

# AAMP UVE – API Documentation

V2.9

## Table of Contents

|   |    |
|---|----|
| Table of Contents .....   | 0  |
| Unified Video Engine (UVE) APIs .....   | 3  |
| Overview .....  | 3  |
| Target Audience .....   | 3  |
| Roadmap .....   | 3  |
| Release Version .....   | 4  |
| Minimal Sample Player .....   | 6  |
| General Setup .....   | 6  |
| Folder Structure: Full Reference Player .....                                 | 7  |
| Universal Video Engine APIs .....   | 8  |
| PROPERTIES: .....   | 8  |
| METHODS: .....  | 8  |
| <i>load( uri, autoplay )</i> .....  | 8  |
| <i>play()</i> .....   | 8  |
| <i>pause()</i> .....  | 8  |
| <i>stop()</i> .....   | 8  |
| <i>seek( offset )</i> .....   | 9  |
| <i>getCurrentState()</i> .....  | 9  |
| <i>getDurationSec()</i> .....   | 10 |
| <i>getVolume()</i> .....  | 11 |
| <i>setVolume ( volume )</i> .....   | 11 |
| <i>setVideoMute( enabled )</i> .....  | 11 |
| <i>getPlaybackRate()</i> .....  | 11 |
| <i>setPlaybackRate( rate )</i> .....  | 11 |
| <i>getVideoBitrates()</i> .....   | 12 |
| <i>getCurrentVideoBitrate()</i> .....   | 12 |
| <i>setVideoBitrate( bitrate )</i> .....                                       | 12 |
| <i>getCurrentAudioBitrate()</i> .....   | 12 |
| <i>setVideoRect( x, y, w, h )</i> .....                                       | 13 |
| <i>setVideoZoom( videoZoom )</i> .....  | 13 |
| <i>addCustomHTTPHeader( headerName, headerValue, isLicenseRequest )</i> ..... | 13 |
| <i>removeCustomHTTPHeader( headerName )</i> .....                             | 14 |
| <i>getAvailableAudioTracks()</i> .....  | 14 |
| <i>getAvailableTextTracks()</i> .....   | 14 |
| <i>getVideoRectangle()</i> .....  | 14 |

|  |    |
|--|----|
| <i>getAudioTrack( )</i> .....  | 14 |
| <i>setAudioTrack( trackIndex )</i> .....                               | 14 |
| <i>getTextTrack( )</i> .....   | 14 |
| <i>setTextTrack( trackIndex )</i> .....                                | 15 |
| <i>setClosedCaptionStatus ( status )</i> .....                         | 15 |
| <i>getTextStyleOptions ( )</i> .....                                   | 15 |
| <i>setTextStyleOptions ( options )</i> .....                           | 15 |
| EVENTS .....   | 15 |
| <i>addEventListener( name, handler )</i> .....                         | 20 |
| <i>removeEventListener( name, handler )</i> .....                      | 20 |
| CDAI Mechanism#1 – Engine Managed CDAI .....                           | 20 |
| <i>setSubscribedTags( tagNames )</i> .....                             | 20 |
| <i>setAlternateContent( reservationObject, promiseCallback )</i> ..... | 21 |
| <i>notifyReservationCompletion( reservationId, time )</i> .....        | 21 |
| CDAI Mechanism#2 – “Player Prebuffering” Feature .....                 | 22 |
| <i>detach()</i> .....  | 22 |
| CONFIGURATION .....  | 23 |
| <i>initConfig( config )</i> .....                                      | 23 |
| <i>setDRMConfig( config )</i> .....                                    | 26 |
| Universal Video Engine Player Errors .....                             | 26 |
| Inband Closed Caption Management .....                                 | 29 |

## Unified Video Engine (UVE) APIs

### Overview

AAMP is an open source native video engine that is built on top of GStreamer and optimized for performance, memory use, and code size. AAMP Reference Player demonstrates how to use the Unified Video Engine (UVE) JavaScript binding APIs to interact with an AAMP player.

The bindings are made available in JavaScript with the help of the injectedbundle component once the DOM elements are loaded by WebKit.

### Target Audience

This document is targeted to OTT app vendors and HTML5 developers who are interested in evaluating/adopting AAMP for their media player applications on settops running RDKV based firmware.

### Features

- Formats: HLS, DASH, Fragmented MP4 HLS
- DRM Systems: Clear Key, Adobe Access, Vanilla AES-128, PlayReady, Widevine
- Captions: CEA-608/708 Captions , WebVTT

### Roadmap

- Video Guard (VGC) DRM
- DVB, EBU-TT captions

## Release Version

| S.No. | Release Version | Release Notes  |
|-------|-----------------|--|
| 1     | 0.7             | Initial draft of UVE APIs implemented  |
| 2     | 0.8             | <p>CDAI support, configuration options for tune optimization</p> <p>API:</p> <ul style="list-style-type: none"><li>• setAlternateContent</li><li>• notifyReservationCompletion</li><li>• addCustomHTTPHeader</li></ul> <p>Configuration:</p> <ul style="list-style-type: none"><li>• stereoOnly</li><li>• asyncTune</li><li>• bulkTimedMetadata</li><li>• useWesterosSink</li><li>• parallelPlaylistDownload</li></ul> <p>Events:</p> <ul style="list-style-type: none"><li>• bufferingChanged</li><li>• timedMetadata</li><li>• adResolved</li><li>• reservationStart</li><li>• reservationEnd</li><li>• placementStart</li><li>• placementEnd</li><li>• placementProgress</li><li>• placementError</li></ul> |
| 3     | 0.9             | <p>“Player Switching” Feature</p> <ul style="list-style-type: none"><li>• load (autoplay=false support)</li><li>• detach() method</li></ul>  |
| 4     | 1.0             | <p>Added support to get available audio track and closed captioning info</p> <p>API:</p> <ul style="list-style-type: none"><li>• getAvailableAudioTracks</li><li>• getAvailableTextTracks</li></ul> <p>Configuration:</p> <ul style="list-style-type: none"><li>• playlistTimeout</li><li>• parallelPlaylistRefresh</li></ul>  |

