



The ODFToEPub User Guide

Copyright © 2012 Pincette bvba. All rights reserved.

TABLE OF CONTENTS

1 Introduction	1
2 Running It	2
2.1 OpenOffice.org Extension	2
2.2 Stand-Alone Version	3
2.3 Batch Version	3
3 Layout	4
3.1 Using Frames	4
3.2 Indexes and Tables	4
3.3 Running Headers And Footers	5
3.4 Segmentation	5
3.5 The Cover Page	5
3.6 Footnotes	6
3.7 Endnotes	6
3.8 The Technical Table of Contents	6
3.9 Font Scaling In Adobe Digital Editions	7
3.10 Meta-Data	8
3.11 Embedding Fonts	8
4 Special Style Summary	9

1 INTRODUCTION

With ODFToEPub creating an e-book has become a very simple matter. You use this extension to export a document in OpenOffice.org Writer to the popular ePub-format. The style features in Open Document Format (ODF), which is the format used by Writer, are translated to an ePub style sheet. This gives you a great deal of control over how the resulting e-book will look. Most structural features, such as tables, itemized lists, footnotes, pictures, links, etc. are also supported. Add to this indexes like the table of contents, an index of tables, an index of illustrations and an alphabetical index and you have everything you need to create a professional e-book in a standard format.

ODFToEPub follows as much as possible what ODF has to offer, but it also introduces a few conventions to help you achieve a better result. This is always done through styles, more specifically style name conventions. There are not many and most of them just provide you with control over how generated material such as the indexes will look.

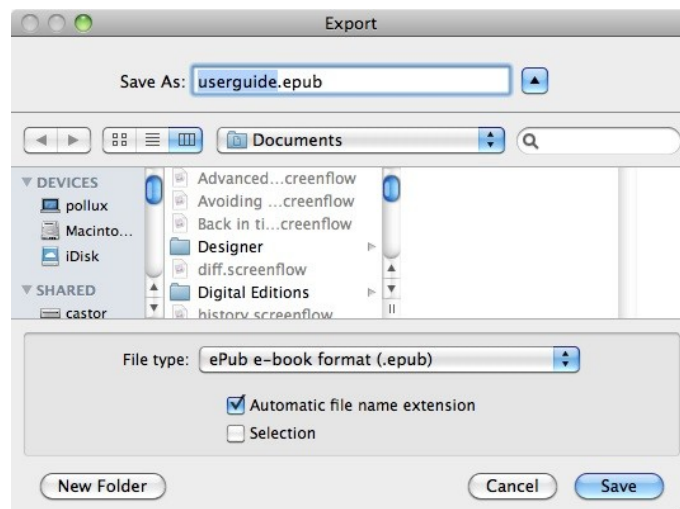
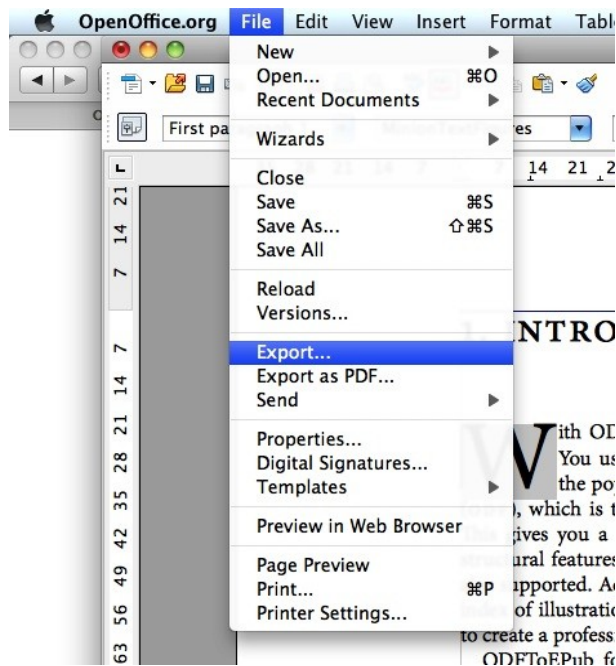
The fact that you have to work in OpenOffice.org to obtain an e-book doesn't mean you have to write the document in it (though it is very convenient and free). OpenOffice is able to open documents in other formats such as Microsoft Word variants, WordPerfect and HTML.¹ Once opened, documents in those formats can also be exported to ePub.

¹ After opening an HTML-file you should export it first as ODF text document, which you can then open to export it to ePub.

