GIT AND GITHUB

Jeff Goldsmith, PhD

Department of Biostatistics
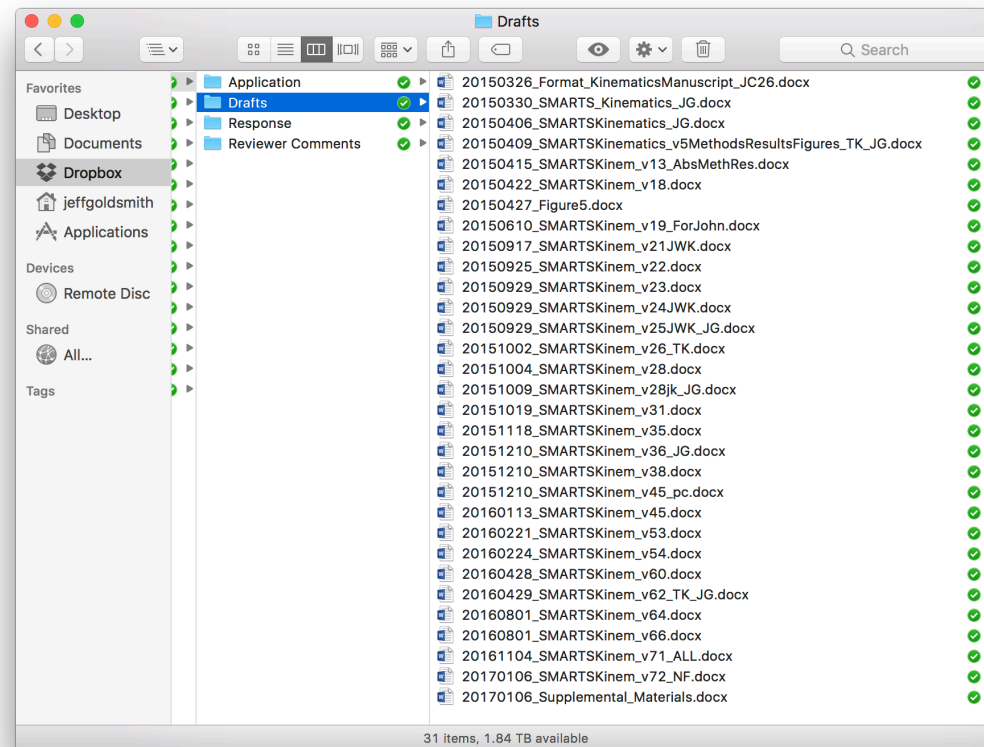
Is Git awesome?

- Yes

Is Git awful?

- Also yes

?????

- The good generally outweighs the bad
  - But there is some bad

So … what is Git?

- Kinda like Google Docs / Dropbox / track changes
- The goal is to avoid this:

