Git and GitHub

Version Control and Social Coding

Dr Lee A. Christie

@javaxnerd

leechristie.com

lee@leechristie.com

xkcd.com/1597

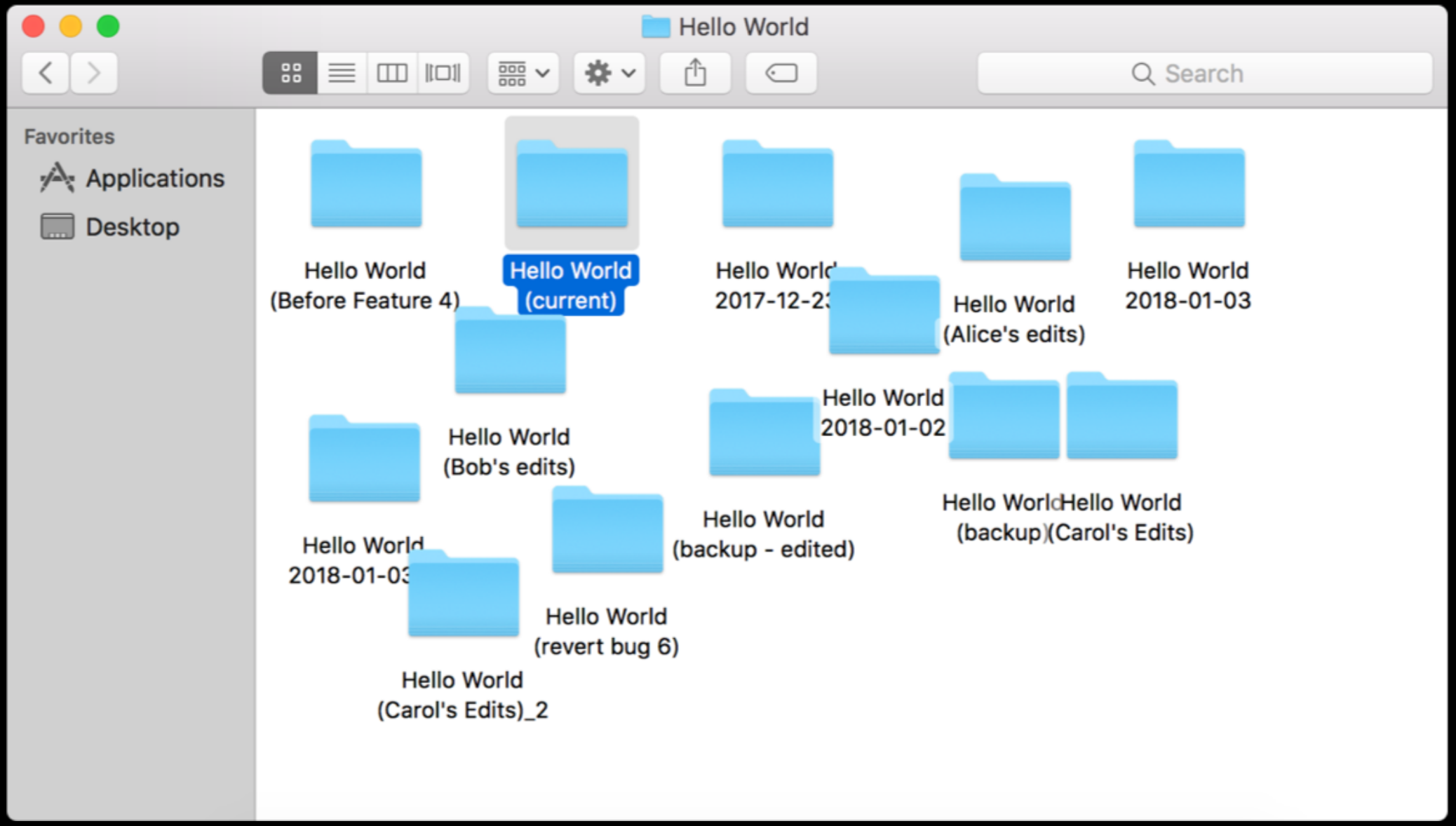
Why Version Control?

