Version Control with Git and Github: the very basics

INST-377: Dynamic Web Applications
Instructor: Alex Leitch

What is Version Control?

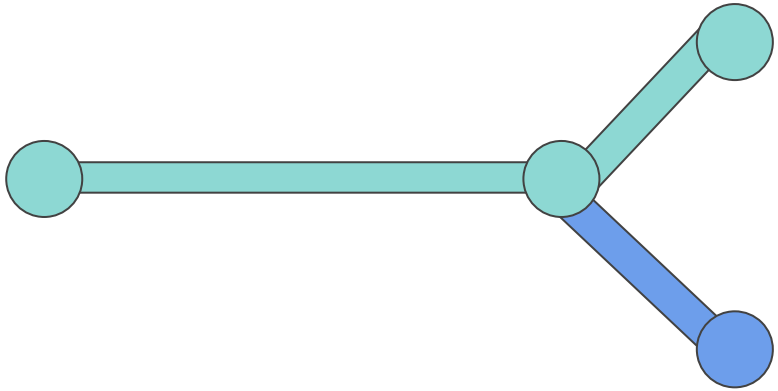


Fig. 1 - make some choices

What is Version Control?

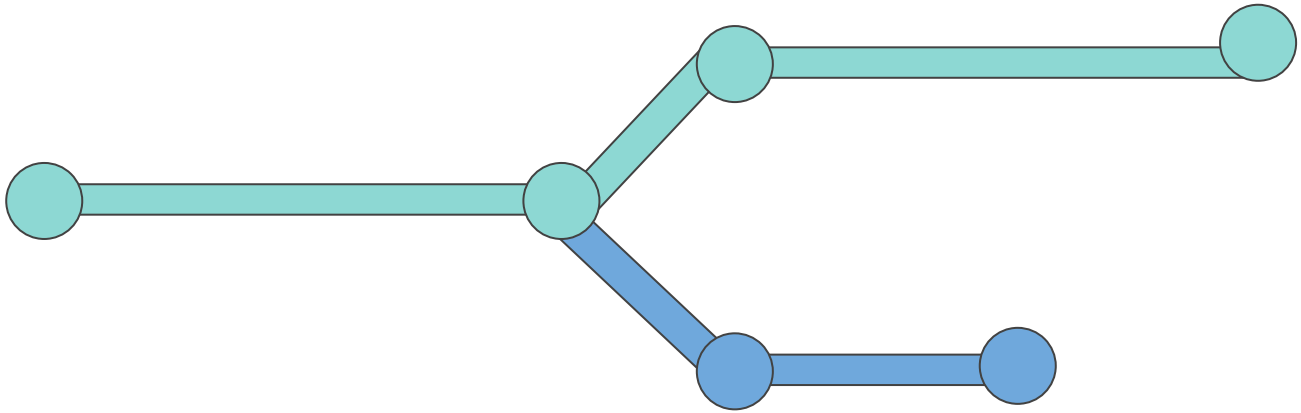


Fig 2. Two different ways of thinking

What is Version Control?

