

Markdown を用いた原稿執筆について

総合情報基盤センター 講師 山下 和也

1 はじめに

富山大学教養教育科目「情報処理」の教科書は Microsoft Word を用いて執筆し Word ファイルを入稿して出版していた。2019 年度教科書の大幅な改編にあたって、原稿執筆方法を見直して PDF ファイルを入稿した。

PDF ファイル入稿に変更したことで、Word を用いる必要は無くなり、簡単な構文で文書作成できる Markdown を用いて原稿執筆した*1。Markdown で作成した原稿を CSS で組版する方法 [1] もあるが、 \LaTeX 文書に変換することで、 \LaTeX で組版して PDF ファイルを生成した。

本稿では、教科書の原稿作成に用いた Markdown, Pandoc, pandoc-crossref と原稿執筆から PDF ファイル生成の流れについて述べる。

2 Markdown

Markdown は、2004 年に John Gruber によって開発された Web 文書作成のための変換ツールである [2]。Markdown は、読みやすく書きやすいプレーンテキスト形式で文書を書くことができ、それを構造的に妥当な XHTML 文書や HTML 文書に変換できる。

簡単な構文で文書構造を記述できるので、 \LaTeX に比べて学習コストが低いことが利点である。また、プレーンテキスト形式で記述

*1 図が大量にある教科書を Word で組版するのは苦行であることも、原稿執筆の方法変更の要因の一つである。

するので、文章のレイアウトよりも内容に集中して書くことができる。

2.1 Markdown の書式

Markdown の書式は非常にシンプルである。例えば、見出し、段落、箇条書き、リンク（画像）は次のように記述することができる*2。

2.1.1 見出し

ソースコード 1 のように、行頭に '#' を付けることで見出しを記述できる。'#' の個数で見出しのレベルを定義できる。HTML 文書では、'#'1 個の見出しが H1 に対応し、'#'2 個から '#'6 個まで順に H2~H6 に対応する。本稿では \LaTeX 文書に変換する際に、'#'1 個の見出しを chapter、'#'2 個から '#'6 個まで順に section, subsection, subsubsection, paragraph, subparagraph に対応させた。

ソースコード 1 見出し

```
# H1見出し (chapter)
## H2見出し (section)
### H3見出し (subsection)
#### H4見出し (subsubsection)
##### H5見出し (paragraph)
##### H6見出し (subparagraph)
```

2.1.2 段落

\LaTeX と同様に空行に挟まれた文章が段落として扱われる。空行ではない単なる改行は無視される。

*2 本稿に記載した構文以外にも多くの構文がある。<https://daringfireball.net/projects/markdown/syntax>

ソースコード 2 段落

ここは段落です。

途中の改行は無視されて出力されます。

空行が段落の区切りです。

2.1.3 箇条書き

ソースコード 3 のように、行頭に ‘-’, ‘+’, ‘*’ のいずれかをつけると番号なしリストを記述できる。これは、 \LaTeX の `itemize` 環境に対応する。

ソースコード 3 番号なしリスト

- リスト1
 - リスト2
 - リスト3
-

ソースコード 4 のように、行頭に数値と半角ドットで番号付きリストを記述できる。これは、 \LaTeX の `enumerate` 環境に対応する。

ソースコード 4 番号付きリスト

1. リスト1
 1. リスト2
 1. リスト3
-

2.1.4 リンク (画像)

ソースコード 5 のように記述することで、原稿に画像 (図) を挿入することができる。これは、 \LaTeX の `figure` 環境に対応する。

ソースコード 5 リンク (画像)

```
! [Alt属性] (画像ファイルのパス)
```

3 Pandoc

Markdown ファイルを \LaTeX ファイルに変換して組版を行うが、その変換には Pandoc を用いる。Pandoc [3,4] を用いることで、Markdown から別の文書形式に変換できる。

