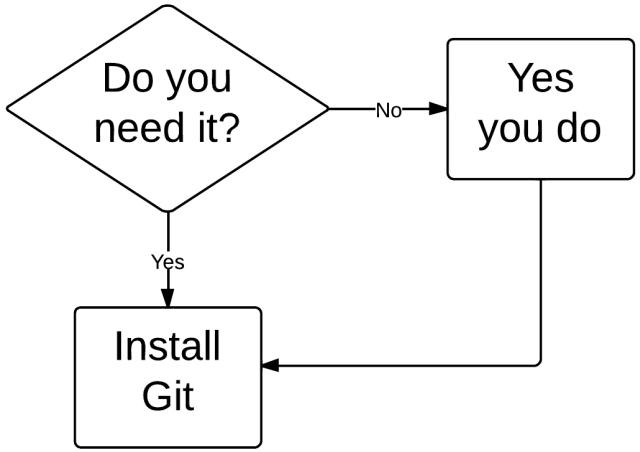
Recitation 2: Gitflow

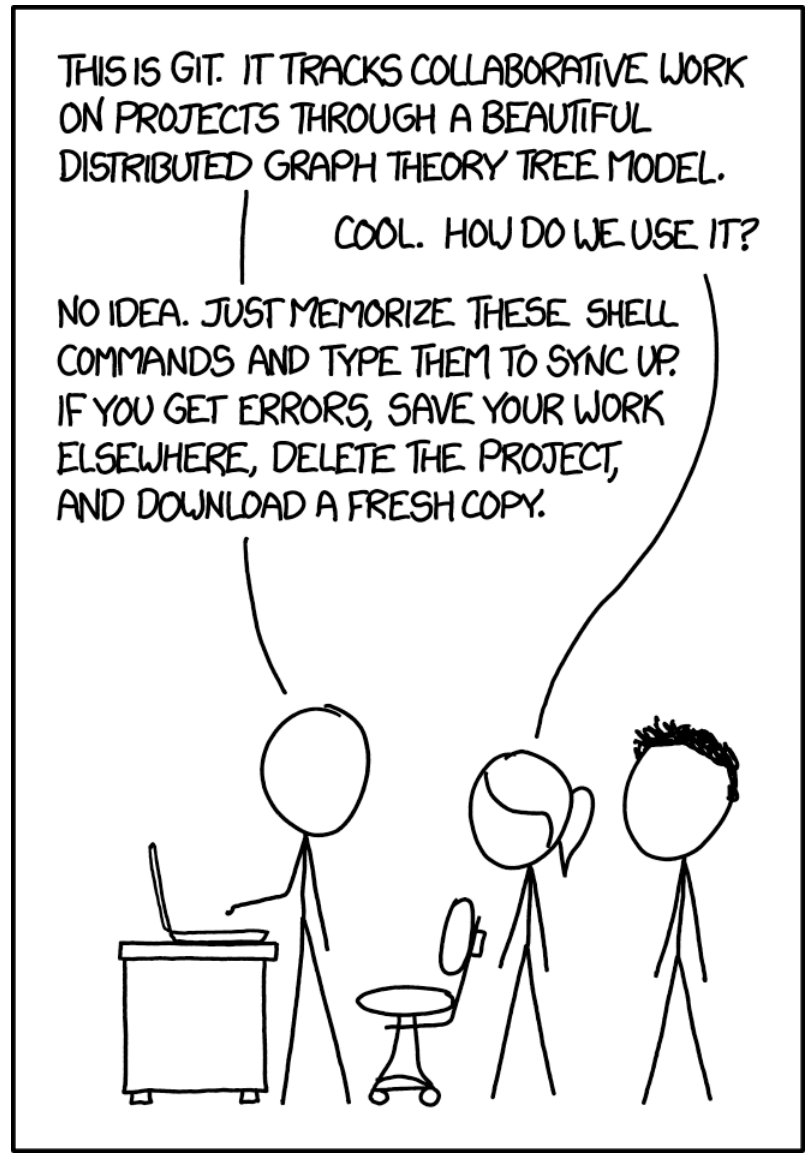
git

Version Control Flowchart

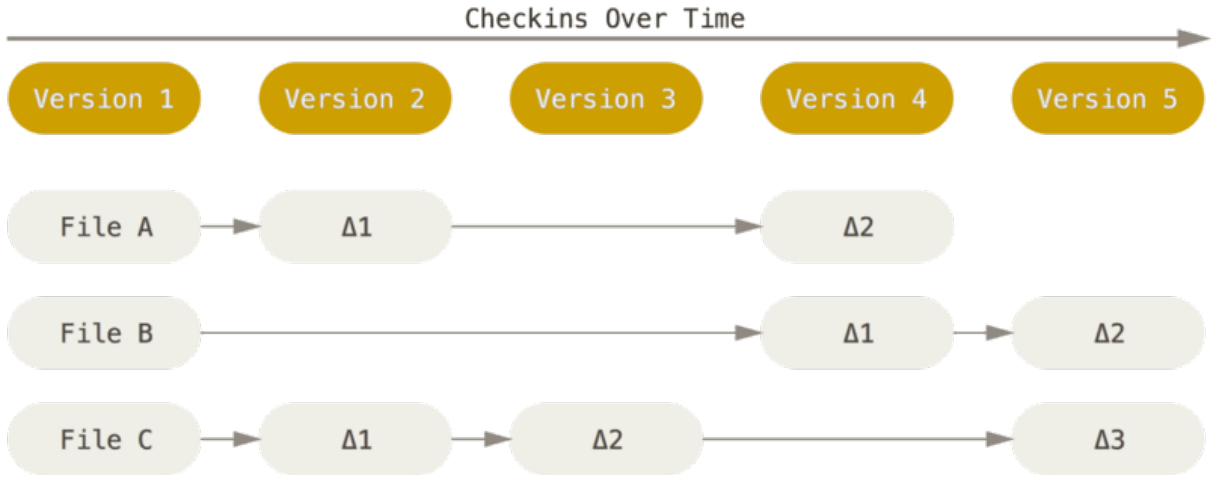


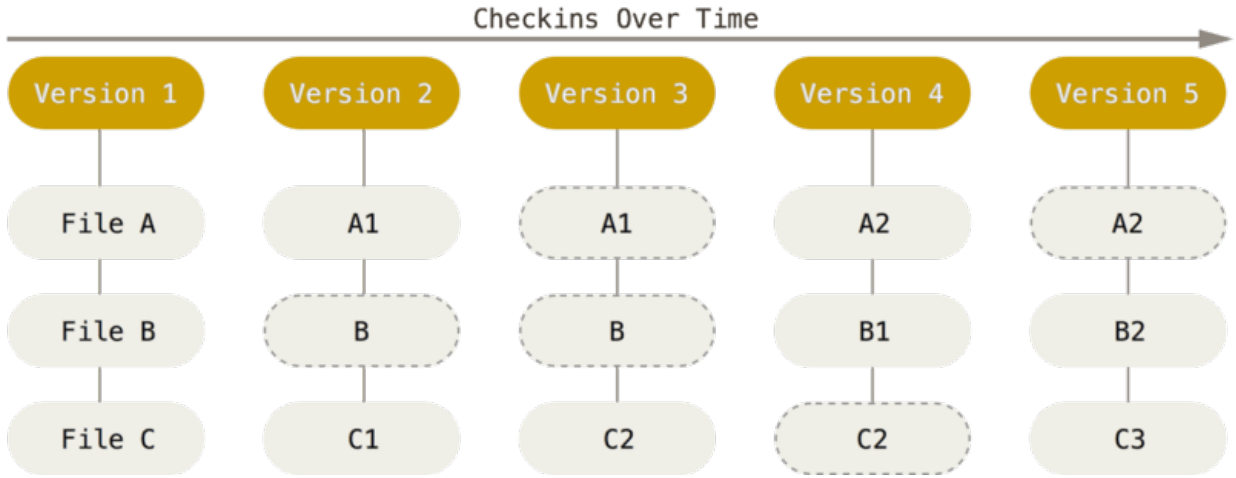
Learning objectives

- Learn how to use git and GitHub for your teamwork in this course
- Gitflow & GitHub flow
- How to collaborate with other developers using git


THIS IS GIT. IT TRACKS COLLABORATIVE WORK ON PROJECTS THROUGH A BEAUTIFUL DISTRIBUTED GRAPH THEORY TREE MODEL.
COOL. HOW DO WE USE IT?
NO IDEA. JUST MEMORIZE THESE SHELL COMMANDS AND TYPE THEM TO SYNC UP. IF YOU GET ERRORS, SAVE YOUR WORK ELSEWHERE, DELETE THE PROJECT, AND DOWNLOAD A FRESH COPY.

