LEARN TO MAKE ALMOST ANYTHING

FAB ACADEMY 2022

https://fabacademy.org

RECITATIONS

Version Control

by Krisjanis Rijnieks (Aalto Fablab)

Overview

- What is version control?

- A brief history

- Centralized vs decentralized

- The story of Git

- How to Git

What is version control?

"The need for a logical way to organize and control revisions
has existed for almost as long as writing has existed..."

Books

	1st edition	2nd edition	3rd edition
	Version 1	Version 2	Version 3
		Foreword changes...	Making-of chapter added...

In Fab Academy version control is used to manage versions of your documentation page.

Engineering

Revision 1

Revision 2

Inner diameter changes...

Revision 3

Thickness adjustments to fit a specific design...

Website and Git

Initial version	Commit 1	Commit 2
– index.html	– index.html – about.html – me.jpg	– index.html – about.html – me.jpg – project.html – project.jpg
	Add about page...	Add final project page...

Semantic versioning

Version 1.0

– Commit 1
– Commit 2
– Commit 3

Version 1.1

– Version 1.0
– Commit 4
– Commit 5
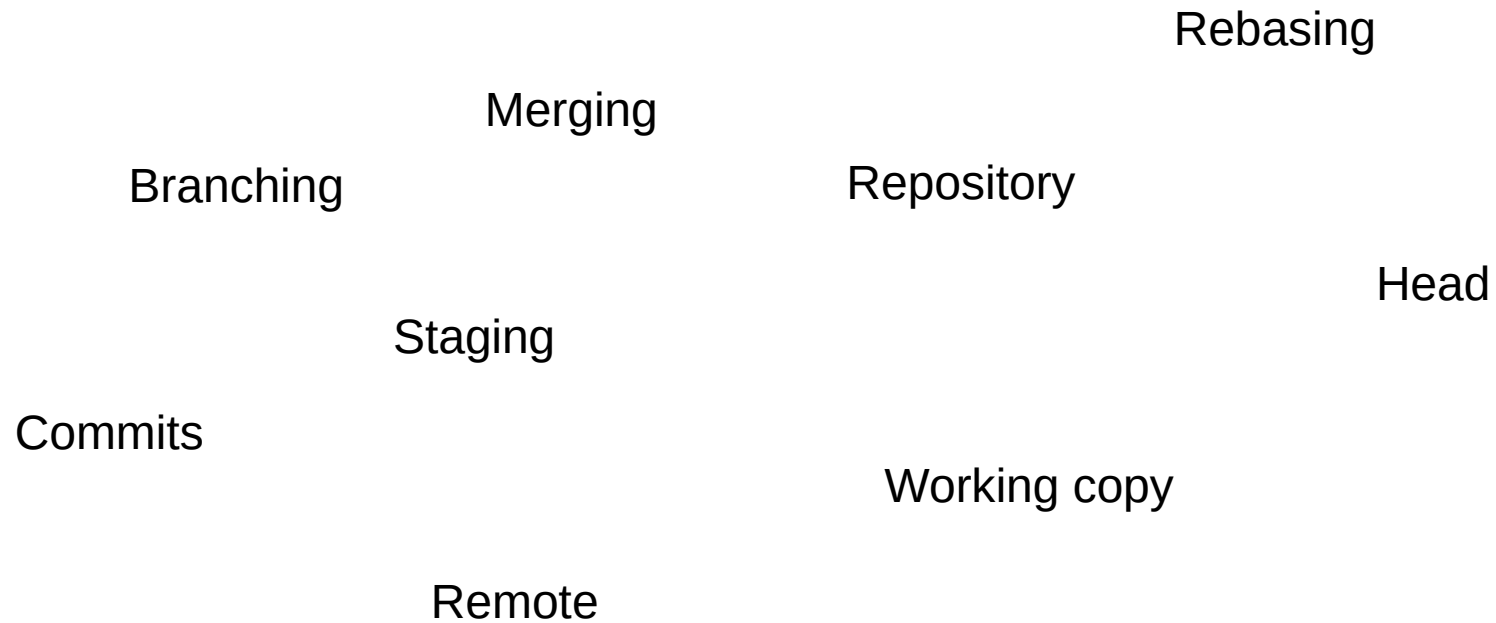
Build on top of version 1.0

Version 1.2

– Version 1.1
– Commit 6
– Commit 7

Optimize images...