Collaborate with multiple developers

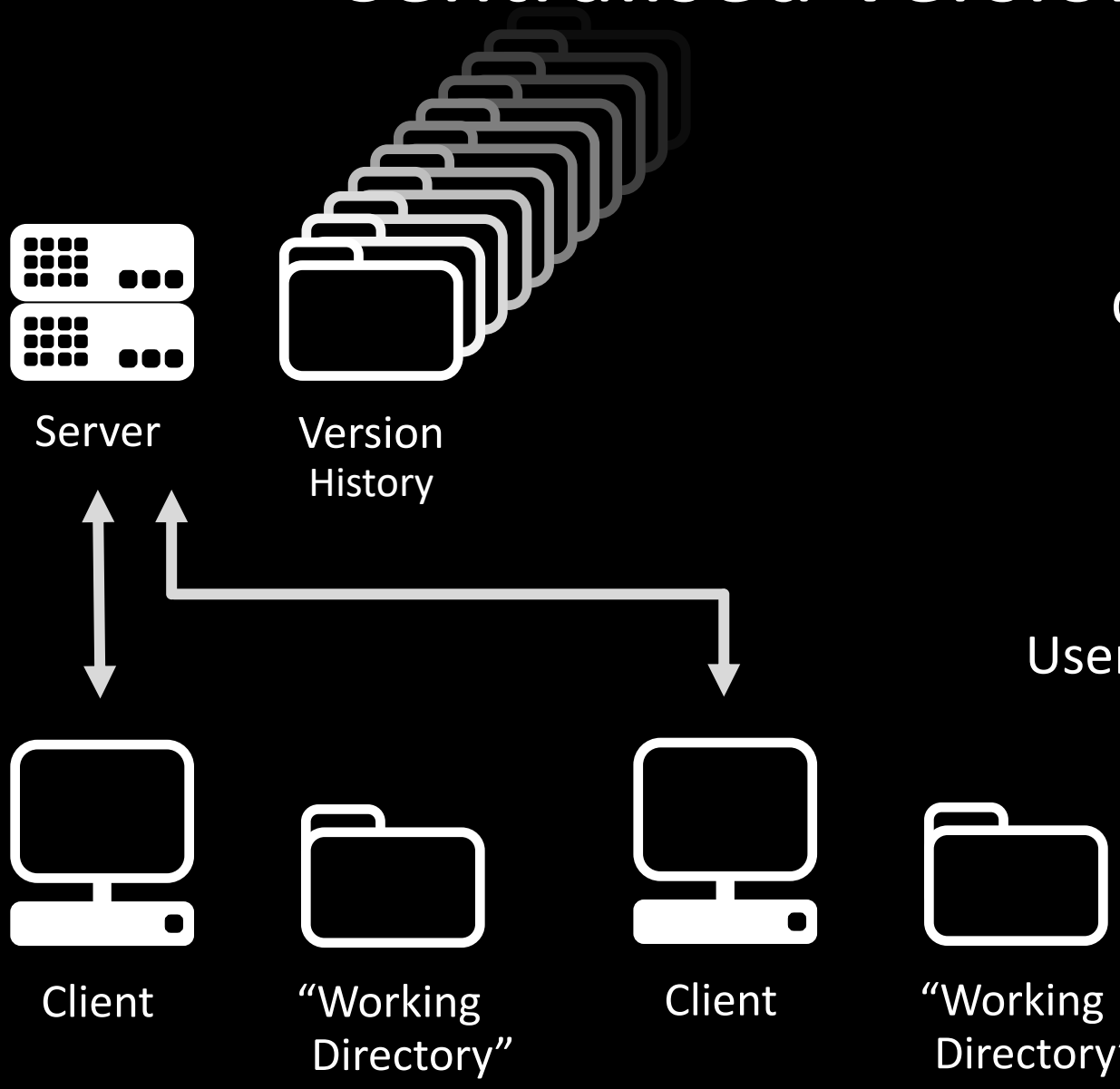
Track changes to project history

Revert changes

A Less-Than Optimal Solution



Centralised Version Control



Server tracks history

Clients "check out" a current copy

Merge changes back to server

Users require server account permissions

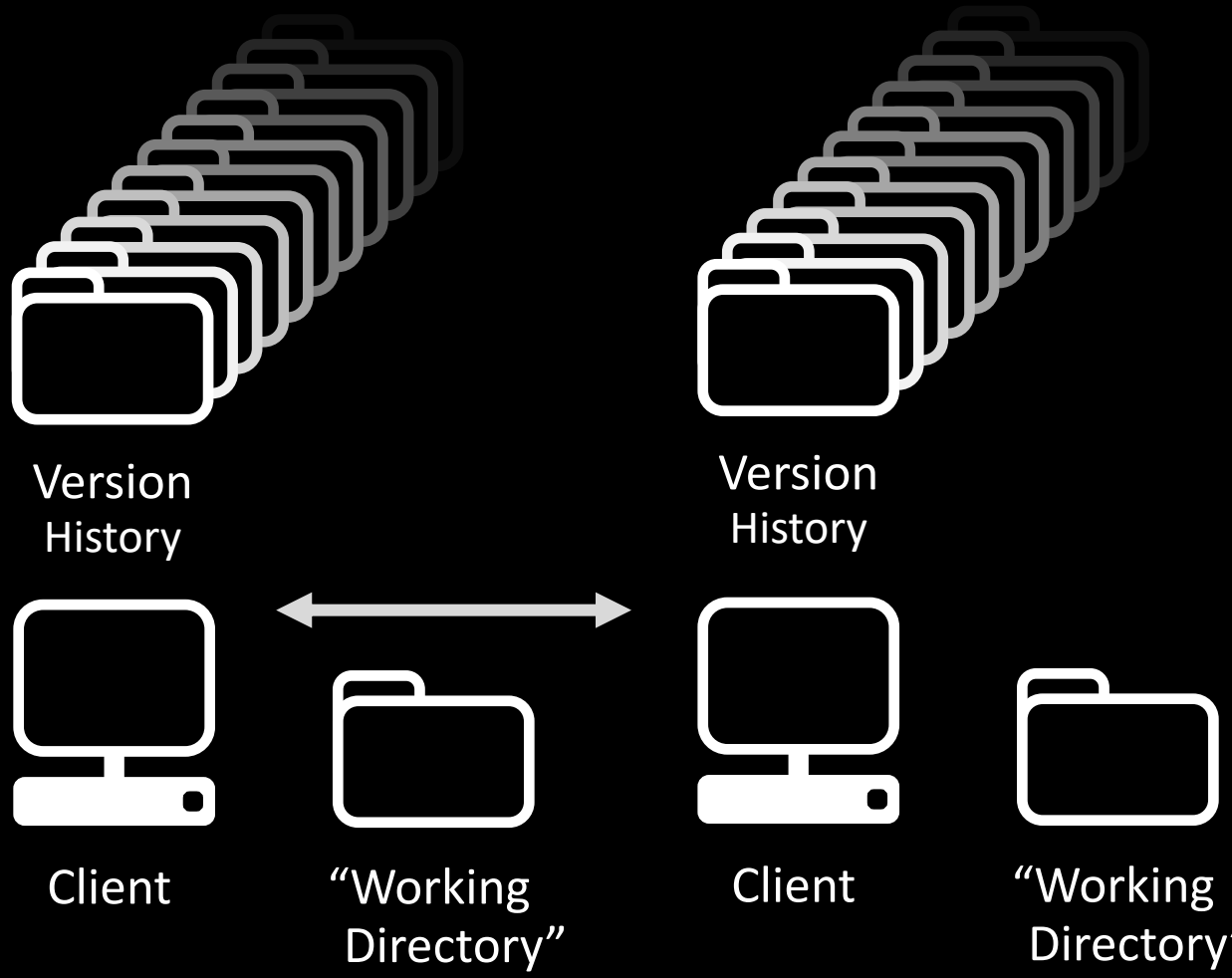
Examples

CVS

Subversion

Perforce

Distributed Version Control



Clients keep a version history

Commit/revert while offline

Re-sync/merge histories

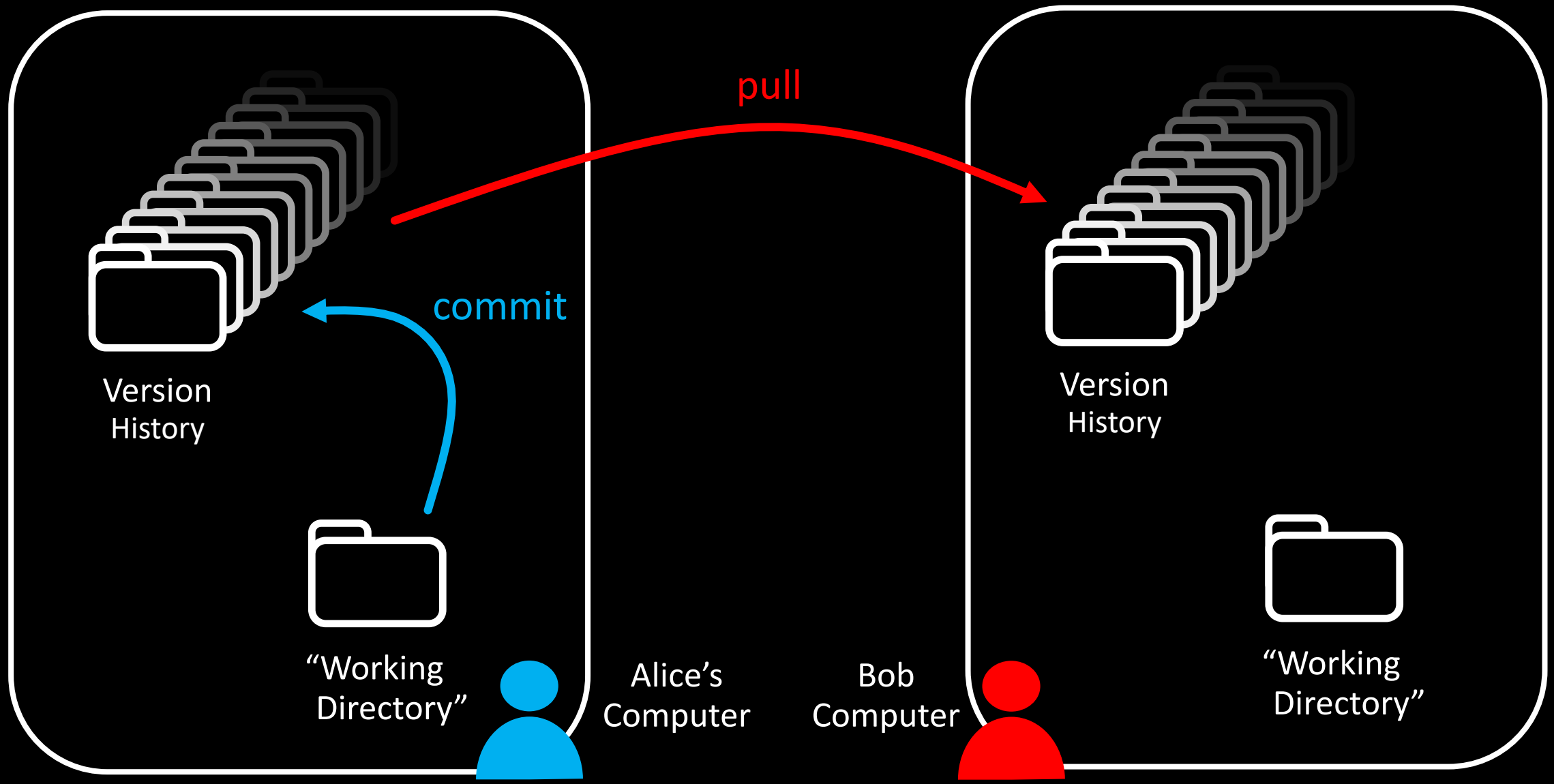
No server accounts needed

Examples

Mercurial

Git

Distributed Version Control



What is GitHub?

