Multi-target publishing

Axel Kielhorn

1 One road leads to one target

The usual target format of my documents was paper: ISO A4, ISO A5 or sometimes 3,5" × 5". My workflow led to an intermediate PDF file which was fine for reading on the screen, especially the smaller formats.

But then mobile devices appeared. The screen was too small to read A4 or even A5 documents. With some effort it was possible to create a document that was readable on *one* mobile device without excessive scrolling.

Having a format that reflows according to the size of the display with a user defined font size would be desirable. Such a format is ePub. It is simply a ZIP archive with a predefined structure and a few XML files that contain the actual content. A CSS file is used to control the appearance.

2 A detour

Luckily there is a program that reads LaTeX and writes ePub: Pandoc [5] (licensed under the GPL).

Unless the LaTeX file is too complicated, Pandoc will understand and convert it. But what is too complicated? The easiest way to find out is to convert a file from LaTeX to LaTeX and see what survives.

```
pandoc -r latex -t latex -o source-pd.tex
      source.tex
```

2.1 A rough road

Pandoc uses UTF-8 encoded files. This shouldn't be problem for most English speakers since they usually only use the first 127 characters of that encoding. But that is a naïve assumption. Even English speakers need non-ASCII characters for foreign words and punctuation characters. LaTeX offers many ways to enter these characters, but the only way that doesn't cause problems is to write them as UTF-8 characters. Thus `\^o` should be written as ô and `\o` as ø. A small few lines of `sed` will help with the conversion.

3 Back to square minus one

Is LaTeX really the starting point? Or should we see LaTeX as *one* backend and the LaTeX file just as an intermediate product?

4 An unusual direction: Markdown instead of markup

Markdown is a markup language developed by John Gruber [1] which looks as if no markup is present:

Editor's note: First published in *Die TEXnische Komödie* 3/2011, pp. 21-32; translation by the author.

A Markdown-formatted document should be publishable as-is, as plain text, without looking like it's been marked up with tags or formatting instructions.

The start of this article originally looked like the following in Markdown:

```
# One road leads to one target

The usual target format of my documents was
paper: ISO A4, ISO A5 or sometimes 3,5"
$\times$ 5". My workflow led to an intermediate
file which was fine for reading on the screen,
especially the smaller formats.

But then mobile devices appeared.  The screen
was too small to read A4 or even A5 documents.
With some effort it was possible to create a
document that was readable on *one* mobile
device without excessive scrolling.
```

This text was created from the original with:

```
pandoc -r latex -t markdown -o Ziele-tug.md
      Ziele-tug.tex
```

Markdown is a very limited language. The man page describing the language has only 16 pages. The "Not So Short Introduction to LaTeX 2ε" has ten times that number.

Converting from a complex language like LaTeX to a simple language like Markdown is difficult. Thus it is understandable that Pandoc only interprets a tiny amount of LaTeX markup. Since it doesn't understand TeX it uses regular expressions to parse the file. This will require additional empty lines in some cases where it is not required by TeX, otherwise the parser misses sectioning commands or environments.

Therefore it is best to convert a document to Markdown once and do all the future editing in Markdown.

5 A new road to an old target: Generating PDF from Markdown via LaTeX

```
pandoc -r markdown -t latex -o source.tex
      source.md
```

5.1 The `default.latex` file

The `default.latex` file distributed with Pandoc (in, e.g., `/usr/local/share/pandoc-X.Y/templates`) is a minimal example. With a little bit of LaTeX knowledge it can be customized to support the layout you need. A version adapted for German users is included in the supplementary material [2]. Modifications are marked with `-ak-`. A more elaborate file using the KOMA-Script class is included as well.

To call Pandoc with a custom template, use the command line:

```
pandoc -r markdown -t latex
       --template=./custom.latex
       -o source.tex source.md
```

5.2 A Shortcut

The fastest way to turn a Markdown file into PDF is:

```
markdown2pdf --template=./custom.latex src.md
```

This will generate an intermediate LaTeX file and call pdfLaTeX to create the PDF.

With the options `--xetex` or `--luatex`, you can select a different engine. The template detects the engine and selects the appropriate code via `ifxetex` and `ifluatex`.

5.3 Postprocessing

The generated LaTeX file is surprisingly good. It matches files written by novice users.

Of course there may be some overfull and underfull hboxes that need further attention.

6 A new target ahead: ePub

The original desire was to create an ePub file in addition to the PDF file. The following command will do that:

```
pandoc -r markdown -t epub
       --epub-cover-image=cover-image.gif -s
       -o Source.epub Source.md
```

The text will be split into separate files according to the structure of the document. Thus it is easy to post-process the file with an ePub editor like Sigil [6].

