website: vanderbi.lt/github

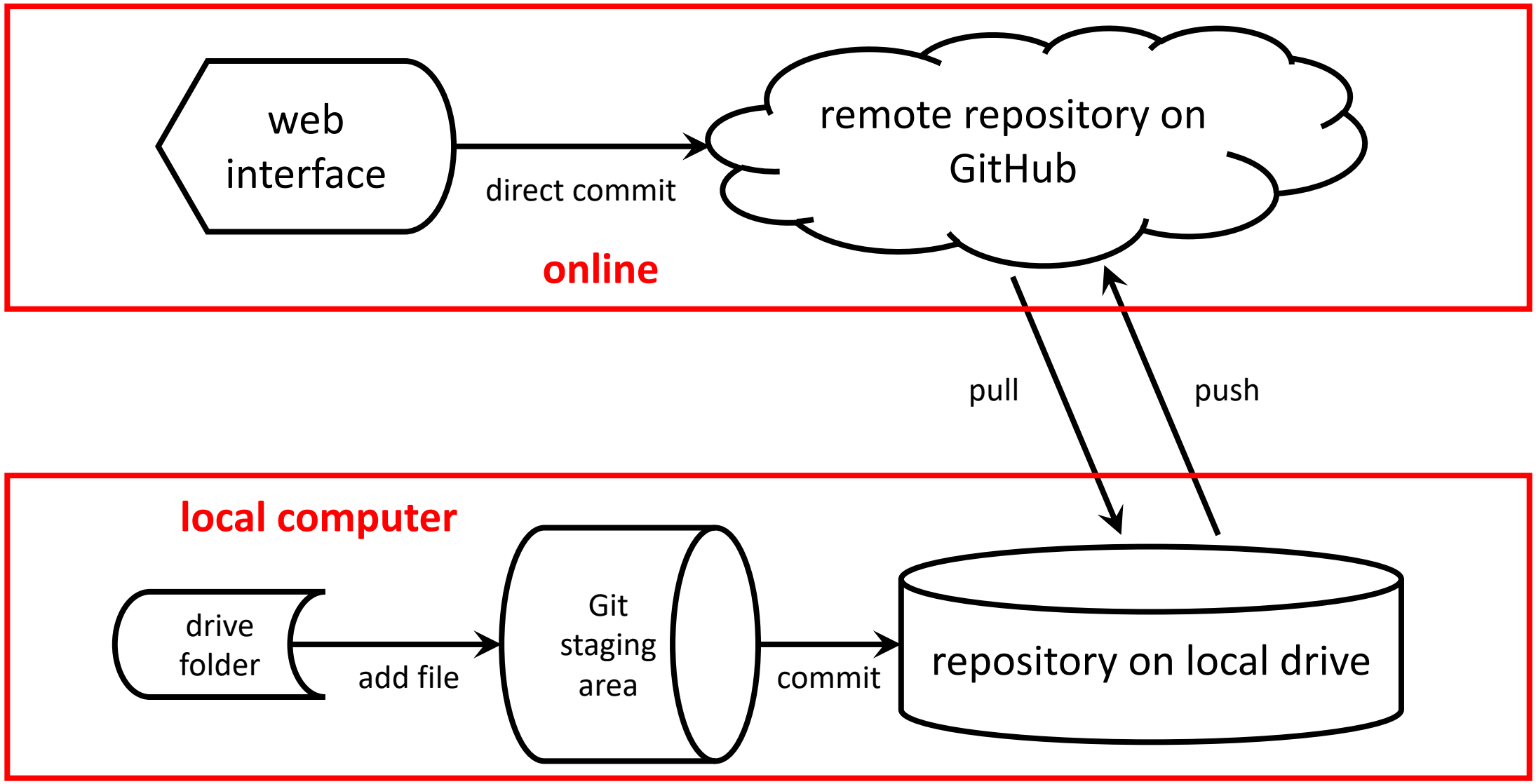
Git and GitHub terms and diagrams

Presenter: Steve Baskauf
steve.baskauf@vanderbilt.edu

DISC DIGITAL SCHOLARSHIP AND COMMUNICATIONS

Jean & Alexander Heard LIBRARIES

The World of Git and GitHub



time

1. To be or not to be
2. What was the question?

shakespere.txt

did NOT use most recent version

should have used most recent version

1. To be or not to be,
2. What was the question?
3. Life = 42

shakespere1.txt

1. To be or not to be,
2. That is the question!

shakespere-a.txt

?