Terms

Rebasing

Merging

Branching

Repository

Head

Staging

Commits

Working copy

Remote

Terms

Repository
A collection of your files that are subject to version control.

Working copy
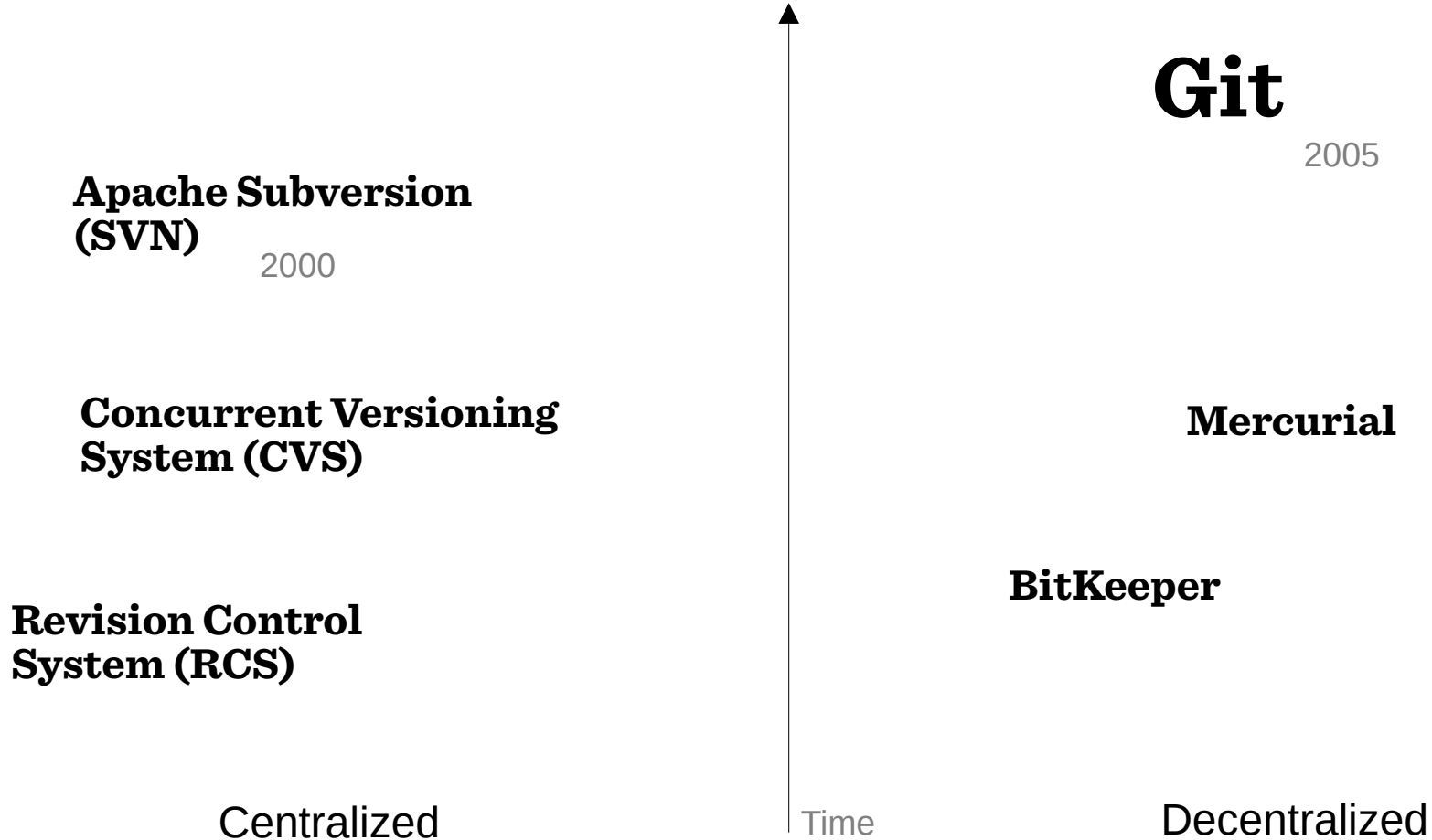Repository that is available on your computer for editing.

