

WCAG 2 Compliance With Flash

A few Notes For This Webinar

1. Please mute your phone line
2. You can post questions in Connect during the session and we will answer what we can at the end of the webinar
 - Further questions will be answered over email or through our blog
 - This session is being captioned and a link to the recording of the webinar will be provided

In This Presentation

1. Background to Flash & Accessibility
2. Methods for Creating Accessible Flash movies
3. Flash & WCAG 2.0 Compliance
 - Principle 1: Perceivable
 - Principle 2: Operable
 - Principle 3: Understandable
 - Principle 4: Robust

Why Flash

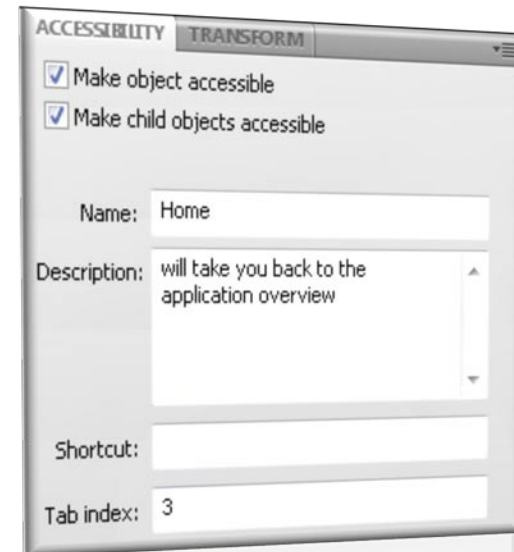
- Provides rich features and full multimedia capabilities
- Only requires a light weight plug-in (as opposed to Applets)
- Early uses:
 - Mostly as eye candy to spice up bland HTML
 - Banners, games, animated movies
- Now , Flash has evolved with the web:
 - Full Flash based RIA's
 - Ecommerce solutions
 - Standardized components
 - Flash's answer to Web 2.0: Adobe Flex

Accessibility in Flash

- Early days (up to Flash Player 6):
 - Virtually inaccessible
 - No information provided at all to assistive technology
 - Keyboard trap: no way out
 - No keyboard accessibility, unless manually scripted
 - Often a nuisance for screen reader users
- Now (version 6 and above):
 - Flash content can be fully accessible
 - All content exposed to AT
 - Keyboard accessibility
 - Accessible interactive components

Features: Accessibility Panel

- Provided with Flash Professional
- Allows you to:
 - Provide text alternatives for any object, image or control
 - Exposed to ATthroughMSAA
 - Modify the tab order
 - Have custom controls be labeled automatically
- All this can be achieved through scripting as well



Features: Accessible Components

- Flash ships with with a set of accessible components:
 - Labeling
 - Role & state information
 - Keyboard accessibility
- Components not accessible by default
 - Accessibility needs to be enabled by developer
 - Each component type requires two added lines of code
 - Button example:
 - `import fl.accessibility.ButtonAcclImpl;`
`ButtonAcclImpl.enableAccessibility();`

Features: Captioning & Audio Descriptions

- The FLVPlaybackCaptioning component
 - Allows closed captioning to be added to video material
 - Recorded sound can be used to provide audio descriptions

Flash and WCAG 2

About WCAG

- Web Content Accessibility Guidelines (WCAG)
 - Explain the requirements for accessible content
 - Version 2.0 released December 2008
- Organized around the following 4 principles:
 - Perceivable
 - Operable
 - Understandable
 - Robust
- Each principle has guidelines
- Each guideline has success criteria

Principle 1: Perceivable

“Information and user interface components must be presentable to users in ways they can perceive.”