1. To be or not to be,
2. That is the question!
3. The end.

shakespere-b.txt

Typical (bad) version control

time

1. To be or not to be
2. that is the question

shakespere-2019-01-25.txt

1. To be or not to be,
2. That is the question!

shakespere-2019-02-08.txt

1. To be or not to be
2. That is the question!
3. Life = 42

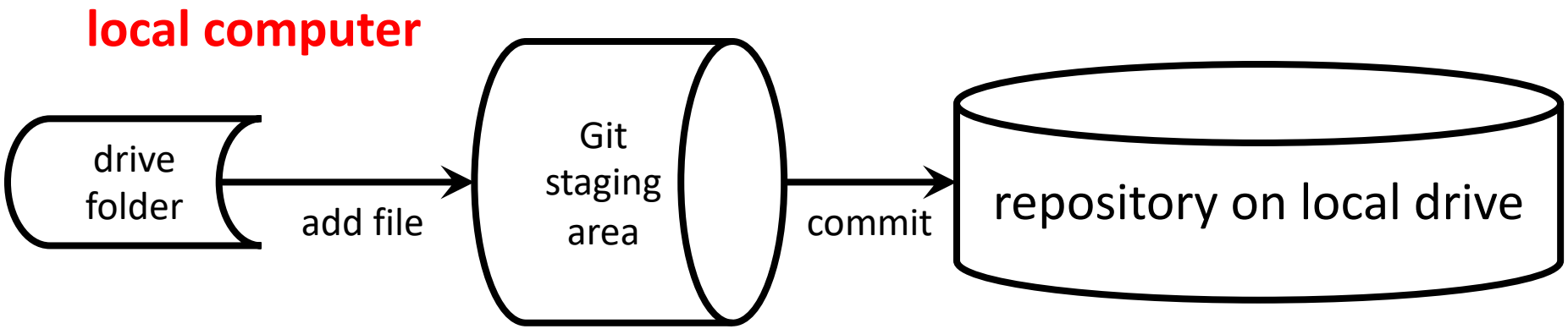
shakespere-2019-02-12.txt

1. To be or not to be
2. That is the question!
3. Life = 42
4. The end.

shakespere-2019-03-02.txt

Simple version control system

Local version control for disaster prevention

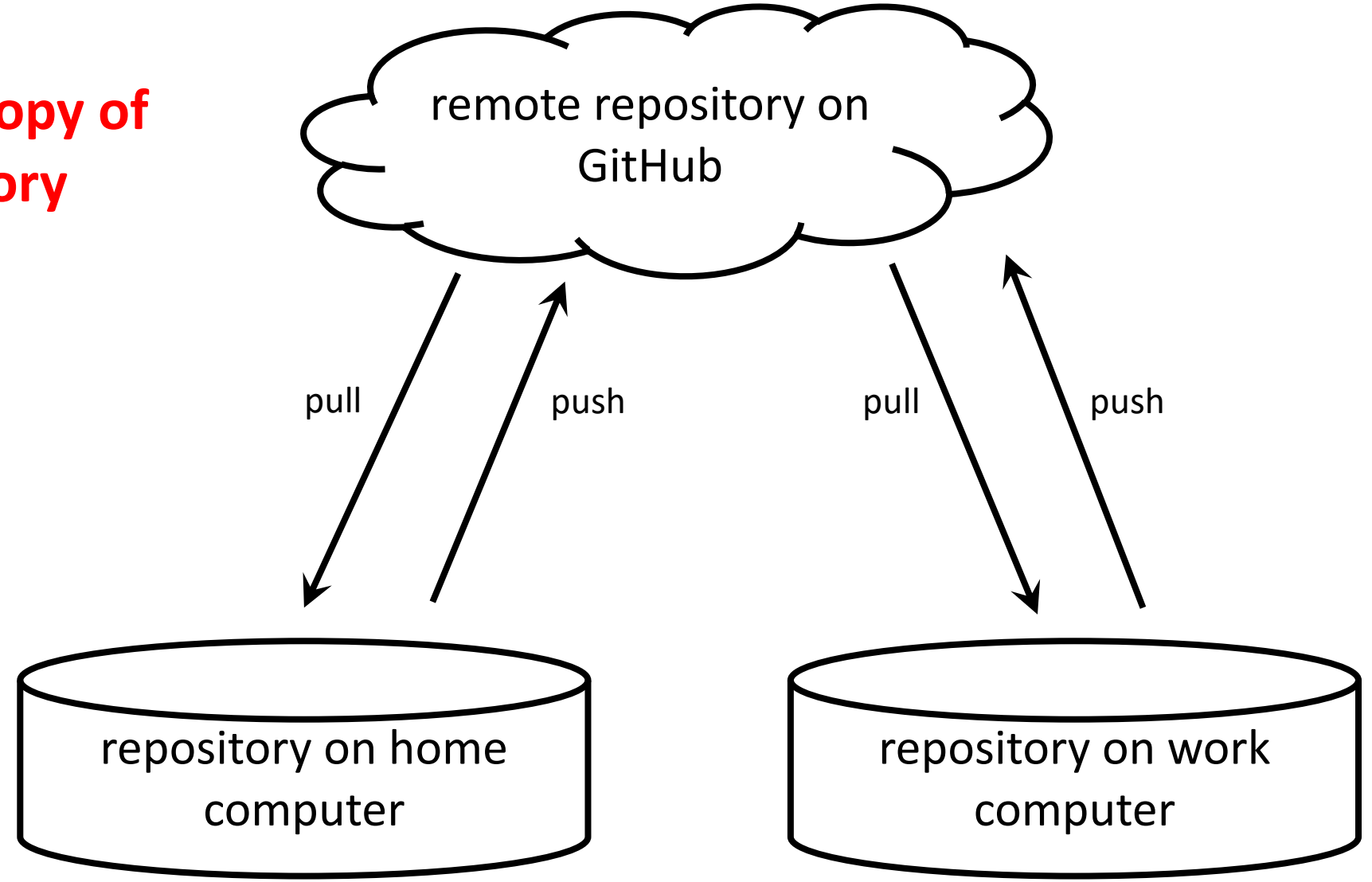


Variation in use of GitHub

- Depends on number of collaborators in the group
- Depends on who has write access to the repository
- Depends on social conventions established by the group

Disaster prevention via synching with GitHub (1 user)

online copy of repository



clones of GitHub repository

time

1. To be or not to be
2. that is the question

shakespere-2019-01-25.txt

used most recent version

did NOT use most recent version

should have used most recent version

1. To be or not to be,
2. catnip the question

shakespere-2019-02-08.txt

1. To be or not to be
2. What was the question?

shakespere-2019-02-12.txt

?

?

