Development Environment

Lecture 2

Hartmut Kaiser

https://teaching.hkaiser.org/spring2025/csc4103/

Git & Github

Managing Source Code Histories

Git and GitHub

- Git and GitHub are common tools used in programming
  - Help managing different versions of your code and collaborate with other developers
- Git was developed in 2005 by Linus Torvalds
  - Open source software for tracking changes in a distributed version control system
- Git is made freely available for anyone to modify and use
  - Available on all platforms, widely used
- Git tracks changes via a distributed version control system
  - Git can track the state of different versions of all files in your project
  - It is distributed because you can access your code files from another computer – and so can other developers.

Git and Github

- GitHub is a web-based platform where Git users build software together
- GitHub is also an hosting provider and version control platform you can use to collaborate on open source projects and share files
- When you're using GitHub, you're working with Git under the hood

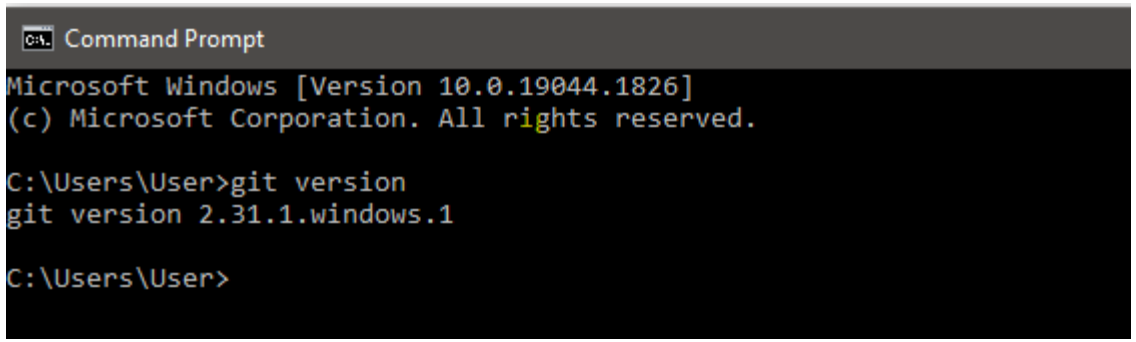
- Git is the (command-line) tool that manages the files
  - VSCode (and many other IDEs) have a graphical user interface that sits on top of Git
- Github is (one of the existing and free) web-platforms you can use to host your Git repositories

Git and Github

- Millions of people all over the world use these tools, and the numbers just keep going up
  - It is being used for any programming language
- More companies are requiring new hires to know how to use Git and GitHub
  - So if you're looking for a developer job, these are essential skills to have

Setting Things Up

- Install Git
  - Comes preinstalled in some Macs and Linux-based systems
  - Simple install for all platforms: https://git-scm.com/download



- Create account on Github: https://github.com

Connect Git to Github

- Set Git user name and email address (do this once)

```
git config --global user.name "Hartmut Kaiser"
git config --global user.email "hartmut.kaiser@gmail.com"
```

- Use same email address as you used for registering on Github

Github Classroom

- Website helping to manage assignments
- Based on starter codes in a repository
  - Manages clones (copies) of this repository for each student
  - All repositories are hosted on Github
- Enables automatic grading
- Enables individual feedback to each student
- Well integrated in VSCode

Version Control

What is Version Control?

- A database that keeps track of all changes to files over time
- Allows for collaborative development
- Allows to track who made what changes and when
- Allows to revert changes and go back to previous state

What is Git

- Distributed version control system
- Entire code and history is kept on user's machine
  - Changes can be made without internet access
  - (Except pushing and pulling from a remote server)
- One of many different version control systems
  - Subversion, Perforce, Mercurial
  - Git one of the most widely used ones

How does Git work?

- Can be complicated at first, but it is based on a few key concepts
- Based on Snapshots
  - All history is based on snapshots
  - Records what all files look like at a given point in time
  - User decides when to take snapshots and of what files
  - Can go back to any previous snapshot
    - Later snapshots are still available

How does Git work?