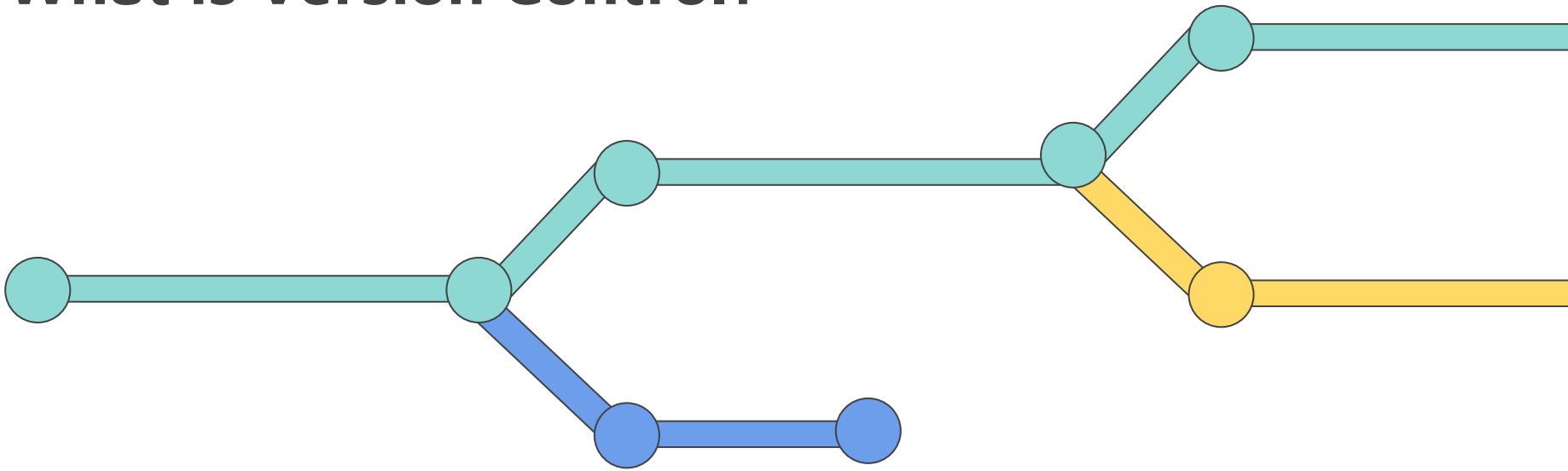


Fig 3. Some good ideas, some not-great ones

What is Version Control?