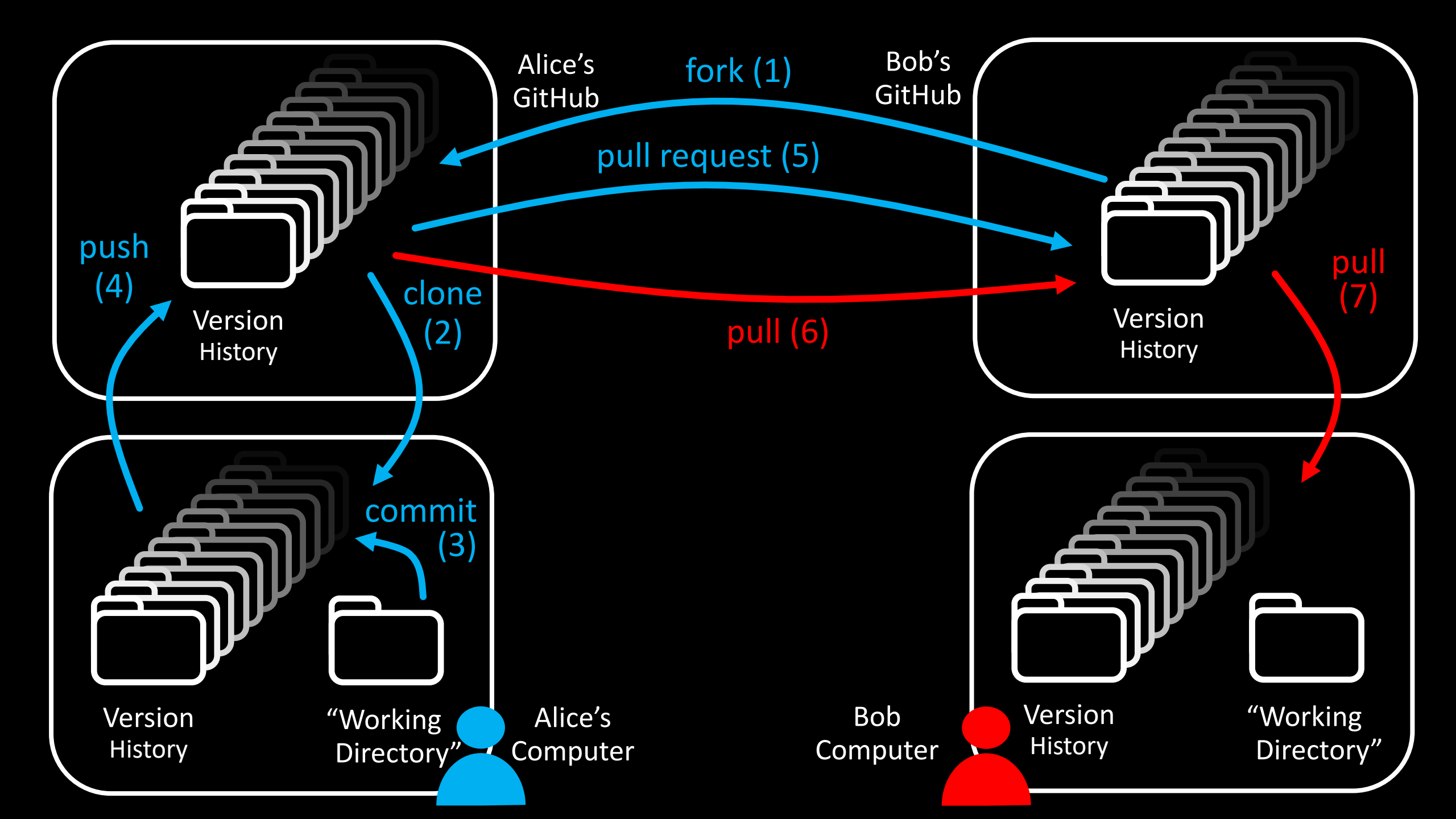
Web hosting for Git repos

Project management tools (Bug tracking, wiki, etc.)

Collaboration via pull requests

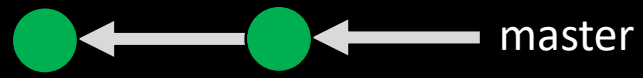
Public / OSS projects hosted free

Private projects: $7/month (**free for students with the Student Developer Pack**)


Alice's GitHub
fork (1)
Bob's GitHub
pull request (5)
push (4)
clone (2)
Version History
pull (6)
pull (7)
Version History
commit (3)
Version History
"Working Directory"
Alice's Computer
Bob Computer
Version History
"Working Directory"

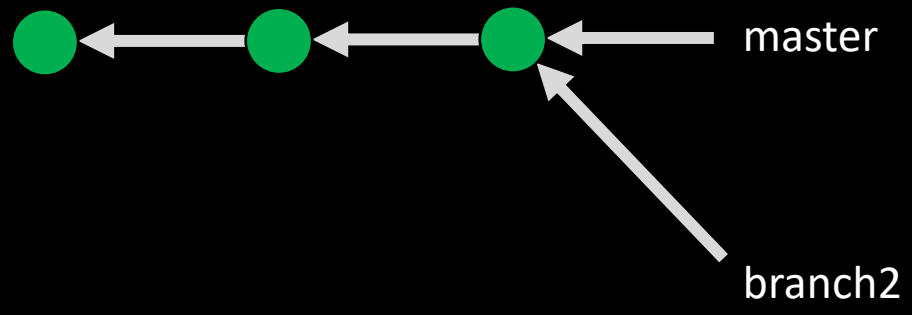
What Does a Git Repo Really Look Like?

What Does a Git Repo Really Look Like?

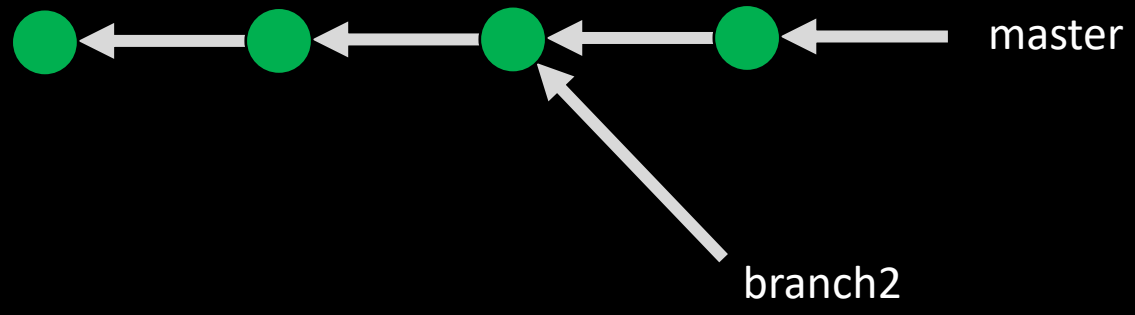


What Does a Git Repo Really Look Like?

