
Intro to VCS, Git, & GitHub

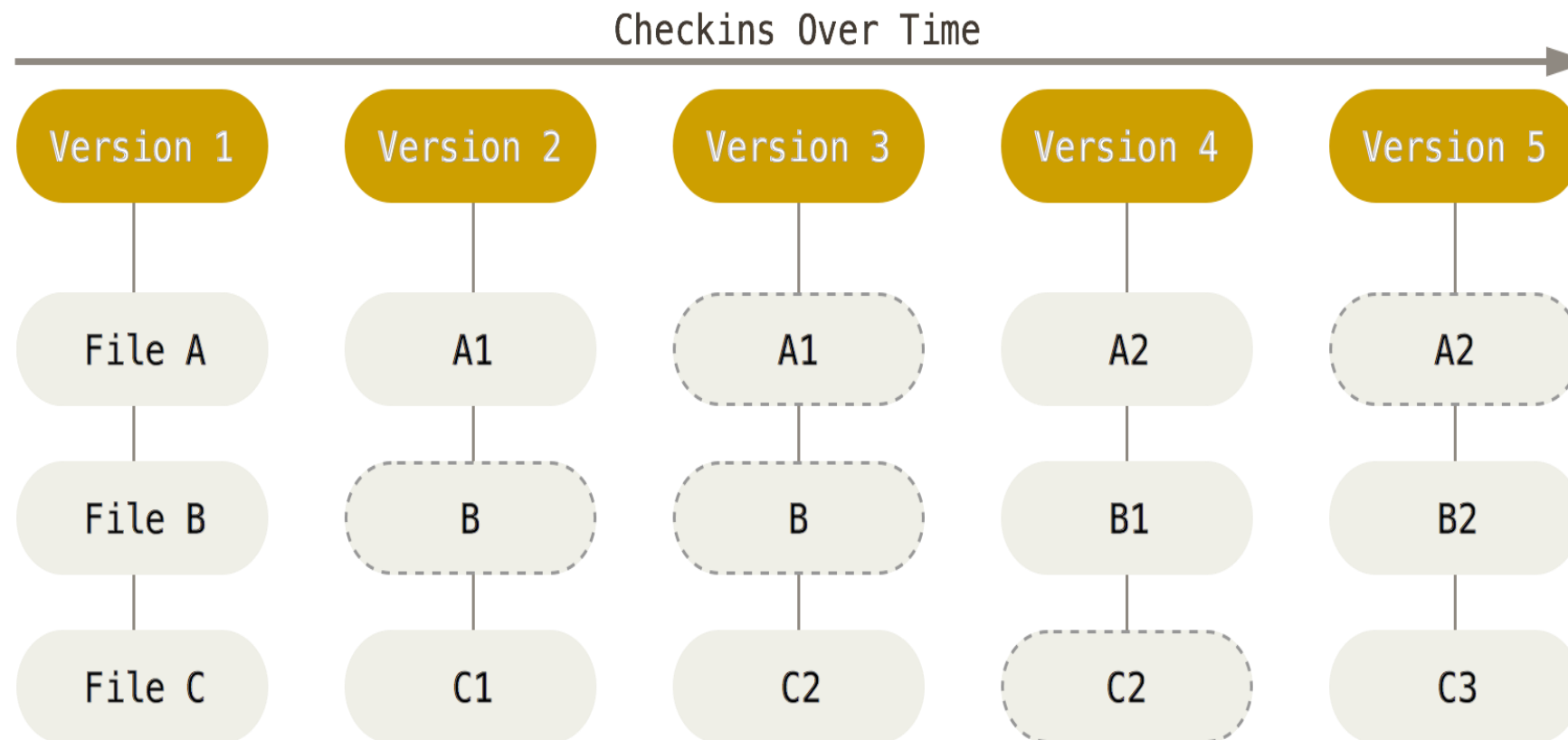
AGENDA

- What is version control?
- Github
- Creating a repo
- Cloning a repo
- Bringing to local computer
- Making changes
- Staging
- Pushing

Version Control

A system that records changes to a file or a set of files over time so that you can recall specific versions later