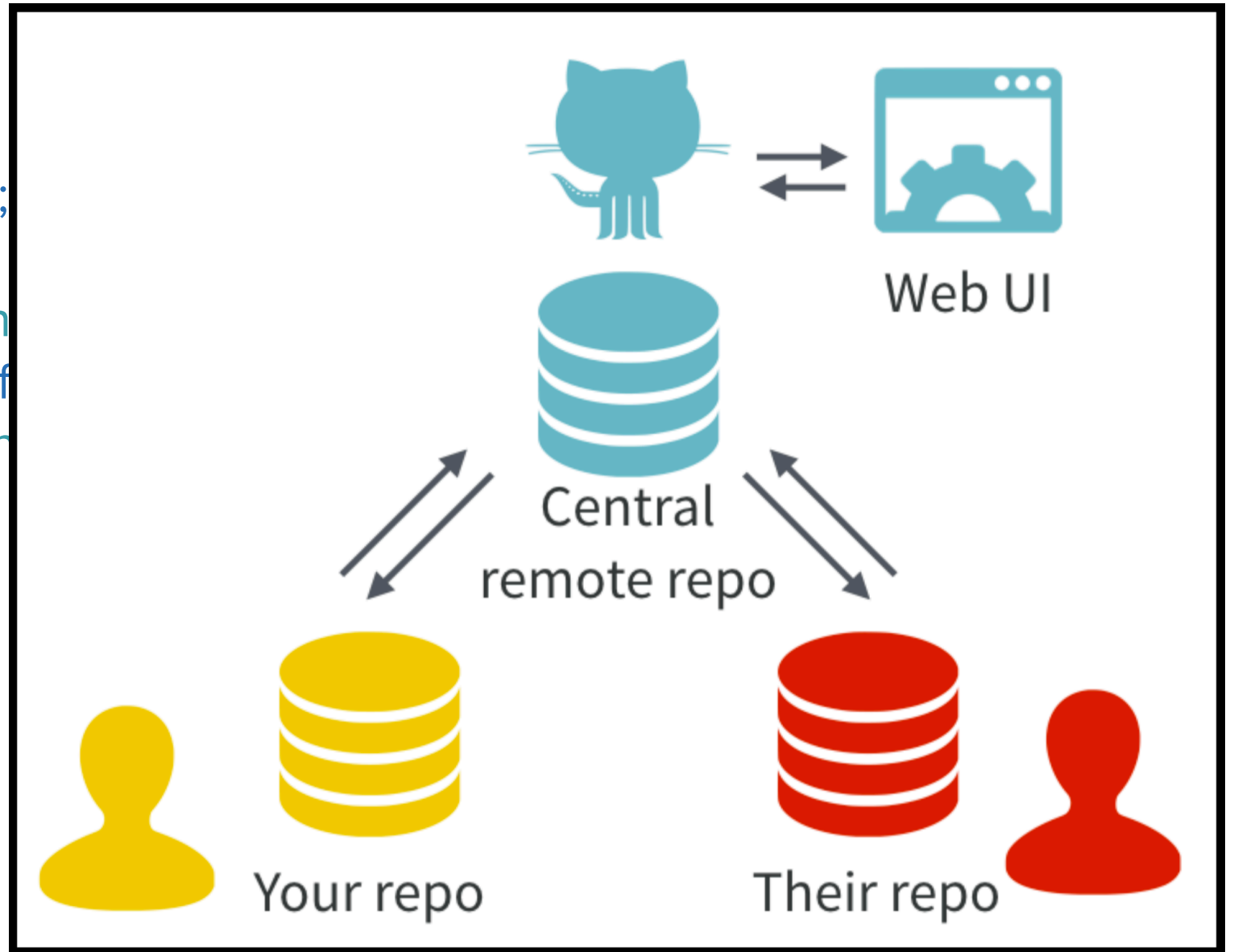# That still doesn't explain it.

- Git watches repositories – folders / directories – for changes
- It asks that you describe changes when they're made
- It remembers old versions if you need them
- It also keeps an eye out for conflicts, and forces you to resolve them
- It allows multiple people to contribute to the same repository, and does all of the above for everyone at once

And GitHub?

- Git lives on your computer; GitHub is a web-based platform for storing repositories
  - Think DropBox, but with Git in your folders (watching you)
- GitHub is a great platform for disseminating work
  - You can easily create and host reports; websites; R packages; …

And GitHub?

- Git lives on your computer; repositories
  - Think DropBox, but with
- GitHub is a great platform f
  - You can easily create ar

"Excuse me, do you have a moment to talk about version control?"

What about RStudio

- Git is something you should be doing, and RStudio tries to make it easy for you to do
- R Projects can initialize Git with a mouse click
- Then, everything in the project is being watched

And a Git client?

- Git is a command-line tool
- Git clients let you do most Git-related stuff in a GUI
  - Git client is to git as RStudio is to R

