

# Practical Approaches to Web Accessibility

Tracy Mitrano

Director of IT Policy and of Institute for Computer Policy & Law  
Cornell University

Irina Zhankov

Designer, Custom Web Development Group  
Cornell Information Technologies

Stacy Pendell

Support Lead, Custom Web Development Group  
Cornell Information Technologies

# Why Accessibility Matters

- Accessibility is about the human experience
  - Not limited to people with disabilities
  - Example: curb cuts for wheelchairs benefit bikes, strollers, shopping carts...
- Accessibility = Availability (Universal Design)
- Accessibility > Compliance
- A device-agnostic approach to accessible web design makes sites available to as many people as possible on as many devices as possible
  - Responsive Web Design benefit

# Accessibility Needs Vary

Think about people who:

- **Can't see or read screen**
  - Visual impairments
  - Cognitive or neurological impairments
  - Reading disabilities
  - Language limitations (English as a second language)
- **Can't use mouse**
  - Mobility or visual impairments
- **Can't hear audio**
  - Hearing impairments
  - Noisy environment or shared cubicle
- **Have other issues**
  - Photosensitive epilepsy
  - Aging population

# Statistics

- According to the World Health Organization (WHO), **15%** of the world population lives with some form of disability, and for 10% of humankind – more than **600 million** people – those disabilities are life-altering.
- United States alone counts more than **54 million** persons with disabilities.
  - Those numbers will dramatically increase over the next decades with the general aging of the population: the U.S. Department of Labor estimates that 20% of the American population will be over 65 years old in 2030, experiencing some limitations in mobility, visual and hearing capabilities, and even some cognitive issues.

# Web Accessibility Empowers People

Shawn Lawton Henry, W3C/WAI:

“With accessible websites, people with disabilities can do ordinary things: children can learn, teenagers can flirt, adults can make a living, seniors can manage their stock portfolios, and so on. With the Web, people with disabilities can do more things themselves, without **having to rely on others.**”

*Source: Daily Tekk, February 24, 2012 - Why Web Accessibility Matters*

# Evolution of Web Accessibility Guidelines

- **WCAG 1.0 (1999)**
  - Finalized in 1999
  - Checkpoint driven
  - Priority 1, 2, and 3 (Level A, AA, and AAA)
  - Specific to HTML
- **Section 508 of the Rehabilitation Act (2001)** ([webaim.org/standards/508/checklist/](http://webaim.org/standards/508/checklist/))
  - Legalistic - easy to verify compliance
  - Applies to federal government
  - Very limited in scope
  - The de facto standard
  - Many states have adopted the guidelines
  - Currently being updated
- **WCAG 2.0 (2008)** ([webaim.org/standards/wcag/checklist/](http://webaim.org/standards/wcag/checklist/))
  - Finalized December 2008
  - Principles based
  - Technology agnostic
  - Maintains levels (A=essential, AA=important, AAA=less important)

## Section 504

“No otherwise qualified individual with a disability in the United States... shall, solely by reason of her or his disability, be excluded from participation in, be denied the benefits of, or be subjected to discrimination under any program or activity receiving Federal financial assistance.”

### Americans with Disabilities Act

- Pre-dates the web
- “Places of public accommodation”
- Current proposal to define ADA and web accessibility

“There is no doubt that the Internet sites of state and local government entities are covered by Title II of the ADA. Similarly, there is no doubt that the websites of recipients of federal financial assistance are covered by Section 504 of the Rehabilitation Act.”

*-Department of Justice*

# Accessibility and Drupal

- Drupal core has accessibility built in
- You can use accessibility as a criterion when choosing modules or themes
  - Look for the accessibility hashtag:  
#D7AX = Drupal 7 Accessibility eXperience
  - Easy to search for reported accessibility issues
- HTML/CSS tips in this presentation help you make your own additions accessible
- Drupal accessibility statement  
<https://drupal.org/about/accessibility>



# Accessibility Steps for Content Contributors

- Link text
- Headings
- Images and alternative text
- Tables
- Multimedia

# Link Text

- Use unique labels. Avoid “Click here,” “More info,” etc.
- Avoid using URLs as link text
  - Example: Does this link text tell you what you’ll find after you click?  
[http://www.section508.gov/docs/RonJones/multimedia\\_files/800x600/Slide1.html](http://www.section508.gov/docs/RonJones/multimedia_files/800x600/Slide1.html)
  - Instead, use the linked page’s title or description as link text
- Omit redundant title attributes `<a...title="Articles">Articles</a>`
- Any information provided about the link should be inside the link text  
`<a...>Article About Apples and Artichokes (4GB PDF)</a>`
  - Examples: format, file size, new window, external site

