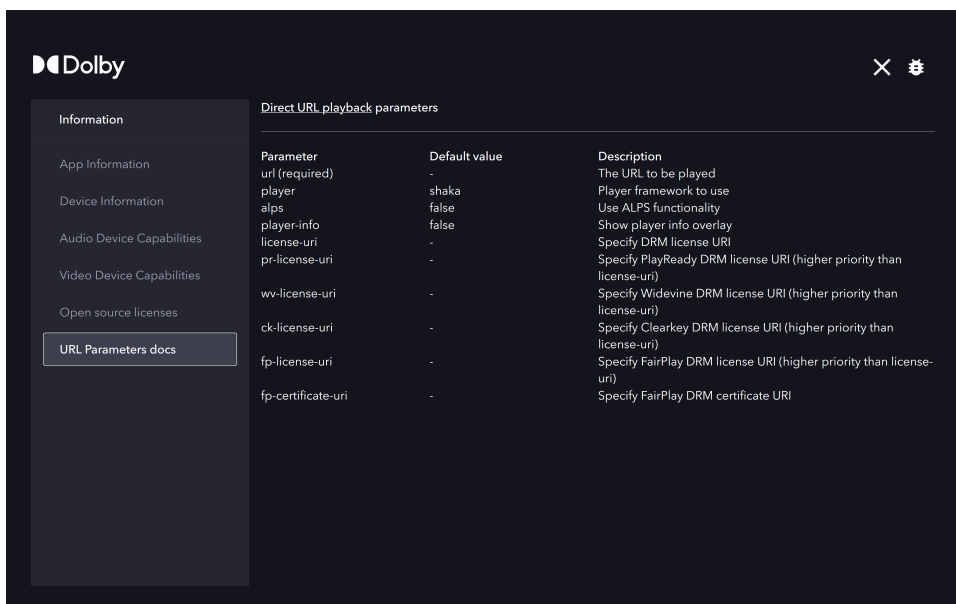
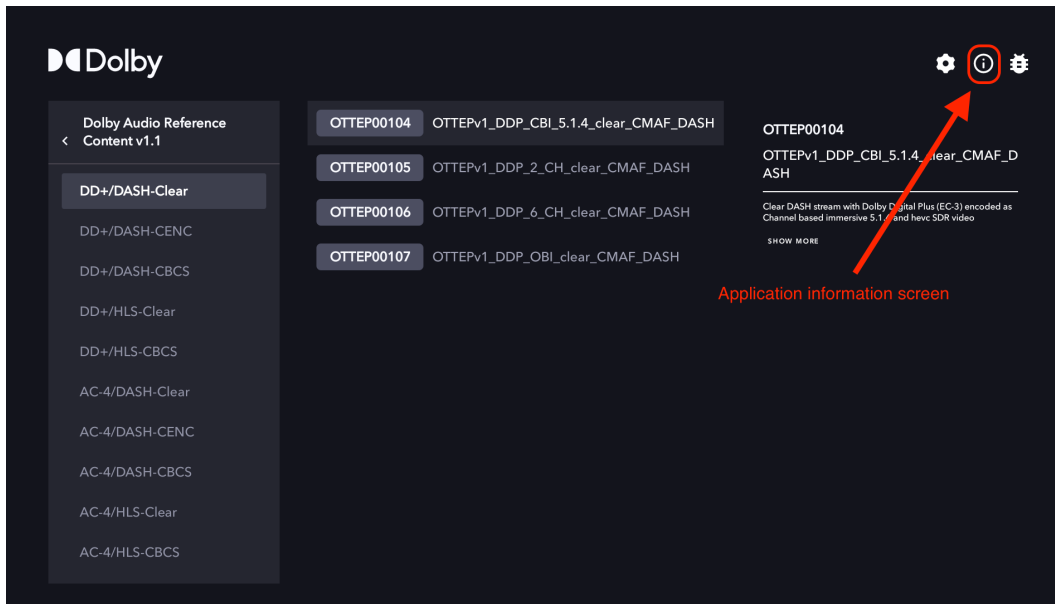
Directplay is available under the `/directplay` path of the SPOTT application. Parameters are passed after a question mark (?) as the key-value (`key=value`) pairs, separated by ampersand (&).

Currently, the following parameters are supported:

Parameter	Type	Description	Default
*url	string	URL of the media asset to play	—
player	"shaka" "dashjs" "hlsjs"	player framework to use for playback	"shaka"
alps	boolean	enable ALPS for presentation selection	false
player-info	boolean	show player information overlay	false
license-uri	string	DRM license server URI	—
pr-license-uri	string	DRM license server URI for Playready	—
wv-license-uri	string	DRM license server URI for Widevine	—
ck-license-uri	string	DRM license server URI for ClearKey	—
fp-license-uri	string	DRM license server URI for FairPlay	—
fp-certificate-uri	string	FairPlay certificate URI	—

Note: Parameters marked with an asterisk (*) are required.

Documentation about the directplay URL parameters is also available under "URL Parameters docs" in the application information screen:



This is an example directplay URL that plays the "Big Buck Bunny" video using Dash.js player: https://spott.ott.dolby.com/v1/directplay?url=https://dash.akamaized.net/akamai/bbb_30fps/bbb_30fps.mpd&player=dashjs

Known limitations

- Not all playback settings are supported in directplay mode yet, especially player-specific settings
- The documentation does currently not specify possible values for some parameters (for example, *player*)

2.4 ADB Control

If you use SPOTT for Android, you can trigger the playback remotely via the Android Debug Bridge (ADB), without interacting with the UI of the app. After you install the app on a DUT and connect to it with

```
adb connect 10.0.0.1 # IP of the DUT
```

You can trigger the playback with the following command:

```
adb shell am start \  
-n com.dolby.android.ott.spott/.MainActivity \  
-a android.intent.action.VIEW \  
-d "https://url.to.the.stream.mpd"
```

If the stream you wish to play is encrypted, a DRM configuration is required for playback. To control this configuration, set these two parameters as follows:

```
adb shell am start \  
-n com.dolby.android.ott.spott/.MainActivity \  
-a android.intent.action.VIEW -d "https://url.to.the.stream.mp4" \  
--es drm_scheme widevine \  
--es drm_license_uri "https://my.la.server.com"
```

The supported values for `drm_scheme` are

- widevine
- playready

The behavior of the DRM configuration depends on which parameters are set:

- Neither `drm_scheme` nor `drm_license_uri` is present -> The player is configured as described in the DRM section of this document, except that URL provided by the backend is not used.
- both `drm_scheme` and `drm_license_uri` are present -> The player is configured with the given DRM keysystem and LA URL.
- Only `drm_scheme` is present -> The player is configured with the given scheme, and retrieves the LA Server URL from its DASH manifest / HLS playlist.
- Only `drm_license_uri` is present -> This configuration is not supported. The `drm_license_uri` must be passed together with `drm_scheme`.

Note: When you use this method to start playback, the app behaves differently than in regular cases. After playback completes, the app closes instead of returning to the list of assets.

