032CM - 2025

PROGRAMMING FOR COMPUTATIONAL CHEMISTRY

Version control with Git

https://hplgit.github.io/teamods/bitgit/Langtangen_bitgit-bootstrap.html

Gianluca Levi

gianluca.levi@units.it, giale@hi.is

Office: Building C11, 3rd floor, Room 329

Fall 2025

Version control with Git

Record the **history of your files and share them among collaborators**

Document changes and ensure everyone has the latest version of the code

Go back to any previous version (a *commit*)

Why do we need it (especially in **collaborative projects**):

- You can try ideas on development versions and **revert changes if needed**
- **Multiple people can work on the *same files*** (Git can automatically merge many changes)
- Useful when you **work on several computers** (e.g. laptop, server of a computer cluster, etc.)

Project hosting services

Web platforms that **store code repositories in the cloud**

Clone and edit locally, then **synchronize** with the remote repository

Advantages over generic cloud storage (e.g. Dropbox, Google Drive)

- Designed for *projects*, not just loose files
- **Keep track of history** (commits)
- Merge changes in files that have been edited simultaneously on different computers
- See *who* changed *what*, *when*, and *why*

Project hosting services

Web platforms that **store code repositories in the cloud**

Clone and edit locally, then **synchronize** with the remote repository

Advantages over generic cloud storage (e.g. Dropbox, Google Drive)

- Designed for *projects*, not just loose files
- **Keep track of history** (commits)
- Merge changes in files that have been edited simultaneously on different computers
- See *who* changed *what*, *when*, and *why*

Main project hosting services:

- **GitHub** https://github.com/
- **GitLab** https://gitlab.com/ ⟵ We are going to use **Gitlab**

Getting started with Gitlab

1. Go to https://gitlab.com and create an account

2. Create a new **repository** (**project**)

3. Create a local copy of the repository on the server

   In the terminal:

   >> git clone <URL_of_the_repository>

Basic Git workflow and commands

git clone <URL> → Download an existing remote repository

git status → Show which files are modified, staged, or untracked

git add <file> → Stage changes in a file so they are included in the next commit

git commit -m "message" → Save a snapshot of the staged changes with a message

git push → Send your local commits to the remote repository

git pull → Fetch and merge the latest changes from the remote into your local branch

git branch → List branches

git checkout <branch> → Move to another (development) branch