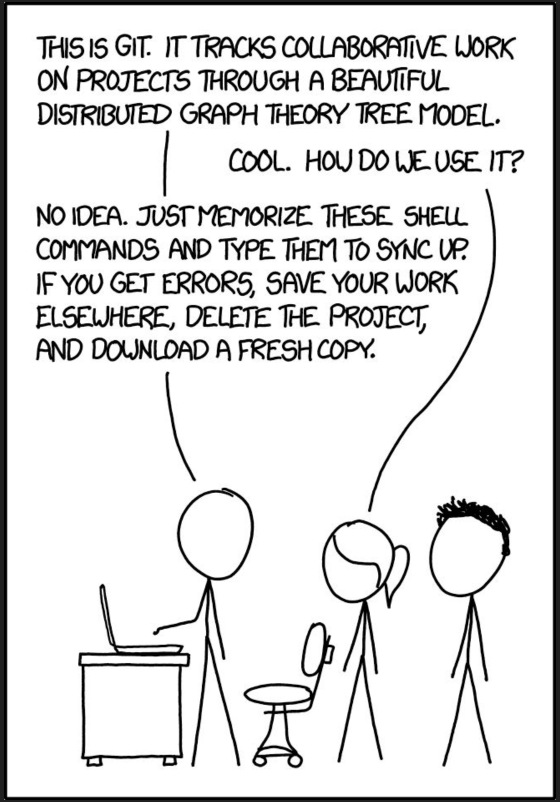
Git

TLDR:


THIS IS GIT. IT TRACKS COLLABORATIVE WORK ON PROJECTS THROUGH A BEAUTIFUL DISTRIBUTED GRAPH THEORY TREE MODEL.
COOL. HOW DO WE USE IT?
NO IDEA. JUST MEMORIZE THESE SHELL COMMANDS AND TYPE THEM TO SYNC UP. IF YOU GET ERRORS, SAVE YOUR WORK ELSEWHERE, DELETE THE PROJECT, AND DOWNLOAD A FRESH COPY.

What is git

Distributed version control system

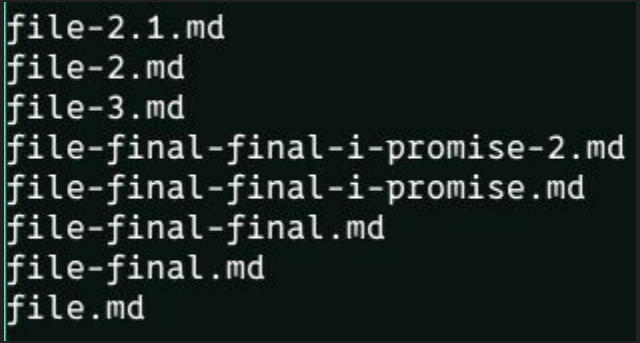
Created by Linus Torvalds in 2005

Used for tracking changes over time

Git != GitHub/GitLab

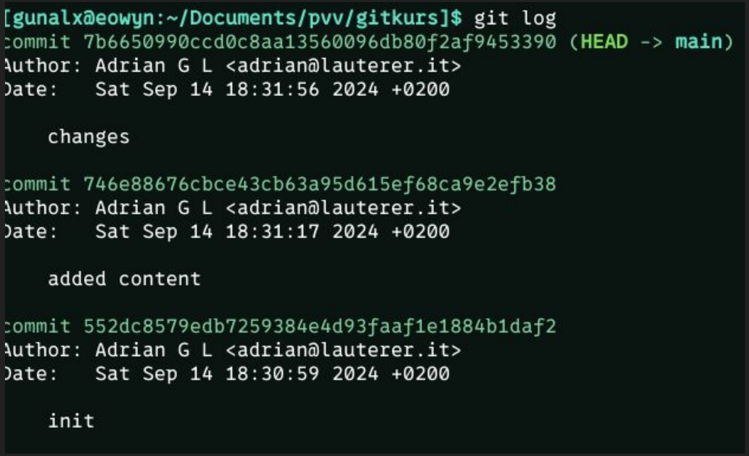
Why use git

Avoid



```
file-2.1.md
file-2.md
file-3.md
file-final-final-i-promise-2.md
file-final-final-i-promise.md
file-final-final.md
file-final.md
file.md
```