3

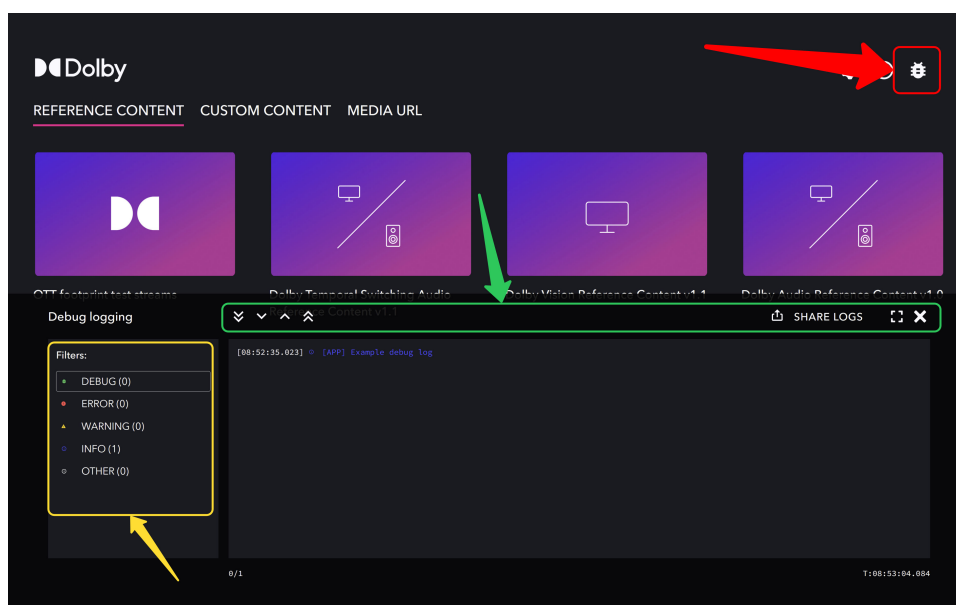
Debug console

The SPOTT application includes a debug console that you can use to view logs. The console is particularly useful when you run the application in environments where accessing logs through standard developer tools is challenging, such as smart TVs or set-top boxes.

To display the debug console, select the debug console button in the application header (marked with red square). This button is present on all screens of the application.

Use the arrow buttons in the debug console toolbar to scroll through the console content (marked with green rectangle). The toolbar also contains buttons to share logs (see [Sharing logs](#) on page 15 section), make console full screen and close the console.

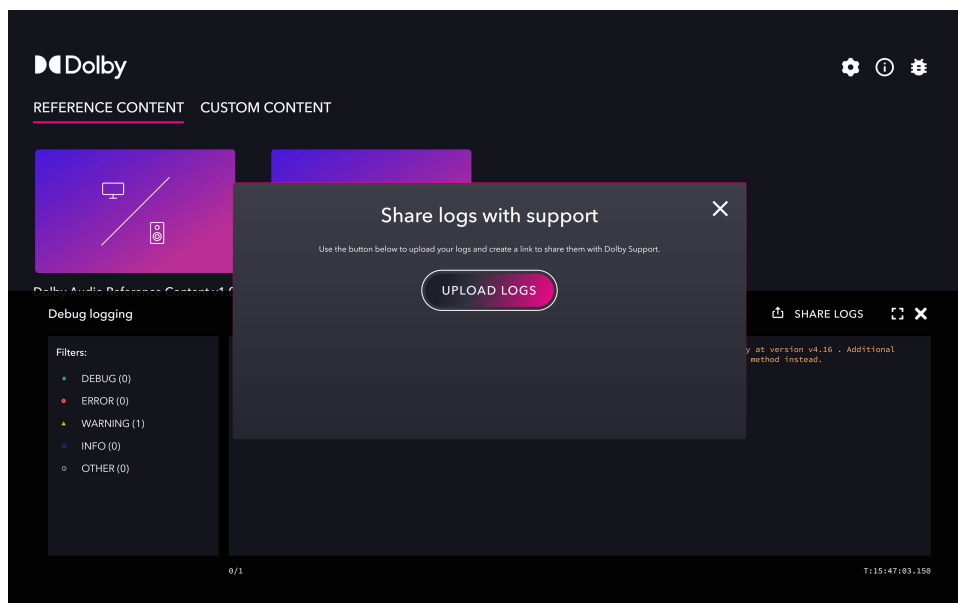
To filter the logs by log level, use the filter panel (marked with yellow rectangle). The log levels are platform-dependent.



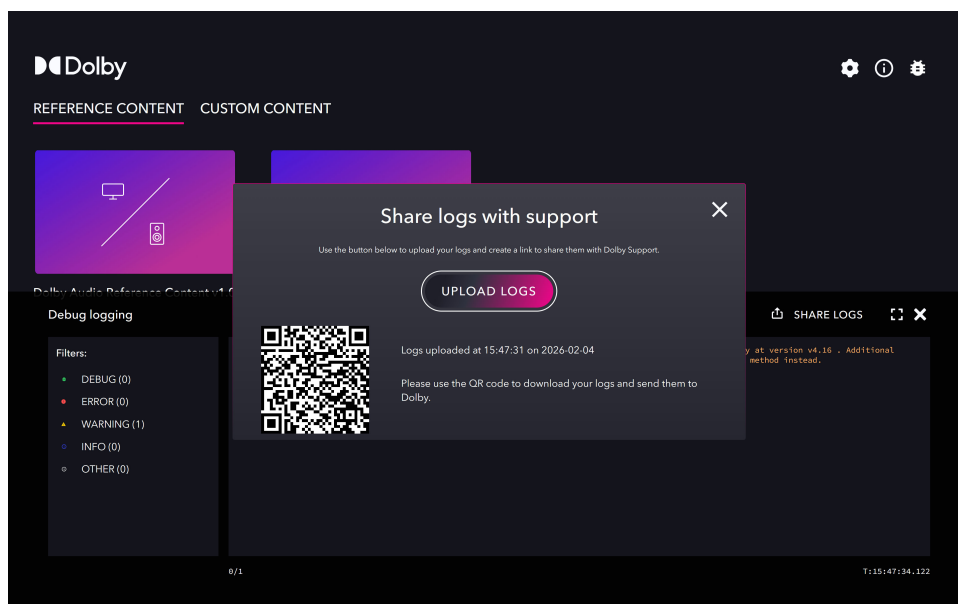
- [Sharing logs](#)

3.1 Sharing logs

To share the logs with others, use the "SHARE LOGS" button in the debug console toolbar.



The logs are uploaded to SPOTT S3 storage after you press the "UPLOAD LOGS" button in the dialog and a QR code is generated. Use the QR code to download the logs. The logs are stored for up to 24 hours.

