

WCAG 2 Compliance With PDF

In This Presentation

1. Background of PDF Files & Accessibility
2. Methods for Creating Accessible PDF Docs
3. PDF & WCAG 2.0 Compliance
 - Principle 1: Perceivable
 - Principle 2: Operable
 - Principle 3: Understandable
 - Principle 4: Robust

Why PDF

- Portability
- Preservation of visual formatting and layout
- Ideal format for long documents intended for printing
- Easy to produce compared to HTML
- Documents can be secured against editing

Examples of PDF use

- Technical documents and manuals
- Reports, especially with complex layouts, graphs, charts etc.
- Forms, meant for print or online delivery
- Scanners and other hardware that output PDF
- Print and design industry
- Large and/or complex documents published on web sites

PDF and accessibility

- Can documents retain these advantages of layout, portability, security and still be accessible?
 - PDF documents support a logical, accessible structure that is independent of the layout
 - Security used to cause an accessibility issue, not any more. Documents can be secured against editing but still be accessible
 - Accessible PDFs can be produced quickly but, as with HTML, the source document must contain structure and semantics

When is PDF not accessible?

- A scanned image is still an image, with no information for screen readers unless the information is added
- Same for PDF, HTML or any document format
- PDF can support a wide array of accessibility information for all content elements, but these must be added, in the source document or in Acrobat

Background of PDF Accessibility

- Until 2001, PDF was not accessible:
 - Content not readable by AT
 - Each page was basically a flat image:
 - Scanned paper document
 - Converted to image from source document
- Since then: PDF can (and should) be fully accessible
 - Accessible tag structure
 - Real text rather than image of text
 - Control over reading order
 - Text alternatives
 - Accessible form controls

PDF Accessibility: Tags

- A tagged document contains an underlying 'tag tree':
 - Represents the structure of your document
 - Perceived by screen reader as document content
 - Not tagged = not announced by AT
 - Similar to tags in HTML
 - Can be edited using the 'Tags Panel'
- **Always** required for accessibility
 - Enables typical screen reader features, such as:
 - Heading navigation
 - Table navigation
 - Link lists



Creating Tagged PDF Documents

- Different 'paths' leading to an accessible PDF doc:

Authoring Application

- Tag structure automatically built

Scanned Documents

- Requires OCR and tagging in Acrobat Pro

Existing PDF doc

- Must be tagged in Acrobat Pro

- Preferred method:
 - Conversion from accessible source document

PDF 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."*
- Images:
 - Apply alternative text in authoring application.
 - Example: Microsoft Word
 - Use Adobe Acrobat Pro
 - Example: use the Touch Up reading Order Tool (TURO).
- Decorative Images
 - Do not need alternative
 - Example: remove image from tag structure using TURO.
- Form Controls
 - Use LiveCycle Designer for large, complex forms
 - Use Acrobat form tools for small, basic forms
 - Example: Making existing form controls interactive.

Guideline 1.2: Time-based Media

- *"Provide alternatives for time-based media."*
- Embedded multimedia must be provided with alternatives
 - Captions
 - Audio descriptions
 - Full text transcripts
- The synchronized alternatives are applied to the actual media, not in PDF

Guideline 1.3: Adaptable

- *"Create content that can be presented in different ways (for example simpler layout) without losing information or structure."*
- Use headings to provide document structure
 - Example: Using MS Word heading styles
 - Example : Tagging headings with Acrobat Pro
- Provide header cells for data tables
 - Example: Using the Table Editor
- Provide a logical reading order
 - Example: Using the Touch Up Reading Order Tool

Guideline 1.4: Distinguishable

- *"Make it easier for users to see and hear content including separating foreground from background. "*
- Adobe Reader provides helpful features:
 - Customizable high contrast colors
 - Unlimited magnification

To fund
incentiv

To fund some of the innovative
incentive programs such as the
program and the State Public
paid 50% of the installed cost

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.
- Ensure the tab order is logical
 - Example: Setting the tab order with the 'fields' panel

Guideline 2.4: Navigable

- *"Provide ways to help users navigate, find content, and determine where they are."*
- Automatically import bookmarks and TOCs
 - Example: Using PDFMaker

Principle 3: Understandable

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

Guideline 3.1: Readable

- *"Make text content readable and understandable."*
- Always set the document's language
 - For the entire document (using document properties)
 - For specific sections (using the tags panel)
- Provide glossaries and Indices
- Mark up abbreviations
 - Example: Using tag properties to add alternative text to an abbreviated text.