1. To be or not to be,
2. ?????

shakespere-2019-03-02.txt

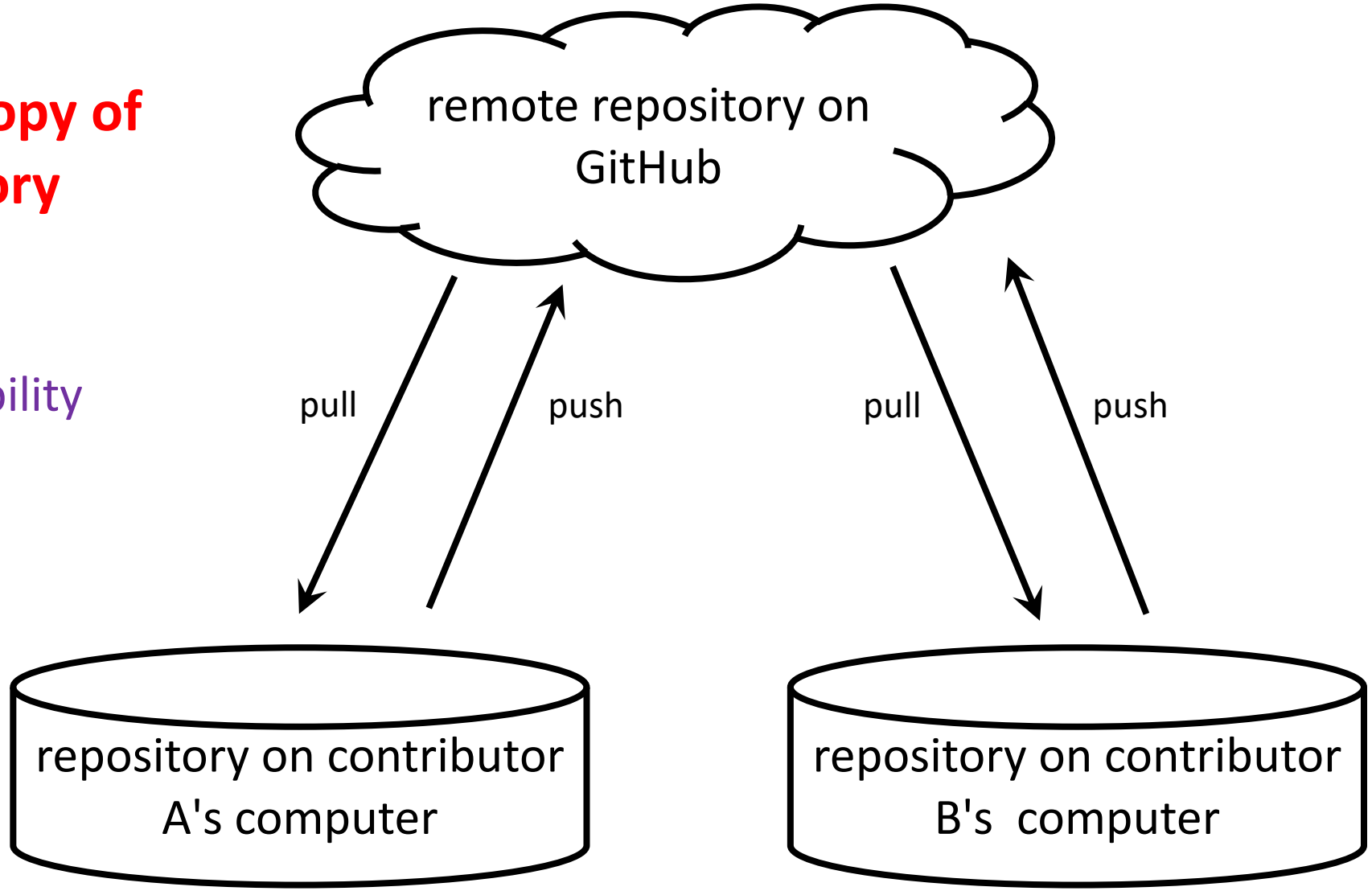
How version conflicts arise

