



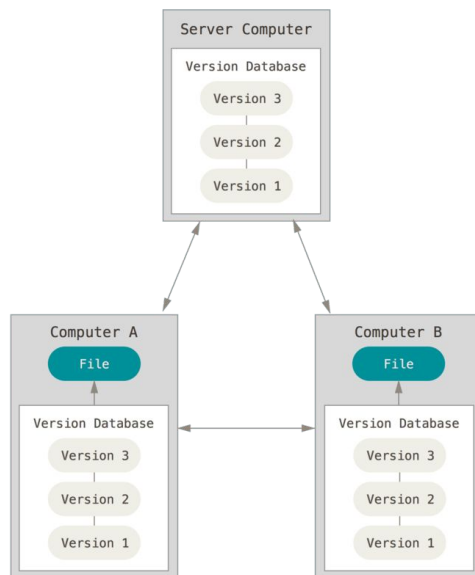
# GIT BASICS

**Sean Ovens & Jan Clarin**



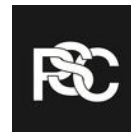
# VERSION CONTROL SYSTEMS

- What are they?
  - Systems that records changes to a set of files so that you can revert to specific versions later
- Basic idea:





# GIT - "THE STUPID CONTENT TRACKER"



- Development started by Linus Torvalds
- Spawned from BitKeeper
  - used to maintain Linux project
- "git"?

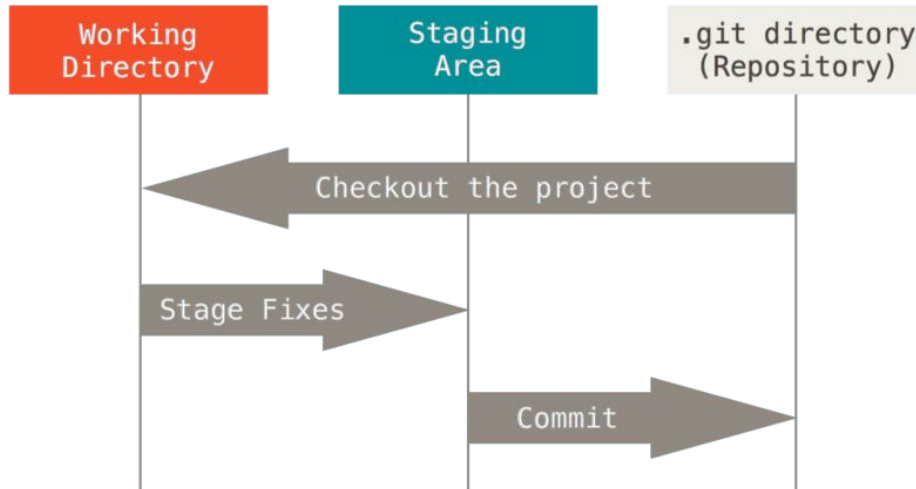
"I'm an egotistical bastard, and I name all my projects after myself. First '[Linux](#)', now 'git'."

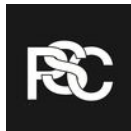


# GIT INTRODUCTION



- What is it?
  - A version control system
- The three file states:





# WHY USE GIT?

- Industry standard
- Versatile
- Can be used offline
- Extremely popular



# GETTING STARTED

- Windows: <http://git-scm.com/download/win>
- Mac: <http://git-scm.com/download/mac>
- Linux: `sudo apt-get install git-all`



# CREATING A LOCAL REPOSITORY

- Create a directory:
  - `mkdir folderName`
  - `cd folderName`
- Initialize a repository:
  - `git init`
- Create a file:
  - `touch octodex.txt`
- Add to stage:
  - `git add octodex.txt`
- Commit:
  - `git commit -m "Created octodex.txt"`

Run 'git status' after every command to see changes



- Git is a distributed version control system (VCS)
  - Others VCSs include CVS, SVN, Mercurial etc.

# GitHub

- GitHub is a remote host for Git repositories
  - Others like it include Gitlab, BitBucket, CloudForge etc.





# GitHub

- What is it?
  - Remote host for git repositories
- Account setup:
  - Go to `github.com`
  - Sign up with username, email, password



# CONFIGURATION

- Link to your GitHub account:
  - `git config --global user.name "Your Name"`
  - `git config --global user.email "your\_email@example.com"`
- Check your work:
  - `git config --list`
- Other configuration options:
  - `git config --global help.autocorrect 30`
  - `git config --global commit.template path/to/file.txt`
  - `git config --global color.* "colour name"`

etc.