Snapshots, not differences



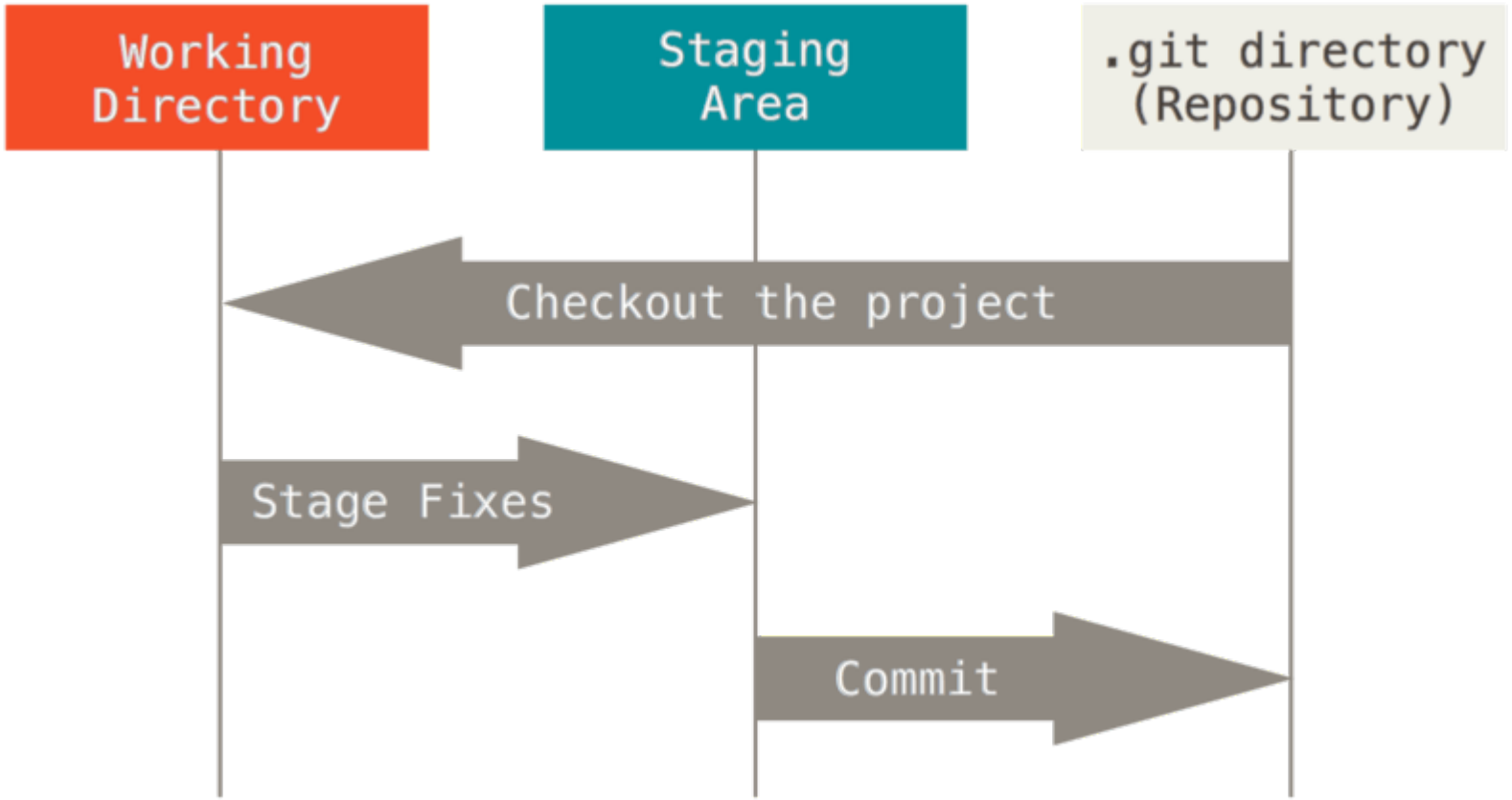


More than a VCS
"A mini-filesystem"

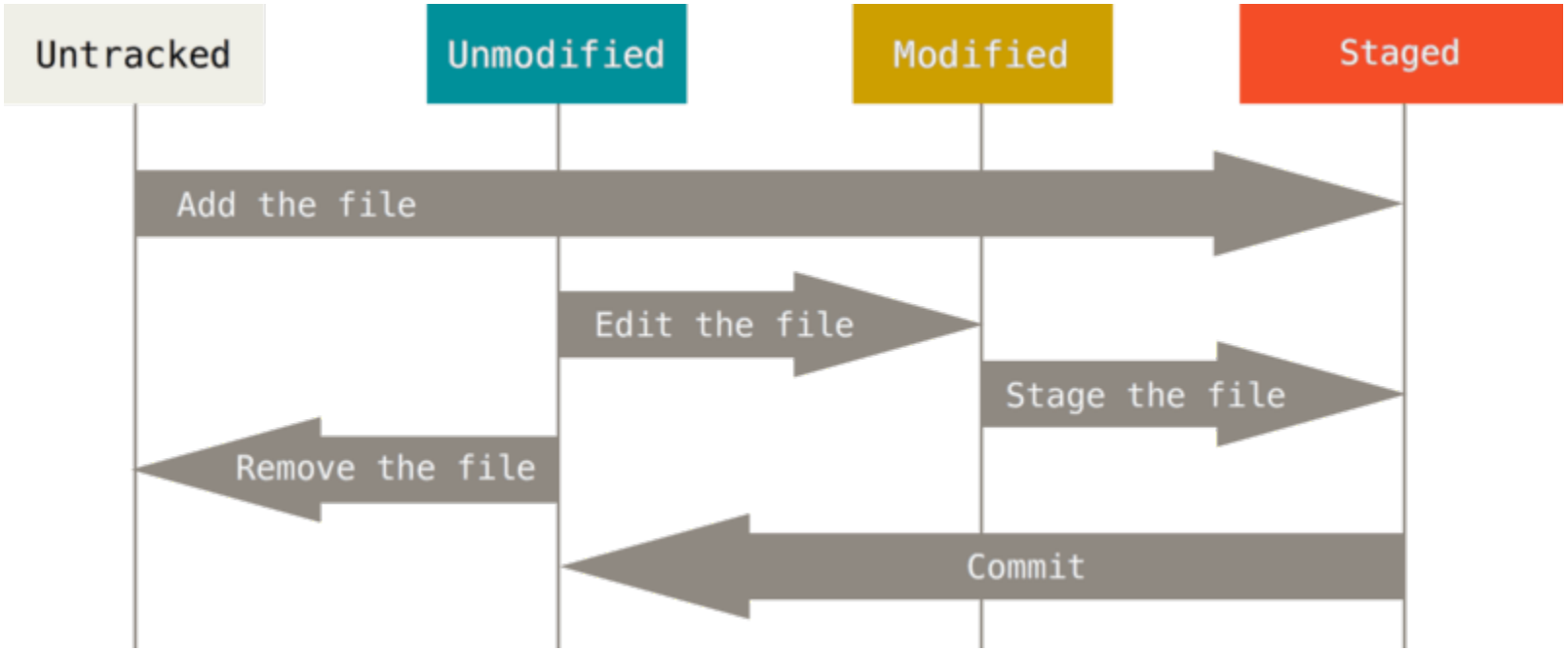
Git: More features

- Nearly every operation is local
- Integrity
- Generally only adds data
  - “It’s hard to get the system to do something that is undoable”