## Other thoughts:

- Absolutely unique, descriptive links are a WCAG 2.0 AAA requirement

# Headings

- Main heading on the page should be an `<h1>`
- Look for skipped levels (e.g., `<h2>` to `<h4>`)
- Use headings to outline structure of page
  - Don't use headings for decoration
  - Empty headings can be confusing
- In WAVE: Headings can also be viewed in the Outline tab in sidebar
  - We'll show this in our demo
- Related thought: Don't forget a unique and descriptive page `<title>`

## Other thoughts

- Heading levels are not a WCAG requirement

# Images and Alternative Text

- Alt text conveys content and function of the image
  - It should be succinct, accurate, and useful
- Can be presented two ways...
  - In the alt attribute of the img element. `<img alt="description of image">`
  - In the context or surroundings of the image itself.  
Then the image should have `<alt="">`
- Instead of saving decorated text as an image, use css styles to accomplish the same presentation

## Other thoughts

- Images that are the only thing in a link must always have a descriptive alternative text
  - Including form buttons that are images
- Image map hot-spots also need alt text
  - Server-side image maps cannot be made accessible

# Tables

- Layout tables
  - Reading order
- Data tables
  - Proper header structure

## Other thoughts

- Watch the first table cell
- Watch for empty table headers
- If the first row spans the whole table, it should be a <caption>
- Table summary attribute no longer recommended

# Table Markup

```
<table>
<caption>Committee Members</caption>
<tr>
  <th scope="col">Name</th>
  <th scope="col">Address</th>
  <th scope="col">Phone</th>
</tr>
<tr>
  <th scope="row">Jane Doe</th>
  <td>123 Main St., Ithaca, NY</td>
  <td>555-1212</td>
</tr>
[more rows]
</table>
```

# Demo: How can I check my site?

- WAVE tool from WebAIM: <http://wave.webaim.org/>
  - Paste in the URL of a site you want to test
- WebAIM also offers a Web Accessibility Toolbar (WAT)
  - Install on your Windows computer
  - Tests more features than the WAVE tool, but is not as up-to-date

The following apply to the entire page:

**WAVE**  
web accessibility evaluation tool

http://www.cio.cornell.edu/planning-go

Styles No Styles Contrast

**Summary**

WAVE has detected the following:

- 10 Errors
- 8 Alerts
- 9 Features
- 18 Structural Elements
- 0 HTML5 and ARIA
- 4 Contrast Errors

**Panel Options**

- DETAILS: A listing of all the WAVE icons in your page.
- DOCUMENTATION: Explanation of the WAVE icons and how you can make your page more accessible.
- OUTLINE: The heading structure of the web page.

**In This Section:**

- Planning & Governance **X2**
- IT Governance Council
- FABIT
- Governance Committees
- Advisory Committees

**IT Governance**

Skipped heading level  
A heading level is skipped.  
[More Information](#)

**h3 Governance structure**

The IT Governance model is based on an interrelated stack of Advisory Committees, Governance Committees, and a high-level Governance Council. It brings central IT **long alt** **IT**

**Governance structure: Top:**  
**IT Governance Council.**  
**Middle: 3 governance committees. Bottom: 3 advisory committees\*units**

**IT Governance Council  
Chief Information Officer**

**long alt** **IT**

# Multimedia

- Automatic captions vs. true captions
  - This funny short video shows dangers of relying on automatic captioning <http://www.youtube.com/watch?v=7ITUXVfTVOg>

## WCAG 2.0 terminology

- Media Alternative for Text
  - Media that presents no more information than is already presented in text (directly or via text alternatives)
  - Example: A video of a news story, but the entire news story (e.g., transcript) is presented on the same page
- Alternative for Time-based Media
  - Document including correctly sequenced text descriptions of time-based visual and auditory information and providing a means for achieving the outcomes of any time-based interaction
  - In English: A descriptive transcript



# Accessibility Steps for Web Designers

When you design a theme or template, worry about these:

- Keyboard navigation
- Reading order
- Color contrast
- Zoom page / enlarge text

If you're not a designer, be aware of these issues to discuss with the person designing your site.

