Real World Web Accessibility With WAI-ARIA

Becky Gibson, IBM

Abstract – The Accessible Rich Internet Applications (WAI-ARIA) specification enables interactive Web applications to be fully accessible to keyboard and assistive technology users. WAI-ARIA adds the semantics necessary for the type and properties of rich user interface elements to be communicated to users of assistive technology. Implementation of keyboard interaction to mimic the desktop paradigm provides easier navigation through and interaction with the scripted components of the Web application. Using WAI-ARIA, updates made to regions of the page via Ajax can be communicated to assistive technology users. Implementation of WAI-ARIA into scripting libraries such as Dojo allows developers of all abilities to create fully accessible applications. No longer must users with disabilities struggle to use the Web or fear being left behind as the Web evolves - WAI-ARIA can enable Web access for all.

I. Introduction

The sophisticated applications enabled by Web 2.0 and Ajax are just hype if they can't be used and accessed by the entire population, including people with disabilities. Those using assistive technologies to interact with the Web need to know the semantics associated with the rich interface controls embedded in Web 2.0 applications. Enter WAI-ARIA – Accessible Rich Internet Applications specification [1] from the Web Accessibility Initiative (WAI) of the W3C. ARIA is currently in last call status and is expected to reach W3C candidate recommendation by the end of 2009.

IBM and many other companies including AOL, Yahoo, Google, Mozilla, Microsoft are adopting WAI-ARIA. It has been implemented into the Dojo Open Source JavaScript Toolkit [2] [3] core widget set and is being implemented in other JavaScript toolkits as well.

II. Methodology

A standard component such as an anchor or input element has a clearly defined role and behavior that the browser can communicate to an assistive technology such as a screen reader using an operating system accessibility application programming Interface (API). Rich interface controls built using JavaScript do not have any semantic information built into them. WAI-ARIA provides the additional metadata to identify the roles, states and properties of these rich interface controls to the browser so it can be communicated to the assistive technologies [4]. For example, with the addition of ARIA roles and properties to a scripted tree control, a screen reader user can hear that focus is on a tree item, what the text of that tree item is, what level that item is within the tree and whether or not the tree item has children and if it is expanded or collapsed [5]. Compare this to just hearing that focus is on a list of items with no knowledge that the list is organized into a tree hierarchy! Instead of tabbing to each of the tree items, adding full keyboard support to mimic the desktop paradigm makes the rich user interfaces more usable for all. A WAI subgroup has created a DHTML Style Guide [6] to define the keyboard behavior of common Web user interface controls.

In addition to richer user interfaces, Web 2.0 is the age of incremental updates via Ajax (Asynchronous JavaScript and XML). Without WAI-ARIA, an assistive technology user may not be aware that parts of the page are being updated. With the use of WAI-ARIA live region roles and properties the updates can be announced and spoken by the screen reader. The author can control the extent of the change announced – either the entire region or just the data that has changed within the marked region. ARIA landmark roles allow regions of the page such as navigation, search, main content and banners to be identified and easily navigated to via the keyboard.

The realization of WAI-ARIA has been a collaboration of effort between the specification process and the implementation within the browsers, assistive technologies and Web developers. The browser and screen reader communicate via the operating system accessibility API. The open source Firefox browser was the first to support WAI-ARIA. Firefox interprets the additional WAI-ARIA roles and properties from the JavaScript Document Object Model (DOM) and, on the Windows platform, communicates that information to the assistive technology via the Microsoft Accessibility API (MSAA). The JAWS 10 screen reader from Freedom Scientific currently provides the best support for WAI-ARIA. Internet Explorer 8 supports WAI-ARIA and Opera and Safari are working on support as well. WAI-ARIA has entered the mainstream and is making it possible to create fully accessible Web 2.0 applications.

III. Results

As the first toolkit to implement WAI-ARIA, the Dojo Open Source JavaScript toolkit provides a fully accessible rich user interface widget set. Using Dojo, Web developers can create full featured, dynamic Web 2.0 applications that are accessible to users of all

B. Gibson is a Senior Technical Staff Member in Emerging Technologies at IBM, Westford, MA.

abilities. In addition, during the development process the Dojo accessibility team provided valuable feedback into the development of the WAI-ARIA specification and to the browser and assistive technology vendors. Implementing a full featured, real world solution via Dojo was vital to moving WAI-ARIA from theoretical specification to a practical, implementable solution.

IV. Conclusion

With support in the mainstream browsers, popular assistive technologies and Web 2.0 JavaScript toolkits, WAI-ARIA is making an impact in the accessibility of rich Web applications. Adoption by companies such as IBM, Google, Yahoo, Microsoft, AOL and others are helping to increase the awareness and adoption rate of ARIA implementations and techniques. It is important to communicate the capabilities of WAI-ARIA to the Web and accessibility communities to increase the usage of WAI-ARIA and to enable development of a truly accessible Web for all.

References

[1] Accessible Rich Internet Applications (WAI-ARIA) Version 1.0. J. Craig, M. Cooper, L. Pappas, R. Schwerdtfeger, L. Seeman, Editors, W3C Working Draft (work in progress), 24 February 2009. This version of WAI-ARIA is available at http://www.w3.org/TR/wai-aria/.

[2] Dojo Open Source JavaScript Toolkit - http://dojotoolkit.org/.

[3] B Gibson, Dojo: An Accessible JavaScript Toolkit, 14 August, 2007, IBM Human Ability & Accessibility Center, http://www-03.ibm.com/able/resources/dojo.html - ibm-content.

[4] B Gibson, R Schwerdtfeger, DHTML Accessibility – Solving the JavasScript Accessibility Problem, conference proceedings The Seventh International ACM SIGACCESS Conference on Computers and Accessibility, 9-12 October, 2005;202-203.

[5] B Gibson, recorded demonstration, 1 May 2009 http://www.weba11y.com/Examples/DojoMailwithJaws10/DojoMailwithJaws10.html.

[6] DHTML Style Guide, 12 December, 2007, AOL Developer Network, http://dev.aol.com/dhtml_style_guide.