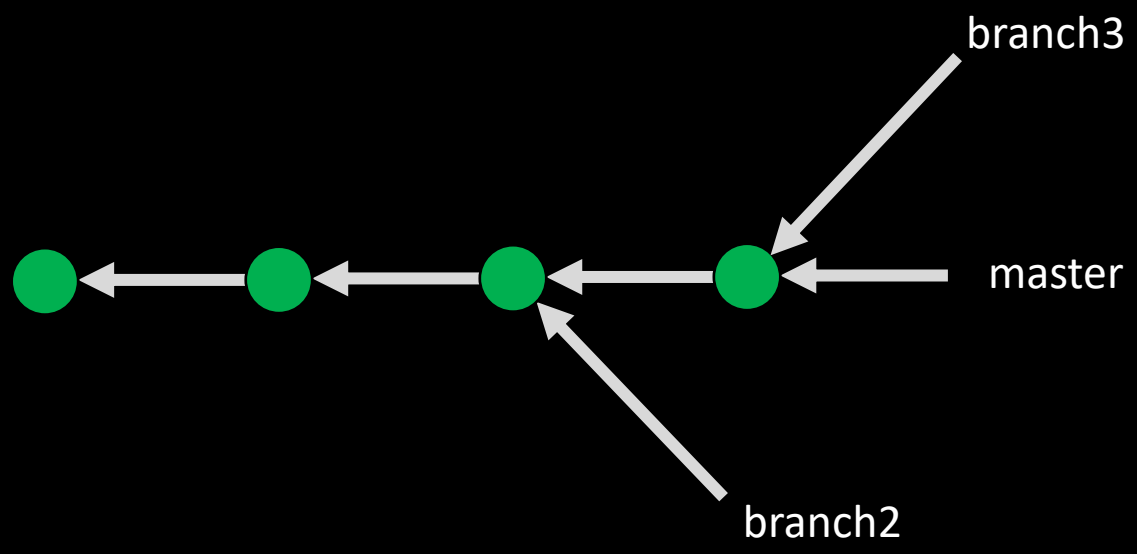
What Does a Git Repo Really Look Like?



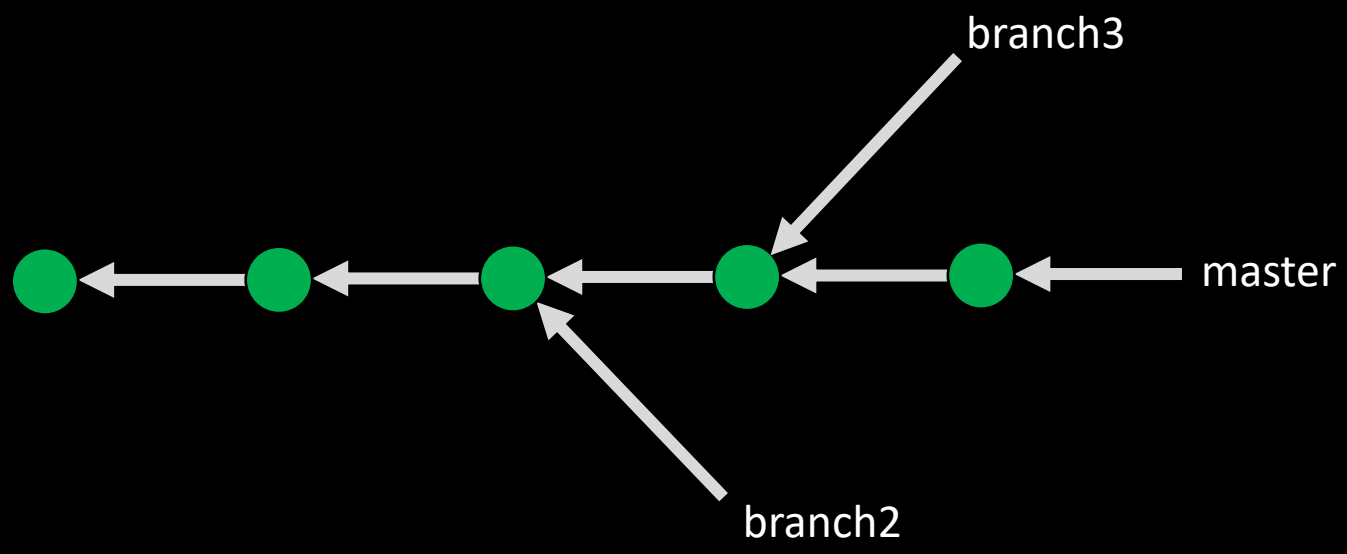
What Does a Git Repo Really Look Like?



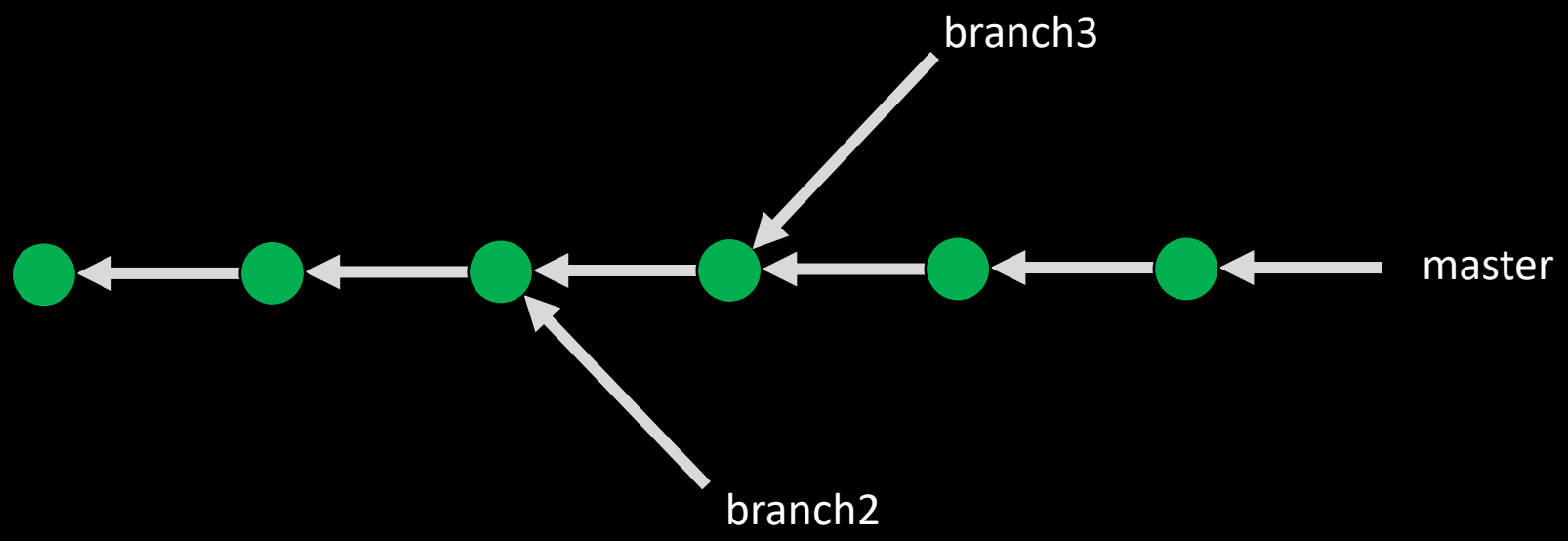
What Does a Git Repo Really Look Like?



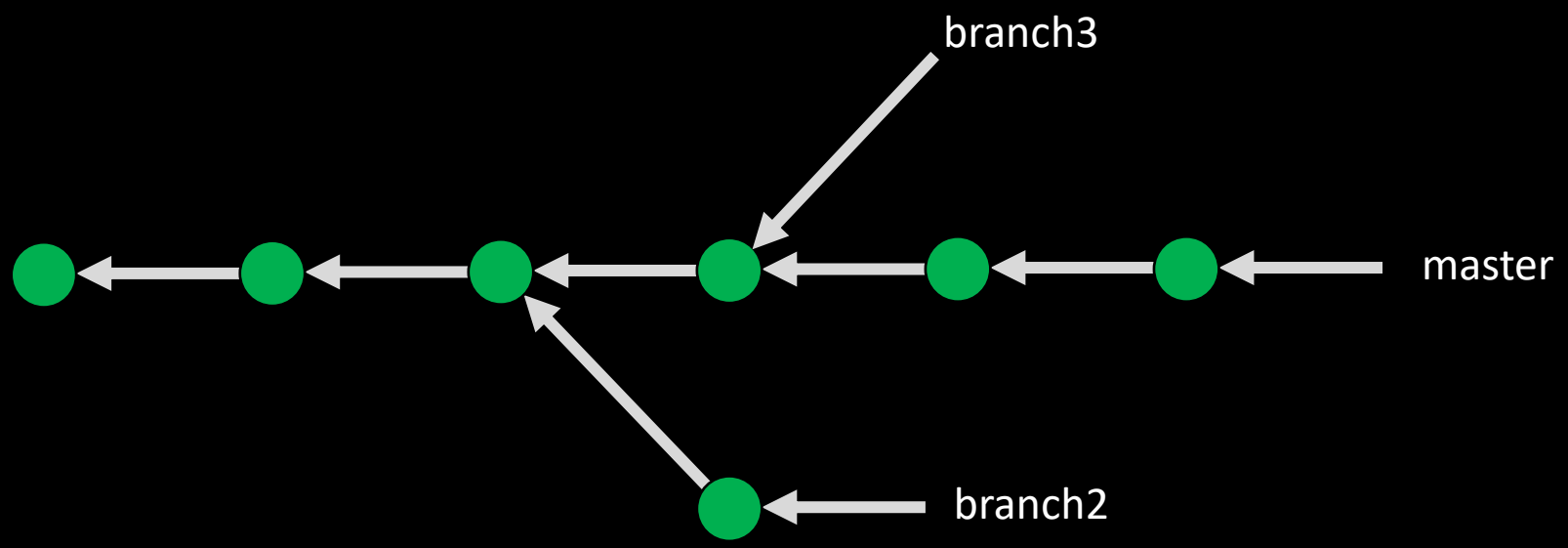
What Does a Git Repo Really Look Like?