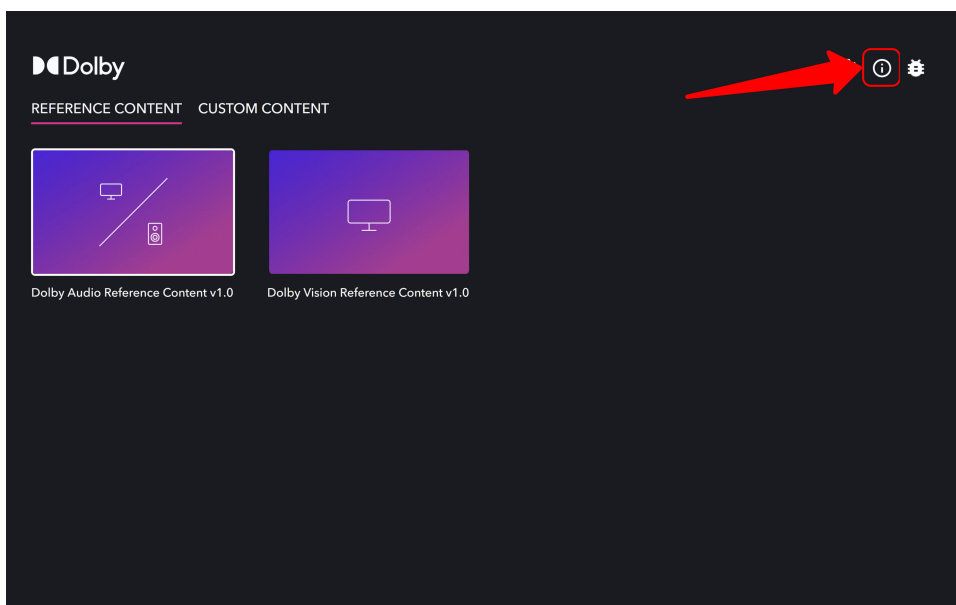


4

Info screen

The info screen provides essential information about the SPOTT application and the device it is running on.

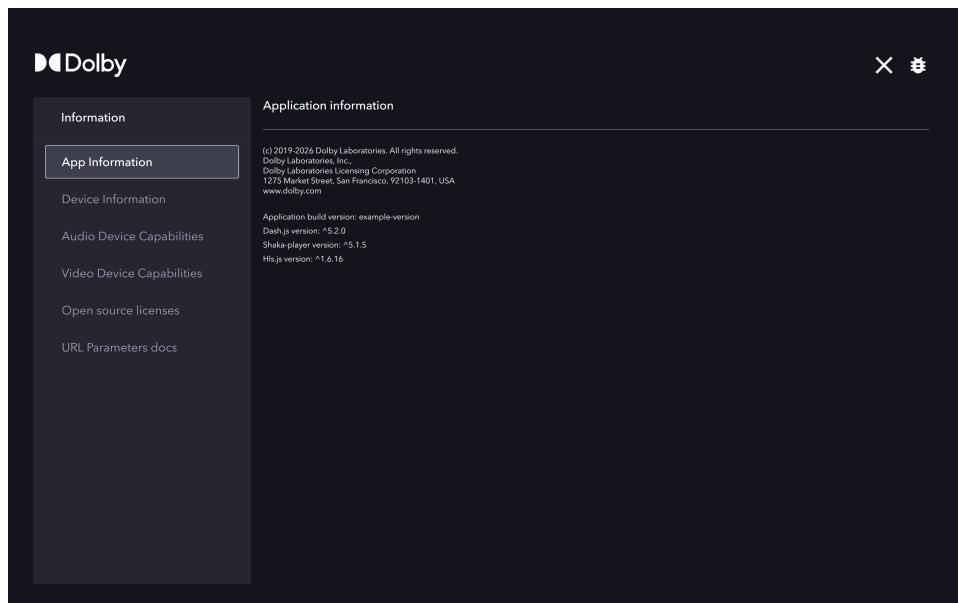
To access this screen, select the info button (marked with a red square) on the toolbar.



- [App Information tab](#)
- [Device Information tab](#)
- [Audio and Video Device Capabilities tabs](#)
- [Open source licenses tab](#)
- [Other](#)

4.1 App Information tab

"App Information" provides general information, such as application version, players versions and copyright information.

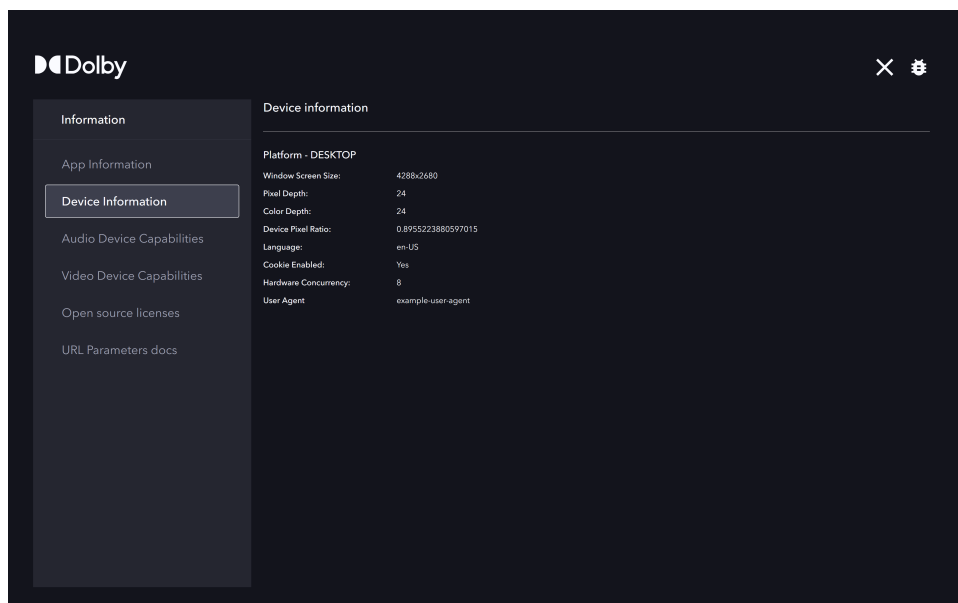


4.2 Device Information tab

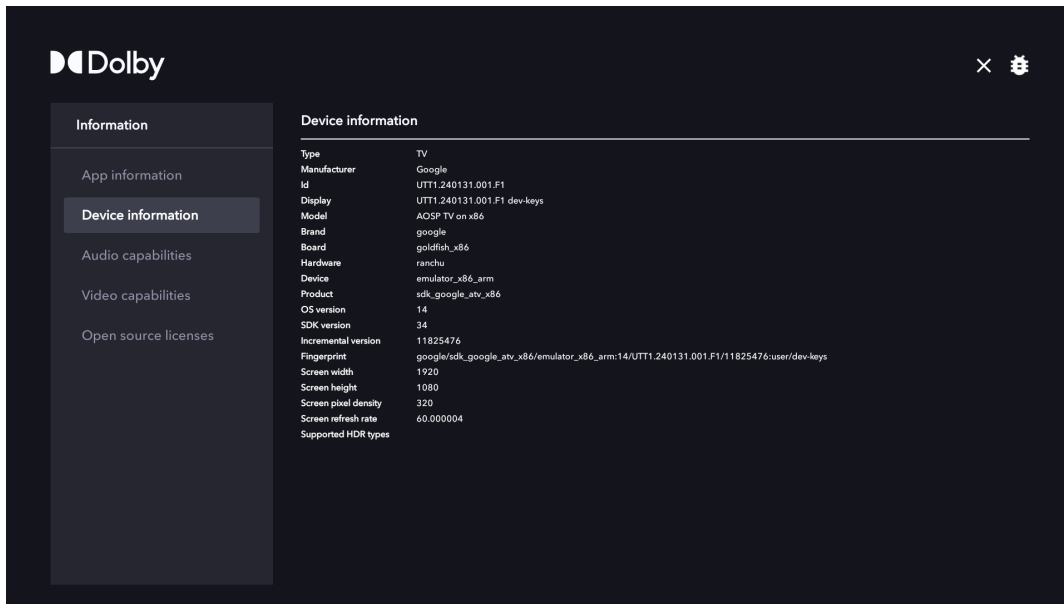
"Device Information" provides information about the device the application is running on, such as screen resolution, OS version or User Agent. The available information varies depending on the platform.

