# Commitment Issues:

## Understanding git for better version control and collaboration

Jacob Adams, UGRC

UGIC 2024

## git: Not Just an Unpleasant Person

## Ever wish baking had an undo button?

Photo by Vaibhav Jadhav: https://www.pexels.com/photo/man-preparing-dough-for-bread-3218467/

## All the benefits

Undo just about everything

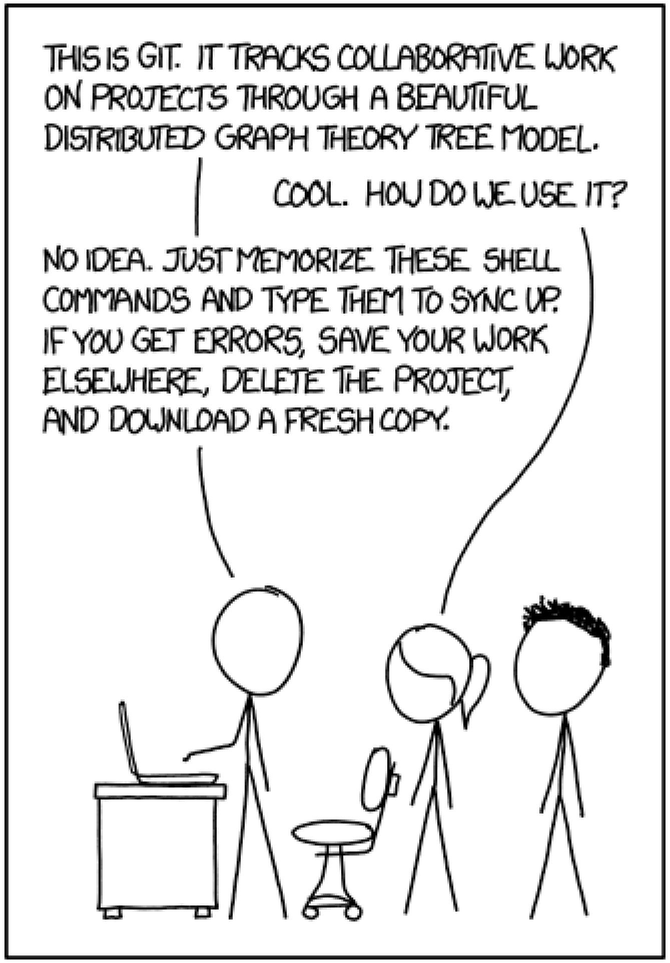
Always have a known-good copy

Try new things without fear

Enable complex collaborations

Share your code easily

DON'T
PANIC

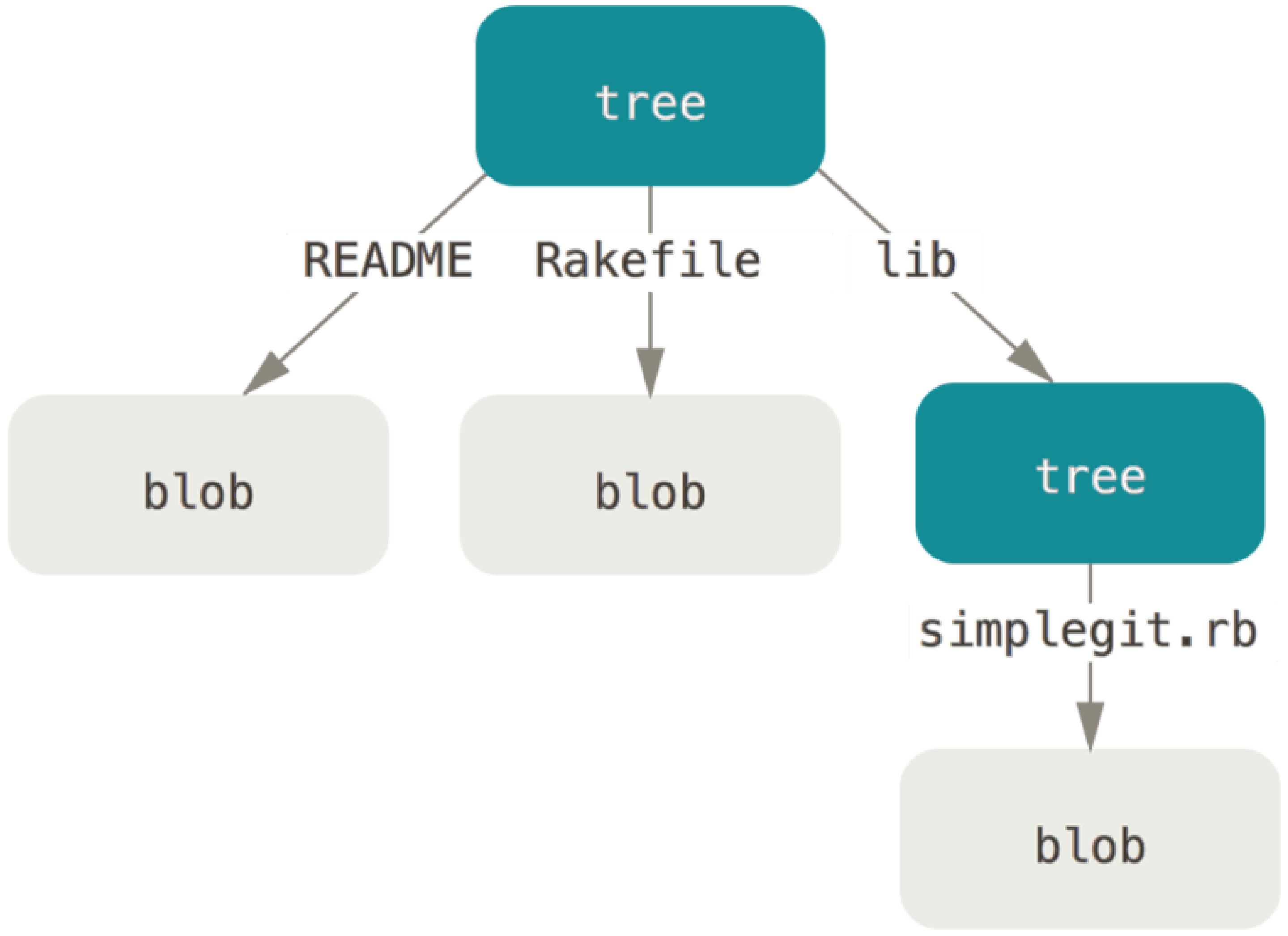

THIS IS GIT. IT TRACKS COLLABORATIVE WORK ON PROJECTS THROUGH A BEAUTIFUL DISTRIBUTED GRAPH THEORY TREE MODEL.
COOL. HOW DO WE USE IT?
NO IDEA. JUST MEMORIZE THESE SHELL COMMANDS AND TYPE THEM TO SYNC UP. IF YOU GET ERRORS, SAVE YOUR WORK ELSEWHERE, DELETE THE PROJECT, AND DOWNLOAD A FRESH COPY.