Instead



```
[gunalx@eowyn:~/Documents/pvv/gitkurs]$ git log
commit 7b6650990ccd0c8aa13560096db80f2af9453390 (HEAD -> main)
Author: Adrian G L <adrian@lauterer.it>
Date:   Sat Sep 14 18:31:56 2024 +0200

    changes

commit 746e88676cbce43cb63a95d615ef68ca9e2efb38
Author: Adrian G L <adrian@lauterer.it>
Date:   Sat Sep 14 18:31:17 2024 +0200

    added content

commit 552dc8579edb7259384e4d93faaf1e1884b1daf2
Author: Adrian G L <adrian@lauterer.it>
Date:   Sat Sep 14 18:30:59 2024 +0200

    init
```

Why use Git

- Track changes over time
- Collaborate async with others
- Revert to previous states
- Test Branches
- Blame/Praise authors

Concepts

Repository - A “folder” git tracks the changes of

Commit - A set of changes.

Branch - Pointers to commits

Merge - Combine branches Or commits

Rebase - Put a branch or commit on top of changes of another commit

Remote - A remote server the code can be synced (pushed or pulled) from/to

Create a repository

- git init - Initialize a new local Git repository
- git clone - Copy a repository from a remote source

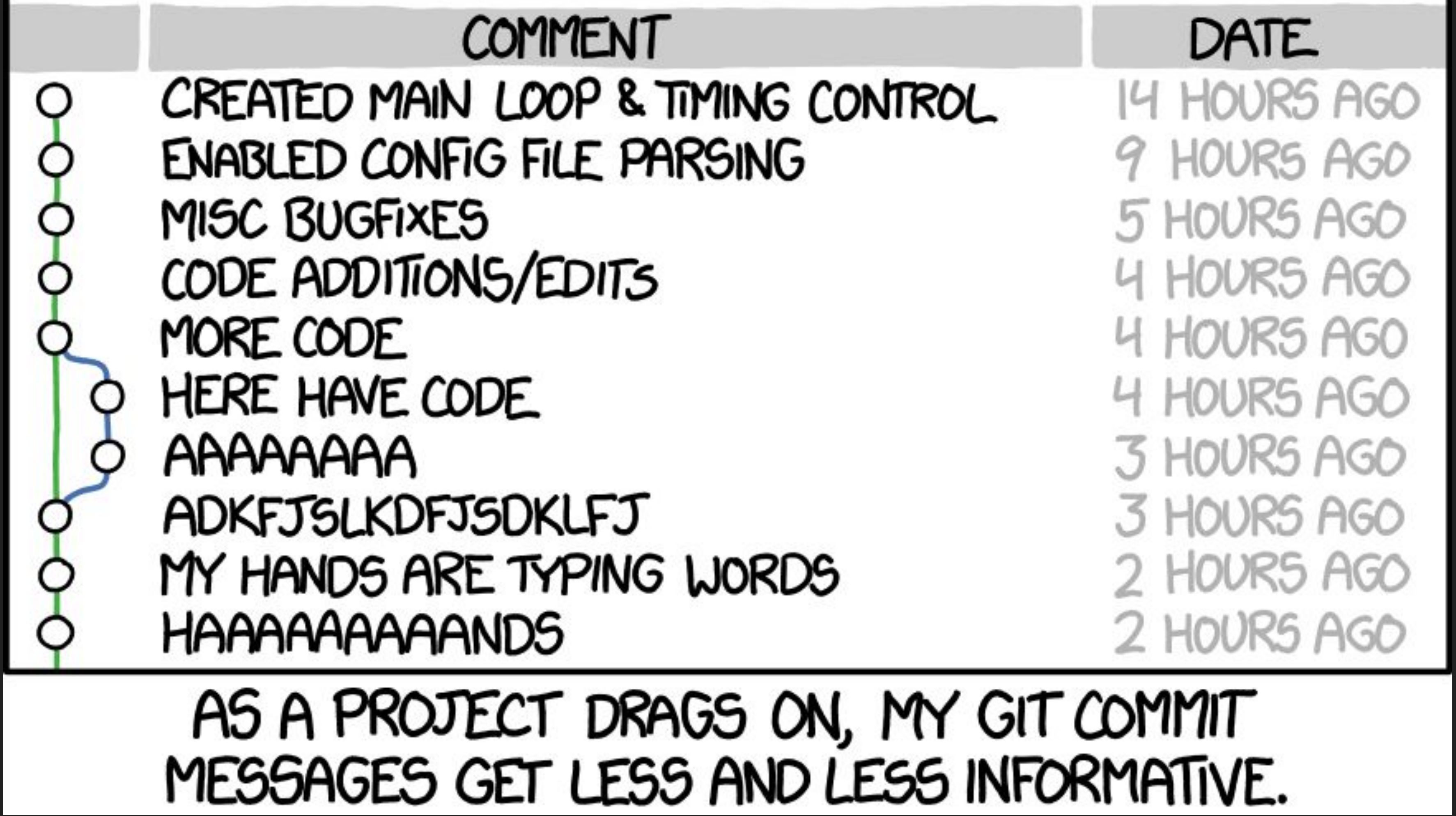
Do changes

git add - Stage changes for commit, select each file you want.

git commit - Track staged changes

Best Practices

- Commit often: Make atomic commits.
- Clear messages: Follow Conventional Commits.
- Use branches: For new features/experiments.
- Review changes: Before merging.
- Protect main: In collaborative projects.



	COMMENT	DATE
○	CREATED MAIN LOOP & TIMING CONTROL	14 HOURS AGO
○	ENABLED CONFIG FILE PARSING	9 HOURS AGO
○	MISC BUGFIXES	5 HOURS AGO
○	CODE ADDITIONS/EDITS	4 HOURS AGO
○	MORE CODE	4 HOURS AGO
○	HERE HAVE CODE	4 HOURS AGO
○	AAAAAAAA	3 HOURS AGO
○	ADKFJSLKDFJSDKLFJ	3 HOURS AGO
○	MY HANDS ARE TYPING WORDS	2 HOURS AGO
○	HAAAAAAAANDS	2 HOURS AGO

AS A PROJECT DRAGS ON, MY GIT COMMIT MESSAGES GET LESS AND LESS INFORMATIVE.

Sync changes

git push - Upload local commits to a remote repository

--force-with-lease

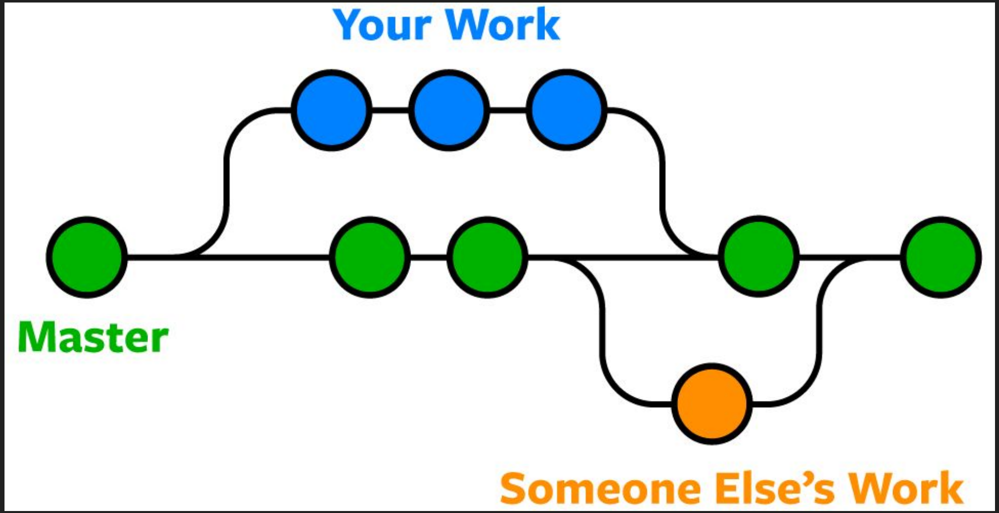
git pull - Download changes from a remote repository.

--rebase to rebase your local changes not in remote on top of remote state.

