

番外篇 01: Git版本控制与Modelica模型开发

作者：航佛儿

课程地址: <https://github.com/hyumo/ModelicaPrimer>

(*本期内容有些整活成分，不喜勿喷，本集超纲不考，完全可以跳过)

主要内容

- 版本控制及版本控制工具git
- 如何利用git提高Modelica模型开发效率

你是否在模型开发中遇到过下面这些情况？

- 情况1：模型吭哧吭哧搞了一个月，头发也掉一桌子了，月底发现"哎呀，其实还是月初那个版本好"，一顿操作猛如虎，但就怎么也恢复不回去了。
- 情况2：想尝试一些新的idea，又不舍得删老代码，复制->粘贴->重命名，一顿操作猛如虎，最后新想法也不行，老代码也找不到了。（你的文件夹是不是长这样？
`foo.mo, foo_v1.mo, foo_v1_new.mo, foo_v1_backup.mo, foo_大猛子ym.mo`）。
- 情况3：一不小心就把整个库文件夹给shift+delete了，也没备份，有备份版本也太旧了，已经在准备跑路了。
- 情况4：组内总是通过QQ微信伊妹儿附件的方式交换代码，搞着搞着就整乱套了，最后谁也不知道自己运行的是哪个版本。别人的模型连上你的跑不通，你的连上人家的也不编译，一顿操作猛如虎，最后你把前两天自己发给别人的模型在机器上一跑，结果也不一样了。

如果你遇到过以上这些情况，那么你需要#版本控制！

何为版本控制

版本控制 是一种记录一个或若干文件内容变化，以便将来查阅特定版本修订情况的系统。<https://git-scm.com/book/zh/v2/起步-关于版本控制>

- 其中git是现在最流行的版本控制工具。（别的都是辣鸡，用git就完事儿了）
- git可以解决上面提到的所有问题，一针就见效。

git的具体用法——我就不细说了

大家现在可以先点个赞，然后再点 **暂停**，去其他家up主学习一下git的使用方法再回来，**不用学太多**，下面这些命令会用就行，自己建个仓库玩一玩，想速成可以试着删一下公司或者学校的库。

- `git init` : 初始化本地仓库
- `git clone` : 克隆远程仓库内容
- `git add` : 将新文件或更改的文件内容加入到本地staging区
- `git commit` : 本地提交更改的内容
- `git push/pull/fetch` : 推，拉，拽远程仓库内容
- `git checkout` : 切换到某个commit或者某个branch
- `git merge/rebase` : 将当前branch与另一个branch合并或rebase(不会翻译)

使用git提高模型开发效率

(现在我假设你已经git入门了，如果你还不会用，先看着，之后再回来再体会一下)

- 如何分阶段记录模型开发的流程？
- 如何通过创建branch快速试验新想法且保留原模型不变？
- 如何回到历史中某一次提交的代码？
- 如何通过git与你的队友愉快地合作？（大型同性社交网站github了解一下）

