

Implementing WCAG 2.0 Requirements using HTML5 and ARIA 1.0 Techniques

Jon Gunderson

Division of Disability Resources and Educational Services

University of Illinois

jongund@illinois.edu

2014 Accessing Higher Ground

Presentation Home Page

Slides

1. Overview
2. Web Content Accessibility Guidelines 2.0
3. Accessible Rich Internet Applications (ARIA) 1.0
4. HTML 5
5. HTML 5: main element
6. Landmarks, HTML5 Section Elements and Headings
7. ARIA Landmarks
8. Landmark Example: Disability Resources and Educational Services (DRES) Website
9. Pitfalls Observed in Initial Use of Landmarks
10. DRES Home Page Landmarks
11. MAIN Landmark Design Patterns
12. NAVIGATION Landmark Design Patterns
13. Landmark Design Patterns
14. Use of Headings (H1-H6) in ARIA Landmarks
15. Landmark Example: Campus Information Technology and Education Services (CITES) Website
16. Landmark Example: International Business Machines (IBM)
17. HTML5 Section Elements
18. HTML5 to ARIA Mappings and Implementation Information
19. Headings in HTML5 are Designed for Outlining Content
20. HTML5 Headings Outline Example
21. Extensions for Viewing HTML5 Outlines
22. What About the Article Role and the Article Element?

- 23. Screen Reader Support
- 24. Link Usability 101
- 25. WCAG 2.0 Link Requirements
- 26. WCAG 2.0 Technique Examples
- 27. Accessible Name Calculation
- 28. Defining Link Context Information
- 29. WCAG Heading Context Example
- 30. Using aria-describedby to define context
- 31. Using aria-labelledby to define accessible name
- 32. Link Keyboard Focus Styling
- 33. WCAG 2.0 Requirements for Forms and Form Controls (Level A and AA)
- 34. ARIA and HTML5 form controls
- 35. Form Techniques: Example 1
- 36. Form Techniques: Example 2
- 37. Form Techniques: Example 3
- 38. Accessible Rich Internet Applications (ARIA) Basics
- 39. Tabindex and keyboard focus
- 40. Example Widget: ARIA Checkbox
- 41. Example Widget: ARIA Checkbox with Focus Styling
- 42. OpenAjax Accessibility Task Force
- 43. OpenAjax Evaluation Library Rulesets
- 44. Alnspector Sidebar for Firefox
- 45. Functional Accessibility Evaluator (FAE) 2.0
- 46. Functional Accessibility Evaluator: Auditor
- 47. Summary of FAE Auditor Results for Illinois
- 48. First Steps
- 49. Next Steps
- 50. Acknowledgements

Overview

- W3C Standards for web accessibility
- ARIA Landmarks Design Patterns
- Programmatic link identification and context using ARIA
- ARIA for accessible Forms
- What is an ARIA Widget?
- Open source tools for evaluating web accessibility

Slides at:
<http://presentations.cita.illinois.edu/2014-11-ahg-aria-html5/>

Web Content Accessibility Guidelines 2.0

Web Content Accessibility Guidelines 2.0

[\[contents\]](#)



Copyright 2014 University of Illinois

Web Content Accessibility Guidelines (WCAG) 2.0

W3C Recommendation 11 December 2008

This version:

<http://www.w3.org/TR/2008/REC-WCAG20-20081211/>

Latest version:

<http://www.w3.org/TR/WCAG20/>

Previous version:

<http://www.w3.org/TR/2008/PR-WCAG20-20081103/>

Editors:

Ben Caldwell, Trace R&D Center, University of Wisconsin-Madison

Michael Cooper, W3C

Loretta Guarino Reid, Google, Inc.

Gregg Vanderheiden, Trace R&D Center, University of Wisconsin-Madison

Previous Editors:

Wendy Chisholm (until July 2006 while at W3C)

John Slatin (until June 2006 while at Accessibility Institute, University of Texas at Austin)

Jason White (until June 2005 while at University of Melbourne)

Please refer to the [errata](#) for this document, which may include normative corrections.

See also [translations](#).

This document is also available in non-normative formats, available from [Alternate Versions of Web Content Accessibility Guidelines 2.0](#)

Accessible Rich Internet Applications (ARIA) 1.0

Accessible Rich Internet Applications 1.0



[\[contents\]](#)

Copyright 2014 University of Illinois

Accessible Rich Internet Applications (WAI-ARIA) 1.0

W3C Recommendation 20 March 2014

This version:

<http://www.w3.org/TR/2014/REC-wai-aria-20140320/>

Latest version:

<http://www.w3.org/TR/wai-aria/>

Previous version:

<http://www.w3.org/TR/2014/PR-wai-aria-20140206/>

Editors:

James Craig, Apple Inc.

Michael Cooper, W3C

Previous Editors:

Lisa Pappas, Society for Technical Communication

Rich Schwerdtfeger, IBM

Lisa Seeman, UB Access

Please check the [errata](#) for any errors or issues reported since publication.

This document is also available as a [single page](#) version.

See also [translations](#).

Copyright © 2008-2014 W3C® ([MIT](#), [ERCIM](#), [Keio](#), [Beihang](#)), All Rights Reserved. W3C [liability](#), [trademark](#) and [document use](#) rules apply.

