

Pandoc と LibreOffice Writer で iD エディタのマニュアルを製本する

のがたじゅん nogajun@gmail.com
2014/12/6

Pandoc と LibreOffice Writer で iD エディタのマニュアルを製本する

この記事は [LibreOffice Advent Calendar 2014](#) 6 日目の記事です。

Web 上から [OpenStreetMap](#) が編集できる、iD というエディタがあります。このエディタの解説が [LernOSM](#) に Public Domain で公開されているのですが、OSM マッピングパーティで人が集まった時には手元で見られるように紙に印刷したいと思い、[Pandoc](#) と LibreOffice Writer を使って冊子を作成しました。

Pandoc とは

[Pandoc](#) は、Haskell で書かれたドキュメント変換ツールです。対応文書フォーマットの多さと変換クオリティの高さに定評があります。Pandoc の出力フォーマットに odt があるので、今回は Pandoc を使って HTML から odt に変換し、LibreOffice Writer を使ってレイアウト編集、それを印刷して製本という手順で冊子を作成します。

Debian を使っている人は、apt でインストールできるのでインストールしておきましょう。

```
$ sudo apt install pandoc
```

それ以外のプラットフォームの方は「[Installing](#)」を参照して下さい。Pandoc の詳しい解説についてはユーザーズガイドの日本語訳もあるので、こちらも参照して下さい。

- Pandoc ユーザーズガイド 日本語版 - Japanese Pandoc User's Association: <http://sky-y.github.io/site-pandoc-jp/users-guide/>

まずは Pandoc を使って odt に変換してみる

冊子の元原稿となる、iD エディタマニュアルの Web ページは以下になります。

- iD エディタ - LearnOSM: <http://learnosm.org/jp/editing/id-editor/>

iD エディタのマニュアルは単一ページです。Pandoc は http でアクセスして HTML 文書を取得できるので、以下のコマンドだけで変換ができます。

```
$ pandoc -f html -t odt -o ideditor.odt http://learnosm.org/jp/editing/id-editor/
```

では、変換した odt ファイルを Writer で開いてみましょう。

パッと見は悪くないように見えますが、フォントの設定などに違和感を感じると思います。これは Pandoc が odt に変換する際に参照している「reference.odt」(テンプレートの ott ではありません)が、日本語の文書に適していない設定になっているためです。

この問題は、日本語の文書向けに設定した odt ファイルをオプション「--reference-odt=FILENAME」で指定すると解決できるので参照用 odt ファイルを作成します。

Pandoc 参照用の odt ファイルを作成する

Pandoc 参照用 odt ファイルは、スタイルを設定しただけの空の odt ファイルですが、独自に作成した odt ファイルを使うとうまく変換できない場合もあります。ですので、オリジナルの reference.odt をひな型にして作成します。

reference.odt については、Debian では /usr/share/pandoc/data/reference.odt にあるので、これをコピーするか、マニュアルにあるように出力して保存したものを用意します。

```
$ pandoc --print-default-data-file reference.odt > reference.odt
```

それでは、用意した reference.odt を Writer で開いてスタイルを編集しますが、スタイルの編集の前に言語設定が日本語環境向けではないので修正しておきましょう。メニューバー[ツール]-[オプション]からオプションダイアログを開き、以下の個所を変更します。

- [言語設定]-[言語]のアジア諸言語が「なし」になっているので、「標準 - 日本語」に変更
- [LibreOffice Writer]-[規定のフォント(西欧諸言語)]が変更されているので、「標準」ボタンを押して戻す
- [LibreOffice Writer]-[規定のフォント(アジア諸言語)]が変更されているので、「標準」ボタンを押して戻す

これで日本語の文書でも大丈夫です。このほか日本語関係の設定で注意する点としては、標準ページスタイルの用紙サイズが「レター」になっていることと余白の幅が変更されています。

日本語の標準設定では、用紙サイズ「A4」、余白「2.0cm」ですが、出力する文書によってサイズ、余白は変わってくるので、メニューバー[書式]-[ページ]から確認して、自分が使用する用紙サイズと余白に設定しておきます。

スタイルを作成する

ここからはスタイルの作成ですが、スタイルについては、手を入れる個所が多く、文章での説明は長くなるので、用意したサンプルファイル(reference-japanese.odt)を元にポイントを説明します。

参照用 odt ファイルには、スタイルのテストを書いておく

参照用 odt ファイルは、中身に何が書いてあっても Pandoc からは無視されます。スタイルを作成するとき見栄えを確認しながら調整すると作成しやすいので、あらかじめ使用するスタイルを適用した文章を書いておくといよいでしょう。

オプションの「規定のフォント」は揃えておく

「規定のフォント」はスタイルでフォントが指定されていない場合に利用されます。最初に「規定のフォント」を戻しましたが、標準では欧文(西欧諸言語)と和文(アジア諸言語)それぞれ違うフォント、違うサイズで設定されるので、見栄えを整えるときに思わぬところで悩む原因になります。