- Key concept: Commit
- The act of taking a snapshot
  - Verb: the user committed the code
  - Noun: the user made a new commit
- Every project is made of many commits
  - List of commits defines the timeline of changes applied to files
- Three pieces of information:
  - How did files changes from previous state
  - A reference of the commit that came before it
    - Parent commit
  - A unique hash code identifying the commit
    - Something like: `c374f26626038f020dd12f842d4dc5d67d02f59d`

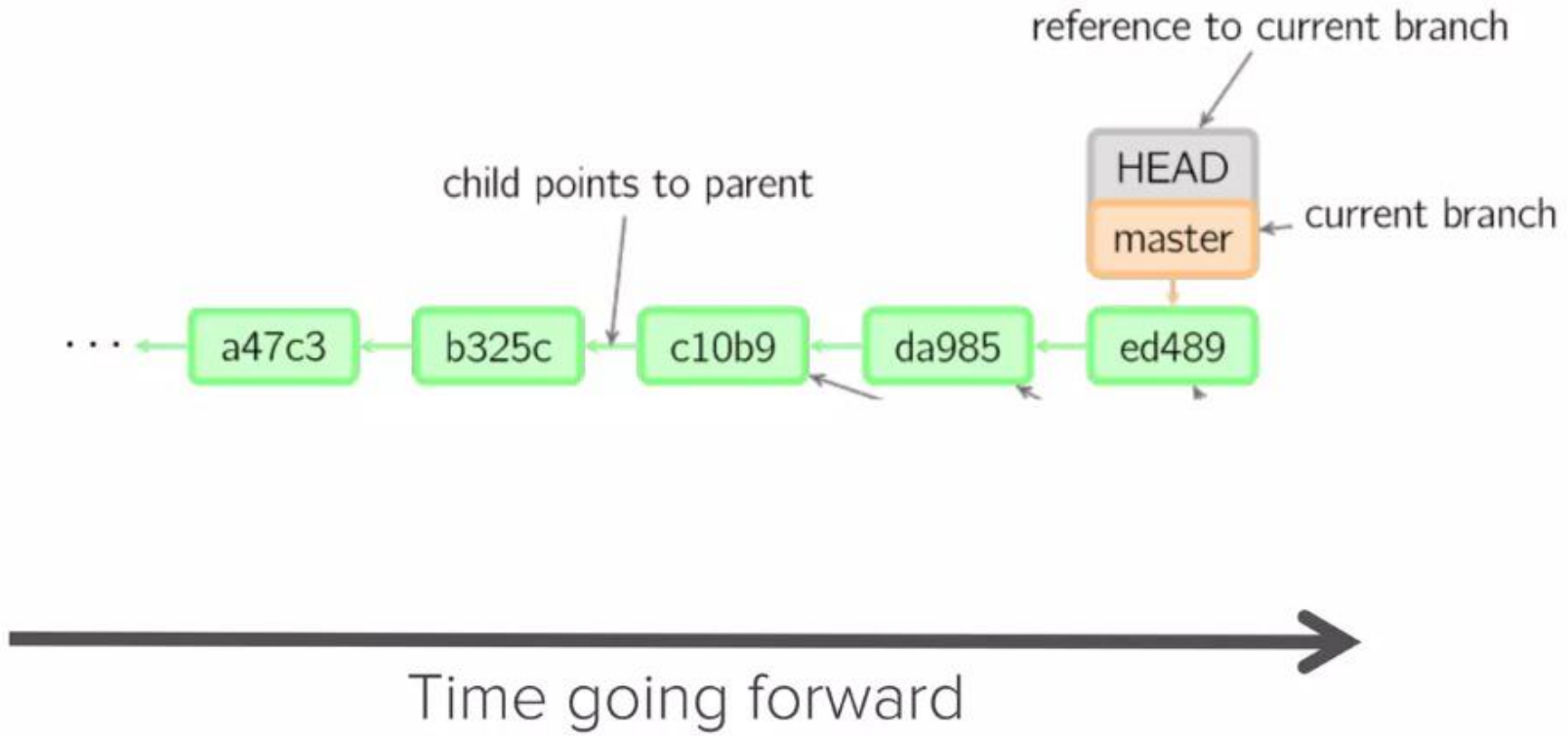
How does Git work?