|   |     |   |
|---|-----|---|
|   |     | <ul style="list-style-type: none"> <li>• useAverageBandwidth</li> <li>• preCachePlaylistTime</li> <li>• progressReportingInterval</li> <li>• useRetuneForUnpairedDiscontinuity</li> <li>• drmDecryptFailThreshold</li> </ul>  |
| 5 | 2.4 | <p>April 2020 Release Update</p> <p>Configuration</p> <ul style="list-style-type: none"> <li>• initialBuffer</li> <li>• useMatchingBaseUrl</li> <li>• initFragmentRetryCount</li> </ul> <p>Event Notification</p>   |
| 6 | 2.6 | <p>June 2020 Release Update</p> <p>Seek while paused, get/set audio and text track supported</p> <p>API:</p> <ul style="list-style-type: none"> <li>• getAudioTrack</li> <li>• setAudioTrack</li> <li>• getTextTrack</li> <li>• setTextTrack</li> <li>• setClosedCaptionStatus</li> <li>• setTextStyleOptions</li> <li>• getTextStyleOptions</li> </ul> <p>Configuration:</p> <ul style="list-style-type: none"> <li>• nativeCCRendering</li> <li>• langCodePreference</li> <li>• descriptiveTrackName</li> </ul> |
| 7 | 2.7 | <p>Aug 2020 Release Update</p> <p>Configuration</p> <ul style="list-style-type: none"> <li>• Deprecated useWesterosSink</li> </ul>  |
| 8 | 2.9 | <p>Sept 2020 Release Update</p> <p>Configuration</p> <ul style="list-style-type: none"> <li>• authToken</li> <li>• useRetuneForGstInternalError</li> </ul>  |

## Minimal Sample Player

```
<html><head><title>IP Video Playback in WPE browser using UVE API</title></head>
  <script>
window.onload = function() {
  var player = new AAMPMediaPlayer();
  var url = "https://cpetestutility.stb.r53.xcal.tv/multilang/main.m3u8";
  player.load(url);
}
</script>
  <body>
    <div id="videoContainer">
      <video style="height:100%; width:100%; position:absolute; bottom:0; left:0">
        <source src="dummy.mp4"> <!-- hole punching -->
      </video>
    </div>
  </body>
</html>
```

## General Setup

To setup the AAMP Reference Player in RDK devices(Comcast):

- a. Host the ReferencePlayer folder in a web server.
- b. Use Comcast's IBIS tool (<https://ibis.comcast.com/app-dev-tool/send-html-app>) to launch the reference player in the device:
  - a. Under Launch HTML App, select **Select a device to get started**.
  - b. From the list, find your device (it should be registered previously).
  - c. Enter the ReferencePlayer URL in the **URL** field.
  - d. Enter any name in the **App name** field.
  - e. Click **Launch**.

## Folder Structure: Full Reference Player

- icons // UI elements of reference players and homepage
- UVE
  - index.html // Homepage of UVE reference player
  - UVEMediaPlayer.js // Includes "AAMPPlayer" JS class which wraps UVE binding object AAMPMediaPlayer
  - UVEPlayerUI.js // JS code for the UI elements and their functionality
  - UVERefPlayer.js // Main JS file
  - UVERefPlayerStyle.js // JS code for reference player and its UI
- index.html // Homepage of reference player
- ReferencePlayer.js // JS code for Homepage and redirection to respective reference players
- URLs.js // list of selectable streams
- ReferencePlayerStyle.css // CSS for Homepage and its UI



## Universal Video Engine APIs

### PROPERTIES:

| Name    | Type   | Description  |
|---------|--------|--|
| version | number | May be used to confirm if RDKV build in use supports a newer feature |

### METHODS:

#### load( uri, autoplay )

- Begin streaming.

| Name     | Type    | Description  |
|----------|---------|--|
| Uri      | String  | URI of the Media to be played by the Video Engine  |
| autoplay | Boolean | optional 2 <sup>nd</sup> parameter (defaults to true)<br><br>If false, causes stream to be prerolled/prebuffered only, but not immediately automatically presented. Available starting with version 0.8. |

#### play()

- Supported UVE version 0.7 and above.
- Start playback (if stream is in prebuffered state), or resume playback at normal speed. Equivalent to setPlaybackRate(1).

#### pause()

- Supported UVE version 0.7 and above.
- Pauses playback. Equivalent to setPlaybackRate(0).

#### stop()

- Supported UVE version 0.7 and above.
- Stop playback and free resources associated with playback.

