

Backbase Accessibility

Whitepaper

Learn about:

- Section 508
- Accessibility requirements
- Backbase compliance

Introduction

This paper discusses the growing importance of Rich Internet Applications (RIA's) and their support for Accessibility. It reviews the Accessibility requirements as set by the World Wide Web Consortium (W3C) through their Web Content Accessibility Guidelines (WCAG) and Section 508 of the U.S. Rehabilitation Act. Because these guidelines are not easily applicable to Rich Internet applications, this paper also details how the Backbase Enterprise Ajax engine and development environment support accessibility – helping developers to build new accessible AJAX web applications (ARIA's).

New standards for accessibility

When different entities try to address a single issue, it's essential to establish standards. In the case of accessibility, standards help designers and developers of Web content address accessibility issues in a common and organized way.

In the first significant effort to establish a common accessibility standard, the World Wide Web Consortium (W3C) established The Web Content Accessibility Guidelines (WCAG). This standard includes 14 guidelines, each with three checkpoint levels for web developers to meet: Priority One, Priority Two, and Priority Three.

Since then, many countries have introduced their own national standards. In the US, Section 508 of the U.S. Rehabilitation Act is based on WCAG Priority One checkpoints. Australia, France, Germany, and many other countries use the same checkpoints. On the other hand, the Common Look and Feel standard in Canada and Guidelines for U.K. Government Web Sites in the United Kingdom are based on Priorities One and Two of the WCAG.

The importance of accessibility

The law is the ultimate motivation to support accessibility, and it's required when dealing with government agencies. With new national requirements in the United States, Canada, and the European Union, and more to come in the near future, there are numerous legal mandates for accessibility. These policies will likely expand in scope.

Companies delivering early-on products that are accessibility compliant will gain a significant competitive advantage.

Another driver for compliance is market demand. As an ancillary benefit, accessibility guidelines encourage good programming practices separating layout from content through the use of cascading style sheets (CSS). This separation lets companies disclose information much faster online, enables shorter development cycles, and simplifies exchange with other Web services (mashups).

Section 508

All U.S. federal agencies were mandated under Section 508 of the U.S. Rehabilitation Act to make their websites accessible by June 21, 2001.

U.S. law and international standards

Most countries, including the European Union, have adopted standards based on the Web Content Accessibility Guidelines (WCAG) from the W3C. In the United States, Section 508 of the U.S. Rehabilitation Act is based on W3C Priority One checkpoints. It mandates that Web content maintained by the federal government must be made accessible to people with disabilities.

Backbase accessibility often refers to Section 508. Section 508 issues are, for the most part, similar to W3C Priority One issues.

Web-based intranet and Internet information and applications (Section 508 §1194.22)

The criteria for Web-based intranet and Internet information and applications are based on guidelines established by the Web Accessibility Initiative of the W3C. Many provisions are meant for people with visual impairments who rely on assistive products like screen readers.

This section promotes the use of text labels and descriptors for graphics and certain elements. It also addresses the usability of style sheets, scripting languages, electronic forms, etc.

§1194.22 is consistent with the priority 1 Checkpoints of the Web Content Accessibility Guidelines 1.0 (WCAG 1.0) (May 5, 1999) published by the Web Accessibility Initiative of the World Wide Web Consortium.

Accessible Rich Internet Applications (ARIA)

The challenge with current legislation and accessibility initiatives is that they are designed for traditional web pages. In effect, HTML was not designed for the highly interactive user Interfaces offered by AJAX – Javascript and XML enable partial page refreshing, dynamic showing and hiding of sections in a page, etc. AJAX propelled web development very quickly towards Graphic User Interfaces, as we know in the desktop environment. However, unlike desktop applications, the HTML realm has no accessibility APIs that let assistive tools ‘understand’ the application. Hence, the paradox is that a richer Graphical User Interface enabled by AJAX reduces accessibility compliancy when measured by the older legislation and HTML shortcomings.

XML (the “x” in AJAX) does have the capacity to offer semantic information specifically for accessibility purposes. However, today’s AJAX frameworks don’t exploit this.

Backbase provides the capabilities for achieving the same level of accessibility as HTML, and to even exceed it.

ARIA based on Backbase

For Backbase, accessibility compliance has always been foremost in our design and development efforts. The Backbase engine enables uniform XML support across all browsers. Backbase continues to lead the Ajax space in its ongoing active support for the W3C standards, including the WCAG Accessibility standard. We think this will emerge as the standard solution for proficient accessibility support and likely most data communication as well.

Below, you will see three areas where Backbase leverages accessible application development: framework design, widgets, and real world testing.

Accessibility by framework design

Cascading Style Sheets driven

Cascading style sheets offer significant benefits for accessibility. While the use of CSS is not mandated by Section 508, national standards in the United Kingdom and Canada do require it. When HTML is structured semantically and CSS is applied to mark up text, screen readers can pick up important headers, and visually impaired users can override styles to format text to meet their needs. CSS can

also be used for the layout of the whole web page. This offers some major advantages over HTML (i.e. table based) layout. Developers can structure pages so screen readers process the most important information first – the menu, or the main content – even when it is graphically positioned at the bottom of the page. Backbase’s design patterns, widgets markup and layout are fully CSS driven.

