Work efficiently, stay organized, collaborate, and achieve your dreams with

GitHub and version control

Snow Naing
PUBS 2018

Adapted from Alison Maxwell, Bootcamp 2018

🥳 Version control 🎉

- Experiment without interfering with working code

- Find when a bug arose in your code

- Maintain multiple versions of script

- Revert back to a working version of code after ruining it

GitHub: What and why

- Git: remote Version Control System (VCS)

- The Hub: a place to store your work, network, and collaborate

- Repositories ("repos") for organization

- Command-line and GUI interfaces

Remote repository

fetch   push

Local repository

commit

pull   **Cache**

add

Working directory
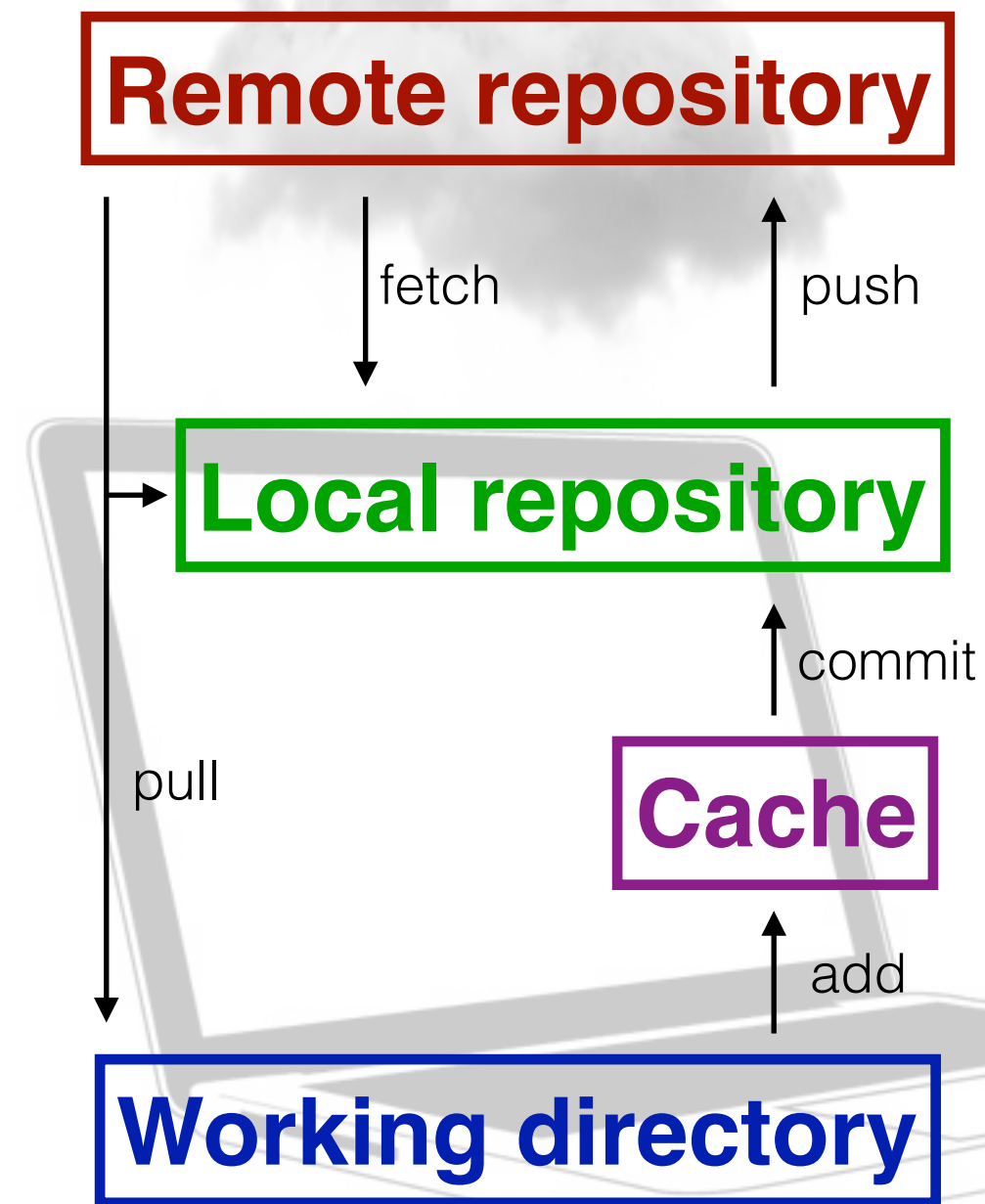
Workflow

1. Create and share a repository

2. Create a branch—your personal version of the master project/branch