### seek( offset )

- Supported UVE version 0.7 and above.
- Specify initial or new stream playback position. May be called prior to first load() call (or implicitly using initConfig's "offset" parameter), or while streaming.

| Name      | Type       | Description   |
|-----------|------------|---|
| offset    | Number (s) | Offset from beginning of VOD asset. For live playback, offset is relative to eldest portion of initial window. Offset value should be in seconds<br><br>Note that ability to seek is currently limited to fragment granularity. |
| keepPause | Boolean    | Flag indicates if player was in paused state before seek then maintain the same state post seek<br>Available starting with version 2.6  |

### getCurrentPosition()

- **Supported UVE version 0.7 and above.**
- Returns current playback position in seconds.

### getCurrentState()

- Supported UVE version 0.7 and above.
- Returns one of below logical player states as number:

| State Name   | Value | Semantics           | Remarks   |
|--------------|-------|---------------------|---|
| idle         | 0     | eSTATE_IDLE         | Player is idle  |
| initializing | 1     | eSTATE_INITIALIZING | Player is initializaing resources to start playback         |
|              | 2     | eSTATE_INITIALIZED  | Player completed playlist download and metadata processing. |
|              | 3     | eSTATE_PREPARING    | Create internal resources required                          |

| State Name | Value | Semantics        | Remarks   |
|------------|-------|------------------|---|
|            |       |                  | for DRM decryption and playback   |
|            | 4     | eSTATE_PREPARED  | Required resources are initialized successfully                                       |
|            | 5     | eSTATE_BUFFERING | When player does internal buffering mid-playback. Note -send out in initial buffering |
| paused     | 6     | eSTATE_PAUSED    | Indicates player is paused  |
| seeking    | 7     | eSTATE_SEEKING   | Indicates player is seeking   |
| playing    | 8     | eSTATE_PLAYING   | Indicates player has started playback   |
|            | 9     | eSTATE_STOPPING  | Not supported, for future   |
|            | 10    | eSTATE_STOPPED   | Not supported, for future   |
|            | 11    | eSTATE_COMPLETE  | When the media reaches end.   |
|            | 12    | eSTATE_ERROR     | In case any error occurred  |
|            | 13    | eSTATE_RELEASED  | Not supported, for future   |

#### getDurationSec()

- Supported UVE version 0.7 and above.
- Returns current duration of content in seconds. Duration is fixed for VOD content, but may grow with DVR content.

### getVolume()

- Supported UVE version 0.7 and above.
- Get current volume (value between 0 and 100). Default audio volume is 100. Volume is normally mapped from remote directly to TV, with video engine used to manage an independent mute/unmute state for parental control.

### setVolume ( volume )

- Supported UVE version 0.7 and above.
- Sets the current volume (value between 0 and 100). Updated value reflected in subsequent calls to getVolume()

| Name   | Type   | Description   |
|--------|--------|---|
| volume | Number | Pass zero to mute audio.<br>Pass 100 for normal (max) audio volume. |

### setVideoMute( enabled )

- Supported UVE version 0.7 and above.
- Enable or black out video for parental control purposes, default is false

| Name   | Type   | Description   |
|--------|--------|---|
| volume | Number | Pass false to black out video.<br>Pass true to resume presenting video. |

### getPlaybackRate()

- Supported UVE version 0.7 and above.
- Returns the current playback rate.

### setPlaybackRate( rate )

- Supported UVE version 0.7 and above.
- Change playback rate, supported speeds are given below -

| Value | Description                           |
|-------|---------------------------------------|
| 0     | Pause                                 |
| 1     | Normal Play                           |
| 4     | 2x Fast Forward (using iframe track)  |
| 16    | 4x Fast Forward (using iframe track)  |
| 32    | 8x Fast Forward (using iframe track)  |
| 64    | 16x Fast Forward (using iframe track) |
| -4    | 2x Rewind (using iframe track)        |
| -16   | 4x Rewind (using iframe track)        |
| -32   | 8x Rewind (using iframe track)        |
| -64   | 16x Rewind (using iframe track)       |

#### getVideoBitrates()

- Supported UVE version 0.7 and above.
- Return array of available video bitrates across profiles.

#### getCurrentVideoBitrate()

- Supported UVE version 0.7 and above.
- Return current video bitrate, as bits per second.

#### setVideoBitrate( bitrate )

- Supported UVE version 0.7 and above.

| Name    | Type   | Description   |
|---------|--------|---|
| bitrate | Number | <p>Pass bitrate from getVideoBitrates to disable ABR and lock playback to single profile.</p> <p>Pass zero to (re)enable ABR, allowing Video Engine to select from available bitrates based on network bandwidth.</p> |

#### getCurrentAudioBitrate()

- Supported UVE version 0.7 and above.
- Return current audio bitrate, as bits per second.

