odt2braille Developer Guide

for version 0.0.2, 30 August 2010

by Bert Frees

Copyright © 2010 by DocArch http://www.docarch.be.

This program is free software: you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU Lesser General Public License for more details.

You should have received a copy of the GNU Lesser General Public License along with this program. If not, see http://www.gnu.org/licenses.

Table of Contents

Introduction

1 Hacking

System Requirements

In order to hack odt2braille, you need the following software:

OpenOffice.org

The latest version of the OpenOffice.org office suite can be downloaded here.

OpenOffice.org Software Development Kit

The OpenOffice.org SDK is an add-on for OpenOffice.org. It provides the necessary tools and documentation for programming the OpenOffice.org APIs and creating own extensions. The latest version of OpenOffice.org SDK can be downloaded here. Make sure you install OpenOffice.org first. The version of OpenOffice.org SDK should be equal or lower than the version of OpenOffice.org.

NetBeans

Odt2braille was developed in NetBeans, an integrated development environment. The latest version of NetBeans can be downloaded here. Make sure the Java SE pack is included.

OpenOffice.org API Plugin for NetBeans

To install the OOo API Plugin for NetBeans, launch NetBeans, go to [*Tools* > *Plugins*], then select *OpenOffice.org API Plugin* in *Available Plugins* and click on *Install*. Now relaunch NetBeans and configure the plugin in [*Tools* > *Options* > *Miscellaneous* > *OOo API Plugin*] by selecting the appropriate OpenOffice.org installation and OpenOffice.org SDK folder.

[@TODO] NetBeans with OpenOffice.org API plugin may be replaced by Eclipse with OOEclipse plugin.

Odt2braille Source Code

The source code is available on Sourceforge and can be downloaded directly into NetBeans as follows:

- [*Team - Subversion - Checkout*]
- *Repository URL*: "https://odt2braille.svn.sourceforge.net/svnroot/odt2braille". Leave "*User*" and "*Password*" fields blank.
- *Next* >
- *Repository Folders*: "DaisyPipeline, Odt2Braille, Odt2BrailleAddOn"
- Check "*Scan for NetBeans Projects after checkout*".
- *Finish*

The source code can also be downloaded without NetBeans:

- `svn co https://odt2braille.svn.sourceforge.net/svnroot/odt2braille path/to/odt2braille/folder/`

Build

Build projects with NetBeans:

- Open "*Projects*" window, than right click a project and choose menu item "*Build*".

Build projects with Apache ANT:

- `cd` to project directory
- `ant`

Create .oxt file with NetBeans:

- Open "*Projects*" window, than right click "*Odt2BrailleAddOn*" and choose menu item "*Create OXT*" or "*Deploy and Run Extension in OpenOffice.org*".
- The .oxt file will be created in Odt2BrailleAddOn/dist.

Create .oxt file with Apache ANT:

- `cd` to Odt2BrailleAddOn directory
- `ant uno-package`
- The .oxt file will be created in Odt2BrailleAddOn/dist.

Documentation

Texinfo:

- `cd` to Odt2BrailleAddOn/doc
-
```
texi2dvi --pdf
odt2braille-user-doc.texi
```
-
```
makeinfo --html
[--no-split]
[--no-headers]
--output=odt2braille-user-doc.html
odt2braille-user-doc.texi
```

Javadoc:

- `cd` to directory containing all projects
-
```
javadoc [-private]
[-author]
[-version]
[-breakiterator]
-d Odt2BrailleAddOn\doc\javadoc
-subpackages be.docarch:org_pef_text
-sourcepath Odt2BrailleAddOn\src;Odt2Braille\src;DaisyPipeline\src
```

Downloads

- Apache Subversion: `http://subversion.apache.org/`
- Apache Ant (1.7.1 or higher): `http://ant.apache.org/`
- Javadoc: `http://java.sun.com/j2se/javadoc/downloads/index.html#findjavadoc`
- Texinfo: `http://www.gnu.org/software/texinfo/`

2 Projects

This is an overview of the projects that are created in NetBeans after odt2braille has been succesfully checked out.

Odt2BrailleAddOn

Odt2BrailleAddOn is the OpenOffice.org extension. It makes extensive use of the OpenOffice.org UNO API. It takes care of the graphical user interface (menu's, dialogs, progress bars, etc.) and allows for loading settings from and saving settings to OpenOffice.org or the OpenOffice.org Writer document. For the actual document processing, Odt2BrailleAddOn relies on the Odt2Braille library.

