Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Applied Data Science

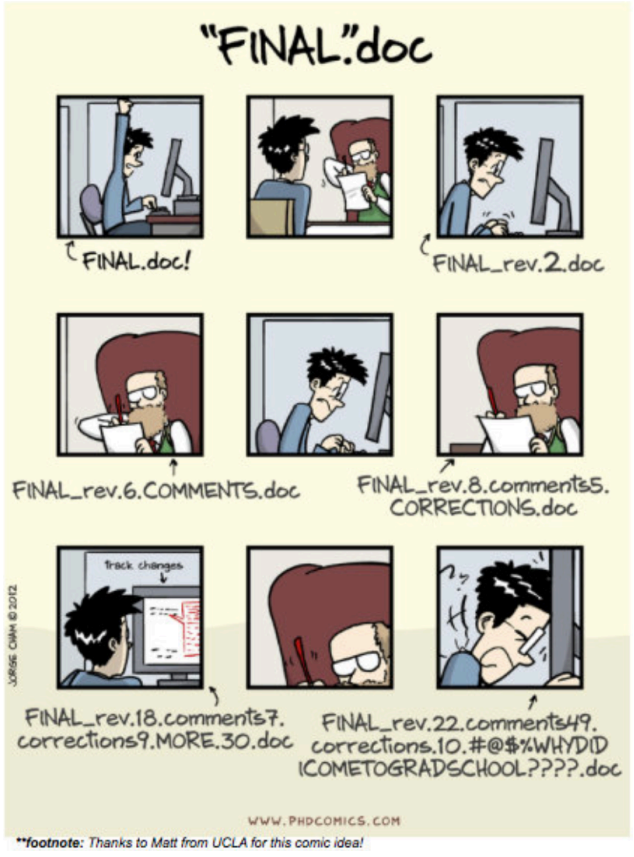
Git Fundamentals

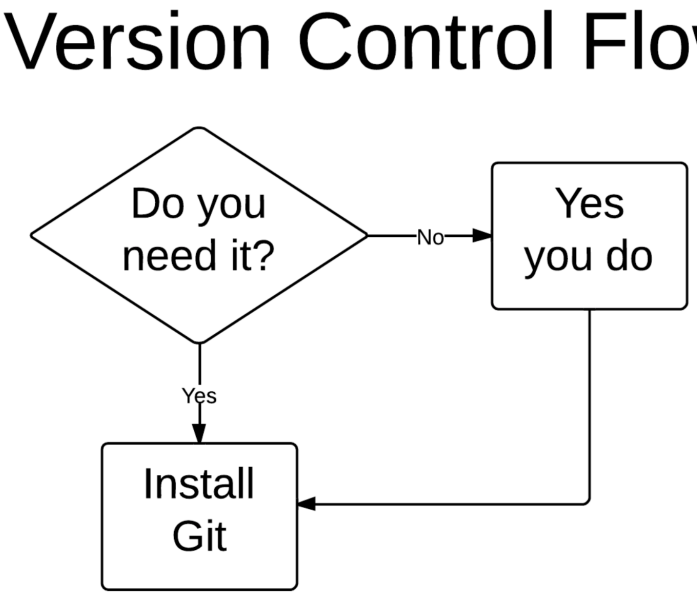
Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Agenda

- What is version control?
- Setting up git
- Create your first repository
- The ”Fork and Branch” workflow

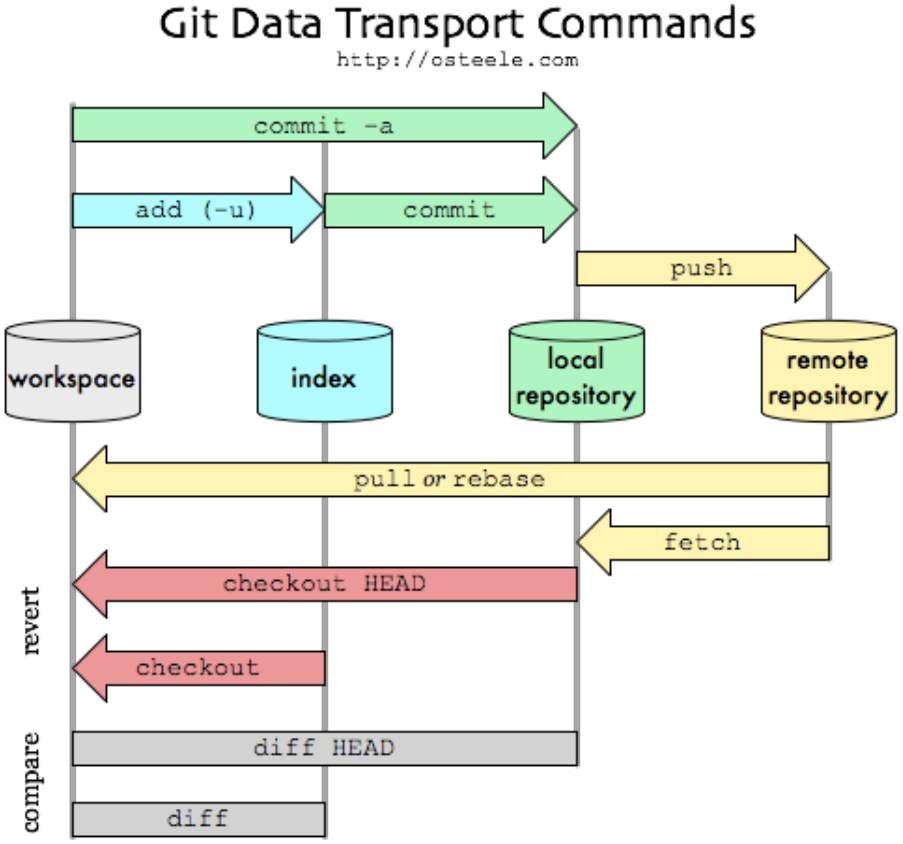
Why do we need version control?





Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

How does it work?



Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Agenda

- What is version control?
- **Setting up git**
- Create your first repository
- The ”Fork and Branch” workflow

Install git

A very good tutorial:

- https://happygitwithr.com/install-git.html

Install Git for Windows:

https://gitforwindows.org

NOTE: When asked about “Adjusting your PATH environment”, make sure to select “Git from the command line and also from 3rd-party software”. Otherwise, we believe it is good to accept the defaults.

The same very good tutorial:

- https://happygitwithr.com/install-git.html

Install Git for Mac:

Go to the Terminal and and install the Xcode command line tools (which include git) by entering:

“xcode-select –install”

Or if you use Homebrew:

“brew install git”

Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Configure Git

- On Windows:
  - Enter the Git Bash shell
- On Mac and Linux:
  - Enter the shell

- Configure your git:
  - *git config --global user.name “Max Mustermann”*
  - *git config --global user.email “max.mustermann@mail.com”*

Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Agenda

- What is version control?
- Setting up git
- **Create your first repository**
- The ”Fork and Branch” workflow

Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Create your first repository

- Go to https://github.com/ and create an account (make sure to use the email address from your git config)
- Click green “New repository” button
- How to fill this in:
  - Repository name: myrepo (or whatever you wish, we’ll delete this soon anyway).
  - Description: “testing my setup” (or whatever, but some text is good for the README).
  - Public
  - YES Initialize this repository with a README

Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Clone the repo to your local computer

- Go to the shell
- Make a new directory and move into it
  - *mkdir test*
  - *cd test*
- Clone your repository (copy the link from github.com)
  - *git clone https://github.com/YOUR-USERNAME/YOUR-REPOSITORY.git*

Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Make a local change, commit, and push

- Add some text to the README with the editor of your choice
- Verify that git notices the change:
  - *git status*
- Stage and commit this change and push it to your remote repo on github
  - *git add -A*
  - *git commit –m “My first commit”*
  - *git push origin master*

Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Clean up

- Delete the local test folder
- Delete the repository on github.com

Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Agenda

- What is version control?
- Setting up git
- Create your first repository
- **The ”Fork and Branch” workflow**

Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

The PDS repository

- We provide a repository for our class at:
  https://github.com/matjesg/PDS19.git

- We will use this repository to upload:
  - Jupyter Notebooks from the lectures
  - Exercises and solutions
  - Interesting and helpful stuff

- Additionally, you will push your solutions for the exercises and the project to the repository