The most important slide

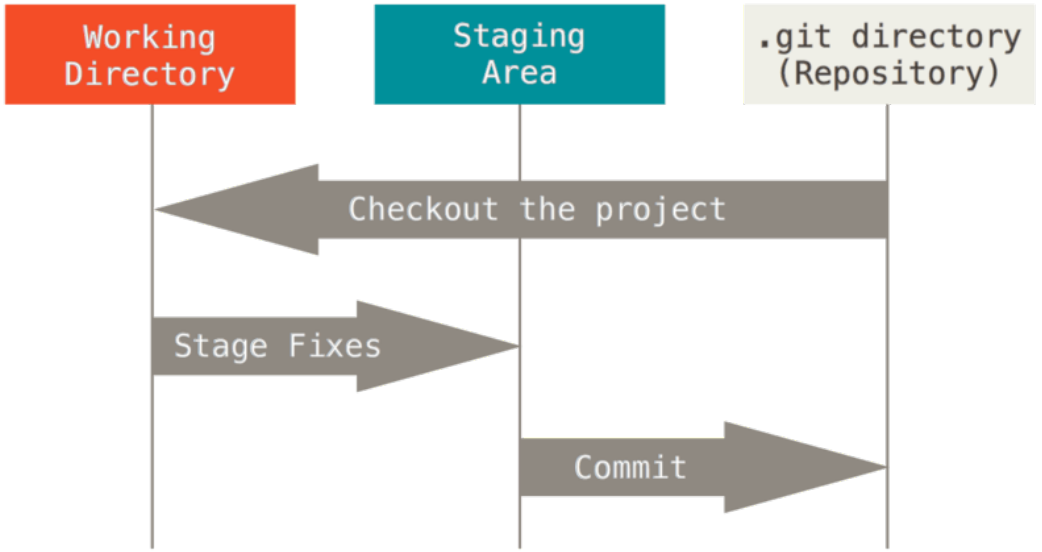


Recording changes
(second most important slide)



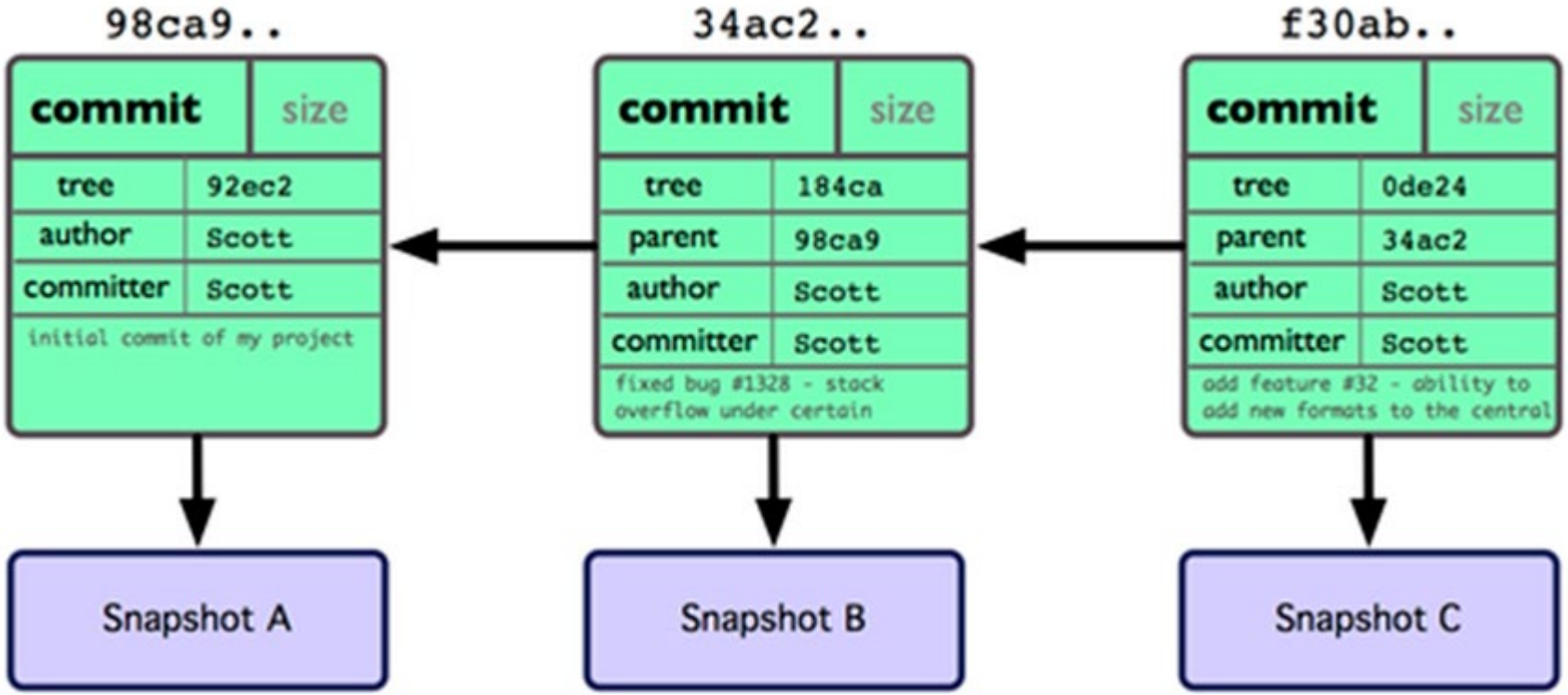
Staging Area

- Why is it useful?

