



Accessible HTML5 Media Players, Captions, Audio Description, and Search

Terrill Thompson

tft@uw.edu

@terrillthompson

<http://terrillthompson.com>

The Title: Deconstructed

- Accessible
- HTML5
- Media Players
- Captions
- Audio Description
- Search

Accessible

Accessible to whom?

- To people who can't hear audio content
 - Captions
- To people who can't see visual content
 - Audio description
- To people who can't see or hear the content
 - Transcript
- To people who aren't using a mouse
 - An keyboard-accessible player

HTML



About HTML5

- Work on the spec began in 2004 by the Web Hypertext Application Technology Working Group (WHATWG), separate from W3C, but now working collaboratively with W3C
- <http://www.w3.org/TR/html5/>
- <http://whatwg.org/html> (more current)
- HTML Accessibility Task Force
<http://www.w3.org/WAI/PF/HTML/>

Everything's easy with HTML5!

Then:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

Now:

```
<!DOCTYPE html>
```

Seriously. *Everything's* easy!

Then:

```
<meta http-equiv="Content-Type" content="text/html;  
      charset=utf-8" />
```

Now:

```
<meta charset="utf-8">
```


Media Players

Media Players Then

```
<object id="mediaPlayer1" width="180" height="50"
classid="CLSID:22d6f312-b0f6-11d0-94ab-0080c74c7e95"
codebase=http://activex.microsoft.com/activex/controls/mplayer/en/nsmp2inf.cab#Version=5,1,
52,701 type="application/x-oleobject"
standby="Loading Microsoft Windows Media Player components..." >
<param name="fileName" value="take5.wmv">
<param name="animationatStart" value="true">
<param name="transparentatStart" value="true">
<param name="autoStart" value="false">
<param name="showControls" value="true">
<param name="ShowAudioControls" value="true">
<param name="ShowStatusBar" value="true">
<param name="loop" value="false">
<embed type="application/x-mplayer2"
pluginspage="http://microsoft.com/windows/mediaplayer/en/download/"
id="mediaPlayer" name="mediaPlayer" displaysize="4" autosize="-1"
bgcolor="darkblue" showcontrols="true" showtracker="-1"
showdisplay="0" showstatusbar="-1" videoborder3d="-1" width="420" height="380"
src="take5.wmv" autostart="false" loop="false">
</embed>
</object>
```

Media Players Now

```
<audio src="take5.mp3" controls>  
</audio>
```

What if your browser doesn't support MP3?

```
<audio controls>  
  <source src="take5.mp3">  
  <source src="take5.ogg">  
</audio>
```

No mouse?

```
<audio controls tabindex="0">  
  <source src="take5.mp3">  
  <source src="take5.ogg">  
</audio>
```

Your browser doesn't support HTML5 <audio>?

```
<audio controls tabindex="0">
```

```
<source src="take5.mp3">
```

```
<source src="take5.ogg">
```

```
<!-- fallback content goes here -->
```

```
</audio>
```

It works in IE9!



- Tab gives the player focus
- Space toggles play/pause
- Right/left arrows fast forward/rewind
- Up/down arrows control volume
- The player is not exposed to screen readers (they read the fallback text)

It works in Firefox!



- Tab gives the player focus
- Space and arrow keys work as they do in IE
- Screen readers announce:
 - Play button / pause button
 - Trackbar
 - Mute button / unmute button

It works in Opera!



- The only player in which individual components are included in the tab order
- However, once track bar has focus, there doesn't appear to be a way to operate it

It works in Chrome and Safari (sort of)



- Tab gives the player focus, but then what?
- Player doesn't respond to keyboard commands

It works in Safari on iPhone (sort of)



- VoiceOver doesn't recognize it

Better accessibility via custom controls

```
<audio controls tabindex="0">  
  <source src="take5.mp3">  
  <source src="take5.ogg">  
  <!-- fallback content goes here -->  
</audio>  
<div id="controls"></div>
```

Add HTML controls using JavaScript

```
var audio=document.getElementById("audio");  
if (audio.canPlayType) {  
    //browser can play this HTML5 audio  
    //add HTML controls (e.g., links, buttons, inputs)  
    //and listeners to control their behavior  
}
```

Accessible Audio Player (AAP) 1.0



- Uses the Yahoo! Media Player as a fallback
- Has more-or-less the same interface for both players
- <http://terrillthompson.com/music/aap>

The Same Code Works for Video

```
<video controls tabindex="0" id="video">  
  <source src="take5.mp4">  
  <source src="take5.ogv">  
  <!-- fallback content goes here -->  
</video>  
<div id="controls"></div>
```

...with the same results per browser



Can't hear the audio?

```
<video controls tabindex="0" id="video">  
  <source src="take5.mp4">  
  <source src="take5.ogv">  
  <track kind="captions" src="take5.srt">  
  <!-- fallback content goes here -->  
</video>  
<div id="controls"></div>
```

WebVTT Format

- Web Video Text Tracks
- Previously "WebSRT"
- Based on SubRip format (*.srt)
- The draft WHATWG spec:
<http://www.whatwg.org/specs/web-apps/current-work/webvtt.html>
- Understanding WebSRT:
<http://www.delphiki.com/websrt/>

WebVTT Example

- 1

00:00:13,678 --> 00:00:16,789

Ladies and gentlemen

2

00:00:23,910 --> 00:00:27,101

Please give it up for
the Dave Brubeck Quartet!