- Key concept: Repositories
  - Often shortened to repo
- A collection of all files and their history
  - Consists of all commits
  - Place were all the work is stored
- Can live on a local machine or on a remote server (Github)
- Copying a remote repository to your local machine is called cloning
  - Allows for teams to work collaboratively
- Downloading commits from remote repository: pulling changes
- Adding local changes to a remote repository: pushing changes

How does Git work?

- Key concept: Branches
- All commits live on a branch
  - Each branch is a sequence of commits
- There can be many branches
- The main branch is often called main or master branch

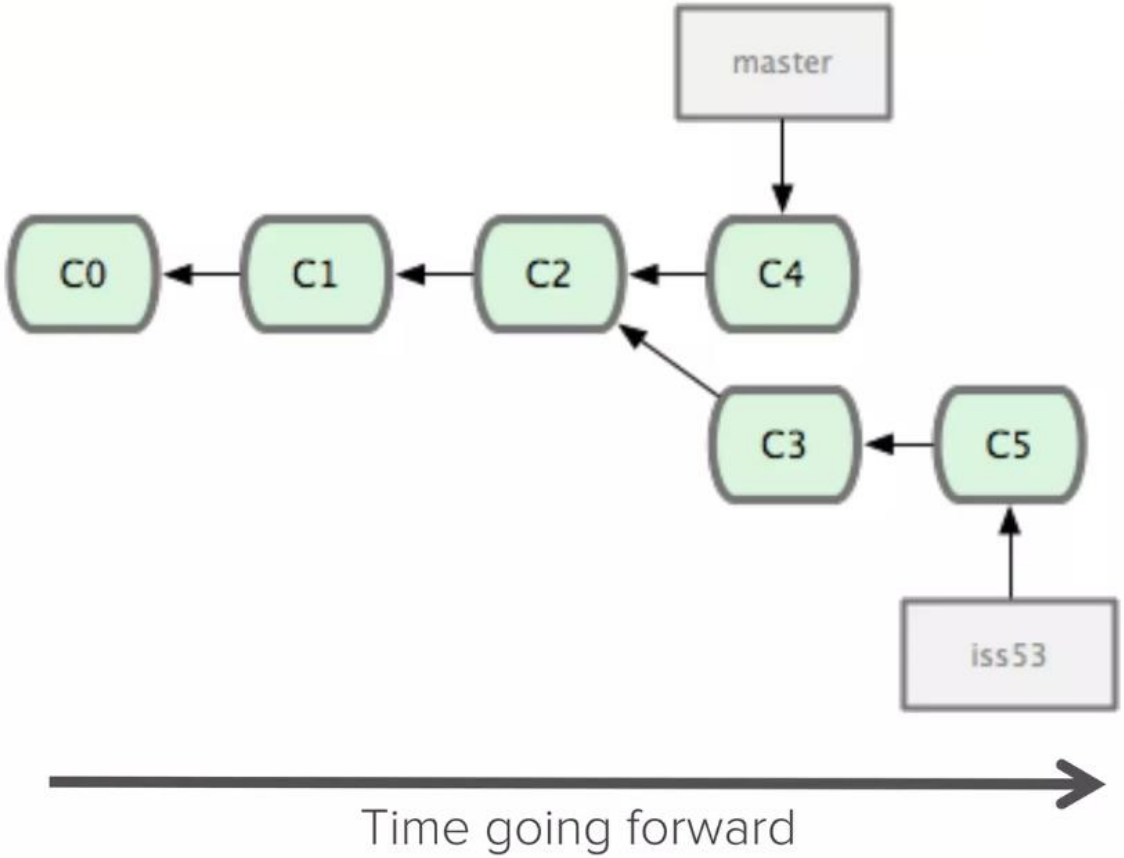
Typical Structure of a Project



Typical Structure of a Project

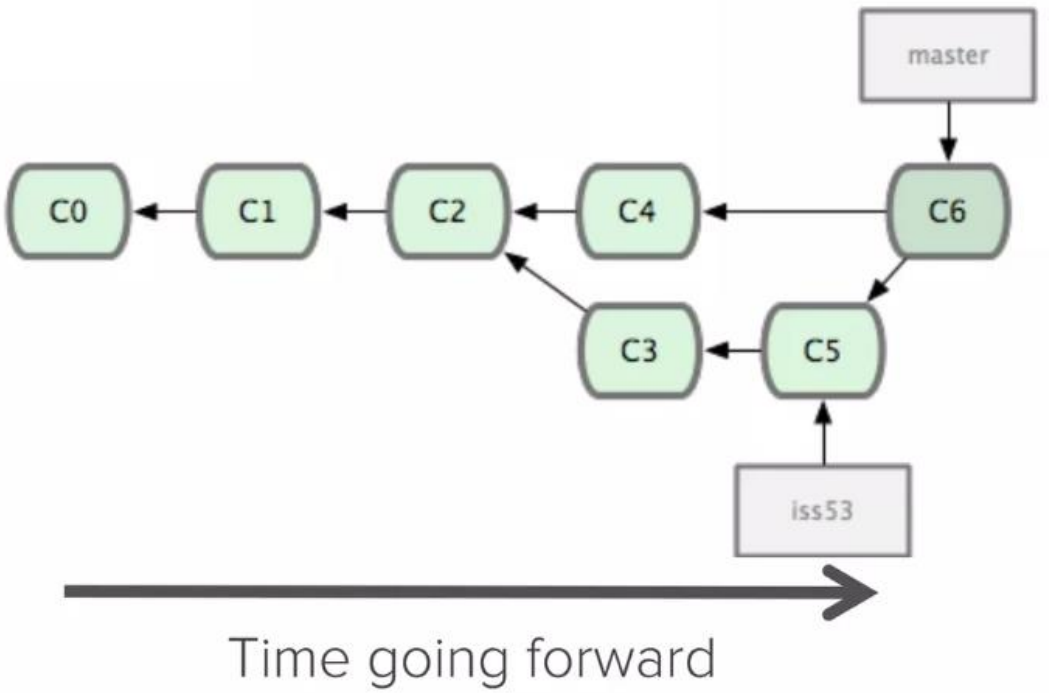
- HEAD: Reference to the most recent commit
- MASTER: The main branch in a project
  - Sometimes called ‘main’
- Key concept: branching off master branch
  - Start of a branch points to a specific commit
  - Any changes to a project should start with a new branch