#### setVideoRect( x, y, w, h )

- Supported UVE version 0.7 and above.
- Set display video rectangle coordinates. Note that by default video will be fullscreen.
- Rectangle specified in “graphics resolution” coordinates (coordinate space used by graphics overlay).
- Window size is typically 1280x720, but can be queried at runtime as follows:  
var w = window.innerWidth || document.documentElement.clientWidth || document.body.clientWidth;  
var h = window.innerHeight || document.documentElement.clientHeight || document.body.clientHeight;

| Name | Type   | Description             |
|------|--------|-------------------------|
| X    | Number | Left position for video |
| Y    | Number | Top position for video. |
| W    | Number | Video width.            |
| H    | Number | Video height.           |

#### setVideoZoom( videoZoom )

- Supported UVE version 0.7 and above.
- Set video zoom, by default its set to “full”

| Name      | Type   | Description  |
|-----------|--------|--|
| videoZoom | String | “none” to disable video zoom mode. “full” to enable video zoom mode. |

#### addCustomHTTPHeader( headerName, headerValue, isLicenseRequest )

- Supported UVE version 0.8 and above.
- Add custom headers to HTTP requests

| Name             | Type         | Description  |
|------------------|--------------|--|
| headerName       | String       | HTTP header name   |
| headerValue      | String Array | HTTP header value  |
| isLicenseRequest | Boolean      | (defaults to false) indicates if the HTTP header is for exclusive use with PlayReady/Widevine license requests |

#### **removeCustomHTTPHeader( headerName )**

- Supported UVE version 0.8 and above.
- Remove a custom header set previously. If called with no arguments, will remove all custom headers.

| Name       | Type   | Description      |
|------------|--------|------------------|
| headerName | String | HTTP header name |

#### **getAvailableAudioTracks()**

- Supported UVE version 1.0 and above.
- Returns the available audio tracks information in the content.

#### **getAvailableTextTracks()**

- Supported UVE version 1.0 and above.
- Returns the available text tracks(CC) in the content.

#### **getVideoRectangle()**

- Supported UVE version 1.0 and above.
- Returns the current video rectangle co-ordinates.

#### **getAudioTrack( )**

- Supported UVE version 2.6 and above.
- Returns the index of current audio track in available audio track list.

#### **setAudioTrack( trackIndex )**

- Supported UVE version 2.6 and above.
- Set the audio track at trackIndex in available audio track list.

| Name       | Type   | Description  |
|------------|--------|--|
| trackIndex | Number | Index of desired audio track in available audio track list |

#### **getTextTrack( )**

- Supported UVE version 2.6 and above.
- Returns the index of current text track in available text track list.

### setTextTrack( trackIndex )

- Supported UVE version 2.6 and above.
- Set the text track at trackIndex in available text track list.

| Name       | Type   | Description  |
|------------|--------|--|
| trackIndex | Number | Index of desired text track in available text track list |

### setClosedCaptionStatus ( status )

- Supported UVE version 2.6 and above.
- Set the ClosedCaption rendering to on/off.

| Name   | Type    | Description                            |
|--------|---------|--|
| status | Boolean | To turn on/off ClosedCaption rendering |

### getTextStyleOptions ( )

- Supported UVE version 2.6 and above.
- Returns the JSON formatted string of current ClosedCaption style options and values.

### setTextStyleOptions ( options )

- Supported UVE version 2.6 and above.
- Set the ClosedCaption style options to be used for rendering.

| Name    | Type   | Description   |
|---------|--------|---|
| options | String | JSON formatted string of different rendering style options and its values |

## EVENTS

| Event Name      | Event Payload | Description |
|-----------------|---------------|-------------|
| playbackStarted |               |             |



|                        |  |   |
|------------------------|--|---|
|                        |  | <ul style="list-style-type: none"> <li>- Supported UVE version 0.7 and above.</li> <li>- fired when playback starts</li> </ul>  |
| playbackStateChanged   | state: number  | <ul style="list-style-type: none"> <li>- Supported UVE version 0.7 and above.</li> <li>- fired as state changes across play/pause seek/not-seek quadruplet</li> </ul>                                     |
| playbackProgressUpdate | durationMilliseconds:<br>number,<br>positionMilliseconds:<br>number,<br>playbackSpeed:<br>number,<br>startMilliseconds:<br>number,<br>endMilliseconds:<br>number | <ul style="list-style-type: none"> <li>- Supported UVE version 0.7 and above.</li> <li>- fired based on the interval set</li> </ul>   |
| bufferingChanged       | buffering: bool  | <ul style="list-style-type: none"> <li>- Supported UVE version 0.8 and above.</li> <li>- fired when AAMP encounters buffering mid-playback, buffering flag indicates buffering status (on/off)</li> </ul> |
| playbackCompleted      |  | <ul style="list-style-type: none"> <li>- Supported UVE version 0.7 and above.</li> <li>- fired when there is nothing left to play</li> </ul>  |

|                      |  |   |
|----------------------|--|---|
| playbackSpeedChanged | speed: number,<br>reason: string   | - Supported UVE version 0.7 and above.  |
| playbackFailed       | shouldRetry: boolean,<br>code: number,<br>description: string  | - Supported UVE version 0.7 and above.<br>- fired when an error occurs  |
| decoderAvailable     | decoderHandle:<br>number   | - Supported UVE version 0.7 and above.<br>- fired when video decoder handle becomes available, required for closedcaption parsing + rendering by RDK ClosedCaptions module                              |
| mediaMetadata        | durationMiliseconds:<br>number,<br>languages: string[],<br>bitrates: number[],<br>playbackSpeeds:<br>number[],<br>width: number,<br>height: number,<br>hasDrm: boolean | - Supported UVE version 0.7 and above.<br>- fired with metadata of the asset currently played, includes duration(in ms), audio language list, available bitrate list, hasDrm, supported playback speeds |
| speedsChanged        | playbackSpeeds:<br>number[]  | - Supported UVE version 0.7 and above.<br>- fired when supported playback speeds changes (based on iframe availability)   |