https://xkcd.com/1597/

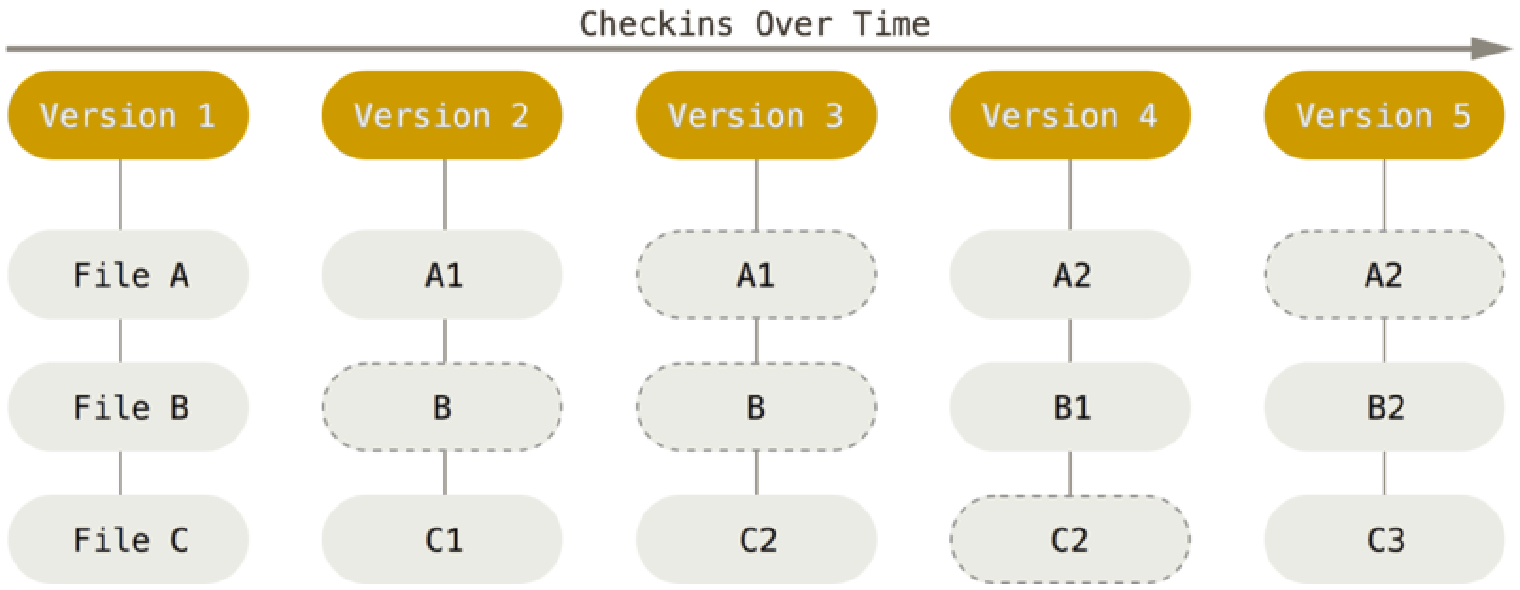
## Laying the Foundation

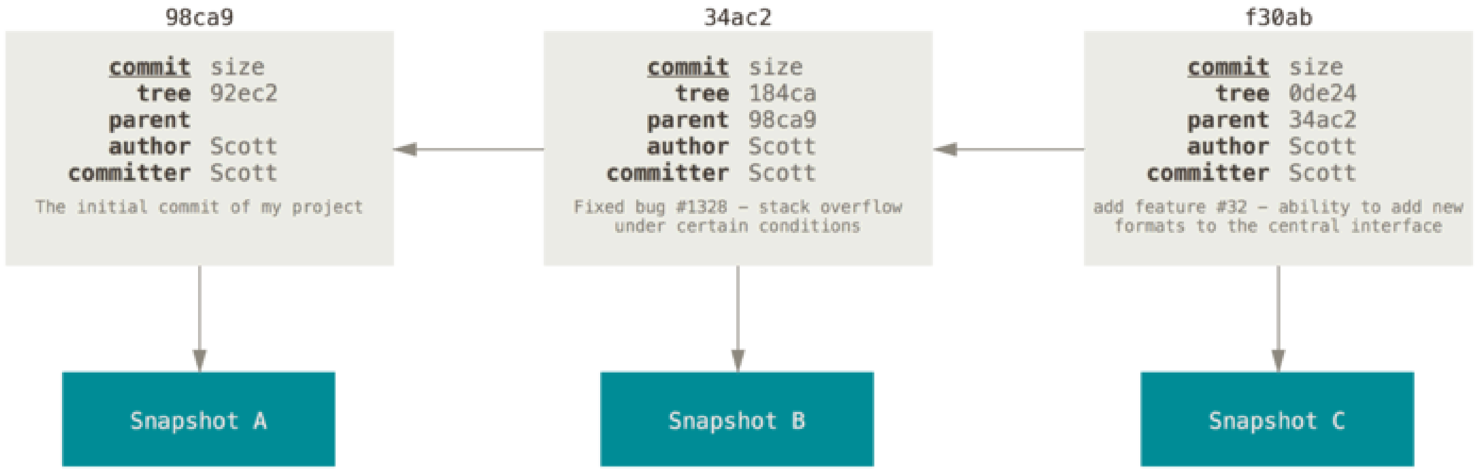
## Basic Units of git

## Blobs and trees: Storing files and directories

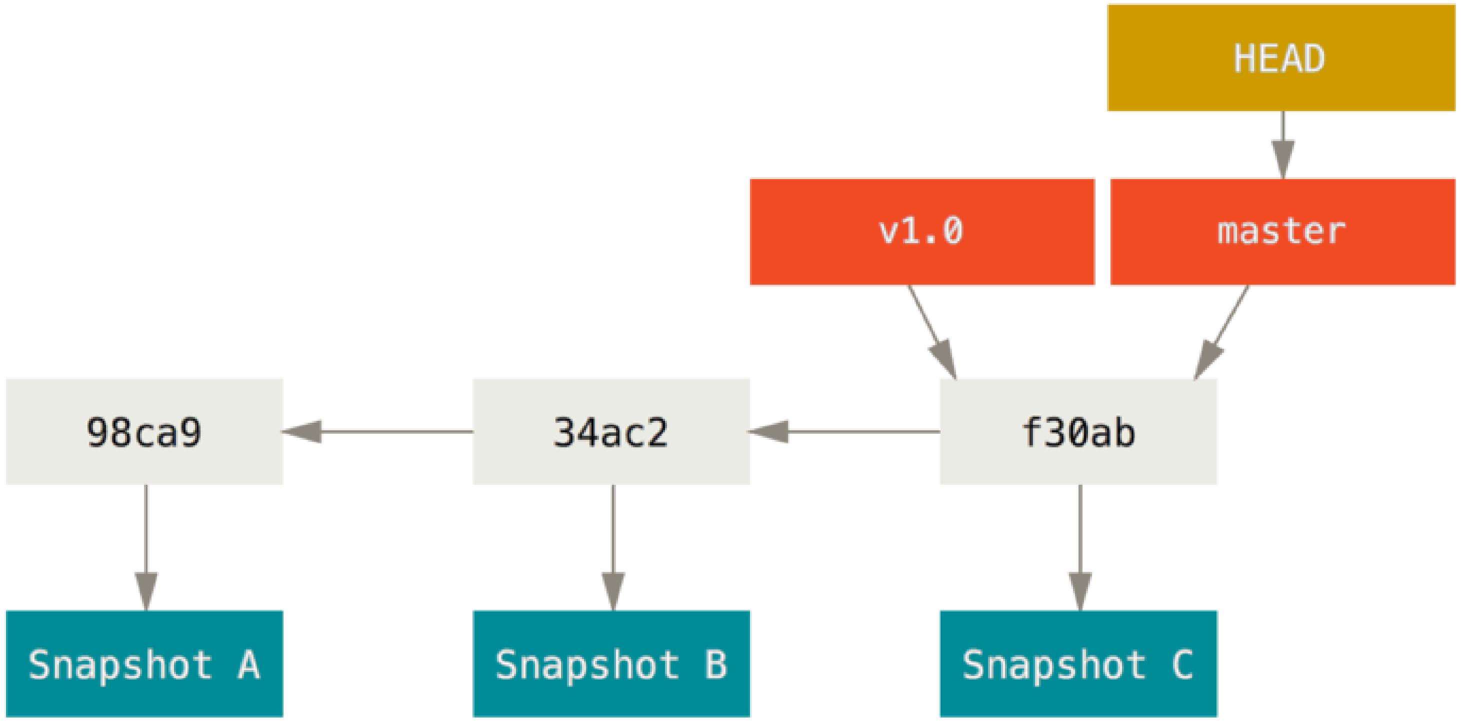


## Commit: A snapshot in time

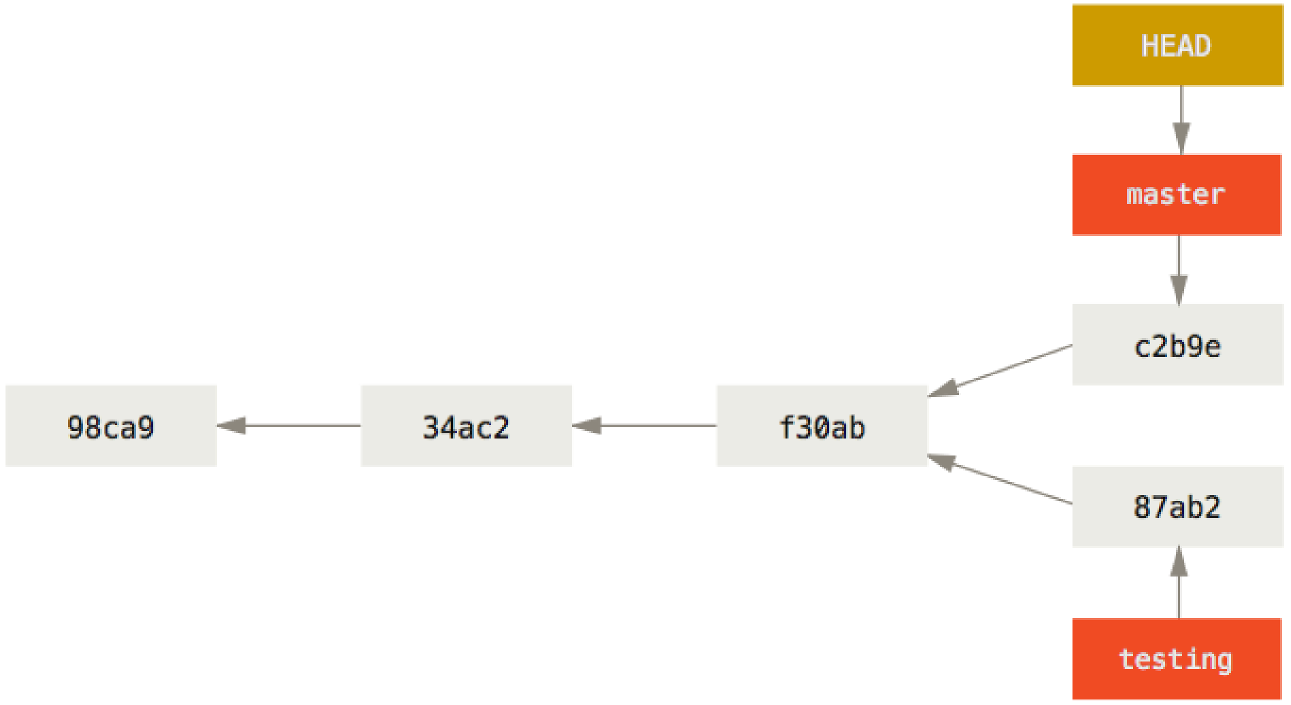



98ca9
commit size
tree 92ec2
parent
author Scott
committer Scott
The initial commit of my project
34ac2
commit size
tree 184ca
parent 98ca9
author Scott
committer Scott
Fixed bug #1328 - stack overflow under certain conditions
f30ab
commit size
tree 0de24
parent 34ac2
author Scott
committer Scott
add feature #32 - ability to add new formats to the central interface
Snapshot A
Snapshot B
Snapshot C