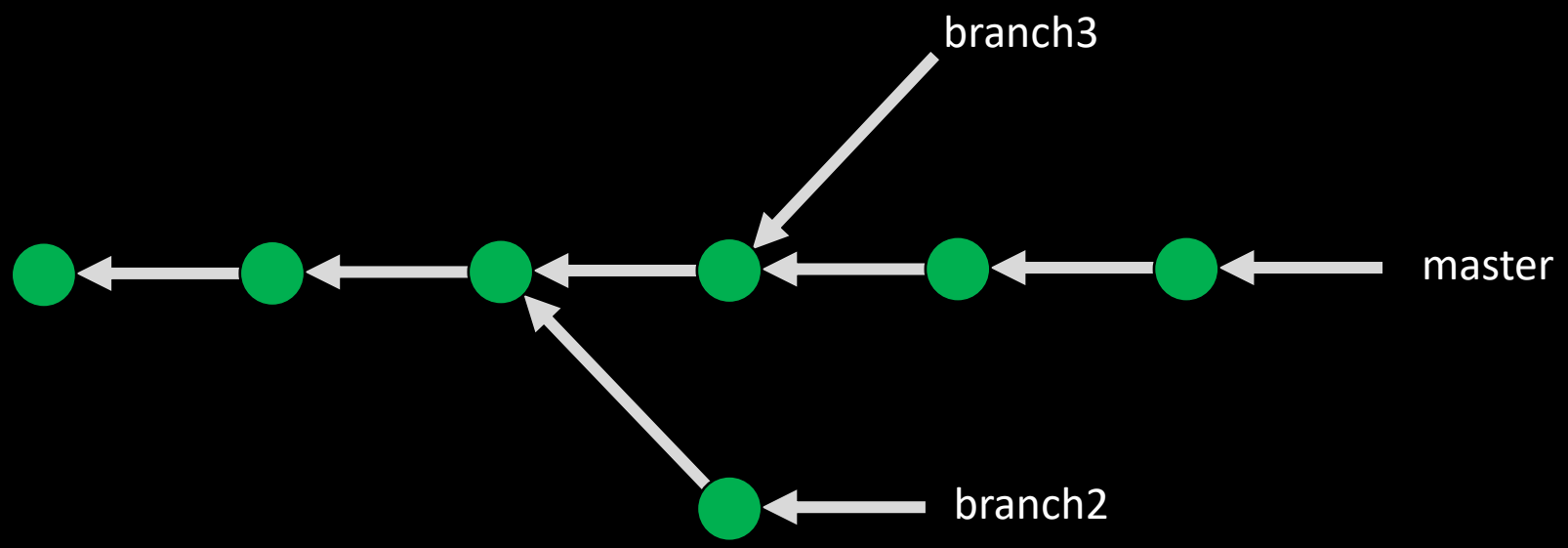
What Does a Git Repo Really Look Like?



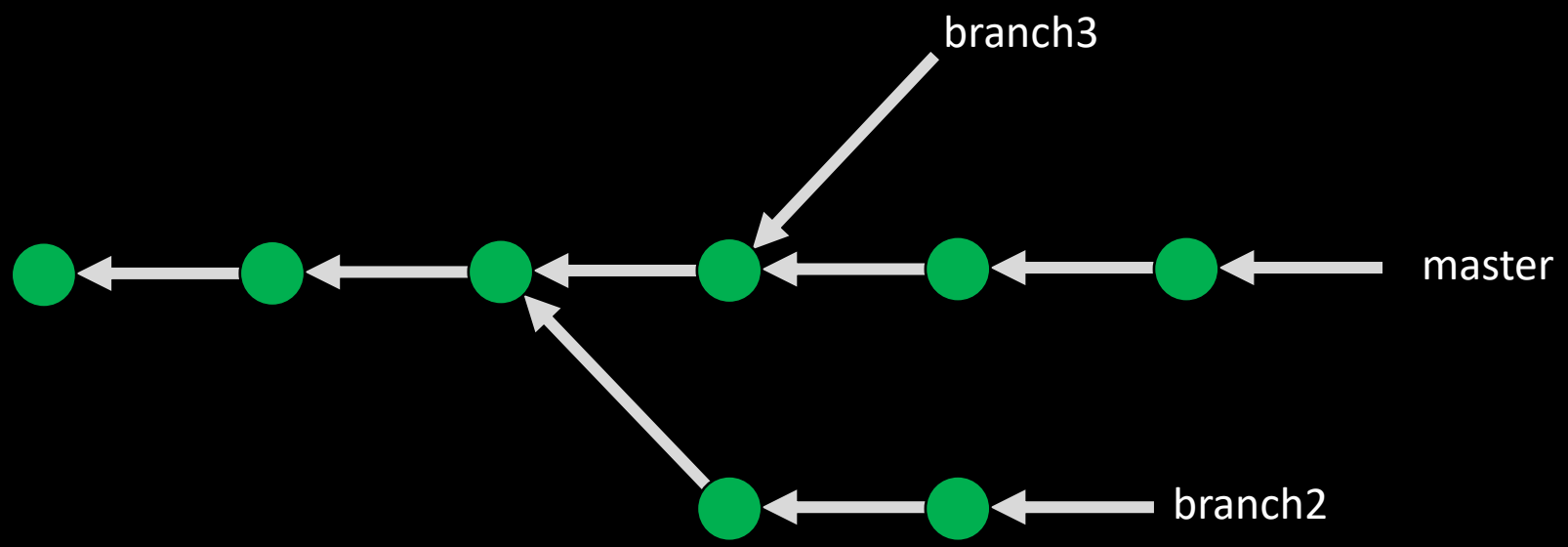
What Does a Git Repo Really Look Like?



