

文档编号：EPPDEV-PANDOC-TEMPLATE-USAGE

EPPDEV-PANDOC-TEMPLATE 使用指南

V1.3

郝金隆



EPPDEV.CN

2019-09

修订记录

序号	修订人	修订时间	修订记录
V1.0	郝金隆	2019-09-25	1. 创建文档 2. 完成第一版的文档说明
V1.1	郝金隆	2019-09-26	增加了修订目录相关内容
V1.2	郝金隆	2019-09-28	1. 修改一级目录引导线的格式, 增加点间距 2. 修改修订目录的格式, 去掉内容空行, 标题粗体
V1.3	郝金隆	2019-10-13	1. 增加了与 DevOps 的自动化处理脚本 2. 增加了与 DevOps 集成的环境配置说明

目录

1 工程说明	1
2 基础环境配置	1
2.1 Pandoc 的安装	1
2.2 Latex 的安装	1
3 安装后的设置	1
3.1 配置 Pandoc 模板	1
3.2 模板的定制	2
3.3 字体设置	3
4 pdf 文件的生成	3
4.1 PDF 文件指定 metadata 信息	3
4.2 Markdown 其他编写要求	4
4.3 文件生成	5
5 与 DevOps 集成	5
5.1 发布与编译环境准备	5
5.1.1 安装 Pandoc	5
5.1.2 安装 TexLive	5
5.1.3 安装 template 文件	6
5.1.4 安装字体	6
5.1.5 配置目标目录	6
5.1.6 安装 pandoc_deploy.sh	6
5.2 DevOps 的配置	7
5.2.1 主要配置方式	7
5.2.2 gitlab-ci 配置说明	7
5.3 下一步的定制	8
6 引用参考	8
7 许可	8
8 附录	9
8.1 Markdown 概述	9
8.2 Markdown 写作环境说明	9

8.3	配置 http 环境的安全认证	9
8.3.1	首先安装 httpd-tools :	9
8.3.2	然后创建密码文件	10
8.3.3	修改 nginx 配置	10

1 工程说明

本工程是个人根据实际需要参考相关网络文档，整理的通过 pandoc 转换为 pdf 的模板，用于实现内部 IT 项目管理文档格式的标准化工作。

2 基础环境配置

markdown 生成 pdf 主要需要使用 Pandoc 和 Latex(texlive/miktex) 两个工具，具体安装方式如下：

2.1 Pandoc 的安装

Pandoc 是由 John MacFarlane 开发的标记语言转换工具，可实现不同标记语言间的格式转换，堪称该领域中的“瑞士军刀”。Pandoc 使用 Haskell 语言编写，以命令行形式实现与用户的交互，可支持多种操作系统。

- Window 下的安装：下载[安装包](#)直接安装即可
- Linux/FreeBSD 下的安装：Pandoc 已经包含在大部分 Linux 发行版的官方仓库中，直接使用诸如 apt/dnf/yum/pacman 之类的安装工具直接安装即可
- MacOS 下的安装：建议通过 HomeBrew 进行安装即可

详细的安装说明参见：[官方安装文档](#)

2.2 Latex 的安装

latex 工具，在 windows 下建议安装 miktex，Linux 和 MacOS 下建议安装 texlive

- Windows 下的安装：下载安装[miktex](#)，注意安装后需要再安装 cjk,cjk-fonts 等相关 package
- Linux/FreeBSD 下的安装：使用 apt/dnf/yum/pacman/pkg 等安装工具安装 texlive、texlive-latex 等相关软件包即
- MacOS 下的安装：使用 HomeBrew 安装 texlive 即可

3 安装后的设置

3.1 配置 Pandoc 模板

为保证生成的 pdf 格式（自动插入封面、目录页、页眉页脚等信息），在本地环境中安装模板，具体步骤是：

- 下载[eppdev-pandoc-template](#) 模板 将其保存到本地
- 根据需要对模板进行相应的定制，具体定制方式参见下节
- 将修改后的模板文件复制到指定目录下¹
 - Window 下：C:/Users/USERNAME/AppData/Local/roaming/pandoc/templates
 - Linux/FreeBSD/MacOS： ~/.pandoc/templates/