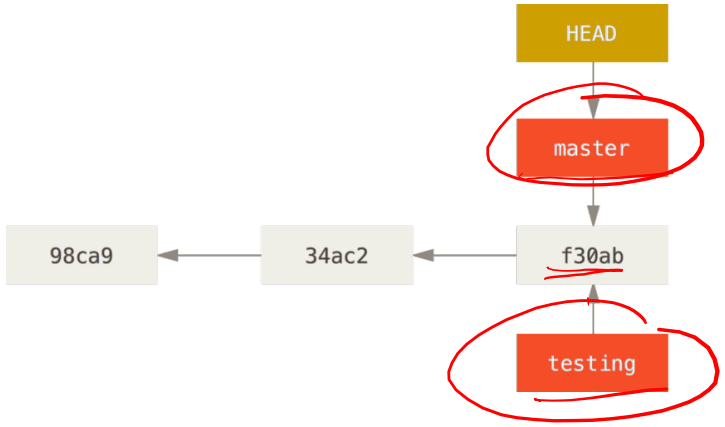


What’s inside the repository? (.git)

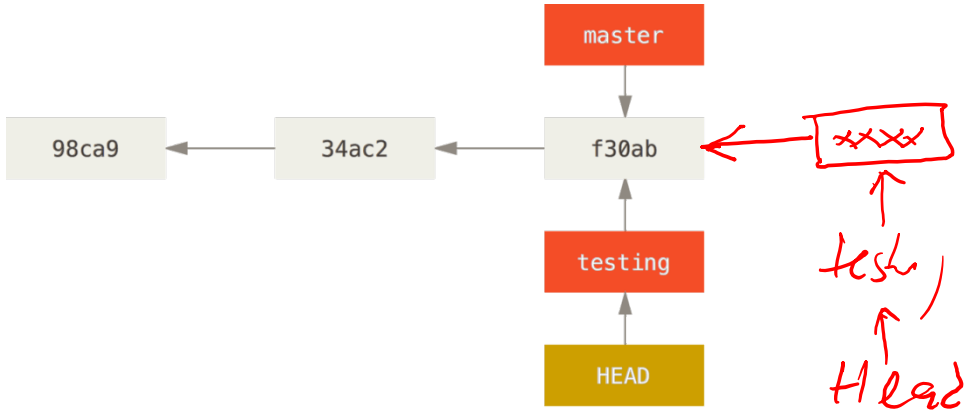
Git isn’t magic!



`git branch testing`

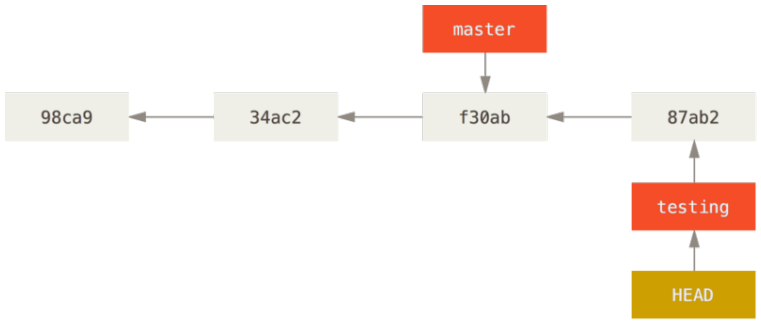


`git checkout testing`

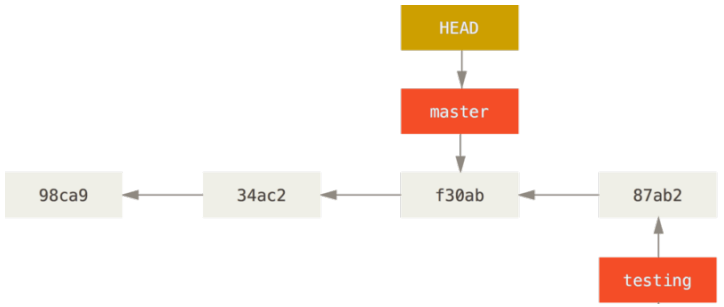


Always remember where is your HEAD!