Simple collaboration (>1 user)

online copy of repository

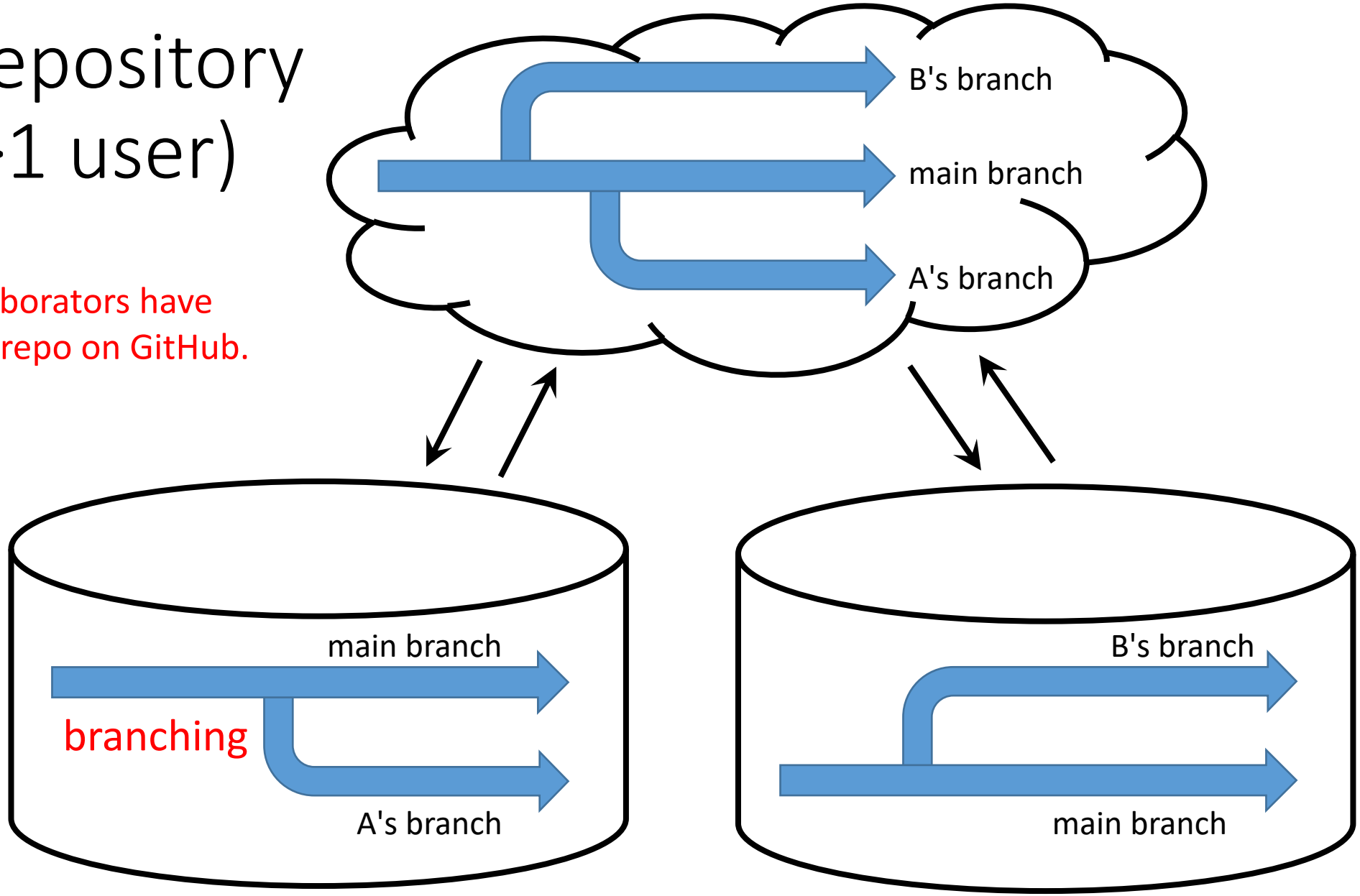
Warning! High probability of version conflicts!