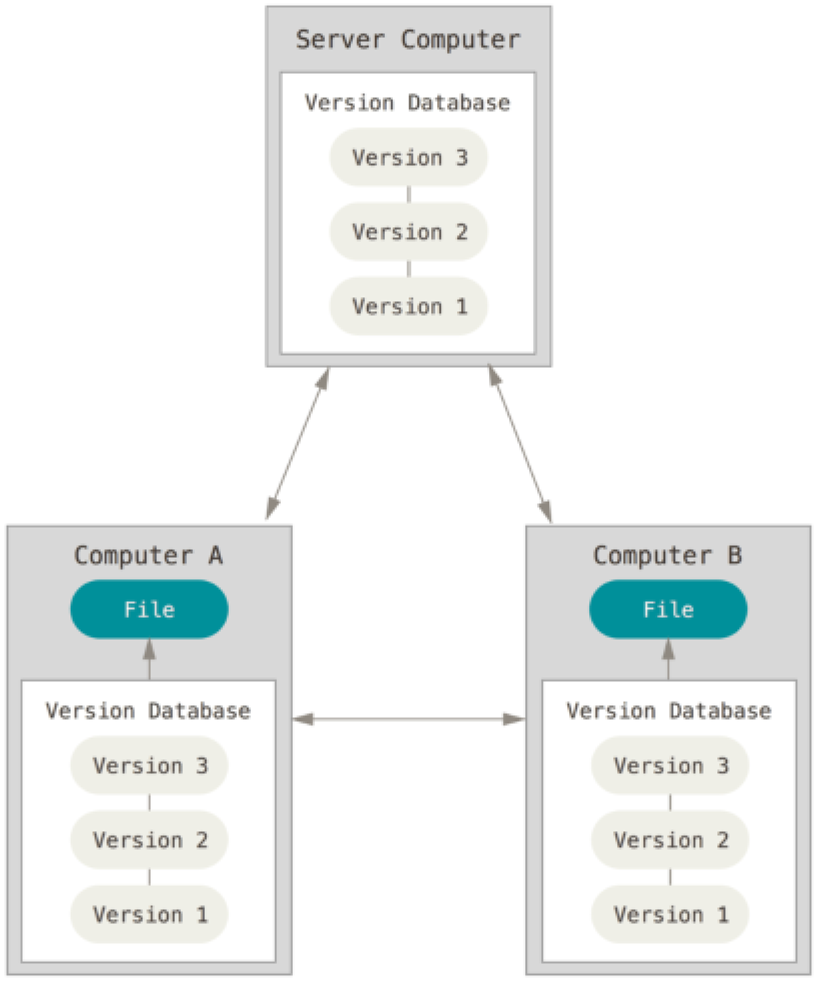
## Branches: A named lineage of commits



## Repository: A collection of all the branches and commits

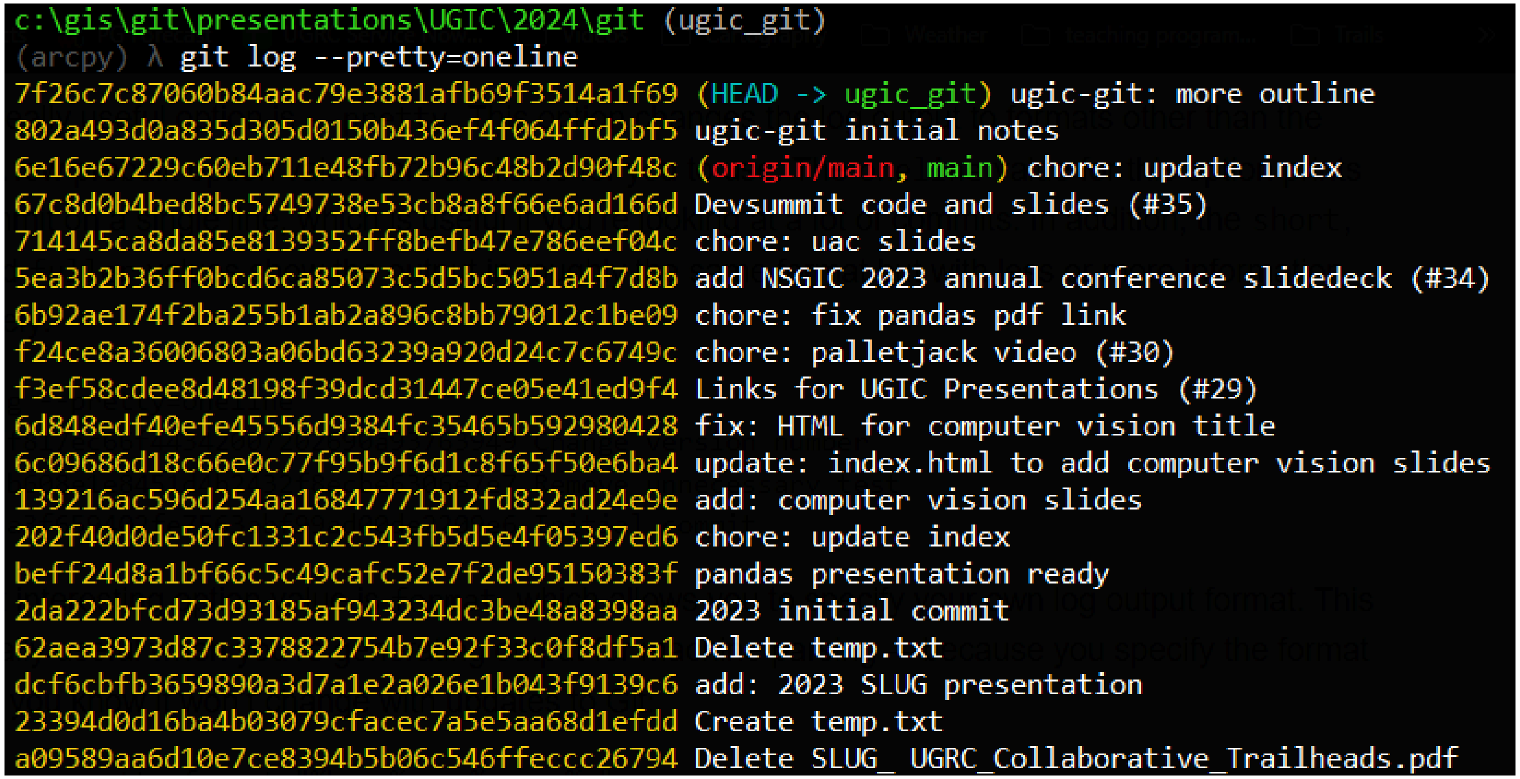


## git is distributed



## Hashes and Immutability

## Every commit has a SHA-1 hash

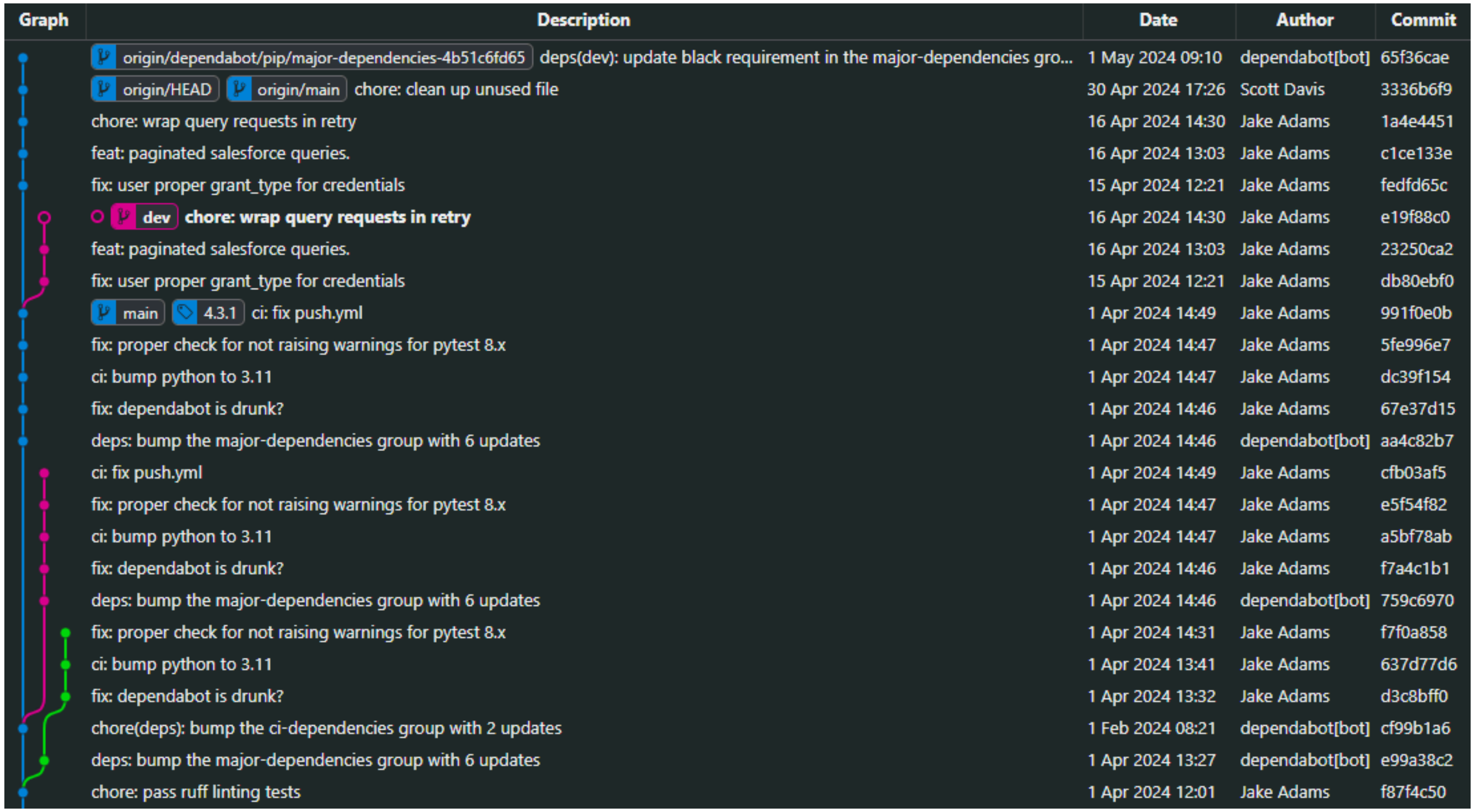


```
c:\gis\git\presentations\UGIC\2024\git (ugic_git)
(arcpy) λ git log --pretty=oneline
7f26c7c87060b84aac79e3881afb69f3514a1f69 (HEAD -> ugic_git) ugic-git: more outline
802a493d0a835d305d0150b436ef4f064ffd2bf5 ugic-git initial notes
6e16e67229c60eb711e48fb72b96c48b2d90f48c (origin/main, main) chore: update index
67c8d0b4bed8bc5749738e53cb8a8f66e6ad166d Devsummit code and slides (#35)
714145ca8da85e8139352ff8befb47e786eef04c chore: uac slides
5ea3b2b36ff0bcd6ca85073c5d5bc5051a4f7d8b add NSGIC 2023 annual conference slidedeck (#34)
6b92ae174f2ba255b1ab2a896c8bb79012c1be09 chore: fix pandas pdf link
f24ce8a36006803a06bd63239a920d24c7c6749c chore: palletjack video (#30)
f3ef58cdee8d48198f39dcd31447ce05e41ed9f4 Links for UGIC Presentations (#29)
6d848edf40efe45556d9384fc35465b592980428 fix: HTML for computer vision title
6c09686d18c66e0c77f95b9f6d1c8f65f50e6ba4 update: index.html to add computer vision slides
139216ac596d254aa16847771912fd832ad24e9e add: computer vision slides
202f40d0de50fc1331c2c543fb5d5e4f05397ed6 chore: update index
beff24d8a1bf66c5c49cafc52e7f2de95150383f pandas presentation ready
2da222bfcd73d93185af943234dc3be48a8398aa 2023 initial commit
62aea3973d87c3378822754b7e92f33c0f8df5a1 Delete temp.txt
dcf6cbfb3659890a3d7a1e2a026e1b043f9139c6 add: 2023 SLUG presentation
23394d0d16ba4b03079cfacec7a5e5aa68d1efdd Create temp.txt
a09589aa6d10e7ce8394b5b06c546ffeccc26794 Delete SLUG_ UGRC_Collaborative_Trailheads.pdf
```

## What goes into the hash?

- The data itself
- Author & committer info
- Timestamp
- The parent commit

# Commits are almost never deleted, just "hidden"



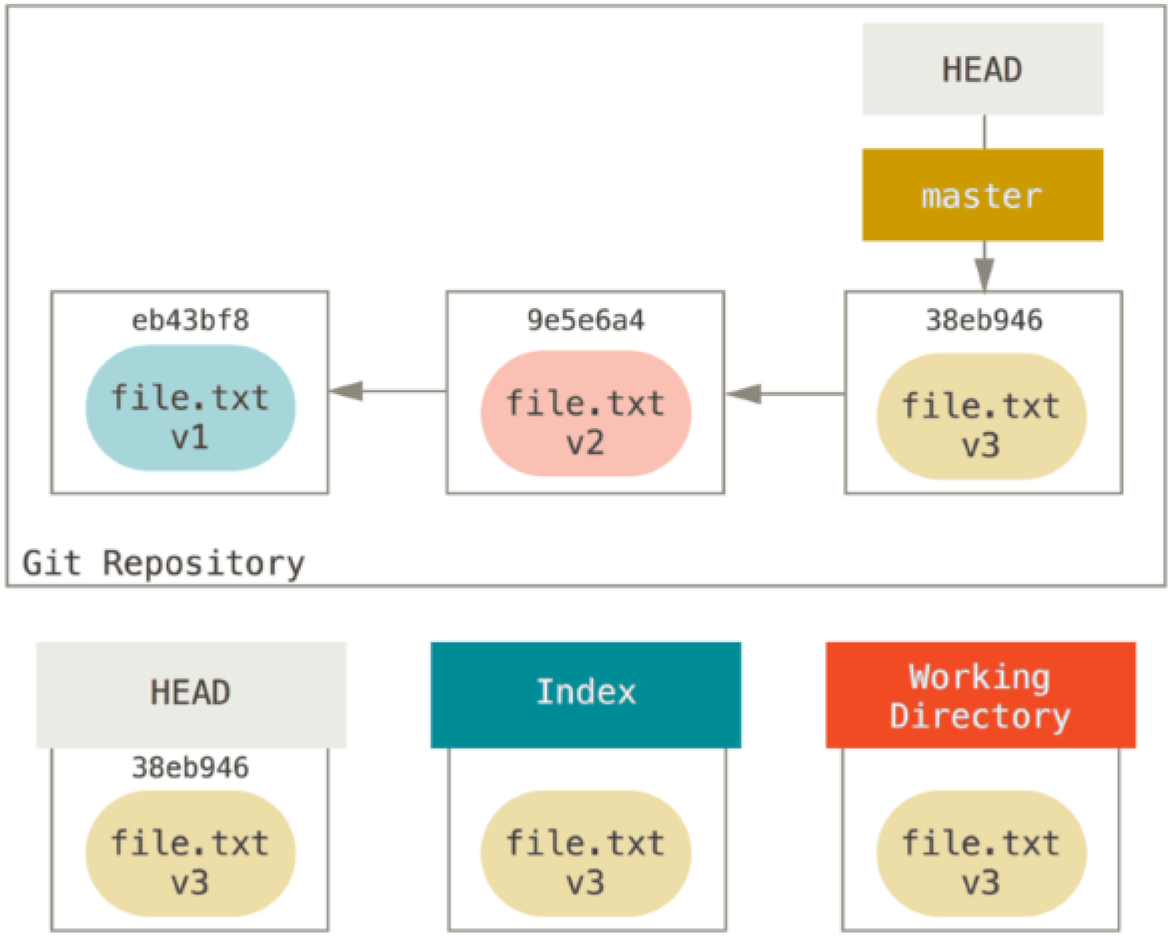
| Graph | Description | Date | Author | Commit |
|---|---|---|---|---|
| | origin/dependabot/pip/major-dependencies-4b51c6fd65 deps(dev): update black requirement in the major-dependencies gro... | 1 May 2024 09:10 | dependabot[bot] | 65f36cae |
| | origin/HEAD origin/main chore: clean up unused file | 30 Apr 2024 17:26 | Scott Davis | 3336b6f9 |
| | chore: wrap query requests in retry | 16 Apr 2024 14:30 | Jake Adams | 1a4e4451 |
| | feat: paginated salesforce queries. | 16 Apr 2024 13:03 | Jake Adams | c1ce133e |
| | fix: user proper grant_type for credentials | 15 Apr 2024 12:21 | Jake Adams | fedfd65c |
| | dev **chore: wrap query requests in retry** | 16 Apr 2024 14:30 | Jake Adams | e19f88c0 |
| | feat: paginated salesforce queries. | 16 Apr 2024 13:03 | Jake Adams | 23250ca2 |
| | fix: user proper grant_type for credentials | 15 Apr 2024 12:21 | Jake Adams | db80ebf0 |
| | main 4.3.1 ci: fix push.yml | 1 Apr 2024 14:49 | Jake Adams | 991f0e0b |
| | fix: proper check for not raising warnings for pytest 8.x | 1 Apr 2024 14:47 | Jake Adams | 5fe996e7 |
| | ci: bump python to 3.11 | 1 Apr 2024 14:47 | Jake Adams | dc39f154 |
| | fix: dependabot is drunk? | 1 Apr 2024 14:46 | Jake Adams | 67e37d15 |
| | deps: bump the major-dependencies group with 6 updates | 1 Apr 2024 14:46 | dependabot[bot] | aa4c82b7 |
| | ci: fix push.yml | 1 Apr 2024 14:49 | Jake Adams | cfb03af5 |
| | fix: proper check for not raising warnings for pytest 8.x | 1 Apr 2024 14:47 | Jake Adams | e5f54f82 |
| | ci: bump python to 3.11 | 1 Apr 2024 14:47 | Jake Adams | a5bf78ab |
| | fix: dependabot is drunk? | 1 Apr 2024 14:46 | Jake Adams | f7a4c1b1 |
| | deps: bump the major-dependencies group with 6 updates | 1 Apr 2024 14:46 | dependabot[bot] | 759c6970 |
| | fix: proper check for not raising warnings for pytest 8.x | 1 Apr 2024 14:31 | Jake Adams | f7f0a858 |
| | ci: bump python to 3.11 | 1 Apr 2024 13:41 | Jake Adams | 637d77d6 |
| | fix: dependabot is drunk? | 1 Apr 2024 13:32 | Jake Adams | d3c8bff0 |
| | chore(deps): bump the ci-dependencies group with 2 updates | 1 Feb 2024 08:21 | dependabot[bot] | cf99b1a6 |
| | deps: bump the major-dependencies group with 6 updates | 1 Apr 2024 13:27 | dependabot[bot] | e99a38c2 |
| | chore: pass ruff linting tests | 1 Apr 2024 12:01 | Jake Adams | f87f4c50 |

