



Media Player Accessibility: Insights from Interviews and Focus Groups

Terrill Thompson
Technology Accessibility Specialist
University of Washington
tft@uw.edu

Media Accessibility in HTML5

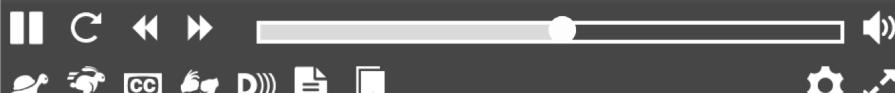
- The `<audio>` and `<video>` elements
- The `<track>` element
 - Captions
 - Subtitles
 - Descriptions
 - Chapters
 - Metadata
- Web Video Text Tracks (WebVTT)
- A robust API that enables development of custom, accessible media players

Able Player 3.1

- Free, open source
- Accessible buttons and controls
- Support for all five kinds of text tracks
- Description text track is exposed in an ARIA live region (announced by screen readers)
- Also supports alternative audio described version of video (user can toggle between versions with a "Description" button)
- Support for interactive transcript, built from chapters, descriptions, & captions
- Support for synchronized sign language
- Highly customizable UI via Preferences
- <https://ableplayer.github.io/ableplayer>



Your website's visitors aren't a
faceless mass of identical mouse-clickers



0:27 / 0:53

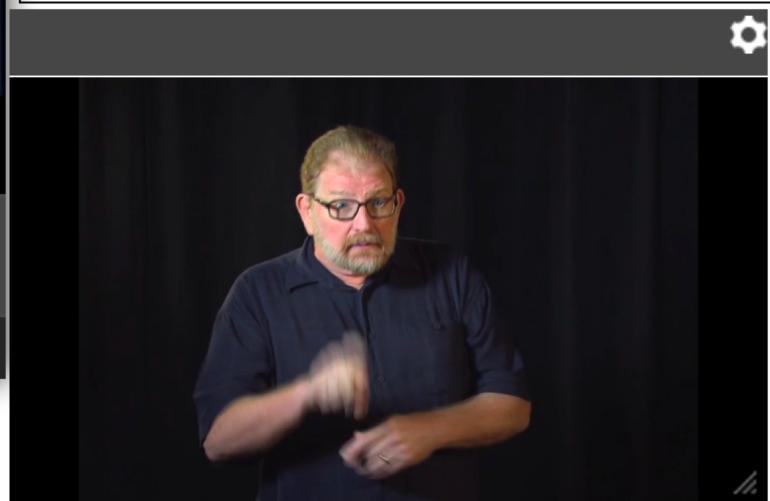
Speed: 1x

Playing

Auto scroll: Language: English

(Narrator) You want these people. They **order** your products, **sign up** for your services, **enroll** in your classes, **read** your opinions, and **watch** your videos. You'll never see them, but they know you- through your website. Or maybe not. **Your website's visitors aren't a faceless mass of identical mouse-clickers** but a vibrant community of individuals with varying tastes, styles, and abilities. This includes people with disabilities.

Terrill Thompson



CSUN 2017

- Interviews and focus groups with 37 persons with disabilities on media player accessibility
- 13 one-hour sessions
 - 10 focus groups (up to 6 participants each)
 - 3 individual interviews

Participants by Disability

Blind	22
Low vision	8
Deaf or hard of hearing	4
Mobility (speech input user)	1
Other (not using AT)	2

"Bring Your Own Device"

- Windows laptops
- iPads
- iPhones
- Android phones
- Various screen readers
- ZoomText
- Dragon Naturally Speaking

Part 1: The Interview

- Describe your preferred technologies for accessing video online.
- Where do you tend to access video online?
- What are your likes and dislikes about the online media players you've encountered?

Part 2: Playing with Demos

- Instruction: For each demo, "explore the video player for 2-3 minutes".
- Demos (all featured the same short video)
 1. Video with text-based audio description
 2. Video with human-narrated described version
 3. Video with synchronized sign language
- Discussion and exploration

**Six prominent issues
emerged.**

Issue 1: Seeking to a new point in the media

- Able Player uses role="slider" and associated ARIA markup
- Able Player also has "Rewind" and "Forward" buttons where default interval is a % of total duration (not to exceed a maximum number of steps)