3.2 模板的定制

模板定制主要修改模板最前面的 “O、模板基础配置” 相关内容，主要可修改的包括：

1. 公司和组织，目前默认是 “EPPDEV.CN”
2. 正文缩进，目前默认是 2em (2 个中文字符，4 个英文字符)
3. 主要中文字体和英文字体：目前都是微软雅黑
4. 页眉、页脚展示内容，目前是：
 1. 左页眉：title
 2. 右页眉：“企业机密，请勿外传”
 3. 作页脚：company
 4. 右页脚：页码

eisvogel 模板原来右页眉默认是 date，如需恢复为 data，只需要将 “EPPDEV.CN” 改为 \$date\$ 即可

现有内容如下：

```
1
2 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
3 %
4 % O、模板基础配置
5 %
6 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
7
8 % 公司和组织
9 \newcommand*{\company}{\if(company)$$company$$else$EPPDEV.CN$endif$}
10
11 % 缩进
12 \newcommand*{\udfparindent}{2em}
13
14 % 主要中文字体
```

¹如有不同需要，可以定制不同的模板，作不同的命名，复制到对应目录下

```
15 \newcommand*{\thecjkmainfont}{\if(CJKmainfont)$$CJKmainfont$$else$
    Microsoft YaHei$endif$}
16
17 % 主要英文字体
18 \newcommand*{\themainfont}{\if(mainfont)$$mainfont$$else$Microsoft
    YaHei$endif$}
19
20 % logo
21 \newcommand*{\thelogo}{\if(logo-url)$$logo-url$$else$logo.png$endif$}
22
23 % 页眉-左
24 \newcommand*{\headerleft}{\if(header-left)$$header-left$$else$$title$$
    endif$}
25
26 % 页眉-右
27 \newcommand*{\headerright}{\if(header-right)$$header-right$$else$企业机
    密，请勿外传$endif$}
28
29 % 页脚-左
30 \newcommand*{\footerleft}{\if(footer-left)$$footer-left$$else$\company$
    endif$}
31
32 % 页脚-右
33 \newcommand*{\footerright}{\thepage}
```

3.3 字体设置

目前页面默认的字体是微软雅黑，对于非 Windows 系统，可能不存在该字体，则有以下两种解决方案：

1. 手工安装微软雅黑字体（需要 msyh,msyhbd 两个文件）
2. 修改为其他字体，如苹方、文泉驿等

若需要多个团队共同使用，建议采用方案一。

4 pdf 文件的生成

4.1 PDF 文件指定 metadata 信息

在每个 markdown 最前面增加以下主要 metadata 信息，metadata 内容开始行内容为三个“-”，结束行为三个“.”，示例如下：

```
1 title: EPPDEV-PANDOC-TEMPLATE使用指南
2 version: 1.0
3 author: 郝金隆
4 date: 2019-09
5 file-code: EPPDEV-PANDOC-TEMPLATE-USAGE
6 history:
7   - version: V1.0
8     author: 郝金隆
9     date: 2019-09-25
10    desc:
11      - 1. 创建文档
12      - 2. 完成第一版的文档说明
13   - version: V1.1
14     author: 郝金隆
15     date: 2019-09-26
16     desc: 增加了修订目录相关内容
```

其他可选配置项目如下：

1. subtitle: 副标题
2. header-left: 左页眉
3. header-right: 右页眉
4. footer-left: 左页脚
5. footer-right: 右页脚
6. company: 公司名称
7. CJKmainfont: 主要中文字体
8. mainfont: 主要字体
9. lot: 是否创建表格目录
10. lof: 是否创建图片目录
11. logo: 是否在封面显示 logo (需要在 markdown 相同文件夹下有 logo.png 文件)

可选配置项中，建议除了 subtitle 外，全部在模板中定制，不在 markdown 文件中定制

4.2 Markdown 其他编写要求

pandoc 默认使用的pandoc_markdown格式，为避免 markdown 转 pdf 格式异常，在编写 markdown 的时候有几个原则要求：

- 每个标题前后都必须有空行
- 每个表格前后都必须有空行

- 每个代码块前后必须要有空行
- 每个列表前后必须要有空行

总而言之，每个不同的格式和内容前后都需要有空行，详细内容参见 [pandoc 官方文档](#)

4.3 文件生成

配置完成后即可在通过 pandoc 命令生成 pdf 文件：

```
1 pandoc --listings --pdf-engine=xelatex --template eppdev-doc a.md -o a.pdf
```

