ACCESSIBILITY ON DEMAND

Getting Started with ARIA: Crash Course 101

Bryan Garaventa
9/17/16

ARIA stands for Accessible Rich Internet Applications, and learning how it works and how to use and recognize what it does will be a critical part of any development related position and task, especially since HTML5 now specifically maps to the ARIA spec for related control types. There are very few places now on the web that do not use ARIA to some extent.

To fully understand how to utilize ARIA, a good working knowledge of HTML, CSS, and JavaScript is important to have in advance.

First, to understand what ARIA is, you first need to understand what the Accessibility API is within platform operating systems and how this evolved into what it is now. The following article explains this, which was written by fellow members of the W3C:

https://www.smashingmagazine.com/2015/03/web-accessibility-with-accessibility-api/

Second, a full breakdown of how to properly use ARIA is important to understand, which will give you the necessary tools for recognizing when it is not being used properly. To accomplish this, the Accessibility Tree Training Guide must be read from beginning to end, which is literally a crash course in ARIA usage starting from no knowledge of what it is and ending with how assistive technologies use it to convey information. After you finish reading the training guide, you will understand the bones of ARIA and the basics for making it work, plus this will act as a reference guide for finding relevant details later on using the Table of Contents.

http://whatsock.com/training/

(It is important to read all of this from beginning to end, preferably while experimenting with the coding examples provided within the guide for practice.)

Since ARIA is an invisible technology, it can be extremely difficult to conceptualize, making the learning process more difficult than it needs to be. To assist with this, you can use the Visual ARIA Bookmarklet to physically see where ARIA is being applied on any public or local webpage, which maps to the ARIA 1.1 specification.

http://whatsock.com/training/matrices/visual-aria.htm

(Add the bookmarklet to your browser Favorites or Bookmark menu as instructed, then run it on any web page to display the ARIA usage there.)

Additionally, it is important to download the Visual ARIA CSUN GitHub project for a systematic introduction regarding Visual ARIA and how to use it to evaluate ARIA usage, which also includes a test development exercise for building accessible simulated radio buttons using ARIA. This is available at:

https://github.com/accdc/csun-2016

(View the ReadMe.txt file contained within the download for additional instructions.)

One of the features that Visual ARIA provides is the ability to hover the mouse over interactive widgets in order to display the computed Name and Description in accordance with the W3C Text Alternative Computation. To understand why this is important and how the algorithm works, read the blog post at:

http://www.ssbbartgroup.com/blog/how-the-w3c-text-alternative-computation-works/
(This will likely be confusing, but it is important to understand.)

By reading, rereading, and utilizing all of the resources here, you will gain a working knowledge of ARIA and how to use and recognize it, plus a working knowledge of tools that can be used to assist with this.

Download AccDC TSG Examples
https://github.com/accdc/tsg

To learn how ARIA works within complex widget examples, and to understand the underlying code that needs to be applied to them for proper functionality, use Visual ARIA on the live widgets within the above TSG download by opening the Coding Arena folder and loading the demo.htm files in your browser. (Some widgets may need a web server to load.)

To better understand how to interact with and test these widgets using the JAWS For Windows screen reader, perform the exercise documented in the Mozilla article “Testing web content with JAWS screen reader”.
http://website-archive.mozilla.org/www.mozilla.org/access/access/qa/win-webcontent-jaws.html

In addition, the following article explains how to use the NVDA screen reader in combination with Firefox to test web technologies more effectively:
https://www.marcozehe.de/articles/how-to-use-nvda-and-firefox-to-test-your-web-pages-for-accessibility/

After learning how ARIA works using the above resources, it should then be possible to accomplish the following when evaluating ARIA usage within web technologies:

1. Load Visual ARIA to see where ARIA is applied, and identify possible issues when interacting with simulated widgets using the keyboard.
2. Open the ARIA Role Matrices and expand the Table of Contents to locate the desired ARIA construct that you wish to learn more about and compare with the widget you are evaluating, which should match the required roles and supporting attributes documented there plus proper focus movement when applicable.
3. Compare the simulated client widget with a provably accessible widget that matches the same functionality within the AccDC TSG GitHub archive download to make sure that the desired functionality is supported by the assistive technology being used for testing.
4. Identify any discrepancies that may be impairing accessibility within the newly constructed web technology.

Related Articles:
"Printout: Using Visual ARIA to Physically See and Learn How ARIA Works"
"Important and valuable support changes to aria-owns "
"For Engineers: A Self-Paced Training Roadmap for Learning ARIA "

Contact Bryan:

Bryan Garaventa
Accessibility Fellow
SSB BART Group, Inc.
bryan.garaventa@ssbbartgroup.com
415.624.2709 (o)