Browsers don't support <track> yet,
but we can do it with JavaScript

```
<div id="player">
```

```
<video controls tabindex="0" id="video">
```

```
<source src="take5.mp4">
```

```
<source src="take5.ogv">
```

```
<track kind="captions" src="take5.srt">
```

```
<!-- fallback content goes here -->
```

```
</video>
```

```
<div id="captions"></div>
```

```
</div>
```

```
<div id="controls"></div>
```

JavaScript & Captions

```
video.addEventListener('timeupdate',function () {  
    //the current time on the media has been updated  
    //if a new caption should be displayed at this time  
    //display the new caption in the #captions div  
});
```

Example of captions in HTML5



Can't see the visual content?

```
<div id="player">  
  <video controls tabindex="0" id="video">  
    <source src="take5.mp4">  
    <source src="take5.ogv">  
    <track kind="captions" src="take5-captions.srt">  
    <track kind="descriptions" src="take5-desc.srt">  
    <!-- fallback content goes here -->  
  </video>  
  <div id="captions"></div>  
</div>  
<div id="controls"></div>
```

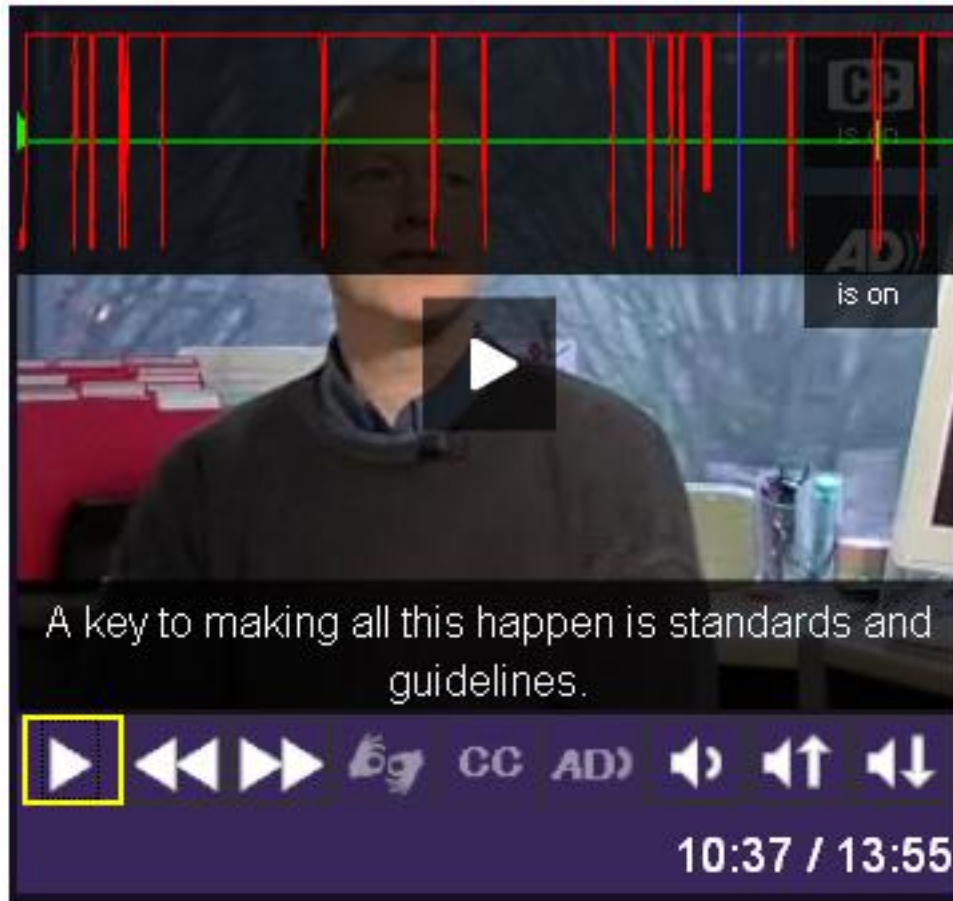
The DO-IT Solution until browsers support closed audio description

```
if (userPrefersDescription) {  
    var videoSrc = "take5_described.mp4";  
}  
else {  
    var videosrc = "take5_not_described.mp4";  
}  
source.setAttribute('src',videoSrc);
```


What about a fallback player?

- JW Player (a Flash player) plays FLV files
- <http://longtailvideo.com/players>
- Has buttons that are clearly labeled for screen reader users
- Has a robust API that allows the player to be operated using a custom HTML controller
- Supports closed captions
- And...

JW Player supports closed audio description!



Functionality Galore



Video Collection

[Preferences](#) | [About DO-IT Video](#) | [FAQ](#)

Search the full text of DO-IT's video library:

Your Search Results

We found **15** captions in **4** videos matching your search terms. Select any video title to go to the home page for that video. Click on any timestamp to view the video starting at that time.



[World Wide Access: Accessible Web Design \(2009\)](#)

Relevance Rating: 7

How to make web pages accessible to people with disabilities.

Matching captions from this video:

- **03:59** ...in developing a website is to comply with **standards**....
- **04:01** ...So, be aware of what the **standards** are,...
- **04:05** ...and develop websites that comply with those **standards**....
- **07:34** ...It's been designed using a **standards**-based approach,...

More Functionality Galore



[Preferences](#)

Access to Technology in the Workplace: In Our Own Words

[Return to main page for this video](#)

Transcript

Click on any spoken text to play the video at that point. No mouse? Tab to the desired text and press Enter.

(Music)

In a blurry image, skyscrapers tower overhead.

(Narrator) Every day, it's a bit of a miracle. Millions of people go to work, interacting in a complex web of interdependency. We communicate across cubicles and across country with equal speed. In a world of technology-based global connections, equal access defines success.

(Mia) Having assistive technology available makes it possible to have people on your workforce who may not have been able to work before, not because they weren't qualified, but because the necessary tools weren't available.

Resources

- DO-IT Video

<http://uw.edu/doit/video>

- My blog

<http://terrillthompson.blogspot.com>