clones of GitHub repository



Shared repository model (>1 user)

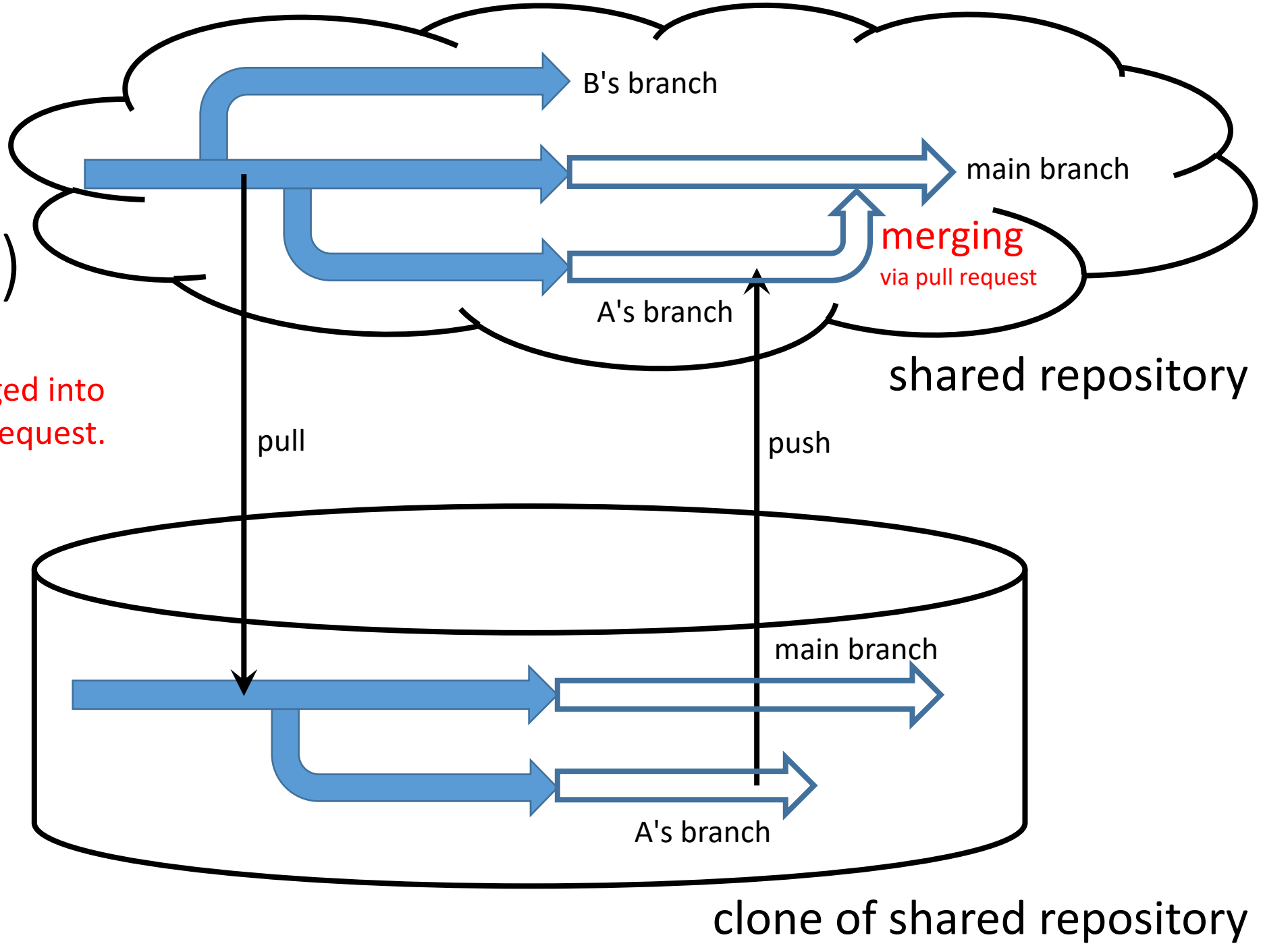
Used when all collaborators have write access to the repo on GitHub.



