

The Malcontent Management System

Ben Klemens

January 10, 2015

Here comes the LaTeX header. It is cut/pasted/modified from another document, where it was cut/pasted/modified from another document, ..., meaning that it has a lot that could be excised.

First the ending, stored in m4 buffer 9, then we return to buffer 0 to start from the beginning.

We write technical and academic documents, and are not happy with typical content management systems.

- The math has to look good.
- We have to get the references in the right style.
- We're posting one possibly very long document, not a daily sequence of short entries.
- We need a PDF and web version of the document.
- We are collaborating on one document, and some of our collaborators may not know the intricacies of LaTeX or HTML.

The MMS is a wrapper around LaTeX to make these things easier. Document parsers/translators often don't produce good output without a ton of tweaking. Markdown is a pain for technical documents ("A * B * C" comes out as "A B C", for example). The solution here is to specify a short list of elements to mark section heads, citations, and so on, and expand those to the appropriate TeX or HTML as needed.

Requisites To compile the LaTeX and do the conversions using the script here, you will need one machine with LaTeX and basic POSIX compliance.

It is easiest to set up POSIX compliance via a package management system. Linux users know their package manager; Mac users can use MacPorts¹ or Fink²; Windows users can use Cygwin³. Each of these package managers will set up a full TeX environment in a few clicks, not to mention thousands of other programs, libraries, language compilers, and games. You will need:

- make
- a LaTeX installation

¹<http://macports.org>

²<http://finkproject.org>

³<http://cygwin.com>

- m4
- a C compiler (probably gcc or clang)

Are you a Github⁴ user? Github offers a simple means to host web pages, so if you have a github repository set up, the MMS has a hook to post to it.

1 The basic idea

We use m4 to convert certain blobs of text to LaTeX-formatted blobs of HTML-formatted blobs, depending on the output type. Anything that is not plain text or math has an associated m4 command.

Both LaTeX and HTML largely ignore the white space in your text file, so feel free to tab-indent whenever it's useful to you.

Math is standard TeX math. We use MathJax to render it in HTML on not-too-old browsers.

Both the LaTeX and HTML headers specify an encoding of UTF-8. On anything but Windows, this is the default; on Windows, please follow the custom of over 80% of the web⁵ and save as UTF-8.

1.1 Some commands

This section is a demo of the various commands, and doesn't make much sense as prose. Compare the source for this manual to the PDF or HTML output to see how the commands work.

1.1.1 Sectioning

There are Chapter, Section, Subsection, and Paragraph commands. If you need more depth than that you're probably trying too hard. The Paragraph is sometimes useful for marking a list of ideas within a section.

1.1.2 Text decoration et al.

changing typeface *Emphasis, probably italic* SMALL CAPS **boldface**, use sparingly
teletype, for computer code-like fragments in text

Linking text describing the url⁶

Citations The MMS was developed for a distant spinoff of a modeling textbook [Klemens, 2008]. It was also used for documenting Klemens [2012].

The second argument to Citep is text for the HTML version; BibTeX autogenerates the right text for the PDF.

⁴<http://github.com>

⁵http://w3techs.com/technologies/overview/character_encoding/all

⁶<http://example.com>

The content of the citations goes into a BibTeX database. See the example, or ask your search engine for hundreds of tutorials.

Lists

- bulleted lists look like this.
- They're unicode bullets. If you don't have an easy way to type them, you could cut/paste the bullets here.
 - Lists may be nested to arbitrary depth.

2 Production

As noted above, you will need a working copy of LaTeX, and a basic POSIX environment to compile the document from m4 to PDF or HTML.

Copy the `makefile` from this project directory (i.e., the `manual` subdirectory of the MMS) to your project directory. It has to be named `makefile` or `Makefile`. Open the `makefile` in your text editor and read the notes on setting the configuration variables.

When you are ready to compile the document, type `make` to generate the PDF, `make html` to generate the HTML, and `make html push` to generate the HTML and push it to Github.

It relies on a lot of search-and-replace via `sed`, `m4` for the above fixes, and a few other hacks. The bibliography is generated via BibTeX and then a `sed` script converts it to HTML.

References

Ben Klemens. *Modeling with Data: Tools and Techniques for Statistical Computing*. Princeton University Press, 2008.

Ben Klemens. Tea for survey processing. In *Proceedings of the Conference of European Statisticians*, Work Session on Statistical Data Editing. United Nations Economic Commission for Europe, September 2012.