Commit
A collection of changes in one or multiple files in working copy submitted to the repository.

More on that later...

A brief history

"A full system (SCSS–Source Code Control System)
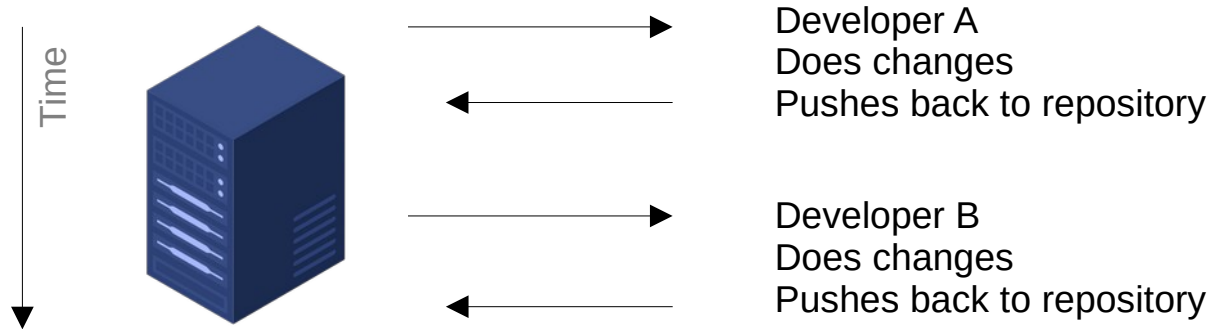designed for source code control was started in 1972..."
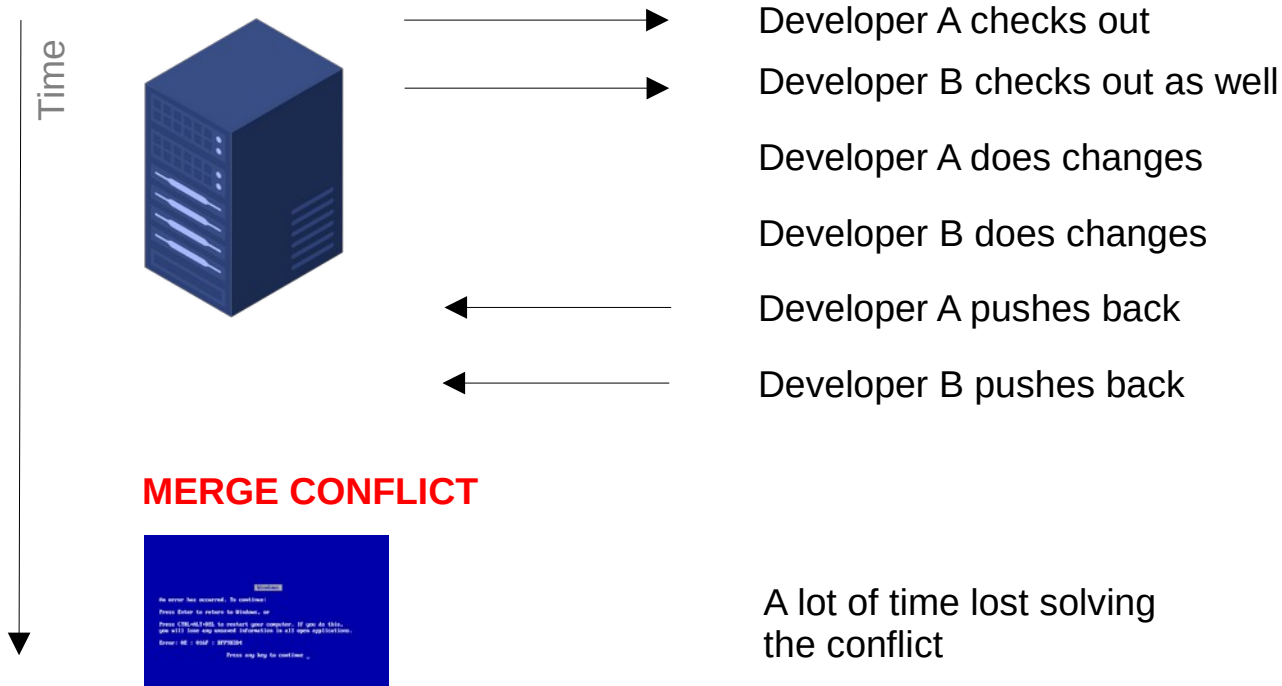
Git