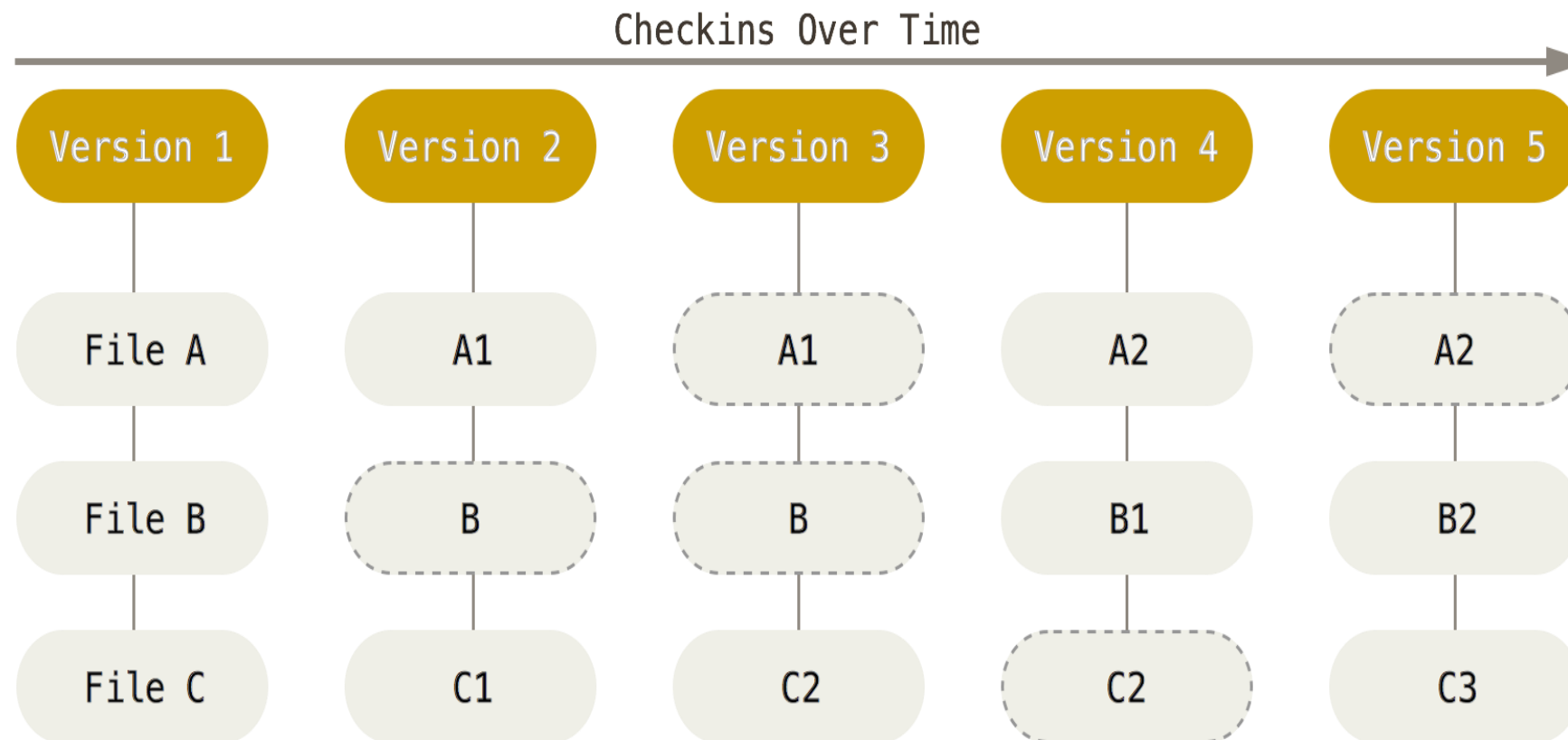
- Local Version Control
- Centralized Version Control (Subversion, Perforce, CVS, etc)
- Distributed Version Control: (Git, Mercurial, etc)



Version Control

A system that records changes to a file or a set of files over time so that you can recall specific versions later

- Local Version Control
- Centralized Version Control (Subversion, Perforce, CVS, etc)
- **Distributed Version Control:** (Git, Mercurial, etc)



VCS Benefits

- Formal system for tracking different versions of projects
- Useful for Software
- Easy Collaboration
- Involvement in Open Source Projects
- Skill for Employment