HTML 5

HTML 5 Specification



Copyright 2014 University of Illinois
[Add developer-view styles](#)

HTML5

A vocabulary and associated APIs for HTML and XHTML

W3C Recommendation 28 October 2014

This Version:

<http://www.w3.org/TR/2014/REC-html5-20141028/>

Latest Published Version:

<http://www.w3.org/TR/html5/>

Latest Version of HTML:

<http://www.w3.org/TR/html/>

Latest Editor's Draft of HTML:

<http://www.w3.org/html/wg/drafts/html/master/>

Previous Version:

<http://www.w3.org/TR/2014/PR-html5-20140916/>

Previous Recommendation:

<http://www.w3.org/TR/1999/REC-html401-19991224/>

Editors:

WHATWG:

[Ian Hickson](#), Google, Inc.

W3C:

[Robin Berjon](#), W3C

[Steve Faulkner](#), The Paciello Group

[Travis Leithead](#), Microsoft Corporation

[Erika Doyle Navara](#), Microsoft Corporation

[Edward O'Connor](#), Apple Inc.

[Silvia Pfeiffer](#)

HTML 5: main element

HTML 5: main element

4.4.14 The `main` element

Copyright 2014 University of Illinois

Categories:

Flow content.
Palpable content.

Contexts in which this element can be used:

Where `flow content` is expected, but with no `article`, `aside`, `footer`, `header` or `nav` element ancestors.

Content model:

Flow content.

Content attributes:

Global attributes

Tag omission in text/html:

Neither tag is omissible

Allowed ARIA role attribute values:

`main` role (default - *do not set*) or `presentation`.

Allowed ARIA state and property attributes:

Global `aria-*` attributes
Any `aria-*` attributes [applicable to the allowed roles](#).

DOM interface:

Uses `HTMLElement`

The `main` element [represents](#) the **main content** of the `body` of a document or application. The main content area consists of content that is directly related to or expands upon the central topic of a document or central functionality of an application.

Note: The `main` element is not [sectioning content](#) and has no effect on the document [outline](#)

The main content area of a document includes content that is unique to that document and excludes content that is repeated across a set of documents such as site navigation links, copyright information, site logos and banners and search forms (unless the document or

Landmarks, HTML5 Section Elements and Headings

- ARIA Roles
- HTML 5 Section Elements
- Headings (H1-H6)

Copyright 2014 University of Illinois

ARIA Landmarks

ARIA Landmark Role Specification

- application
- banner
- complementary
- contentinfo
- form
- main
- navigation
- region (sub sections)
- search

Landmark Example: Disability Resources and Educational Services (DRES) Website

Landmark Example: DRES Website

THE DIVISION OF
DISABILITY RESOURCES & EDUCATIONAL SERVICES
College of Applied Health Sciences at the University of Illinois at Urbana-Champaign

Copyright 2014 University of Illinois

My Quick Links About Knowledge Center Contact

THINKING ABOUT ILLINOIS APPLYING FOR SERVICES GRADUATION & BEYOND

Academic Support >

Living Accommodations >

Getting Around >

Health >

Campus Life >

Athletics >


PIONEERS OF EMPOWERMENT

Spotlight [ALL SPOTLIGHTS »](#)

Brad Hedrick Retires
Brad Hedrick retired after 33 years of service to the Division of Disability Resources and Educational Resources. A reception was held on September 12, 2014 to honor his many accomplishments - see the video at:



Events Calendar [ALL EVENTS »](#)

General Therapy Group
Mon, November 17, 1:00 pm

Academic Lab
Mon, November 17, 3:30 pm

Academic Lab

Pitfalls Observed in Initial Use of Landmarks

- ARIA menu roles were used in flyout menus:
 - No keyboard support to manage focus
 - Menu roles override link role
- Nested Navigation Landmarks, resulting in too many navigation landmarks
- Landmarks indicated programmatic/styling sections, not semantic sections
- Some content in landmarks some content not in landmarks
- `article` role was used as a landmark role

DRES Home Page Landmarks

Disability Resources & Educational Services - University of Illinois - Mozilla Firefox
 File Edit View History Bookmarks Tools Accessibility Help
 Persons for Firefox | Harry Potter and... Disability Resources & Educational Se...
 http://dres-testing-csug3.devobjtech.com/

My Quick Links About Knowledge Center Contact

Go back one page Copyright 2014 University of Illinois
 Navigation Landmark label: Top Menu

Main Landmark Label: Disability Resources and Educational Services

DISABILITY RESOURCES & EDUCATIONAL SERVICES
 College of Applied Health Sciences at the University of Illinois at Urbana-Champaign

Thinking About Illinois Applying for Services Graduation & Beyond

Academic Support Living Accommodation Getting Around Health Campus Life Athletics

Navigation Landmark (Child of main) Label: Services and Resources

NEERS OF EMPOWERMENT

Spotlight ALL SPOTLIGHTS » Region Landmark (Child of main) Label: Spotlight

Test Spq Photo of a DRES More

◀ Prev Next ▶

Calendar ALL EVENTS » Region Landmark (child of main) Label: Calendar

16th Mahomet Half Marathon & 5K Sat, August 20 - 7:00 am Med... Sun, October 1 Sat, October 1 Sat, October 30

A History of Firsts ALL STORIES » Region Landmark label: History

give online

GIVE US FEEDBACK »

DISABILITY SERVICES AT THE UNIVERSITY OF ILLINOIS

1207 S. Oak St. Champaign, IL 61820 Phone: (217) 244-51820 Hours: 8:00 AM - 4:30 PM DRES access to campus maps

About DRES Resources for Faculty/Staff with Disabilities

Region Landmark (child of complementary) Label: Contact and Maps

Frequently Asked Questions Menu Navigation Landmark (child of complementary) Label: Frequently Asked Questions