Key takeaway: commits are never modified

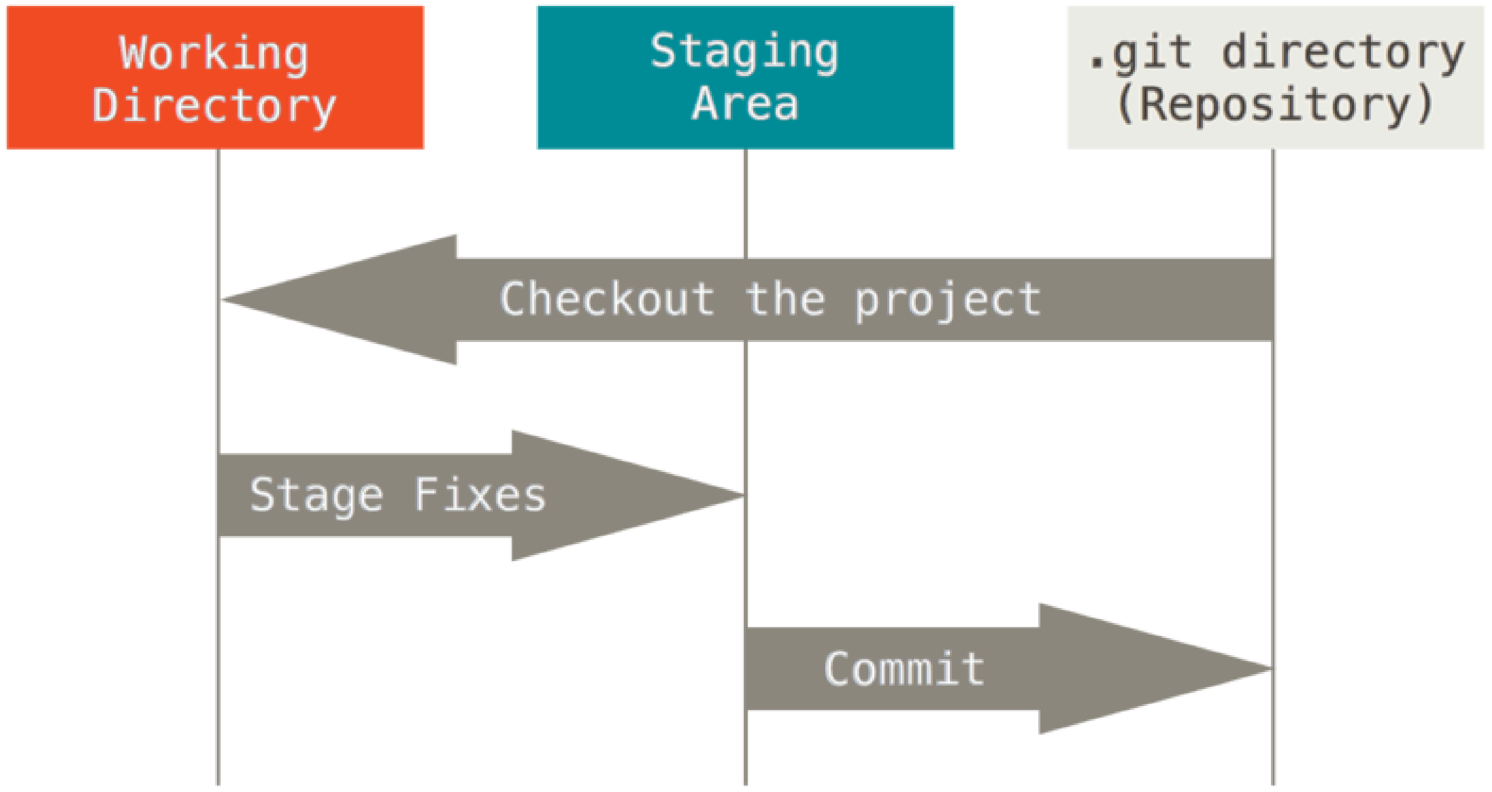
☝️ ☝️ ☝️

You may make a new commit that contains the same blob but the timestamp will be different

## The Four Storage Areas



- **Repository**: All the blobs, trees, and commits as binary files in the `.git` folder
- **Staging Area/Index**: Your proposed next commit
- **Working Directory/Tree**: The current commit "de-blobbed" as a normal folder and files
- **HEAD**: The currently checked out branch



## The same file can exist in all three areas with different content

- **Repository**: All your previous work on the file
- **Index**: Changes you've made and staged for a future commit
- **Working Tree**: Current edits that aren't finalized