Examples for device information on different platforms:

Firefox (desktop):



Android TV (emulator):



4.3 Audio and Video Device Capabilities tabs

"Audio Device Capabilities" and "Video Device Capabilities" provide information about the audio and video capabilities of the device the application is running on. This information is based on the results of various API calls that check for support of different codec and DRM combinations.

Note: Be aware that API responses may not always be accurate and can sometimes be conflicting.

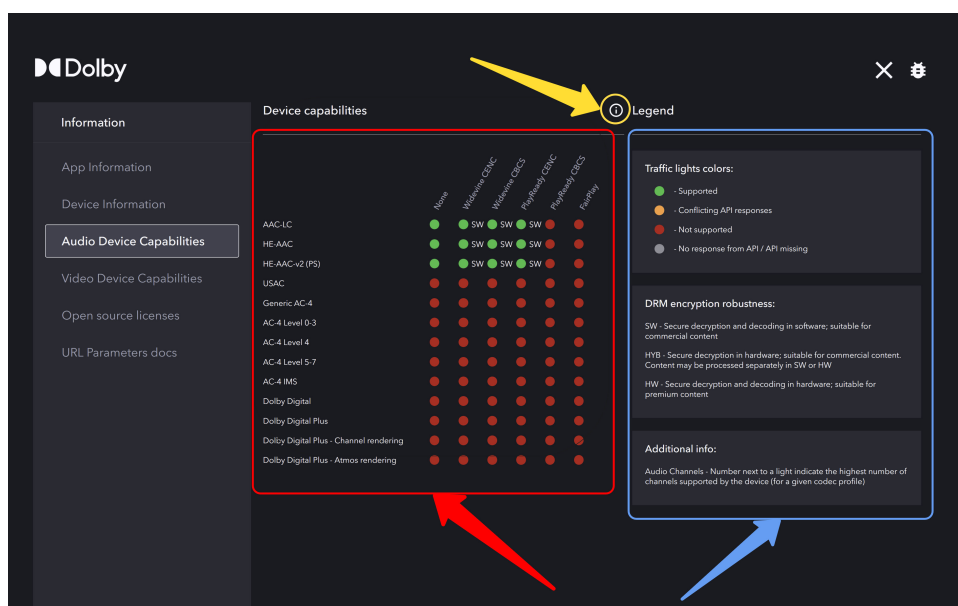
The capabilities screen has two main sections:

- in the middle, there is a board with traffic light style indicator showing support for a given codec and DRM combination (the board is marked with a red square)
- on the right side, there is a panel showing either the legend explaining the meaning of the indicators or detailed information about a selected codec/DRM combination (the panel is marked with a blue square)

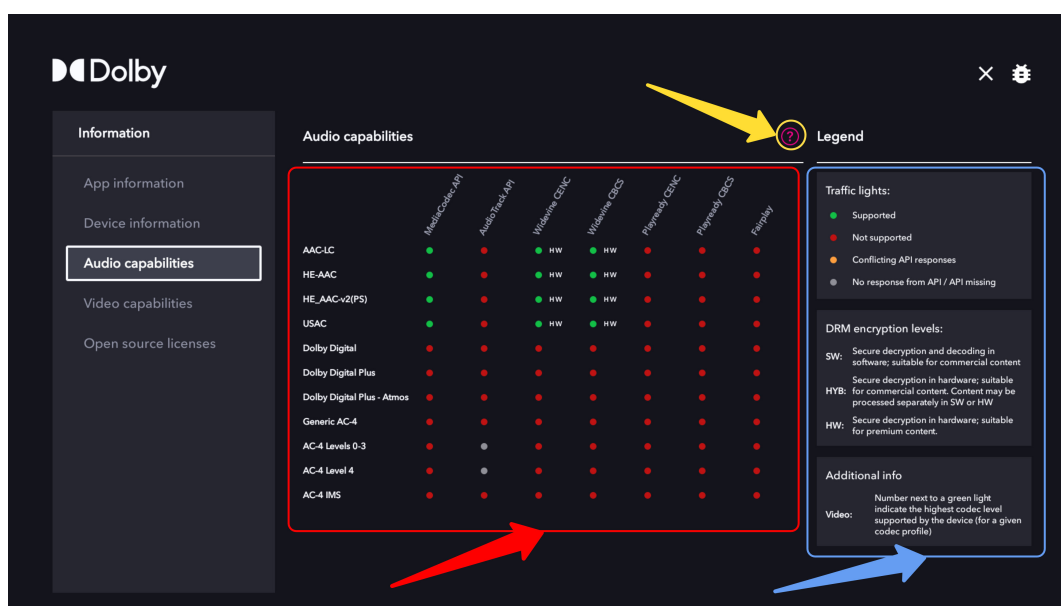
To view detailed information about a specific codec/DRM combination, select the corresponding indicator on the board. The board displays the detailed information as a JSON blob with input data and data derived from the available capability checks.

To return to the legend view, select the help icon (marked with yellow circle).

Firefox Audio Device Capabilities:



Android TV Audio Device Capabilities:



Note: Video Device Capabilities screens look similar, so this section does not include them.

Capabilities indication legend:

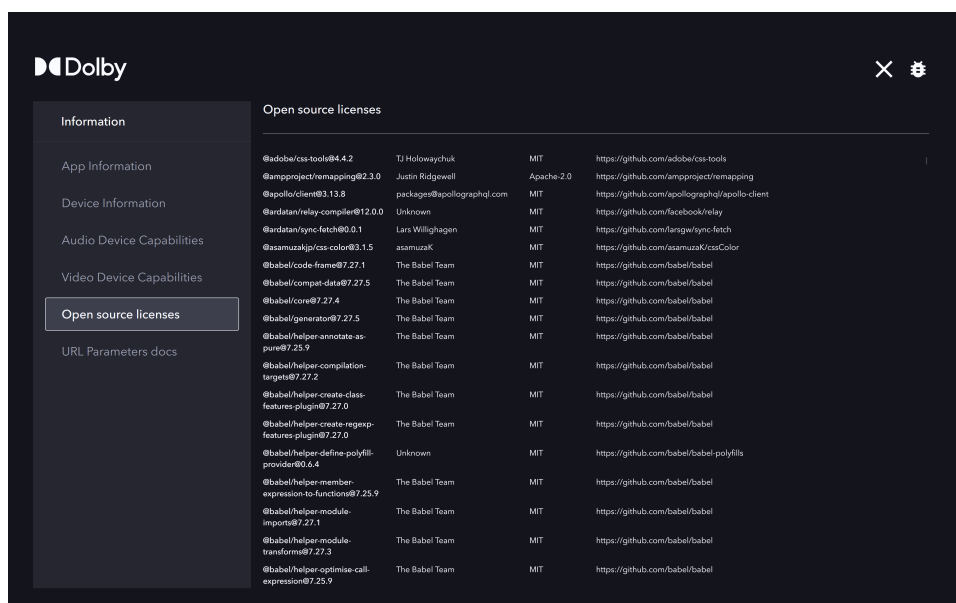
- Color:
 - Green - API responses indicate support for a codec/DRM
 - Red - API responses indicate lack of support for a codec/DRM
 - Yellow - conflicting API responses
 - Gray - no responses from API / API missing
- DRM Encryption level:
 - SW - Secure decryption and decoding in software
 - HW - Secure decryption and decoding in hardware
 - HYB - Decryption and decoding may be done separately in hardware and software
- Additional info:

- Audio - maximum number of channels supported
- Video - highest codec level supported

Both web and Android platforms use multiple APIs to gather device capabilities. In web, the results are gathered in single columns and both API responses are shown in the detailed view. In Android, the non-DRM results are split into separate columns for MediaCodec API and AudioTrack API, while DRM results are gathered in single columns similar to web.