College of Applied Health Sciences Grievances Staff Directory A-to-Z Directory

Complementary Landmark

MAIN Landmark Design Patterns

- Typically one `main` landmark per page
- Each `main` landmark has a `h1` element
- If more than one `h1` element and `main` landmarks use `aria-labelledby` for naming the `main` landmark

One Main Landmark

```
<div role="main">  
  
<h1>title for main content<h1>  
  
.... main content....  
  
</div>
```

Multiple Main Landmarks

```
<div role="main" aria-labelledby="title1">  
  
<h1 id="title1">title for main content 1<h1>  
  
.... main content....  
  
</div> .... <div role="main" aria-labelledby="title2">  
  
<h1 id="title2">title for main content 2<h1>  
  
.... main content....  
  
</div>
```

NAVIGATION Landmark Design Patterns

- Use `navigation` landmark on a element *containing* the list and `h2` element
- Each `navigation` landmark should have a `h2` element before the list of links
- If there is more than one `navigation` landmark use `aria-labelledby` for labeling the landmark with the `h2` element
- Use `ul` element and `li` elements for the list of links in the navigation bar

One `navigation` landmark

```
<div role="navigation">  
  
  <h2>title for navigation</h2>  
  
  <ul>  
    <li><a href="page1.html">Link 1</a></li>  
    <li><a href="page2.html">Link 2</a></li>  
    <li><a href="page3.html">Link 3</a></li>  
    <li><a href="page4.html">Link 4</a></li>  
    ....  
  </ul>  
</div>
```

More than one `navigation` landmark

```
<div role="navigation" aria-labelledby="nav1">  
  
  <h2 id="nav1">title for navigation 1</h2>  
  
  <ul>  
    <li><a href="page11.html">Link 1</a></li>  
    <li><a href="page12.html">Link 2</a></li>  
    <li><a href="page13.html">Link 3</a></li>  
    <li><a href="page14.html">Link 4</a></li>  
    ....  
  </ul>  
</div>
```

```
<div role="navigation" aria-labelledby="nav2">  
  <h2 id="nav2">title for navigation 2</h2>  
  
  <ul>  
    <li><a href="page21.html">Page 10</a></li>  
    <li><a href="page22.html">Page 11</a></li>  
    <li><a href="page23.html">Page 12</a></li>  
    <li><a href="page24.html">Page 13</a></li>  
    ....  
  </ul>  
</div>
```

Landmark Design Patterns

- All content should be contained in a landmark:
 - banner (top level)
 - complementary
 - contentinfo (top level)
 - main (top level)
 - navigation
 - search
- *Must* use labels when a landmark of the same type is used more than once
- *Should* not nest landmarks of the same type

Use of Headings (H1-H6) in ARIA Landmarks

- main landmarks should contain h1 element at the beginning of section:
 - h1 element content describes the content or the purpose of the page
 - Provides a redundant means for assistive technology users to navigate to main content
 - Use headers in a structural way or in combination with HTML5 section elements
 - If more than one main landmark on the page use aria-labelledby to use the heading as accessible name to the landmark
- navigation and complementary landmarks should contain h2 element at the beginning of section:
 - h2 element content describes the content or the purpose of the page
 - Provides a redundant means for assistive technology users to navigate to content
 - If more than one navigation or complementary landmark on the page use aria-labelledby to use the heading as accessible name for each landmark
- contentinfo landmark should contain h2 element at the beginning of section:
 - h2 element identifies the section as footer information
 - Provides a redundant means for assistive technology users to navigate to content
- banner landmark may contain h1 element at the beginning of section:
 - h1 element identifies the website information
 - Provides a redundant means for assistive technology users to orient to the website

Copyright 2014 University of Illinois

Landmark Example: Campus Information Technology and Education Services (CITES) Website

Landmark Example: Campus Information Technology and Education Services (CITES) Website

HOME ABOUT US CONTACT INFO WEBSTORE CITES HELP DESK Stay Updated: NEWS FEEDS FACEBOOK STATUS

CITES Campus Information Technologies and Educational Services University of Illinois Urbana-Champaign

Copyright 2014 University of Illinois powered by Google Search

Help & Support

Password Reset

 Reset Your Password

Services & Info

Unified Communications
Learn more about UC at Illinois

Accounts and Passwords
On this page you can find help with passwords, links to account login pages, and help to [reset your password](#).

Connecting to the Internet
CITES Networking provides wired and wireless network services to students, faculty and staff at the University of Illinois at Urbana-Champaign.

Safety and Security
Here you can find information about computer security and life safety services on campus.

Email
Here you can find help with Google Apps@Illinois, MS Exchange, Spam Control, and other email issues.

Collaboration and Calendaring

U of I and Mayo Clinic to Revolutionize Gene Data Research
Massive amounts of big data will be shared and analyzed across the Campus Advanced Research Network Environment.



<< Prev 2 of 5 Next >>

Item Title
Posted in [CITES News](#)
Monday, November 17, 2014 - 17:45
For the time being, people wanting to use the Illinois webmail service will need to use a slightly different web address to reach the webpage.

Team Folders Available for U of I Box
Posted in [CITES News](#)
Thursday, November 13, 2014 - 15:00
Team Folders make it easier for departments, organizations and groups to collectively...