Guideline 1.1: Alternative text

- *“Provide text alternatives for any non-text content so that it can be changed into other forms people need, such as large print, braille, speech, symbols or simpler language.”*
- Embedded Flash movies
 - Example: Use the accessibility panel while the main stage is selected
 - Choose ‘make movie accessible’ and ‘Make child elements accessible’
- Images / objects / animations:
 - Example: Use the accessibility panel to specify an accessible ‘name’ and a ‘description’
- Decorative Images
 - Example: Hide content using the accessibility panel
- Form Controls
 - Each control needs to be labeled

Guideline 1.2: Time-based Media

- *“Provide alternatives for time-based media.”*
 - Closed captions provide alternative for audio info
 1. Create captions file
 - Timed Text XML file (DFXP)
 - Not created in Flash (Captivate, MAGpie, Hi-Caption Studio)
 - Requires skill & time: Outsource?
 2. Assign Flash captioning component to movie player
 - Components used: FLVPlayback & FLVPlaybackCaptioning
 3. Optionally: Modify captions display
 - Inside / outside player
 - Text Formatting



Guideline 1.2 (ctd)

- Audio descriptions provide alternative for visual information
 1. Identify sections in movie that need descriptions
 2. Record descriptions as MP3
 3. Create SMIL document to assign MP3 files to timecodes
 - Alternatively, use cuepoints for extended audio descriptions
- Full text transcripts provide complete overview of all information
 - Not synchronized with the movie
 - Either provide inside or outside Flash movie

Guideline 1.3: Adaptable

- *“Create content that can be presented in different ways (for example simpler layout) without losing information or structure.”*
- Label form controls
 - Using standard form components
 - Using accessibility panel to label controls
 - Using auto labeling
- Datagrid component
 - Automatically inserts column header in front of every cell
- Provide a logical tab order
 - Tab & reader order are the same in Flash
 - Modify the ‘tabindex’ property of the objects on your stage

Guideline 1.4: Distinguishable

- *“Make it easier for users to see and hear content including separating foreground from background.”*
- Provide different contrast schemes
- Allow text to be enlarged
 - Flash’s vector based content ideal for zooming
- Do not intrude with audio
 - Do not automatically start audio over 3 seconds
 - Provide easy access to audio control

Principle 2: Operable

“User interface components and navigation must be operable.”

Guideline 2.1: Keyboard Accessible

- *“Make all functionality available from a keyboard.”*
 - All interactive controls must be reachable and operable without a mouse.
- Where possible, use standard components
 - Keyboard accessibility built in
 - For custom controls, manually handle keyboard input
- Logical tab order

Guideline 2.4: Navigable

- *“Provide ways to help users navigate, find content, and determine where they are.”*
- Provide ways of skipping content
 - Add buttons / menus
 - Ensure the purpose of each link is clear
 - Use accessibility panel if needed
- Allow users to take their time
 - Prevent timeouts

Principle 3: Understandable

“Information and the operation of user interface must be understandable.”

Guideline 3.1: Readable

- *“Make text content readable and understandable.”*
- Set the language
 - Flash inherits language settings from containing HTML
- Provide glossaries and Indices
- Prevent disruptive changes of context
- Form validation & handling
 - Easier to use Flex

Principle 4: Robust

“Content must be robust enough that it can be interpreted reliably by a wide variety of user agents, including assistive technologies.”

Guideline 4.1: Compatible

- *“Maximize compatibility with current and future user agents, including assistive technologies.”*
- Use standardized components
 - Role will be automatically provided
 - Custom interactive controls will be ‘buttons’

Using Flex

- Create Flash based forms and web apps
- Same accessibility as Flash + more
 - Forms
 - Mark form fields as required
 - Provide validation info
 - Temporary save results on server



The screenshot shows a web form with three input fields: "Social Security Number", "Date of Birth", and "Place of Birth". The "Social Security Number" field contains "xx" and is highlighted with a red border. A red error message box on the right states: "The Social Security number must be 9 digits or in the form NNN-NN-NNNN." The "Date of Birth" field is a three-part input with the first part containing a vertical bar. The "Place of Birth" field is empty. A "Save and move to Step 2: Contact Details" button is at the bottom.

Social Security Number * xx

Date of Birth * |

Place of Birth *

The Social Security number must be 9 digits or in the form NNN-NN-NNNN.

Save and move to Step 2: Contact Details

Wrapping Up

- To Summarize:
 - Flash content can be made fully accessible and WCAG 2.0 compliant
 - Use standard form controls, and enable their accessibility
 - Provide captions, audio descriptions and text transcripts
- Adobe and Accessibility
 - For more information, visit the Adobe Accessibility Portal:
 - <http://www.adobe.com/accessibility/products/flash>