Branching off master



Typical Structure of a Project

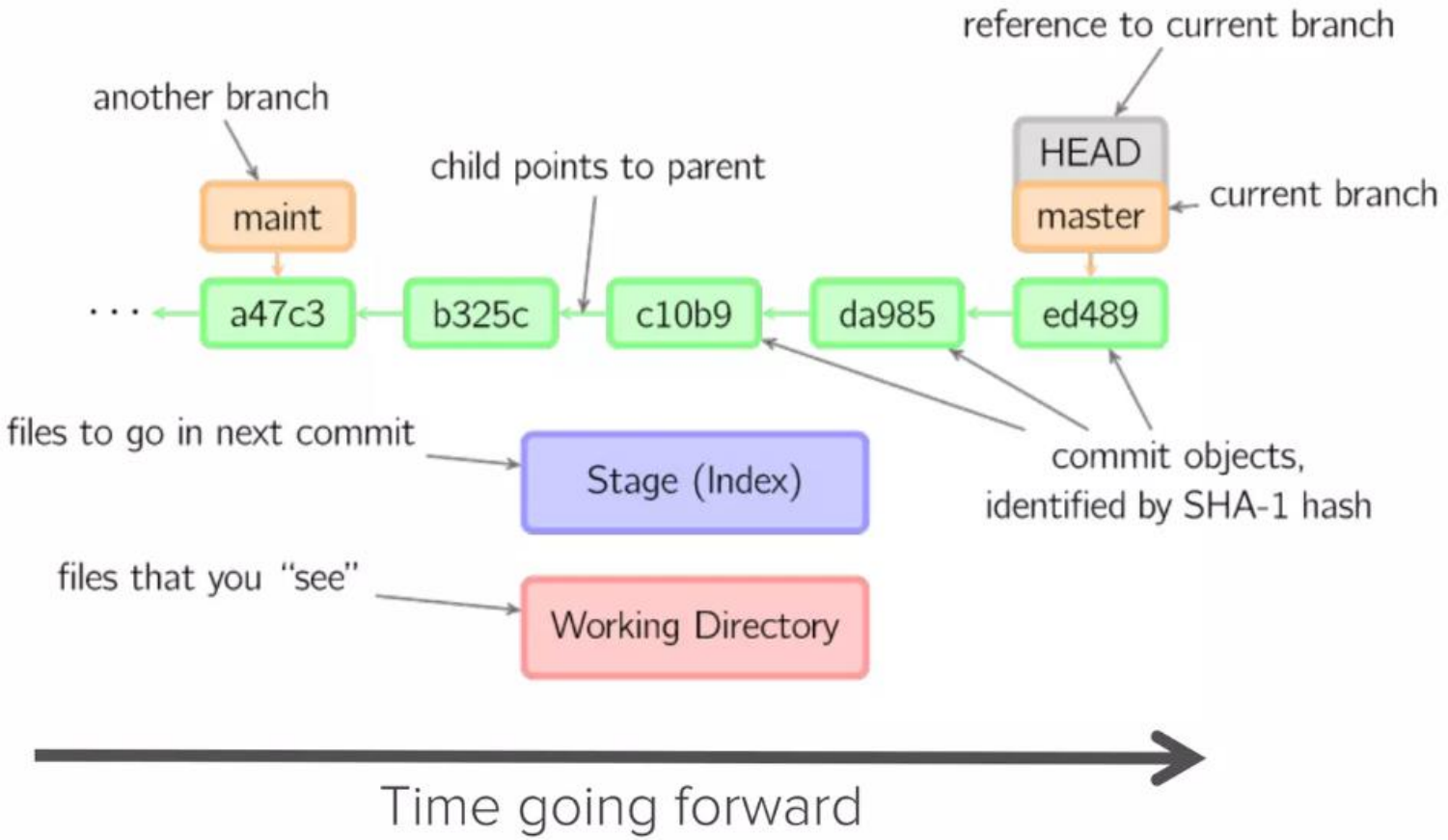
- Key concept: Merging
- Once done with a feature the branch will be merged back to master



