

FireboltMediaPlayer (Phase1)

- [Versions](#)
- [Notes](#)
- [Methods](#)
 - [create\(v1\)](#)
 - [release \(v1\)](#)
 - [load \(v1\)](#)
 - [play \(v1\)](#)
 - [pause \(v1\)](#)
 - [stop \(v1\)](#)
 - [seekTo \(v1\)](#)
- [Events](#)
 - [playbackStarted\(v1\)](#)
 - [playbackStateChanged\(v1\)](#)
 - [playbackProgressUpdate\(v1\)](#)
 - [bufferingChanged\(v1\)](#)
 - [playbackSpeedChanged\(v1\)](#)
 - [playbackFailed\(v1\)](#)
- [Constants:](#)
 - [PlaybackState\(v1\)](#)
 - [PlaybackError\(v1\)](#)

Versions

org.rdk.FireboltMediaPlayer.1 - This API provides the ability to playback IP media streams

Notes

- Wrapper for AAMP HLS/DASH player for RDK community as an alternative to native/UVE-JS integrations
- AAMP dedicated process with env AAMP_ENABLE_OPT_OVERRIDE=1 (enables /opt/aamp.cfg) and AAMP_ENABLE_WESTEROS_SINK=1
- Are setWaylandDisplay and setVideoRectangle equivalents needed for hole-punching and scaling of out of process player instance?
- Todo: add 'initConfig' and 'drmConfig' methods

Methods

create(v1)

Description: will instantiate a new (AAMP) player instance, suitable for playback of IP feeds. If a player identified by the specified id already exists it will be ref-counted.

Arguments:

- id : string - requested player id.

Returns:

- success : bool

```
Request : { "jsonrpc": "2.0", "id": 3, "method": "org.rdk.FireboltMediaPlayer.1.create", "params": {  
  "id" : "MainPlayer" } }  
  
Response: { "jsonrpc": "2.0", "id": 3, "result": {  
  "success": true  
} }
```

release (v1)

Description: will decrease the ref-count of the player. When the ref-count reaches 0 the player will be destroyed.

Arguments:

- id: string - id of the associated player instance

Returns:

- success : bool

```
Request : { "jsonrpc": "2.0", "id": 3, "method": "org.rdk.FireboltMediaPlayer.1.release", "params": {
    "id": "MainPlayer" } }

Response: { "jsonrpc": "2.0", "id": 3, "result": {
    "success": true
} }
```

load (v1)

Description: associated locator with a player instance.

Arguments:

- id: string - associated player instance
- url : string - locator of the stream to play
- autoplay: bool (Default: true) - true instructs the player to start playing immediately.

Returns:

- success : bool

```
Request : { "jsonrpc": "2.0", "id": 3, "method": "org.rdk.FireboltMediaPlayer.1.load", "params": {
    "id": "MainPlayer",
    "url": "https://cpetestutility.stb.r53.xcal.tv/VideoTestStream/main.m3u8",
    "autoplay": true
} }

Response: { "jsonrpc": "2.0", "id": 3, "success": true }
```

play (v1)

Description: begin/resume streaming content associated with this player instance. Sets the playspeed to 1

Arguments:

- id: string - associated player instance

Returns:

- success : bool

```
Request : { "jsonrpc": "2.0", "id": 3, "method": "org.rdk.FireboltMediaPlayer.1.play", "params": {
    "id": "MainPlayer"
} }

Response: { "jsonrpc": "2.0", "id": 3, "success": true }
```

pause (v1)

Description: pause streaming content associated with this player instance. Sets the playspeed to 0

Arguments:

- id: string - associated player instance

Returns:

- success : bool

```
Request : { "jsonrpc": "2.0", "id": 3, "method": "org.rdk.FireboltMediaPlayer.1.pause", "params": {
  "id": "MainPlayer"
} }

Response: { "jsonrpc": "2.0", "id": 3, "success": true }
```

stop (v1)

Description: stop streaming content - new 'load' request required to recycle player instance. Note that for ideal performance, we recommend NOT calling stop in between new load() - avoids unnecessary resource deallocation/reallocation.

Arguments:

- id: string - associated player instance

Returns:

- success : bool

```
Request : { "jsonrpc": "2.0", "id": 3, "method": "org.rdk.FireboltMediaPlayer.1.stop", "params": {
  "id": "MainPlayer"
} }

Response: { "jsonrpc": "2.0", "id": 3, "success": true }
```

seekTo (v1)

Description: Moves the media to a specific position.

Arguments:

- id: string - associated player instance
- sec: integer - position in seconds
 - for VOD streams:
 - relative to the beginning of the media
 - for LIVE streams:
 - relative to the position of the first tune or eldest part of the live window if culling has since occurred
 - -1 - position of the current live point

Returns:

- success: bool

```
Request : { "jsonrpc": "2.0", "id": 3, "method": "org.rdk.FireboltMediaPlayer.1.seekTo", "params": {
  "id": "MainPlayer",
  "sec": 30
} }

Response: { "jsonrpc": "2.0", "id": 3, "success": true }
```

Events

playbackStarted(v1)

Description: A event notifying playback started

Payload:

- "<player name>" - object - "<player name>" is actual name of the player. Object Payload is empty

```
{
  "jsonrpc": "2.0",
  "method": "org.rdk.FireboltMediaPlayer.1.playbackStarted",
  "params": {
    "MainPlayer": {}
  }
}
```

playbackStateChanged(v1)