Version 1.8.1.2 added the option to include a cover image (as shown above), thus reducing the need for post-processing.

7 The road to OpenOffice

"May I have this as a Word file?" Who doesn't know this question? Let's meet in the middle of the road with a LibreOffice file.[1]

```
pandoc -r markdown -t odt
       --reference-odt=./reference.odt -s
       -o source.odt source.md
```

The file `reference.odt` will be used as a template for the formatting of the document. If you want to change the design, you should modify the file supplied with Pandoc to make sure the internal style names match the ones used by Pandoc.

---

[1] `Writer2LaTeX` can convert LibreOffice files into LaTeX.

If you get an error when opening the `odt` file complaining about a corrupt file, you need to update Pandoc — a bug prior to version 1.8.1.3 led to the creation of invalid files when images were included.

Including images is still problematic. The images are in the final document, but they have to be rescaled.

8 Travel preparations

A small `sed` program removes some markup and converts LaTeX characters to UTF-8:

```
s/\\LaTeX/LaTeX/g
s/\\TeX/TeX/g
s/\\ConTeXt/ConTeXt/g
s/\\begingroup//
s/\\endgroup//
s/\\^o/ô/
s/\\o/ø/
```

Call this program on the command line with:

```
sed -f tex2mdtex.sed Source.tex
    >Source-clean.tex
```

The result can be converted to Markdown with:

```
pandoc -r latex -t markdown -s
    -o Source-clean.md  Source-clean.tex
```

9 Road signs

9.1 Sectioning commands

Markdown supports six hierarchy levels for sectioning commands. The number of `#` signs indicates the level. There has to be an empty line in front of the sectioning command.

```
# Top level


## Second level


### Third level


#### *Important information* hidden in
    the fourth level
```

An alternative form of sectioning commands only supports two levels:

```
First Level
===========


Second and last level
---------------------
```

9.2 Block Quotations

Markdown uses email conventions for quoting blocks of text. Lines starting with a `>` character are treated as block quotations.

```
> This is a block quotation
>
> > And this is a block quotation
> > inside a block quotation.
>
>
> The > sign is only needed in the first
line of the quotation.
```

A special kind of quotation is a quotation from a program. This is usually printed in a monospaced font. If a line starts with four spaces, it is treated as a verbatim text.

```
␣␣␣␣\documentclass[a4paper]{ltugboat}
␣␣␣␣\usepackage[utf8]{inputenc}
```

If you don't want to indent every line, you can use a delimited block, which begins with 3 or more tilde (~) characters and ends with at least the same number of tilde characters. If the code already contains a row of tilde characters, use more to delimit the quotation.

```
~~~~~~~~
This is a program listing
~~~~
Header preceded by tildes
~~~~
Body preceded by tildes
~~~~~~~~
```

9.3   Lists

There are several list types in Markdown that we already know from LaTeX:

9.3.1   The `itemize` list

The `itemize` list is started with a bullet character (*, + or -).

```
* one
* two
* three
        - three a
        - three b
* four
```

If a list entry contains several paragraphs, the paragraphs should be indented with four spaces or one tab.

```
* one
* two
* three
        - three a
        - three b
* four
```

Axel Kielhorn

```
    As usual, we hide important information
    in the fourth item.

    To be really sure, the 4 space rule is only
    mentioned in the last paragraph.
```

9.3.2   The enumerate list

An ordered list is like a bullet list, but it starts with an enumerator (`1.`, `(1)`, or `i.`) instead. The enumerators need not be in the correct order, even if that looks funny.

This kind of enumeration automatically loads the `enumerate` package to get custom enumerators. The generic enumerator `#.` uses the enumerators defined by the document class and avoids loading an additional package.

```
1. one
2. two
4. three
        a) three a
        b) three b
5. four
    Hiding important information ...
```

9.3.3   The description list

Sadly these animals from the German `lshort` haven't made it into the English version. Therefore I will introduce them here.