GitHub

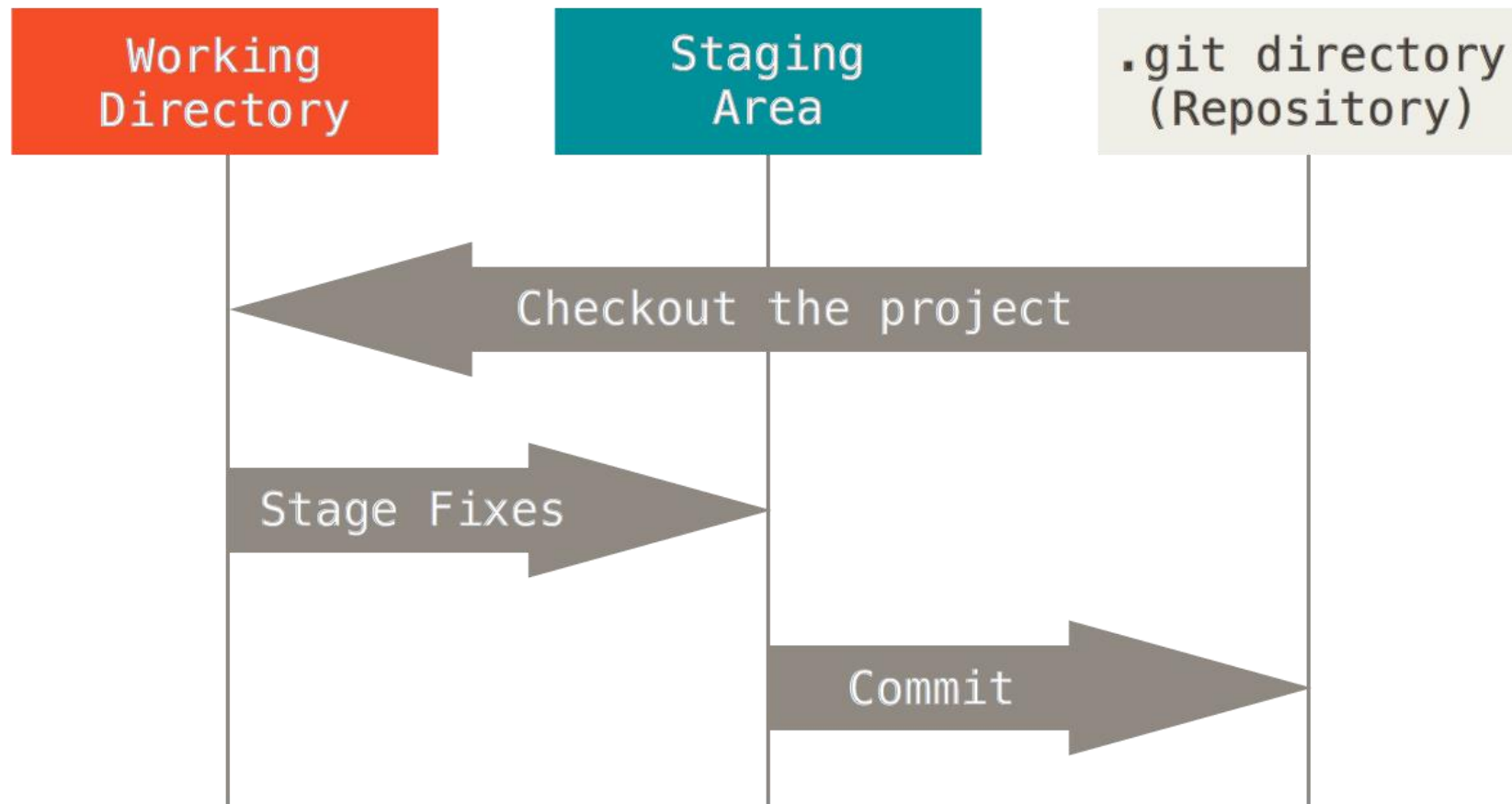
- › A website that allows you to put Git repos online
- › Not a VCS
- › Box for Git

Benefits:

