LEVEL access

STATIC VS. INTERACTIVE WIDGET ROLES: ENSURING PROPER FUNCTIONALITY IN ARIA

Washington, D.C. Office
1593 Spring Hill Road, Suite 720
Vienna, VA 22182

Silicon Valley Office
114 Sansome Street, Suite 950
San Francisco, CA 94104

New England Office
250 Commercial Street, Suite 3007A
Manchester, NH 03101

Static vs. Interactive Widget Roles: Ensuring Proper Functionality in ARIA

CSUN presenters: Bryan Garaventa and Terri Fellers, Level Access (Formertly known as SSB BART Group, Inc.)

Using the ARIA Role Matrices as a guideline, developers at all levels will learn the use of ARIA roles from the basic to the advanced.

CSUN 2017 presentation download: https://github.com/accdc/csun-2017

What is ARIA

ARIA stands for Accessible Rich Internet Applications, and is a technical specification published by the World Wide Web Consortium (W3C) Web Accessibility Initiative (WAI). To learn more about how ARIA works, download the presentation "The Accessibility Tree: How ARIA Works": https://github.com/accdc/csun-2015

ARIA Categories

There are three primary categories of ARIA usage that apply during development. One involves static ARIA roles that don't require any scripting, the second involves ARIA roles that are basically static but still require some scripting to properly implement, and the third involves ARIA roles that cannot be used without comprehensive scripting in order to properly implement. Not having a proper understanding which roles belong to which category is a mistake that developers often make, which can cause interactive web technologies to become totally inaccessible to assistive technology users when the wrong roles are used in the wrong places or when the proper scripting is missing from roles that require it.

This is especially important when building markup templates as part of framework designing, and how this compares with reusable interactive component design, the two of which are not equal. The reason being that only static ARIA roles can be safely added to static markup templates without concern for scripting, yet all others such as pseudo and fully interactive ARIA roles cannot be safely added to templates without proper scripting to process these roles after content is rendered. These differences directly impact the job roles of those who are implementing

ARIA roles within web technologies. For example, it is safe for web designers with little to no knowledge of scripting to utilize the static ARIA roles within content templates without negatively impacting accessibility. However, if the same people attempt to do the same thing using interactive ARIA widget roles without strict cohesion with developers or a framework for synchronizing the requisite scripting, then they will break accessibility for assistive technology users instead.

The ARIA Role Matrices can be used to quickly identify these categories, including specific implementation guidance associated with each role: http://whatsock.com/training/matrices/

Static Roles

The following static ARIA roles can be included within standard HTML markup and require no scripting. These can be safely used by web designers with little to know knowledge of scripting.

- Landmark and Region Roles: application, banner, complementary, contentinfo, form, main, navigation, region, search
- Structural Roles: article, definition, directory + listitem, document, feed + article, figure, heading, img, list + listitem, math, note, separator, table + cell, term, toolbar
- Specialty Roles: none, presentation

Role names that include a + character followed by another role name refer to parent roles that require specific child roles.

Pseudo Interactive Roles

The following pseudo interactive ARIA roles are static roles that can be included within standard HTML markup, but require some accompanying scripting to ensure proper functionality. If used by web designers with little to no knowledge of scripting, then they must collaborate with knowledgeable developers so that the HTML markup and client side scripting correctly work together.

- Dialog Roles: alertdialog, dialog
- Live Region Roles: alert, log, marquee, status, timer

Interactive Widget Roles

The following interactive widget roles can be included within standard HTML markup, but require precise scripting to ensure proper functionality. Web designers with little to no knowledge of scripting cannot safely use any of these roles without full collaboration with knowledgeable developers to handle the required scripting, or without strict adherence to the rules for implementing frameworks to handle this required scripting automatically.

- Simple Interactive Widget Roles: button, checkbox, link, progressbar, searchbox, scrollbar, slider, spinbutton, switch, textbox, tooltip
- Complex Interactive Widget Roles: combobox, grid or treegrid + rowgroup + row + columnheader + rowheader + gridcell, listbox + option, menu or menubar + menuitem or menuitemcheckbox or menuitemradio, radiogroup + radio, tablist + tab + tabpanel, tree + treeitem + group

Tools and Resources

Visual ARIA can be used to visually identify the location of ARIA usage within web technologies and to assist in determining whether ARIA usage is correctly implemented.
http://whatsock.com/training/matrices/visual-aria.htm

An archive of open source provably accessible widgets is available for use within any web technology project, all of which map to the same standards documented in the ARIA Role Matrices and are visually confirmable using Visual ARIA.

- Powered by jQuery: https://github.com/accdc/tsg
- Powered by MooTools: https://github.com/accdc/tsg-mootools
- Powered by Dojo: https://github.com/accdc/tsg-dojo

The Accessibility Tree Training Guide provides a detailed training course for learning ARIA from beginning to end.
http://whatsock.com/training/