

EasyBroadcast For Dashjs Player

Description: This manual is a complete guide to ensure a integration of EasyBroadcast Library into your Dashjs Player 2.4 - 2.5.

Require: Dashjs Player 2.4 - 2.5, Chrome, Code Editor

Edited by	Timestamp	Note
Zhejiayu Ma	13th September 2017	First edition

1. Create a html file and import library scripts

First, create a `index.html` that includes scripts of EasyBroadcast Library `eb.js` and then dashjs, finally `dashjs-easybroadcast2_4.js` which is an adapter for dashjs. Do keep the order of them so that they can work as we want them to.

```
<!Doctype html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Your title here</title>
    <style>
      button.enabled {
        background: green;
      }

      button.disabled {
        background: red;
      }

      button.selected {
        background: blue;
      }
    </style>
  </head>
  <body>
    <div>
      <video id="custom-player" class="" preload="none" controls>
      </video>
    </div>
  </body>
  <script src="../../path/to/eb.js"></script>
  <script src="https://cdn.bootcss.com/dashjs/2.4.0/dash.all.min.js">
</script>
  <script src="../../path/to/dashjs-easybroadcast2_4.js"></script>
</html>
```

2. Configure the video

To start up, create a script tab below those three script, and add all these into it. And, here you go.

```

/**
 * Obligatoire
 */
// 1. Create a dashjs player instance
var player = dashjs.MediaPlayer().create()

// 2. Easybroadcast bind to player and initialize easybroadcast
easybroadcast.bind(player)({
  broadcasterId: 'test',
  manager: 'wss://manager4.easybroadcast.fr',
})

// 3. Initialize Player
player.initialize()
player.attachView(document.getElementById('custom-player'))
player.attachSource('http://olive.fr.globecast.tv/live/ramdisk/demo_hd/dash-mp4/demo_hd.mpd');

/**
 * Optional
 */
player.getDebug().setLogToBrowserConsole(false)
player.on(dashjs.MediaPlayer.events['PLAYBACK_STARTED'], function () {
  player.setAutoSwitchQualityFor('video', false)
  player.setAutoSwitchQualityFor('audio', false)
  player.setInitialBitrateFor('audio', 321)
  player.setInitialBitrateFor('video ', 321)
  player.setQualityFor('video', 1)
})

```

3. Create a local server

Due to security issue, the library can't be used on file system. It should be served by a server. So to be simple, we use npm to install globally a lite server.

If you haven't install npm, check [this](#) out, you can install it in couples of minutes.

Then run in terminal `npm install -g lite-server` and go to the directory containing **the** html file.

Run `lite-server`, the result below should be seen from the terminal. Open the url

`http://localhost:3000`

```
root$ lite-server
Did not detect a bs-config.json or bs-config.js override file. Using lite-
server defaults...
** browser-sync config **
{ injectChanges: false,
  files: [ './*/*.html,css,js' ],
  watchOptions: { ignored: 'node_modules' },
  server: { baseDir: './', middleware: [ [Function], [Function] ] } }
[BS] Access URLs:
-----
      Local: http://localhost:3000
    External: http://xxxxxxx:3000
-----
      UI: http://localhost:3001
  UI External: http://xxxxxxx:3001
-----
[BS] Serving files from: ./
[BS] Watching files...
```