Branches

git branch <branch-name> or git switch -c <branch-name>

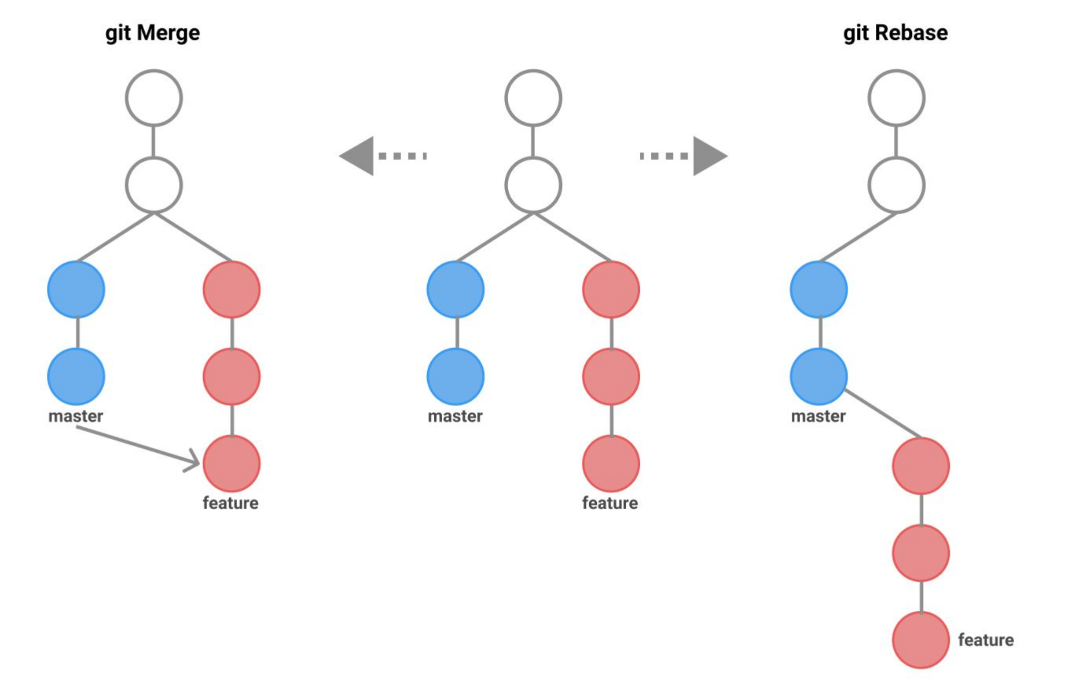
git checkout <branch-name> or git switch <branch-name>



Rebase & Merge

git merge

git rebase



https://www.atlassian.com/git/tutorials/merging-vs-rebasing

Git remotes

SSH or HTTPS

git remote list

git remote add navn url

git remote get-url navn

git remote set-url navn url

Hosting locations

Gitea -> git.pvv.ntnu.no

GitHub -> github.com eller i fremtiden git.ntnu.no

GitLab -> gitlab.com eller https://gitlab.stud.idi.ntnu.no/

Bitbucket -> bitbucket.org

LIKELIHOOD YOU WILL GET CODE WORKING BASED ON HOW YOU'RE SUPPOSED TO INSTALL IT:
VERY LIKELY
APP STORE OR PACKAGE MANAGER
GITHUB LINK
SOURCEFORGE LINK
GEOCITIES/TRIPOD LINK
COPY-AND-PASTE EXAMPLE CODE FROM PAPER'S APPENDIX
ANYTHING THAT "REQUIRES ONLY MINIMAL CONFIGURATION AND TWEAKING"
UNLIKELY

Other Nice to Know

git stash: Temporarily store modified files

git blame -L 5,30 filepath

git log

git diff

git rebase -i HEAD~n (n numbers of commits back from current)

Config

```
git config --global user.name "navn"

git config --global user.email "bruker@pvv.ntnu.no"

git config --global alias.bb "branch"

git config --global branch.autosetuprebase always

git config --global rerere.enabled true
```

Reference Log

git reflog - log of git changes

git reset --hard <commit-hash> - Recover a lost commit:

git reset --hard HEAD@{2} (go back 2 positions in reflog)

DEMO

Git filter repo

Remove file or string from entire git history of repo/branch

git filter-branch

git-filter-repo

Maintenance

git maintenance start / run

optimize git in LARGE repos

https://git-scm.com/docs/git-maintenance

Cheatsheet

https://education.github.com/git-cheat-sheet-education.pdf

https://www.youtube.com/watch?v=aolI_Rz0ZqY #Avansert

https://git-scm.com/docs/