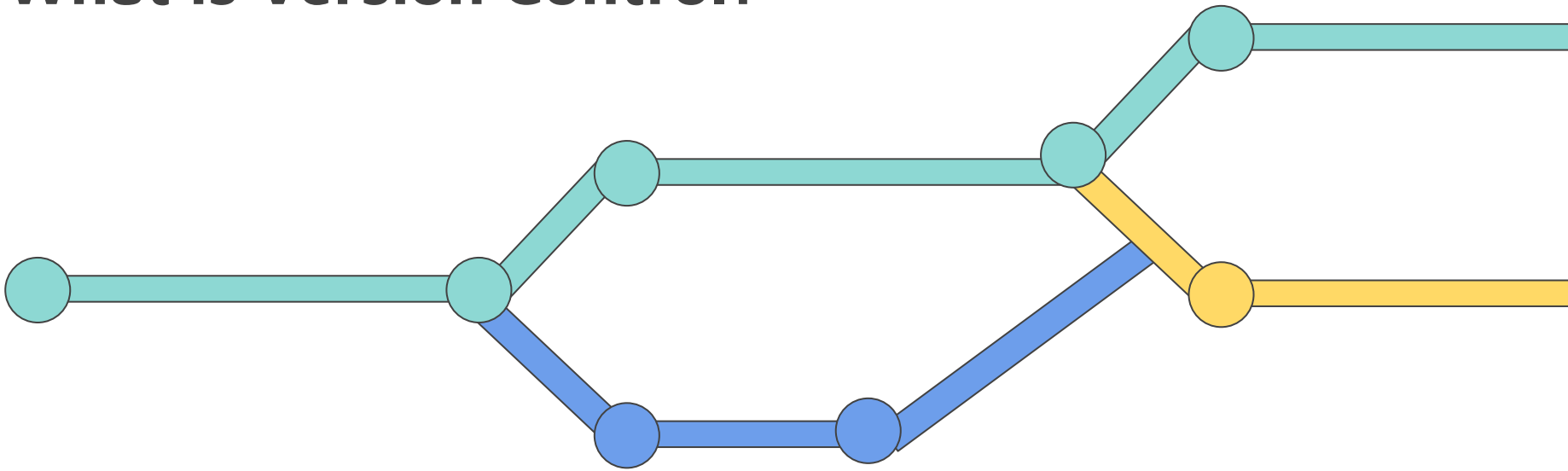


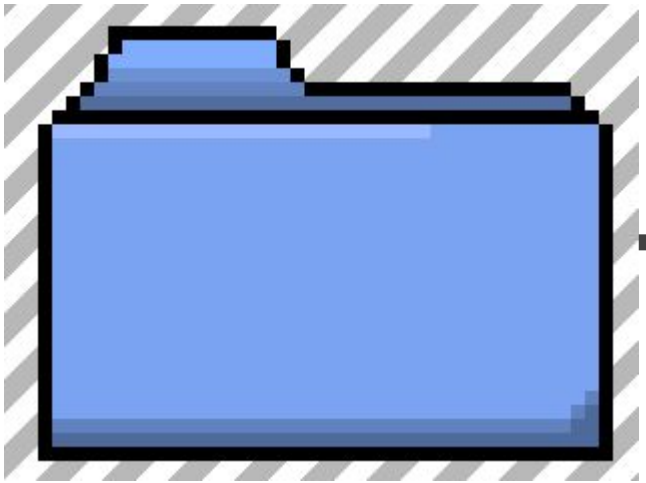
Fig 4. Wait, I needed that.

What is Git, specifically

- One of about six systems for this
- Made popular by Github
- **Git** is the software that lets us make version control choices
- **Github** is the company that helps us have conversations with our peers about those choices

git

What is Git on your computer?



.git
.gitignore
.gitattributes
index.html
/whatever_folder

Git Exercise - work in pairs and talk about it

- On **github.com**, please make a new account, with a **professional name.**
  - You'll use this to submit work for this class, and to back up your work.

- Install **either** Github Desktop **or** the Github command-line interface (CLI)
  - Optional/not really: Install a graph visualizer, like GH Desktop or Sourcetree or Tower for Mac.

- Connect to your Github account with eithwe

- Make a new directory, and initialize a new repository in that directory.

- Add a text file to the repository folder, type something into it, and save it as `index.html`
  - Commit that file
  - Open up your **graph visualizer** of choice - Github desktop, etc.
  - Have a look at your file

