
Accessibility

In this short chapter, we're going to look at accessibility. Accessibility (in the context of video streaming) refers to the practice of making your videos accessible to people of all abilities and those with disabilities by adding subtitles and closed captions to them.

Adding subtitles to your videos means more people will be able to watch them—unfortunately it won't help you make videos that people will want to watch.

In some cases, you have to; it's a legal requirement. The U.S. government recently signed into law the Twenty-First Century Communications and Video Accessibility Act (CVAA). As of September 2013, any content aired on television with subtitles must also have subtitles when it's distributed on the web.

We'll start by learning what subtitles are and how to add to them to your videos. We'll then take a short video and add English and Spanish subtitles to it. By the end of the chapter you'll know enough (hopefully) to start making your own videos more accessible to a wider audience.

8.1 Understanding Subtitles

Subtitles are a text representation of the dialog in a video. They allow people who are hard of hearing or deaf to understand what is being said. However, this is not their only use. They can also be used to provide the translation of dialog in a foreign language. For example, if you are watching a film in a language you don't speak, without subtitles you won't be able to understand what is going on.

We can add subtitles to our videos by using Web Video Text Tracks¹ (WebVTT). WebVTT is a plain-text format for displaying subtitles, originally designed for use with HTML5 video.

Here's an example of a WebVTT file:

1. <http://dev.w3.org/html5/webvtt/>

```

WEBVTT
X-TIMESTAMP-MAP=MPEGTS:900000,LOCAL:00:00:00.000
00:00:01.000 --> 00:00:03.500
Have you had the opportunity to be in Columbia
00:00:04.000 --> 00:00:06.000
Belgium, Denmark, France
00:00:06.000 --> 00:00:10.200
United States, Spain, Holland, Poland, Germany, Sweden
00:00:10.300 --> 00:00:11.300
in the same week.

```

Each subtitle has a start time, an end time, and the text to be displayed. Timing cues are in the format `HH:MM:SS.mmm` and are relative to the start of the video. For example, the first subtitle (“Have you had the opportunity to be in Columbia”) is displayed for a duration of 2.5 seconds, starting at 1 seconds from the beginning of the video and ending at 3.5 seconds.

Subtitles for the Deaf and Hard of Hearing

Subtitles for the deaf and hard of hearing (SDH) can also contain cues that describe the audio and music in a video. For instance, the crowd cheering at a sporting event; music playing in the background. They can also include information about the person who is speaking.

We can use cue settings² to position the text exactly where we want it to go. Here's an example:

```

00:00:15.000 --> 00:00:45.000 line:0 align:middle
[ Crowd cheers ]

```

The position of the text is specified immediately after the timing information. In this instance, the text is positioned at the top of the screen (*line:0*) and centered horizontally (*align:middle*). You'll often see them displayed at the top of the screen—this is to distinguish them from the subtitles.

Styling Subtitles

You can also format the subtitle text by adding some styling information to it. For instance, parts of the text can be displayed in bold (``), italics (`<i>`) or underlined (`<u>`). Let's look at an example. The following line was taken from `english.vtt` and modified to include some basic styling information:

```

00:00:18.500 --> 00:00:20.500
so that you live a new experience.

```

2. https://developer.mozilla.org/en-US/docs/HTML/WebVTT#Cue_Settings

When the text is rendered on screen, the word “you” will be displayed in italics and “new” will be underlined.

8.2 Adding Subtitles to Videos

In this section, we're going to learn how to add subtitles to a video. Just like with video and audio, subtitles are divided into separate files and referenced from a playlist. We can use `mediasubtitlesegmenter` to do this.

The video we are going to use is a short promotional video for a festival in Colombia. You can download it here³. Once you've downloaded the video, rename it to `promo.mp4`.

First, create the directory `subtitles` inside `/Library/WebServer/Documents`. This is where we'll put all the files. Now we need to prepare the video for streaming with HLS. Open up a terminal and run the following command from the directory where you saved the video:

```
$ mediafilesegmenter -I -t 9 -f /Library/WebServer/Documents/subtitles promo.mp4
```

The next step is to create a variant playlist. (We'll see why shortly.) Run this command from the same directory as the previous one:

```
$ variantplaylistcreator -o /Library/WebServer/Documents/subtitles/master.m3u8 \
prog_index.m3u8 promo.plist
```

This will write the variant playlist (`master.m3u8`) to the `subtitles` directory alongside the video segments and playlist.