|                    |   |   |
|--------------------|---|---|
| vttCueDataListener | start: number,<br>duration: number,<br>text: string   | <ul style="list-style-type: none"> <li>- Supported UVE version 0.7 and above.</li> <li>- fired for VTT cue parsed from the WebVTT playlist in the asset</li> </ul>  |
| drmMetadata        | code: number,<br>description: string  | <ul style="list-style-type: none"> <li>- Supported UVE version 0.7 and above.</li> <li>- fired when there is a change in DRM metadata (especially expiration of DRM auth data)</li> </ul>                 |
| enteringLive       |   | <ul style="list-style-type: none"> <li>- Supported UVE version 0.7 and above.</li> <li>- fired when entering live point of a live playlist during/after a seek/trickplay operation</li> </ul>             |
| timedMetadata      | time: number,<br>duration: number,<br>name: string,<br>content: string,<br>type: number,<br>metadata: object,<br>id: string             | <ul style="list-style-type: none"> <li>- Supported UVE version 0.8 and above.</li> <li>- fired when a subscribed tag is found in the playlist</li> </ul>  |
| bitrateChanged     | time: number,<br>bitRate: number,<br>description: string,<br>width: number,<br>height: number,<br>framerate: number<br>position: number | <ul style="list-style-type: none"> <li>- Supported UVE version 0.7 and above.</li> <li>- fired when video profile is switched by ABR with the metadata associated with newly selected profile.</li> </ul> |
| adResolved         | resolvedStatus: bool,   | <ul style="list-style-type: none"> <li>- Supported UVE version 0.8 and above.</li> </ul>  |

|                   |  |  |
|-------------------|--|--|
|                   | placementId: string,<br>placementStartTime:<br>number,<br>placementDuration:<br>number | - Confirmation that an upcoming ad's main manifest has been successfully downloaded and parsed.  |
| reservationStart  | adbreakId: string,<br>time: number   | - Supported UVE version 0.8 and above.<br>- Sent upon playback into an ad break (one or more ads).   |
| reservationEnd    | adbreakId: string,<br>time: number   | - Supported UVE version 0.8 and above.<br>- Sent upon completion of an ad break (back to main content) - it is NOT sent (per previously agreed contract) if user does trickplay or seek to abort ad playback         |
| placementStart    | adId: string,<br>time: number  | - Supported UVE version 0.8 and above.<br>- This is sent in real time when injecting first frame of a new ad on content->ad or ad->ad transition. Should be accurate compared to onscreen frames.                    |
| placementEnd      | adId: string,<br>time: number  | - Supported UVE version 0.8 and above.<br>- This is sent in real time after passively playing to end of an ad - it is NOT sent (per previously agreed contract) if user does trickplay or seek to abort ad playback. |
| placementProgress | adId: string,<br>time: number  | - Supported UVE version 0.8 and above.<br>- Sent periodically while ad is being played   |

|                |   |   |
|----------------|---|---|
|                |   | out, giving an estimate percentage-watched metric. It's interpolated based on elapsed time, and should repeat same value if paused. |
| placementError | adId: string,<br>time: number,<br>error: number | - Supported UVE version 0.8 and above.<br>- Generated only for exception while attempting to play out ad content.                   |

#### addEventListener( name, handler )

| Name    | Type     | Description                    |
|---------|----------|--------------------------------|
| name    | String   | Event Name                     |
| handler | Function | Callback for processing event. |

#### removeEventListener( name, handler )

| Name    | Type     | Description                    |
|---------|----------|--------------------------------|
| Name    | String   | Event Name                     |
| handler | Function | Callback for processing event. |

### CDAI Mechanism#1 – Engine Managed CDAI

Supported for DASH Linear, working with period structure and SCTE35 markers, with optional replacement for like-amount of content.

#### setSubscribedTags( tagNames )

- Supported UVE version 0.8 and above.
- Subscribe to specific tags / metadata in manifest

| Name     | Type         | Description                    |
|----------|--------------|--------------------------------|
| tagNames | String Array | List of tag names of interest. |

|  |  |  |
|--|--|--|
|  |  | Examples:<br>#EXT-X-IDENTITY-ADS<br>#EXT-X-MESSAGE-REF<br>#EXT-X-CUE<br>#EXT-X-ASSET-ID<br>#EXT-X-TRICKMODE-RESTRICTION<br>#EXT-X-CONTENT-IDENTIFIER |
|--|--|--|

#### setAlternateContent( reservationObject, promiseCallback )

- Supported UVE version 0.8 and above.

| Name              | Type     | Description  |
|-------------------|----------|--|
| reservationObject | Object   | <p>reservationObject provides context for alternate content to be played out at ad opportunities.</p> <pre> {   "reservationId": "1234", // period id from DASH manifest   "reservationBehavior": number,   "placementRequest":   {     // uuid generated to identify this placement     "id": string,      // position at which placement will begin playback on the main timeline     "pts": number,      "url": "",   }, } </pre> |
| promiseCallback   | Function | Signals success/failure while retrieving ad manifest and preparing for playback.   |

#### notifyReservationCompletion( reservationId, time )

- Supported UVE version 0.8 and above.