4.4 Open source licenses tab

"Open source licenses" provides a list of open source libraries used in the SPOTT application along with their authors and licenses.



4.5 Other

4.5.1 URL Parameters docs

When the app runs in a browser, a "URL Parameters docs" tab is available in the info screen. This tab shows a list of all supported URL parameters that you can use to run playback directly from the browser address bar. Each parameter is described with its name, default value, and description.

Select the "Direct URL playback" link (marked with red square) to be redirected to directplay subpage. The parameters must be added manually.

Dolby

Information

App Information

Device Information

Audio Device Capabilities

Video Device Capabilities

Open source licenses

URL Parameters docs

Direct URL playback parameters

Parameter	Default value	Description
url (required)	-	The URL to be played
player	shaka	Player framework to use
alps	false	Use ALPS functionality
player-info	false	Show player info overlay
license-uri	-	Specify DRM license URI
pr-license-uri	-	Specify PlayReady DRM license URI (higher priority than license-uri)
ww-license-uri	-	Specify Widevine DRM license URI (higher priority than license-uri)
ck-license-uri	-	Specify Clearkey DRM license URI (higher priority than license-uri)
fp-license-uri	-	Specify FairPlay DRM license URI (higher priority than license-uri)
fp-certificate-uri	-	Specify FairPlay DRM certificate URI

5

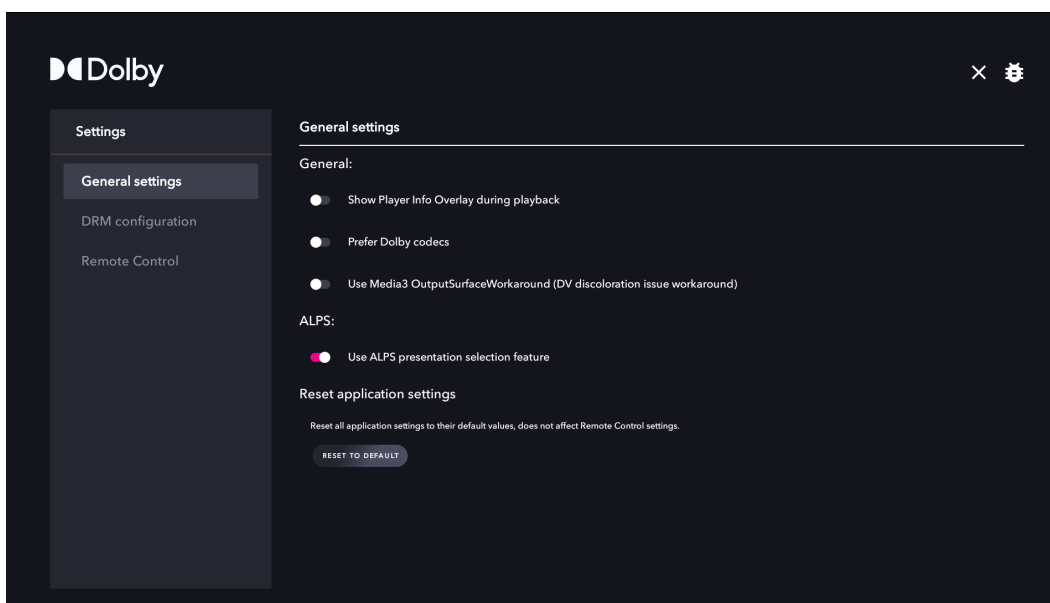
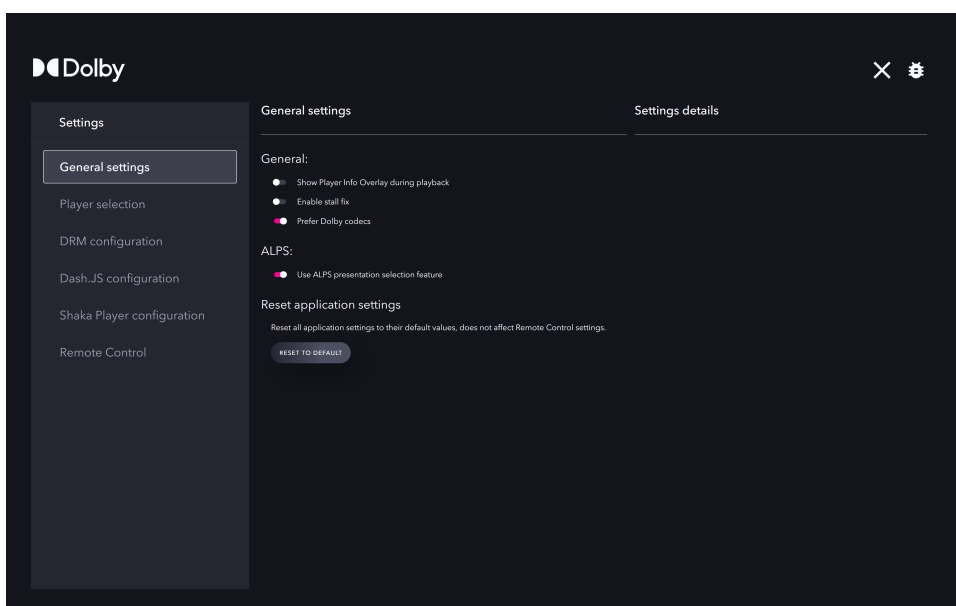
Settings

The settings screen allows you to configure playback and application options.

The available settings vary depending on the platform. Platform specific settings are highlighted.

- [General settings](#)
- [DRM configuration](#)
- [SPOTT Web specific settings tabs](#)

5.1 General settings



The general settings section allows you to configure general application options:

- Show Player Info Overlay during playback - Shows or hides the player information overlay visibility at the start of playback. You can still show and hide the overlay manually during playback.
- Prefer Dolby codecs – When enabled, instructs the active player to **prefer** Dolby audio and video codecs (for example, Dolby AC-4, Dolby Digital Plus with Dolby Atmos (E-AC-3 JOC) and Dolby Vision) over non-Dolby alternatives during track selection.
- Use ALPS presentation selection feature - Enables or disables the ALPS presentation selection feature for compatible players and streams. Currently, ALPS is supported in Dash.js, Shaka, Hls.js and ExoPlayer/Media3 players.

There is also a "**RESET TO DEFAULT**" button that resets all application settings (except for the Remote Control settings) to default values.