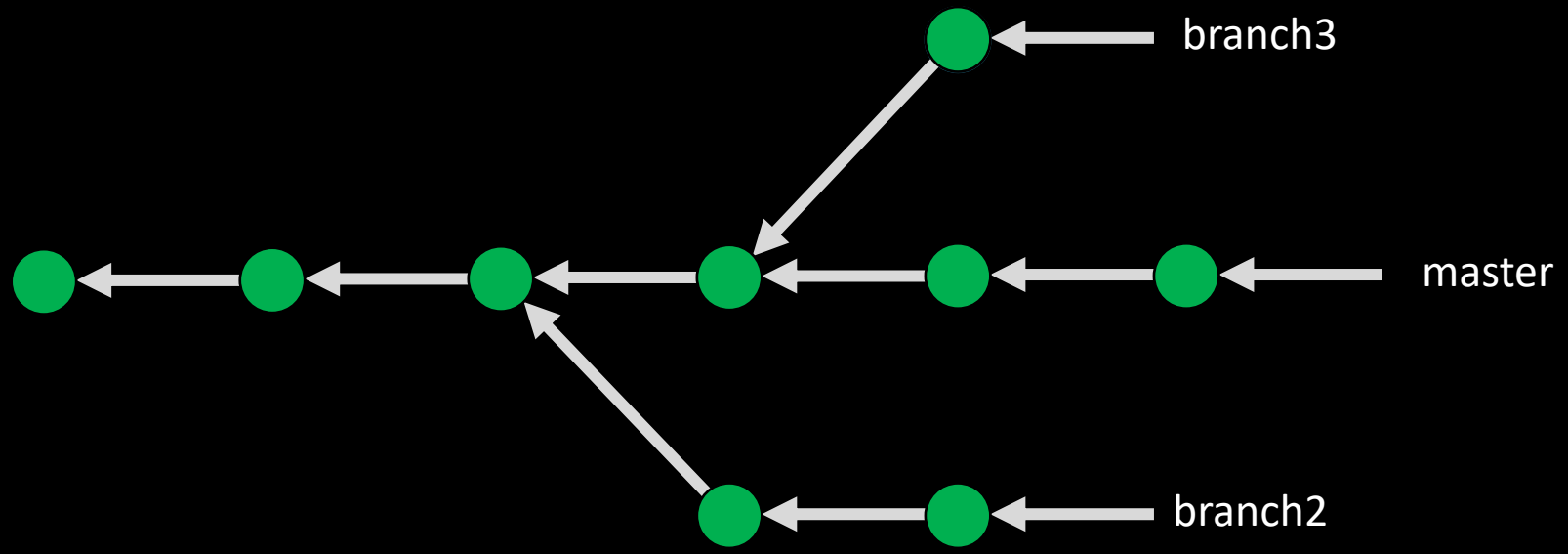
What Does a Git Repo Really Look Like?



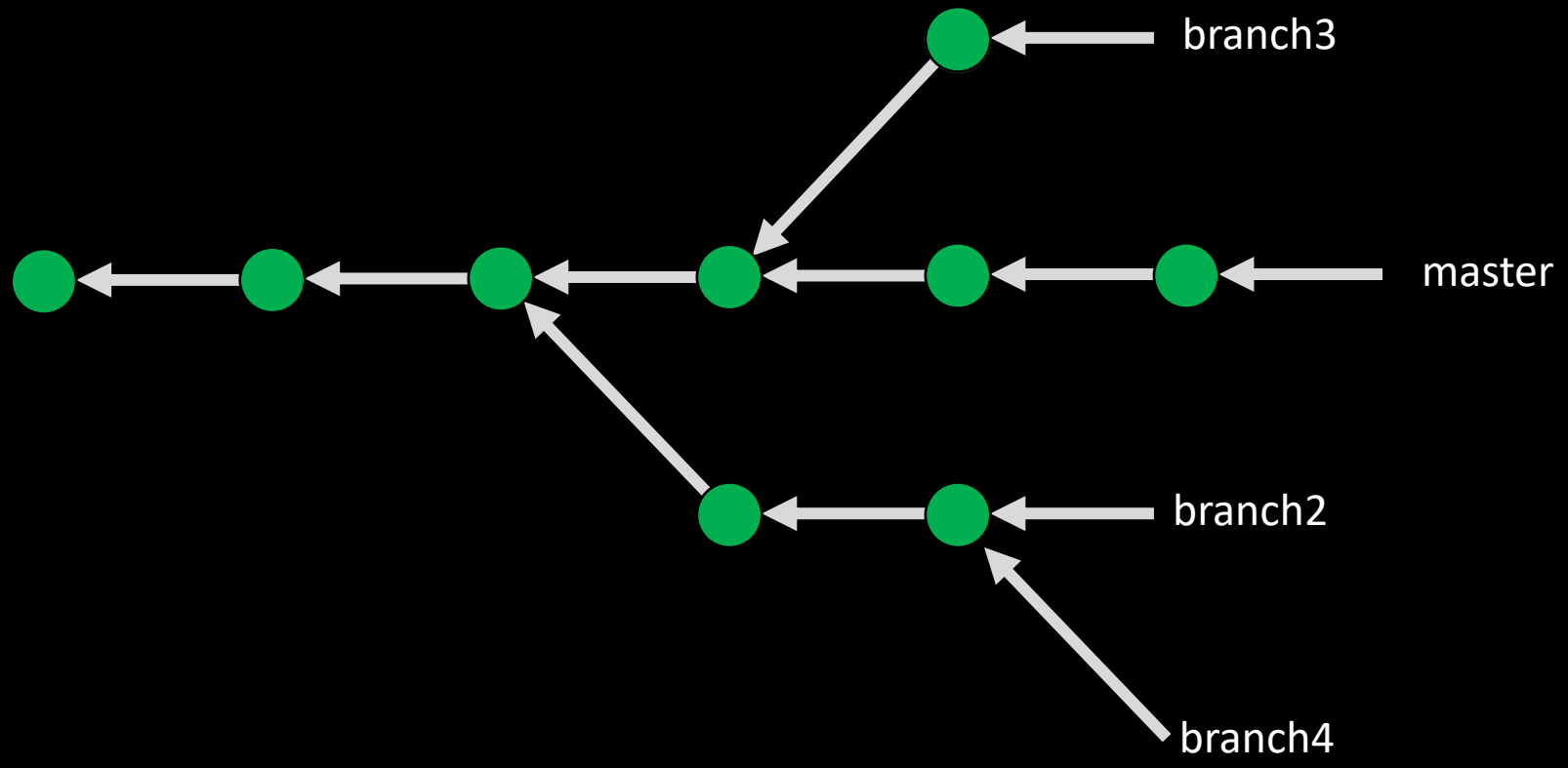
What Does a Git Repo Really Look Like?



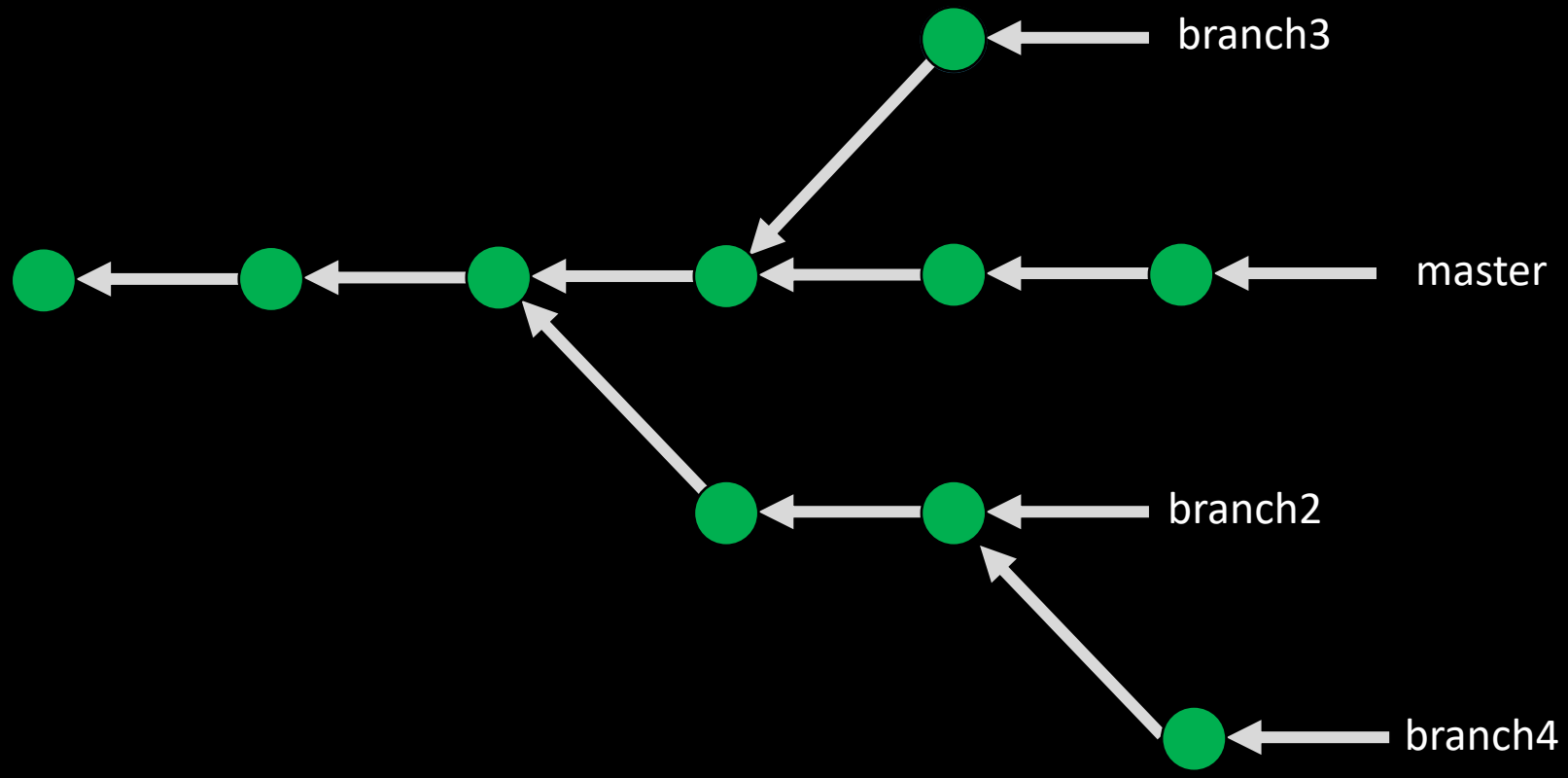
What Does a Git Repo Really Look Like?