若定制的模板修改了文件名，需要将命令中的 eppdev-doc 修改为修改后的文件名

5 与 DevOps 集成

本 pandoc 模板可以很容易的与 gitlab-ci 等 devops 工具相集成，实现 pdf 文件的自动生成，这样就无需每个人都安装一套 pandoc, texlive 等环境，下文以 gitlab 为例，说明如何与 DevOps 进行集成

5.1 发布与编译环境准备

首先需要再 DevOps 的执行节点进行环境配置²，要配置的环境包括：

5.1.1 安装 Pandoc

安装方式可参考上文，注意很多服务器操作系统 (如 CentOS) 官方库的 pandoc 还是 1.x 版本，使用该版本将不支持修订目录功能，建议手工安装最新的 pandoc，具体可以通过官网下载最新版本，解压缩以后，设置 PATH 环境变量即可

5.1.2 安装 TexLive

安装方式同样可以参考上文，但是服务器操作系统版本一般相对比较第，建议手工安装最新版本，可以从国内镜像如[华为](#)、[清华](#)等站点下载最新的 iso 文件，当前最新为 2019。具体安装命令如下：

²是指具体进行编译执行的环境，如 gitlab-ci 中的 gitlab-runner 所在的服务器

```
1 wget https://mirrors.tuna.tsinghua.edu.cn/CTAN/systems/texlive/Images/texlive2019.iso
2 sudo mount ./texlive2019.iso /mnt
3 cd /mnt
4 sudo ./install_tl
```

安装完成后，需要将修改环境变量，将最新的 texlive 目录下的 bin 目录添加到环境变量中

5.1.3 安装 template 文件

根据需要，将 eppdev_doc.latex 文件复制到进行编译执行的服务器用户主目录 (如 gitlab-ci 对应的 /home/gitlab-runner) 下的.pandoc/templates 目录下

5.1.4 安装字体

具体需安装的字体根据 eppdev_doc.latex 中的配置为准，本文以微软雅黑为例。将 msyh.ttf, msyhbd.ttf 两个文件上传到服务器上，然后进行安装：

```
1 sudo mkdir /usr/share/fonts/chinese
2 sudo cp msyh*.ttf /usr/share/fonts/chinese
3 cd /usr/share/fonts/chinese
4 sudo mkfontscale
5 sudo mkfontdir
```

5.1.5 配置目标目录

创建一个要发布的目标文档目录，提供编译执行者可读写的权限，以 gitlab 为例：

```
1 sudo mkdir /srv/doc
2 sudo chown gitlab-runner /srv/doc
```

配置好目录以后后续即可通过 nginx/httpd 的文件列表功能，或者通过 rsync 同步到对应的文件管理服务器中，本文对此就不再赘述

5.1.6 安装 pandoc_deploy.sh

将工程下载的 pandoc_deploy.sh 文件进行修改，将 target_dir 修改为上文创建的目标目录。然后将其复制到/usr/bin 目录下，并为其添加可执行权限

```
1  chmod a+x pandoc_deploy.sh
2  sudo cp pandoc_deploy.sh /usr/bin/
```

5.2 DevOps 的配置

5.2.1 主要配置方式

环境安装好了以后，即可在 DevOps 的环境中定义任务，完成自动的 pdf 生成工作，具体是的调用命令为：

```
1  /usr/bin/pandoc_deploy.sh 本工程对应的目录
```

注意：此处的目录为针对本工程的代码对应的所有目录

5.2.2 gitlab-ci 配置说明

在仓库主目录下创建.gitlab-ci.yml 文件，示例内容如下：

```
1  stages:
2    - deploy
3
4
5  doc-deploy:
6    stage: deploy
7    script:
8      - /usr/bin/pandoc_deploy.sh tmp_doc
9    tags:
10     - pandoc
```

上述配置主要说明如下：

1. stages: deploy, 是指本工程编译只有一个 stage
2. doc-deploy: 是指定义的一个任务，其所属的 stage 为 deploy
3. /usr/bin/pandoc_deploy.sh tmp_doc 是指本工程生成的 pdf 将复制到/srv/doc/tmp_doc 目录下³
4. tags:pandoc, 是因为环境中有多多个 gitlab-runner，仅有该标签的 runner 配置了 pandoc 环境

