THE UNIVERSITY OF MAINE

# Reproducible & Collaborative Research

Dudu Meireles
jose.meireles@maine.edu

# Research today…

is ***collaborative***!

must be ***reproducible***!

is **complex**! Big data & Large models

# Code must be…

be ***clearly written*** and ***structured***

must run anywhere, i.e. ***self contained***

Safely stored, ***version controlled***

***Accessible*** to collaborators all anywhere

Argued over. ***Science is social***

# Solutions

- **Structured R projects & Studio**
  - Set *paths relative* to where the .Rproj file is
  - Be *consistent* with your code style
  - Think modularly
  - Structure your project

- **Git and Github**
  - ***Version control*** and "*backup*" your data / code
  - ***Share*** your project with collaborators / world
  - ***Comment*** on code, **discuss** issues and ideas directly on GitHub

# Code style and structure

# To master project Reproducibility

- **Study and follow a style guide:** https://style.tidyverse.org/
- **Comment your code!** Try to convey intent.
- Set *file paths **relative*** to where the **.Rproj** file is
- **Structure your scripts:**
    1. Load packages
    2. Read data
    3. do work
    4. Save work

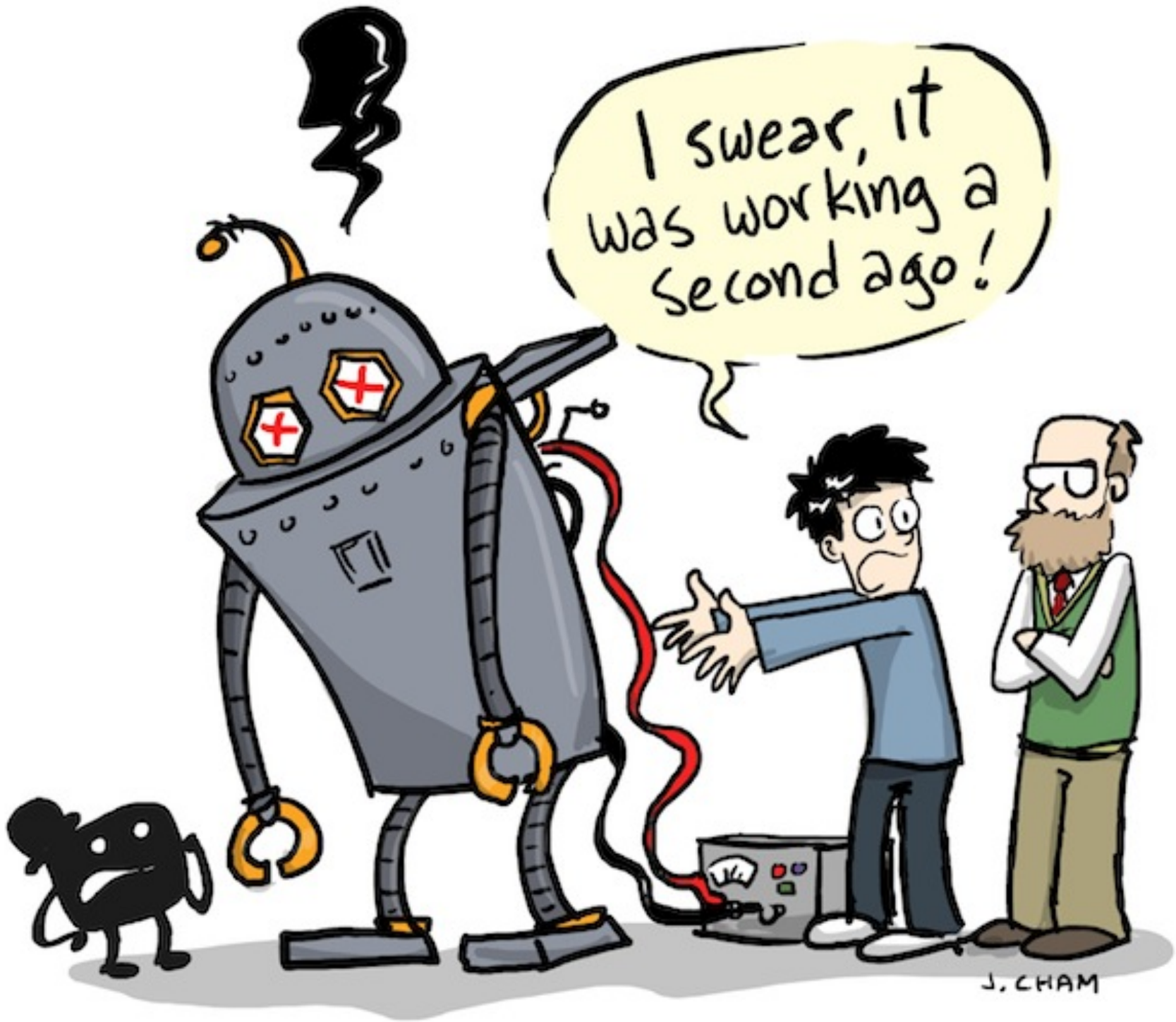
# Example

# How do you get that example project?

## Using git and Github!

# Version control

# Why do we need it?


I swear, it was working a second ago!
J. CHAM

# Why Version Control?

**Have a record of the history of your project (code, manuscript, data), along with metadata (who, when, what)**

1. Go back to a version that was working

2. Try new things without fear of breaking stuff

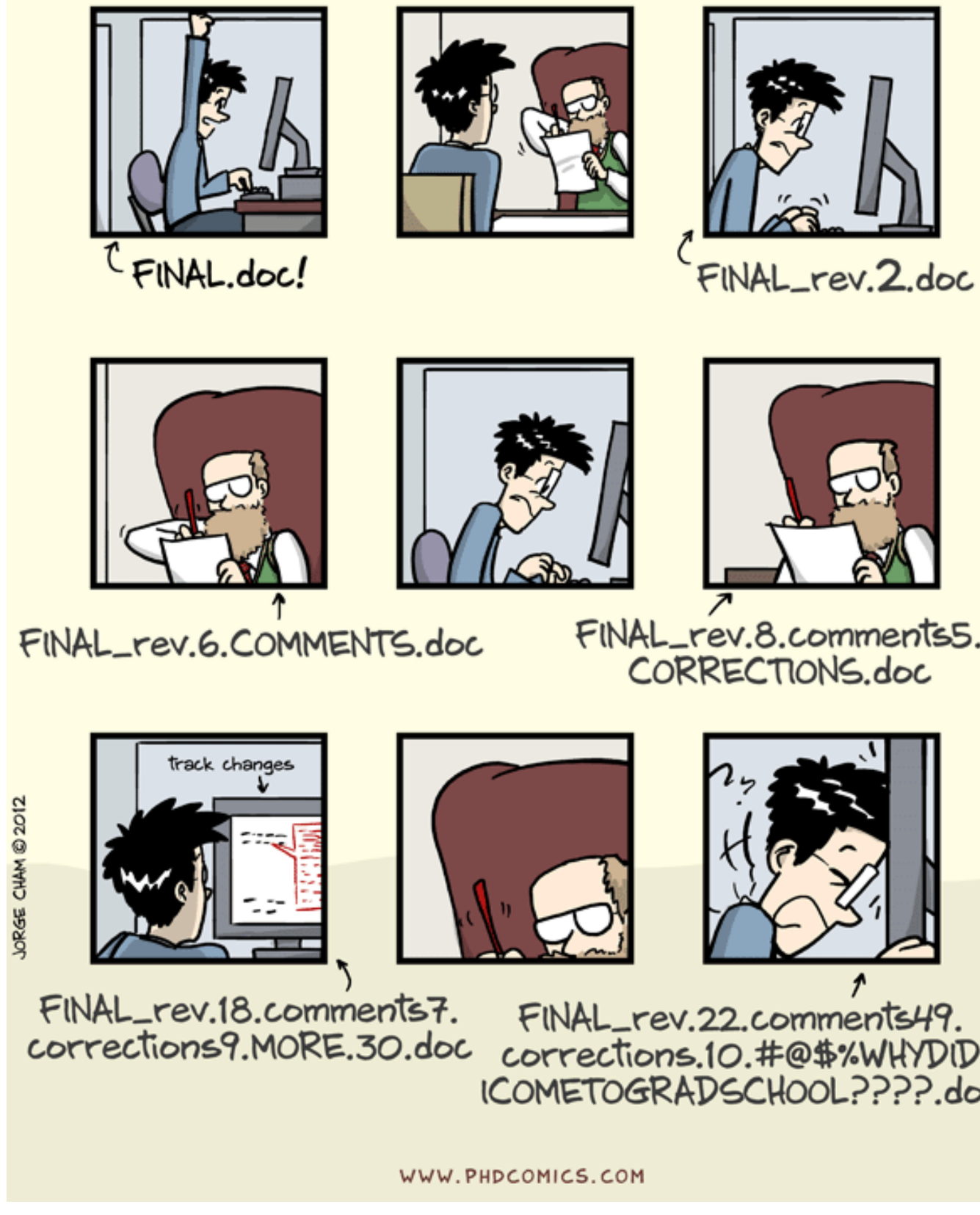
3. *Collaborating can be dangerous!* Prevent collaborators from messing your work