2 RUNNING IT

2.1 OPENOFFICE.ORG EXTENSION

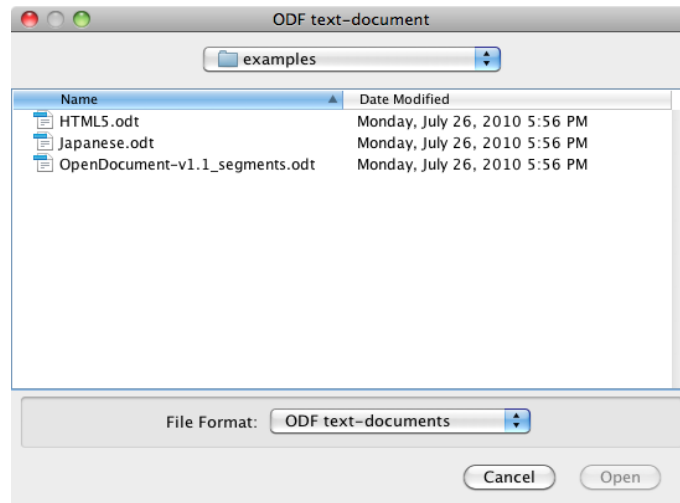
To run ODFToEpub you first have to open the document in OpenOffice.org. Then you choose the menu-item File→Export, as shown in the first figure below. A dialog box will pop up. In the “File type” drop-down box you have to select “ePub e-book format”. The document will then be saved with the “.epub” extension. Alternatively you can use the toolbar button.



Running It

2.2 STAND-ALONE VERSION

The stand-alone version is a Java-archive. On most systems you can double-click on it to launch it. Otherwise you have to open a terminal window and type “java -jar odf_to_epub.jar”. It is also possible to launch the application via the link http://www.pincette.biz/odftoepub/odf_to_epub.jnlp. The program will ask for an ODT-file, i.e. one with an “.odt” extension, and save an ePub-document next to it.



2.3 BATCH VERSION

If you have a batch license you can use ODFToEPub as a library. Consult the [Javadoc](#) to learn how. You can also run it on the command-line like this:

```
> java -jar odf_to_epub.jar document.odt [document.epub]
```

3 LAYOUT

The layout of an e-book should reflow in order to produce good results on the different types of e-readers in the market. That is because they all have different screen sizes. Therefore, as a general rule, it is advisable to use styles as much as possible and relative sizes wherever possible. Absolute widths or heights, fixed offsets on a page, tab-stops, hard spaces, those are things that might look nice in the word processor, but when the screen size of the e-reader is only half that of a normal page, some material might fall off the screen.

However, a WYSIWYG type of word processor doesn't always let you specify relative sizes. When you create a table, for example, you drag the lines between the columns to give them their width. This will result in absolute column widths being saved in the document. The same goes for pictures you import from a file or frames you insert into the text. To accommodate this ODFToEPub calculates the various widths in proportion to the containing element, i.e. the page gallery, the surrounding frame, etc. The produced ePub-file will express those widths in percentages.

3.1 USING FRAMES

With frames you can isolate a picture, a table or any other content in a box and manage its position on the page. For example, when you import a picture from a file it will be automatically placed in a frame. It is important that you never anchor a frame to the page, but instead to a paragraph or as a character. Otherwise the frame won't appear in the flow of the document. This is because it is unknown up front how the pages in the e-reader will be cut.

Frames can be centred or left or right aligned. Through the Format→Wrap menu-item in OpenOffice.org you can determine how the text will be laid out in relation to the frame. The option “Optimal Page Wrap”, for example, will flow the text around it.

3.2 INDEXES AND TABLES

If you want an inline table of contents in your e-book, you have to insert a table of contents using the menu-item Insert→Indexes and Tables→Indexes and Tables. However, the resulting table will be ignored. ODFToEPub uses it as a trigger to generate its own table. Those are produced from the headings in the document with the corresponding outline levels. The entries will be hyperlinked to the headings.

You can control the style of the table entries by adding paragraph styles with the names “TableOfContentsLevel1”, “TableOfContentsLevel2”, etc. If there are no such styles a default layout will be generated.

In the same way you can add an index of tables or an index of illustrations. The paragraph styles to control their entries are “TableIndex” and “IllustrationIndex” respectively.

The alphabetical index is the last in the row that can be added like this. Its generated structure is, however, a bit more complex. First, an enumeration of all used begin letters is displayed. Each is linked to the section of entries that starts with it. The layout of the enumeration can be controlled with the paragraph style “Alphabetic-

allIndexLetters”. Second, for each used begin letter there is a section with a section title. Changing the layout of the title is done with the “AlphabeticalIndexSection” paragraph style. The index entries in a section are styled with the “AlphabeticalIndexEntry” paragraph style.

3.3 RUNNING HEADERS AND FOOTERS

In principal you can use any kind of header or footer in your page masters. However, keep in mind that an e-book is presented on a rather small screen, so it is better to limit headers and footer to just one line of text. Note also that quite a few e-readers don't yet support running headers and footers. When they don't the header and footer text will appear in the text flow on top of the first page they were created on.

