set up the Emacs org and ox-hugo to export the org files in markdown and latex format.

1 Introduction

I use hugo before it support Emacs org file. Then I use an Emacs extension called ox-hugo which is a carefully crafted Org exporter back-end for Hugo. Ox-hugo meets all my requirement and definitely worth a try when you are an advanced user of Emacs Org. By the way, if you are quite familiar with Emacs Org, there is no need to take time to learn markdown. Emacs Org, I think, is more elegant than markdown.

2 set up ox-hugo

I use Spacemacs and integrate ox-hugo into my config is easy. You can add it into your private layer or just add it in your `user-config` . Please refer to the repo. and read the document for more information.

The first time you want to export an Org file or a subtree of an Org file, use `M-x org-hugo-export-to-md` . There will be a hugo-related section in the `org-dispatcher` whenever you use `C-c C-e` .

3 use Org Macro replacement

Often, you want different actions when you export the Org file to hugo-compatible markdown and latex file. For example, hugo or hugo theme supports summary and shortcodes such as `alert note` and `alert warning` (see Writing content with Markdown, LaTeX, and Shortcodes | Academic)

When you export the snippet to markdown then hugo will display the content according to the css. However, latex does not recognize the shortcodes. So latex will display `{ {% alert note %} }` literally which is clutter. The Org Macro may help. For more information please refer to The Org Manual: Macro replacement and fniessen/org-macros: Shared macros for Org mode . I also define my own Org Macro in my setup file for every .org file. With Org Macro, I control some contents and options exist only for certain backend.

Once you are familiar with Org Macro, It behaves beyond your imagination.

4 use the setupfile for each Org file

Often, you options for a org file grows more and more. If you think it kind of clutter at the top of your Org files, you can group them in a setupfile then include it at the top of your Org file.

For my setupfiles, please refer to my org-templates. As you can see, I have several setupfiles for different aims. To include one of them in my Org file, I use:

```
#+SETUPFILE: <path+name of the setup files>
```

By using `#+SETUPFILE` ,I only have to option lines at the top of my org file. So it looks clean.

Also, you can notice that in my setupfile named enpost.org exists options for different export targets.