5.1.1 Web specific general settings

- Enable stall fix - Enables or disables the fix that performs a seek (0.1 seconds forward) when a playback stall is detected.

5.1.2 Android specific general settings

- Use Media3 OutputSurfaceWorkaround – Forces Media3 to use the 'OutputSurfaceWorkaround,' preventing the use of placeholder surfaces and eliminating the risk of incorrect dynamic surface switching. Enable this option only if a Dolby Vision discoloration issue occurs and the workaround is required. Dolby appreciates users sharing logs when this issue is encountered.

5.2 DRM configuration

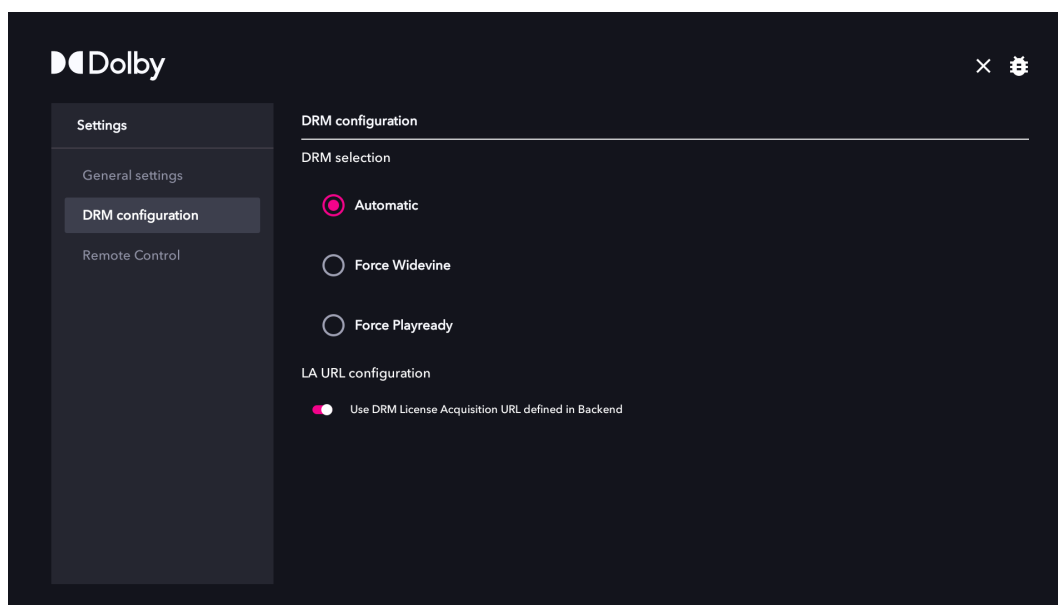
SPOTT allows you to force a specific DRM key system for playback. By default, it is set to "Automatic", which lets the player choose the key system automatically based on the manifest and capabilities.

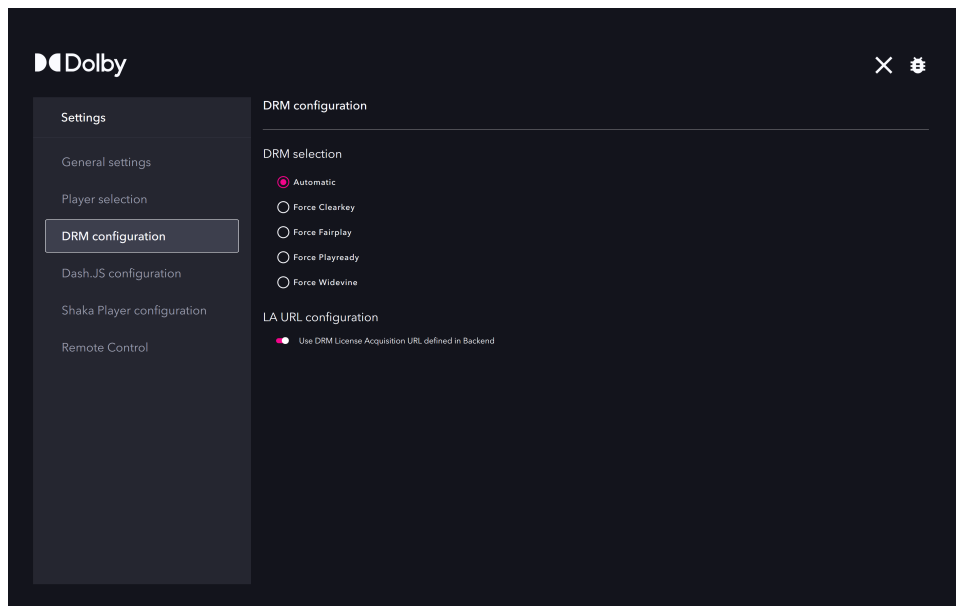
When set to "Automatic", SPOTT uses the following key system priority (from highest to lowest):

- Android:
 1. Widevine
 2. PlayReady
- Windows:
 1. PlayReady
 2. Widevine
 3. FairPlay
 4. ClearKey
- Other web platforms:
 1. Widevine
 2. PlayReady
 3. FairPlay
 4. ClearKey

You can also configure other DRM-related settings:

- Use the DRM License Acquisition URL defined in the backend - Enables or disables the use of a DRM LA URL that is defined for streams. To use an LA URL defined in the manifest, disable this option.





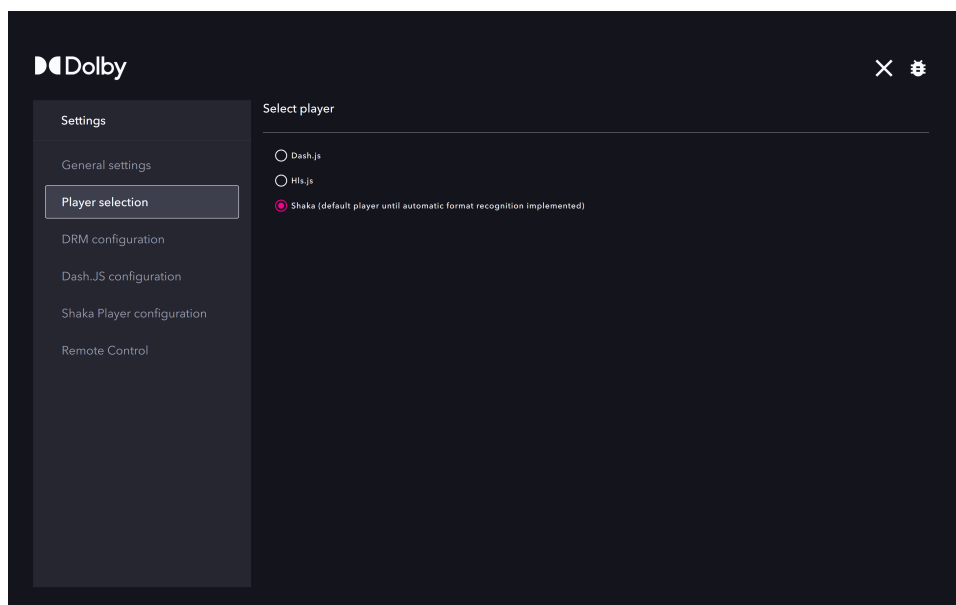
5.3 SPOTT Web specific settings tabs

5.3.1 Player selection

SPOTT Web supports multiple player frameworks:

- Dash.js
- Hls.js
- Shaka

Shaka is selected by default. To change the active player, select a player from the list.