Declarative language for easy interpretation by assistive technologies

Backbase Enterprise Ajax uniquely includes a double API (XML and JavaScript). The declarative language approach from the XML API is superior to ‘loose ends’ JavaScript. The advantage of this programming approach is that predefined events and event handlers are available. The semantics of these declarative tags will assist accessibility. The W3C identifies declarative languages as one of the key methods for mitigating HTML shortcomings in Web applications.

[Accessibility in widgets](#)

Progressive HTML enhancement

Backbase supports a programming technique called “progressive HTML enhancement.” Here, the widget has an HTML object as a starting situation, and the Backbase engine reads the data from this object. Next, the AJAX widget is ‘laid over’ this HTML widget and thereby provides all richness.

There are two advantages to this technique. First, content is accessible to both assistive technologies and Web crawlers like Google. Second, the widgets have ‘graceful degradation,’ meaning it is still readable for users after turning off CSS or JavaScript. A good example is the list grid as can be found in the BDN community. With Backbase enabled, live editing and keyboard navigation of table cells are available. When Backbase is disabled, a traditional table is displayed.

Keyboard navigation

Many Enterprise Ajax widgets are keyboard navigable by default. Backbase has invested significant effort in advancing our focus model to its current level. Elements can almost always get focus using the keyboard – for example cells in a table grid. For visually- and motor-disabled users this is a very important feature. Tabindex is used for navigating things like links and forms.

Implementing the focus model is fast and straightforward. Event handling and bubbling implementations conform to W3C standards and are fully unified across all supported browsers.

Access key

In addition to HTML elements like input or textfield, Backbase supports key accessibility for all Backbase widgets in the BTL namespace, including widgets like b:slider and b:calendar.

Font scaling

Many Enterprise Ajax widgets support font scaling. Users can scale fonts both up and down several steps without breaking the layout or functionality.

Skin colors

The default CSS skins available for the widgets are designed so that colorblind users can discern the different interaction states. When hovering, clicking or selecting, both the hue and the brightness of colors of the objects are used to help colorblind people differentiate interaction states.

[Accessibility by real world testing](#)

Accessibility form demo

Backbase’s accessible forms demo is available at Backbase.com. The demo shows how to build client

side validation in a Web form accessible by screen readers. It also illustrates graceful degradation when CSS and JavaScript are switched off. The demo can be used as a design pattern for Backbase forms.

Accessibility testing

Backbase's User Experience Center and R&D team test the main widgets with people who are visually impaired. Feedback gathered from this real world testing drives further software improvements.

Consulting screen reader developers

Backbase consults Freedom Scientific (developers of Jaws, a common used screen reader for visually impaired people) to test the accessibility of widgets. Currently, Jaws can pick up link lists and header lists in Backbase applications and additional uses are planned.

Compliance testing

Backbase widgets are regularly tested for compliance with section 508. Widgets and the Backbase framework are scored for their degree of accessibility using the articles specified in the rehabilitation act. For an up-to-date list of accessibility widgets compliance, check Backbase.com.

Backbase and accessibility: what to expect

Backbase ranks accessibility as very important for both customers and end users. However, we recognize that standards for RIA accessibility are not finalized and there are opportunities for even more comprehensive support. Several additional accessibility topics are currently on our roadmap, both for developers and end users.

Future improvement for developers

Design patterns

Our technical consultants will be publishing accessibility articles on our new developer's network. These articles explain our best practices and design patterns based on Backbase's application development experience.

In-browser accessibility validation scripts

We will soon offer exchangeable user installed validation scripts for our in-browser Debugger tool. After downloading the 'accessibility script', developers can check the built-in 'logger' for accessibility errors and warnings.' for example, the blogger can warn that a button does not have a text equivalent. These validation methods will be published through the Developer's network, allowing the community to provide input.

Future improvement for real world accessibility

Support for W3C WAI-ARIA namespaces

On short term, we will support the standards and drafts created by the Web Accessibility Initiative for RIA accessibility. These specifications are still in draft, but are an important source for accessibility in Accessible Rich Internet Applications. Support entails the ARIA-ROLE and ARIA-STATE. W3C has proposed special namespaces for ROLE and STATE. These namespaces can be used by assistive technologies to better understand the application. Support for the Resource Description Framework (RDF) will further enable the understanding of data.

Semantic data in the Backbase abstraction layer

MVC paradigm implementation is central in the Backbase 4 framework. This client-server design pattern is a very powerful concept for separating the model from the view and the controller. This

abstraction layer can be used by assistive technologies like screen readers, allowing the addition of extensive semantic information to describe the application state and role. In combination with the W3C WAI-ARIA conventions and RDF support, this abstraction layer can be used to define a true accessibility API similar to desktop GUIs.

Conclusion

Backbase leads in W3C support for Ajax and is fully committed to support the new W3C standards as they become available, including the WCAG Accessibility standard. Backbase is also committed to the continued support of Section 508 of the U.S. Rehabilitation Act.

Equally important, Backbase aims to set the tone for accessibility in RIAs, in terms of framework, design patterns and widgets. And this is not just a statement. Our software is designed with accessibility built-in from the ground up. Our vision is that XML standards will be the foremost enabler of cooperation across devices, platforms, and ultimately context.

Contact Backbase, The Ajax Company:

Americas:

635 Mariners Island Blvd Suite 200
San Mateo, CA 94404
USA

Tel: +1 866 800 8996
Email: sales-us@backbase.com

Europe and Asia:

Stephensonstraat 19
1097 BA Amsterdam
The Netherlands

Tel: +31 20 465 8888
Email: sales-eu@backbase.com