## Basic git Operations

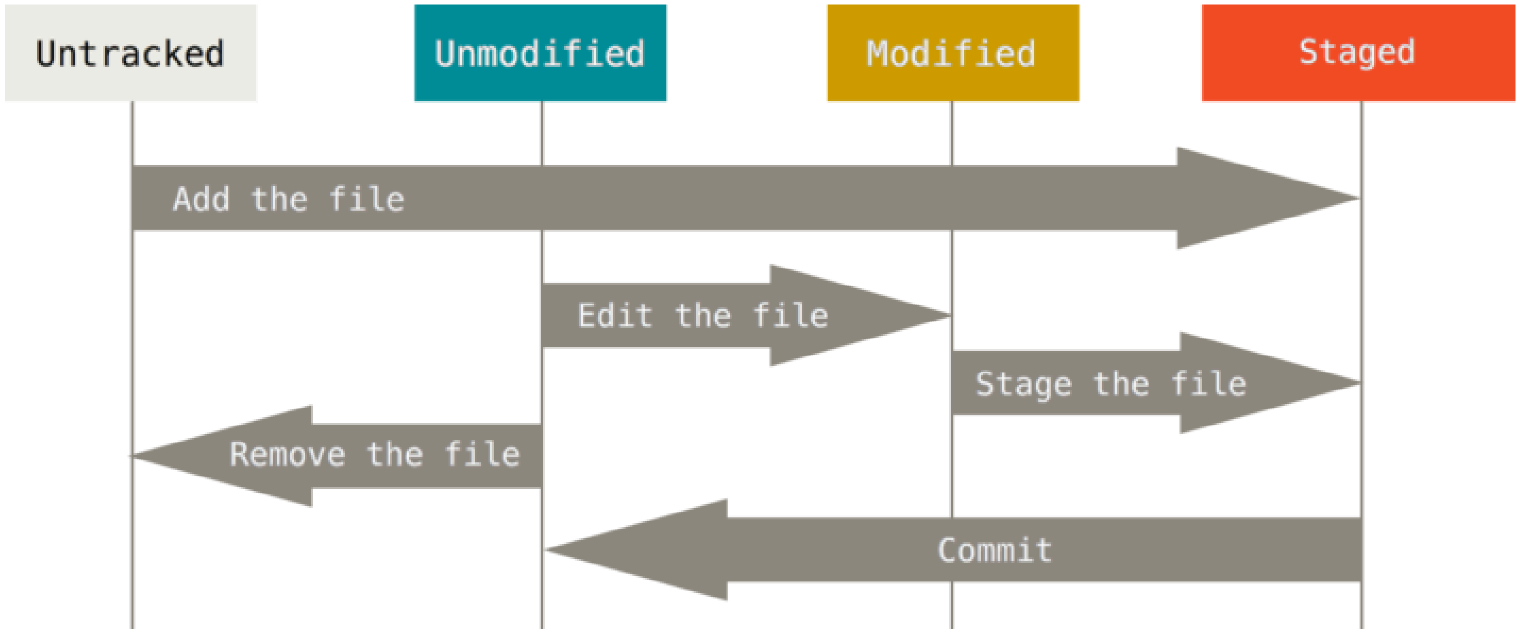
Comitting edits

Branching

Merging

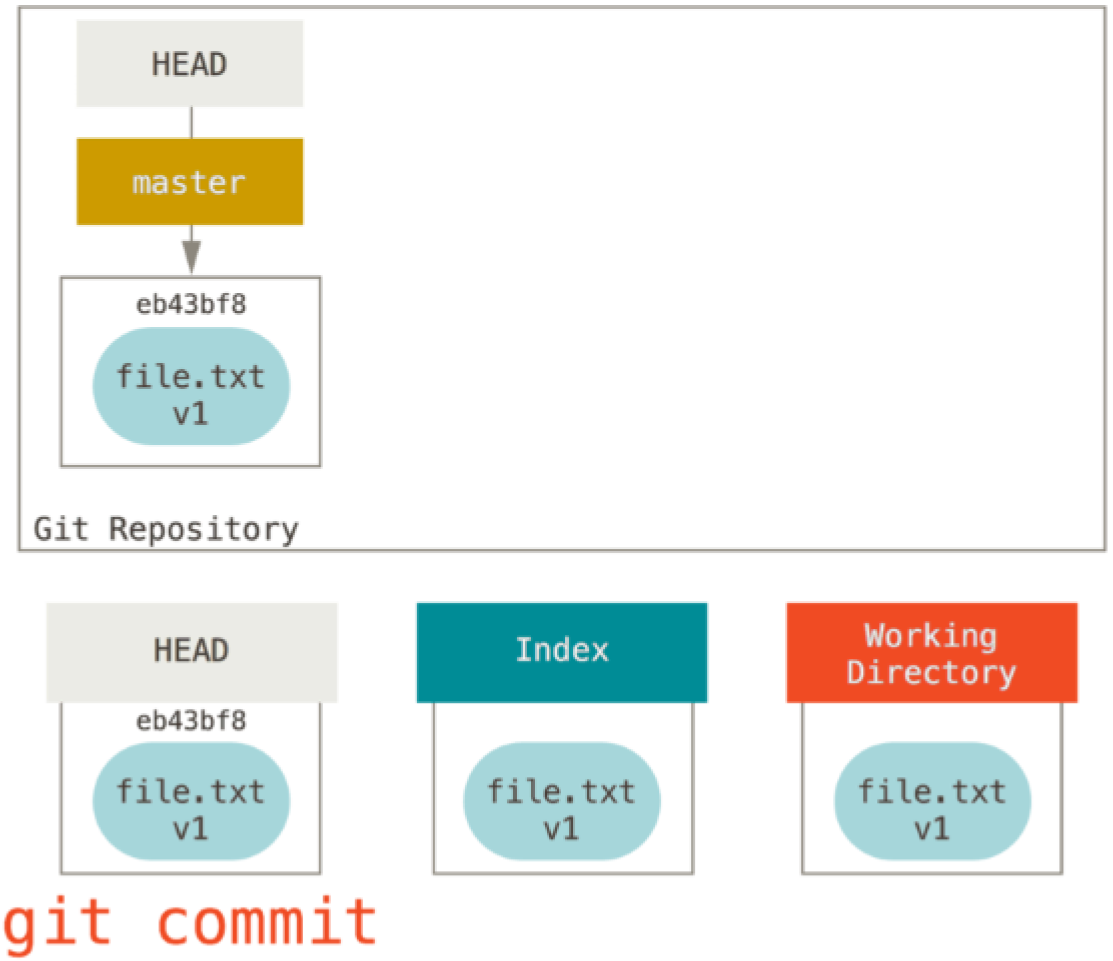
Merge conflicts

# Life of a file

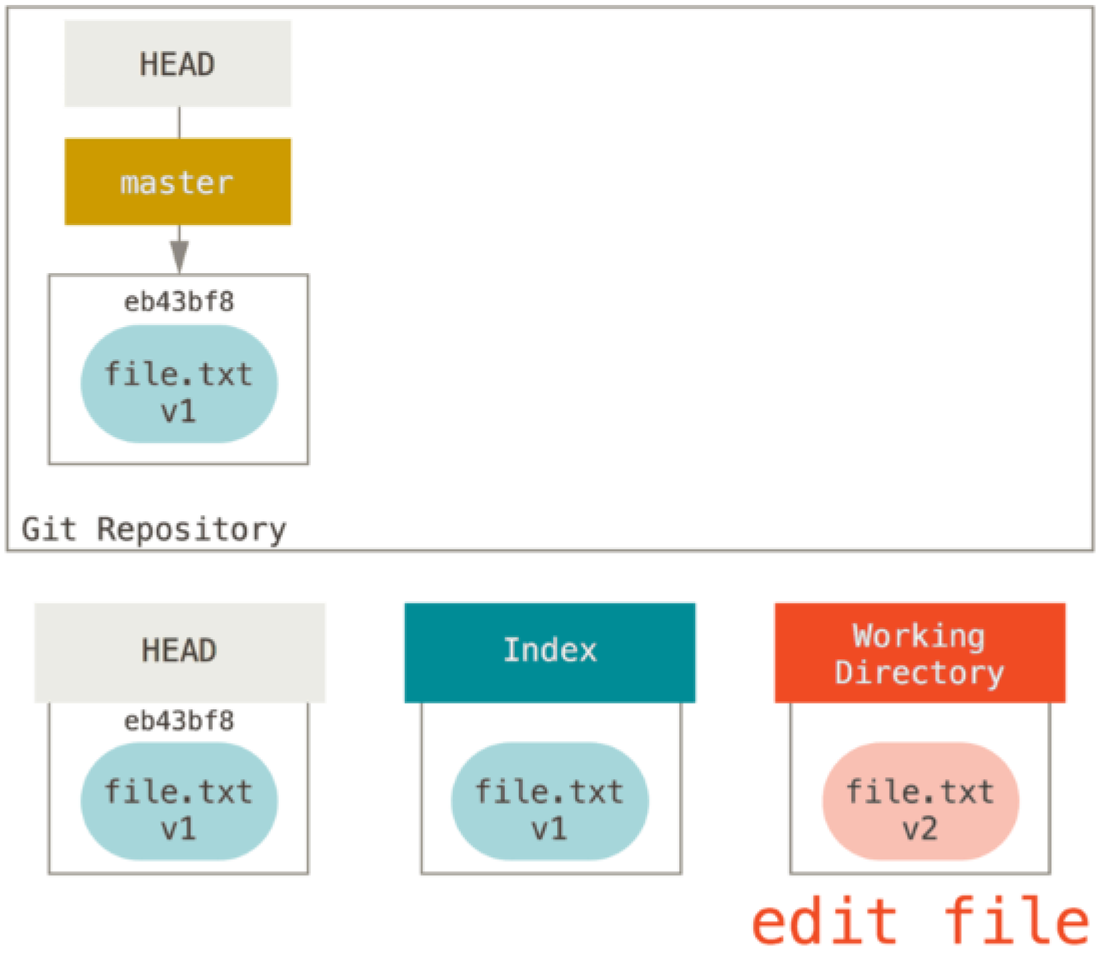


## Committing a file: The bedrock operation

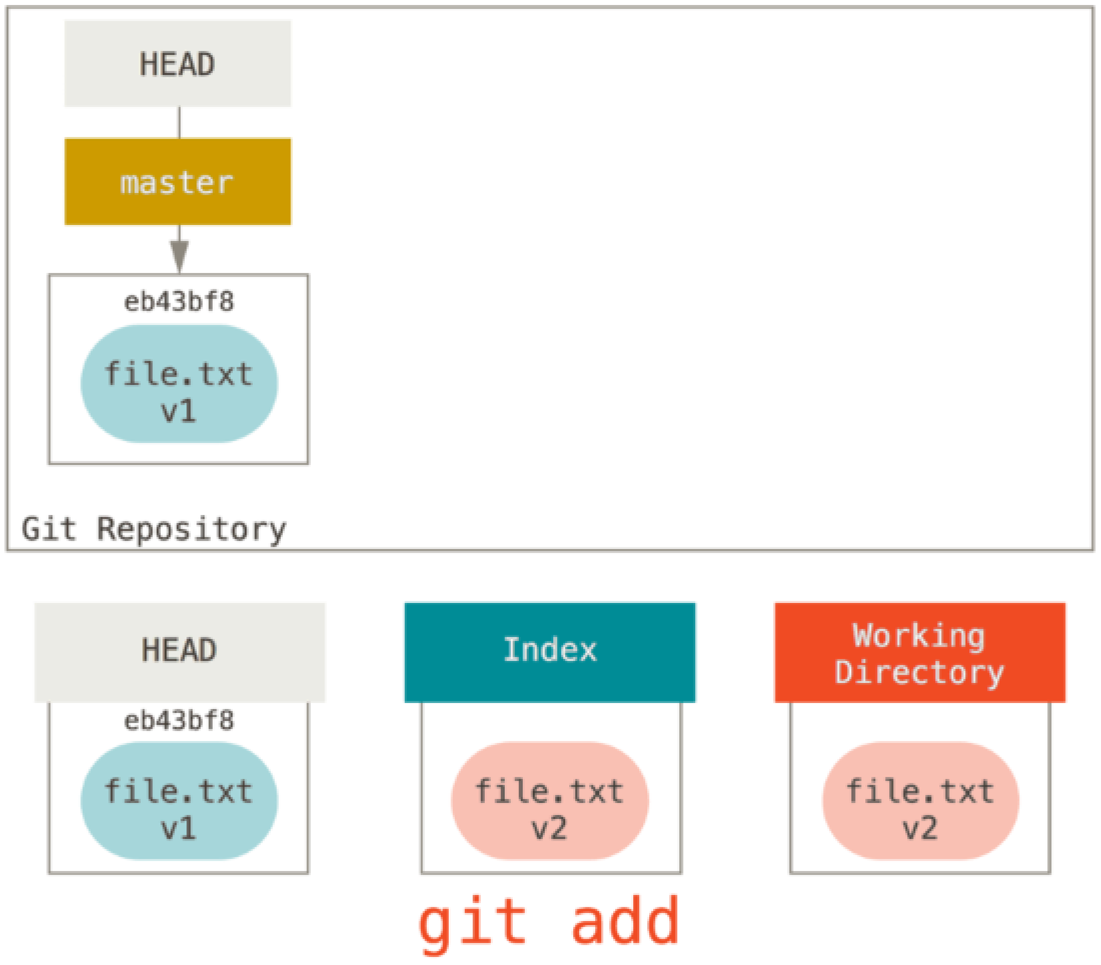
## Starting point: previous work was committed



