Table of Contents

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.

Commented [Ruzzi, Jo1]: Consider using "the injectedbundle component" or "the injectedbundel API"

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	CDAI support, configuration options for tune optimization API: • setAlternateContent • notifyReservationCompletion • addCustomHTTPHeader Configuration: • stereoOnly • asyncTune • bulkTimedMetadata • useWesterosSink • parallelPlaylistDownload Events: • bufferingChanged • timedMetadata • adResolved • reservationStart • reservationEnd • placementStart • placementEnd

		• placementProgress • placementError
3	0.9	"Player Switching" Feature • load (autoplay=false support) • detach() method
4	1.0	Added support to get available audio track and closed captioning info API: • getAvailableAudioTracks • getAvailableTextTracks Configuration: • playlistTimeout • parallelPlaylistRefresh • useAverageBandwidth • preCachePlaylistTime • progressReportingInterval • useRetuneForUnpairedDiscontinuity • fragmentRetryLimit • drmDecryptFailThreshold

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 (i.e. one present from 0.7, or more recently introduced with UVE versions 0.8 or 0.9)

METHODS:

load(uri, autoplay)

- Supported UVE version 0.7 and above.
- Begin streaming.

Name	Type	Description
uri	String	URI of the Media to be played by the Video Engine
autoplay	Boolean	optional 2nd parameter (defaults to true) 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.

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. Note that ability to seek is currently limited to fragment granularity.

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
idle	0	eSTATE_IDLE
initializing	1	eSTATE_INITIALIZING
	2	eSTATE_INITIALIZED
	3	eSTATE_PREPARING
	4	eSTATE_PREPARED
	5	eSTATE_BUFFERING
paused	6	eSTATE_PAUSED
seeking	7	eSTATE_SEEKING

State Name	Value	Semantics
playing	8	eSTATE_PLAYING
	9	eSTATE_STOPPING
	10	eSTATE_STOPPED
	11	eSTATE_COMPLETE
	12	eSTATE_ERROR
	13	eSTATE_RELEASED

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. 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.

Name	Type	Description
volume	Number	Pass false to black out video. 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.

Value	Description
0	Paused
1	Normal Play Rate
2	2x Fast Forward (using iframe track)
4	4x Fast Forward (using iframe track)
8	8x Fast Forward (using iframe track)
16	16x Fast Forward (using iframe track)
-2	2x Rewind (using iframe track)
-4	4x Rewind (using iframe track)
-8	8x Rewind (using iframe track)
-16	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	Pass bitrate from getVideoBitrates to disable ABR and lock playback to single profile. Pass zero to (re)enable ABR, allowing Video Engine to select from available bitrates.

getCurrentAudioBitrate()

- Supported UVE version 0.7 and above.
- Return current audio bitrate.

setVideoRect(x, y, w, h)

- Supported UVE version 0.7 and above.
- Set display video rectangle coordinates relative to (0,0,1280,720)

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

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	
headerValue	String Array	
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	
headerValue	String Array	

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.

EVENTS

Event Name	Event Payload	Description
playbackStarted		- Supported UVE version 0.7 and above. - fired when playback starts
playbackStateChanged	state: number	- Supported UVE version 0.7 and above. - fired as state changes across play/pause seek/not-seek quadruplet
playbackProgressUpdate	durationMiliseconds: number, positionMiliseconds: number, playbackSpeed: number, startMiliseconds: number, endMiliseconds: number	- Supported UVE version 0.7 and above. - fired based on the interval set
bufferingChanged	buffering: bool	- Supported UVE version 0.8 and above. - fired when AAMP encounters buffering mid-playback, buffering flag indicates buffering status (on/off)

playbackCompleted		- Supported UVE version 0.7 and above. - fired when there is nothing left to play
playbackSpeedChanged	speed: number, reason: string	- Supported UVE version 0.7 and above.
playbackFailed	shouldRetry: boolean, code: number, description: string	- Supported UVE version 0.7 and above. - fired when an error occurs
decoderAvailable	decoderHandle: number	- Supported UVE version 0.7 and above. - fired when video decoder handle becomes available, required for closedcaption parsing + rendering by RDK ClosedCaptions module
mediaMetadata	durationMiliseconds: number, languages: string[], bitrates: number[], playbackSpeeds: number[], width: number, height: number, hasDrm: boolean	- Supported UVE version 0.7 and above. - fired with metadata of the asset currently played, includes duration(in ms), audio language list, available bitrate list, hasDrm, supported playback speeds
speedsChanged	playbackSpeeds:	- Supported UVE version 0.7 and above.

	number[]	- fired when supported playback speeds changes (based on iframe availability)
vttCueDataListener	start: number, duration: number, text: string	- Supported UVE version 0.7 and above. - fired for VTT cue parsed from the WebVTT playlist in the asset
drmMetadata	code: number, description: string	- Supported UVE version 0.7 and above. - fired when there is a change in DRM metadata (especially expiration of DRM auth data)
enteringLive		- Supported UVE version 0.7 and above. - fired when entering live point of a live playlist during/after a seek/trickplay operation
timedMetadata	time: number, duration: number, name: string, content: string, type: number, metadata: object, id: string	- Supported UVE version 0.8 and above. - fired when a subscribed tag is found in the playlist
bitrateChanged	time: number, bitRate: number, description: string, width: number, height: number, framerate: number	- Supported UVE version 0.7 and above. - fired when video profile is switched by ABR with the metadata associated with newly selected profile.