- Backup
- Collaboration
- Visual Interface

Three File States

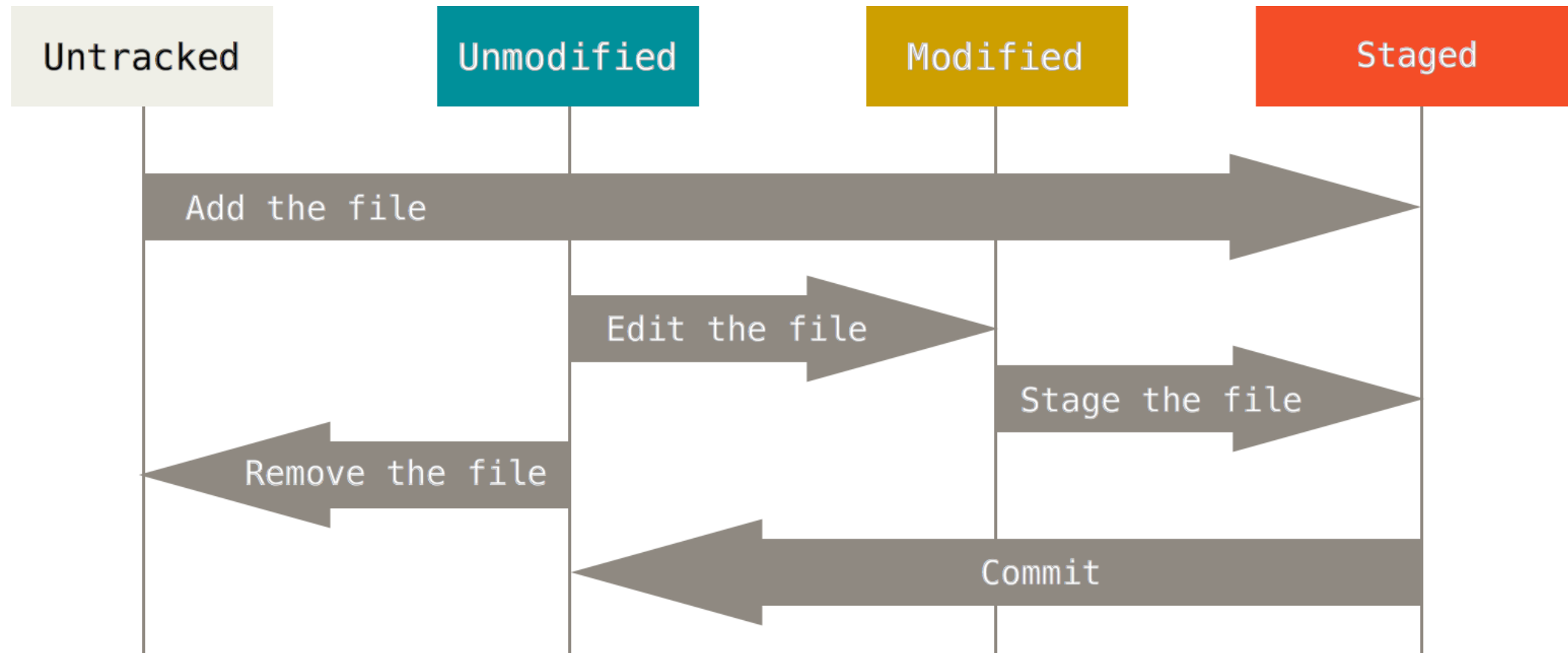
- Committed: Data stored in local database
- Modified: Changed but not committed yet
- Staged: Marked in its current version to go into next commit



3 Sections of a Git Project

- Working Directory
- Staging Area
- Git Directory

Basic Workflow



Commands



Fork a repo: Make a copy of a repo in your account.

- Log into GitHub
- Go to repository
- Click fork

Pull: Pull your changes into a repo

Create a new repository

- Click on + then new repository
- Choose name and description
- Public
- Initial readme
- Click create

Copying repo to local computer:

- Open git, terminal
- Change working directory to where you want to save the repo
- `git clone [url of repo]`
- Hit enter
- Type password
- Repo is copied to a sub directory of working directory

Commands

Status: Status of files (tracked, untracked, modified or not)

- `git status`

Staging: Prepping files for committing.

- `git add filename`
- `git add .`

Committing: Committing the changes to Git

- `git commit -m "Comment for the commit"`

Log: Log of changes

- `git log`

Push: Push the changes to the GitHub repo

- `git push origin master`
- Push master branch to the origin remote
- Type password and you're done

To Do

- Fork the course repo
- Create a new repo for homework
- Clone to local computer
- Change README.md in homework
- Commit changes
- Push to GitHub