2005

Apache Subversion (SVN)

2000

Concurrent Versioning System (CVS)

Mercurial

BitKeeper

Revision Control System (RCS)

Centralized

Time

Decentralized

Apache Subversion (SVN)

Centralized
Current version is stored on a computer somewhere in a megacorp

Time

Developer A
Does changes
Pushes back to repository

Developer B
Does changes
Pushes back to repository

Happy times!

Time

Developer A checks out

Developer B checks out as well

Developer A does changes

Developer B does changes

Developer A pushes back

Developer B pushes back

MERGE CONFLICT

A lot of time lost solving
the conflict

The story of Git

Basics

Initial version by Linus Torvalds
First version done over a weekend and sice then the development has been tracked using Git itself.

As an alternative to BitKeeper
BitKeeper was not open source.

To version control the Linux kernel
A new way of working needs new set of tools.

Decentralized
So that people could commit without internet connection.

Most popular version control system today (31 Jan 2022)

Read more:
https://git-scm.com/book/en/v2/Getting-Started-A-Short-History-of-Git
https://www.linuxfoundation.org/blog/10-years-of-git-an-interview-with-git-creator-linus-torvalds/

Merriam-Webster **SINCE 1828**

git

git *noun*

Save Word

\ ˈgit \

Definition of *git* (Entry 1 of 2)

British

: a foolish or worthless person

How to Git

Installations

Watch installation videos by Oleksandr

+

Fedora   **OSX**   **Windows**   **Ubuntu**

Find videos here:
https://fabacademy.org/2022/recitation/version-control/

Where are the videos?