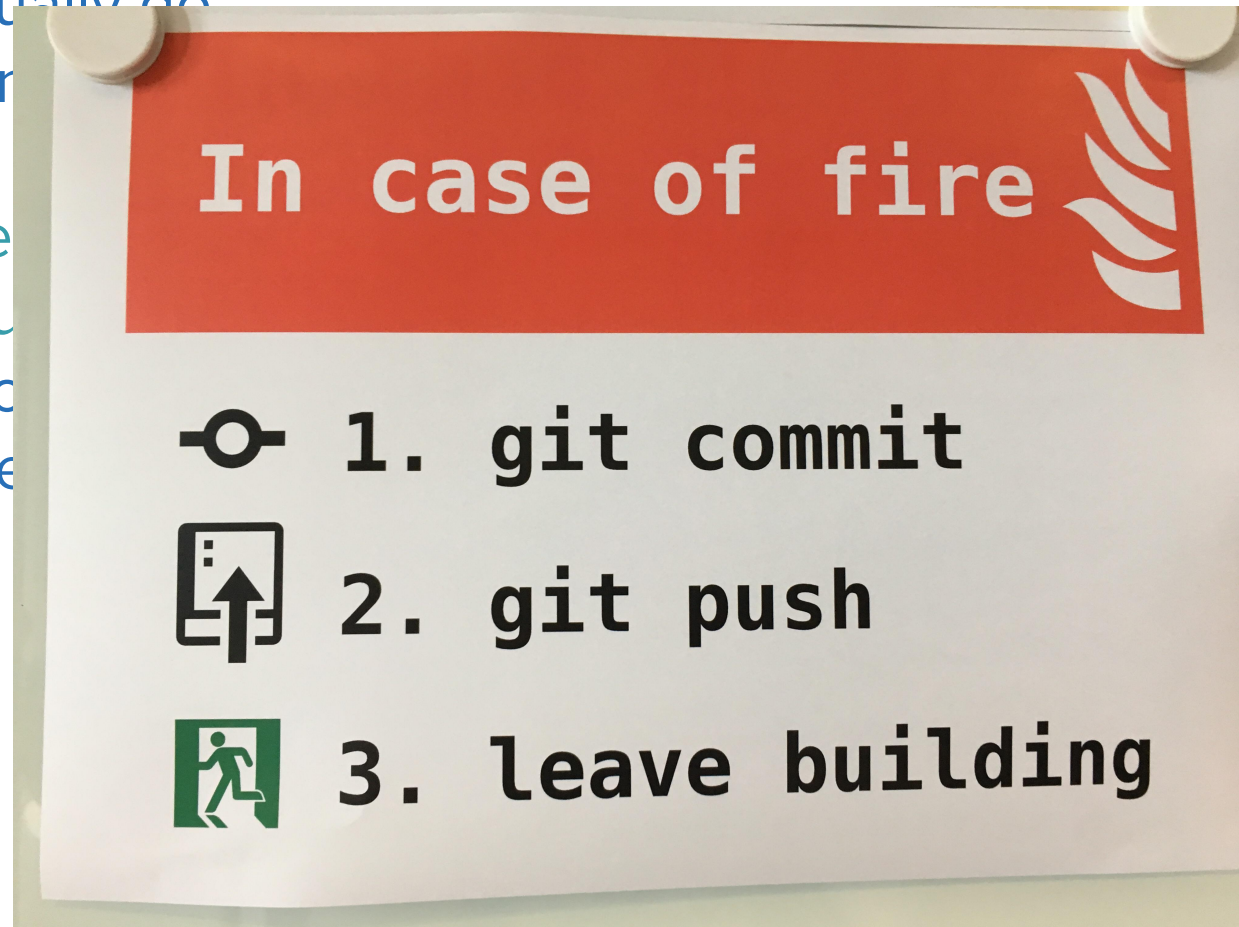- RStudio has a bare-bones Git client which will work for most stuff

Workflow

- When starting a new analysis / project / whatever, I
  - Create GH repo
  - Create linked R Project using repo URL
  - Do stuff

"Doing stuff" in a git repo

- You do whatever you would usually do
- Once you've done some amount of stuff, you commit the changes
  - "commit" = "fancy save"
  - Git will keep track of changes between commits
  - Your commit message will summarize what's different
- Then you do more stuff, then you commit, then you do more stuff …
- Push changes to GitHub – more on that soon

"Doing stuff" in a git repo

- You do whatever you would usually do
- Once you've done some amour
  - "commit" = "fancy save"
  - Git will keep track of change
  - Your commit message will su
- Then you do more stuff, then yo
- Push changes to GitHub – more

Pros:

- You can revert to earlier commits if you mess something up
- You can quickly review the development process
- You can see what collaborators are doing, where they're doing it, and why
- You're forced to resolve conflicts (two people changing the same thing at the same time) as they arise

Cons:

- There is a lot of overhead, and it's worst at the beginning
- "Resolving conflicts" can be awful
- Everyone on a project is required to stick with the same development pipeline

Vocab

- Repository
- Commit
- Push / Pull
- Branch / Merge
- Cloning? Forking??

Not going to cover

- Messaging
- Issue tracking

Confidentiality

- You have to watch out for data confidentiality – GitHub is public!