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."*
- Manually Provide Roles
 - Example: Using the tags panel

Ensuring Accessibility at the Source

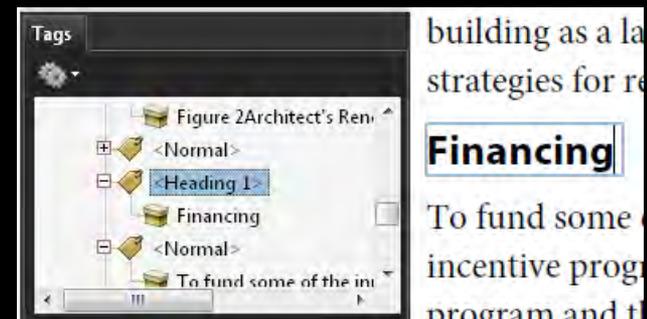
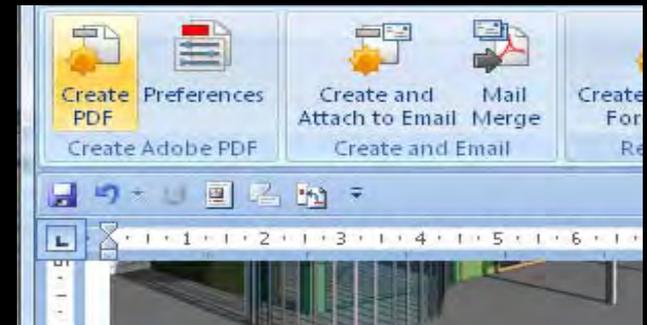
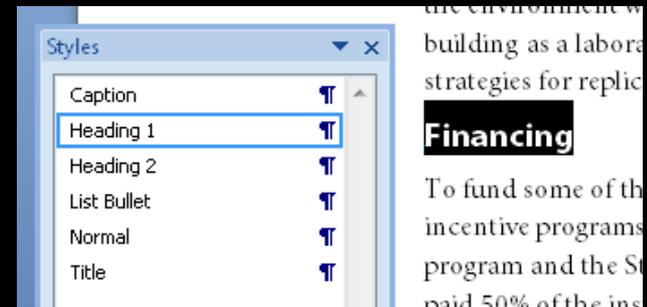
Conversion to accessible PDF from authoring application

Conversion to PDF From Accessible Source Document

- Advantages:
 - Tag structure automatically built based on styles / markup used
 - Significantly more efficient / effective than tagging PDF after conversion
 - Most accurate tag structure
 - Ability to edit content without having to reapply tags in Acrobat
- PDFMaker
 - Add-on for popular business applications, such as MS Office and Lotus Notes

Example Workflow: Creating an Accessible PDF Using MS Word

1. Use appropriate styles, e.g. Heading, list and paragraph.
2. Use PDFMaker add-on to create a tagged PDF File.
3. Use Acrobat Pro to inspect and repair document if needed.



Legacy Documents

Repairing existing inaccessible PDF
files

Repairing Scanned Documents

- Use Acrobat's OCR feature
 - Scanned text will be recognized and converted to actual text.
 - OCR Makes a text in a scanned document:
 - Searchable
 - Selectable
 - Perceivable by assistive technology
 - OCR is not perfect
 - You must manually fix 'OCR suspects'
 - Document still requires a tag structure

Adding and Editing the Tag Structure

- Acrobat Pro provides several tools for tagging:
 - ‘Add Tags to Document’ command
 - Accessibility Check
 - Creates overview of accessibility issues, including tag related
 - Provides quick link to each issue
 - Touch Up Reading Order (TURO) Tool
 - Allows content to be selected and assigned a tag type
 - Tags Panel
 - Allows tag structure to be edited manually

Wrapping Up

- To Summarize:
 - PDF documents can always be made accessible and WCAG 2.0 compliant
 - Create accessible source documents, then convert to tagged PDF.
 - Use Acrobat Pro tools to repair accessibility problems

Further Reading

- Go to:
 - <http://www.adobe.com/accessibility/>
 - Detailed guides for creating accessible PDF documents and forms
- Questions?
 - cpike@paciellogroup.com
 - access@adobe.com

Seminar Series

- Next seminars

Also at <http://adobe.acrobat.com/wcag2/>

- **Webinar 2: “Flash and WCAG 2.0”**
Tuesday, March 31 12:00 noon Eastern
- **Webinar 3: “PDF Forms and WCAG 2.0”**
Wednesday, April 8 12:00 noon Eastern