Popular Topics

- Pinnacle Service Management System
- Accounts and Passwords
- Virtual Private Networking (VPN)
- Email
- [Guide to Google Apps @ Illinois Email Accounts](#)
- CITES Help Desk
- CITES Exchange Services
- Email Web Interface Links
- Your NetID
- Guest Services
- Illinois Compass 2g

Network Access, Computer Labs, and Classrooms

IllinoisNet Wireless Locations Go

Upcoming Events

Tue 11/18/14

[U of I Webstore Training and Events](#)
Access 2013: Getting Started
10:00 AM

[U of I Webstore Training and Events](#)

Landmark Example: International Business Machines (IBM)

Landmark Example: International Business Machines (IBM)

United States Welcome | IBM Sign in / Register |

IBM Industries & solutions Services Products Support & downloads My IBM Search



Copyright 2014 University of Illinois

Why would your fridge need a password?

More intelligent devices means more risk from the Internet of Things

Learn more in the latest IBM X-Force Threat Intelligence Quarterly →



@IBM: You are invited to experience a #NewWaytoWork. Register for tomorrow's livestream: <http://t.co/ZLrYjVBQto>



Follow IBM on: 

Connect with us	Key topics	Information for	Try & buy	About IBM	Popular links
<ul style="list-style-type: none"> ■ Create a profile ■ Communities ■ Academic collaboration ■ Research collaboration ■ IBM Next and present 	<ul style="list-style-type: none"> ■ Smarter Planet ■ Analytics ■ Big Data ■ IBM Cloud ■ IBM PureSystems 	<ul style="list-style-type: none"> ■ C-suite executives ■ Industries ■ Small and medium business solutions ■ Developers 	<ul style="list-style-type: none"> ■ IBM Cloud marketplace ■ Cloud solutions ■ Bluemix ■ SoftLayer Cloud ■ Offer your services 	<ul style="list-style-type: none"> ■ Latest news ■ IBM's Centennial ■ IBM Research ■ Case studies ■ Corporate responsibility 	<ul style="list-style-type: none"> ■ Apple + IBM partnership ■ Careers ■ Fix central ■ Made with IBM ■ Personal Advantages

HTML5 Section Elements

Proposed HTML5 section element mapping to ARIA roles

HTML5 section elements and allowed ARIA roles

Section Element	Allowed ARIA Roles
address	contentinfo region (<i>proposed</i>) presentation
article	article main document application region (<i>proposed</i>) presentation
aside	complementary note search region (<i>proposed</i>) presentation
footer	contentinfo presentation
header	banner presentation
main	main presentation
nav	navigation region presentation
section	region alert alertdialog application contentinfo dialog document log main marquee presentation search status presentation

HTML5 to ARIA Mappings and Implementation Information

- Proposed HTML5 to ARIA Mappings
- HTML 5 Accessibility Mapping Implementation (Steve Faulkner, Pacilleo Group)

Copyright 2014 University of Illinois

Headings in HTML5 are Designed for Outlining Content

- HTML5 Outlining Algorithm
- Relative heading levels in sections define the outline structure
- Search engine optimization
- Mobile technology outlines
- Improved and rationale structure supports accessibility
- Accessibility APIs will support heading outlining

HTML5 Headings Outline Example

Example from HTML5 specification

HTML5 Outline Algorithm

1. Apples (H1 in accessibility API)
 1. Taste (H2 in accessibility API)
 1. Sweet (H3 in accessibility API)
 2. Color (H2 in accessibility API)

```
<body>
  <h1>Apples</h1>
  <p>Apples are fruit.</p>
  <section>
    <h1>Taste</h1>
    <p>They taste lovely.</p>
    <section>
      <h1>Sweet</h1>
      <p>Red apples are sweeter than green ones.</p>
    </section>
  </section>
  <section>
    <h1>Color</h1>
    <p>Apples come in various colors.</p>
  </section>
</body>
```

same as

```
<body>
  <h1>Apples</h1>
  <p>Apples are fruit.</p>
  <section>
    <h2>Taste</h2>
    <p>They taste lovely.</p>
    <section>
      <h3>Sweet</h3>
      <p>Red apples are sweeter than green ones.</p>
    </section>
  </section>
  <section>
    <h2>Color</h2>
    <p>Apples come in various colors.</p>
  </section>
</body>
```

Copyright 2014 University of Illinois

Extensions for Viewing HTML5 Outlines

- [HTML5 Outliner for Chrome](#)
- [HTML5 Outliner for Firefox](#)

Copyright 2014 University of Illinois

What About the Article Role and the Article Element?

- `article` role (ARIA)
 - Is **NOT** a landmark element
 - Nested discussion lists (e.g. forum post, a blog entry or a user-submitted comment)
- `article` element (HTML5)
 - HTML5 section element
 - Self-contained compositions (e.g. magazine or newspaper article)
 - Nested discussion lists (e.g. forum post, a blog entry or a user-submitted comment)
 - Interactive widget or gadget

Screen Reader Support

Jaws 15 with Firefox/IE`on MS Windows

- Penn State: JAWS commands for web
- WebAIM: JAWS commands for web
- Freedom Scientific: JAWS commands for web

JAWS 15 Landmark Keyboard Information

- Region List INSERT+CTRL+R
- Q: Next Region
- R: Main Region

NVDA with Firefox`on MS Windows

- Penn State: NVDA Keyboard Commands
- Webaim: NVDA Keyboard Commands
- NVDA: Web Browse Keyboard Commands

NVDA Landmark Keyboard Information

- d: next landmark
- shift+d: previous landmark
- NVDA+f7: List of elements including landmarks

Voice over for Safari on OS X

- Penn State: Voice Over for OS X
- Apple Voice Over Commands

Apple Support for Landmarks

- Voice Over does not support `region` roles as landmarks

Link Usability 101

- What makes a good link?
- Why does someone follow a link?

