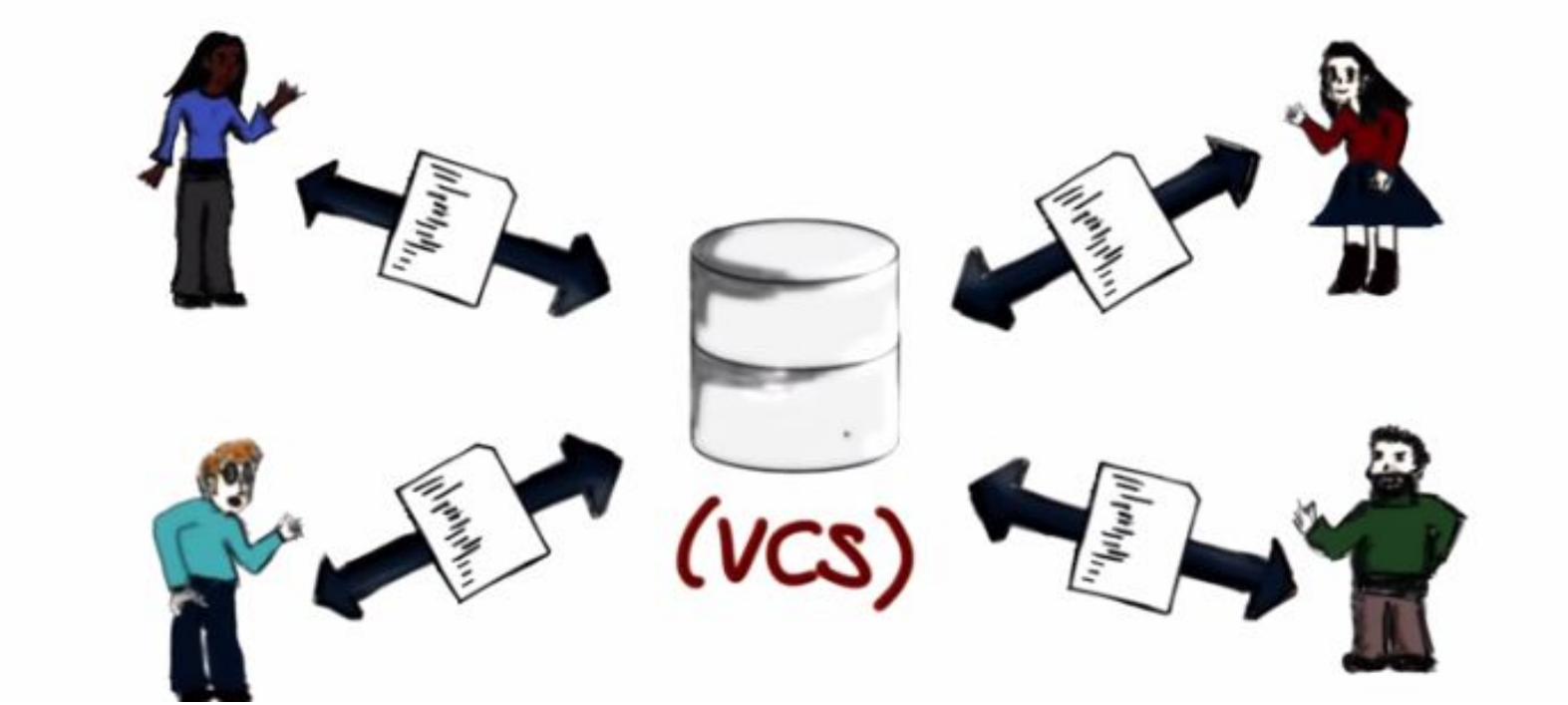
Version Control

Lab

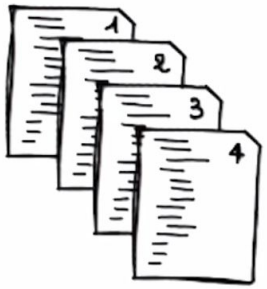
Version Control

- Udacity about GitHub



Why Version Control Systems?