adResolved	resolvedStatus: bool, placementId: string, placementStartTime: number, placementDuration: number	- Supported UVE version 0.8 and above. - Confirmation that an upcoming ad's main manifest has been successfully downloaded and parsed.
reservationStart	adbreakId: string, time: number	- Supported UVE version 0.8 and above. - Sent upon playback into an ad break (one or more ads).
reservationEnd	adbreakId: string, time: number	- Supported UVE version 0.8 and above. - 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, time: number	- Supported UVE version 0.8 and above. - 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, time: number	- Supported UVE version 0.8 and above. - 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,	- Supported UVE version 0.8 and above.

	time: number	- 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, time: number, error: number	- Supported UVE version 0.8 and above. - 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: #EXT-X-IDENTITY-ADS #EXT-X-MESSAGE-REF #EXT-X-CUE #EXT-X-ASSET-ID #EXT-X-TRICKMODE-RESTRICTION #EXT-X-CONTENT-IDENTIFIER

setAlternateContent(reservationObject, promiseCallback)

- Supported UVE version 0.8 and above.

Name	Type	Description

reservationObject	Object	reservationObject provides context for alternate content to be played out at ad opportunities.
		
		`{`
		`"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": "",`
		`},`
		`}`

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 Switching" 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
```

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	Description
initialBitrate	Number	max initial bitrate (bps)
initialBitrate4K	Number	max initial bitrate for 4k video playback (bps)
offset	Number (s)	start position offset (same as seek() method)
networkTimeout	Number (s)	network request timeout for fragment/playlist/manifest downloads
manifestTimeout	Number (s)	Manifest download timeout; overrides networkTimeout if both present; available starting with version 0.8 . Applied to Main manifest in HLS and DASH manifest download.
playlistTimeout	Number (s)	HLS playlist download timeout; overrides networkTimeout if both present; available starting with version 1.0
downloadBuffer	Number	max amount of time to download ahead of playhead (fragments) example: - with a downloadBuffer of 3 (default) there will be 3 fragments (typically 2s each) of video or audio harvested and buffered in advance, in additional to internal playback buffering
minBitrate	Number	Optional profile clamping
maxBitrate	Number	Optional profile clamping
preferredAudioLanguage	String	ISO-639 audio language preference; for more than one language, provide comma delimited list from highest to lowest priority: '<HIGHEST>,<...>,<LOWEST>'
timeShiftBufferLength	Number	(not yet supported)
stereoOnly	Boolean	Optional forcing of playback to only select stereo audio track (defaults to false); available starting with version 0.8
liveOffset	Number (s)	Allows override default/stream-defined distance from live point for live stream playback
asyncTune	Boolean	Return control to JS immediately when tuning with load() method (defaults to false); (not supported)
bulkTimedMetadata	Boolean	Send timed metadata using single stringified JSON array instead of individual events (defaults to false); available starting with version 0.8
useWesterosSink	Boolean	Use westerossink for playback instead of Broadcom brcmVideoSink (defaults to false); 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)	Allow fast-failure for class of curl-detectable mid-download stalls
downloadStartTimeout	Number (s)	Allow fast-failure for class of curl-detectable stall at start of download
preferredSubtitleLanguage	String	ISO-639 language code used with VTT OOB captions
parallelPlaylistDownload	Boolean	Optional optimization – download audio and video playlists in parallel for HLS; available starting with version 0.8
parallelPlaylistRefresh	Boolean	Optionally disable audio video playlist parallel download for linear (only for HLS)
useAverageBandwidth	Boolean	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)	Optionally change Progress Report Interval (in seconds)
useRetuneForUnpairedDiscontinuity	Boolean	Optional unpaired discontinuity retune config (version 1.0)
fragmentRetryLimit	Number	Maximum number of fragment download failures before reporting playback error (version 1.0)
drmDecryptFailThreshold	Number	Maximum number of fragment decrypt failures before reporting playback error (version 1.0)

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

Universal Video Engine Player Errors

Error code	Code	Error String
AAMP_TUNE_INIT_FAILED	10	AAMP: init failed Fragmentcollector initialization 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 Playlist refresh failed
AAMP_TUNE_AUTHORISATION_FAILURE	40	AAMP: Authorization failure
AAMP_TUNE_FRAGMENT_DOWNLOAD_FAILURE	10	AAMP: fragment download failures
AAMP_TUNE_INIT_FRAGMENT_DOWNLOAD_FAILURE	10	AAMP: init fragment download failed
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 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_DATA	51	AAMP: DRM failure due to Corrupt DRM files
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_GST_PIPELINE_ERROR	80	AAMP: Error from gstreamer pipeline
AAMP_TUNE_PLAYBACK_STALLED	7600	AAMP: Playback was stalled due to lack of new fragments

AAMP_TUNE_CONTENT_NOT_FOUND	20	AAMP: Resource was not found at the URL(HTTP 404)
AAMP_TUNE_DRM_KEY_UPDATE_FAILED	50	AAMP: Failed to process DRM key
AAMP_TUNE_DEVICE_NOT_PROVISIONED	52	AAMP: Device not provisioned
AAMP_TUNE_HDCP_COMPLIANCE_ERROR	53	AAMP: HDCP Compliance Check Failure
AAMP_TUNE_INVALID_MANIFEST_FAILURE	10	AAMP: Invalid Manifest, parse failed
AAMP_TUNE_FAILED_PTS_ERROR	80	AAMP: Playback failed due to PTS error
AAMP_TUNE_MP4_INIT_FRAGMENT_MISSING	10	AAMP: init fragments missing in playlist
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);
   XREReceiver.onEvent("onDecoderAvailable", { decoderHandle: event.decoderHandle });
}
```

Toggle CC display on or off at runtime:

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

Set CC track at runtime:

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

When closing stream, detach decoder handle:

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