Now, we can turn our attention to the subtitles. In the code download for this book, there are two sets of subtitles. One set is in English (`english.vtt`) and the other is in Spanish (`spanish.vtt`). Both files are in the WebVTT format.

We'll start with the English language subtitles. In the same directory as the subtitle files, open up a terminal and run the following command:

```
$ mediasubtitlesegmenter -f /Library/WebServer/Documents/subtitles \
-B english -i english.m3u8 english.vtt
```

We use the `-i` option to set the name of the playlist to `english.m3u8` and we set the base name (`-B`) of the subtitle files to `english`. This means the files will be called `english0.webvtt`, `english1.webvtt`, and so on. We do this to distinguish them from subtitles in other languages.

3. <http://vimeo.com/29790482>

If you look in the `subtitles` directory in the web root, you will see some files with the extension `.webvtt` and the subtitle playlist (`english.m3u8`). Let's take a look at the playlist. Open it in a text editor. It should look like this:

```
#EXTM3U
#EXT-X-TARGETDURATION:28
#EXT-X-VERSION:3
#EXT-X-MEDIA-SEQUENCE:0
#EXT-X-PLAYLIST-TYPE:VOD
#EXTINF:28.00000,
english0.webvtt
#EXT-X-ENDLIST
```

The playlist contains the target duration and the locations of the subtitle files. Now, you may be wondering how can subtitles have a target duration? In our playlist the target duration is 28 seconds, which means that the subtitle file contains the subtitles that appear during that 28 second interval. In this instance, we only have one subtitle file. The start time of a subtitle (and the target duration) determines which file it is added to. For example, if the target duration was 30 seconds and we have a subtitle that begins at 40 seconds, it would go in the second file. (The default target duration is 60 seconds.)

Now it's your turn. Repeat the process for the Spanish subtitles (`spanish.vtt`). Call the playlist `spanish.m3u8` and use `spanish` as the base name for the subtitle files.

How does the player know what subtitles are available? We use a variant playlist. We add a reference to each set of subtitles and then associate them with the video stream(s). Let's do this for the English and Spanish subtitles. Open up the variant playlist (`master.m3u8`) and modify it to look like this:

```
#EXTM3U
#EXT-X-MEDIA:TYPE=SUBTITLES,GROUP-ID="subs",NAME="Español",
DEFAULT=NO,FORCED=NO,URI="spanish.m3u8",LANGUAGE="es"
#EXT-X-MEDIA:TYPE=SUBTITLES,GROUP-ID="subs",NAME="English",
DEFAULT=NO,FORCED=NO,URI="english.m3u8",LANGUAGE="en"
#EXT-X-STREAM-INF:BANDWIDTH=1118592,CODECS="mp4a.40.2, avc1.64001f",
RESOLUTION=640x360,SUBTITLES="subs"
prog_index.m3u8
```

We define each set of subtitles by using the `EXT-X-MEDIA` tag and setting the type to `SUBTITLES`. Each entry contains the location of the playlist and a name. The `LANGUAGE` attribute is a 2-letter code⁴ that indicates the language the subtitles are in. Subtitles are grouped together with the `GROUP-ID` attribute.

4. http://en.wikipedia.org/wiki/List_of_ISO_639-1_codes

To associate the subtitles with the video, we set the `SUBTITLES` attribute of the video stream to the group id of the subtitles (“subs”). When you’re done, save the file.

Now we’re ready to test what we’ve done. Open up Safari and enter the URL of the variant playlist: <http://localhost/subtitles/master.m3u8>.



In the bottom right-hand corner you should see an icon that looks like a speech bubble. If you click on it, a list of available subtitles will appear. There should be two: English and Spanish. Notice that the names of the subtitles in the list match the names in the variant playlist. By default, subtitles aren’t displayed. However, you can alter this behaviour. Let’s say you want to turn on the English subtitles by default. Open the variant playlist and change the `DEFAULT` attribute to `YES` for the English subtitles. Save the file and reload the playlist in the browser. Now the English subtitles will appear automatically when the video starts to play.

Trying playing the video on iOS, either in the simulator or on a device if you have one. You can use the application in [Section 3.3, Playing Video in iOS](#). If you run the application on a device, don’t forget to change `localhost` to the IP address of your machine (System Preferences > Network).

8.3 Summary

In this chapter we’ve learned that adding subtitles in different languages makes our videos more accessible to a wider audience, which can result in more views. And in some cases, adding subtitles is mandatory—it’s a legal requirement. We then looked at how to add English and Spanish subtitles to an HLS video stream.