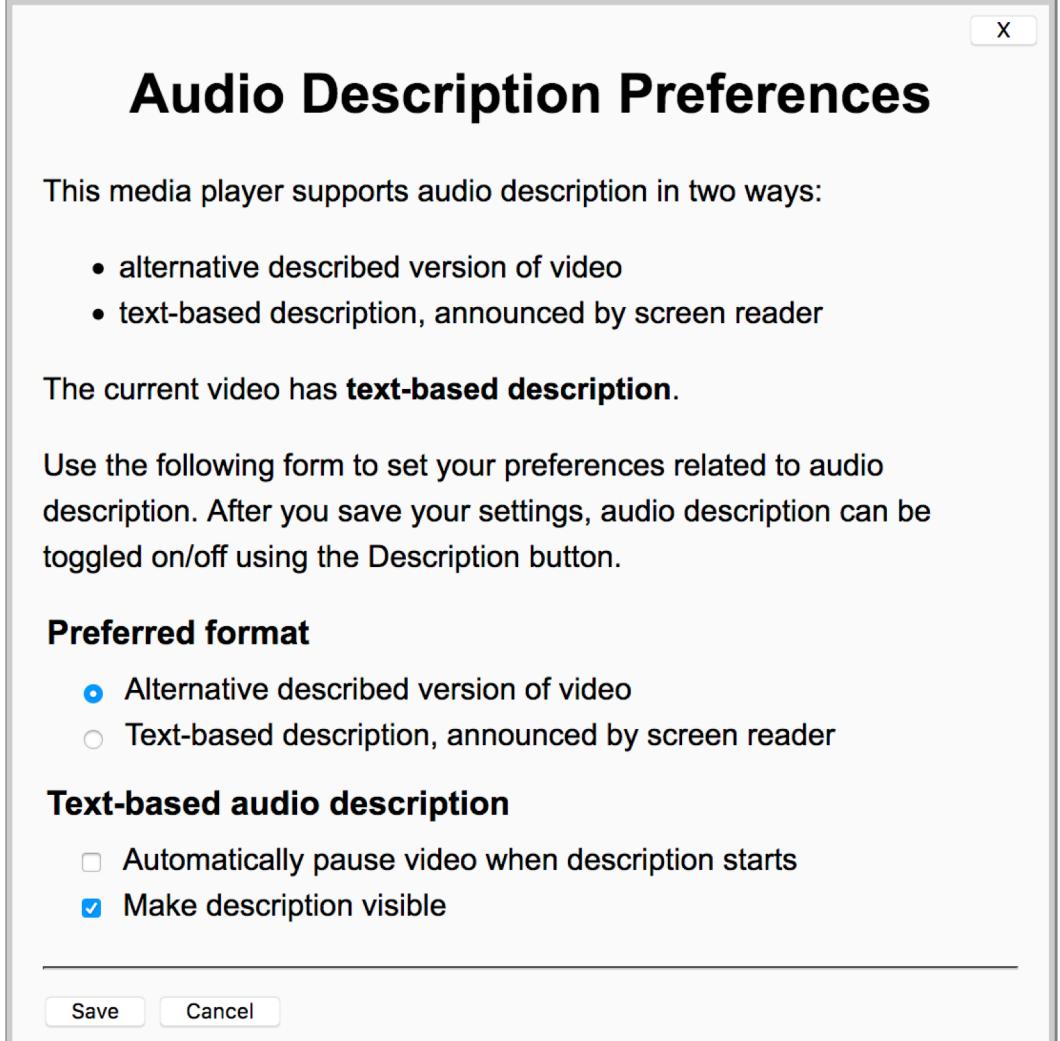


Issue 1 (continued)

- Controls should disclose the seek interval (e.g., "Forward 10 seconds")
- Neither slider nor buttons have enough granularity. Need a "Jump to time" edit box.
- Dragon user would like to use natural language commands like "Forward eight minutes"

Issue 2: Audio Description Preferences

Human-narrated description is nearly always preferred over text-based description, if given a choice



The dialog box is titled "Audio Description Preferences" and includes a close button (X) in the top right corner. The content area begins with a statement: "This media player supports audio description in two ways:" followed by a bulleted list: • alternative described version of video • text-based description, announced by screen reader. Below this, a message states: "The current video has **text-based description**." Further instructions advise: "Use the following form to set your preferences related to audio description. After you save your settings, audio description can be toggled on/off using the Description button." A "Preferred format" section contains two radio buttons: one selected (blue outline) for "Alternative described version of video" and one unselected (gray outline) for "Text-based description, announced by screen reader". A "Text-based audio description" section contains two checkboxes: one unselected (gray outline) for "Automatically pause video when description starts" and one selected (blue outline) for "Make description visible". At the bottom of the dialog are "Save" and "Cancel" buttons.

Audio Description Preferences

This media player supports audio description in two ways:

- alternative described version of video
- text-based description, announced by screen reader

The current video has **text-based description**.

