

Optional Part 4 with a Focus on Development

Additional/Alternate Part 4:

- Goal: To investigate developer- and business-orientated solutions (including cost/benefits)
- Reason: To answer the question: “That’s nice, but how do I *do* it?”
- Could also be a supplement to Part 3 for what to look for

Microsoft Active Accessibility (MSAA):

<http://msdn.microsoft.com/library/default.asp?url=/library/en-us/vsent7/html/vxconactiveaccessibility.asp>

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dv_foxhelp/html/foconaccessibilitybrowser.asp

<http://www.mozilla.org/projects/ui/accessibility/msaa-server-impl.html#quirks>
(Mozilla.org’s comments on the quirks for developer usability of MSAA)

GNOME and GTK+ accessibility:

<http://developer.gnome.org/projects/gap/>

<http://developer.gnome.org/projects/gap/testing/index.html> (Testing guidelines)

Developing end-user accessibility tools for Gecko applications:

<http://www.mozilla.org/projects/ui/accessibility/vendors-win.html>

How Mozilla.org implements MSAA:

<http://www.mozilla.org/projects/ui/accessibility/msaa-server-impl.html>

Mozilla.org 508 Compatibility (including their explanation why it is concerned with it):

<http://www.mozilla.org/projects/ui/accessibility/section508.html>

Macromedia guidelines for creating accessible Flash:

<http://www.macromedia.com/macromedia/accessibility/features/flash/>

Based on information read, write up a simple report stating their opinions of reasonability for their own usage of the MSAA or other developer-offered toolkits.

Discussions could include the feasibility of a cross-platform and cross-language (with a similar API where feasible) accessibility library for software developers, cross-platform development issues (ATK is available for FreeBSD, Linux, and Windows [OS X support through X11 support?], and is toolkit-independent, should this be preferred over MSAA? However, existing screen-reading software is most effective with MSAA.)

Taking into account the extra development time and costs, how should it be handled in a production environment? Should accessibility items be added in a later release of the software, or always be concurrent? Should it be “mostly” functional, with a few features not available?