和文と欧文をそれぞれ違うフォント・サイズを割り当てる設定は、スタイルを利用してもできるので、別々に設定したい場合は、スタイル側で設定をするようにして、オプションの「規定のフォント」は和文フォントに揃えておくといよいでしょう。(reference-japanese.odt では、見出し以外は IPAex 明朝に揃えてあります。)

段落スタイルの調整ポイント

- 「標準」は最初に設定する
- 「本文」を設定したら、Pandoc 独自スタイルの「First Paragraph」も忘れずに設定する
- 「見出し」は作りこみましょう。
 - ・ 「見出し 1」は体裁の区切りを「挿入」にチェックを入れる

段落スタイルの「標準」はすべての段落スタイルの大元になるので、LibreOffice のスタイル作成の基本としては、なにはななくとも最初に設定をします。とはいえ、「標準」のスタイルを作り込み過ぎると継承するスタイルにも影響が出るので、設定は基本的な設定だけにとどめておきましょう。

「標準」の次に設定するのは「本文」になりますが、Pandoc では「本文」のほかに独自スタイルの「First Paragraph」も設定されているので忘れずに設定します。

「見出し」については、日本語ではなじまない斜体を使った見出しがあるので、斜体を使わない形に作りなおしておくで見栄えが良くなります。

大見出しである「見出し 1」は、体裁タブにある区切り「挿入」にチェックを入れて、種類を「ページ」にしておくと、「見出し 1」があればページの先頭から始まるようになるので設定しておくといよいでしょう。

これら以外のスタイルについては、最初からすべてを作りこむのは大変なので、足りない部分を継ぎ足していく形で作成すると楽に作業を進められます。

変換したファイルが使用しているスタイルを調べるには、「スタイルと書式」の表示を「使用したスタイル」にすると、使用しているスタイル一覧が表示されるので、これを元にとよいでしょう。

参照用 odt ファイルを適用して変換する

作成した参照用ファイルを適用して HTML 文書を odt に変換します。

変換の前に、毎回、ネットワークからダウンロードするのは時間もかかるので、ローカルに保存をしてから pandoc で変換します。

HTML の保存は、Firefox/Iceweasel の「ページを保存」を使うと手軽に画像を含めた完全な形で保存できるのでお勧めです。それ以外でも画像なども含めて保存できるなら、どの方法を使っても構いません。

では、odt に変換してみましょう。最初と違うのは「--reference-odt=reference-japanese.odt」で作成した参照用 odt を指定している部分が違います。

```
$ pandoc -f html -t odt --reference-odt=reference-japanese.odt -o ideditor.odt
LearnOSM.html
```

変換した odt ファイルを開いてみます。

```
$ xdg-open ideditor.odt
```

なかなか、いい感じに変換できました。これをそのまま印刷してもよいのですが、HTML 文書で使われていたメニュー部分は不要ですし、レイアウトも紙の文書としてはおかしいので、これを調整して冊子に仕上げます。

Writer を使って冊子として編集する

LibreOffice の印刷には、用紙を真ん中で折ると冊子になるように印刷する「[パンフレットの印刷](#)」という機能があります。この機能を使うと簡単に冊子用の印刷・PDF の出力ができるので、A5 サイズの冊子として編集していきます。

ページサイズを A4 から A5 に変更

「パンフレットの印刷」をするには、ページを印刷する用紙(A4)の半分のサイズ(A5)のページを 2 ページ面付けする必要があるので用紙サイズを A5 に変更します。

変更は、サイドバーのプロパティから、標準ページスタイルの用紙サイズを「A5」余白を「狭い」(1.27cm)に変更しました。そして、表紙もつけたいのでページスタイルの「最初のページ」も同じ設定にしました。これでA4に2ページ分、面付けできるようになりました。

Pandoc で変換した画像に注意

あとは用紙に収まるように編集していきますが、画像については注意が必要です。

Pandoc は画像のアンカーをすべて「文字として」に設定して変換するので、画像をドラッグしても行内しか移動しませんし、画像の大きさを変更するとつられて文字組みもおかしくなります。ですので、画像を移動したりレイアウトを変更する場合は、アンカーを「段落に」に変更する必要があります。

タブを挿入して一気に表組みを作る

iD エディタの説明で、アイコンの機能説明が箇条書きで書いてありますが、これはちょっと不自然なので表組みに変更しました。

表を作成するとき、文字列がタブで区切られていると表の区切りとして認識されるので、箇条書きを解除して説明文とアイコンの間にタブを挿入。表組みにする個所を選択して[表]ボタンを押すと一気に表組みが作成できます。

