OpenOffice.org Conference 2006

2006-09-13, Lyon, France

OpenOffice.org & ODF Accessibility

Malte Timmermann
Technical Architect
StarOffice/OpenOffice.org
Sun Microsystems

About the Speaker

- Technical Architect in StarOffice/OpenOffice.org Development
- Working on StarOffice since 1991
- Main focus: Core Development
  - Accessibility
  - Security
  - Architecture and Performance
  - Past: EditEngine, VCL, Help System, BasicIDE, ...
- http://blogs.sun.com/malte

Agenda

- Accessibility Features
- Assistive Technology
- Accessible Document Creation
- OASIS OpenDocument Accessibility SC
- Q&A

Accessibility

- People with disabilities should be able to work with OpenOffice.org
- Access for people with disabilities must be comparable like for people without disabilities

Accessibility Features

- Keyboard Navigation
- Colors and Fonts
- Assistive Technology
- Documentation
- Additional OOo Accessibility Features

Keyboard Navigation

- All features must be usable via the keyboard!
- Some users cannot use a mouse
  - > If they can't access every control and function in your application without one, productivity may suffer, or the application might be simply unusable.
- Some users cannot see the screen
  - > But they still have a good mental picture of what's there, so require consistent, reliable keyboard navigation to get around.

Keyboard Navigation

- Most of us find it faster to use the keyboard for some things
  - But when it doesn't work as expected, we usually end up taking longer than if we'd just used the mouse, even if we don't realize it!
- Accessibility improves the user experience for everyone, not just for those with special needs!

Colors & Fonts

- Adapt to the color schemes and contrast display settings of the operating system
- Offers additional settings for colors which are not specified in any system color schemes
- Support for High Contrast and Large Fonts
  - > Document content might be rendered differently
- OpenOffice.org User Interface uses the default system font, but offers to choose a different one

Documentation

- Documentation accessible, available in HTML and PDF format
- The help system is accessible
- Templates and sample documents follow rules for accessible document creation

Other Features

- Read-Only cursor
  - Makes it much easier to navigate in read only documents
  - Text selection with keyboard is possible
- Theming for help viewer
- Animations for images and text can be switched off
- “Automatic Font Color” as the default for new documents
  - To make sure text is readable regardless of the color theming

Assistive Technology

- Screen Magnifiers
- Screen Readers
- Voice Control
- Onscreen keyboards
- Keyboard/mouse enhancement utilities
- Alternate input/output devices
- Eye-gaze, simple switches, Braille devices

Evolution of Assistive Technology

- History
  - First generation - access to TTY systems
    - Access to content was quite ease (video buffer)
  - Second generation - access to the GUI
    - Access very difficult
      - Accessing Windows Screen was not possible for a long time!
    - Patching of OS and video drivers was needed
    - A lot of guessing from AT tools
  - Third generation - access via API
    - Well defined API, no guessing from AT
    - AT can be quite application independent
      - Special knowledge about application only for efficiency and customizations

Evolution of Assistive Technology

- Accessibility API
  - AT shouldn't do guess or patch the system
  - Sun defined the Java Accessibility API
    - Because Java didn't use standard system controls
  - GNOME Accessibility Architecture
    - Derived from Java Accessibility API, to have built in accessibility framework in GNOME.
    - Other are following this approach: Mac OS X, Windows Vista
  - OpenOffice.org Accessibility API
    - Derived from Java and from GNOME Accessibility, to have it on different platforms

OpenOffice.org Accessibility API

- Derived from Java and GNOME Accessibility
  - To support multiple platforms
- Used for UI and for document representation
  - Enhancements for exposing the document content
  - Led also to improvements in Java and GNOME APIs
  - OpenOffice.org is probably the first application that exposes the document content via Accessibility API
- Bridged to Java Accessibility and UNO-ATK
  - AT doesn't have to know anything about OpenOffice.org or about that new Accessibility API

AT Tools working with OpenOffice.org

- Windows
  - AT that supports Java Accessibility API
    - JAWS Screen Reader
    - ZoomText Screen Reader and Magnifier
    - Supernova Screen Reader
- GNOME (Linux/Solaris)
  - AT that supports the GNOME Accessibility Architecture
    - Gnopernicus Screen Reader and Magnifier
    - GNOME On Screen Keyboard (GOK)
    - Orca Screen Reader and Magnifier
    - Dasher
    - All above is open source!

