

# 纯文本做笔记 — 使用 Pandoc 与 Markdown 实践

2017-12-10 00:10

本文主要写如何使用 Pandoc 从 Markdown 文件生成格式精美的 PDF 文件，也包含了一些遇到的问题的解决办法。

## 你需要什么

- 安装 Pandoc，下载[最新的安装包](#)，安装完把 Pandoc 安装目录加入系统 PATH 变量。
- TeX 发行版。请确保你的系统已经安装了 TeX 软件，你可以使用 [TeX Live](#) 或者 [MiKTeX](#)，安装完成之后可能需要设置 PATH，推荐安装 TeX Live。
- 一个强大的文本编辑器，当然首推 [Sublime Text](#)。当然了，[Visual Studio Code](#) 和 [GitHub Atom](#) 也是不错的选择。

## 使用 Pandoc 从 Markdown 生成 PDF 文件

[Pandoc](#) 可以很方便地在不同 Markup 语言之间进行格式转换，被誉为格式转换的“瑞士军刀”。使用 Pandoc 把 Markdown 文件转为 PDF 实际上有两个步骤：第一步，.md 文件被转为 .tex 文件，第二步，调用系统的 pdf<sub>l</sub>atex 或者相关 TeX 命令从 .tex 文件生成最终的 PDF 文件 (见上图)。由于我的文档是中文，并且包含引用，表格等比较复杂的格式，因此遇到了一些问题，下面介绍具体解决办法。

## 如何处理中文

Pandoc 默认使用的 pdf<sub>l</sub>atex 命令无法处理 Unicode 字符，因此在生成 PDF 的过程中会报错，我们需要使用 xelatex 来处理中文。并且，需要指定支持中文的字体。在 Windows 系统中，对于 pandoc 2.0 版本以上，可以使用以下的命令：

```
pandoc --pdf-engine=xelatex -V CJKmainfont="KaiTi" test.md -o test1.pdf
```

CJKmainfont 后面跟的是支持中文的字体名称。如何找到支持中文的字体？首先需要知道所使用的语言的 [language code](#)，Chinese 的 language code 是 zh。然后使用 fc-list 查看支持中文的字体 (对于 Windows 系统，fc-list 命令在安装 TeX Live 完整版以后可以使用，Unix 系统一般会预装这个程序)：

```
fc-list :lang=zh # zh 是 Chinese 的 language code
```

系统输出如下图所示

字体的名称就是字体文件位置后面的字符串，由于字体名称很多包含空格，在引用的时候需要加上引号，例如 -V CJKmainfont="Source Han Serif CN"。

在 Pandoc 2.0 版本中，[上述 --pdf-engine 命令](#)取代了原有的 [--latex-engine 命令](#)。对于 Linux 系统，Pandoc 版本可能比较老，上述生成 PDF 的命令可能并不适合，正确的命令如下 (在 Pandoc 1.12.3.1 验证)：

```
pandoc --latex-engine=xelatex -V mainfont='WenQuanYi Micro Hei' test.md -o test.pdf
```

# 从 Markdown 生成 PDF 遇到问题及解决方法

## block quote, table 以及 list 未能正确渲染

原因是在 Pandoc 中 block quote, list 以及 table 前需要空一行。另外 block quote 中每一行渲染成 PDF 未能正确换行, 所有行的文字都跑到了一行, 可以通过强制在原 block quote 每一行后面加上空格来解决。

## 给 code block 加上 highlight

Pandoc 支持给 code block 里面的代码进行高亮, 提供了不同的高亮主题, 并且支持非常多的语言。要列出提供的高亮方案, 使用下面命令,

```
pandoc --list-highlight-styles
```

要列出所有支持的语言, 使用下面命令,

```
pandoc --list-highlight-languages
```

要使用语法高亮, Markdown 文件中的 code block 必须指定语言, 同时在命令行使用 --highlight-style 选项, 例如

```
pandoc --pdf-engine=xelatex --highlight-style zenburn -o test.pdf test.md
```

以上命令, 使用了 zenburn 主题, 推荐使用 tango, zenburn 或者 breezedark 高亮主题。

## 给 section 加上编号以及给文章加上目录

默认情况下, 生成的 PDF 不含目录, 同时各级标题不含编号, 仅仅字体大小有变化, 要给各个 section 加上编号, 可以用 -N 选项; 加上目录, 可以使用 --toc 选项。一个完整例子如下:

```
pandoc --pdf-engine=xelatex --toc -N -o test.pdf test.md
```