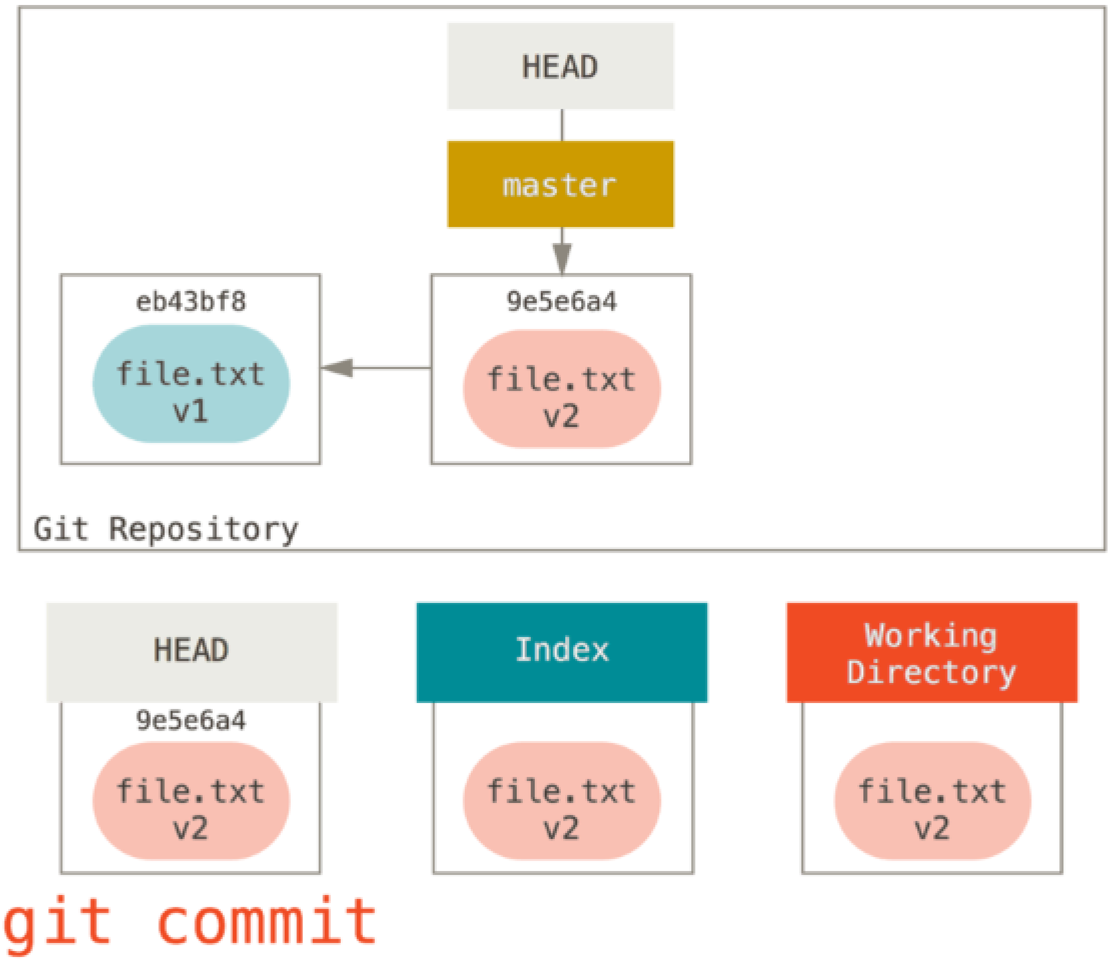
## Make some edits



## Stage your edits



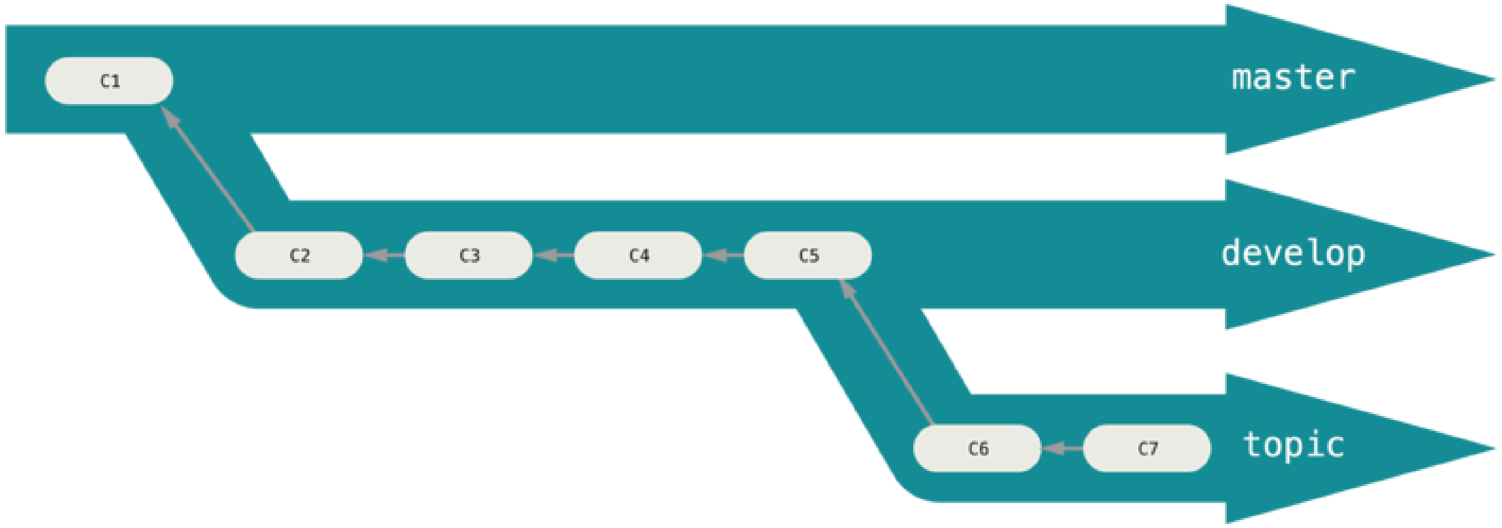
## Commit your edits

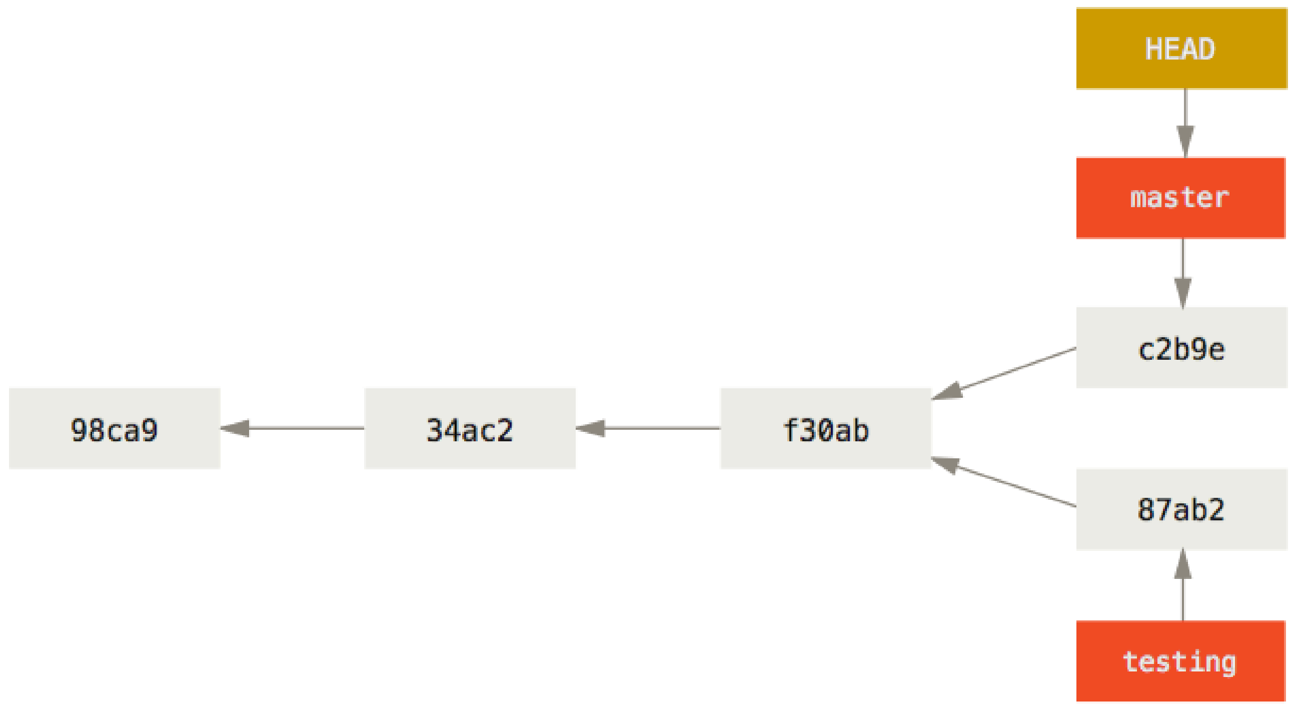


## Branching

Experiment to your hearts content

Branches are cheap- don’t be nervous about using them



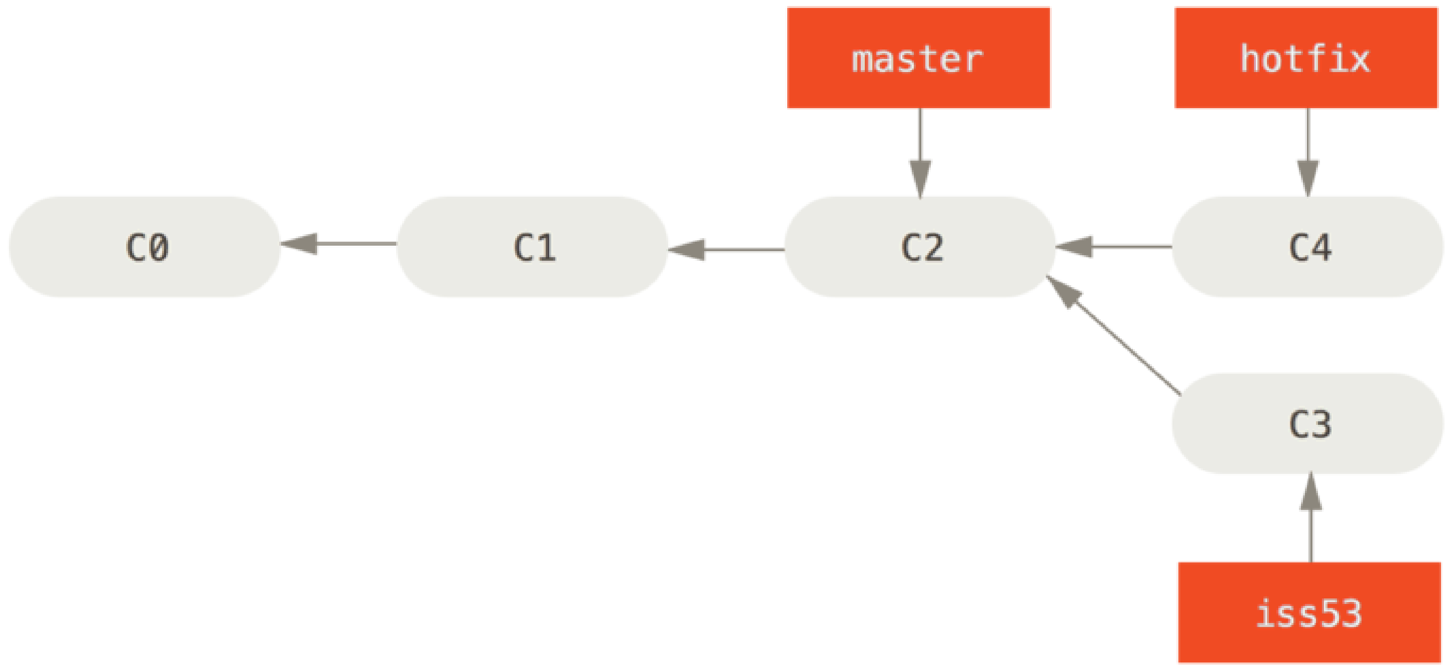

HEAD
master
c2b9e
98ca9
34ac2
f30ab
87ab2
testing

## Checkout: Switch between branches

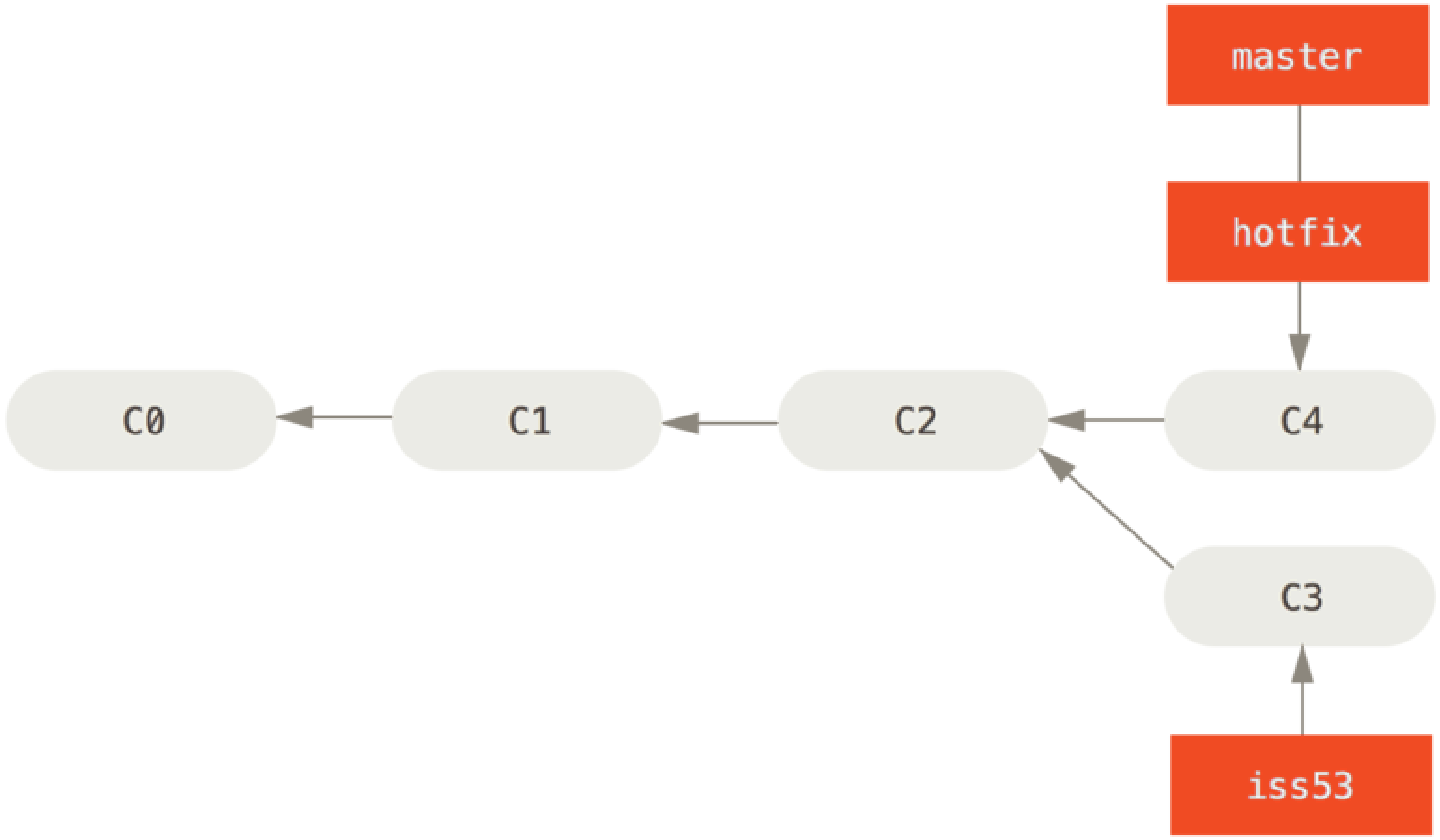
Replaces content in working tree, index, and Head with the last commit in the checked-out branch

## Merging: Bringing it all back together

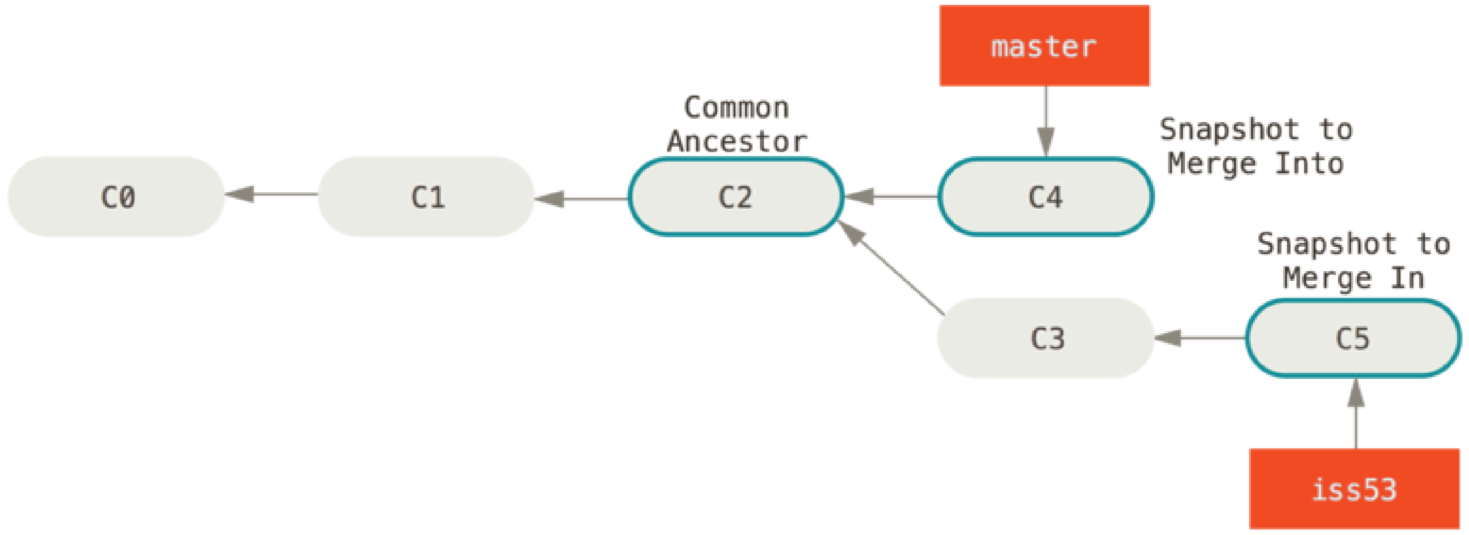
## Fast Forward: Simply move master forward



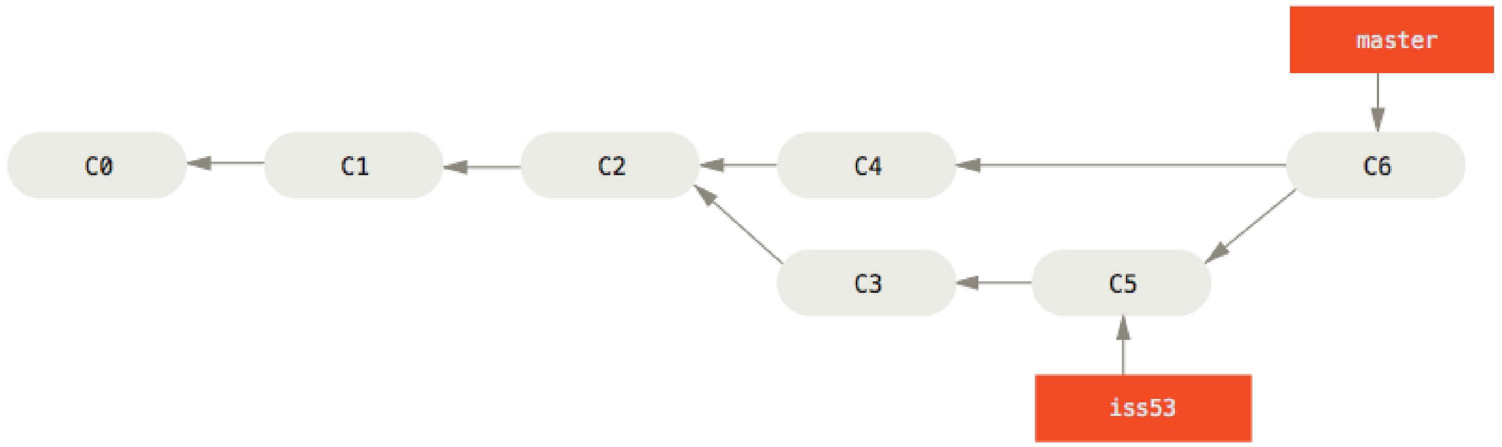
## Fast Forward: Simply move master forward



## Merge Commit: One commit, two ancestors



## Merge Commit: One commit, two ancestors



## Conflict is inevitable

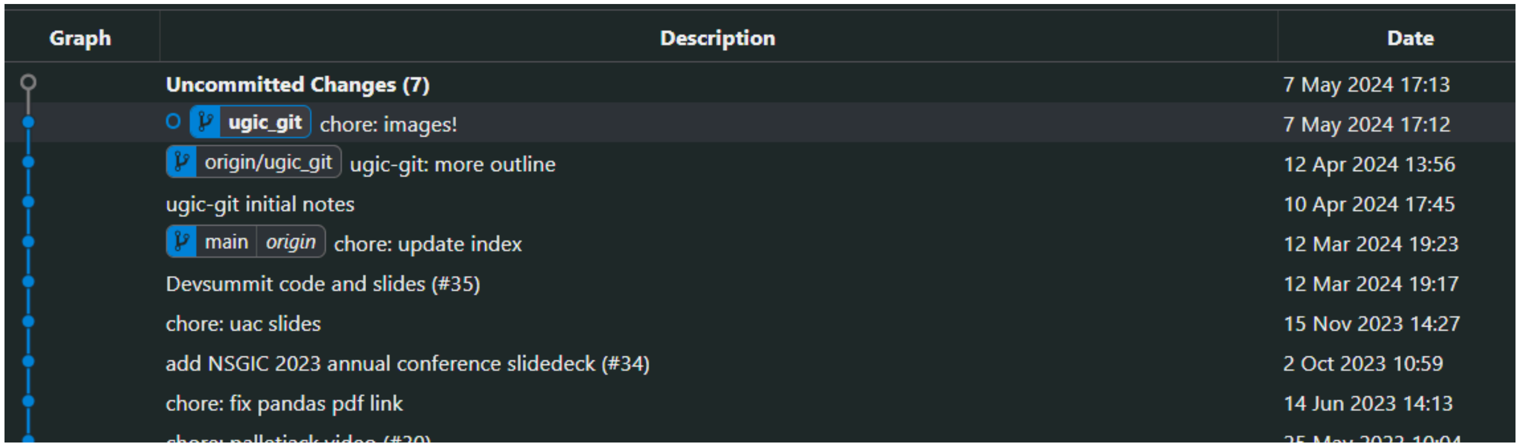
Merge conflict markers

Edit, save, commit, continue

## GitHub: A cloud git server with collaboration tools

GitHub

## Git is distributed: Remotes



| Graph | Description | Date |
|---|---|---|
| | **Uncommitted Changes (7)** | 7 May 2024 17:13 |
| | **ugic_git** chore: images! | 7 May 2024 17:12 |
| | origin/ugic_git ugic-git: more outline | 12 Apr 2024 13:56 |
| | ugic-git initial notes | 10 Apr 2024 17:45 |
| | main *origin* chore: update index | 12 Mar 2024 19:23 |
| | Devsummit code and slides (#35) | 12 Mar 2024 19:17 |
| | chore: uac slides | 15 Nov 2023 14:27 |
| | add NSGIC 2023 annual conference slidedeck (#34) | 2 Oct 2023 10:59 |
| | chore: fix pandas pdf link | 14 Jun 2023 14:13 |