BUT: Don’t clone the repository

Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

The fork and branch workflow

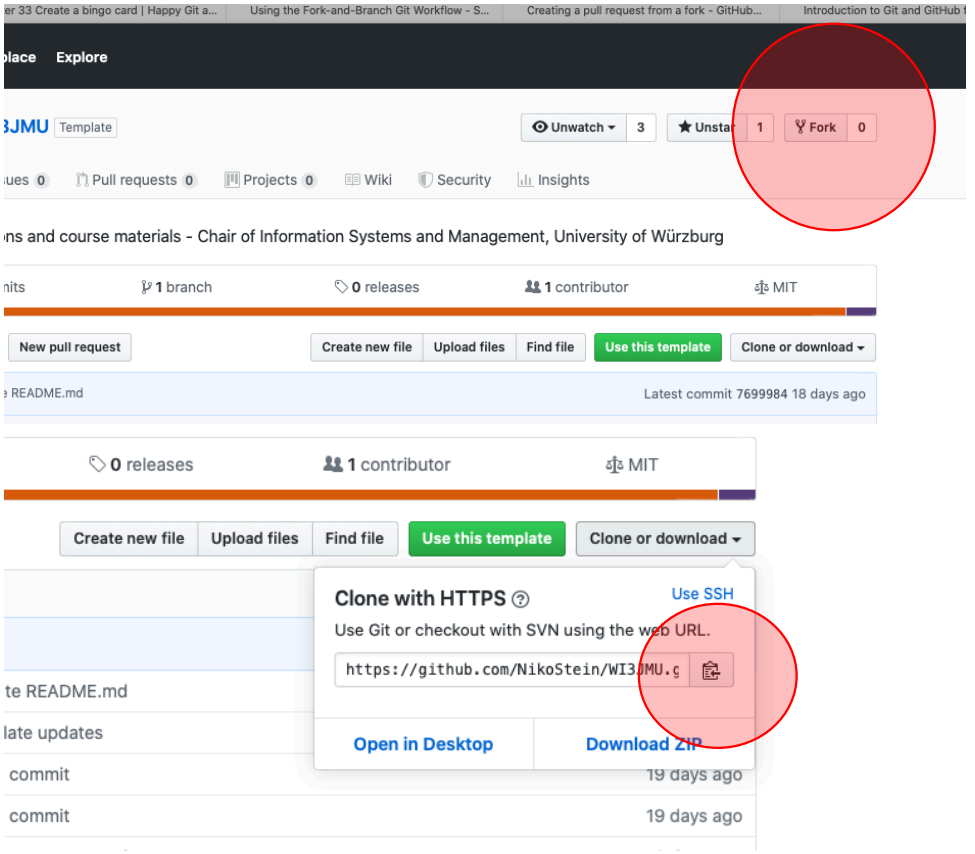
- The “fork and branch” workflow is a common way of collaborating on projects
- Basically, the “fork and branch” workflow looks like this:
  - Fork a GitHub repository
  - Clone the forked repository to your local system
  - Add a Git remote for the original repository
  - Create a feature branch in which to place your changes
  - Make your changes to the new branch
  - Commit the changes to the branch
  - Push the branch to GitHub
  - Open a pull request from the new branch to the original repo
  - Clean up after your pull request is merged

- For more details see https://blog.scottlowe.org/2015/01/27/using-fork-branch-git-workflow/

Fork the PDS repository and make a clone

- First, fork our PDS repository:
  - Go to www.github.com
  - Make sure you're logged into GitHub with your account
  - Find the GitHub repository with which you'd like to work on
  - Click the Fork button on the upper right-hand side of the repository's page

- Clone the fork to your local machine using
  git clone https://github.com/abc/abc.git



Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Add a git remote

- If you were interested in making a fork of the current state of the project and not contributing back to the origin repository, you could stop here
- But:
  - You have to frequently pull changes from the class repository
  - You have to push your solutions for the exercise and the final project to the repository of our class
- Therefore, you have to add a Git remote pointing back to the **original** repository like this:
  git remote add upstream https://github.com/abc/abc.git
- Now, you can update your local copy from the original repository:
  git pull upstream master
  git push origin master

Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Work in a feature branch

- Now you are all set-up to start making changes to your local Git repository
- However, you should use branches to effectively collaborate with others on the same repo
- To create a new branch and check it out use:
  git checkout –b <new branch name>
- You can switch you active branch by
  git checkout <branch name>
- Finally, you can work on your code and push changes to the new branch in your forked repository by
  git push origin <branch name>

Opening a pull request

- Prior to pushing your changes to the official class repo you should update your feature branch

  git checkout master *(switch to master branch)*

  git fetch -p upstream *(update from upstream)*

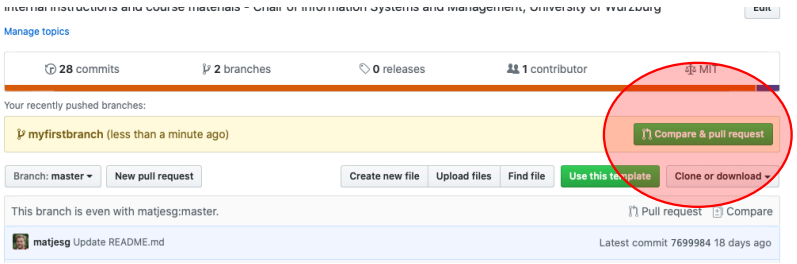
  git merge upstream/master *(merge to master)*

  git checkout <feature-branch> *(switch branch)*

  git merge master *(merge master into new)*

  git push origin <feature-branch> *(push changes)*

- To submit your changes to our official class repo you have to create a Pull Request for the feature branch on the github website



Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Cleaning up after a merged pull request

- **After** we accept your pull request and add your changes to the official repository you can clean up the fork
- First, update your local clone by using
  git pull upstream master
- Delete the feature branch
  git branch -d <branch name>
- Update the master branch in the forked repo
  git push origin master
- Delete the feature branch on your GitHub repository
  git push --delete origin <branch name>

Thank you!

Nikolai Stein
Lehrstuhl für WI & IM
Julius-Maximilians-Universität Würzburg
Josef-Stangl-Platz 2, 97070 Würzburg

nikolai.stein@uni-wuerzburg.de
www.bwl.uni-wuerzburg.de/lehrstuehle/bwl12/