4. Allow you and other people to see *what*, *when* and *who* changed something.

# You already use it!

- Does this file renaming scheme seem familiar ***my_script_v1.R***, ***script_v2.R***, …?

# "FINAL".doc



# You already use it!

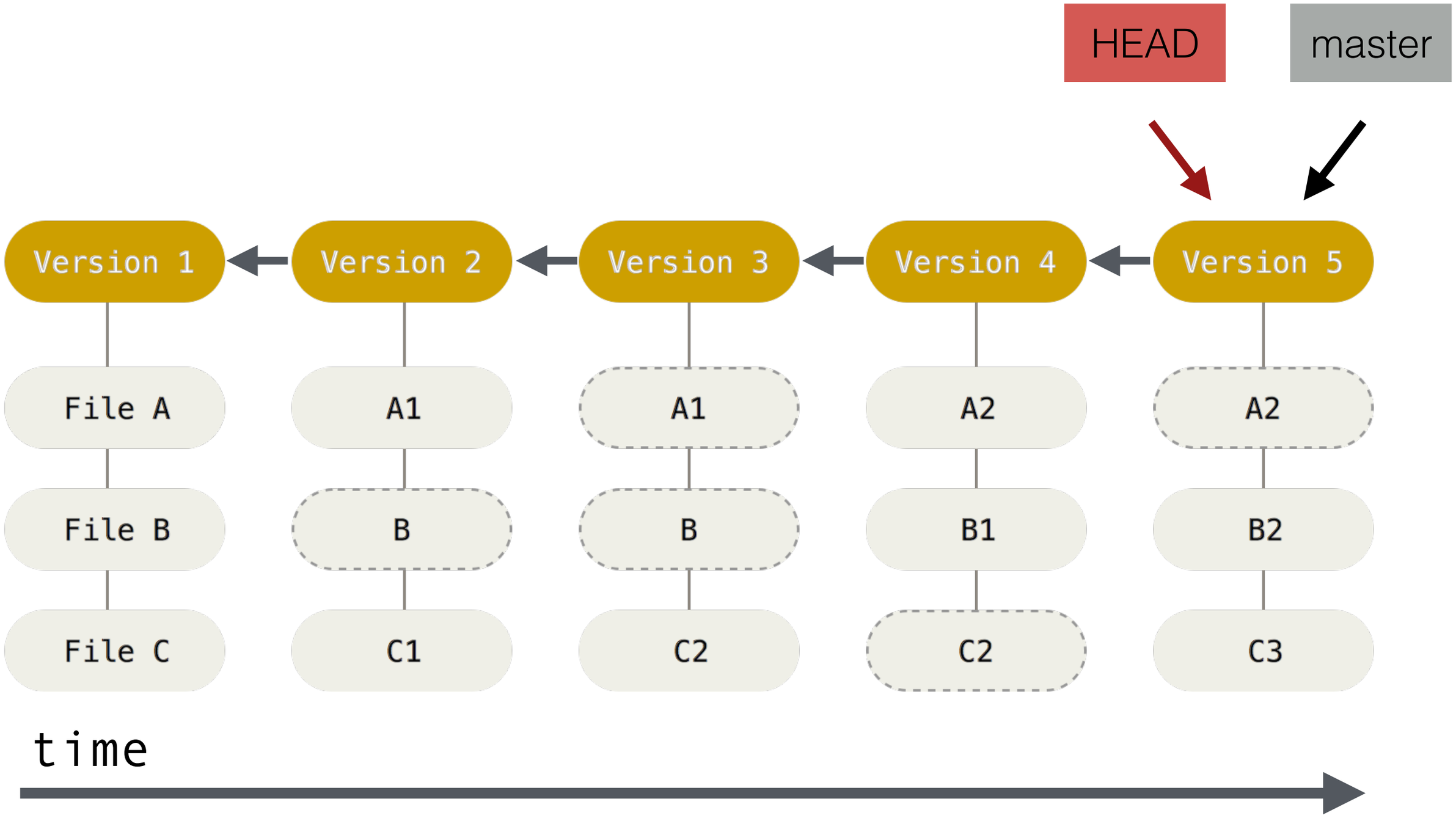
- Does this file renaming scheme seem familiar ***my_script_v1.R***, ***script_v2.R***, …?
- Go back button
- Word track changes

