

A very intriguing article title

Author One^{*†1,3}, Author Two^{†2}, and Author Three^{*3}

¹Affiliation 1

²Federation of Planets

³Acme Corporation

[†] These authors contributed equally to this work.

^{*} Correspondence: Author One <author.one@example.edu>
Author Three <author.three@example.edu>

Abstract

This template article is made using the markdown-based ‘pandoc-scholar’ approach. It uses a simplified (hassle-free) markdown input file, which is then processed to simultaneously produce many output formats including html, pdf, latex, docx (word), odt, epub. The goal is to produce a manuscript for submission to journals where only minor modifications (if any) need be applied to the final submitted file. When in doubt, let the copy editors take care of it.

Significance or key points

Some journals require significance statements or key points of the article. If not desired, just delete this section. This may be in the form of a short paragraph. Or it can be as a set of bullet points:

- Key point 1
- Key point 2
- Key point 3

Keywords: keyword1, keyword2, keyword3

1 Introduction

A lot of time is wasted on formatting for journals, which completely reformat the article to their liking in the final stage anyway. Let’s avoid that.

The ‘pandoc-scholar’ software provides a set of generalized templates that automatically creates nice academic manuscript files in a huge set of formats simultaneously. This allows you to write the manuscript in plain text markdown, and then distribute to colleagues or submit to journal in any required format. The key insight to this approach is **Do not waste effort fiddling with the**

format. The manuscript file contains all the content, and journals can worry about formatting details. Any changes that are absolutely required for submission should be applied manually to the output file just before submission.

2 Methods

Creating a beautiful manuscript is simple with pandoc-scholar.

2.1 Workflow

To create a manuscript using this template, follow these steps:

- 0) Be sure that required software is installed on your machine (see Appendix Sec. 5.1 for details).
- 1) Copy the entire pandoc-scholar-template repo <https://github.com/aswol/pandoc-scholar-template.git> and delete the git repo data (`rm .git`)
- 2) Rename directory and initialize new git repository if desired
- 3) Edit Makefile to point at pandoc-scholar run `make init` to initialize directory
- 4) Begin editing the article file (`article.md` by default)
- 5) run `make` to make all output files
- 6) Revise manuscript and optionally commit changes to git (really, you should use git)
- 7) return to step 5
- 8) submit article in required format

It is important to remember that both the main text and the YAML header must be edited to create a completed manuscript. The YAML header (everything above the Keywords & Abstract above) contains vital information for both the printed manuscript (like title and authors) as well as meta-data for the manuscript. Don't forget to also update or remove the project section to link to your own files as desired.

2.2 Basic Document Elements

2.2.1 Level 3 subsection

2.2.1.1 Level 4 subsection

2.2.2 Document Section References

Automating document references is one of the best features for making manuscript revisions painless. Using pandoc, any section can be given a

reference id by following tacking it onto the end of the header, e.g.:

```
## Section title {#sec:id}
```

This allows the section to be referenced using the standard markdown reference syntax `@sec:id`. For example, here is a reference to the Workflow discussion found in Section 2.1.

2.2.3 Inserting citations

For inserting a reference, the database key is given within square brackets, and indicated by an '@.' It is also possible to add information, such as page:

```
[@suber_open_2012; @benkler_wealth_2006, 57 ff.]
```

gives (Suber 2012; Benkler 2006, 57 ff.).

2.2.4 Figures

Pandoc-flavored markdown (<http://pandoc.org/>) adds several extensions which facilitate the authoring of academic documents and their conversion into multiple output formats. Table 1 demonstrates the simplicity of MD compared to other markup languages. Figure 1 illustrates the generation of various formatted documents from a manuscript in pandoc MD. Some relevant functions for scientific texts are explained below in more detail.

2.2.5 Tables

There are several options to write tables in markdown. The most flexible alternative - which was also used for this article - are pipe tables. The contents of different cells are separated by pipe symbols (|):

```
Left | Center | Right | Default
:-----|:-----:|-----:|-----
  LLL  |  CCC   |  RRR   |  DDD
```

gives

Left	Center	Right	Default
LLL	CCC	RRR	DDD

The headings and the alignment of the cells are given in the first two lines. The cell width is variable. The pandoc parameter `--columns=NUM` can be used to define the length of lines in characters. If contents do not fit, they will be wrapped.

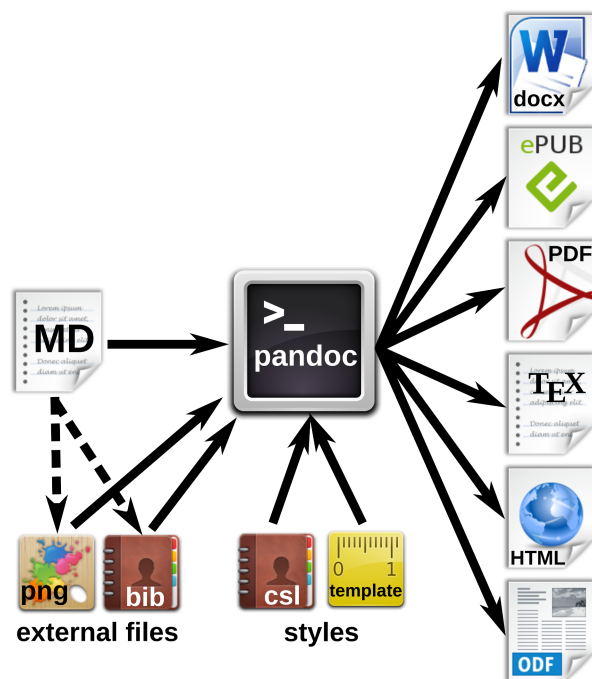


Figure 1: Workflow for the generation of multiple document formats with pandoc. The markdown (MD) file contains the manuscript text with formatting tags, and can also refer to external files such as images or reference databases. The pandoc processor converts the MD file to the desired output formats. Documents, citations etc. can be defined in style files or templates.

