Software version control

Stefan Richter (DESY, MCnet)

Why version control?

- Combine work of multiple collaborators
- Understand changes
- Support incremental development
- Compare and revert to earlier versions
- Backup
- Parallel versions
- Document development (for other developers and yourself, not for users)

→ **version control is awesome. Use it all the time.**

What is Git?

A distributed **version control system** (VCS) whose primary user interface is the Unix command line. It basically keeps a "non-human-readable" database of the files you put under version control ("track") and provides commands to access and update that database.

Graphical user interfaces, integration in Integrated Design Environments, and **web platforms GitHub/GitLab/…** have formed around the Git core software.

The aim here is not to tell you every single Git command in existence or even to teach you all the functionality. The aim is to familiarise you with the *principles of version control*, *some good practices*, and *get you started on the practical matters*.

Practical introduction by example

We're going to walk you through an example. The things we show you here will teach you all you need to know to collaborate on your team project using Git.

Setting up

To initialise a new local repository do