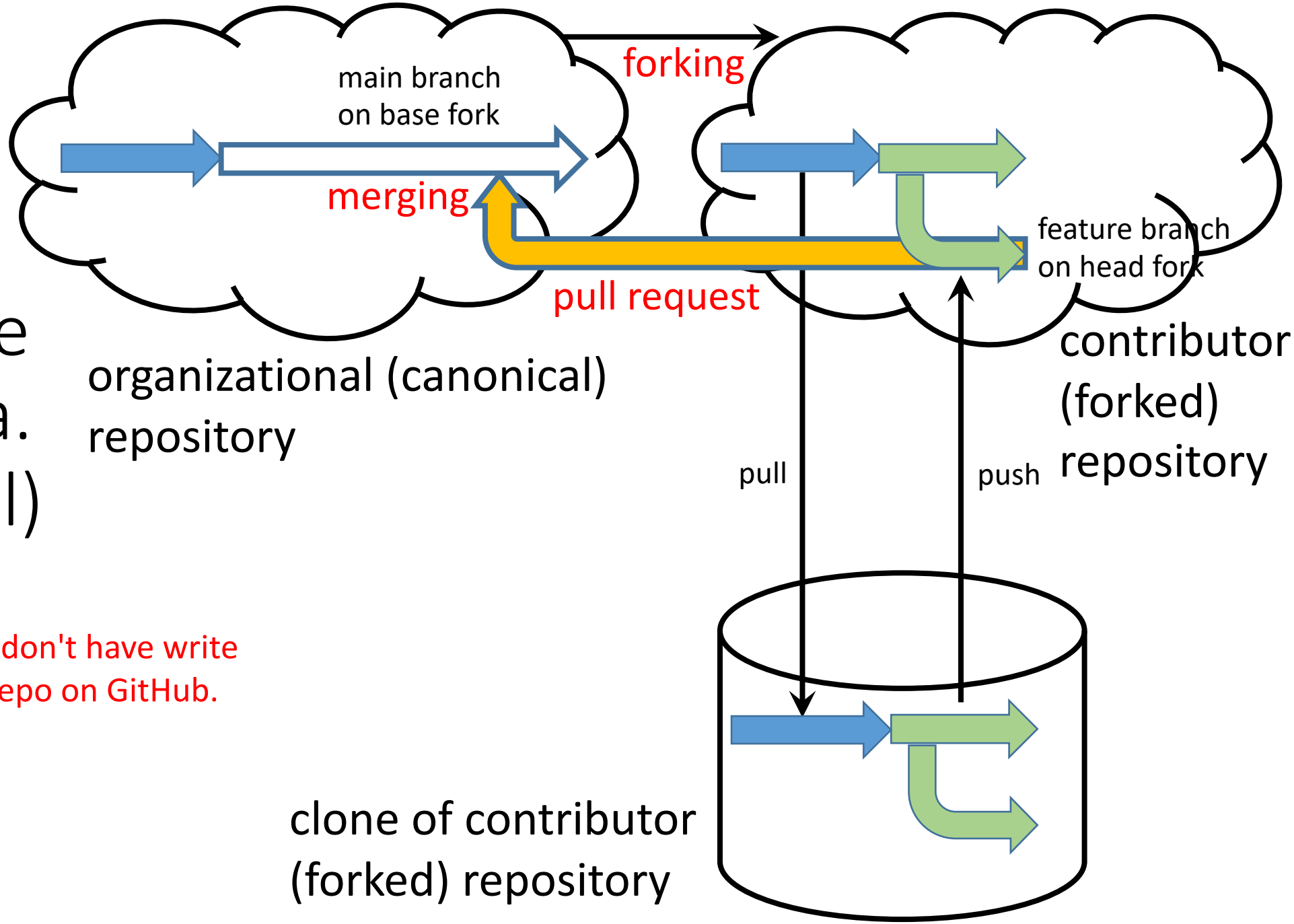
Version conflicts avoided when working on separate branches