The term described is on a line of its own; the description follows in the next lines. The description is started with a colon or tilde, indented by one or two spaces. A term may have multiple descriptions, and each description may have one or more paragraphs.

```
Gelse
  : a small animal, living east of the
    Semmering, that chases tourists away.

Gemse
  : a large animal, living west of the
    Semmering, chased away by tourists.

    A long paragraph discussing
    whether it should be Gemse or Gämse.

Gürteltier
  ~ A medium sized animal. It only appears here
    because it has a long name.

  ~ In Austria, Gürteltiere are usually seen
    only in zoological gardens.
```

```
--------------------------------------------------------------------
Centered        Default          Right    Left
Header          Header          aligned   aligned
-------------   -------    -------------   ---------------------
First           row               12.0    Example of a row that
                                          spans multiple lines.


Second          row                5.0    Here's another one. Note
                                          the blank line between rows.
--------------------------------------------------------------------
```

Figure 1: A multiline table

9.3.4 Numbered lists with references

Usually a new list starts with number 1. If you want your items numbered throughout the document, Markdown offers a special list marker that is not reset. These list markers can be used as a reference later, or earlier. This is comparable to the caption counters used by LaTeX, but doesn't use the \label/\ref mechanism and does not require a second LaTeX pass.

```
(@Statement) Here I state something.
```

```
The statement (@Statement)
will be proved in (@Proof).
```

```
(@Proof) This is the proof.
```

9.4 Tables

Starting with version 1.8.1.2 Pandoc uses the `ctable` package to create tables. When entering tables it is best to use spaces instead of tabs to align the columns. There are three kind of tables.

A simple table:

```
  Right   Left   Center   Default
-------   ------  --------  -------
     12   12        12          12
    123   123      123         123
     ab   ab        ab          ab
```

```
Table: A simple table
```

The table header and the table rows must be written on one line. The alignment is defined by the dashed line below the header.

- If the dashed line is flush with the header text on the right side but extends beyond it on the left, the column is right-aligned.
- If the dashed line is flush with the header text on the left side but extends beyond it on the right, the column is left-aligned.

```
+------------+---------+----------------------+
| Fruit      | Price   | Advantages           |
+============+=========+======================+
| Bananas    | $3.14   | - built-in wrapper   |
|            |         | - bright color       |
+------------+---------+----------------------+
| Oranges    | $2.82   | - cures scurvy       |
|            |         | - tasty              |
+------------+---------+----------------------+
```

Figure 2: A grid table

- If the dashed line extends beyond the header text on both sides, the column is centered.
- If the dashed line is flush with the header text on both sides, the default alignment is used (in most cases, this will be left).

A table must be terminated by an empty line.

You can provide a caption starting with the string `Table:`, or just the character `:`. Any `Table` and the colon will be removed from the output. The caption may appear either before or after the table. When a caption is used, the table will be set in a `table` environment, otherwise it will appear in the body text.

Multiline tables allow headers and table rows to span multiple lines of text. The rows must be separated by empty lines. An example is shown in figure 1.

A grid table is shown in figure 2. The cells may contain arbitrary block elements, including lists.

9.5 Title

Information about the title, author and publication can be given at the beginning of the file.

```
% Multi-target publishing
% Axel Kielhorn
% TUGboat Volume vv
```

Long titles may be broken into several lines.

```
% Viele Ziele\
  (Multi-target publishing)
% Axel Kielhorn
  Babel Fisch (Trans.)
% TUGboat Volume vv
```

The \ in the first line will be translated to \\ in the LaTeX output.

9.6 Footnotes

A footnote consists of two parts, the footnote marker and the footnote text.

```
This is a footnote marker[^1]
and this is another footnote[^fussnote]

[^1]: Here is the footnote text

[^fussnote]: This footnote is slightly longer.

    It contains a second paragraph.
```

9.7 Inline formatting

Italic text is surrounded by *one* * or _. Bold text is surrounded by *two* * or _. If you want to emphasize only a part of a word, you have to use * because _ is often used as part of a name.

```
This text was emphasized _with underlines_
and this *with asterisks*.
For __bold text__ you need **two** characters.
```

Superscripts are surrounded by ^, subscripts by ~ characters:

```
H~2~O is water, 2^10^ is 1024.
2^2^^2^ is 2^22^.
```

The output of the last line may be unexpected, but note that these are *text* super- and subscript commands, not math commands.

9.8 Math(s)

Inline math is surrounded by $ characters. It is processed by LaTeX, thus everything allowed in LaTeX is permitted.

```
$2^{2^2} != 2^{22}$
```

When using a different output format, the result depends on the capabilities of that format.

Display math can be entered as raw LaTeX.

9.9 Raw LaTeX

Everything between a \begin and an \end will be copied verbatim to the LaTeX output and ignored in all other formats.

9.10 Raw HTML

Markdown was designed to create HTML. Therefore it is possible to include raw HTML, which will be ignored by non-HTML based formats.

