

Commit early! Commit often!

A gentle introduction to the joy of Git and GitHub

Isaiah Lankham

Legacy Health
System Office
Portland, OR

Matthew T. Slaughter

Kaiser Permanente
Center for Health Research
Portland, OR

Half-Day Class • 1:30 p.m. • 02NOV2023

Who "gits" it?

Raise Your Hand if...

- ✓ you've had issues working with code sent to you as an email attachment?
- ✓ you've had issues determining which version of a program is the "most recent"?
- ✓ you've suspected the word "final" in a filename is a lie?

But what is "git"?



standalone software
created by Linus Torvalds
in 2005 for distributed
version control



web-based software
development platform
founded in 2008, using
Git version control

Common Workflows for GitHub

Theme. GitHub lowers the friction of collaboration (even when working alone)

Workflow 1. Commits in "Main Branch"

Workflow 2. Commits in "Feature Branch" + Pull Request

Workflow 3. Fork + Commits in "Feature Branch" + Pull Request

Workflow 4. Fork + Clone + Commits in "Feature Branch" + Push + Pull Request

Common Workflows for GitHub

Theme. GitHub lowers the friction of collaboration (even when working alone)

Workflow 1. Commits in "Main Branch"



Advantages

- Uses the fewest Git concepts
- Immutable commit history provides a detailed project audit trail

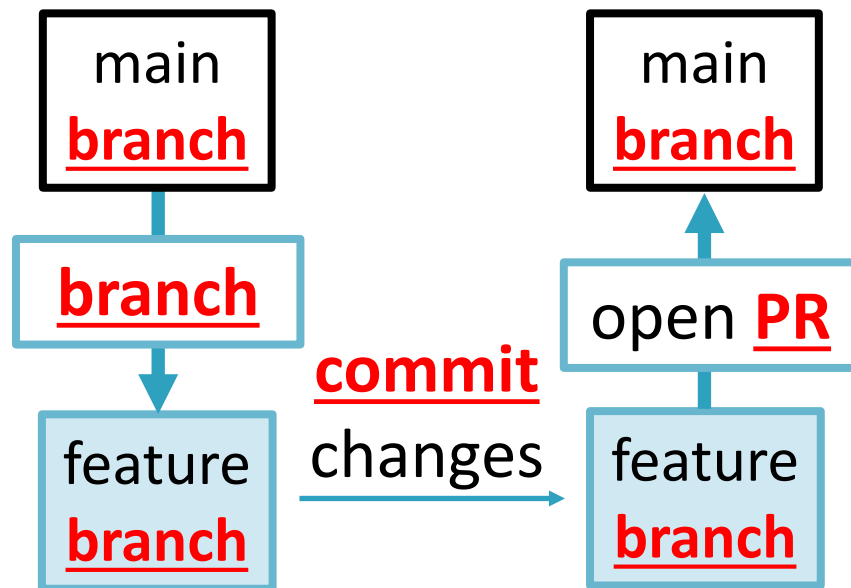
Disadvantages

- Discourages "exploratory coding"
- Difficult to simultaneously collaborate on the same file
- Everyone has the same permissions to change the main branch

Common Workflows for GitHub

Theme. GitHub lowers the friction of collaboration (even when working alone)

Workflow 2. Commits in "Feature Branch" + **Pull Request** (aka PR)



Advantages

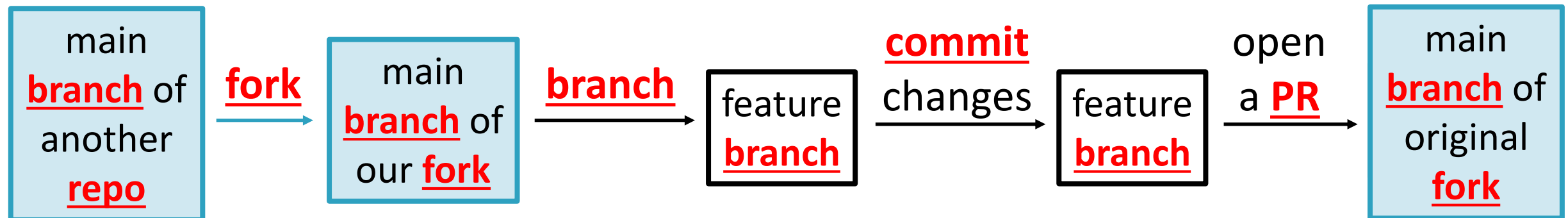
- Immutable commit history
- Enables "exploratory coding"
- Enables simultaneous collaboration
- Enables differentiated permissions (controlling changes to main branch)