But you can do better…

by using **Git**!

# How does Git work?

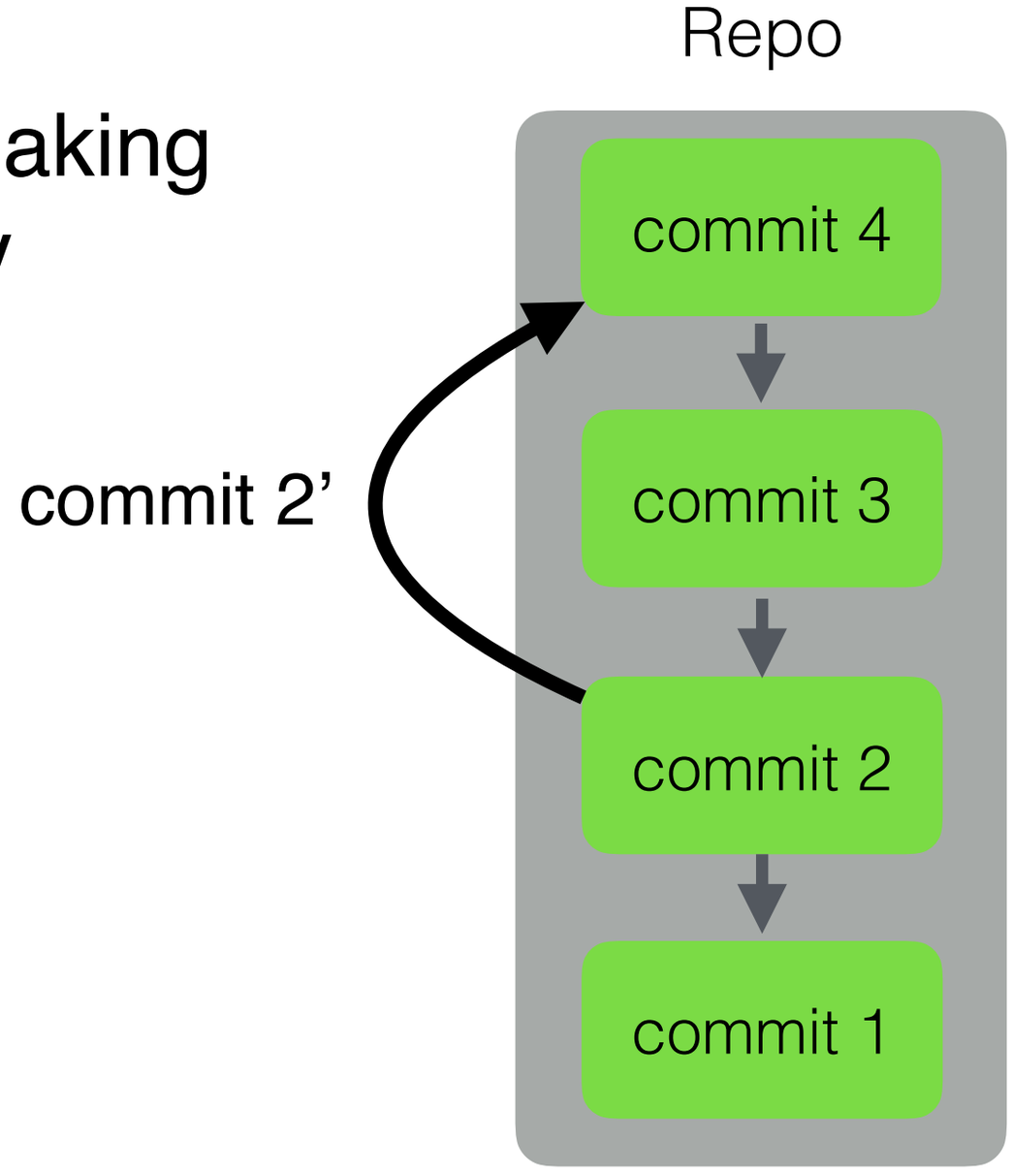
# Chain of Snapshots (***commits***)