Description: A event notifying playback state changed

Payload:

- "<player name>" - object - "<player name>" is actual name of the player.
Object Payload:
 - state: integer - The current state of the player instance, one of [PlaybackState\(v1\)](#)

```
{
  "jsonrpc": "2.0",
  "method": "org.rdk.FireboltMediaPlayer.1.playbackStateChanged",
  "params": {
    "MainPlayer": { "state" : 8 }
  }
}
```

playbackProgressUpdate(v1)

Description: A event notifying about playback progress

Payload:

- "<player name>" - object - "<player name>" is actual name of the player.
Object Payload:
 - durationMiliseconds: integer - current duration of the stream
 - positionMiliseconds: integer - current play or pause position relative to:
 - for VOD - the beginning of media
 - for LIVE - position of the first tune or eldest part of the live window if culling has since occurred
 - playbackSpeed: integer - current playback speed, 1 for playback in normal speed
 - startMiliseconds: integer - 0 for VOD or first tune or eldest part of the live window if culling has since occurred for LIVE
 - endMiliseconds: integer - duration of stream for VOD or time of end of live window

```
{
  "jsonrpc": "2.0",
  "method": "org.rdk.FireboltMediaPlayer.1.playbackProgressUpdate",
  "params": {
    "MainPlayer": { "durationMiliseconds": 734000, "positionMiliseconds": 4458, "playbackSpeed": 1, "
startMiliseconds": 0, "endMiliseconds": 734000 }
  }
}
```

bufferingChanged(v1)

Description: A event notifying if the buffering state changed

Payload:

- "<player name>" - object - "<player name>" is actual name of the player.
Object Payload:
 - buffering: bool - 'true' if buffering started, 'false' if ended

```
{
  "jsonrpc": "2.0",
  "method": "org.rdk.FireboltMediaPlayer.1.bufferingChanged",
  "params": {
    "MainPlayer": { "buffering": true }
  }
}
```

playbackSpeedChanged(v1)

Description: A event notifying if playback speed has changed

Payload:

- "<player name>" - object - "<player name>" is actual name of the player.
Object Payload:
 - speed: integer - current playback speed, 1 for playback in normal speed

```
{
  "jsonrpc": "2.0",
  "method": "org.rdk.FireboltMediaPlayer.1.playbackSpeedChanged",
  "params": {
    "MainPlayer": { "speed": 0 }
  }
}
```

playbackFailed(v1)

Description: A event notifying about playback error

Payload:

- "<player name>" - object - "<player name>" is actual name of the player.
Object Payload:
 - shouldRetry: bool - if 'true', the user should try to stop and load the stream again
 - code - one of [PlaybackError\(v1\)](#) Codes
 - description - human-readable description of the error, one of [PlaybackError\(v1\)](#) Description String

```
{
  "jsonrpc": "2.0",
  "method": "org.rdk.FireboltMediaPlayer.1.playbackFailed",
  "params": {
    "MainPlayer": { "shouldRetry": true, "code": 10, "description": "AAMP: Manifest Download failed : Curl Error Code 7" }
  }
}
```

Constants:

PlaybackState(v1)

Description: Specifies the state of the playback - will be changed based on the content being played.

Value	Description
0	Player is idle
1	Player is initializing a particular content
2	Player has initialized for a content successfully

3	Player is loading all associated resources
4	Player has loaded all associated resources successfully
5	Player is in a buffering state
6	Playback is paused
7	Seek is in progress
8	Playback is in progress
9	Player is stopping the playback
10	Player has stopped playback successfully
11	Playback completed
12	Error encountered and playback stopped
13	Player has released all resources for playback

PlaybackError(v1)

Code	Description String
10	AAMP: init failed Fragmentcollector initialization failed
10	AAMP: init failed (unable to download manifest)
10	AAMP: init failed (manifest missing tracks)
10	AAMP: init failed (corrupt/invalid manifest)
10	AAMP: init failed (unsynchronized tracks)
10	AAMP: Manifest Download failed Playlist refresh failed
40	AAMP: Authorization failure
10	AAMP: fragment download failures
10	AAMP: init fragment download failed
50	AAMP: DRM error untracked error
50	AAMP: DRM Initialization Failed
50	AAMP: InitData-DRM Binding Failed
50	AAMP: DRM Session ID Empty
50	AAMP: DRM License Challenge Generation Failed
50	AAMP: DRM License Request Timed out
50	AAMP: DRM License Request Failed
50	AAMP: Invalid Key Error, from DRM
50	AAMP: Unsupported Stream Type Unable to determine stream type for DRM Init
50	AAMP: No supported Audio Types in Manifest
50	AAMP: Failed to parse key id from PSSH
50	AAMP: Failed to get access token from Auth Service
	AAMP: DRM failure due to Corrupt DRM files

50	AAMP: DRM failure due to Bad DRMMetadata in stream
50	AAMP: DRM Decryption Failed for Fragments
80	AAMP: Error from gstreamer pipeline
7600	AAMP: Playback was stalled due to lack of new fragments
20	AAMP: Resource was not found at the URL(HTTP 404)
50	AAMP: Failed to process DRM key
52	AAMP: Device not provisioned
53	AAMP: HDCP Compliance Check Failure
10	AAMP: Invalid Manifest, parse failed
80	AAMP: Playback failed due to PTS error
10	AAMP: init fragments missing in playlist
100	AAMP: Unknown Failure