

# Introduction to the Kaltura Player

## Overview

This article includes everything you need to know to get started with the Kaltura player. Our industry-leading player offers a high-performance, fast-loading HTML5 player, with a cutting-edge user experience. Learn how to create the perfect playback experience for your users, set up monetization tracks, and optimize content with granular analytics.

Using the Kaltura Player Studio you can create, configure, update, and design media players. You can add, remove and adjust multiple buttons and features, and design a player to match the look & feel of your site. To learn more, please visit [The Kaltura Player Studio Getting Started](#).

## User Experience

The Kaltura Player provides a rich and varied interface that has all the controls expected from a video player, coupled with a set of capabilities to drive engagement, including accessibility and localization. The player may be fully customized to your needs.

The Kaltura Player provides a rich user interface including several useful and intuitive buttons for play, pause, seeking forward and backward, selecting audio and subtitles tracks, selecting quality selection, and controlling the volume.

The Kaltura Player provides indicators of total duration and current position (for VOD) and also a “Live” indication for linear streams. These features create a riveting and rich user interface.

## Player Controls

By default, the Kaltura Player has a play/pause button to control the playback, and it allows skipping backwards and forwards (10 seconds).

The Kaltura Player provides the total duration and current position indication for VOD assets (and also for live channels with a catch-up buffer).

## Quality/Speed Selection

The Kaltura Player allows you to set the quality selection, either automatically (allowing the [ABR](#) mechanism to choose the appropriate quality) or manually.

You can also set the playback speed. By default, the available speeds are x0.5, normal,

x2 and x4. However, the speeds may be customized to other playback speeds the application may allow its viewer to use.

## Picture in Picture

You may watch videos in a floating Picture in Picture (PiP) window which is always on top of other windows, even if the tab is switched. This feature allows users to continue consuming media while they interact with other content sites.

The Picture in Picture feature also works when a user scrolls down on a webpage past the video they are watching. For example, a user may continue watching a video while continuing to scroll through news articles or could switch to their email or text messaging app and respond to an email or text message, without having to pause the video.

On mobile devices, this capability leverages the native PiP functionality, which was introduced in iOS 14 and Android 8.0.

## Viewability

Viewability is how visible the player is for the viewers. This is mostly important for ads, as an ad which is less than 50% viewable (the remaining 50% is hidden or off-screen) does not count as viewed. However, this can also be used to save on bandwidth, pausing the playback when player is out of view.

The Kaltura Player provides several capabilities to make sure the player is viewable during playback:

1. **Floating player:** When this feature is enabled, once playback space has less than 50% in view, the player is minimized to the bottom right corner of the window so that the one viewing the page could still interact with both the content and the article itself.
2. **Auto-pause:** When this feature is enabled, the playback would automatically pause once it's less than 50% viewable. This also serves CDN savings.
3. **Auto-play:** The Kaltura Player has a standard auto-play feature; however, it may be set to execute the auto playback *only if the Player is in view*. Auto-play will not start if the player is located at the bottom of the page, only when a user seeks to it.

## Timeline Scrubber

The timeline component allows you to seek through content, as preview thumbnails are displayed to enrich the viewer experience.

The timeline may be [fully customized](#): timeline markers may be added by the application, and their shape and color can be customized as well as the preview image. One use case which is provided out of the box is to display the ad markers on the timeline.

However, the application can leverage this capability for various other use cases, such as chapters, highlights, and others.

## Branding

The Kaltura Player offers several branding opportunities to customize the look and feel of the UI.

## UI Customization

The Kaltura Player user interface is created using HTML and CSS, so you can [tailor the UI](#) to your needs. Any aspect of the player may be modified, the colors, the fonts, the buttons, the icons, etc.

## Localization

The Kaltura Player is fully localized. All its menus, tooltips and aria labels (for screen readers) are translated. There are translations available to all common languages. A [custom translation](#) may easily be uploaded and bundled with the player.

## Logo

Connect your users with your brand by adding your own branded logo (by default: bottom right) to the player. You can make the logo clickable, and clicking it sends the user to the given webpage URL.

## Accessibility

The Kaltura Player is fully accessible. It is possible to control it with a standard keyboard (arrows, tab, space, enter, escape). Each of its controls is set with an aria-label, allowing screen readers to render the text as speech.

The Kaltura Player provides flexibility for users that may speak a different language, or have difficulty hearing or seeing. Player flexibility also serves users that cannot use a mouse.

The Kaltura Player colors follow the color contrast guidelines to help users clearly see what is playing.

You have the ability to customize the look of the captions, as shown here.

## Audio Language

The Kaltura Player allows selecting the audio language and the language of the captions (when applicable). The selection is then saved for subsequent playbacks and is used for upcoming content.

## Responsiveness

The Kaltura Player is implemented with a responsive web design, which means it will adjust for different screen sizes automatically.

## Screen Orientation

The application can decide to lock the screen orientation on mobile browsers [that support it](#). Use this, for example, to lock the orientation to portrait, when switching to full screen on mobile Web.

## Engagement

The Kaltura Player includes several features to drive engagement and by that increase viewership.