Anatomy of a Commit

Chaos ⟶ Nice

Anatomy of a Commit

Unstaged
files

Staging
area

History
log

Chaos ⟶ Nice

Anatomy of a Commit

Unstaged
files

```
git add file.jpg
git add file.html
```

Staging
area
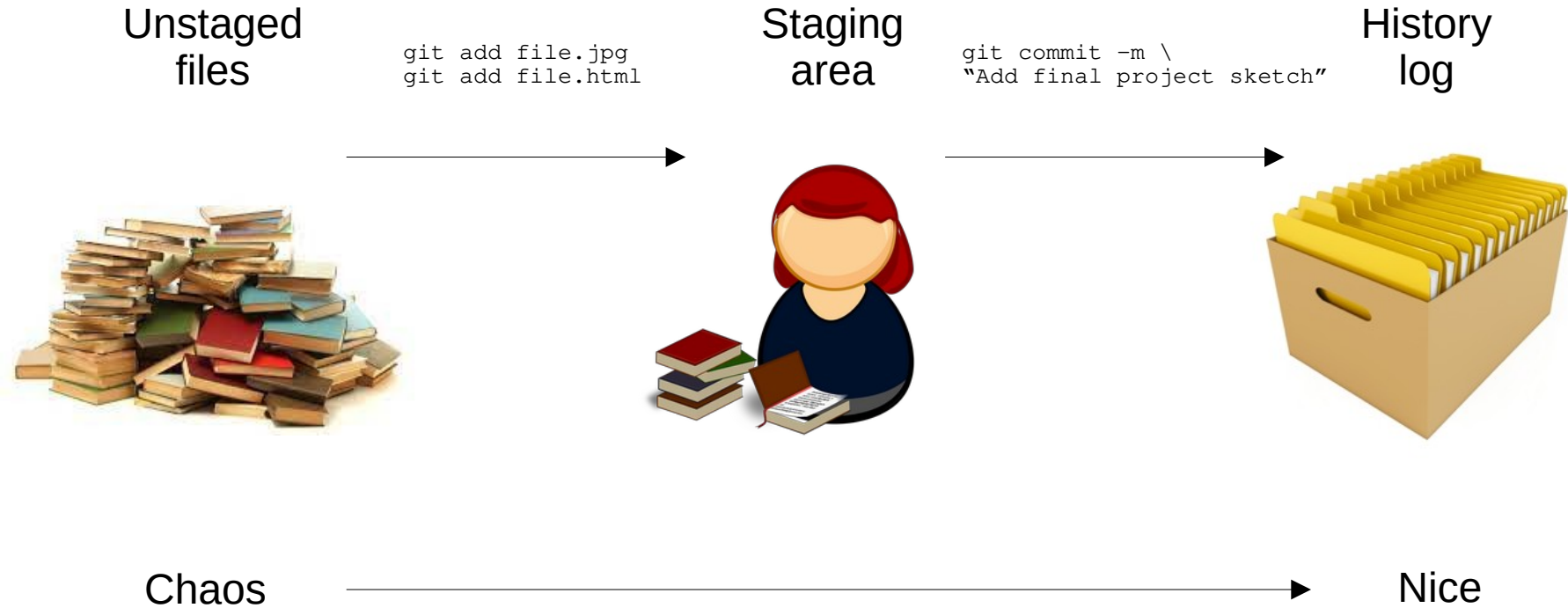
```
git commit -m \
"Add final project sketch"
```

History
log

Chaos ⟶ Nice

Anatomy of a Commit

Time

Commit 1
- index.html

Commit 2
- about.html
- me.jpg

Commit 2
- project.html
- final.jpg

From https://git-scm.com/about/staging-area
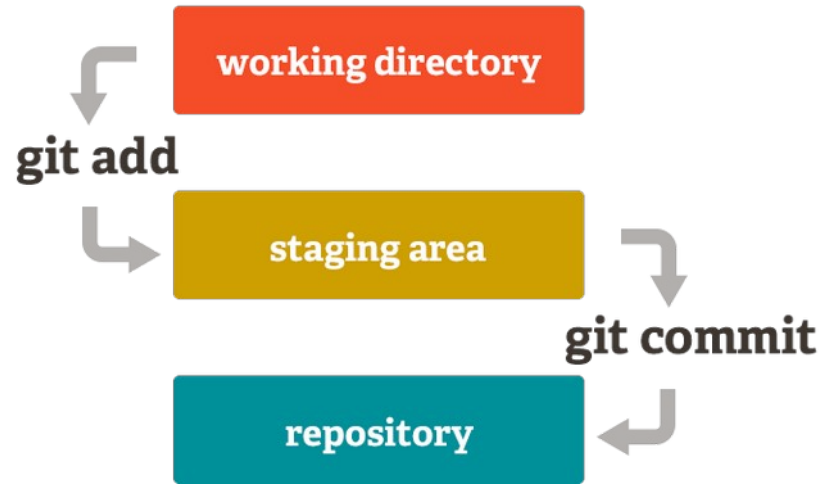
Demo

Collaboration

Branches
Create a work-in-progress (wip) branch.

```
> git checkout -b wip
```

Merging
When you are ready to push changes, switch to main branch and pull.

```
> git checkout main
> git pull
```

Rebasing
Use rebasing to avoid conflicts.
Check out your working branch again and rebase it on top of the main branch.

```
> git checkout wip
> git rebase main
> git checkout main
> git merge wip
> git push
```

The power of pull requests

Ask your instructor!