LESSONS

Lesson 2. An introduction version control Intro version control git

Software Carpentry, NEON Data Skills, Max Joseph, Leah Wasser

🎓 Learning objectives

At the end of this activity, you will be able to:

- Explain how version control is useful in a scientific workflow
- Define version control

☑ What you need

- A GitHub user account
- A terminal running bash, and
- Git installed and configured on your computer.

Follow the setup instructions here:

- Setup instructions

In this page, you will be introduced to the importance of version control in scientific workflows.

The text and graphics in the first three sections were borrowed, with some modifications, from Software Carpentry's Version Control with Git lessons.

What is Version Control?

A version control system maintains a record of changes to code and other content. It also allows us to revert changes to a previous point in time.

Many of us have used the "append a date" to a file name version of version control at some point in our lives. Source: "Piled Higher and Deeper" by Jorge Cham www.phdcomics.com

Types of Version control

There are many forms of version control. Some not as good:

- Save a document with a new date (we've all done it, but it isn't efficient)
- Google Docs "history" function (not bad for some documents, but limited in scope).

Some better:

- Git
- Mercurial
- Subversion

Why Version Control is Important

Version control facilitates two important aspects of many scientific workflows:

1. The ability to save and review or revert to previous versions.
2. The ability to collaborate on a single project.

This means that you don't have to worry about a collaborator (or your future self) overwriting something important. It also allows two people working on the same document to efficiently combine ideas and changes.

★ **Thought Questions:** Think of a specific time when you weren't using version control that it would have been useful.

- Why would version control have been helpful to your project & work flow?
- What were the consequences of not having a version control system in place?

How Version Control Systems Works

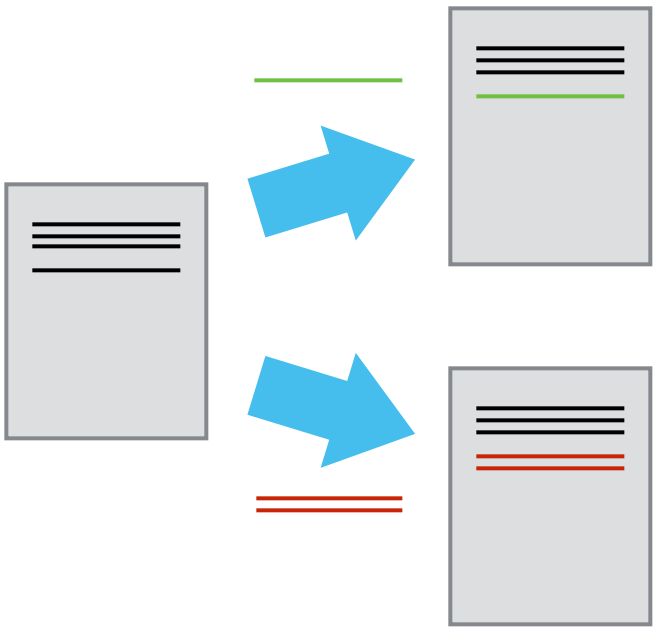
Simple Version Control Model

A version control system tracks what has changed in one or more files over time. Version control systems begin with a base version of a document. They then save the committed changes that you make. You can think of version control as a tape: if you rewind the tape and start at the base document, then you can play back each change and end up with your latest version.

A version control system saves changes to a document, sequentially as you add and commit them to the system. Source: Software Carpentry

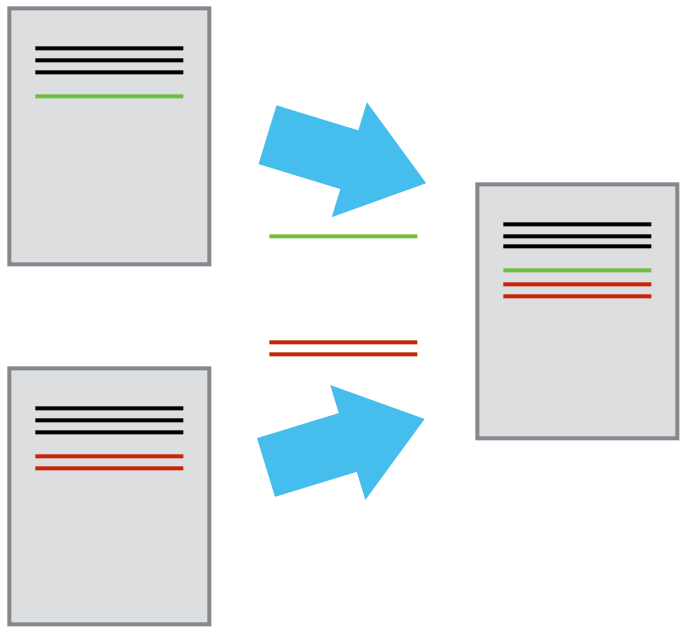
Once you think of changes as separate from the document itself, you can then think about "playing back" different sets of changes onto the base document. You can then retrieve, or revert to, different versions of the document.

Collaboration with version control allows to users to make independent changes to the same document.



Different versions of the same document can be saved within a version control system. Source: Software Carpentry

If there aren't conflicts between the users' changes (a conflict is an area where both users modified the same part of the same document in different ways) you can review two sets of changes on the same base document.