Packages in Odt2BrailleAddOn:

`be.docarch.odt2braille.addon`

Odt2Braille

The Odt2Braille library takes care of the actual document processing. It enables the conversion of a flat xml odt file to a pef (Portable Embosser Format) file. Furthermore, this pef file can be converted to a variety of other generic braille formats, or it can be converted to an embosser-specific braille file (and optionally sent to an embosser device). The braille transcription is powered by liblouisxml, and for the pef processing, Odt2Braille uses the DaisyPipeline library. The Java OpenDocument Library (JODL) is used for creating and cleaning the flat .odt file (see `http://odt2daisy.sourceforge.net/downloads/`).

Packages in Odt2Braille:

`be.docarch.odt2braille`

DaisyPipeline

The DaisyPipeline project contains the `org_pef_text` and `org_pef_text.pef2text` packages. These packages have been adopted from the DAISY Pipeline and have been slightly modified. Their purpose is to convert a PEF 2008-1 document into a plain text braille file (see also the `pef2text` documentation).

Packages in DaisyPipeline:

`org_pef_text` and `org_pef_text.pef2text`

3 Liblouisxml

Liblouisxml is the heart of the braille transcription. It is an open-source library intended to provide complete Braille transcription services for XML documents. Liblouisxml is built on top of liblouis, its translation engine. The translation is driven through text based translation tables which define the translation rules. The formatting of braille is defined in semantic mappings that define how a specific XML input tag is to be rendered in the Braille output. Liblouisxml is embedded in odt2braille as an executable. It can be found in the Odt2BrailleAddOn project under `liblouis/bin`. The translation tables and configuration files are kept under `liblouis/share`.

For more information read the `liblouisxml` and `libouis` manuals.

4 Tutorials

Adding an embosser

To add support for an embosser, the protocol for giving print instructions to that embosser has to be known. This includes e.g.

- the configuration of the header which initiates the print job and gives general printing information,
- the way Braille pages and Braille lines are represented,
- the way each Braille character is represented (the character set),
- the footer which ends the print job, etc.

In addition, you should know the dimensions of cell spacing and line spacing, whether the embosser can print interpoint (duplex), which paper sizes are supported, etc.

Once the protocol is known, it can be implemented. What follows is a more or less general way of adapting the code in order to add an embosser. But because each embosser is different, extra adjustments may have to be made.

- In `org_pef_text.pef2text.EmbosserFactory`, expand the `EmbosserType` enumeration with a new embosser type. Add a `case` for this new embosser type to the `switch` statement in the `newEmbosser` function.
- Possibly, a new character set may have to be defined in `org_pef_text.TableFactory` as well. Edit the `TableType` enumeration and the `newTable` function.
- Finally, in `be.docarch.odt2braille.Settings`, the functions `embosserIsSupported`, `changeEmbosser`, `tableIsSupported`, `paperSizeIsSupported`, `changePaperSize`, `getMaxPaperWidth`, `getMaxPaperHeight`, `getMinPaperWidth`, `getMinPaperHeight` and `duplexIsSupported` need adjustments.

5 Task list

- More flexibility by providing more settings!
  - Footnotes, endnotes, transcriber's notes...
  - Volume info and transcription info => tags gebruiken die worden vervangen: <transcriber> = instelbaar <volumenaam> = instelbaar per volume <datum> <aantal volumes> <aantal supplementen> <aantal preliminaire volumes> <beginbraillepagina> <eindbraillepagina> <beginprintpagina> <eindprintpagina>
  - 'Continued' suffix.
  - Tables
  - Textboxes
  - Images
  - Table Of Contents: - headings tot level x weergeven ? - printpagina nummers weergeven ? - braillepagina nummers weergeven ?
  - Bibliography
  - Special symbols: knop om default waarde van special symbol in te stellen => getDefaultSpecialSymbol(SpecialSymbolType type, String language)
  - ...
- Mac OS, Linux, ... !!!
- Add more "Braille formatting standards" (UK, Zwitserland, België, Nederland ?) (=> zie: odt2braille_Formatting_Standards.odt)
- Javadoc bijwerken.
- Support more embossers.
- Only Interpoint55 is tested.
- Tooltips in dialogs.
- Keyboard shortcut for Braille menu.
- Howto handle unknown characters? (-> now: dots 3456) (-> "undefined" opcode ?)
- Hidden paragraphs (or paragraphs in hidden sections) => transcriber's notes (braille-only material). TN => settings!
- 8 dot Braille.
- Tactile graphics.
- If the liblouisxml process takes to long (e.g. with Chinese), OpenOffice.org might think the program is not responding.
- OpenOffice.org accessibility for screen readers (on Windows)?
- http://www.thessalonica.org.ru/en/index.html ?
- Check if the number of cells per line (and the number of lines per page) is sufficient (if too small, this might cause liblouisxml to fail).
- Endnotes in preliminary pages: fix bug.
- makePEF(): should be made more robust.