Keep your users engrossed in your content by directing their attention after they finish watching their chosen asset and even when they interact with other content sites.

## Playlist

The Kaltura Player provides a Binge Watch experience by playing videos one after another without the viewer lifting a finger.

Other options include:

- Have an **Up Next** indication pop-up towards the end of the video showing what's coming up, show the end screen after the end of the playlist.
- Play a pre-created playlist or a dynamic one (by providing a list of entry IDs, media IDs or playback URLs). You can loop a playlist as well to keep playing the playlist again and again.

## Related Content

Keep your viewers watching by suggesting related content. Related content can be set either automatically (by tagging all related content with the same metadata) or manually (by providing a list of entry IDs, a list of media IDs, or a list of external URLs).

The functionality includes:

1. Showing the related grid upon player pause (configurable)
2. Showing the related grid when video playback ends (configurable)
3. Automatically go to next item after countdown is over

## Social Share

Encourage your users to share your videos with others, on social networks, via email, or as an embedded code. They can share videos so that the videos start from a given position of the user's choosing.

You can manage the set of sharing options, removing any and adding new custom ones.

## Playback

### Captions Support

The Kaltura Player supports rendering of subtitles and closed captions in both VOD and live content. Captions are a text version of the dialog and other sound content of the video. Subtitles are a text version of the dialog of the video. They are most often used to translate dialog into other languages to extend the reach of the video to viewers who by viewers who do not understand the language of the video.

The Kaltura Player supports several types of captions/subtitle formats, both in-band (part of the streaming manifest itself), or out-of-band (using an external file).

The Kaltura Player supports the following caption types:

Caption Format	Description
<a href="#">WebVtt</a> (.vtt)	This is the most common format used for online video. WebVtt captions can be embedded as part of the stream or be side-loaded as a separate file.

Caption Format	Description
SubRip Text (.srt)	SubRip File Format (SRT) is a common format used for online video. SRT files include only the captions without any metadata.
SMPTE-TT	A common format that supports DVB subtitles, which are common in live liner streams, as well as image-based or character-based captions which are embedded in the stream. SMPTE-TT is supported by the DASH streaming format.

## Captions Styling

**Note:** When using the Safari browser to play content, switching to full screen playback is handled by the native player. In such a case, caption styling is not in effect.

## Captions Position

The Kaltura Player provides a way to set the position of the captions on screen through the player APIs. Captions may be positioned statically in an exact location, or dynamically so that captions move up when something is shown on the bottom of the screen. This is relevant both for the Kaltura Player and the native SDK players (Android and iOS).

## Thumbnails

A thumbnail is a reduced-size image of a specific frame, which helps viewers to seek a specific event or scene in a video. For the Kaltura Player, a timeline scrubber is provided as part of the player user interface.

For the native SDK players (Android, iOS) there are no user interface components. However, the SDK player exposes the thumbnails to the application, allowing the developer to easily implement a timeline scrubber.

The Kaltura Player supports three types of thumbnails:

1. **Standard thumbnails:** For customers using Kaltura VOD transcoding. These thumbnails are generated automatically in a “just in time” manner and displayed on the player. The application can request as many thumbnails as needed at the

desired dimensions.

2. **VTT thumbnails:** VTT or WebVTT is a plain text format which is part of the HTML5 standard and also used to provide close captioning. When VTT is used for thumbnails, the VTT files contains links to the actual thumbnail's images (JPG, PNG or GIF format).
3. **DASH in-stream thumbnails:** Image based tracks in the DASH streaming format. Thumbnails are specified in the stream as a track, just like audio or video. To learn more about DASH in-stream thumbnails see [Seek Preview and Thumbnail Navigation](#) in *Guidelines for Implementation: DASH-IF Interoperability Points*.

## Casting Support

Casting allows a user to cast or send a video to another display for viewing, while continuing to use their device and without interrupting the viewing experience. The Kaltura Player supports casting content (VOD, Live, DVR) and ads. Moreover, if the stream is DRM copyright protected, it will be cast in the same protected format.

The Kaltura Player supports two popular casting technologies Google Cast and Apple AirPlay.

Casting Technology	Description
AirPlay	Allows casting to an AirPlay compatible device, available for Safari on iOS and on OSX.
Chromecast	<p>Chromecast is a device which uses the Google Cast technology to allows viewers to cast to a Chromecast-enabled device on a shared network connection. It is available for the Chrome browser on Android, Windows and Mac. It is not available for Chrome on iOS.</p> <p>When casting to a Chromecast device, a media receiver is required to ensure the same user experience on the device receiving the casting. A sample receiver can be found <a href="#">here</a>.</p>

## Monetization

Kaltura's monetization options allow you to create revenue from your your content with

video ads, while still keeping users engaged. Ads can be Skippable, Clickable, or Overlay/Banner ads, and include different capabilities depending on the type of ad.

Monetization is achieved using client-side ads or server-side ads.