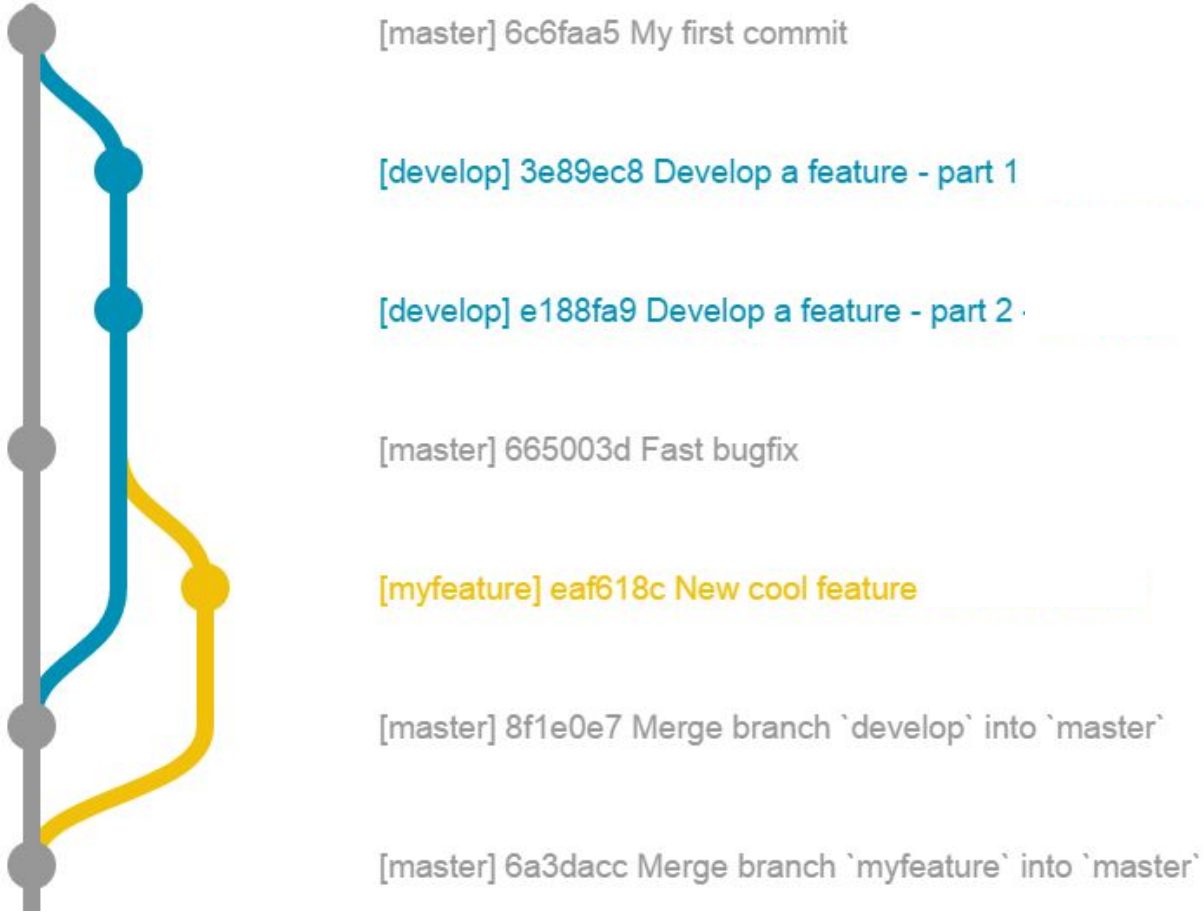


Fig. 5 - Sample Git Chart

Fig. 6 - `master` Branch - this goes public.