Accessible Document Creation

- Accessibility features alone are not enough
- Authors of documents must make sure to help for good accessibility!

Accessible Document Creation

- Some simple rules
  - Create well structured content
    - Use named styles (“Heading 1”, ...) instead of font changes
    - Mark table column and row headers
    - For forms, use logical tab order
  - Don't rely on color alone
    - Ensure that information conveyed with colors is also available without color
  - Use luminosity contrast ratio
  - Provide text alternatives for all non-text content
    - Each image and each object should have a (unique) name and a description
    - Provide names and descriptions for hyper links

Accessible Document Creation

- Make sure to keep you documents as accessible as possible
  - > When exporting documents to an other portable format, use HTML or PDF
    - > There is more AT support for browsers than for PDF viewers
    - > Latest versions of Adobe Reader are better accessible now, including Self-Voicing on Windows and Linux
  - > When using OpenOffice.org to create PDF files, make sure to create “Tagged PDF”
  - > When using Adobe Acrobat for this, also make sure to not lock out AT by setting security restrictions

Accessible Document Creation

- W3C Web Accessibility Initiative
  - Web Content Accessibility Guidelines
    - Version 1 concentrated on HTML
    - Version 2 (working draft) will be more general
- OASIS OpenDocument TC Accessibility Subcommittee
  - Determine whether any accessibility support is missing in the file format, and if so add what is missing
  - Guidelines for ODF applications

OASIS OpenDocument Accessibility SC

- Statement of purpose
  - Ongoing review of the OpenDocument specification for accessibility, both to discover potential accessibility issues and to improve the usability and functionality of creating, reading, and editing office documents for people with disabilities.
  - To provide accessibility related feedback to the OpenDocument Technical Committee and implementors of the OpenDocument specification.
- Deliverables
  - Proposals for accessibility enhancements to the OpenDocument specification.
  - Accessibility related guidelines for OpenDocument implementors and users.

OASIS OpenDocument Accessibility SC

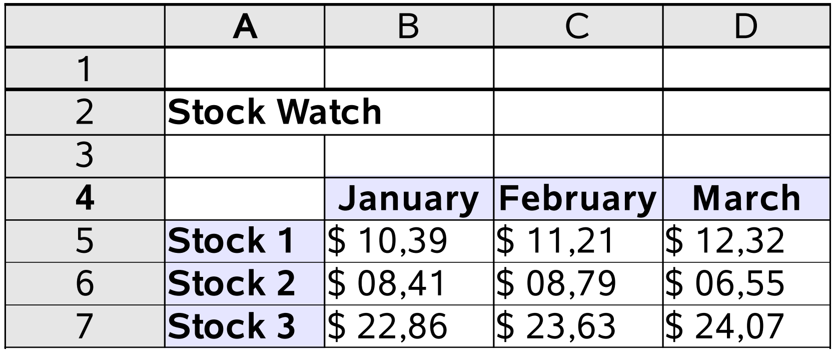
- Started in January 2006
  - Identified 9 accessibility issues in ODF 1.0
    - This is really not much, considering a 700 pages specification!
  - Proposed changes for Accessibility to OASIS OpenDocument Format in June 2006
  - Most of them to be integrated into ODF 1.1 specification
- With proposed changes, we believe that ODF will meet or exceed the accessibility support provided in all other office file formats.
- Continuing the work on further improvements and on guidelines for ODF implementors

Accessibility Changes for ODF 1.1

- #1: Encoding of pagination information
  - When a document is paginated, the soft page break elements should be exported
A new attribute is introduced, which specifies whether this document contains soft page breaks.
  - This enables direct ODF to DAISY book conversion, with same pagination like when the document is rendered in an ODF application
  - Applications loading such documents should use the soft page breaks for pagination