画像の枠はスタイルで

HTML 文書の画像には灰色の枠がついていましたが、LibreOffice では枠スタイルの「画像」に「外枠」をつけることで似たような感じにしました。

表紙をつける

冊子なので表紙をつけたいと思います。

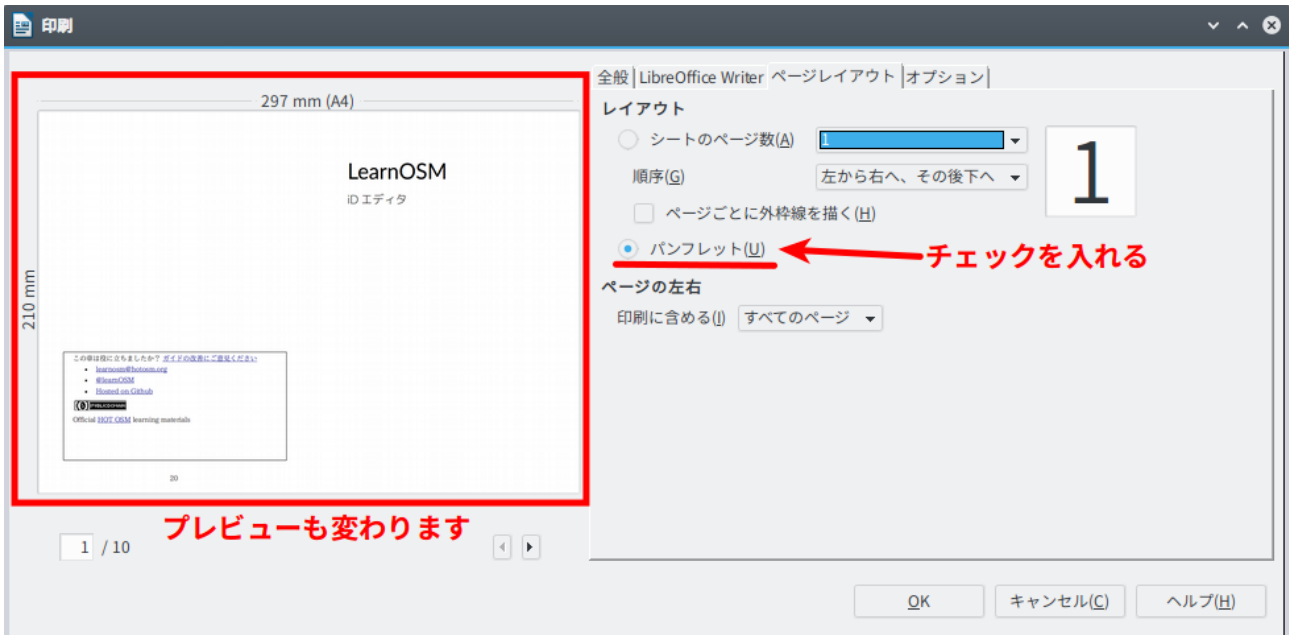
表紙をつけるには、「タイトルページ」機能(メニューバー[書式]-[タイトルページ])を使うと、表紙ページとページ番号を自動で設定してくれるので便利です。

今回はページ番号を内容ページからつけたかったので利用しましたが、時間があれば表紙に凝ってみるものいいかもしれません。

「パンフレットの印刷」で出力

編集が終わったら「パンフレットの印刷」で出力します。

「パンフレットの印刷」で出力するには、メニューバー[ファイル]-[印刷]で印刷ダイアログを開き、「ページレイアウト」タブにあるラジオボタン「パンフレット」をクリックします。



印刷する場合は、このままプリンターを選んで **両面印刷** をするとパンフレットとして印刷できます。

PDFに出力するには、「オプション」タブにある「ファイルへ出力」にチェックを入れると「印刷」ボタンが「ファイルへ出力」ボタンに変わるので、PDFとして保存できます。

PDFに出力するとコンビニなどで印刷もできるので、とても便利です。

完成

こんな感じにできあがりました。

(印刷中)

おまけ

作成した reference-japanese.odt ですが、常用する場合は毎回オプションをつけるのは面倒なので、ファイル名を「reference.odt」に変更して以下のディレクトリに保存しておく、odt に変換時に自動的に適用してくれます。

- Linux: \$HOME/.pandoc
- Windows: C:\Users\USERNAME\AppData\Roaming\pandoc

また、変換に使うテンプレートも変更できるので、以下のように出力した odt テンプレートを

```
$ pandoc -D odt ~/.pandoc/templates/default.opendocument
```

odt ファイルに含まれる content.xml を参考に変更すると、よりよい形に出力できると思います。

最後に

例では、HTML で書かれた文書を odt に変換して編集しましたが、Pandoc はさまざまな文書フォーマットに対応しているので、長文の原稿を Markdown で書いて、LibreOffice で編集して印刷といったこともできます。

ということで、Pandoc と LibreOffice Writer を使うと、テキストエディタで書いて、そこそこ見られる形に整形した文書が作成できるので活用してください。