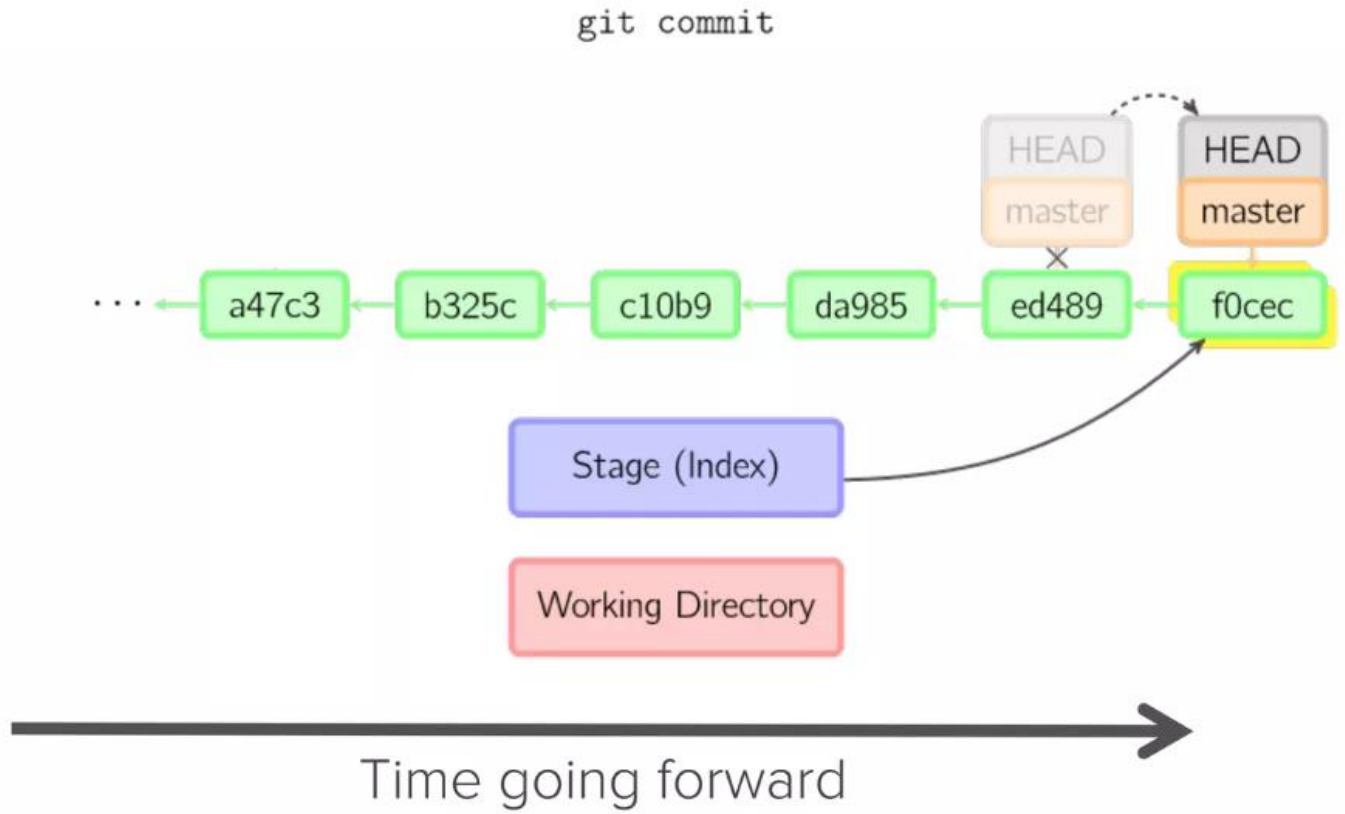
Making a Commit

- Files can be a in a lot of states and places
- Files are being edited in your local file system
  - The working directory
- A file that is ready to be committed needs to be staged (added to the index)
  - Use `'git add ...'` command to define the set of files that should be part of a commit
  - Use `'git commit ...'` command to create actual commit

Making a Commit



Making a Commit



Additional Resources

- Official Git site: https://git-scm.com
- Github guides: https://guides.github.com
- Interactive Git tutorial: https://try.github.io
- Git cheatsheet: https://www.ndpsoftware.com/git-cheatsheet.html

Remote Repositories on Github

- Pulling and pushing code
- Pull requests
- Github Classroom creates pull request ‘Feedback’
  - Allows to see all changes to start code
  - Don’t close this PR

LSU CENTER FOR COMPUTATION & TECHNOLOGY

STE||AR GROUP