Use the following form to set your preferences related to audio description. After you save your settings, audio description can be toggled on/off using the Description button.

Preferred format

Alternative described version of video

Text-based description, announced by screen reader

Text-based audio description

Automatically pause video when description starts

Make description visible

Save Cancel

Issue 2 (continued):

Problems with Text-based Description

- Users might ignore screen reader if it starts talking during video playback.
- The role of screen readers is to respond to user's commands, and while there can be latitude for alerts and other critical information, automatic reading of content generally falls outside this scope.
- Pressing a key interrupts the description, thereby breaking accessibility
- Extended audio description poses challenges (can pause video when description starts, but end time is unknown)

Issue 2 (continued): Pros of Text-based Description

- It's easy to produce, with minimal cost
- "Any description is better than no description"
- It's searchable and browsable
(e.g., in interactive transcript)

Issue 3. Accessibility of Subtitles to Screen Reader Users

- Able Player currently uses `aria-hidden="true"`, based on the belief that a screen reader reading content in synch with spoken word in the program audio would be distracting and undesirable (the content is accessible in the transcript)

Issue 3 (continued): One User's Perspective

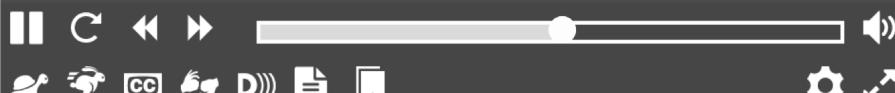
- Foreign language subtitles are critical for access in real-time.
- Same-language captions can help clarify difficult-to-understand audio.
- Most users can handle simultaneous audio tracks. However, independent volume controls would maximize usability.

Issue 4. Synchronized Sign Language

- Sign language window should be semi-transparent so it can be placed on top of the video without completely obstructing content.
- Able Player users can already drag and resize the sign language window, but they should also be able to crop it and/or resize it *without* preserving the aspect ratio.
- The video window itself should also be draggable and resizable, not just the sign language window.



Your website's visitors aren't a
faceless mass of identical mouse-clickers



0:27 / 0:53

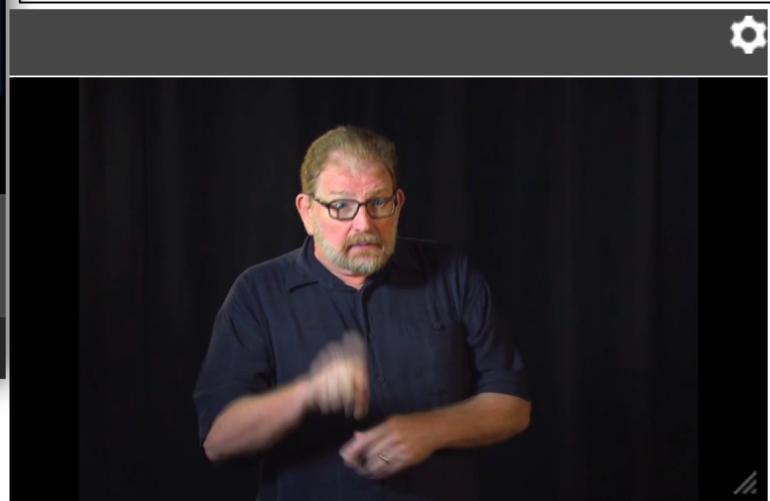
Speed: 1x

Playing

Auto scroll: Language: English  

(Narrator) You *want* these people. They **order** your products, **sign up** for your services, **enroll** in your classes, **read** your opinions, and **watch** your videos. You'll never see them, but they know you- through your website. Or maybe not. **Your website's visitors aren't a faceless mass of identical mouse-clickers** but a vibrant community of individuals with varying tastes, styles, and abilities. This includes people with disabilities.

Terrill Thompson



Issue 5. Quantity/Visibility/ Order of Controls

- A common complaint: "Too many buttons"
- A common solution: Hide less frequently used buttons behind a "More" button.
- Which buttons should be hidden? There is no consensus.
- Recommended possible solutions:
 - Customize per individual's usage patterns
 - Allow users to select which buttons are visible within Preferences
 - Both solutions would result in a non-standard UI across users

Issue 5 (continued)

- Multiple sighted users expressed a desire to hide the controls during playback.
- One user described this as an accessibility problem, as the controls are distracting and prevent them from fully focusing on the content of the video
- This is especially desirable in full screen mode
- Controls should be easily recoverable if user moves the mouse or presses a key.

**The full paper is expected to
be published June 1 in
CSUN's Journal on
Technology & Persons
with Disabilities**

<https://csun.edu/cod/journal>