# Keyboard Navigation

- Navigate the site using only the keyboard (Tab, Shift + Tab, Enter, etc.).
  - Is all functionality available?
  - Is the navigation order logical?
- Is a visible keyboard focus indicator/ outline present?

Demo: <http://www.w3.org/WAI/demos/bad/before/home.html>

## Other thoughts

- Be careful when default form control behavior is modified (auto-tabbing, “jump” select menus, etc.)
- Use buttons and links appropriately
- JavaScript “jump” menus
- Keyboard accessibility is different when using a screen reader
- Visible focus indicators are essential (even though WCAG 2 rates them as AA)
- Look for empty links or buttons
- Requiring JavaScript is not an accessibility issue (but it can be a usability one)

# “Skip Navigation” Link

- Is a link available?
- Does it function correctly?
- Is it visible on focus?
- Example from Cornell template:  

```
<div id="skipnav">Skip directly to:  
<a href="//www.cornell.edu/#maincontent">content</a> |  
<a href="//www.cornell.edu/#content">section navigation</a> |  
<a href="//www.cornell.edu/#search-form">search</a></div>
```

## Other thoughts

- To ensure the skip link is functioning correctly do the following:
  - Tab to link > Enter > Tab again
- Not required by 508 or WCAG, but still a good idea
- (Possibly) Benefits sighted users more than screen reader users
- Don't “display:none” the 'skip' link (not readable to screen readers)
  - Drupal's CSS class “.element-invisible” is a convenient way to provide text for screen readers only

# Mushrooms

The green mushrooms listed here are okay to eat.

The red mushrooms will kill you.

- Amanita
- Chanterelle
- Porcini
- Shitake
- Tylopilus

- Amanita
- Chanterelle
- Porcini
- Shitake
- Tylopilus

# Color Reliance

- Don't rely on color as the only visual distinction between items.
- Provide other means to indicate the difference.

## Contrast

- AA Ratios – 4.5:1, 3:1 for large text
- AAA Ratios – 7:1, 4.5:1
- Logos and “incidental” text exempted
- **Non-underlined links have additional requirements**
  - 3:1 contrast ratio between link and link text
  - AND the 4.5:1 ratios above
  - AND the links must present a “non-color designator” (typically underline) on mouse hover and keyboard focus

## Resources

- [webaim.org/resources/contrastchecker/](http://webaim.org/resources/contrastchecker/)
- Use Colorzilla or Firebug (or both) to find color values
- Contrast ratios are not perfect

# Zoom page/Enlarge text

- Enlarge the text size in your web browser to ~150%. Note that WCAG 2.0 requires 200%.
  - Is the page readable and usable?
  - Is horizontal scrolling minimized?
- Zoom the web page in your browser (enlarge fonts and images).
  - Is text in images readable?

## Other thoughts

- Zoom meets WCAG requirement for text resizing, so text resizing is almost impossible to fail
- Users that require 200% text are not using text sizing
- 200% text sizing is VERY difficult
- Consider ~130% - 150% text sizing and 200% zoom
- No WCAG or 508 requirements for minimum text size (but it's still a good idea)

# Other issues for designers and content contributors

- Page language (e.g., `<html lang="en">`)
  - “Language of parts”
- Shape, size, location - don't rely on users being able to see these
  - Avoid writing “see list at left” or “click on the blue box”
- Animated or auto-updating content - allow user to control it
- Strobing content - can trigger photosensitive seizures
  - Large area that flashes more than 3 times/second with bright contrast
- Sensory characteristics - can make text hard to read
  - Fuzzy font, textured background, etc.
- Validate HTML/CSS - excellent starting point before accessibility checking
  - <http://validator.w3.org/>
- PDFs used to be very difficult to make accessible, but no longer
  - Latest versions of Adobe Acrobat have much better tools

# Accessibility Steps for Developers

- Form labels
- Captcha



# Forms

- Drupal's Form API makes accessible field labeling easy
  - #title\_display property
- Click on the text next to a form field. If the field receives focus, the label is correctly associated.
  - This makes fields easier to click, especially checkboxes and radio buttons
- Buttons should not have labels
- 1:1 form to label relationship
- Support for ARIA labeling (e.g., aria-labelledby, aria-describedby) is good, but testing still required
  - Start with HTML labeling and use ARIA if necessary
- Fieldsets: use to create groups of elements in your form
  - Other ways to group large lists: tabindex="0" or ARIA

# CAPTCHA Is Hard to Make Accessible

- CAPTCHA is not the best way to block spam
  - Instead, use a simple “prove you’re a human” task such as adding two small numbers
  - Computers are getting smarter. We are not.
- CAPTCHA is permitted by WCAG if audio CAPTCHA is provided
- Audio CAPTCHA must be accessible, but that’s very hard to do:
  - Link must be clearly described
  - Audio must not interfere with screen reader
  - Etc.