- Notify video engine when all ad placements for a particular reservation have been set via setAlternateContent.

| Name          | Type   | Description |
|---------------|--------|-------------|
| reservationId | String |             |
| Time          | Number |             |

## CDAI Mechanism#2 – “Player Prebuffering” Feature

Can be leveraged for quick stream transitions. Suitable for preroll, and midroll insertions. No limitations with respect to content type – can transition between DASH and HLS.

### detach()

- Supported UVE version 0.9 and above.
- Optional API that can be used to quickly stop playback of active stream before transitioning to 2nd prebuffered stream.

### Example use of detach and buffering:

```
var player = new AAMPMediaPlayer();
player.load( "http://test.com/content.m3u8" ); // begin streaming main content
...
var adPlayer = new AAMPMediaPlayer(); // create background player
adPlayer.load( "http://test.com/ad.m3u8", false ); // preroll
...
player.detach(); // stop playback of active player
adPlayer.play(); // activate background player (fast transition)
player.stop(); // release remaining resources for initial player instance
```

### Example of midroll Ad insertions and resume main content playback:

|                               |                    |                     |                                 |
|-------------------------------|--------------------|---------------------|---------------------------------|
| Main content<br>(0 – 180 Sec) | AD1<br>(0 -40 Sec) | AD2<br>(0 – 30 Sec) | Main Content<br>(180 – 600 Sec) |
|-------------------------------|--------------------|---------------------|---------------------------------|

|                               |   |
|-------------------------------|---|
| Main Content<br>(0 – 180 sec) | <b>create foreground player and start streaming of main content</b><br>var player = new AAMPMediaPlayer();<br>player.load( "http://test.com/content.mpd" );<br><b>create background player and preload AD1</b><br>var adPlayer1 = new AAMPMediaPlayer();<br>adPlayer1.load( "http://test.com/ad1.mpd", false ); |
|-------------------------------|---|

|                                 |   |
|---------------------------------|---|
| AD1<br>(0 – 40 sec)             | <b>time of AD1 start, stop active player and activate background player for AD1</b><br>var position = Player.getCurrentPosition() // get current playback position<br>player.detach();<br>adPlayer1.play();<br>player.stop();<br><b>preload AD2 in background player</b><br>var adPlayer2 = new AAMPMediaPlayer();<br>adPlayer2.load( "http://test.com/ad2.mpd", false ); |
| AD2<br>(0 – 30 sec)             | <b>EOS of AD1, stop active player and activate background player for AD2</b><br>adPlayer1.detach();<br>adPlayer2.play();<br>adPlayer1.stop();<br><b>preload Main content in background and set last playback position</b><br>var player = new AAMPMediaPlayer();<br>player.Seek (position)<br>player.load( "http://test.com/content.mpd", false );                        |
| Main Content<br>(180 – 600 sec) | <b>EOS of AD2, stop active player and activate background player for main content</b><br>adPlayer2.detach();<br>player.play();<br>adPlayer2.stop();   |

## CONFIGURATION

### initConfig( config )

Configuration options are passed to AAMP using the UVE initConfig method. This allows the application override default configuration used by AAMP player to give more control over player behavior. Parameter is a JSON Object with one or more attribute/value pairs as follows:

| Property       | Type   | Default Value | Description               |
|----------------|--------|---------------|---------------------------|
| initialBitrate | Number | 2500000       | max initial bitrate (bps) |



|                        |            |          |  |
|------------------------|------------|----------|--|
| initialBitrate4K       | Number     | 13000000 | max initial bitrate for 4k video playback (bps)  |
| Offset                 | Number (s) | 0        | start position offset in seconds(same as seek() method)  |
| networkTimeout         | Number (s) | 10       | network request timeout for fragment/playlist/manifest downloads (in seconds)  |
| manifestTimeout        | Number (s) | 10       | Manifest download timeout; overrides networkTimeout if both present; available starting with version 0.8 . Applied to Main manifest in HLS and DASH manifest download. (in seconds)  |
| playlistTimeout        | Number (s) | 10       | HLS playlist download timeout; overrides networkTimeout if both present; available starting with version 1.0 (in seconds)  |
| downloadBuffer         | Number     | 3        | max amount of time to download ahead of playhead (fragments)<br>example:<br>- with a downloadBuffer of 3 (default) there will be 3 fragments (typically 2s each) of video or audio harvested and buffered in advance, in addition to internal playback buffering |
| minBitrate             | Number     | -        | Optional profile clamping (in bps)   |
| maxBitrate             | Number     | -        | Optional profile clamping (in bps)   |
| preferredAudioLanguage | String     | en       | ISO-639 audio language preference; for more than one language, provide comma delimited list from highest to lowest priority:<br>'<HIGHEST>,<...>,<LOWEST>'   |
| timeShiftBufferLength  | Number     | -        | (not supported, for future)  |
| stereoOnly             | Boolean    | False    | Optional forcing of playback to only select stereo audio track available starting with version 0.8   |
| liveOffset             | Number (s) | 15       | Allows override default/stream-defined distance from live point for live stream playback (in seconds)  |
| asyncTune              | Boolean    | False    | Return control to JS immediately when tuning with load() method (not supported)  |
| bulkTimedMetadata      | Boolean    | False    | Send timed metadata using single stringified JSON array instead of individual events available starting with version 0.8   |
| networkProxy           | String     | -        | Network proxy to use (Format <SCHEME>://<PROXY IP:PROXY PORT>)   |
| licenseProxy           | String     | -        | Network proxy to use for license requests (Format same as  |