Copyright 2014 University of Illinois

WCAG 2.0 Link Requirements

2.4.4 Link Purpose (In Context)

WCAG 2.0 Level A requirement

The purpose of each link can be determined from the link text alone or from the link text together with its programmatically determined link context, except where the purpose of the link would be ambiguous to users in general.

2.4.9 Link Purpose (Link Only)

WCAG 2.0 Level AAA requirement

A mechanism is available to allow the purpose of each link to be identified from link text alone, except where the purpose of the link would be ambiguous to users in general.

WCAG 2.0 Technique Examples

- Headings
- Paragraph
- Table cell
- List item

Copyright 2014 University of Illinois

Accessible Name Calculation

- Child text nodes of a element
- aria-label attribute
- aria-labelledby attribute
- title attribute (rare)

Copyright 2014 University of Illinois

Defining Link Context Information

- `aria-describedby` attribute
- `title` attribute (tooltip)

Copyright 2014 University of Illinois

WCAG Heading Context Example

Links with Heading Context

Link Context Example

Royal Palm Hotel

[Map](#) [Photos](#) [Directions](#) [Guest reviews](#) [Book now](#)

Hotel Three Rivers

[Map](#) [Photos](#) [Directions](#) [Guest reviews](#) [Book now](#)

Source Code

```
<h3><a href="royal_palm_hotel.html">Royal Palm Hotel</a></h2>
<ul class="horizontal">
  <li><a href="royal_palm_hotel_map.html">Map</a></li>
  <li><a href="royal_palm_hotel_photos.html">Photos</a></li>
  <li><a href="royal_palm_hotel_directions.html">Directions</a></li>
  <li><a href="royal_palm_hotel_reviews.html">Guest reviews</a></li>
  <li><a href="royal_palm_hotel_book.html">Book now</a></li>
</ul>

<h3><a href="hotel_three_rivers.html">Hotel Three Rivers</a></h2>
<ul class="horizontal">
  <li><a href="hotel_three_rivers_map.html">Map</a></li>
  <li><a href="hotel_three_rivers_photos.html">Photos</a></li>
  <li><a href="hotel_three_rivers_directions.html">Directions</a></li>
  <li><a href="hotel_three_rivers_reviews.html">Guest reviews</a></li>
  <li><a href="hotel_three_rivers_book.html">Book now</a></li>
</ul>
```

Copyright 2014 University of Illinois

Using aria-describedby to define context

Link Context Example

Royal Palm Hotel

Map Photos Directions Guest reviews Book now

Hotel Three Rivers

Map Photos Directions Guest reviews Book now

Source Code

```
<h3><a id="ref1" href="royal_palm_hotel.html">Royal Palm Hotel</a></h2>
<ul class="horizontal">
  <li><a href="royal_palm_hotel_map.html" aria-describedby="ref1">Map</a></li>
  <li><a href="royal_palm_hotel_photos.html" aria-
describedby="ref1">Photos</a></li>
  <li><a href="royal_palm_hotel_directions.html" aria-
describedby="ref1">Directions</a></li>
  <li><a href="royal_palm_hotel_reviews.html" aria-describedby="ref1">Guest
reviews</a></li>
  <li><a href="royal_palm_hotel_book.html" aria-describedby="ref1">Book
now</a></li>
</ul>

<h3><a id="ref2" href="hotel_three_rivers.html">Hotel Three Rivers</a></h2>
<ul class="horizontal">
  <li><a href="hotel_three_rivers_map.html" aria-describedby="ref2">Map</a>
</li>
  <li><a href="hotel_three_rivers_photos.html" aria-
describedby="ref2">Photos</a></li>
  <li><a href="hotel_three_rivers_directions.html" aria-
describedby="ref2">Directions</a></li>
  <li><a href="hotel_three_rivers_reviews.html" aria-describedby="ref2">Guest
reviews</a></li>
  <li><a href="hotel_three_rivers_book.html" aria-describedby="ref2">Book
now</a></li>
</ul>
```

Copyright 2014 University of Illinois

Using aria-labelledby to define accessible name

Link Context Example

Royal Palm Hotel

[Map](#) [Photos](#) [Directions](#) [Guest reviews](#) [Book now](#)

Hotel Three Rivers

[Map](#) [Photos](#) [Directions](#) [Guest reviews](#) [Book now](#)