# WAI-ARIA Overview

WAI-ARIA, the Accessible Rich Internet Applications Suite, defines a way to make Web content and Web applications more accessible to people with disabilities. It especially helps with dynamic content and advanced user interface controls developed with Ajax, HTML, JavaScript, and related technologies.

WAI-ARIA provides Web authors with the following:

- Roles to describe the type of widget presented, such as "menu," "treeitem," "slider," and "progressmeter"
- Roles to describe the structure of the Web page, such as headings, regions, and tables (grids)
- Properties to describe the state widgets are in, such as "checked" for a check box, or "haspopup" for a menu.
- Properties to define live regions of a page that are likely to get updates (such as stock quotes), as well as an interruption policy for those updates—for example, critical updates may be presented in an alert dialog box, and incidental updates occur within the page
- Properties for drag-and-drop that describe drag sources and drop targets
- A way to provide keyboard navigation for the Web objects and events, such as those mentioned above

<http://www.w3.org/WAI/intro/aria>

# Tools for Checking Accessibility

- Windows

- IE
  - Web Accessibility Toolbar
- Firefox
  - WAVE toolbar ([wave.webaim.org](http://wave.webaim.org))
  - Colorzilla
  - Web Developer Toolbar (optional)
- NVDA (screen reader)
  - [nvda-project.org](http://nvda-project.org)
  - [webaim.org/articles/nvda/](http://webaim.org/articles/nvda/)

- Mac

- Firefox
  - WAVE toolbar ([wave.webaim.org](http://wave.webaim.org))
  - Colorzilla
  - Web Developer Toolbar (optional)
- Safari settings
  - Safari > Preferences > Advanced > Check “Press Tab to highlight each item on a webpage”
- VoiceOver (screen reader)
  - [webaim.org/articles/voiceover/](http://webaim.org/articles/voiceover/)

# Testing Strategies

- Use online tools first to correct issues that tools can find
- Then have people test
  - Including testers with disabilities is ideal
  - But any group of testers is better than none
  - Educate first - testers need to know what to look for

## Known issues when testing

- Each tester with a disability can only verify what works in his/her specific situation
- Assistive technologies may auto-correct and hide issues that should be fixed
  - JAWS screen reader is notorious for this
- Testing with a screen reader yourself doesn't work well because you're not an expert user of that technology

## Other thoughts

- Some institutions have labs equipped with assistive technologies for testing
- Companies that do professional testing can teach us how to do it ourselves

# Articles about Accessibility in Drupal

- Accessibility tools and best practices for site builders  
<https://drupal.org/node/394094>
- Creating accessible structure and markup  
<https://drupal.org/node/1638020>
- Accessibility in Drupal - Special Topics Course  
<http://accessibility.micnap.com/part-3>
- A content editing tool with accessibility checking built in, as you type  
EditLive <http://ephox.com/editlive/demo>
  - Not a free tool, but has 30-day free trial

# Quick Videos about Who Benefits from Accessibility

- Keeping Web Accessibility in Mind (from WebAIM, 11 minutes)  
Users with various disabilities demonstrate their experiences  
<http://www.youtube.com/watch?v=yx7hdQqf8IE>
- A Personal Look at Accessibility in Higher Education (6 minutes)  
Highlights the experiences of students, faculty, and staff with disabilities  
<http://www.youtube.com/watch?v=PQGFshzLPXE>
- Web Accessibility: What Does It All Mean (3 minutes)  
<http://www.youtube.com/watch?v=bEM9Fn9aOG8>

## General Accessibility Resources

- WebAIM <http://webaim.org/>
- W3C's Web Accessibility Initiative, WAI <http://www.w3.org/WAI/>

# The End

This cat is not accessible!