### Syncing work with fetch, pull, and push

- **fetch**: Download all the commits from the remote that are not in your local repo
- **pull**: Do a fetch and then merge your branch into the latest new commit (usually just a fast-forward)
- **push**: Send your commits to the remote

### Always do a fetch or pull before starting to work locally!

## GitHub development model

1. Create repo on GitHub
2. Clone it to your local machine
3. Create a local branch and commit to that
4. Push your branch to GitHub
5. Create Pull Request
6. Rebase and merge into GitHub main branch, delete GitHub branch
7. Pull GitHub main into local main
8. Delete local branch

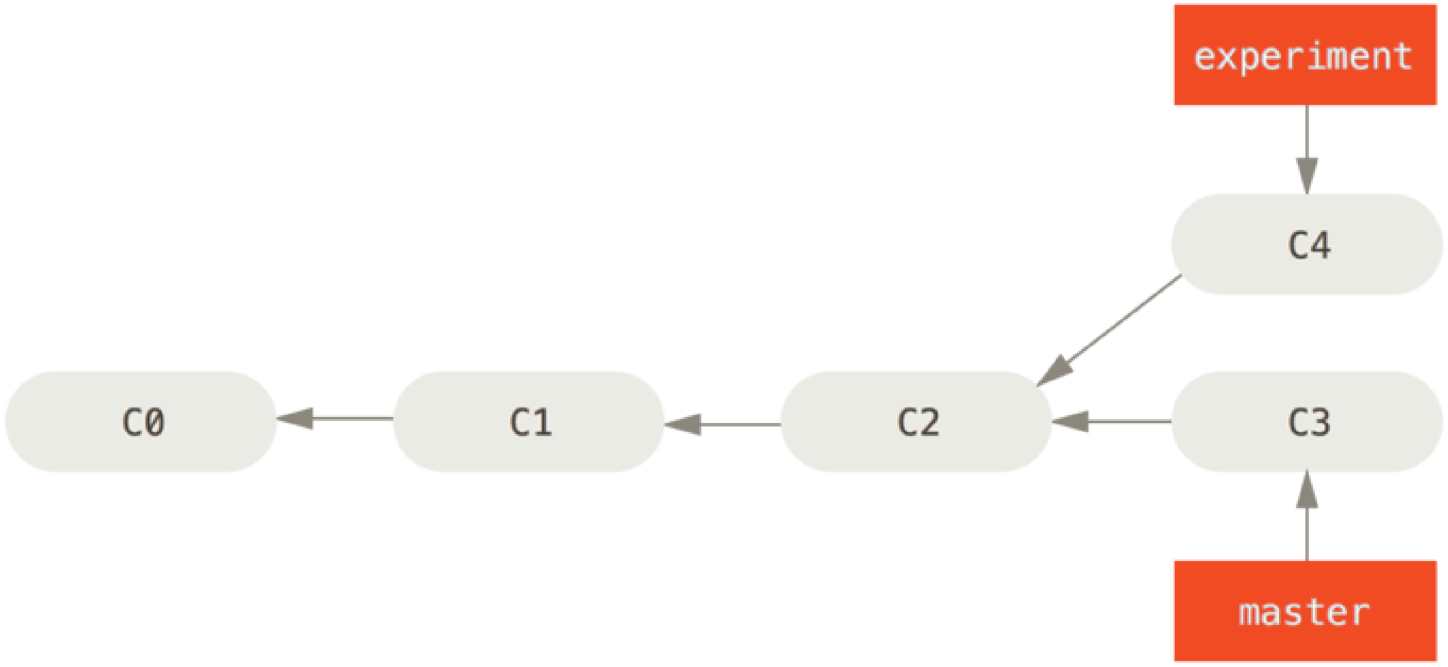
## Pull requests

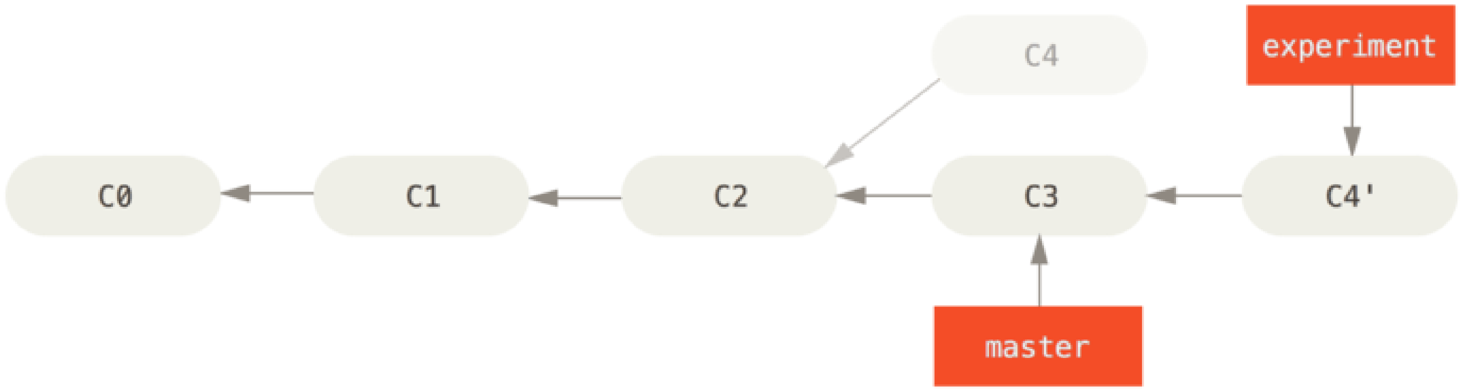
## Issue tracking

Stick a fork in it

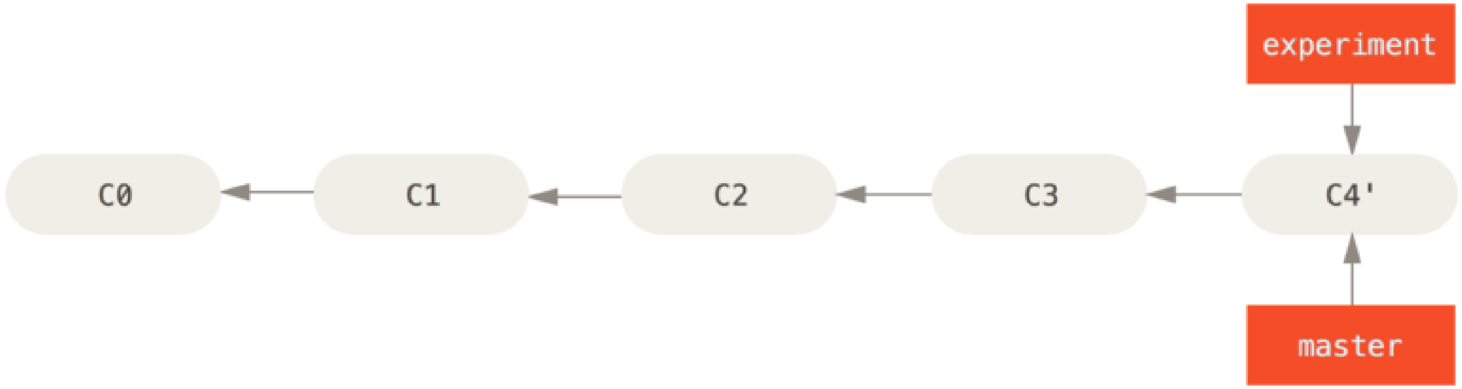
## Advanced git: Rebase, Reset, and Recover