|                                   |            |       |  |
|-----------------------------------|------------|-------|--|
|                                   |            |       | network proxy)   |
| downloadStallTimeout              | Number (s) | -     | Optional optimization - Allow fast-failure for class of curl-detectable mid-download stalls (in seconds)   |
| downloadStartTimeout              | Number (s) | -     | Optional optimization - Allow fast-failure for class of curl-detectable stall at start of download (in seconds)  |
| preferredSubtitleLanguage         | String     | en    | ISO-639 language code used with VTT OOB captions   |
| parallelPlaylistDownload          | Boolean    | True  | Optional optimization – download audio and video playlists in parallel for HLS; available starting with version 0.8  |
| parallelPlaylistRefresh           | Boolean    | True  | Optionally disable audio video playlist parallel download for linear (only for HLS)  |
| useAverageBandwidth               | Boolean    | False | Optional Average bandwidth for ABR switching ( version 1.0)  |
| preCachePlaylistTime              | Number (s) | -     | Optionally enable PreCaching of Playlist and TimeWindow for Cache(minutes) ( version 1.0)  |
| progressReportingInterval         | Number (s) | 1     | Optionally change Progress Report Interval (in seconds)  |
| useRetuneForUnpairedDiscontinuity | Boolean    | True  | Optional unpaired discontinuity retune config ( version 1.0)   |
| drmDecryptFailThreshold           | Number     | 10    | Maximum number of fragment decrypt failures before reporting playback error (version 1.0)  |
| initialBuffer                     | Number     | -     | Optional pre-tune buffering (in seconds) before playback start (version 2.4)   |
| useMatchingBaseUrl                | Boolean    | False | use DASH main manifest hostname to select from multiple base urls in DASH (when present). By default, will always choose first (version 2.4)   |
| initFragmentRetryCount            | Number     | 1     | Maximum number of retries for MP4 header fragment download failures (version 2.4)  |
| nativeCCRendering                 | Boolean    | False | Use native ClosedCaption support in AAMP (version 2.6)   |
| langCodePreference                | Number     | 0     | Set the preferred format for language codes in other events/APIs (version 2.6)<br>NO_LANGCODE_PREFERENCE = 0,<br>3_CHAR_BIBLIOGRAPHIC_LANGCODE = 1,<br>3_CHAR_TERMINOLOGY_LANGCODE = 2,<br>2_CHAR_LANGCODE = 3 |
| descriptiveTrackName              | Boolean    | False | Use descriptive audio track naming format which is a combination of <lang>-<role> (version 2.6)  |

|                              |         |      |  |
|------------------------------|---------|------|--|
| authToken                    | String  | -    | Optional field to set AuthService token for license acquisition(version 2.7) |
| useRetuneForGstInternalError | Boolean | True | Optional Gstreamer error retune config ( version 2.7)                        |

### setDRMConfig( config )

DRM configuration options are passed to AAMP using the setDRMConfig method. Parameter is JSON object with pairs of protectionScheme: licenseServerUrl pairs, along with preferredKeySystem specifying a preferred protectionScheme.

| Property                | Type   | Description   |
|-------------------------|--------|---|
| com.microsoft.playready | String | License server endpoint to use with PlayReady DRM.<br>Example:<br>http://test.playready.microsoft.com/service/rightsmanager.asmx    |
| com.widevine.alpha      | String | License server endpoint to use with Widevine DRM.<br>Example: https://widevine-proxy.appspot.com/proxy                              |
| preferredKeysystem      | String | Used to disambiguate which DRM type to use, when manifest advertises multiple supported DRM systems.<br>Example: com.widevine.alpha |

### Universal Video Engine Player Errors

| Error code                                   | Code | Error String                                    |
|--|------|---|
| AAMP_TUNE_INIT_FAILED                        | 10   | AAMP: init failed                               |
| AAMP_TUNE_INIT_FAILED_MANIFEST_DNLD_ERROR    | 10   | AAMP: init failed (unable to download manifest) |
| AAMP_TUNE_INIT_FAILED_MANIFEST_CONTENT_ERROR | 10   | AAMP: init failed (manifest missing tracks)     |
| AAMP_TUNE_INIT_FAILED_MANIFEST_PARSE_ERROR   | 10   | AAMP: init failed (corrupt/invalid manifest)    |