Main Disadvantage: Added complexity

Common Workflows for GitHub

Theme. GitHub lowers the friction of collaboration (even when working alone)

Workflow 3. Fork + Commits in "Feature Branch" + Pull Request (aka PR)

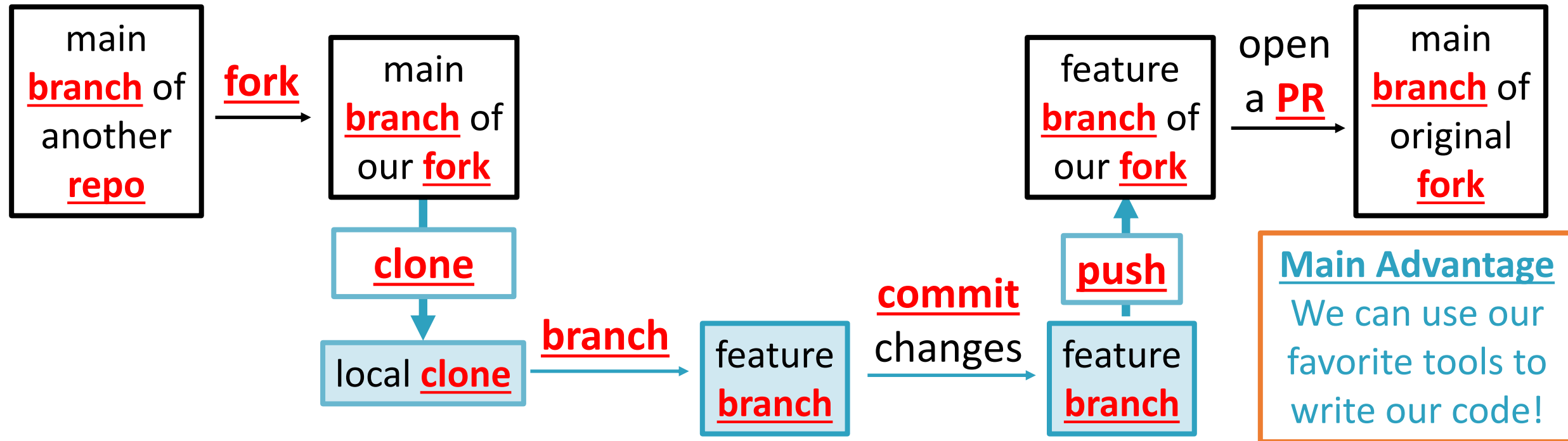


Main Advantage: This enables us to "suggest" unsolicited changes to someone's project (e.g., open-source projects on GitHub).

Common Workflows for GitHub

Theme. GitHub lowers the friction of collaboration (even when working alone)

Workflow 4. Fork + **Clone** + Commits in "Feature Branch" + **Push** + Pull Request



Common Workflows for GitHub

Theme. GitHub lowers the friction of collaboration (even when working alone)

Workflow 1. Commits in "Main Branch"

Workflow 2. Commits in "Feature Branch" + Pull Request

Workflow 3. Fork + Commits in "Feature Branch" + Pull Request

Workflow 4. Fork + Clone + Commits in "Feature Branch" + Push + Pull Request




Today I Learned...



Source: <https://xkcd.com/1597>

In case of fire



-  1. **git commit**
-  2. **git push**
-  3. **leave building**

Source: <https://github.com/hendrixroa/in-case-of-fire>

Call to Action!

- **Takeaway.** Git and GitHub are amazing tools for collaboration, even when collaborating with just your (past or future) self!
- **Just Do It.** Start putting code for personal projects on GitHub, even if it's not polished!
- **Embrace the Open Source Ethos.** Go fork a GitHub repo, make a small change (e.g., fix a typo), and open a PR today!
- **Keep up the Good Work.** All of the materials for this class are on GitHub.