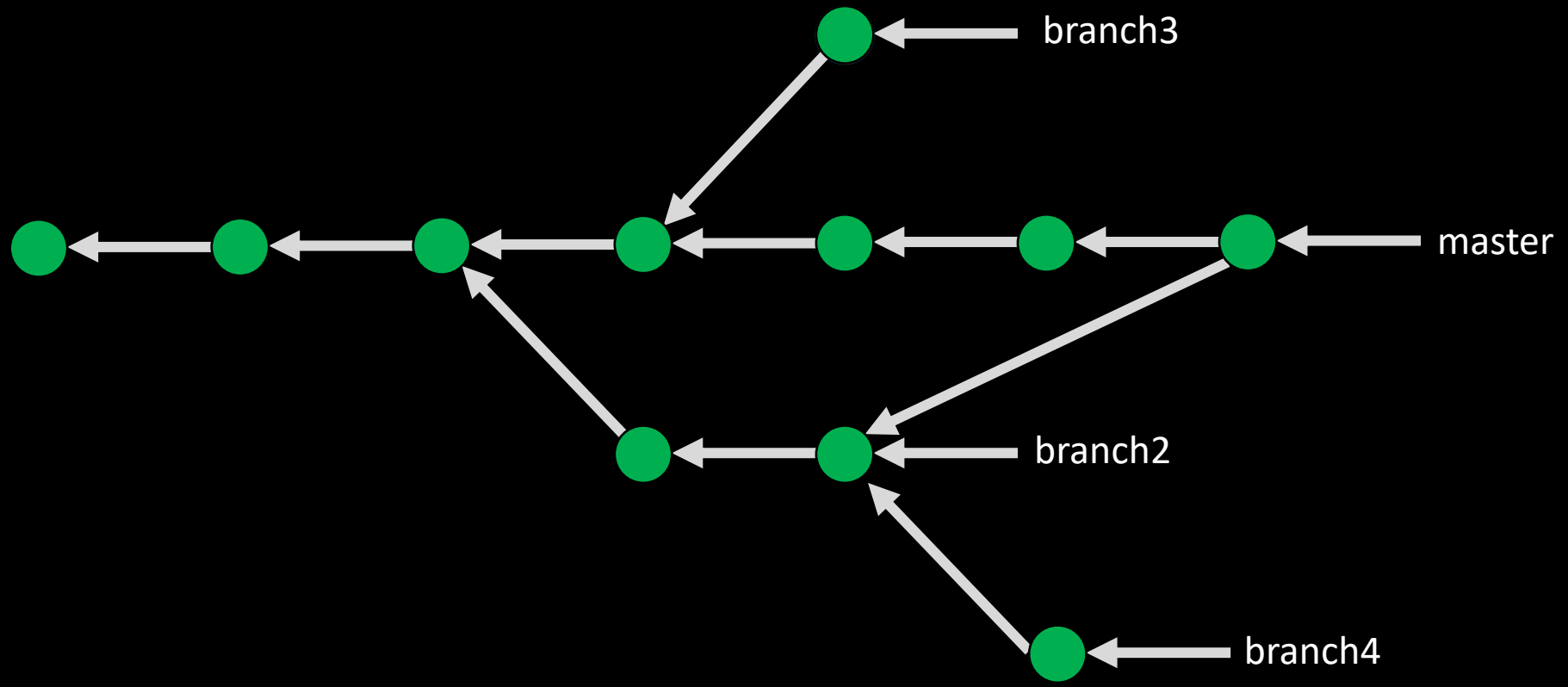
What Does a Git Repo Really Look Like?



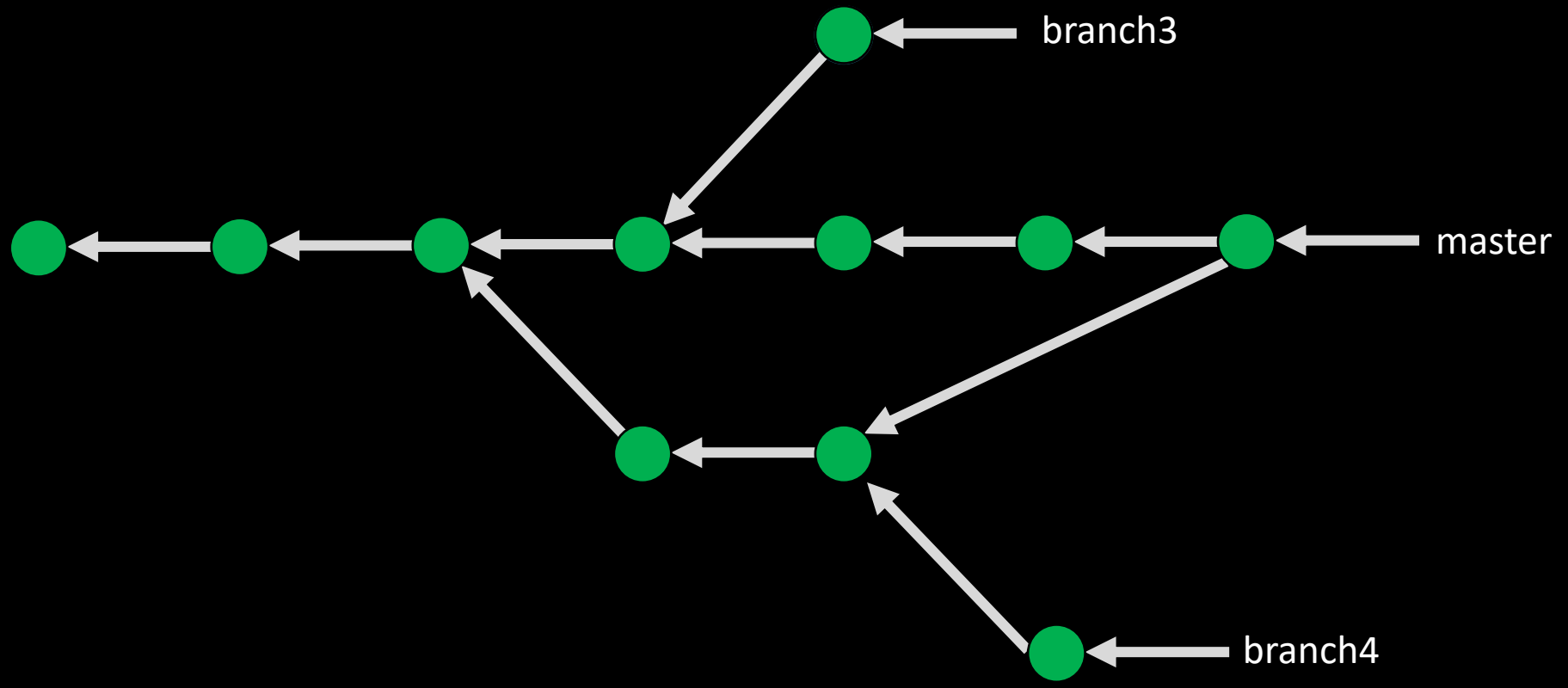
What Does a Git Repo Really Look Like?