## Rebase: Rewriting history

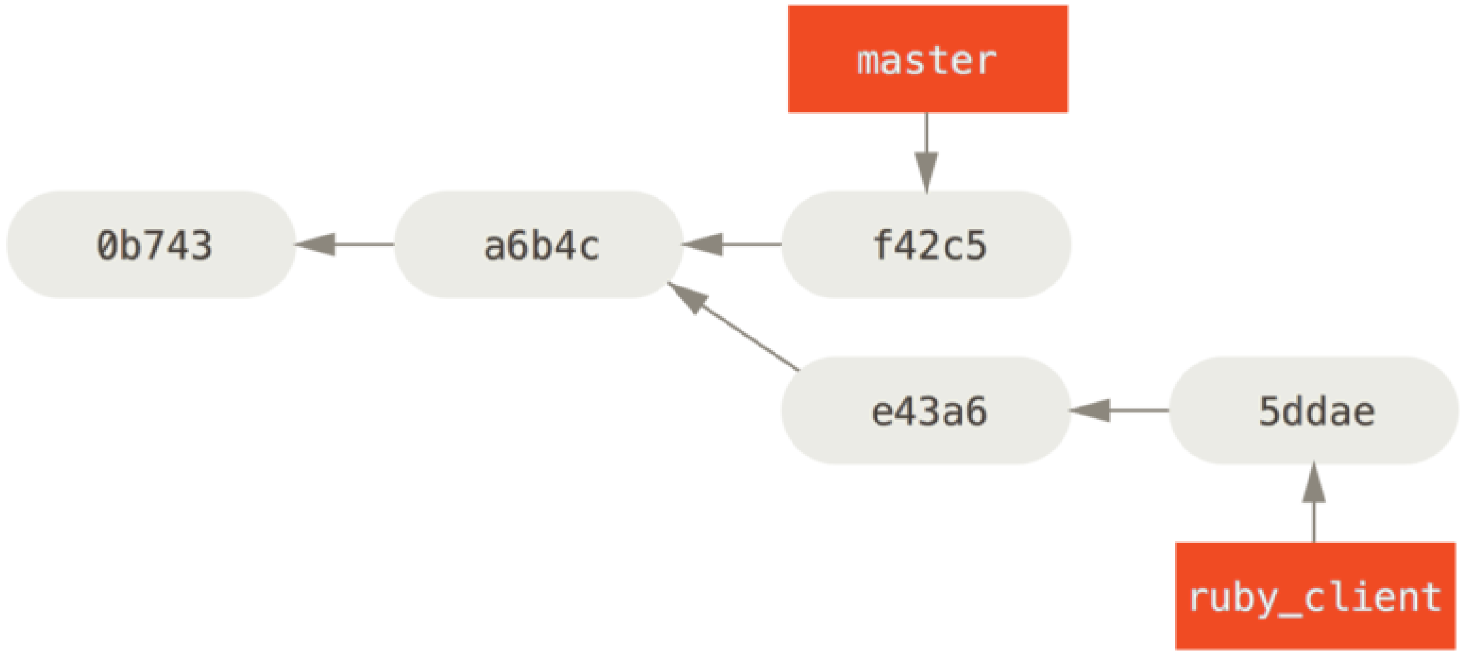

experiment
C4
C0
C1
C2
C3
master

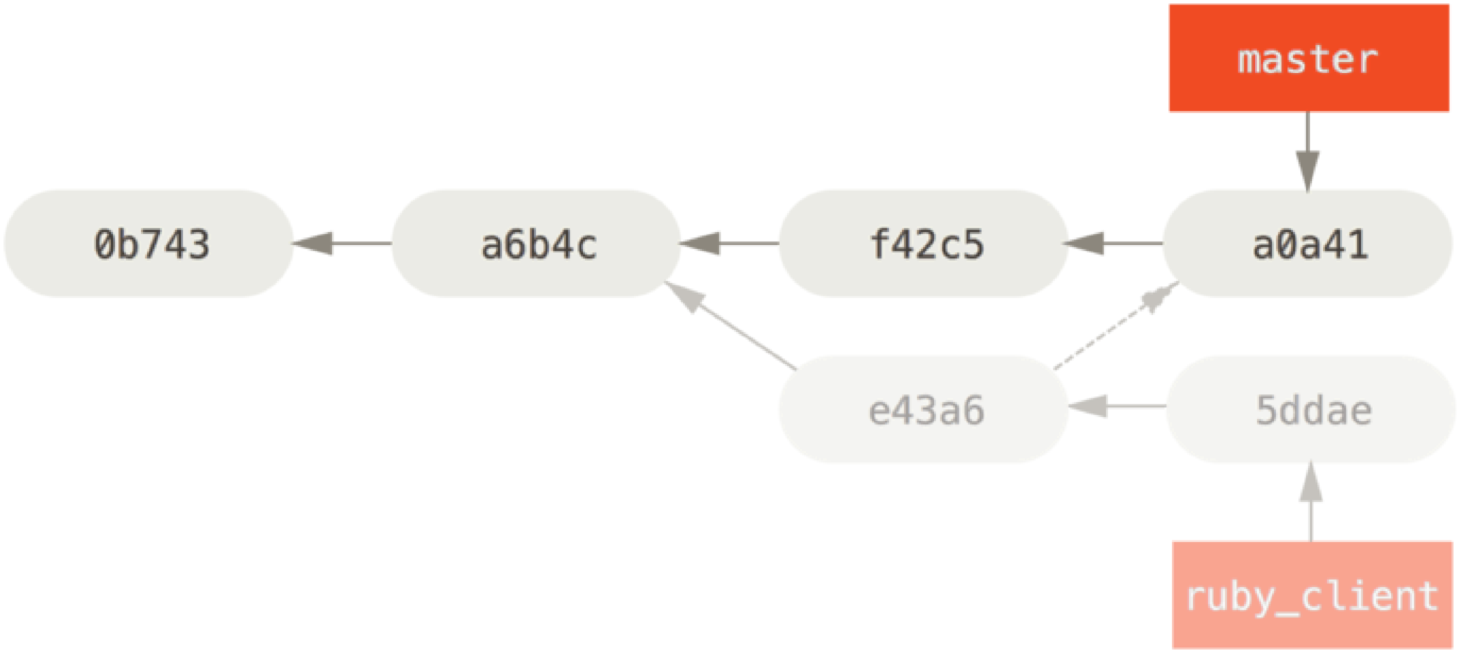

C4
experiment
C0
C1
C2
C3
C4'
master

## Rebase and merge: ”Clean” merges


experiment
C0
C1
C2
C3
C4'
master

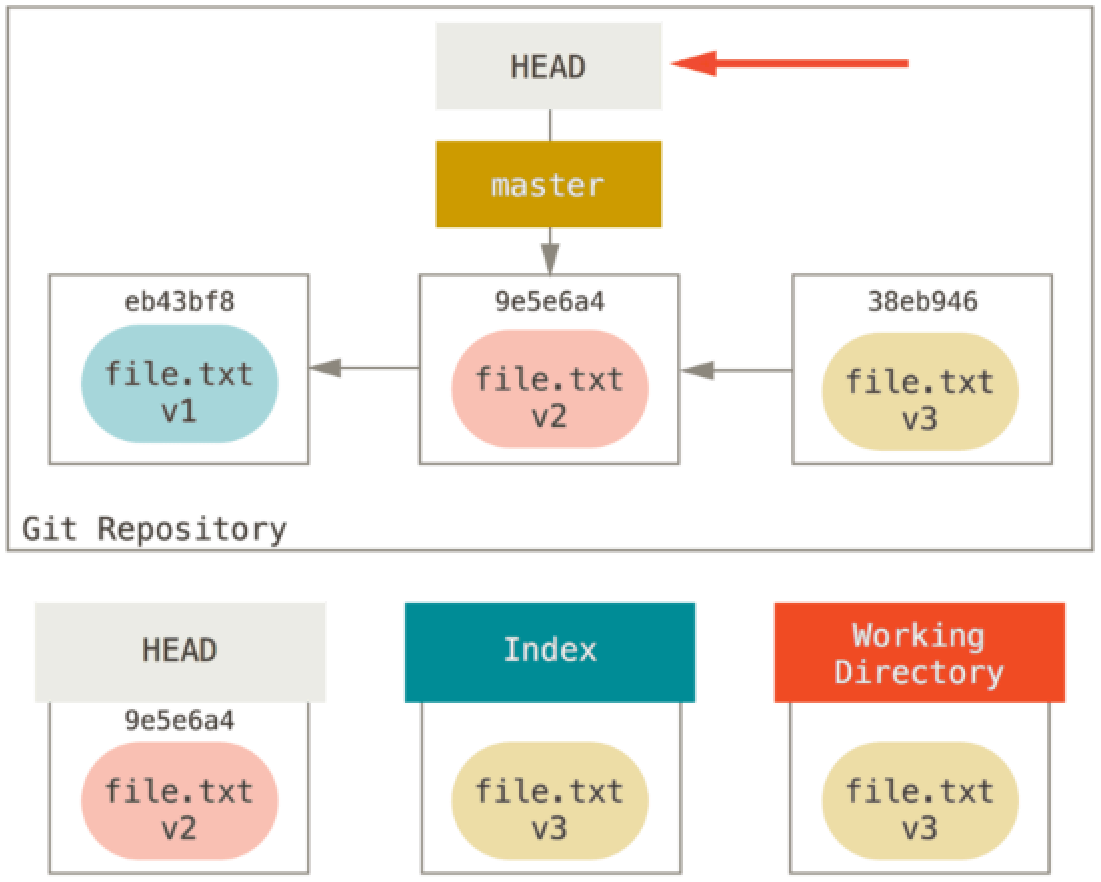
## Cherry pick: Just one, please


master
0b743
a6b4c
f42c5
e43a6
5ddae
ruby_client


master
0b743
a6b4c
f42c5
a0a41
e43a6
5ddae
ruby_client

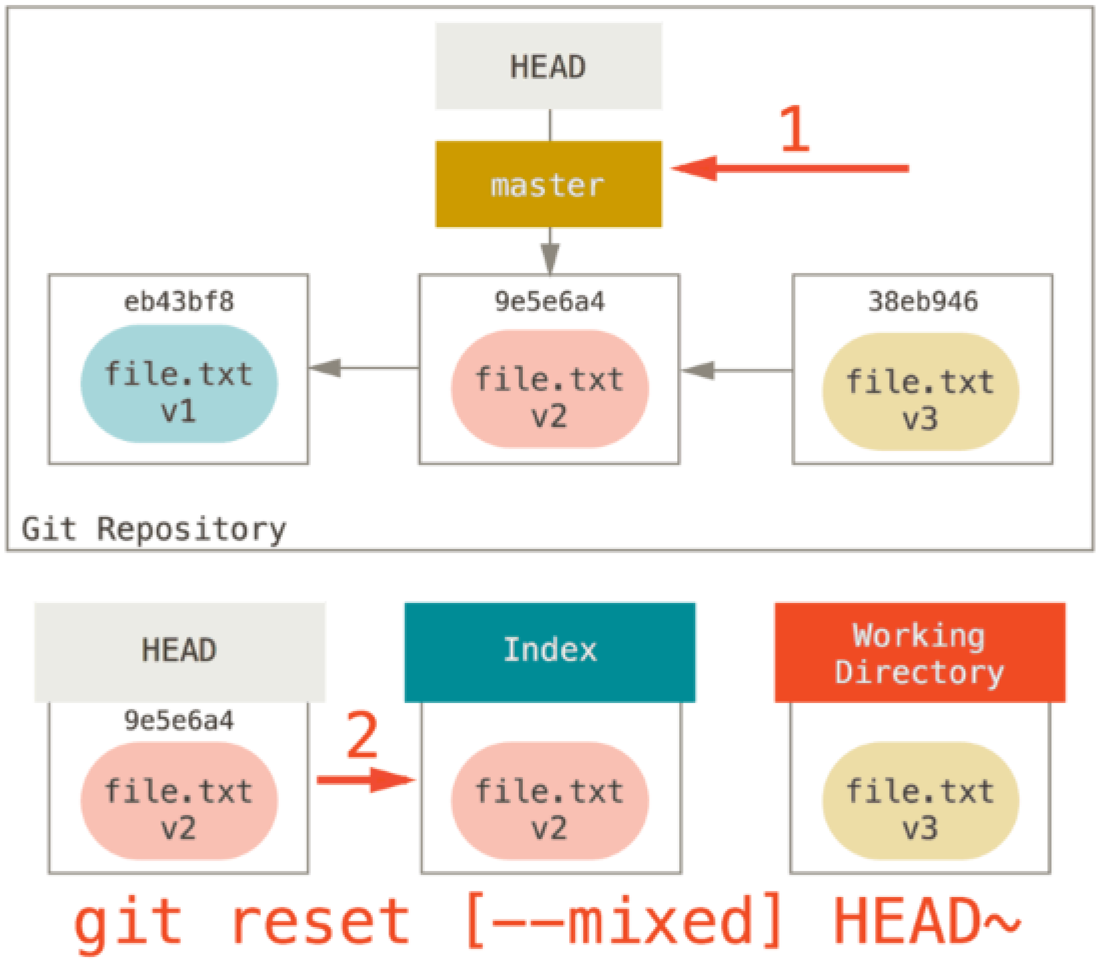
## Reset: The ultimate ctrl-z

## Soft: Move HEAD back, leave changes staged

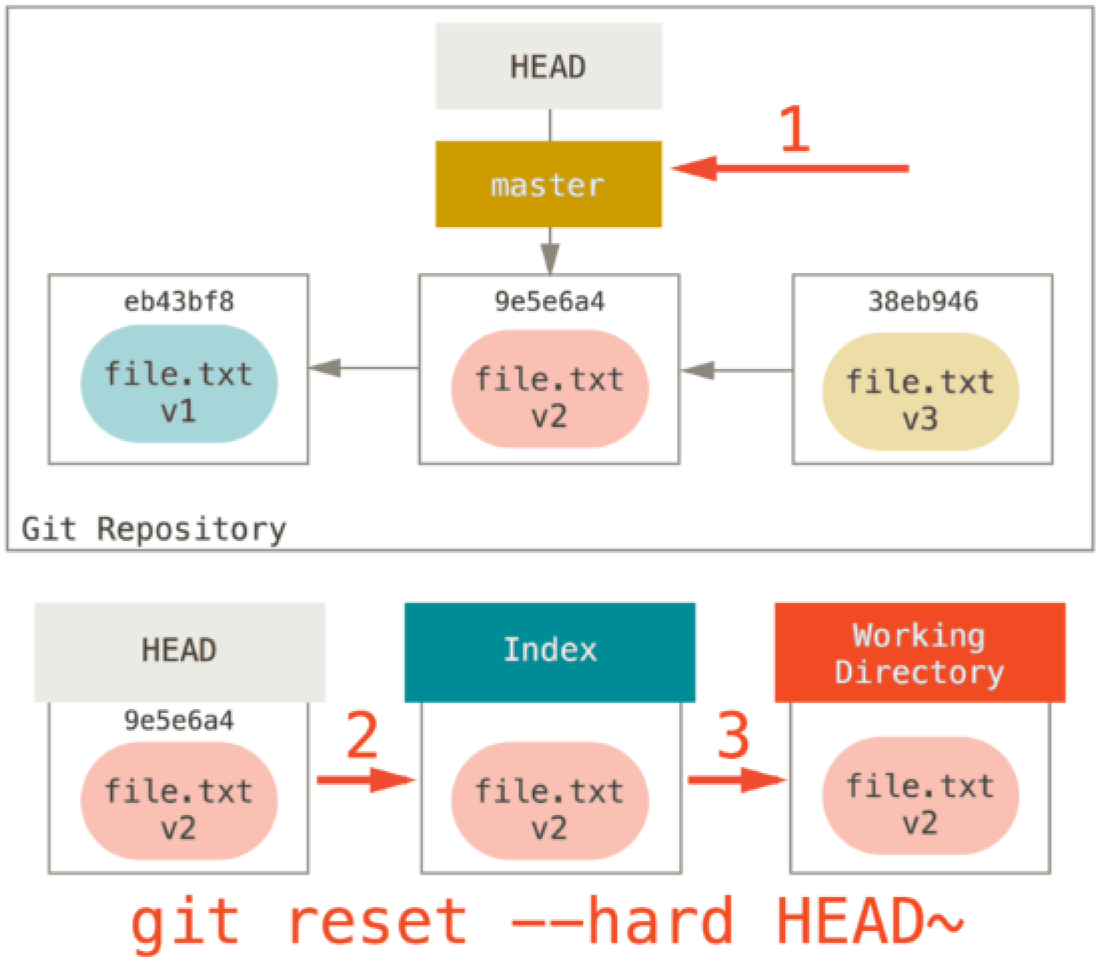


```
git reset --soft HEAD~
```

## Mixed (defualt): Move HEAD, unstage changes



## Hard: Blow everything away



```
git reset --hard HEAD~
```

## Which reset should I use?

I want to completely abandon my current line of work and pretend it never happened

- hard
- (or really just checkout a new branch at the last point you want to continue from)

I don't have any work in progress or anything I want to keep, I just want to point my branch at a different commit

- hard

I did a commit but I want to go back and change something about my edits without adding an extra commit in the repo tree

- mixed

I did a commit but I want to go back and make and stage more edits in addition to my original edits or change the commit message

- soft
- also `git commit --amend` for just the last commit

## Recovery: Commits are loyal friends, always there when you need them

```
git reflog
```

git-graph: include commits mentioned by reflog

## Help! I committed a password!

Check out git-filter-repo

## Resources

The git docs

The git book

Stack Exchange

YouTube

Pluralsight