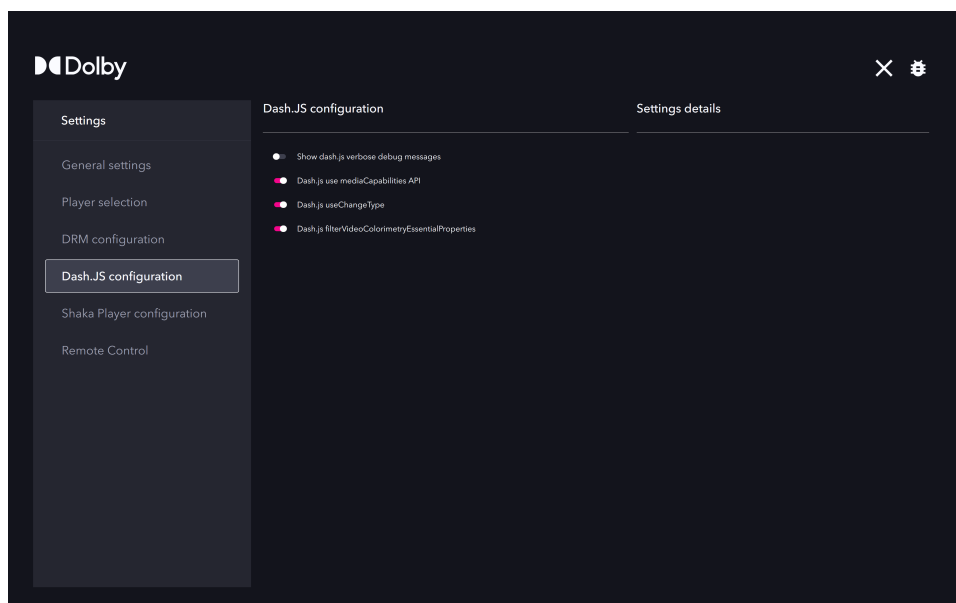


5.3.2 Dash.JS configuration

SPOTT Web allows you to customize Dash.js player settings:

- Show dash.js verbose debug messages - Enables or disables verbose debug messages from the Dash.js player.

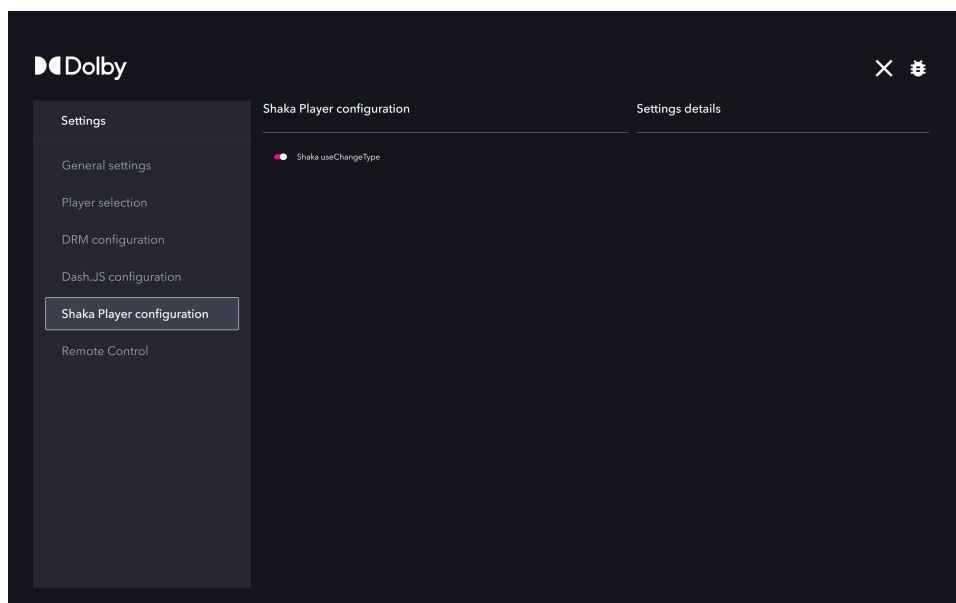
- Dash.js use mediaCapabilities API - Enables or disables the MediaCapabilities API that checks whether codecs are supported. When disabled, Dash.js uses the MSE.isTypeSupported API instead.
- Dash.js useChangeType - Enables or disables the MSE v.2 API "changeType()" before switching to a different codec family.
- Dash.js filterVideoColorimetryEssentialProperties - Enables or disables Dash.js to query MediaCapabilities API for signaled Colorimetry EssentialProperties. When disabled, registered properties per supportedEssentialProperties are allowed without any checks.



5.3.3 Shaka Player configuration

SPOTT Web allows you to customize the Shaka player settings:

- Shaka useChangeType - Enables or disables the MSE v.2 API "changeType()" usage before switching to a different codec family.





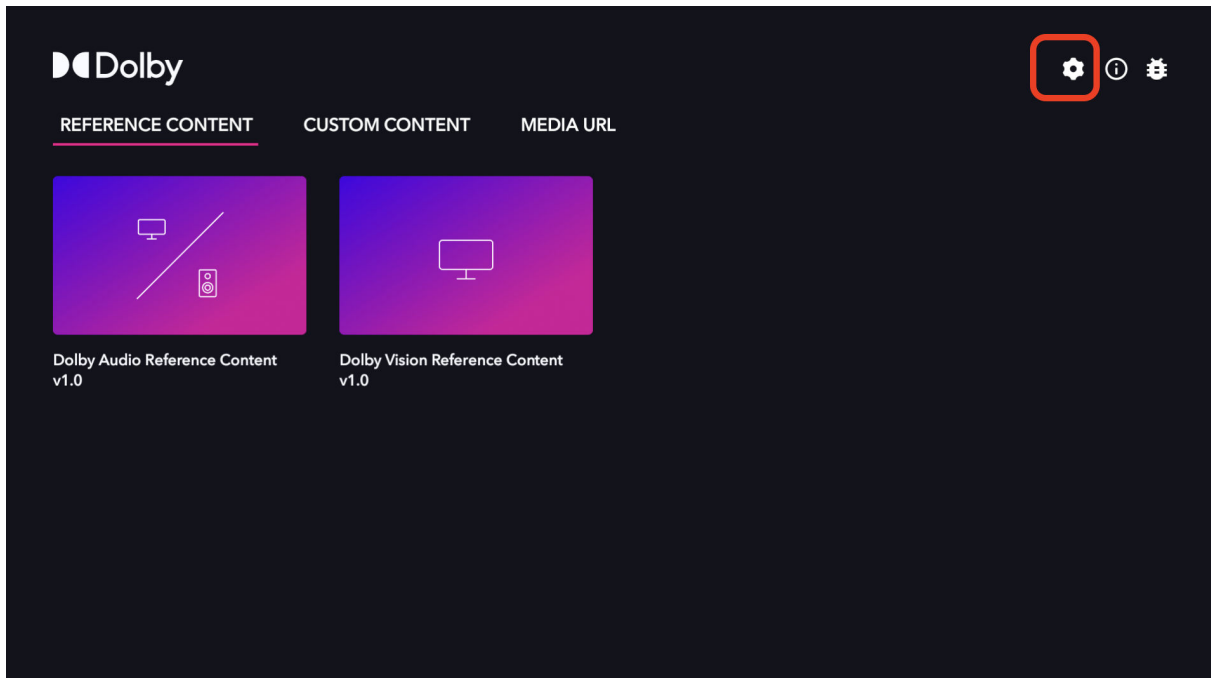
Remote Control

The Remote Control feature allows you to control the SPOTT application without physically interacting with the device it is running on.