3.4 SEGMENTATION

The text of an ePub-file is mostly split up in several parts, because many readers have problems with opening a large document in one big chunk. They usually don't have enough working memory for that.

ODFToEPub will always split on headings with outline level 1. For documents with long chapters this can still be too big. Therefore, you can control segmentation through page breaks, either manually through Format→Paragraph→TextFlow or by specifying a “page break before” on a style, e.g. “Heading 2”. If two consecutive elements have a page break set, the second one won't start a new part, because that would look silly. There should be elements in between in order to produce two parts for this.

If some style has the “page break before” set, there may be places where the split doesn't look well. In that case you can overrule this by removing the break in the paragraph properties of the particular element.

Segmentation cannot be done with elements that are inside other elements, such as a section, a table or a frame.

3.5 THE COVER PAGE

There are two ways to create a cover page. The safest one is using a real picture, because that works on most e-readers. You start your document with a picture and give it the name “Cover”.² It doesn't matter how this picture is laid out in the document, because it will be removed from the flow.

If there is no cover image, the first part that is created during segmentation is taken as the cover page. You can, for example, import the artwork for the cover on the first page and have it occupy the complete page. You then start the next page with a “page break before”. This will only work for e-readers that support XHTML-entities for cover pages.³

² In OpenOffice.org you have to select the picture and then choose the menu-item “Format→Picture...”. In the box that appears you go to the tab “Options” and change the name.

³ iBooks doesn't support this.

3.6 FOOTNOTES

Since the division in pages in the e-reader is not known up front (some even use scrolling instead of pagination), all footnotes in a part are collected at the end of the part as end notes. Footnotes and their references are mutually hyperlinked.

The footnotes configuration, which can be changed through the menu-item Tools→Footnotes/Endnotes, determines how the footnote numbers are generated. Changing the start-value has no effect at the moment because of issue [116994](#) in OpenOffice.org.

There are three special paragraph styles you can add to the document in order to modify the layout of the footnote block at the end of a part. The style “Footnotes” lets you change the block as a whole. The style “FootnoteReference” is for the cell that contains the reference symbol of a footnote. The style “FootnoteBody” governs the cell that contains the text of the footnote.

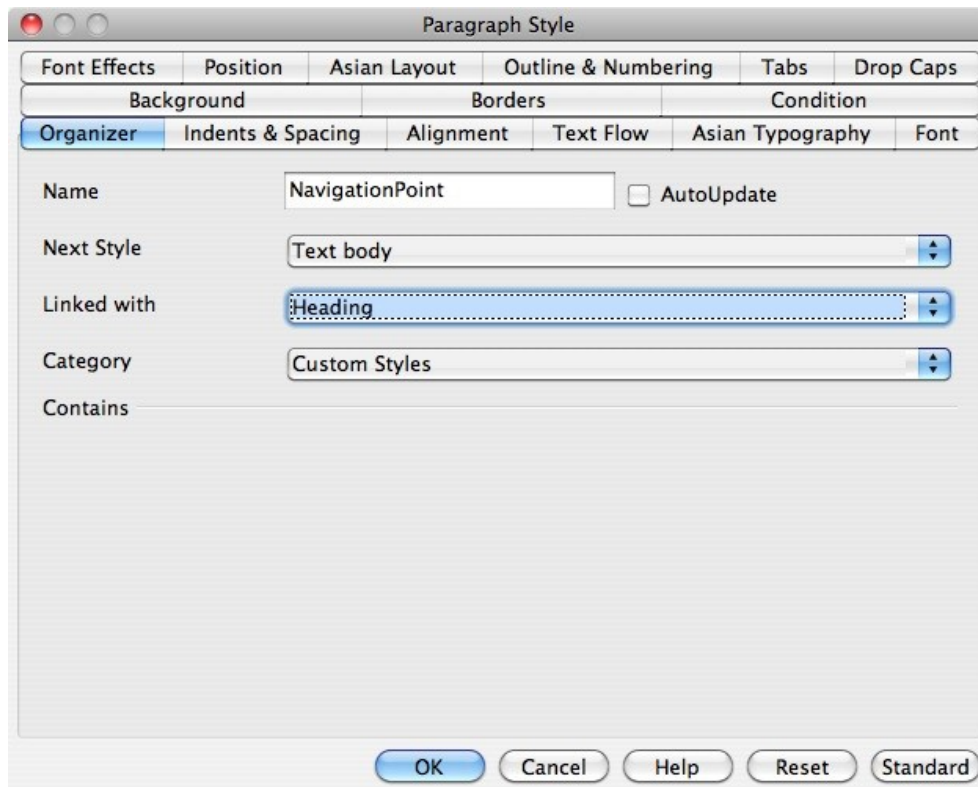
3.7 ENDNOTES

The treatment of endnotes resembles that of footnotes. In order to trigger the generation of endnotes you have to provide a heading somewhere in the document that has the style “EndnotesTitle” or a style that is derived thereof. The endnotes will be placed after the heading.