## 给 URL link 以及 pure URL 加上颜色

根据 Pandoc 官方教程以及 [这个 issue](#), 在命令行指定选项: -V urlcolor=cyan, 该选项可以给 URL link 加上颜色, 但是并不能给文中直接展示的 URL link 加上颜色。要给直接展示的 URL link 加上颜色, 需要使用选项 -f gfm, 参考[这里](#)。完整命令如下,

```
pandoc --pdf-engine=xelatex -f gfm -Vurlcolor=cyan -o test.pdf test.md
```

## 更改 PDF 的 margin

使用默认设置生成的 PDF margin 太大, 根据 [Pandoc 官方 FAQ](#), 可以使用下面的选项更改 margin:

```
-V geometry:"top=2cm, bottom=1.5cm, left=2cm, right=2cm"
```

完整命令为:

```
pandoc --pdf-engine=xelatex -V geometry:"top=2cm, bottom=1.5cm, left=2cm, right=2cm" -o test.pdf test.md
```

## Github flavored Markdown 不支持公式生成

使用 -f gfm 的一个缺点是 [gfm 不支持公式](#), 因此如果在 Markdown 中包含公式, 将不能正确渲染。解决办法是去掉 -f gfm flag, 或者使用 Pandoc 自带的 markdown 格式。

## Markdown 文件中使用 backslash 会出错

原始的 Markdown 格式，支持在文件中使用 backslash，但是 Pandoc 把 backslash 以及后面的内容理解成 LaTeX 命令，因此在编译包含 backslash 的文件时，可能会遇到 `undefined command` 错误或者更加奇怪的错误。参考[这里](#)以及[这里](#)，解决办法是让 Pandoc 把 Markdown 文件当成正常的 Markdown，不要解读 LaTeX 命令，使用下面的 flag:

```
pandoc -f markdown-raw_tex
```

或者也可以用两个 backslash 表示字面意义的 backslash，例如 `\\columnwidth` 或者如果 backslash 及后面文字原本就是一个命令，用 inline code block 包含起来。

## 使用 Sublime Text build system 生成 PDF

写完 Markdown 再切换到 console 使用 Pandoc 生成 PDF 以及预览颇为麻烦，因此我使用了 Sublime Text 的 build system 来完成编译生成 PDF 以及预览的整个过程。PDF 预览使用了轻量级的 [Sumatra PDF](#)。一个示例的 build system 如下，

```
{
  "shell_cmd": "pandoc --pdf-engine=xelatex -f gfm --toc --highlight-style zenburn -Vurlcolor=cyan -V CJKm",
  "file_regex": "^(..[^:]*):([0-9]+):?([0-9]+)??:? (.*)$",
  "working_dir": "${file_path}",
  "selector": "text.html.markdown",

  "variants":
  [
    {
      "name": "Convert to PDF and Preview",
      "shell_cmd": "pandoc --pdf-engine=xelatex -f gfm --toc --highlight-style zenburn -Vurlcolor=cyan -V CJKm"
      // "shell_cmd": "start \"$file_base_name\" call $file_base_name"
    }
  ]
}
```

在命令中使用到了引号，必须使用 backslash escape。可以从[这里](#)下载这个 build system。

## 系统无法识别 Pandoc 的问题

不知什么原因，前几天还用的好好的上述把 Markdown 文件转为 PDF 的 build system 在编译 Markdown 文件时，突然给出如下提示：>'pandoc' is not recognized as an internal or external command, operable program or batch file.

通过查询 [sublime 文档](#)，发现我们可以在 build system 里面加上 path 变量，因此我对上述的 build system 进行了调整，新的 build system 如下：

```
{
  "shell_cmd": "pandoc --pdf-engine=xelatex -f gfm --highlight-style zenburn -Vurlcolor=NavyBlue -V CJKm",
  "path": "C:/Users/east/AppData/Local/Pandoc/;%PATH%",
  "file_regex": "^(..[^:]*):([0-9]+):?([0-9]+)??:? (.*)$",
  "working_dir": "${file_path}",
  "selector": "text.html.markdown",

  "variants":
  [
    {
      "name": "Convert to PDF and Preview",
      "shell_cmd": "pandoc --pdf-engine=xelatex -f gfm --highlight-style zenburn -Vurlcolor=NavyBlue",
      "path": "C:/Users/east/AppData/Local/Pandoc/;%PATH%",
    }
  ]
}
```

```
        // "shell_cmd": "start \"${file_base_name}\" call ${file_base_name}"
    }
}
}
```

使用新的 build system 以后，一切恢复正常。

## 云同步

云同步可以选择[坚果云](#)。同步速度快，并且支持多个文件夹同步，每个月有 1G 免费上传流量，对于文档同步来说，足够使用了。

## 参考

- [Pandoc 处理中文](#).
- [Pandoc 对 block quote 的处理](#).
- [Pandoc 语法高亮设置](#).
- [Pandoc 设置 section number](#)
- [解决 pandoc 命令找不到的问题](#).