Accessibility Changes for ODF 1.1

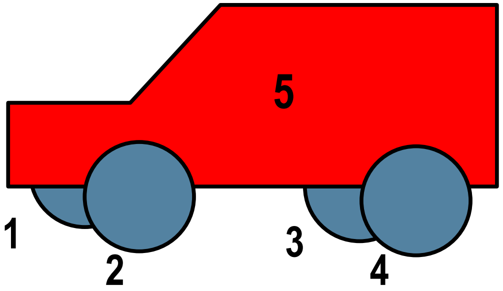
- #2: Proper encoding of authored table header content
  - Authors of documents must be able to specify header row and column.
  - Using styles for this is not enough, it must be indicated with <table:table-header-rows> and <table:table-header-columns>
  - Example:
    - Header row is *4*
    - Header column is *A*



	A	B	C	D
1				
2	**Stock Watch**			
3				
4		**January**	**February**	**March**
5	**Stock 1**	$ 10,39	$ 11,21	$ 12,32
6	**Stock 2**	$ 08,41	$ 08,79	$ 06,55
7	**Stock 3**	$ 22,86	$ 23,63	$ 24,07

Accessibility Changes for ODF 1.1

- #3: Author-defined logical navigation of page objects in presentations
  - Authors should be able to indicate a logical navigation order for traversing through objects in ODF presentations as distinct from z-order
  - The inability to specify a logical navigation order makes it difficult for some users to properly understand a drawing or presentation
  - Example:
    - First toggle through tires, before reaching the car

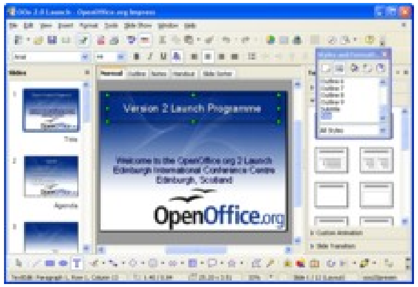


Accessibility Changes for ODF 1.1

- #4-6: Allow alternative text for image maps, drawing layers and drawing objects
  - All these objects should (optionally) have an (accessible) name (<svg:title>) and description (<svg:desc>)
  - This information should also be used when converting from/to HTML or other formats
  - Example:
    - svg:title: Smiling face
    - svg:desc: A circle with 2 small circles as eyes and a curve as the mouth

Accessibility Changes for ODF 1.1

- #7: Establish clear relationships between objects and their captions
  - Establish clear relationship between a drawing objects and its caption by including a new optional draw:describedby attribute
  - Provide this information via Accessibilty API to Assistive Technology
  - Example:
    - Caption text for a picture



Version 2 Launch Programme

Welcome to the OpenOffice.org 2 Launch
Edinburgh International Conference Centre
Edinburgh, Scotland

Screenshot of OpenOffice.org
Presentation module

Accessibility Changes for ODF 1.1

- #8: Establish text hints for hypertext links
  - Hypertext links do not provide hints as to the destination of a link.
  - <svg:title> element must be provided as an optional element to <text:a>
  -
  - Example:

xlink:href:	http://openoffice.org
office:name:	OpenOffice.org Web Site
office:title:	The home of OpenOffice.org

Accessibility Changes for ODF 1.x

- #9: Provide for structured tables in presentations
  - Users importing non-ODF slides that contain tables need access to the table structure via their assistive technology.
  - Native table support with access to full semantic structure will be addressed in a future release of the ODF specification.
  - Meanwhile, tables imported into ODF from another file format must have their structure preserved.
    - Embedded spreadsheets as an interim solution

Q/A

References:

OOo Accessibility	http://openoffice.org/access
OASIS ODF Accessibility SC	http://www.oasis-open.org
W3C Web Accessibility	http://www.w3.org/WAI

OpenOffice.org & ODF Accessibility

Malte Timmermann

Malte.Timmermann@sun.com

http://blogs.sun.com/malte

[DEV] Keyboard Navigation

- What programmers can do
  - Label every control and menu item with a context-unique mnemonic shortcut
  - Use a logical tab order in dialogs
  - Provide keyboard equivalents for every function that can be performed with the mouse
  - Unplug the mouse and check if you can use your features without the mouse, this can lead to better design/usability
  - Ensure custom controls follow the same keyboard navigation rules as standard controls
  - Don't break standard navigation key bindings
  - Ensure custom controls have a focused state that is always visually discernible

[DEV] Assistive Technology

- What programmers can do
  - > Expose information for UI and document content via Accessibility API
  - > Use standard UI elements
  - > Support schemes so that the user can use a GUI that better supports his needs
  - > Support Speech input – Commands and Dictation