|   |    |   |
|---|----|---|
| AAMP_TUNE_INIT_FAILED_TRACK_SYNC_ERROR          | 10 | AAMP: init failed (unsynchronized tracks)                 |
| AAMP_TUNE_MANIFEST_REQ_FAILED                   | 10 | AAMP: Manifest Download failed<br>Playlist refresh failed |
| AAMP_TUNE_INIT_FAILED_PLAYLIST_VIDEO_DNLD_ERROR | 10 | AAMP: init failed (unable to download video playlist)     |
| AAMP_TUNE_INIT_FAILED_PLAYLIST_AUDIO_DNLD_ERROR | 10 | AAMP: init failed (unable to download audio playlist)     |
| AAMP_TUNE_FRAGMENT_DOWNLOAD_FAILURE             | 10 | AAMP: fragment download failures                          |
| AAMP_TUNE_INIT_FRAGMENT_DOWNLOAD_FAILURE        | 10 | AAMP: init fragment download failed                       |
| AAMP_TUNE_INVALID_MANIFEST_FAILURE              | 10 | AAMP: Invalid Manifest, parse failed                      |
| AAMP_TUNE_MP4_INIT_FRAGMENT_MISSING             | 10 | AAMP: init fragments missing in playlist                  |
| AAMP_TUNE_CONTENT_NOT_FOUND                     | 20 | AAMP: Resource was not found at the URL(HTTP 404)         |
| AAMP_TUNE_AUTHORISATION_FAILURE                 | 40 | AAMP: Authorization failure                               |
| AAMP_TUNE_UNTRACKED_DRM_ERROR                   | 50 | AAMP: DRM error untracked error                           |
| AAMP_TUNE_DRM_INIT_FAILED                       | 50 | AAMP: DRM Initialization Failed                           |
| AAMP_TUNE_DRM_DATA_BIND_FAILED                  | 50 | AAMP: InitData-DRM Binding Failed                         |
| AAMP_TUNE_DRM_SESSIONID_EMPTY                   | 50 | AAMP: DRM Session ID Empty                                |
| AAMP_TUNE_DRM_CHALLENGE_FAILED                  | 50 | AAMP: DRM License Challenge Generation Failed             |
| AAMP_TUNE_LICENCE_TIMEOUT                       | 50 | AAMP: DRM License Request Timed out                       |
| AAMP_TUNE_LICENCE_REQUEST_FAILED                | 50 | AAMP: DRM License Request Failed                          |

|                                      |    |   |
|--------------------------------------|----|---|
| AAMP_TUNE_INVALID_DRM_KEY            | 50 | AAMP: Invalid Key Error, from DRM   |
| AAMP_TUNE_UNSUPPORTED_STREAM_TYPE    | 50 | AAMP: Unsupported Stream Type<br>Unable to determine stream type for DRM Init |
| AAMP_TUNE_UNSUPPORTED_AUDIO_TYPE     | 50 | AAMP: No supported Audio Types in Manifest                                    |
| AAMP_TUNE_FAILED_TO_GET_KEYID        | 50 | AAMP: Failed to parse key id from PSSH  |
| AAMP_TUNE_FAILED_TO_GET_ACCESS_TOKEN | 50 | AAMP: Failed to get access token from Auth Service                            |
| AAMP_TUNE_CORRUPT_DRM_METADATA       | 50 | AAMP: DRM failure due to Bad DRMMetadata in stream                            |
| AAMP_TUNE_DRM_DECRYPT_FAILED         | 50 | AAMP: DRM Decryption Failed for Fragments                                     |
| AAMP_TUNE_DRM_KEY_UPDATE_FAILED      | 50 | AAMP: Failed to process DRM key   |
| AAMP_TUNE_CORRUPT_DRM_DATA           | 51 | AAMP: DRM failure due to Corrupt DRM files                                    |
| AAMP_TUNE_DEVICE_NOT_PROVISIONED     | 52 | AAMP: Device not provisioned  |
| AAMP_TUNE_HDCP_COMPLIANCE_ERROR      | 53 | AAMP: HDCP Compliance Check Failure   |
| AAMP_TUNE_GST_PIPELINE_ERROR         | 80 | AAMP: Error from gstreamer pipeline   |
| AAMP_TUNE_FAILED_PTS_ERROR           | 80 | AAMP: Playback failed due to PTS error  |

|                            |      |   |
|----------------------------|------|---|
| AAMP_TUNE_PLAYBACK_STALLED | 7600 | AAMP: Playback was stalled due to lack of new fragments |
| AAMP_TUNE_FAILURE_UNKNOWN  | 100  | AAMP: Unknown Failure                                   |

## Inband Closed Caption Management

To use inband closed captions, first register an event listener to discover decoder handle:

```
player.addEventListener("decoderAvailable", decoderHandleAvailable);
```

Along with corresponding event handler to publish the decoder handle to CC subsystem as follows:

```
function decoderHandleAvailable(event) {
    console.log("decoderHandleAvailable " + event.decoderHandle);
    XRReceiver.onEvent("onDecoderAvailable", { decoderHandle: event.decoderHandle });
}
```

**Toggle CC display on or off at runtime:**

```
XRReceiver.onEvent("onClosedCaptions", { enable: true });
XRReceiver.onEvent("onClosedCaptions", { enable: false });
```

**Set CC track at runtime:**

```
XRReceiver.onEvent("onClosedCaptions", { setTrack: trackID });
```

**Set CC style options at runtime:**

```
XRReceiver.onEvent("onClosedCaptions", { setOptions: defaultCCOptions});
```

defaultCCOptions is a JSON object of various style options and its values

**When closing stream, detach decoder handle:**

```
XRReceiver.onEvent("onDecoderAvailable", { decoderHandle: null });
```

Environments without the XREReceiver JS object may exist in future. Applications may use alternate CC rendering methods to avoid dependency on XREReceiver object.

To use, turn on nativeCCRendering init configuration value to true as follows:

```
player.initConfig( { nativeCCRendering: true } );
```

Toggle CC display on or off at runtime:

```
player.setClosedCaptionStatus(true);  
player.setClosedCaptionStatus(false);
```

Get/Set CC track at runtime:

```
player.getTextTrack();  
player.setTextTrack(trackIndex);
```

Get/Set CC style options at runtime:

```
player.getTextStyleOptions();  
player.setTextStyleOptions(options);  
options in a JSON formatted string of style options and its values.
```