Fig. 7 - `develop` Branch - finished but not public

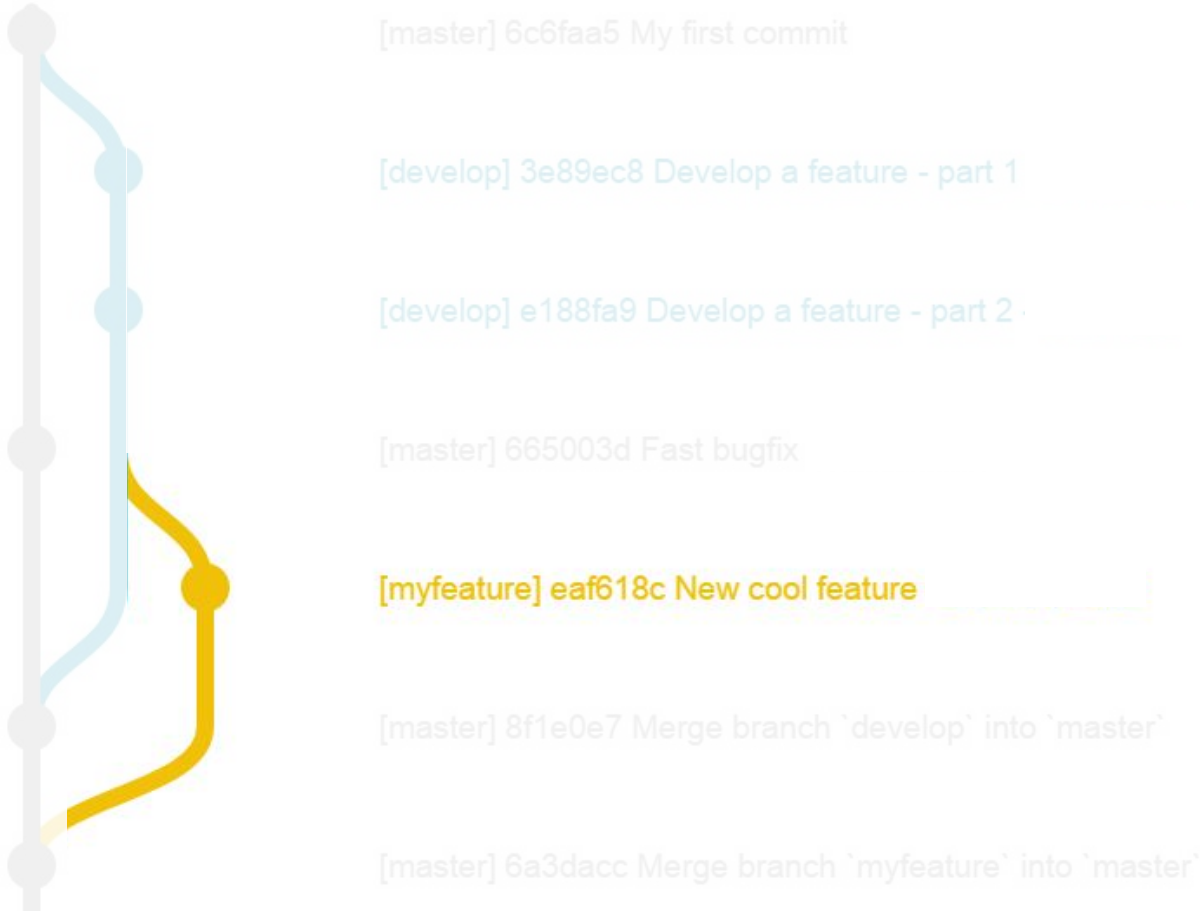


Fig. 8 - `feature` Branch - what you're *working* on

[master] 6c6faa5 My first commit

[develop] 3e89ec8 Develop a feature - part 1

[develop] e188fa9 Develop a feature - part 2

[master] 665003d Fast bugfix

[myfeature] eaf618c New cool feature

[master] 8f1e0e7 Merge branch `develop` into `master`

[master] 6a3dacc Merge branch `myfeature` into `master`

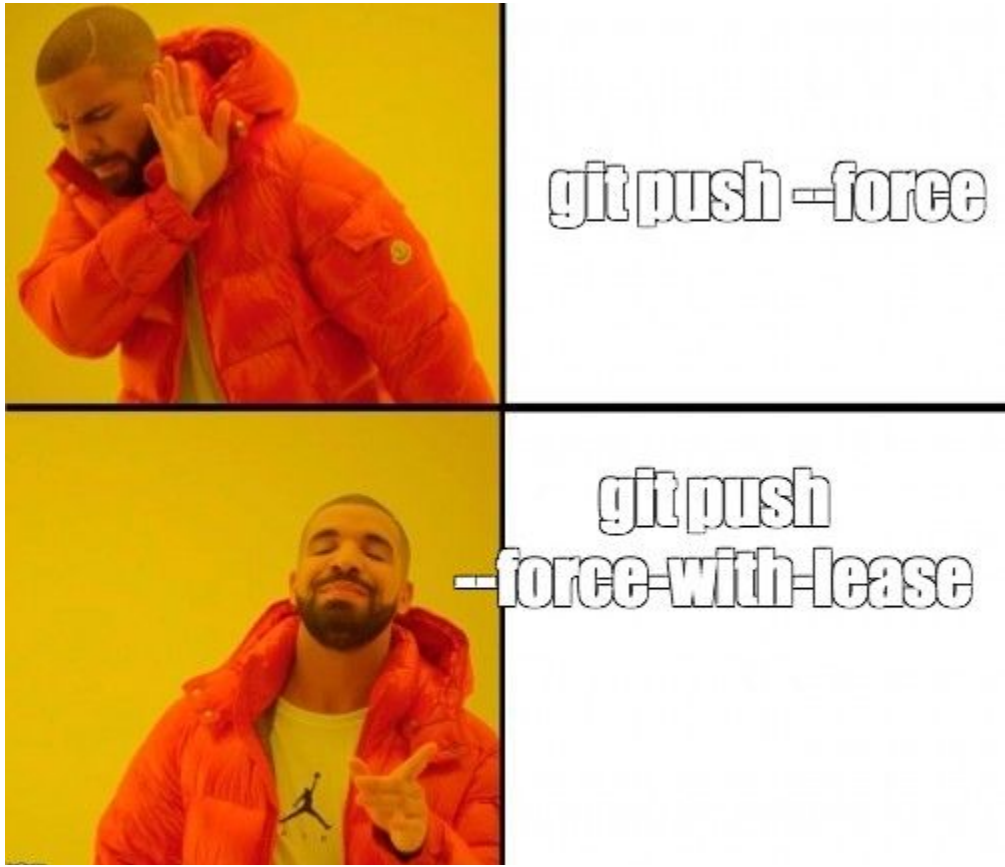
The goal is to keep it legible.

git push --force-with-lease

This is a more polite way of forcing a change into a separate branch of code.

Please do not git push --force.

It will overwrite your remote branch even if someone else has made fun changes.



Fun final git tips

- Please go to **ohshitgit.com** and bookmark it. It is the single best source for command-line guidance on handling git.

- Consider picking up the zine “oh shit git” from https://wizardzines.com/zines/oh-shit-git/ - it is a printable resource and very worthwhile for learning how Git actually works.