# “Undo” (e.g. go to a previous version)

```
git revert commit3
```

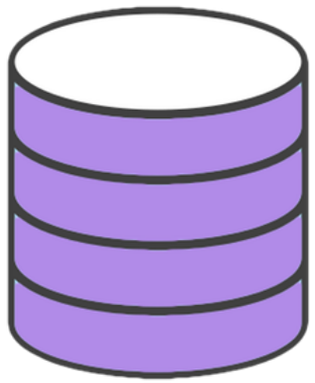
- Does not erase commit 3, making a new commit 4 that is really commit 2’.
- Keeps history!



# Sync to Server

## (backup and sharing)

`git remote add`



**Remote**
**e.g. Github**

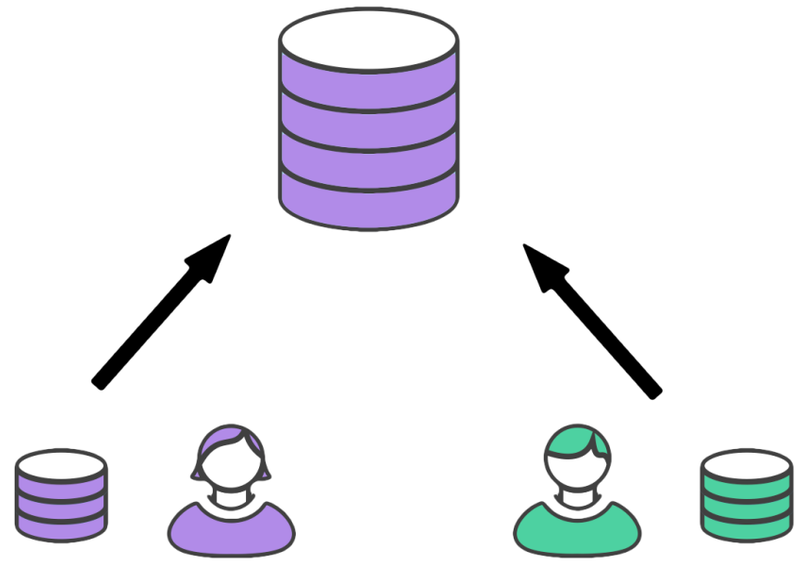
`git push` `git pull`

**Local**
e.g. Your machine

Remember: **Git and Github are not the same thing!**

# Collaborate

- A remote repository (e.g. in Github) is **owned** by someone
- Collaborators copy (`clone or fork`) and then
  - Pull updates from remote
  - Do work and Commit
  - Push changes to remote



- **Conflicts may arise** when people are working in parallel.