

Blaise IS and Accessibility

Jim O'Reilly
Westat

FedCasic Conference
March 16, 2005

Agenda

- Background
- Blaise® for Windows accessibility
- Blaise Internet Services accessibility - greater challenges and relevance
- Blaise IS architecture
- Accessibility testing and actions

Background

- Statistics Netherlands, Blaise team and Westat
 - Committed to supporting accessibility and U.S. ‘508’ standards
- Blaise and Microsoft Windows®
 - Formerly sole development and application environment
 - Accessibility support based on
 - Windows standards and support of accessibility and universal design
 - Blaise system’s rigorous implementation of Windows standards
 - Westat’s statement on Blaise and accessibility at www.westat.com/blaise/508info/blaise508stmt.htm
 - Responding to reported issues

Blaise for Windows and Screen Readers

In 2003 customer reported

- Blind application software tester could not make Blaise applications in the Windows environment run with JAWS screen reader.

Investigated and learned

- Custom-developed scripts for JAWS are common to cue screen reader to idiosyncracies of an application's user interface
- Consultants experienced in JAWS scripting available through www.blindprogramming.com

Westat sponsored development of a JAWS script for the Blaise Data Entry Program (DEP) UI

- Initial script developed in about 12-15 hours
- Testing by blind user found script worked very well
- Script available to others (contact blaise@westat.com)

Blaise IS (Internet Services)

New challenges

- Browser environment diverse and controlled by user
- Users are often the general public not survey staff
- Richer presentation features and user-interface elements

Accessibility testing and evaluation must be

- Pro-active and continuing
- At the application as well as systems levels

IS architecture - a higher level approach

Core technology is ASP, XML, XSL

- High level approach, little or no direct HTML coding
- Developer builds datamodel in same way as Win CAPI or CATI
 - Web look-and-feel set with LAYOUT statements and the mode library UI parameters
 - Mixed-mode CATI-CAWI can be authored in one code set
- Web survey characteristics set in specification file
 - Start and end ASP pages, XSL stylesheet to generate HTML pages, and others
- On server IS receives response, runs datamodel rules, generates next page's information in XML, & XSL generates the HTML page
- XSL stylesheet key to look-and-feel and HTML code
 - Single location for implementing accessibility enhancements

Accessibility testing of Blaise IS Surveys

IS surveys may have hundreds of page. How to test that 508 standards are met?

- Free online tools require submitting individual pages
- Needed a high volume process
- AccVerify (www.hisoftware.com) able to evaluate multiple pages in batch mode
- Capturing Blaise IS pages
 - Adapted IS ASP page handler process (biPagHan.asp) to save to a file every page generated in a test interview
- Passed generated pages into AccVerify
- Reviewed reports

AccVerify Testing

- Reports used to identified “failed” pages
 - Mainly graphical elements lacking a text attribute “Alt=”

A. 508 Standards, Section 1194.22, (a) A text equivalent for every non-text element shall be provided (e.g., via "alt", "longdesc", or in element content).

Rule: 1.1.2 - All INPUT elements are required to contain the alt attribute or use a LABEL.

Failure - INPUT Element, of Type RADIO, found at Line: 620, Column: 18

Failure - INPUT Element, of Type RADIO, found at Line: 661, Column: 18

Failure - INPUT Element, of Type RADIO, found at Line: 702, Column: 18

- Issues report to Blaise team and quickly fixed
 - Changes made to interview page XSL stylesheet
 - Apply directly to subsequent IS surveys

Screen Readers

- Automated testing of HTML code is useful but limited
- Screen readers handle things AccVerify fails
- Conversely, readers can be stumped by survey elements that meet the specific 508 standards
- Readers ability to interpret page correctly is acid test
- Testing with reader can be difficult for inexperienced
 - Readers rely heavily on key-combinations entered by user to navigate about the page
 - JAWS has ~66 of them

Screen Reader Testing

Two tests with JAWS of Blaise IS samples surveys

1. Westat user with normal vision and a few hours experience with JAWS
2. An accessibility consultant who is blind and experienced with JAWS and other screen readers

Test 1 (sighted Blaise developer)

- Reads header but not footer (Prev, Next buttons)
- Items with preface text—not reading preface
- At times not reading enumeration sets past first empty selection
- Tables—column labels not read
- Items with icons to left (Help & Comment buttons) are not read properly

Test 2: (blind consultant)

- Overall 70-80% of full accessibility
- Need explicit titling of columns with <TH> tag
- Form elements need a <Label for="idname">
- Recommends alt-key combinations for key navigational elements, <Prev>, <Next> etc.
 - Not 508 requirement but handy for users

Actions

- Pursuing specifications based on tests to address all identified issues
- Plan on integrating other guidelines
 - E.g. BLS draft guidelines
- Revise interview XSL stylesheet to make IS surveys accessible to 9x%
- Continue evaluation and testing