- *Client-side ads* work by making calls from the video player to an ad server that responds with the ad. Client-side ads are easier to use, but they are much easier to block, by blocking the call from the video player.
- *Server-side ads* work by inserting (or “stitching”) the ad media directly into the video stream, avoiding the need for making calls to a server to receive ads. Server-side ads (stitching), is harder and more expensive to set up, but is very hard to block.

For both client-side and server-side ads, the Kaltura Player provides a set of [APIs and events](#) to control and monitor the ad playback.

## Client-side Ads

For client-side ads, the Kaltura Player integrates with the [IMA](#) (Interactive Media Ads) SDK, which allows the player to request VMAP, VAST and VPAID ads (described below).

### Ad types:

**Skippable Ad:** Can play before, during, or after the video, and allows the user to skip after X seconds.

**Clickable Ad:** Allows the user to click away from the video to the website being advertised.

**Linear or Non-Linear Ad:** An overlay or banner that appears during the video, but does not hinder watching the video.

These attributes are defined on the Ad Server and the Kaltura Player knows how to manage each of them.

**VOD content:** Client-side ads are typical for VOD streaming and the player supports *linear ads*, including: *pre-rolls*, *mid-rolls*, and *post-rolls* where ads run before, during or after the video content.

**LIVE content:** LIVE content ads are typically *non-linear* ads that run over the video content and do not interrupt the video play, or *companion ads* that appear outside the video player. An example is a banner that plays just above or below the video content.

### Ad definition files:

- **Video Multiple Ad Playlist (VMAP):** Describes the pre-rolls, mid-rolls, post-rolls, and non-linear ads



for a video and when they should be shown.

- **Video Ad Serving Template (VAST):** VMAP says there is a pre-roll, VAST describes that pre-roll. VAST allows viewer [targeting by demographics](#), such as certain positions on a webpage, or a particular page or pages on a given topic.
- **Video Player Ad-Serving Interface Definition (VPAID):** VPAID is like VAST for interactive ads.

There are [various capabilities](#) for the many types of client-side ads.

## Server-Side Ads

For server-side ads, the Kaltura Player is integrated with the [IMA-DAI](#) (Interactive Media Ads - Dynamic Ad Insertion) SDK, which enables the player to request a combined (“stitched”) stream from the Google ad stitching service, which includes both the video content and the ads.

The integration allows the player to be notified whenever an ad break is coming in the (live or VOD) stream, enabling the player to change the UI if needed (such as adding an “advertisement” banner) and also enabling to capture the ad data and send it to any analytics service.

## Ad Scheduler

The Kaltura *Ad Scheduler* in the player provides a way for the application to create its own ad break schedule. Each ad break contains one or more ads (or even an ad group known as an *add-pod*). For each ad break, multiple ad tag URLs can be provided, allowing the player to display them one after the other so that no ad impression is missed.

The Ad Scheduler also enables periodical ad playback, such as every X seconds or every X percent of the content duration (such as every 25%). The periodical ad playback can also be used for live streams, allowing to monetize such streams without the overhead of an ad stitcher.

Another feature of the ad scheduler is to [add header bidding](#) for specific ad breaks. This is done using an integration with [Prebid](#), which is an open-source header bidding solution that enables the player to request bids from multiple ad networks, take the winning bid and select its associated advertisement.

## Ad Analytics

During ad playback, various [events](#) are triggered, allowing us to monitor the ad state, its progress, and more.

The Kaltura Player has an OOTB integration with [NPAW's Youbora](#), in which the player sends all the ad events to it.

For any other analytics service, the application can subscribe to the relevant events and send them to the analytics system.

## Analytics

The Kaltura Player tracks a myriad of video events, which can (if configured to do so) be automatically sent to the pre-integrated analytics systems. Moreover, all of those events are exposed to the front-end developer, who can choose to subscribe to the relevant ones and send them to an analytics system of his choice.

Video Analytics enables many kinds of insights about the viewer's behavior (such as what kinds of assets they prefer), about the playback behavior (errors, buffering events, quality changes and more) and also about the advertisement metrics.

The Kaltura Player comes pre-integrated with the following Analytics Systems:

1. **KAVA (Kaltura Advanced Video Analytics):** Kaltura's in-house Analytics System. All player events (impressions, plays, seeks, flavor switches, etc.) are automatically sent to the Analytics System, along with multiple dimensions (playback type, device type, OS version and more).
2. **Youbora (by NPAW):** The Kaltura Player is pre-integrated on all platforms (Web, Android, iOS). You only need to set your Youbora's account code and that's it. All data will flow to Youbora.
3. **Google Analytics:** The Kaltura player is pre-integrated with Google Analytics. This includes both the Universal Analytics and the Google Analytics 4. Just provide your account/data-stream ID and all information will flow to Google.
4. **(OTT only) Bookmarks:** The Kaltura player, by default, sends beacons ("bookmarks") to the Kaltura's OTT BE (play, stop, etc.) and also periodical 'HIT's to keep track of last watched position.

To allow other custom integrations, the Kaltura Player exposes an extensive list of events for [playback](#) and for [ads](#).

These events allow easy integration with any third-party Analytics System.

[template("cat-subscribe")]

---