Rotterdam School of Management
Erasmus University

RSM Erasmus

Git Tutorial

Dr. Jeroen Engelberts, Research Software Engineer, Information Management and Consulting

Outline

- Introduction IMC and Scientific Development Team

- Continues where “Introduction to Git” left off:
  - https://jjengelberts.github.io/eur-git-101/
- Hands-On
  - https://learngitbranching.js.org/
  - https://jjengelberts.github.io/eur-git
- Github
  - Forks, Pull requests and collaborators
  - https://guides.github.com/activities/hello-world
- Questions & Answers

IMC – Information Management & Consulting

- Consultancy
- Functional Application Management
- Learning & Innovation Team
- IT Development
  - Web Development
  - Research Software Engineering & Consulting

Govert Buijs, Erik Kemperman, Paul Kievits and Jeroen Engelberts

Email: rsec@rsm.nl

Telephone: 010-4081930

Location: Mandeville (T) 06-13

Version Control – What & Who?

- What is version control?

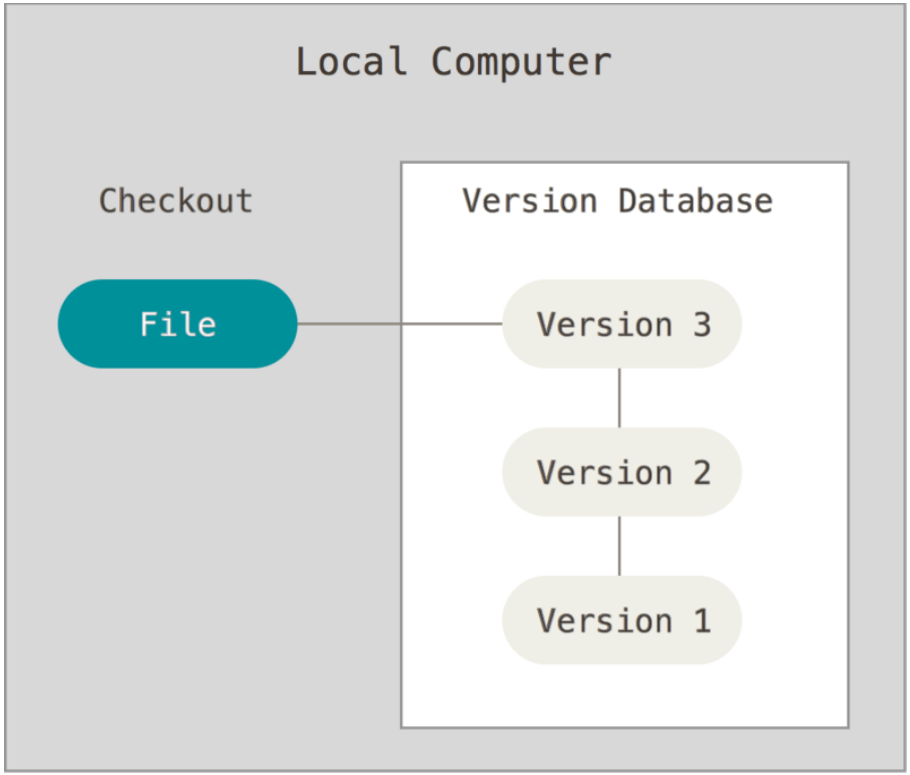
  "Version control is a system that records changes to a file or set of files over time so that you can recall specific versions later." [Pro Git, Apress, 2nd – 2014, Scott Chacon, Ben Straub]
- Who should use, or who could benefit from version control?
  - Any software developer
  - Anyone writing a bunch of articles, like a PhD thesis ☺
  - Any group of people working together with many (text) files

Version Control – Why?

- Advantages
  - Differences over time can be reviewed easily
  - Changes can be unrolled
  - It enables concurrent working on the same files
  - Changes are linked to individuals
  - Logging is kept (metadata) for each change
    - Date
    - Time
    - Description

Local Version Control System

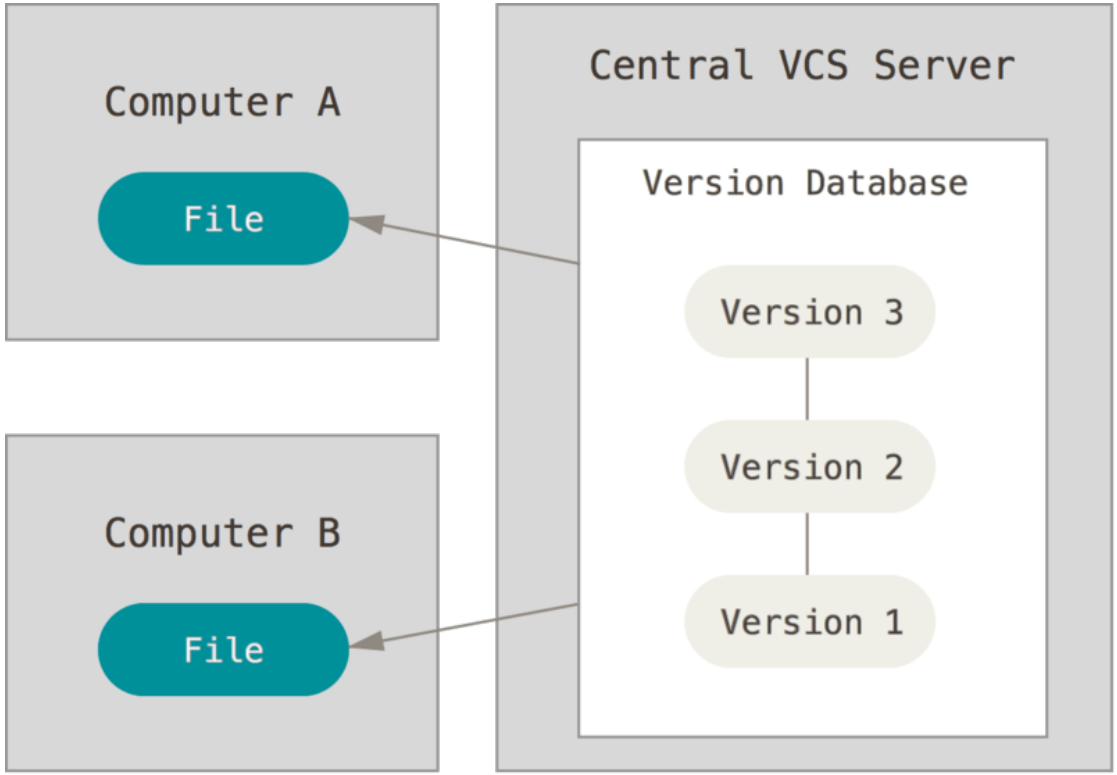
- Pro
  - Version control!
- Cons
  - Single user
  - Limited collaboration