Source Code

```
<h3><a id="ref1" href="royal_palm_hotel.html">Royal Palm Hotel</a></h2>
<ul class="horizontal">
  <li><a id="ref11" href="royal_palm_hotel_map.html" aria-labelledby="ref11 ref1">Map</a></li>
  <li><a id="ref12" href="royal_palm_hotel_photos.html" aria-labelledby="ref12 ref1">Photos</a></li>
  <li><a id="ref13" href="royal_palm_hotel_directions.html" aria-labelledby="ref13 ref1">Directions</a></li>
  <li><a id="ref14" href="royal_palm_hotel_reviews.html" aria-labelledby="ref14 ref1">Guest reviews</a></li>
  <li><a id="ref15" href="royal_palm_hotel_book.html" aria-labelledby="ref15 ref1">Book now</a></li>
</ul>

<h3><a id="ref2" href="hotel_three_rivers.html">Hotel Three Rivers</a></h2>
<ul class="horizontal">
  <li><a id="ref21" href="hotel_three_rivers_map.html" aria-labelledby="ref21 ref2">Map</a></li>
  <li><a id="ref22" href="hotel_three_rivers_photos.html" aria-labelledby="ref22 ref2">Photos</a></li>
  <li><a id="ref23" href="hotel_three_rivers_directions.html" aria-labelledby="ref23 ref2">Directions</a></li>
  <li><a id="ref24" href="hotel_three_rivers_reviews.html" aria-labelledby="ref24 ref2">Guest reviews</a></li>
  <li><a id="ref25" href="hotel_three_rivers_book.html" aria-labelledby="ref25 ref2">Book now</a></li>
</ul>
```

Copyright 2014 University of Illinois

Link Keyboard Focus Styling

- Use CSS Pseudo selectors :focus and :active to style link with focus

Copyright 2014 University of Illinois

WCAG 2.0 Requirements for Forms and Form Controls (Level A and AA)

- 2.4.6 Headings and Labels (AA)
- 3.2.1 On Focus (A)
- 3.2.2 On Input (A)
- 3.3.1 Error Identification (A)
- 3.3.2 Labels or Instructions (A)
- 3.3.3 Error Suggestion (AA)
- 4.4.1 Parsing (A)

ARIA and HTML5 form controls

Labeling form controls

- `label` element (HTML 4.01)
- `title` attribute (HTML 4.01)
- `fieldset/legend` elements (HTML 4.01)
- `aria-labelledby` attribute (ARIA 1.0)
- `aria-label` attribute (ARIA 1.0)

Instructions

- `tabindex="0"` attribute to add instructions to tab order (convention)
- `aria-describedby` attribute (ARIA 1.0)

Identifying Required and Invalid Form Controls

- Image with ALT Text
- Text content (hidden or visible)
- `aria-required` attribute (ARIA 1.0)
- `aria-invalid` attribute (ARIA 1.0)
- `required` attribute (HTML 5)
- alert role for announcing errors

Form Techniques: Example 1

Example 1

Form Control Accessibility

[Example 1](#) [Example 2](#) [Example 3](#)

Example 1

Features

- HTML4 Techniques
- `label` element for labeling
- `fieldset/legend` elements for labeling groups of controls (e.g. crust and toppings)
- Image used to indicate required fields
- Dialog box for errors

Pizza Order Form

* means required field

Name *

Address *

Phone *

Form Techniques: Example 2

Example 2

Form Control Accessibility

[Example 1](#) [Example 2](#) [Example 3](#)

Example 2

Features

- HTML4 Techniques
- `label` element for labeling
- Required fields is a part of label text
- `fieldset/legend` elements for labeling groups of controls (e.g. crust and toppings)
- Error feedback through a list of links with error messages and headings to indicate the number of errors
- `focus` and `blur` events to improve focus highlighting of form controls

Pizza Order Form

Name (required)

Address (required)

Phone (required)

Form Techniques: Example 3

Example 3

Form Control Accessibility

[Example 1](#) [Example 2](#) [Example 3](#)

Example 3

Features

- ARIA Accessibility Techniques
- `aria-label` and `aria-labelledby` attribute for labeling
- Error feedback through pop messages
- `aria-describedby` used to reference error messages and instructions (e.g special)
- `aria-required` and `aria-invalid` attributes to mark required and invalid fields
- `focus` and `blur` events to improve focus highlighting of form controls

Pizza Order Form

Name

Address

Phone

Accessible Rich Internet Applications (ARIA) Basics

- **TABINDEX:** Used for identifying elements that can receive keyboard focus.
- **focus():** Method to move keyboard focus to a DOM node using javascript
- **ROLE:** Attribute to define the type of widget in the web resource.
 - XHTML Role Attribute Module
 - Roles for Accessible Rich Internet Applications (WAI-ARIA Roles)
- **Properties and States**
 - States and Properties Module for Accessible Rich Internet Applications (WAI-ARIA States and Properties)
 - aria-checked
 - aria-labelledby
 - aria-describedby
 - aria-haspopup
 - aria-expanded
 - many others...
- WAI-ARIA 1.0 Authoring Practices

Tabindex and keyboard focus

- `tabindex` attribute is part of the HTML 4 specification
- Support in **all** modern major browsers
- Values:
 - -1: Support keyboard focus/blur events, but do not add to tab order
 - 0: Support keyboard focus/blur events and add to tab order
 - >0: Support keyboard focus/blur events and add to tab order and tab using the numerical order of the tabindex values
- The `tabindex` is not currently valid for most elements
- Style focus visually to make it easy to find what has focus (i.e. do NOT rely on browser focus styling of focus)

Example Widget: ARIA Checkbox

ARIA checkbox examples

Features

- Responds to mouse events using `onClick` event
- Responds to keyboard events using `onKeyDown` event
- `tabindex=0` makes widget keyboard focusable and include in tab order of page
- `role="checkbox"` identifies widget to assistive technologies as a "checkbox"
- `aria-checked` attribute identifies the state as being "checked" or "unchecked"
- Accessible name comes from child text content of the `div[role="checkbox"]` element
- CSS attribute selectors are used to keep the visual indication of checked in sync with accessibility API

- [Example 1](#)
- [Example 2](#)