3. Make and commit changes to your branch

4. Submit and review a pull request

5. Merge the pull request with the master

Remote repository

fetch   push

Local repository

commit

pull

Cache

add

Working directory
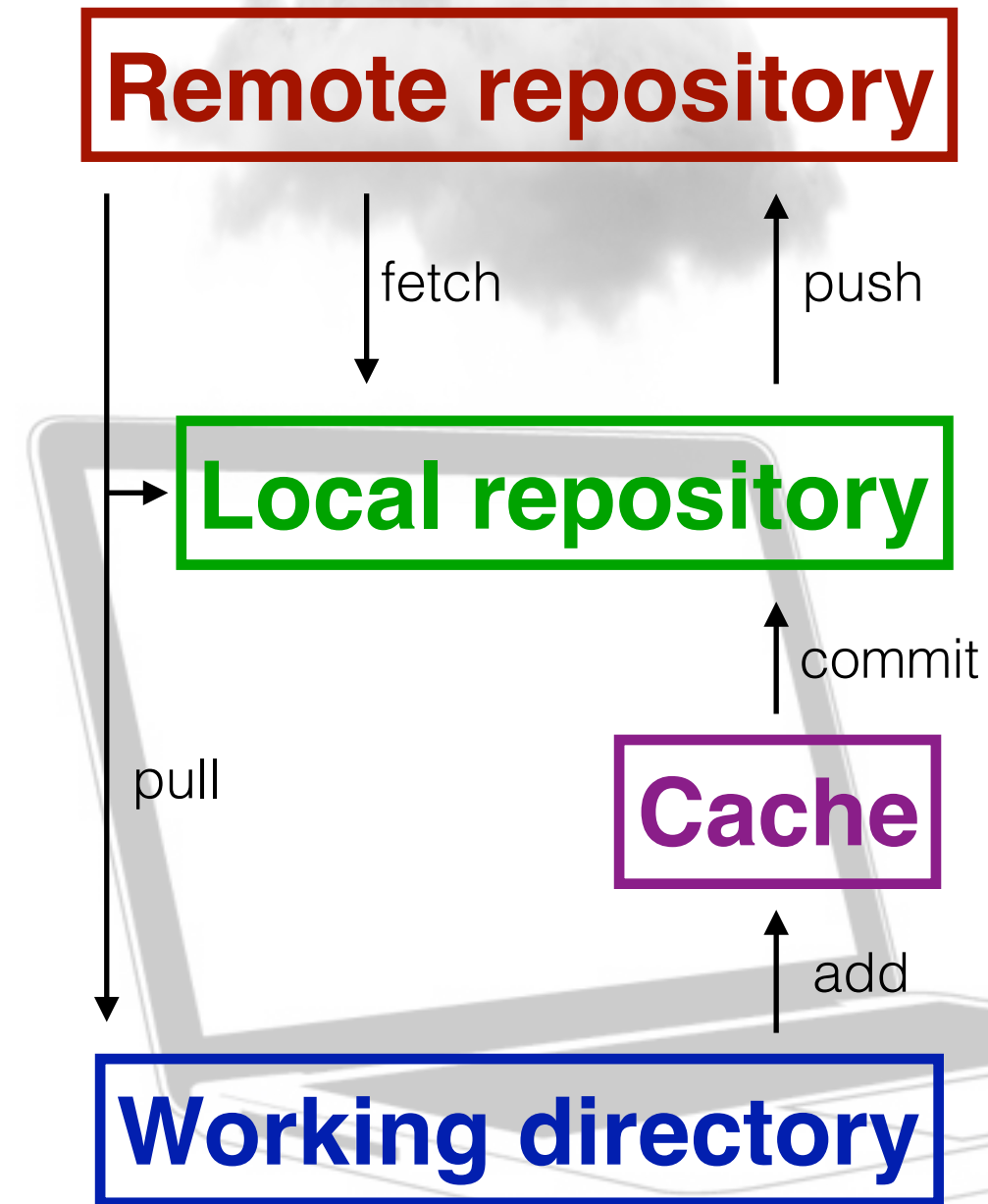
Creating a repository from Terminal

$ cd git_repo_directory #go to that directory from which you want to create the repo

$ touch README.md #make a file; you can't make a repo out of an empty folder

$ git init #initiate git

$ git add README.md #add the files to git

$ git commit -m "write a message about your file - what changes you have made"

$ curl -u 'github_username' https://api.github.com/user/repos -d '{"name":"REPONAME"}'

#This will then ask for your GitHub Password

$ git remote add origin git@github.com:github_username/REPONAME.git

$ git push -u origin master

Add Collaborators