You also have three special paragraph styles called “Endnotes”, “EndnoteReference” and “EndnoteBody” that have the same function as the corresponding styles have for footnotes.

3.8 THE TECHNICAL TABLE OF CONTENTS

This is the control structure in the ePub-file that lets the e-reader generate a navigation tree, which is directly accessible from anywhere in the book through a special button of some sort. ODFToEPub puts all headings up to outline level 3 in



it. It is possible to add additional titles. You have to assign the style “Navigation-Point” to the element that represents the title. If you already have a style for such titles, all you have to do is insert the “NavigationPoint” style anywhere in the style hierarchy, as shown in the above figure.

As an example, suppose you want all elements that have the style “Heading”, directly or indirectly, to be navigation points. Also suppose the “Heading” style is linked to “Default”. You can then create the “NavigationPoint” style via the Format→Styles and Formatting menu-item. Select “Default”, right-click on it and select “New”. The new style will be automatically linked with “Default”. Then select “Heading”, right-click and select “Modify”. In the box that appears you have to change the field “Linked with” to “NavigationPoint”.

3.9 FONT SCALING IN ADOBE DIGITAL EDITIONS

In an e-reader the user can change the overall font size to accommodate her or his needs. There is a severe problem with Adobe Digital Editions when it comes to scaling fonts. It can't scale fonts with a font size expressed in points (pt). This is strange because that font size unit is not fixed to physical dimensions, unlike the pixel unit (px). The DPI of the raster, combined with a scaling factor, determines exactly how many pixels go into one point.

ODFToEPub works around this problem by generating only proportional font sizes, i.e. expressed in percentages. This has the additional advantage that the generated document adapts better to the e-reader, because everything starts from the

base font size chosen by the user. Therefore, you should simply use the default font sizes for text and make sure that other elements such as headings look good relative to the text.

3.10 META-DATA

Meta-data is taken from the input document. The standard properties that are supported in both ODF and ePub are transferred. There is a special case for the identifier of the book. If the DublinCore property `dc:identifier` is present (not in OpenOffice.org) it becomes the book identifier, otherwise the custom property “Identifier” is used. If that also doesn't exist a UUID will be generated instead.

A few other custom properties are translated to their DublinCore counterparts. These are “Contributor”, “Coverage”, “Language”, “Publisher”, “Relation”, “Rights”, “Source” and “Type”. Their definition can be found at <http://dublincore.org/documents/dces/>.

With the custom property “Author” the default author can be overridden.

3.11 EMBEDDING FONTS

It is possible to embed fonts in the generated ePub-document with a separate tool. You can launch it through the following link: https://www.pincette.biz/odftoepub/-embed_fonts.jnlp. The program will ask for an ePub-file, i.e. one with an “.epub” extension. Then it asks for the font files that should be embedded. You can select multiple such files. The ePub-file will be updated. You can run the tool several times on the same ePub-file, which may be necessary if the font files are spread over several folders.

The need for embedding fonts depends on the e-reader. Most of them have all contemporary fonts at their disposition, but e-readers based on Adobe Digital Editions often lack glyphs for special characters in their fonts. Since adding fonts increases the size of the ePub-file, it is best to test your e-book on several devices.

4 SPECIAL STYLE SUMMARY

The optional paragraph styles in the following table let you control certain features.

<i>Style name</i>	<i>Function</i>
AlphabeticalIndexEntry	Layout of the entries in the alphabetical index.
AlphabeticalIndexLetters	Layout of the enumeration of the begin letters in the alphabetical index.
AlphabeticalIndexSection	Layout of the titles of the sections per begin letter in the alphabetical index.
EndnoteBody	Layout of the cell that contains the endnote text.
EndnoteReference	Layout of the cell that contains the reference symbol of the endnote.
Endnotes	Layout of the complete endnote block.
EndnotesTitle	The heading style that triggers the generation of the endnotes. The actual style that is used in the document may be derived from it.
FootnoteBody	Layout of the cell that contains the footnote text.
FootnoteReference	Layout of the cell that contains the reference symbol of the footnote.
Footnotes	Layout of the complete footnote block at the end of a part.
IllustrationIndex	Layout of the entries in the illustration index.
NavigationPoint	Marker to add a title in the technical table of contents.
TableIndex	Layout of the entries in the index of tables.
TableOfContentsLevel<1..n>	Layout of the entries in the table of contents.