GitHub flow (social conventions)

Working branch is merged into main branch via a pull request.

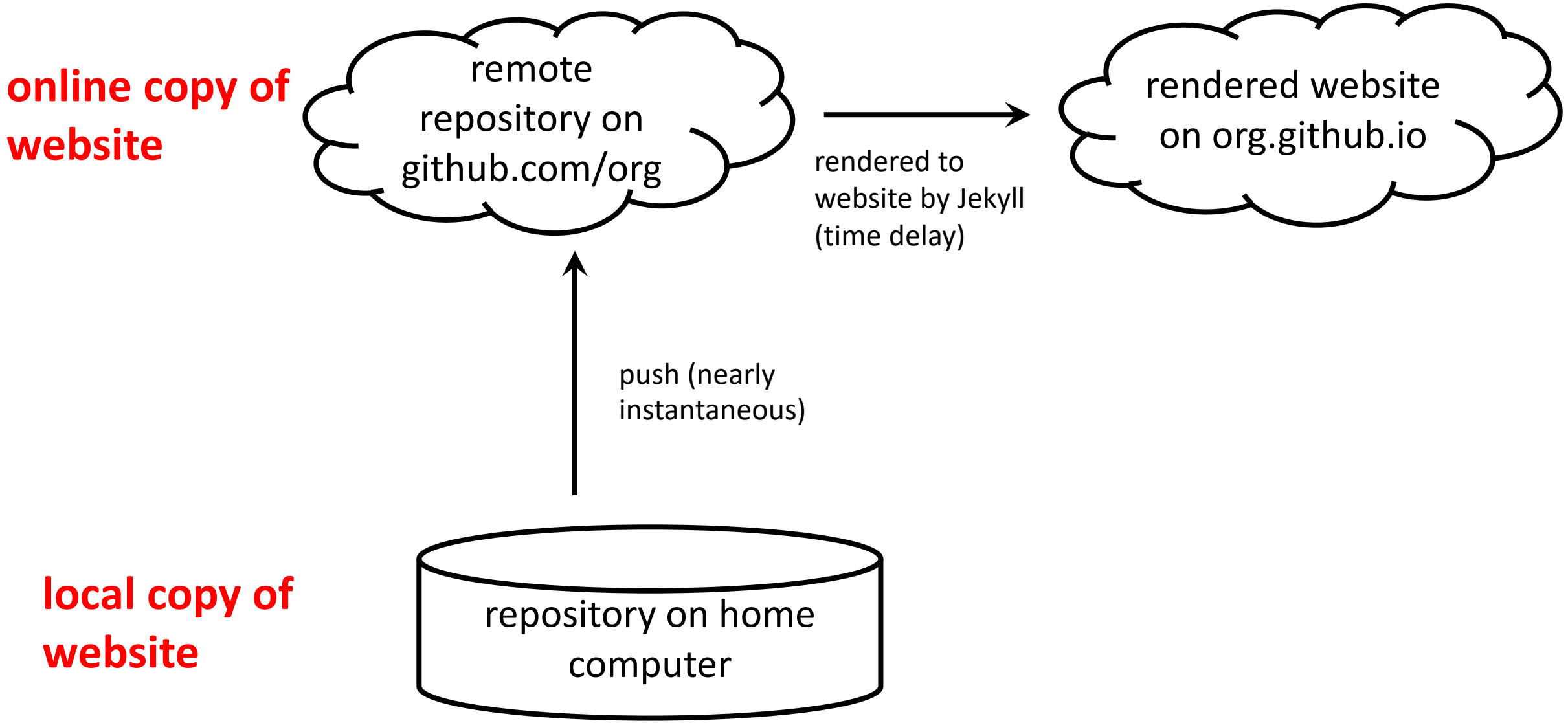
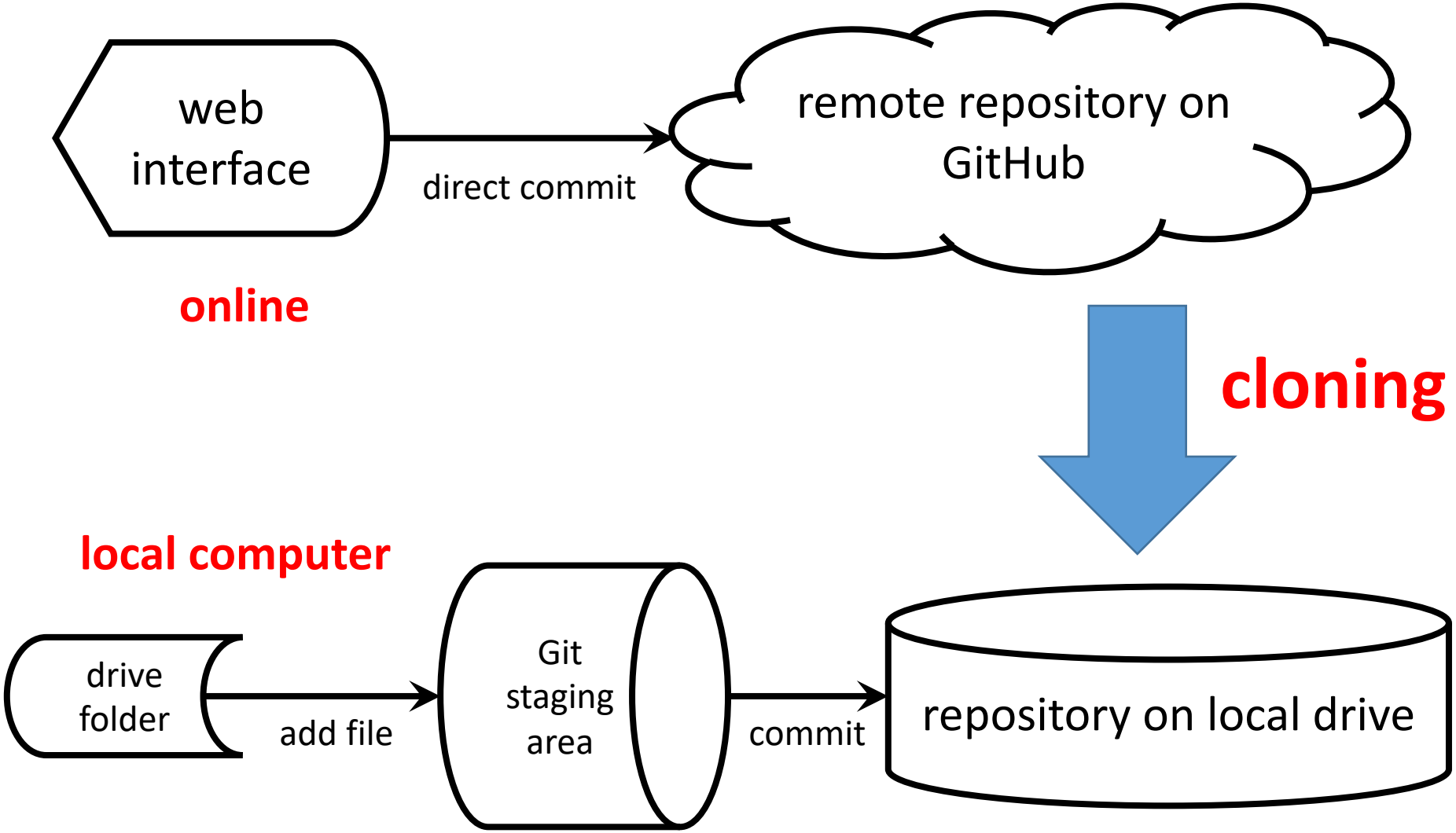


Open Source model (a.k.a. fork and pull)



Used when contributors don't have write access to the canonical repo on GitHub.

Managing a website using GitHub Pages (single user)



Cloning a repo to the desktop