9.11 Links

It is not surprising that a language designed to create web pages supports hyperlinks. Everything included between angle brackets is considered a link.

```
<http://johnmacfarlane.net/pandoc/>
```

A link may appear in a paragraph.

```
Documentation may be found on the [pandoc
web site](http://johnmacfarlane.net/pandoc/).
```

9.12 Pictures

A picture is included by providing a link to that picture and starting that link with a !.

```
![A blue picture](blau.jpg "Blue picture")
```

If the picture appears on a line of its own, it will be set in a `figure` environment and the text in the square brackets will be used as a caption text. Otherwise, it will be included in the body text.

```
The ![red square](rot.png "red square") appears
in the body text.
```

There is no way to scale pictures, they need to be in the correct size and resolution. This causes problems when the same picture is used for the web (72 dpi) and printing (300 dpi).

10 Next exit: ConTeXt

Pandoc is able to generate ConTeXt files. This is an easy way to convert LaTeX files to ConTeXt.

With a filter module, ConTeXt is able to directly process Markdown by calling Pandoc with parts of the document. See the Pandoc Extra Wiki [4] for details.

11 Large documents . . .

The default settings for Pandoc is to create a document without section numbers and without a table of contents. This is fine as long as you write a short document, but when the size exceeds a few pages it would be better to have the sections numbered and a short overview over the contents:

```
pandoc -r markdown -t latex --number-sections
       -o md-test.tex md-test.md
```

numbers the sections, and with

```
pandoc -r markdown -t latex --toc
       -o md-test.tex md-test.md
```

a table of contents will be generated. Of course you can combine the options.

Axel Kielhorn

12   ... to full books

When you use the `report`, `book`, or `memoir` class, Pandoc will use the `\chapter` command as the highest sectioning level. It doesn't know about more exotic classes, like `scrbook`. If you want to use these, you have to request the chapters yourself:

```
pandoc -r markdown -t latex --chapters
       -o md-test.tex md-test.md
```

You can split the source text into several files and combine them when calling pandoc. This isn't as elaborate as the `\include` mechanism in LaTeX, but considering the speed of current computers, there is little need for `\include`/`\includeonly`. For a very long document you have to call Pandoc with a very long command line.

```
pandoc -r markdown -t latex
       --number-sections --toc
       --template=./report.latex
       -o md-test.tex
       md-test-intro.md
       md-test-ch1.md
       md-test-ch2.md
       md-test-ch3.md
```

13   Conclusion

With Pandoc, it is straightforward to create PDF files with LaTeX without knowing anything about LaTeX. All of the TeX processing can be hidden by using `markdown2pdf` and the various options in a shell script (or batch file). For example, I wrote an engine file for TeXShop that calls `markdown2pdf` instead of TeX. "That's too difficult" is no longer a reason not to use LaTeX.

Pandoc, or rather the Markdown language, has its limitations. If you need several kinds of foot- or endnotes, several bibliographies or lots of math, Pandoc is certainly underpowered. But if you don't need these features, Pandoc is an easy way to write structured documents without a high learning threshold.

The main work goes into the design of the `template` files, which should be created and maintained by a LaTeX expert.

14   Acknowledgements

15   Appendix: Supplementary material

The following are included in the supplementary material you can download from the TUG server [2]:

md-test-tug.md The examples from this article.

md-test-tug.tex Converted LaTeX.

md-test.pdf Set with pdfLaTeX.

md-test.epub Converted to ePub.

tex2mdtex.sed A sed script to make general LaTeX documents palatable to Pandoc.

pandoc.pdf Pandoc man page.

pandoc-markdown.pdf Man page describing the Markdown syntax used in Pandoc.

markdown2pdf.pdf markdown2pdf man page.

engines A folder with engine files for TeXShop. (They will be in the next official release.)

⋄ Axel Kielhorn
  Lesumstraße 10
  D-27283 Verden
  Germany
  `A dot Kielhorn (at) web dot de`

References

[1] John Gruber. *Daring Fireball: Markdown*, 2004. `http://daringfireball.net/projects/markdown/`.

[2] Axel Kielhorn. *Supplementary material*, 2011. `http://tug.org/TUGboat/tb32-3/tb102kielhorn-supp.zip`.

[3] Axel Kielhorn. Viele Ziele — Multi-Target Publishing. *Die TeXnische Komödie*, 3:21–32, 2011.

[4] John MacFarlane. *Pandoc Extras*, 2011. `http://github.com/jgm/pandoc/wiki/Pandoc-Extras`.

[5] John MacFarlane. *Pandoc — About pandoc*, 2011. `http://johnmacfarlane.net/pandoc/`.

[6] Strahinja Marković. *Sigil — A WYSIWYG ebook editor*, 2011. `http://code.google.com/p/sigil/`.