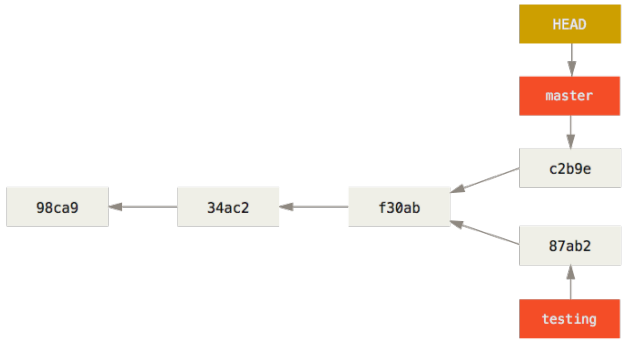
git commit



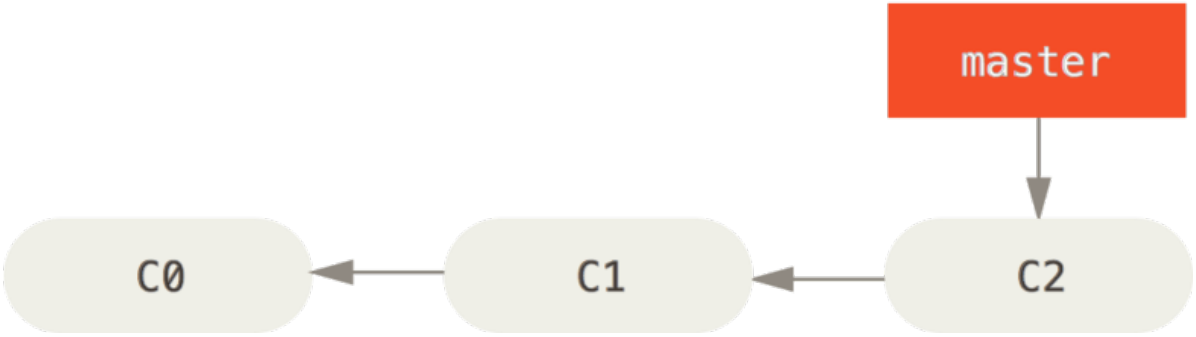
git checkout master



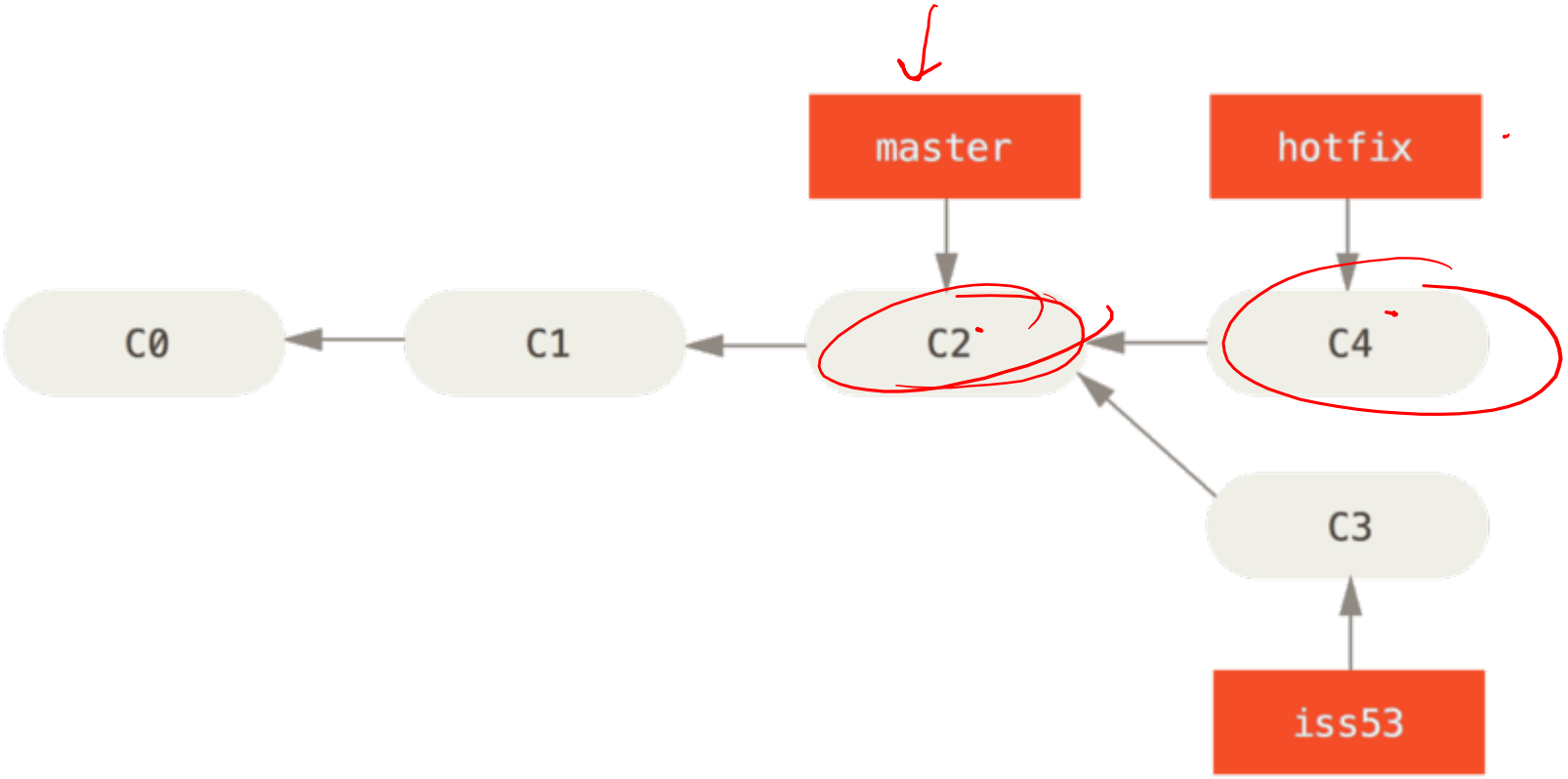
git commit



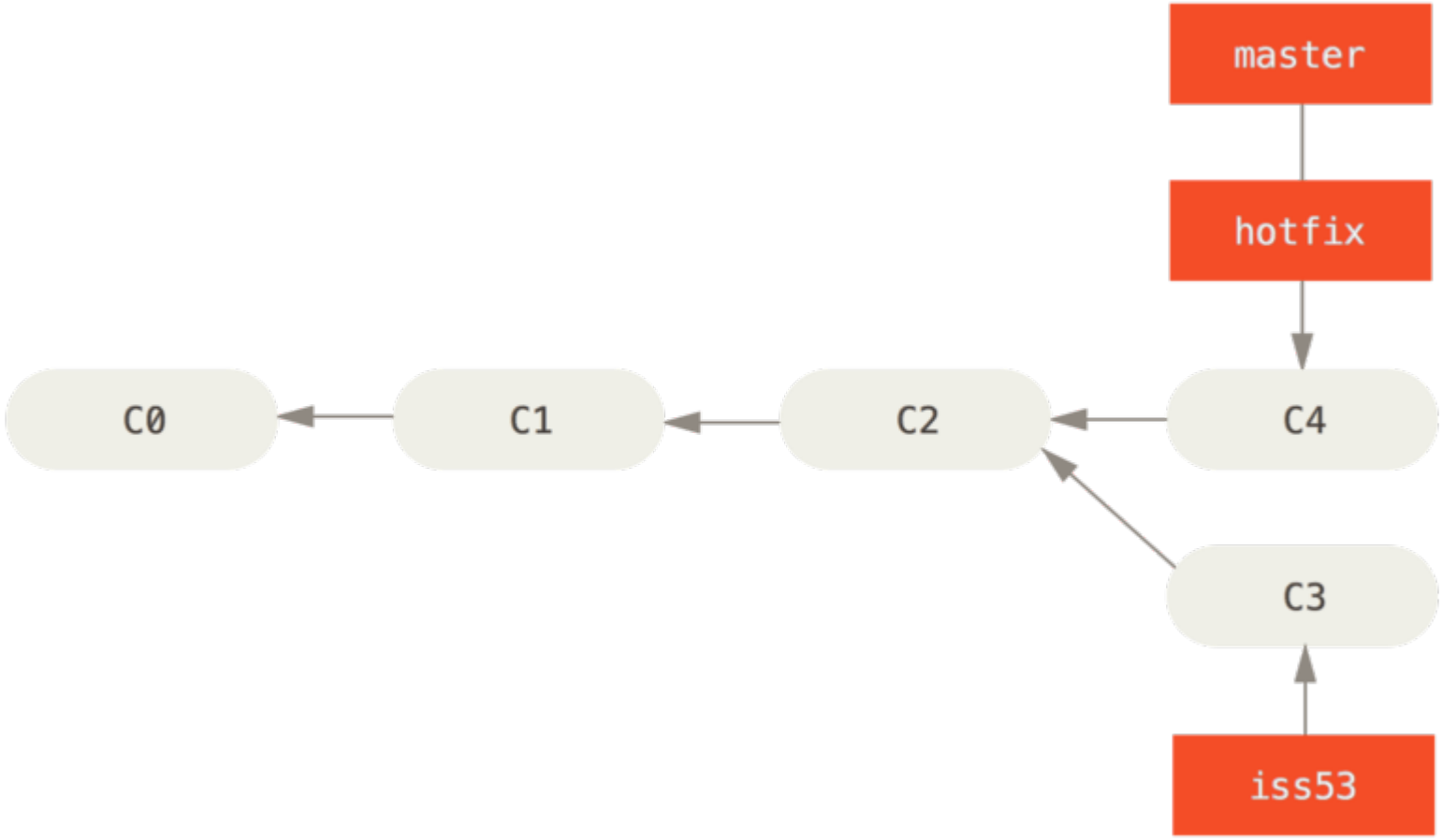
Basic merging



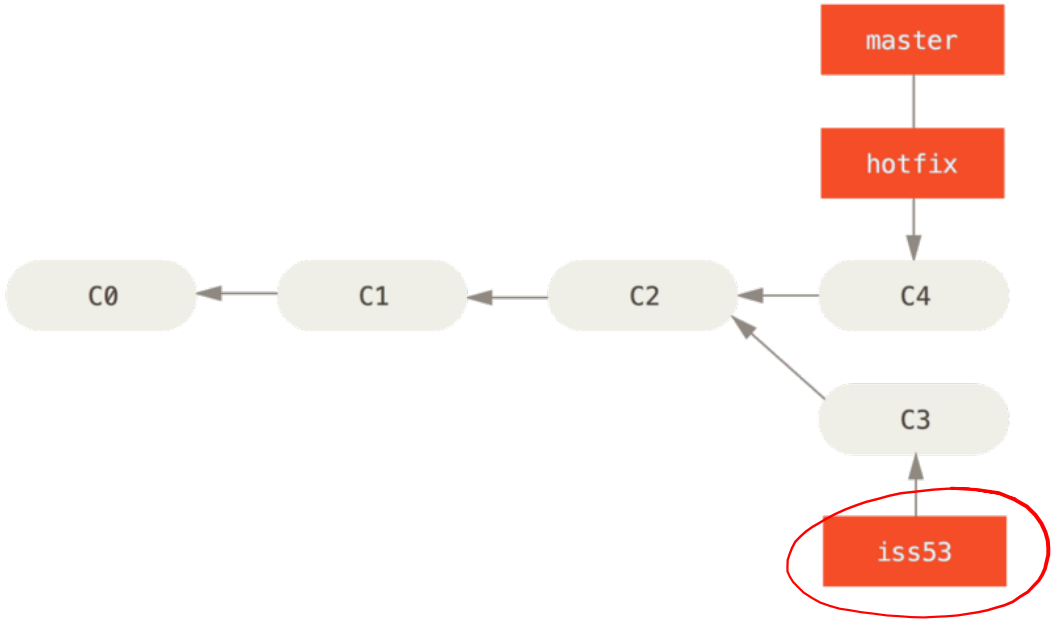
Basic merging

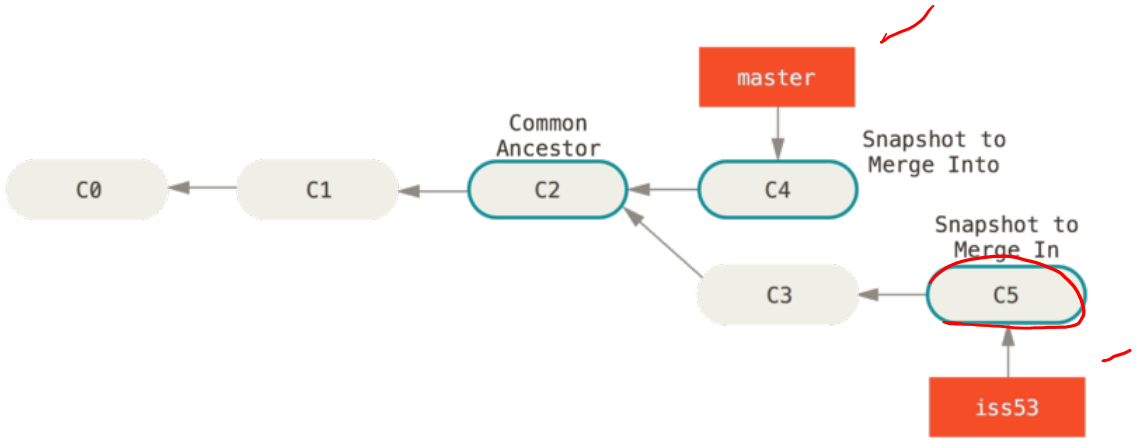


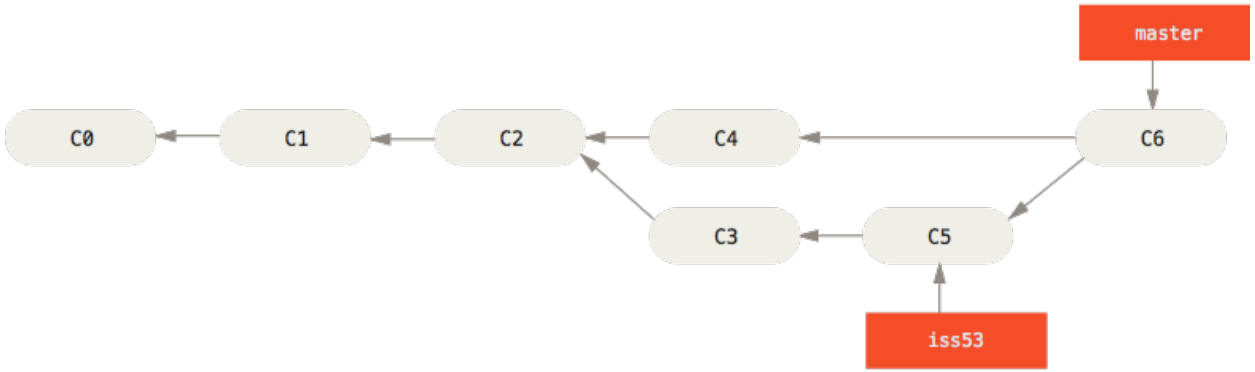
Basic merging (fast-forwarding)



Basic merging







Fixing conflicts

- Conflicted files will have conflict regions indicated by:

```
<<<<<<< Your changes
the content on your branch
=======
the content on their branch
>>>>>>> Their changes
```

- To fix them, edit the files to the state you would like them to be, then add and commit the changes.
- I like to use `git mergetool` to fix conflicts.

Git Review

- git status
  - Use it often to find out what is happening!
- git add
  - Stages changes for commit (-p for staging partial changes of a file)
- git commit
  - Creates a commit with all staged changes
- git branch <name>
  - Creates a new branch from the current commit called name
- git checkout <name>
  - Switches to a branch called name

Undoing changes

- **git restore <file>**: before `git add`
- **git restore --staged <file>**: after `git add` but before `git commit`
- `git reset:` after `git commit`

- Image credits:
  - https://www.slideshare.net/ThomasRausch4/git-introduction-tutorial
  - https://medium.com/@srebalaji/a-very-basic-intro-of-git-b9cab0e64153

- Download Pro Git book:
  - https://git-scm.com/book/en/v2