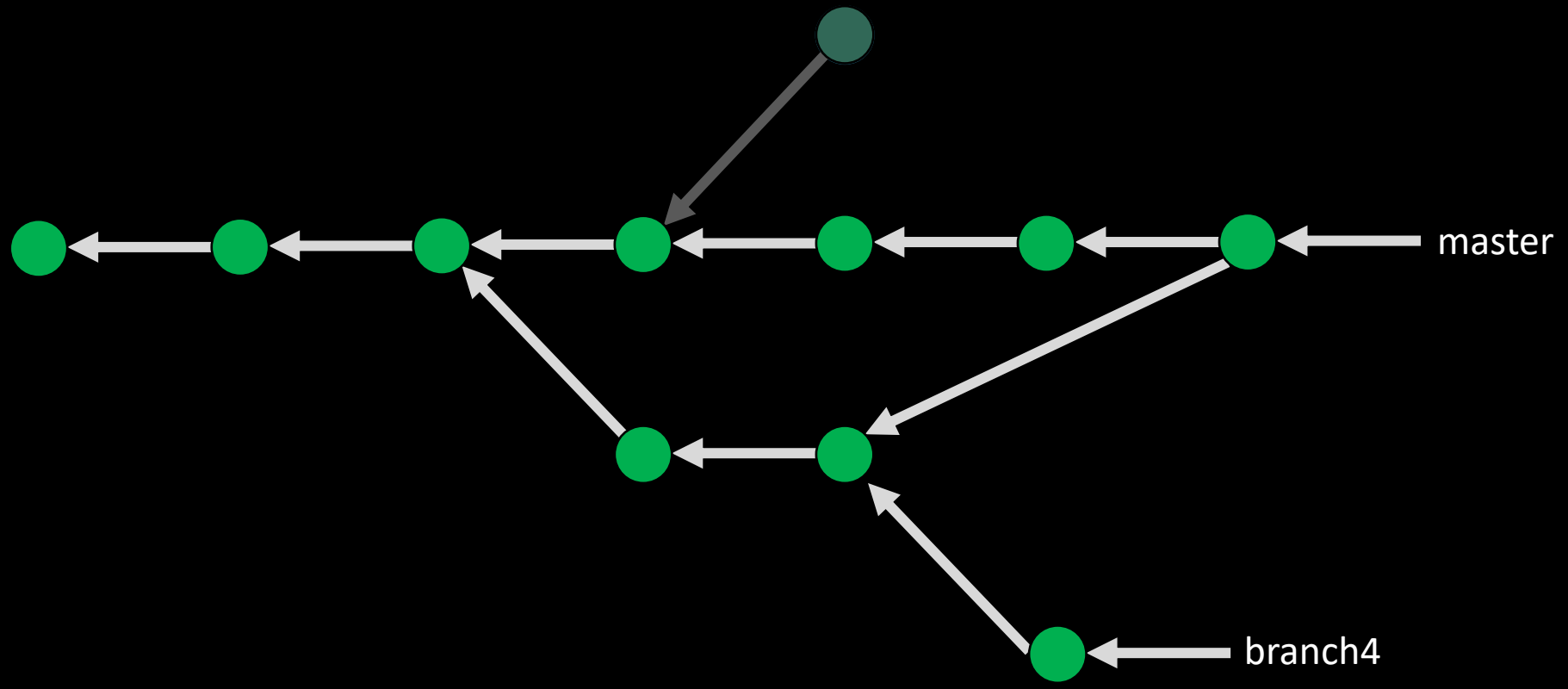
What Does a Git Repo Really Look Like?



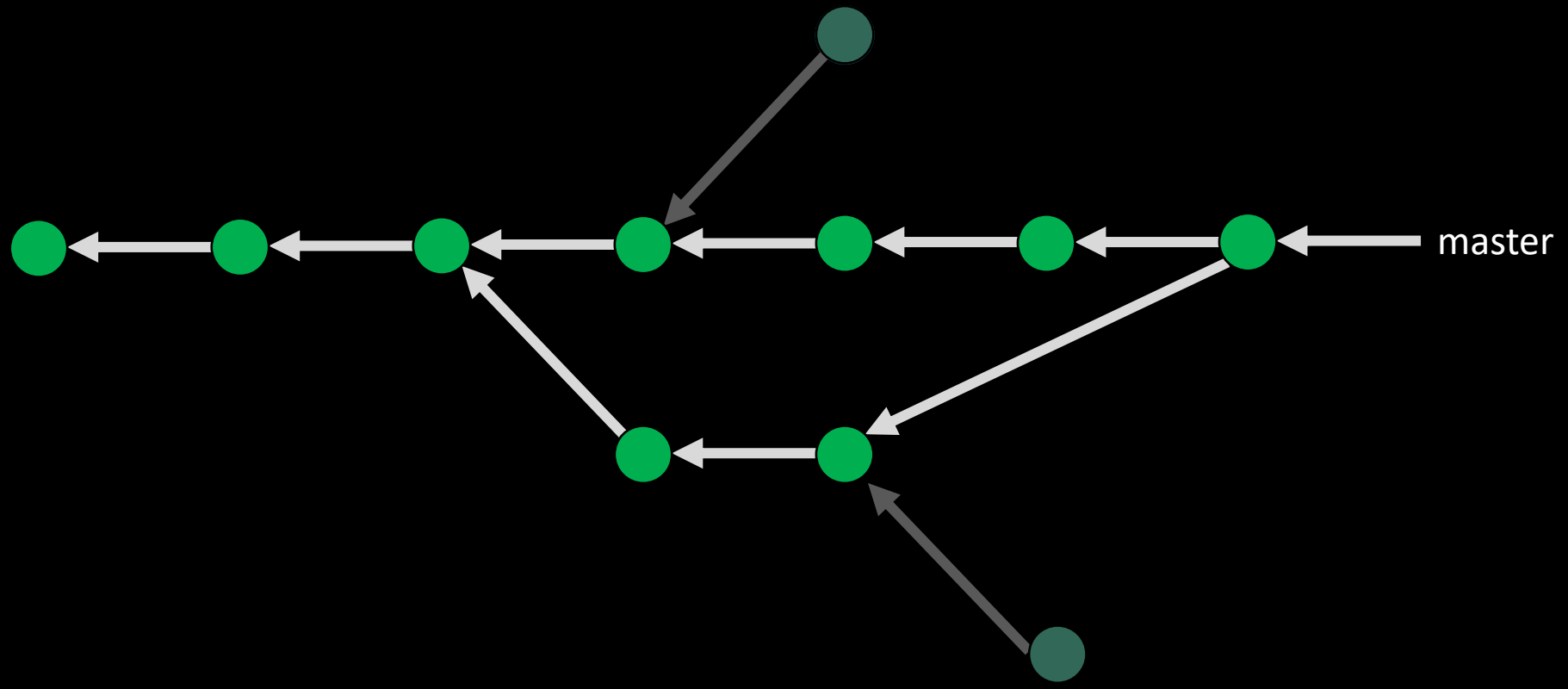
What Does a Git Repo Really Look Like?



What Does a Git Repo Really Look Like?



What Does a Git Repo Really Look Like?



Demo Time…

Thank You

@javaxnerd

leechristie.com

lee@leechristie.com