ARIA Checkbox Example

Standard checkbox 1

Standard checkbox 2

ARIA Checkbox 1

ARIA Checkbox 2

Copyright 2014 University of Illinois

Example Widget: ARIA Checkbox with Focus Styling

ARIA checkbox examples with focus styling

Additional Features

- Use `:focus` pseudo element focusing only focuses the checkbox and not the label
- Use `onFocus` and `onBlur` events to add and remove class for identifying focus
- Use `:hover` pseudo class to style checkboxes for mouse overs
- Standard checkbox 3 doesn't have label so you have to click directly on the checkbox image to change state

[Example 1](#) [Example 2](#)

ARIA Checkbox Example with Focus Styling

Standard checkbox 1 (input:focus pseudo element for focus styling)

Standard checkbox 2 (focus and blur event label styling)

Standard checkbox 3 (no label)

ARIA Checkbox 1

ARIA Checkbox 2

Copyright 2014 University of Illinois

OpenAjax Accessibility Task Force

- Working on new rule development
- Support ARIA and HTML5

OpenAjax Alliance Accessibility Task Force

The screenshot shows a Wikipedia-style page for the "Accessibility" category of the OpenAjax Alliance. The page has a sidebar on the left containing a navigation tree for the category. The main content area includes links to other categories like "Downloads" and "Evaluation Library Objects and APIs". The footer of the page contains copyright information for the University of Illinois.

OpenAjax alliance
Standardizing Ajax Development

page discussion view source history Log in

Accessibility

[Accessibility Home](#) | [Downloads](#) | [Evaluation Library Objects and APIs](#) | [Rule Format](#) | [Issue Tracker](#)

Contents [hide]

- [1 Accessibility Tools Working Group](#)
- [2 Objectives](#)
- [3 Current Documents](#)
- [4 Support Resources](#)
- [5 Tools using OAA Rulesets](#)
- [6 Related Resources](#)
- [7 Discussions](#)
- [8 Participating Companies and Organizations](#)
- [9 Committee Email list](#)
- [10 Teleconference information](#)
 - [10.1 Minutes from previous calls](#)
 - [10.1.1 Instructions to post minutes here](#)
 - [10.1.2 Minutes](#)
 - [10.1.2.1 2009](#)
 - [10.1.2.2 2010](#)
 - [10.1.2.3 2011](#)
 - [10.1.2.4 2012](#)
 - [10.1.2.5 2013](#)
 - [10.1.2.6 2014](#)
 - [11 Formal Working Group Formation](#)

Copyright 2014 University of Illinois

Accessibility Tools Working Group

This is the Wiki for the Open AJAX Alliance Accessibility Tools Task Force. The growing adoption of Rich Internet Applications (RIA) has resulted in the development of new accessibility standards and specifications such as the [W3C WAI Accessible Rich Internet Application Specification \(WAI-ARIA\)](#) and the [W3C Web Content Accessibility Guidelines 2.0 \(WCAG2\)](#). Although accessibility tool providers and vendors are moving to support WCAG 2, compliance is not fully addressed or understood in a RIA environment. For example, in WCAG2, [Guideline 4.1 Compatible: Maximize compatibility with current and future user agents, including assistive technologies](#), there is a subtle dependence upon the WAI-ARIA specification in success criterion 4.1.2. This requirement dictates that, for all user interface components, the name and role of these components are determined and that the states, properties, and values of such components can be programmatically determined and possibly

www.openajax.org/member/wiki/Accessibility#2010

OpenAjax Evaluation Library Rulesets

OAA Ruleset: WCAG + HTML5 + ARIA by Rule Category

fae Functional Accessibility Evaluator 2.0 beta (Public Testing Version)
Version 2.0.0a
A project of the Open Accessibility Alliance and OpenAjax Accessibility Task Force

About FAE Rulesets Register Login

Run FAE Rulesets Copyright 2014 University of Illinois

Rulesets

Ruleset: WCAG + ARIA
WCAG 2.0 Level A and AA requirements using ARIA and HTML5 accessibility features.

Evaluation of WCAG 2.0 Level A and AA conformance including the use of ARIA and HTML5 accessibility features. ARIA Landmarks support the identification of the types and sections of content on the page, and is the first step in using ARIA technologies to improve accessibility. This ruleset includes the more complex ARIA widget validation rules and assumes developers understand the focus management and keyboard support needed for implementing ARIA-enabled widgets. If you don't understand the design of ARIA-enabled widgets, you should remove widget roles from your design and instead use the WCAG + Landmarks ruleset as a first step in understanding the accessibility features of ARIA technologies. As you develop skill and understanding of ARIA accessibility features you can switch back to this ruleset.

- **Version:** 0.9.4 Beta
- **Number of Rules:** 117

Ruleset: WCAG + Landmarks
WCAG 2.0 Level A and AA requirements using ARIA Landmarks and HTML5 accessibility features.