Pandoc は、Markdown, HTML, \LaTeX , Word などの文書ファイルを、様々な形式 (Markdown, HTML, \LaTeX , Word など) に変換できる非常に強力な Universal Document Converter である。

Markdown 形式のファイル (input.md) から \LaTeX 形式のファイル (output.tex) への変換は次のようなコマンドで行う。

```
$ pandoc input.md -o output.tex
```

3.1 pandoc-crossref

\LaTeX であれば、“`\label`” や “`\ref`” を用いれば相互参照を記述できるが、Markdown では相互参照を記述できない。

そのため、Pandoc のプラグインである `pandoc-crossref` [5] を用いて相互参照を記述する。 \LaTeX では、図や表のいずれでも参照元は “`\label{name}`”、参照先は “`\ref{name}`” のように記述できるが、`pandoc-crossref` の相互参照は、参照するもの (図、表など) によって記述内容が異なることに注意が必要である。

図の場合、参照元は Markdown ファイルでソースコード 6 のように記述する。“`{#fig:name}`” の部分が、 \LaTeX の “`\label{name}`” にあたる。

ソースコード 6 相互参照 (label)

```
! [Alt属性] (画像ファイルのパス){#fig:name}
```

参照先は、Markdown ファイルでソースコード 7 のように記述する。“`[@fig:name]`” の部分が、 \LaTeX の “`fig. \ref{name}`” にあたり。 \LaTeX とは異なり “`fig.`” が変換時に付与される。“`fig.`” の部分は YAML で設定ファイルを書くことで “図” に変更できる [6]。

～を[@fig:name]に示す。

表の場合、参照元は “{#tbl:name}”、参照先は “[@tbl:name]” を記述する。式の場合、参照元は “{#eq:name}”、参照先は “[@eq:name]” を記述する。コードの場合、参照元は “{#lst:name}”、参照先は “[@lst:name]” を記述する。

pandoc-crossref を用いて相互参照を記述した Markdown を変換する場合には、次のようにフィルタ (-F pandoc-crossref) を指定する必要がある。

```
$ pandoc input.md -o output.pdf \
-F pandoc-crossref
```

4 原稿作成から PDF ファイル生成までの流れ

Markdown, Pandoc, pandoc-crossref, L^AT_EX を用いることで、Markdown の原稿から PDF ファイルを生成できる。原稿作成から PDF ファイル生成までの流れは次のとおりである。

1. Markdown + pandoc-crossref 形式の原稿ファイルを作成
2. Pandoc を用いて Markdown + pandoc-crossref 形式から L^AT_EX 形式へ変換
3. L^AT_EX を用いて組版
4. L^AT_EX ソースファイルをコンパイルして PDF ファイルを生成

5 おわりに

教科書の原稿作成にあたり、Markdown を用いて原稿ファイルを作成し、Pandoc を用いて L^AT_EX ファイルを生成、L^AT_EX で組版して PDF ファイルを生成した。

Markdown を用いて原稿執筆することで、文書内容に集中して執筆することができ、L^AT_EX で組版して PDF ファイルを生成することが可能になる。

参考文献

- [1] 吉川雅彦, “Markdown + CSS 組版での技術系同人誌制作時の環境”, <https://yoshikawaweb.com/markdown-css.html> (2019/3/31)
- [2] J. Gruber, “Daring fireball: Markdown”, <https://daringfireball.net/projects/markdown/> (2019/3/31)
- [3] J. MacFarlane, “Pandoc a universal document converter”, <https://pandoc.org/> (2019/3/31)
- [4] J. MacFarlane, “Pandoc User’s Guide”, <https://pandoc.org/MANUAL.pdf>, 122 pages, (2019)
- [5] N. Yakimov, “pandoc-crossref”, <https://github.com/lierdakil/pandoc-crossref> (2019/3/31)
- [6] さえきよしやす, “Markdown で相互参照を使うなら pandoc-crossref”, <https://laclefyoshi.hatenablog.com/entry/20150616/crossref> (2019/3/31)