Enforce discipline



Archive versions

Maintain historical information

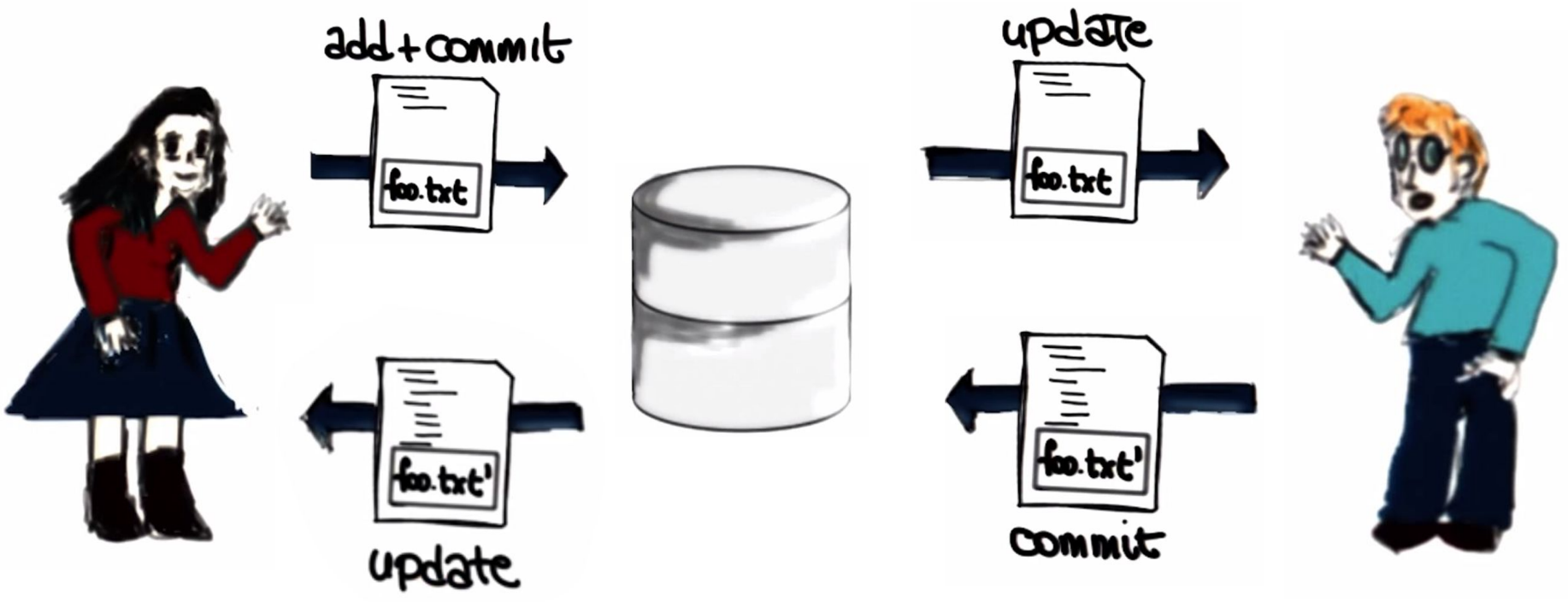
Enable collaboration

Recover from accidental deletions or edits

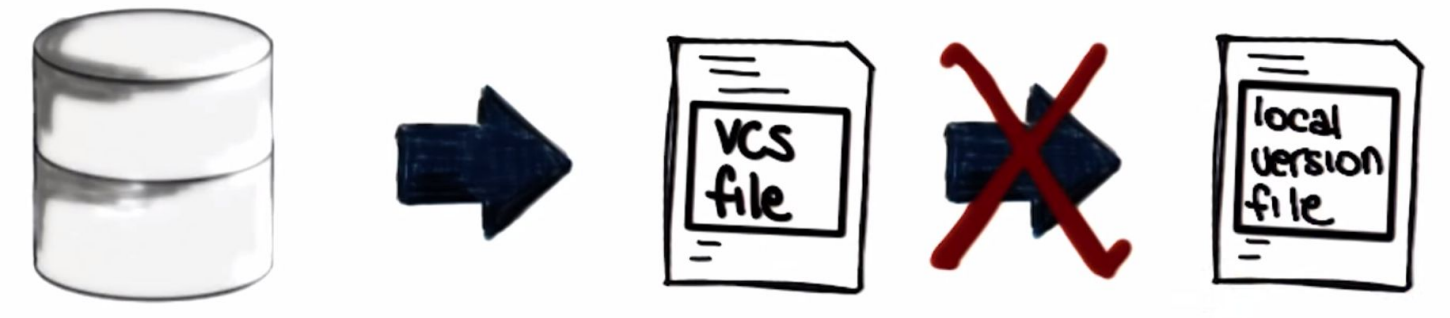
Conserve disk space

Essential Actions

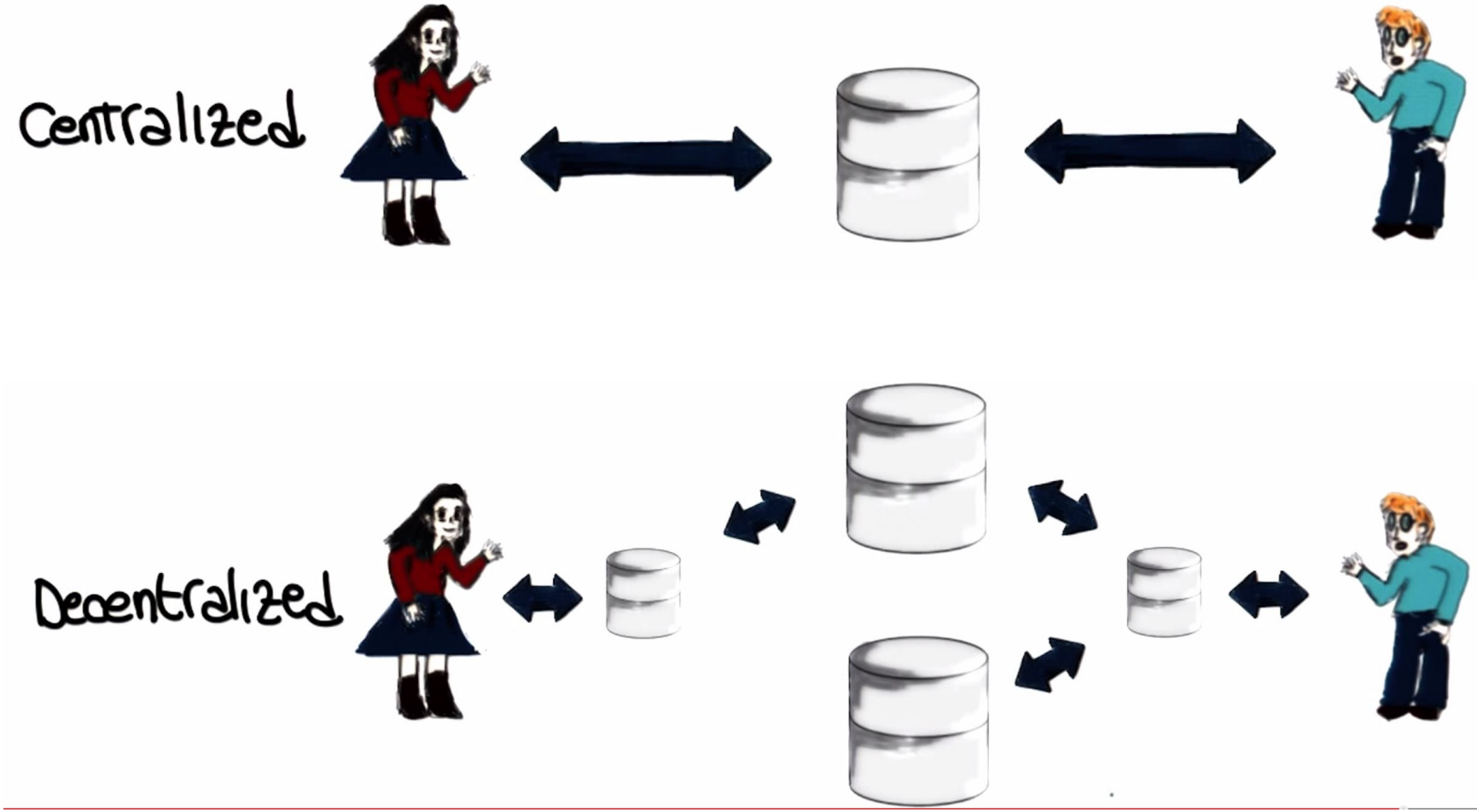
VCS Workflow



Don't do it



Types of VCS



GIT

Lab

About GIT

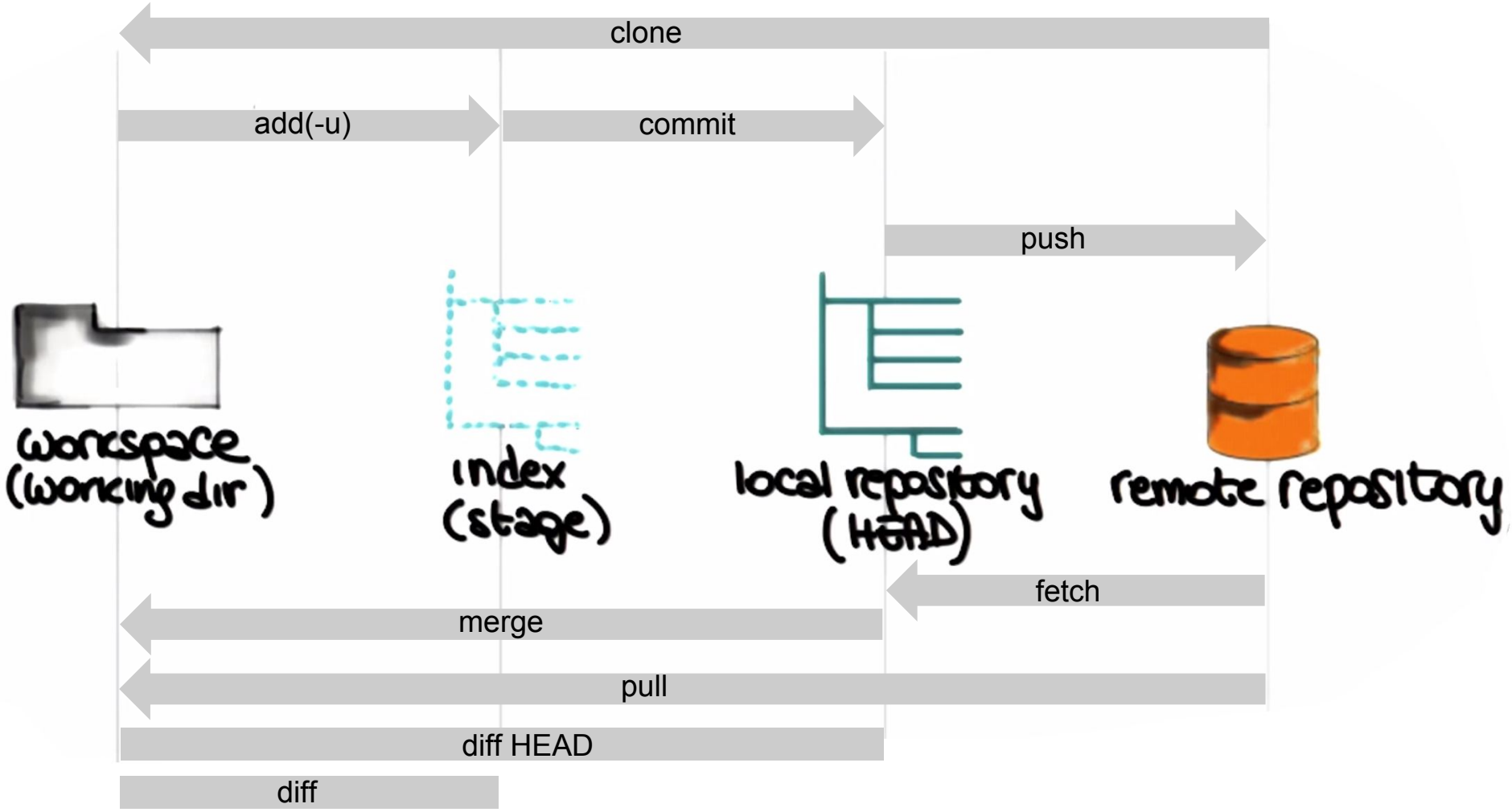
- Decentralized VCS
- Linus Thorvalds
- Large Projects in an efficient way
- For example Linux

Install GIT

- Linux: Package
- Mac OS: XCode, Package
- Windows: Package with Installer

- see http://git-scm.com/downloads

Git Workflow



GIT DEMO 1

Basics

Local Repositories

Create

A possible workflow
(This is just an example!)

- some editing
- git status
  to see what files you changed
- git diff [files]
  to see the changes
- git commit -a [-m <message>]

git status
git diff
git show (last commit)

GIT DEMO 2

Advanced

Remote Repositories

Copy repository

- git clone <repository>
  - repository = URL (file, http, ssh, ...)
  - creates a complete local copy of the repository
  - links it to the remote repository (origin)
    (you can also link to <repository> later

Receive changes

- git pull

Send changes

- git push

GIT HUB

LAB

Projects

JHotDraw

LAB TODOs

- Clone remote repository from GitHub
- Get familiar with Git commands
  - see this slide for an overview
- Commit your initial project code using the Git commands
  add, commit, pull, push...
- Input change request as an Issue at GitHub