Complex tables, e.g. tables featuring multiple headers or those containing cells spanning multiple rows or columns, are currently not representable in markdown format. However, it is possible to embed LATEX and HTML tables into the document. These format-specific tables will only be included in the output if a document of the respective format is produced. This is method can be extended to apply any kind of format-specific typographic functionality which would otherwise be unavailable in markdown syntax.

Although the content elements of documents, such as title, author, abstract, text, figures, tables, etc., remain the same, the syntax of the file formats is rather different. Table 1 demonstrates some simple examples of differences in different markup languages.

Table 1: Examples for formatting elements and their implementations in different markup languages.

Element	Markdown	LATEX	HTML
structure			
section	# Intro	\section{Intro}	<h1>Intro</h1>
subsection	## History	\subsection{History}	<h2>History</h2>
text			
style			
bold	**text**	\textbf{text}	text
italics	<i>*text*</i>	\textit{text}	<i>text</i>
links			
HTTP	<https://	\usepackage{url}	<a href="https://
link	arxiv.org>	\url{https://arxiv.org}	arxiv.org">

2.2.6 Equations

Formulas are written in LATEX mode using the delimiters \$. E.g. the formula for calculating the standard deviation s of a random sampling would be written as:

$$s = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \overline{x})^2}$$

and gives this ($s = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2}$) for an inline figure. In the form of a labeled equation, we get the following:

$$s = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (x_i - \bar{x})^2} \quad (1)$$

with x_i the individual observations, \bar{x} the sample mean and N the total number of samples. And we can even reference the Equation 1 like this.

2.3 Super-fancy formatting with pandoc markdown

See pandoc documentation for all of the lovely details on pandoc-flavored markdown to achieve most any formatting you want. If it's too complicated for markdown, ask yourself if it really matters, or is this a form of procrastination for writing. If needed, the final output document (e.g. latex file) can be edited to achieve desired result. But in all likelihood, it isn't needed, so just move on.

3 Results

This method of manuscript preparation has a number of primary benefits:

- simplicity: fully separating content and presentation
- complete history: storing everything in git maintains full history of revisions
- flexibility: working with colleagues and journals in any desired file format

4 Conclusion

Authoring scientific manuscripts in markdown (MD) format is straight-forward, and manual formatting is reduced to a minimum. The simple syntax of MD facilitates document editing and collaborative writing. The rapid conversion of MD to multiple formats such as DOCX, LATEX, PDF, EPUB and HTML can be done easily using pandoc, and templates enable the automated generation of documents according to specific journal styles.

The additional features we implemented facilitate the correct indexing of meta information of journal articles according to the 'semantic web' philosophy.

Altogether, the MD format supports the agile writing and fast production of scientific literature. The associated time and cost reduction especially favours community-driven publication strategies.

Acknowledgments

We thank the project participants and maintainers for pandoc and pandoc-scholar.

References

- Benkler, Yochai. 2006. *The Wealth of Networks: How Social Production Transforms Markets and Freedom*. New Haven, CT, USA: Yale University Press.
- Suber, Peter. 2012. *Open Access*. Cambridge, Mass: The MIT Press.

5 Appendix

Pandoc allows you to easily include an appendix after the references. But getting the numbering right is tricky. This is the best simple solution, where the numbering is still there to enable references throughout the document, but it does not reset as expected. See section 5.2 for details on how to manually fix this in the latex file if needed for submission (Note: it probably isn't necessary most of the time).

5.1 Required software

Here is the small list of the software-dependencies. Bold items must be installed by you. Other items are installed automatically by this template:

- **pandoc** see <https://pandoc.org>
- **pandoc-scholar** Note that this template uses a modified fork of the main project to properly render a number of important document elements <https://github.com/aswolf/pandoc-scholar.git>
- **pandoc-scholar-template** repo containing this file
- *python*: this is required but is probably already installed
- *pandoc-xnos*: this is installed by the scripts in this project template

5.2 Manual Correction for Appendix Numbering

Renumbering appendix sections is not currently incorporated into pandoc. So the Appendix section numbers will continue from the main text in the resulting manuscript. This is fine for many uses, but manually fixing this prior to submission is a good idea. This can be done in the latex output file by surrounding the appendix content with the following two code blocks:

Before appendix:

```
\appendix
\renewcommand{\thesection}{\Alph{section}.\arabic{section}}
\setcounter{section}{0}

\begin{appendix}
\section{Appendix}
```

After appendix:

`\end{appendix}`

This will completely fix the numbering issue.