- Volledige conversie (DOM + XSLT) vervangen door een reeks van kleinere XSLT's (eenvoudiger, sneller) * Talen * Styles (in het begin: nodig voor analyse door Settings) * Lege paragrafen, ... * Secties verwijderen * Paginanummering * Lijstnummering & bullets * Kopnummering * Linken van captions * Omzetten naar "DAISY" * Titel pagina * Special typeface * Splitsen in volumes (pas op einde) (=> gebruik maken van "odt2braille" namespace voor eigen tags) (=> Vincent Spiewak is rewriting JODL in XSLT 2.0)
- Java bindings?
- UTD & liblouisutdml?
- Presentations & spreadsheets.
- Commandline tool (Odt2Braille.jar).
- Er kunnen nog lege paragrafen voorkomen (als er lege 'span's inzaten: bv door special typeface / languages) Door 'clear formatting' toe te passen op deze paragrafen is dit probleem weg.
- Uitleg van WinBraille over g0,g1,g2,... * g0 = one to one * g1 = literary one to one with capital and figure prefix * g2,g3,g4,... = contracted braille
- Wanneer de instelling "export/ emboss naar specifieke brailleprinter" is opgeslagen => OOo crasht (Windows XP) bij oproepen van dialoogvenster => is dit nog steeds zo?
- Mac OS: crasht soms bij laden van settings => nog steeds zo?
- Voor Ubuntu: OOo 3.2.1 nodig ?
- Settings in Tools > Options...
- Voor elke nieuwe release: kijken of liblouis zijn tabellen heeft upgedate.
- Zoveel mogelijk originele tabellen includen en indien nodig (deels) overschrijven met eigen tabellen.
- Voor elke (paragraph) stijl: gebruiker kan formatting parameters instellen (Feature request 3058074). => geen effect op headings, captions, listitems, notes, table cells, title page ? => default = overerven, keuze aan gebruiker (Voor elke (paragraph / character) stijl: gebruiker kan aangeven of special typeface moet overgenomen of genegeerd worden.) (=> effect op headings ?)
- odt2braille_Formatting_Standards.odt: roodgekleurde regels zijn degene die nog niet ondersteund worden.
- Automatic splitting of volumes
- Varia
  - text:list-item[@text:style-override] ?
  - Images are omitted in Braille if they are anchored to a page => alert?
  - Special-typeface. Er wordt enkel gekeken naar @style:style-name (+ @fo:font-style en @fo:font-weight) en @style:parent-style-name (in het geval van Emphasis en Strong_20_Emphasis). Todo: tot aan top-stijl gaan om stijl te achterhalen => recursief.
  - Pagenum inside notesection ?
  - Footnotes onder tabel: ook footnotes in captions meegerekend (terwijl die al onder de caption zelf gezet zijn)

  - als Format – section – options – collect at end of section niet expliciet wordt ingesteld => notes-configuration mogelijk niet aanwezig => notesection in Braille soms op verkeerde plaats (=> uitzoeken hoe OOo wel weet waar notes moeten staan ?)
  - OOo: linking sections?
  - Volume information block: preliminary braille page numbers (p...) => p moet zonder letter indicator
  - Special symbols list: Dot locator: http://www.iceb.org/rep9910f.html ?
  - Underline: dots 7 & 8 ?
  - Veranderen van typeface in midden van woord => documentatie
  - Kan paginanummering mislopen indien pagebreak wordt bekomen door een reeks line breaks ?
  - Supplementary volumes: * pagenumber prefixed by s ? * 1st pagenum should be 1.
- ...