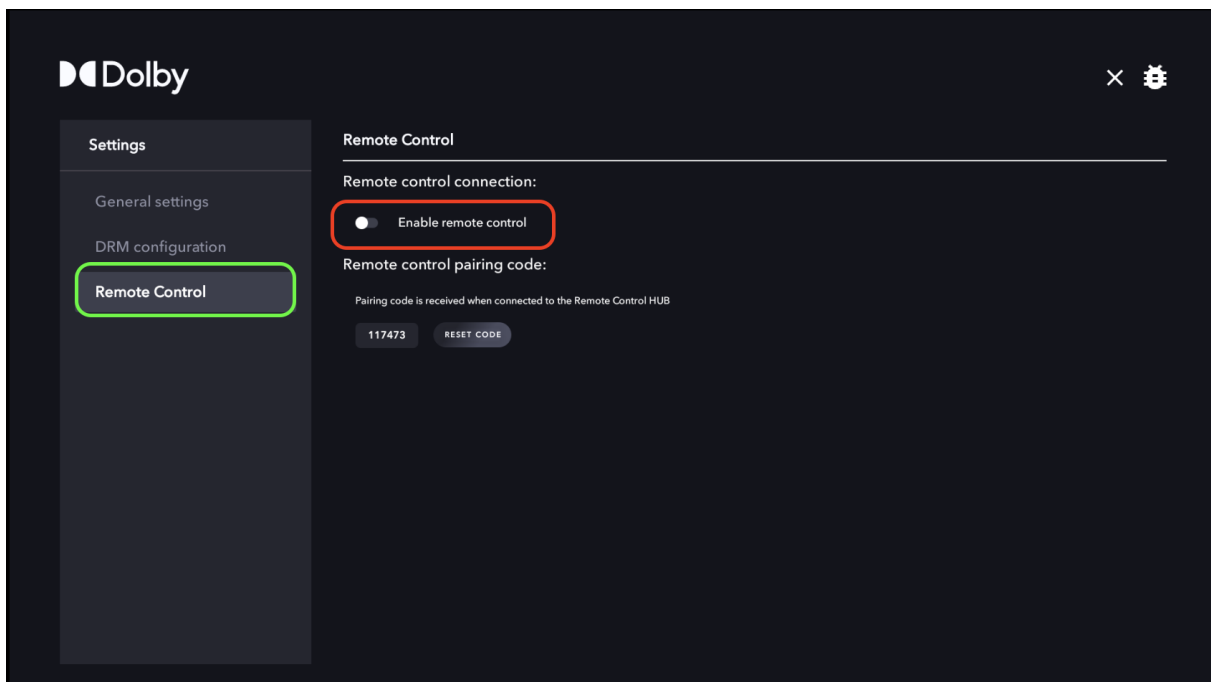
- [Enable Remote Control](#)

6.1 Enable Remote Control

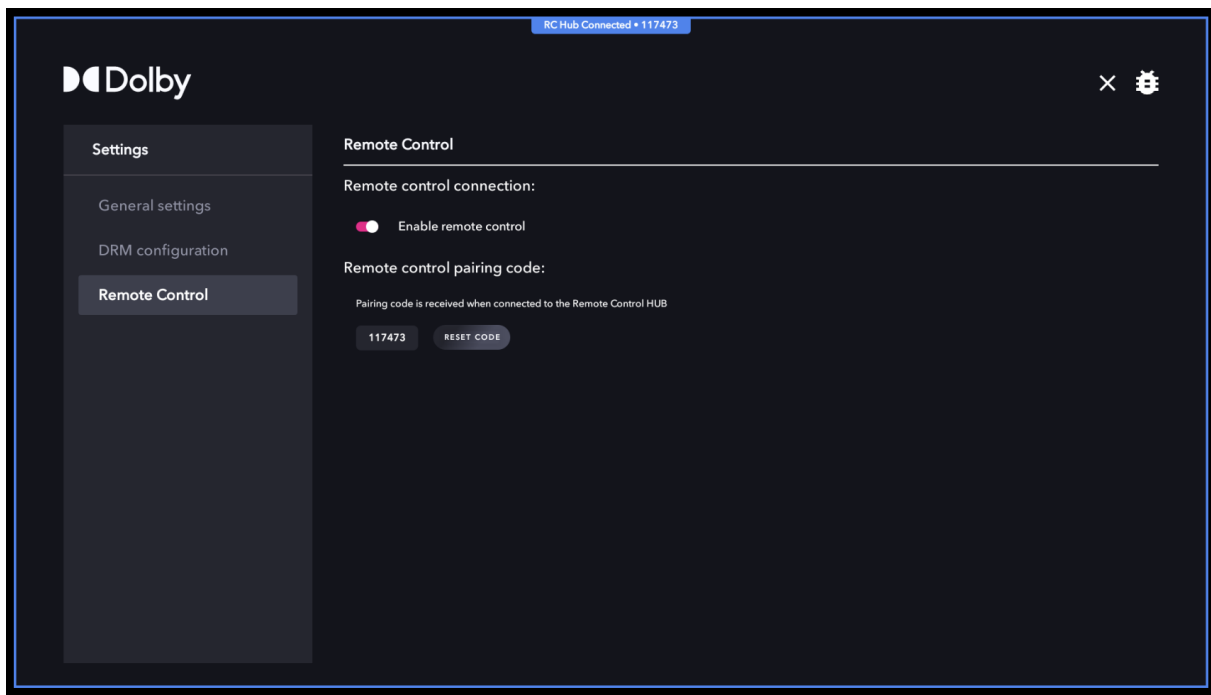
The Remote Control feature is disabled by default. To enable it, go to the Settings page



Under the Remote Control section you will find a toggle that enables the Remote Control feature.



Once the feature is enabled, a frame appears around the interface of the application, indicating the state of the remote controller.



The color of the frame indicates the state, as follows:

- Blue means the app is connected to the Remote Control server, but is not currently being controlled by any RC client application
- Green indicates there is an Remote Client application currently controlling the device
- Red means there was an error establishing a connection to the Remote Control server
- Orange indicates a transition state, either connecting to the server for the first time, or reconnecting after a connection loss

Both on the Settings page, and on the Remote Control overlay, a 6-digit pairing code visible. Enter this code into an Remote Client application to connect to the device.

Once a device receives its pairing code from the RC server, it will remember the code and attempt to use it for future Remote Control sessions. The "Reset code" button under the "Remote Control" settings section clears the saved code, and, if the RC feature is currently enabled, request a new one from the server.



Dolby Laboratories, Inc. 1275 Market Street, San Francisco, CA 94103-1410 USA.

© 2026 Dolby Laboratories. All rights reserved. Dolby and the double-D symbol are registered trademarks of Dolby Laboratories.

All other trademarks remain the property of their respective owners.