Evaluation of WCAG 2.0 Level A and AA conformance including the use of ARIA Landmarks and HTML5 accessibility features. ARIA Landmarks support the identification of the types and sections of content on the page, and is the first step in using ARIA technologies to improve accessibility. This ruleset does not include the more complex ARIA widget validation rules found in the WCAG + ARIA ruleset. Instead it includes rules that produce warnings to discourage developers from using widget roles until they fully understand the focus management and keyboard support needed for implementing ARIA-enabled widgets. If you understand the design of ARIA-enabled widgets and are including widget roles in

About Options
Overview
Rulesets
Usage Statistics
Concepts and Terms
Disclaimer
Version History
Contact

AInspector Sidebar for Firefox

- Designed for UX designers, developers and quality assurance testers to understand the accessibility of a specific page
- Open source
- Update your version of Firefox to the latest version to ensure compatibility
- Analyzing current web page DOM for accessibility features
- Organizes rules by Rule Categories and WCAG Guidelines
- Map rules to WCAG 2.0 Single A and AA requirements
- Highlighting element information

AInspector Sidebar form Mozilla Add-ons Website

The screenshot shows the Mozilla Add-ons website. At the top, there's a navigation bar with links for 'Register' or 'Log in', 'Other Applications', and the 'mozilla' logo. Below the navigation is the 'ADD-ONS' header with links for 'EXTENSIONS', 'THEMES', 'COLLECTIONS', and 'MORE...'. A search bar with the placeholder 'search for add-ons' is also present. A yellow banner at the bottom of the header area encourages users to download Mozilla Firefox. The main content area displays the 'AInspector Sidebar 0.16.4' add-on page. It features a green puzzle piece icon, the add-on name, the developer's name 'Nicholas Hoyt', and a brief description stating that it evaluates accessibility compliance based on WCAG 2.0 requirements. A 'Download Now' button is visible. Below the description, a note says the add-on has been preliminarily reviewed by Mozilla. The page also indicates compatibility with Firefox 7.0 - 28.* and provides a link to view other versions. To the right of the add-on details, there are user reviews (1 user review, 174 users), and links to 'Add to collection' and 'Share this Add-on'. At the bottom, two screenshots of the add-on in use are shown, displaying weather forecast data from the National Weather Service.

Functional Accessibility Evaluator (FAE)

2.0

- Designed for project managers, UX designers, developers and quality assurance testers to understand the level of accessibility of a web site they are developing or have administration responsibility
- Open source
- Analyzing multiple pages for accessibility features
- Organizes rules by Rule Categories, Rule Scope and WCAG Guidelines
- Website summary feature and implementation status
- Map rules to WCAG 2.0 Single A and AA requirements

FAE 2.0 Test Prototype

fae

Functional Accessibility Evaluator 2.0 beta (Public Testing Version)
Version 2.0.0a

A project of the Open Accessibility Alliance and OpenAjax Accessibility Task Force

About FAE Rulesets My Account Logout: jongund

<p>Run FAE</p> <p>Report Processing</p> <p>Archived Reports</p> <p>Manage Reports</p> <p>Reports</p> <p>Summary</p> <p>All Pages</p> <p>Page : 1</p> <p>Previous Next</p> <p>First Last</p> <p>View Options</p> <p>Rule Categories</p> <p>WCAG Guidelines</p> <p>Rule Scope</p> <p>View Filters</p> <p>All Categories</p> <p>Landmarks</p> <p>Headings</p> <p>Styling/Content</p> <p>Images</p> <p>Links</p>	<p>Run FAE: Website</p> <p>URL <input type="text" value="http://"/></p> <p>Title <input type="text"/></p> <p>Depth of Evaluation</p> <p><input type="radio"/> Top-level page only</p> <p><input checked="" type="radio"/> Include second-level pages</p> <p><input type="radio"/> Include third-level pages</p> <p>Follow Links in</p> <p><input checked="" type="radio"/> Specified domain only</p> <p><input type="radio"/> Next-level subdomains</p> <p>Ruleset</p> <p><input checked="" type="radio"/> WCAG + ARIA</p> <p><input type="radio"/> WCAG + Landmarks</p> <p><input type="button" value="Evaluate"/></p>	<p>Copyright 2014 University of Illinois</p>
---	---	--

Functional Accessibility Evaluator: Auditor

- Designed for administrators to get a high level view of campus accessibility
- Open source
- Analyzing multiple *websites* for accessibility features
- Organizational summary feature and implementation status

Illinois Audit From September 20th

Copyright 2014 University of Illinois

Summary of FAE Auditor Results for Illinois

Total websites: 467

Total pages: 25,932

Feature	Percent Websites	Description
Landmarks	>90%	Websites not using landmarks
Headings	~50%	Websites using hidden and/or empty headings (e.g. not in an accessible way), maybe being used to satisfy FAE 1.1 without understanding access
Styling/Content	~50%	Websites having some color contrast issues
Images	>50%	Websites are missing ALT attribute on images
Links	>98%	Websites have no accessible name for some links (probably Facebook and other social media links)
Tables	~70%	Websites with tables not being identified for layout or data; and most data tables have problems
Forms	~50%	Websites with missing labels for form controls
Widgets/Scripting	~50%	Websites with onClick and mouse events on non-interactive elements (e.g. div or span element) and have no widget ROLE describing their behavior or keyboard support
		Websites that may have some type of media

Audio/Video	~25%	content
Keyboard Support	~%50	Websites with widget roles that do not have keyboard event handlers

Copyright 2014 University of Illinois

First Steps

- Working with specific web developers on campus to bring their web resources into WCAG 2.0 compliance:
 - Disability Services
 - IT Accessibility
 - College of Applied Health Studies
 - Library
 - Office of Public Affairs
- Verifying the accessibility features of common web development libraries (e.g. bootstrap.js and jquery-ui) and adding supplemental scripts to improve accessibility
- Testing and validating the OpenAjax rules (Issue Tracker)
- Testing the and improving the utility and usability of the open source tools