

# DASH | Provider API Guide

The JW Player Provider API is a partner-restricted API to load external, independent media playback engines into JW Player. These engines can be written in HTML5 and/or Flash. Existing providers include:

- [Akamai HD Network](#) (flash): Akamai
- [MPEG DASH](#) (html5): DASH.js
- [360 Video](#) (flash): Surround Definition
- [P2P Streaming](#) (html5 & flash): Streamroot.io

This document provides an overview of the provider API, to help partners in developing their own providers for JW Player.

Do note the API documentation and implementation do not have the same level of robustness as those of the JW Player JavaScript API. Therefore, do not hesitate to ping us with any questions.

## HTML5 Providers

The HTML5 provider API was made available in JW Player 6.10. It allows partners to plug in their HTML5 media playback engines into the “guts” of JW Player.

Here is an example setup, using the demo provider bundled with this SDK. The actual provider is the *MyMediaProvider.js* file, an external component loaded at runtime:

```
<script src="http://jwpsrv.com/library/KEY.js"></script>
<script src="MyMediaProvider.js"></script>

<script type="text/javascript">
  jwplayer("player").setup({
    file: 'dummy.timer',
    duration: 10
  });
</script>
```

In this setup, the *MyMediaProvider* registers with the player it can play files with the extension *.timer* and therefore will be selected to play *dummy.timer*. The *duration* option shows as an example to provide providers with configuration info.

## Building HTML5 Providers

To build an HTML5 provider, you need only the JW Player Provider SDK, which includes:

- This guide
- The *MyMediaProvider* example provider
- A simple test page

The example provider showcases the following options of media providers:

- Registering itself with JW Player and signalling supported (file) types.
- Loading player configuration options (*duration*).
- Rendering DOM elements in the player.
- Providing playback state and time position updates.
- Responding to player commands like `play()`, `pause()`, `stop()` and `seek()`.

Since the internal JW Player providers use the same APIs than 3rd party providers, you can see the [JW Player source code](#) for more implementations. These are:

- Video (MP4/WebM and MP3/M4A/Ogg)
- YouTube

## Flash Providers

The Provider API was originally designed for JW Player version 5. This architecture still drives the Flash side of 3rd party providers.

Here is an example setup, using the demo provider bundled with this SDK. The actual provider is the *MyMediaProvider.swf* file, an external component loaded at runtime:

```
<script type="text/javascript">
  jwplayer("player").setup({
    playlist: [{
      file: 'http://example.com/video.mp4',
      provider: '../MyMediaProvider.swf',
      type: 'flv',
    }],
    MyMediaSaturation: -100
  });
</script>
```

The example showcases some of the compatibility quirks, notably the *type=flv* to force the player into using the Flash rendering mode.

## Building Flash Providers

To build a Flash provider, you need:

- [The Adobe Flex SDK](#)
- This JW Player Provider SDK, containing:
  - This guide
  - A simple build script
  - The JW Player library SWC
  - An example provider
  - A simple test page

The example provider enables basic playback of MP4 videos. It showcases the following options of media providers:

- Loading player configuration options (*MyMediaSaturation*).
- Exposing of quality levels to JavaScript API and controlbar buttons.
- Sending out custom metadata through `onMeta()`.
- Sending of custom error events to the player.
- Buffer and time position management.
- Volume and mute management.

Since the internal JW Player providers use the same APIs than 3rd party providers, you can see the [JW Player source code](#) for more implementations. These are:

- MP3 (Sound)
- RTMP
- MP4/FLV (Video)
- YouTube

The example Flash provider included in this SDK is a derivative of the MP4 one.