³之所以是这个目录是因为 pandoc_deploy.sh 中的 target_dir 为/srv/doc

5.3 下一步的定制

可以根据需要对 `pandoc_deploy.sh` 进行修改，通过 `rsync` 将其与其他文档管理系统进行同步。当然，若只是简单的通过 `httpd` 或者 `nginx` 进行文件展示，则该文件无需修改，但是注意至少需要通过 `htpasswd` 进行权限验证，以避免企业机密的外泄！

6 引用参考

本工程主要参考包括：

- [Eisvogel 模板](#)：参考 [Eisvogel](#) 模板，针对国内环境的需要，做了个性化定制，主要取便在于
 - 指定了字体为微软雅黑，默认解决中文问题
 - 指定目录编号层级为 5 级
 - 解决了 4、5 级目录的格式问题，确保目录后有换行
 - 定制了页眉页脚内容
 - 增加了 `version` 属性，展示在封面上
 - 增加了 `file-code` 文档编号属性，用于在封面上的展示
 - 增加了 `company` 可选（默认在模板中设置，可以手工修改）属性，展示在封面和页脚上
 - 修改了页码格式
 - ...
- [formal-boot-title-page](#)：参考该模板修改了封面格式，主要修改点
 - 增加文档编号
 - 调整了 `author` 的展示
 - `publisher` 改为 `company`
 - 日期调整到 `company` 下

7 许可

版权所有：2019，[郝金隆](#)

软件许可：[ANTI-996 License v1.0](#)

8 附录

8.1 Markdown 概述

Markdown 是一种轻量级标记语言，创始人为约翰·格鲁伯（英语：John Gruber）。它允许人们“使用易读易写的纯文本格式编写文档，然后转换成有效的 XHTML（或者 HTML）文档”。这种语言吸收了很多在电子邮件中已有的纯文本标记的特性。

由于 Markdown 的轻量化、易读易写特性，并且对于图片，图表、数学式都有支持，当前许多网站都广泛使用 Markdown 来撰写帮助文档或是用于论坛上发表消息。例如：GitHub、reddit、Diaspora、Stack Exchange、OpenStreetMap、SourceForge 等。甚至 Markdown 能被用来撰写电子书。

Markdown 语法参见：[Markdown 语法介绍 \(coding\)](#)、[Markdown 基础介绍 \(简书\)](#)

8.2 Markdown 写作环境说明

Markdown 的规范本身就是文本文档，故任何文本文件编辑器均可进行 markdown 文件的编辑，编辑 markdown 文件可以使用专业的编辑器，也可以在 IDE 环境中安装 markdown 的插件。

目前业内常用的 markdown 文件编辑器主要包括：

- sublime
- remarkable
- markdownpad
- Typora

IDE 环境：

- IntelliJ IDEA：可以安装[markdown-navigator](#)插件或者[markdown](#)插件
- Eclipse: 可以安装[markdown-text-editor](#)插件
- VSCode: 可以安装[markdown](#)插件

8.3 配置 http 环境的安全认证

以 CentOS 环境下的 nginx 为例

8.3.1 首先安装 httpd-tools：

```
1 sudo yum install httpd-tools
```

8.3.2 然后创建密码文件

```
1 sudo mkdir /etc/nginx/pass
2 sudo htpasswd -cb /etc/nginx/pass/mypass eppdev 123
```

上述命令的作用是在该目录下创建一个 mypass 文件，用户名是 eppdev，密码是 123

8.3.3 修改 nginx 配置

在/etc/nginx/nginx.conf 文件中添加以下配置

```
1     location /doc/ {
2         alias /srv/doc/ ;
3         auth_basic "Input Password:";
4         auth_basic_user_file "/etc/nginx/pass/mypass";
5         autoindex on;
6         autoindex_exact_size off;
7         autoindex_localtime on;
8     }
```

上述代码的说明：

1. 前两行是指创建一个虚拟的链接子目录，映射到/srv/doc/目录下，即访问 http://host:port/doc/即为对应的/srv/doc 目录
2. 第 3，4 行是指使用的是基础的认证机制，认证文件即为上文创建的密码文件
3. 第 5,6,7 行是指启动文件夹文件索引功能，直接通过链接可以访问目录下的文件