如何对 Word 文件进行「纯文本式」版本控制？

**Second 2 测试版本2**

**Git & Pandoc**

Git 最初是被用来对软件源代码进行版本控制，对 .docx 这样的二进制文件的支持不太友好。但是借助 Pandoc，我们可以实现「曲线救国」，通过将 .docx 文件转换为 .md 文件，来实现 Git 对 .docx 进行「纯文本式」版本控制。

**安装 Git 与 Pandoc**

* 前往 [Git 官网](https://sspai.com/link?target=https%3A%2F%2Fgit-scm.com%2Fdownloads)，选择对应的操作系统版本，下载安装 Git。
* 前往 [Pandoc 官网](https://sspai.com/link?target=https%3A%2F%2Fpandoc.org%2Finstalling.html)，下载安装对应系统版本的 Pandoc。

如果你的操作系统是 macOS，推荐使用 [Homebrew](https://sspai.com/link?target=https%3A%2F%2Fbrew.sh%2F) 来安装 Git 和 Pandoc：

brew install git pandoc

**配置 Git**

在用户文件夹下找到文件 .gitconfig，使用文本编辑器打开它，加入下面的内容，然后保存。

[diff "pandoc"]  
  textconv=pandoc --to=markdown  
  prompt = false  
[alias]  
  wdiff = diff --word-diff=color --unified=1

新建一个文件夹，这里命名为 git-word-demo。

# 进入 git-word-demo 文件夹  
cd git-word-demo  
​  
# 创建 Git 仓库  
git init

在 git-word-demo 目录下，新建一个文件 .gitattributes，写入下面的内容，然后保存。

\*.docx diff=pandoc

**编辑 Word 文件**

git-word-demo 文件夹下，新建一个 Word 文件，这里命名为 example.docx，然后按照平常一样进行编辑，完成之后保存并提交，进入该目录，在终端中输入：

git wdiff example.docx

就可以像纯文本一样，看到增加或删除的内容。当然，使用 Git 的 GUI 客户端会更加直观，比如 [GitHub Desktop](https://sspai.com/link?target=https%3A%2F%2Fdesktop.github.com%2F)，如下图所示，绿色表示增加的部分，红色表示减少的部分。