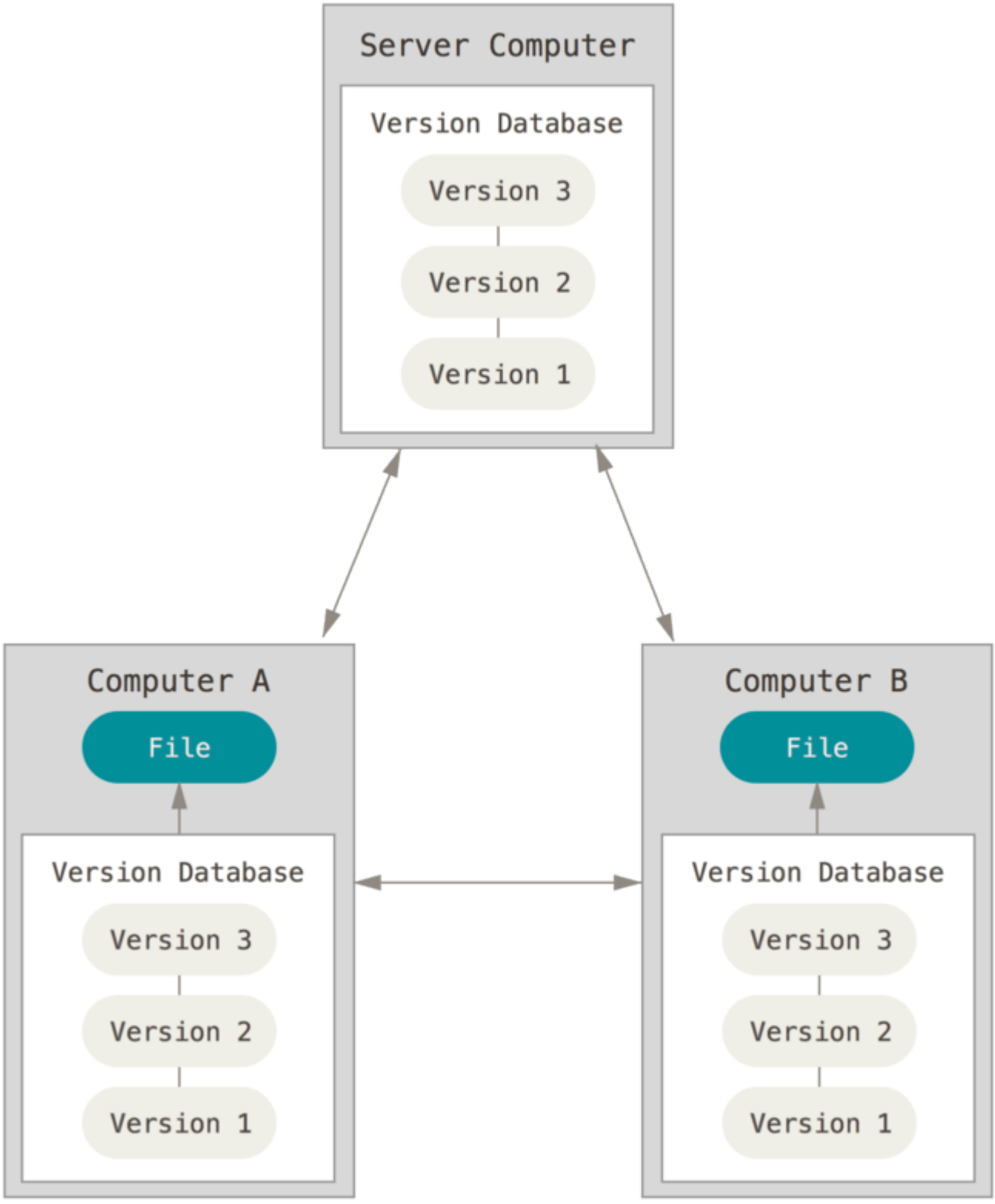
Two sets of changes to the same base document can be reviewed together, within a version control system **if** there are no conflicts (areas where both users **modified the same part of the same document in different ways**). Changes submitted by both users can then be merged together. Source: Software Carpentry

A version control system is a tool that keeps track of these changes for us. Each version of a file can be viewed and reverted to at any time. That way if you add something that you end up not liking or delete something that you need, you can simply go back to a previous version.

Git & GitHub - A Distributed Version Control Model

Git uses a distributed version control model. This means that there can be many copies (or forks/branches in GitHub world) of the repository. When working locally, git is the program that you will use to keep track of changes to your repository. GitHub is a location on the internet (a cloud web server) that acts as a remote location for your repository. GitHub provides a backup of your work, that can be retrieved if your local copy is lost (e.g., if your computer falls off a pier).

GitHub also allows you to share your work and collaborate with others on projects.



One advantage of a distributed version control system is that there are many copies of the repository. Thus, if any server or computer dies, any of the client repositories can be copied and used to restore the data! Every clone (or fork) is a full backup of all the data.
Source: Pro Git by Scott Chacon & Ben Straub

Additional resources

- About version control

- Visit the version control Wikipedia list of version control platforms.
- Read the Git documentation explaining the progression of version control systems.

Workshop overview & setup

Basic git commands

Tags **Reproducible science and programming:** git , version control
Updated: September 03, 2019