Images from: Pro Git, Apress, 2nd – 2014, Scott Chacon, Ben Straub

Centralized Version Control System

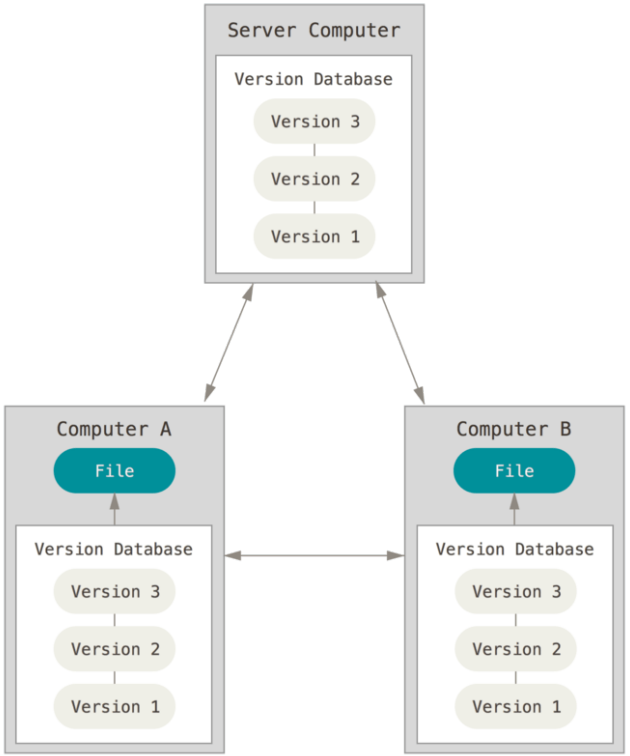
- Pros
  - Version control!
  - Multiple users
  - Collaboration
- Cons
  - Revisions not local
  - Merging can be challenging



Images from: Pro Git, Apress, 2nd – 2014, Scott Chacon, Ben Straub

Distributed Version Control System

- Pros
  - Version control!
  - Multiple users
  - Collaboration
  - Revisions are locally cloned
- Con
  - Merging complete branches can be hard or even impossible



Images from: Pro Git, Apress, 2nd – 2014, Scott Chacon, Ben Straub

Git commands (recap of 101)

- git clone – Clones a remote repository
- git add – Stage files and directories to be committed
- git commit – Commit changes to staged files

Git commits

- Commits can be traced in five ways
  - Their unique hexadecimal string
  - Their comment, or message
  - Their branch label
  - Their tag
  - Their relative position (HEAD)

Git commit, branch, merge and rebase

- Please go to the following site:
  https://learngitbranching.js.org/
- Make the exercises in the “Introduction Sequence” section

- Open up “git bash”, or a terminal where “git –V” gives you the version of Git.
- Follow along with me, I’ll show you:
  - git status & git log
  - git commit
  - git add
  - git branch
  - git merge, git rebase, git rebase -i

Conflicts

- Conflicts arise when Git is not able to unravel commits
- Occur, when many branches are used (even for a single user)
- Or, when different people are working on the same files

Follow me on “git bash” to create a conflict ourselves.

… and let’s resolve it.

HEAD, HEAD^, HEAD^*n* and HEAD~*n*

- Please go to the following site:
  https://learngitbranching.js.org/
- Make the exercises in the “Ramping Up” section

- Open up “git bash”, or a terminal where “git –V” gives you the version of Git.
- Follow along with me, I’ll show you:
  - HEAD
  - HEAD^, and HEAD^2
  - HEAD~*n*

→ Now, make exercises in the “Moving Around” and “A Mixed Bag” sections.

Github

- Hopefully, you created a Github-account (otherwise, do so now)
- Go to https://learngitbranching.js.org/
  - Click on “Remote” tab
  - Make exercises in section “Push & Pull -- Git Remotes!”
- Type along with me and create a Fork of https://github.com/jjengelberts/eur-git
- Make a modification (locally), and create a Pull Request
- I’ll show you how to handle the Pull Requests

- In the meantime, I have added your Github-account as collaborator to “eur-git”
- Clone the repo, make some changes, commit, and push.

Questions and Answers