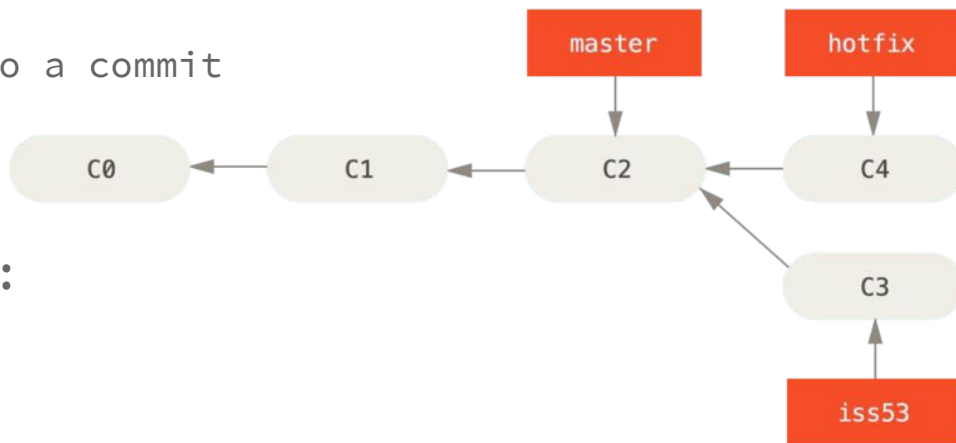
# CREATING A REMOTE REPOSITORY

- On GitHub:
  - New repository
  - Set to public
  - Create repository
- Back up your local repo:
  - `git remote add origin https://github.com/Username/Repository`
  - `git push -u origin master`
- Add contributors:
  - Settings/Contributors



# BRANCHING

- What is a branch?
  - An object that points to a commit

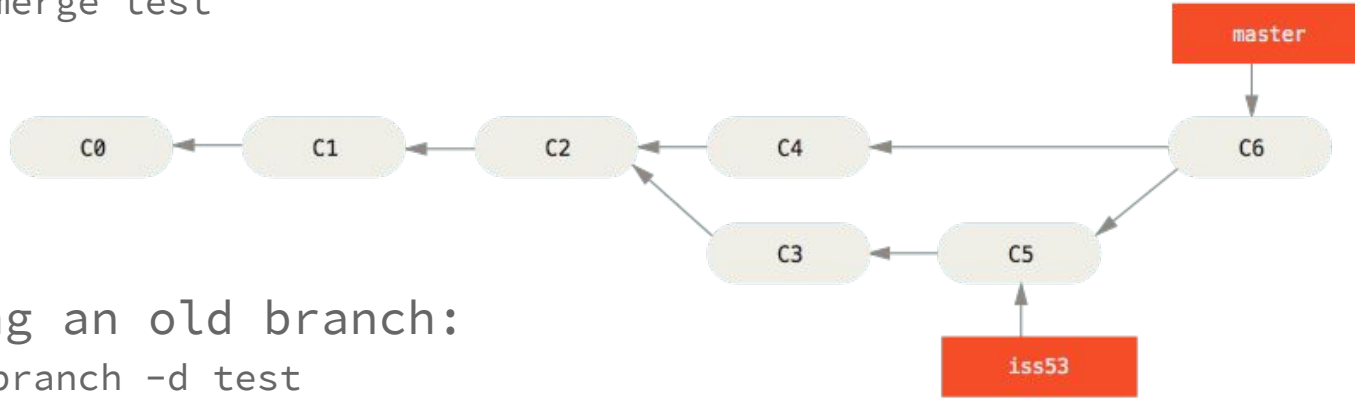


- Creating a new branch:
  - `git branch newBranch`
  - `git checkout newBranch`

Retrieved: <https://git-scm.com/book/en/v2/book/03-git-branching/images/basic-branching-4.png>

# MERGING

- Merging two branches:
  - `git checkout master`
  - `git merge test`



- Deleting an old branch:
  - `git branch -d test`

Retrieved: <https://git-scm.com/book/en/v2/book/03-git-branching/images/basic-merging-2.png>



# UNDOING

- See old commits:
  - `git log`
- Return to an old commit:
  - `git reset commitHash`
- Reset moves current branch
  - Does not change working directory
  
- Change last commit:
  - Stage changes with `git add *`
  - `git commit --amend`

# EXERCISE

Go to <https://github.com/janclarin/git-workshop>