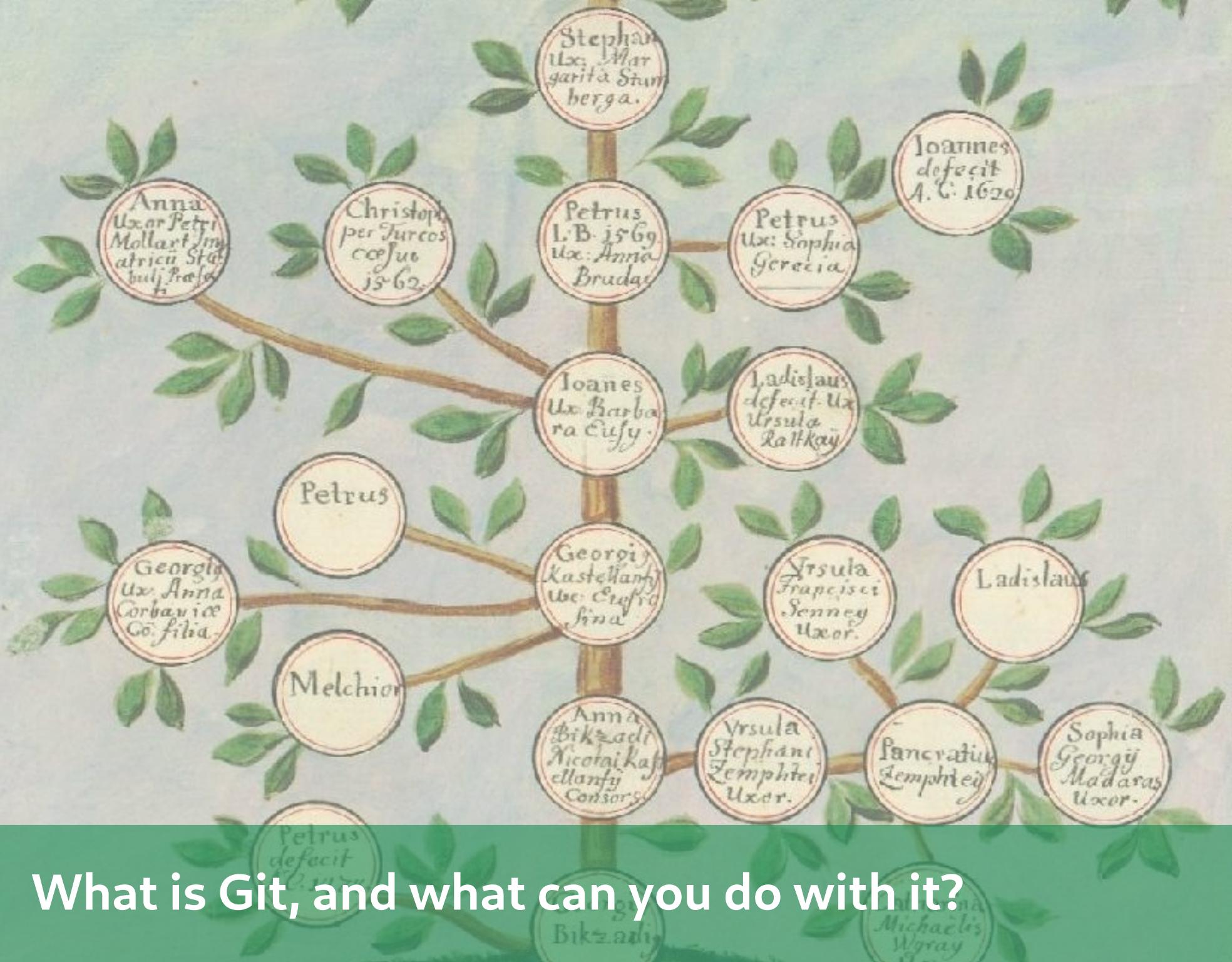


Git for Everybody

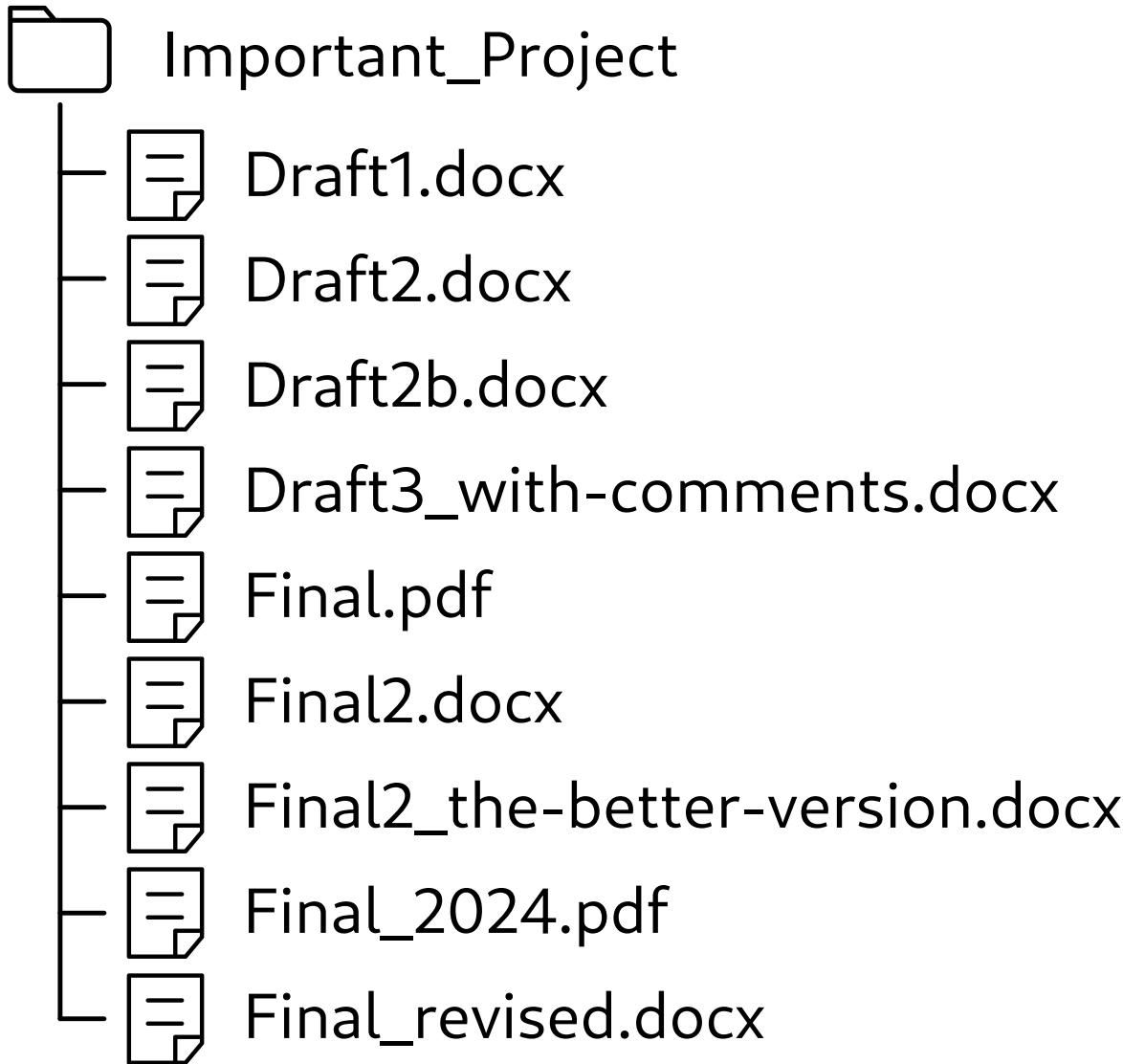
Intro to Git and GitHub
for Non-Coders

github.com/tiago-rorke/git-for-everybody



What is Git, and what can you do with it?

Version Control



Version Control



[Why opensource collaboration is complex]

Git

An open source project

Development started by
Linux creator Linus Torvalds
in 2005



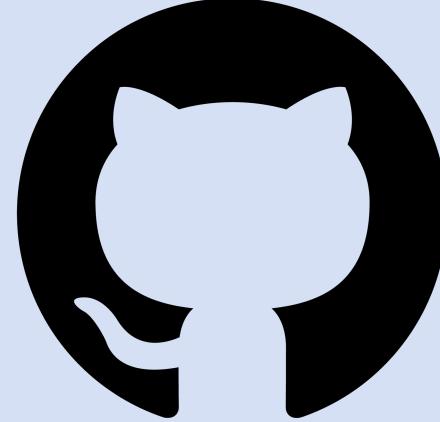
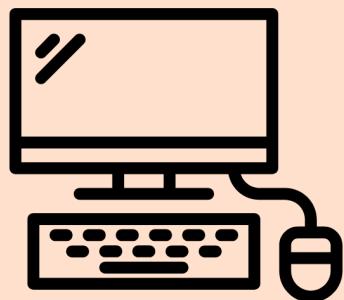
Distributed Version Control

[distributed version control diagram]



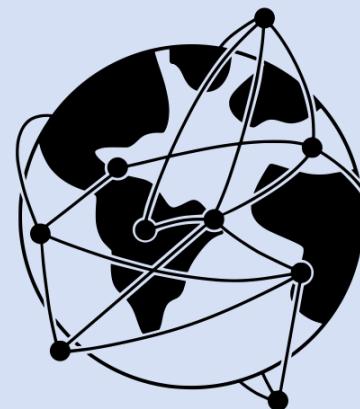
git

git-scm.com



GitHub

github.com



Other Git Hosting Platforms/Networks



GitLab



Gitea



Bitbucket

Forgejo



Git Is a Language

pull stash

commit

stage

push

diff

branch

fetch

repo

rebase

add

fork

revert

remote

clone

checkout

merge

HEAD

origin

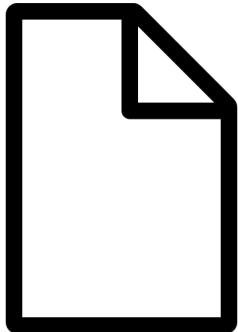
[What "should" you put in Git repo?
What is the point?
When is it truly useful?]

[how I treat a git repo
to help organise my project]

[What files work well in Git and what doesn't
Practical file size limits]

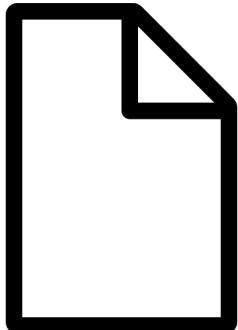
Binary vs Text

Git efficiently tracks changes in text files, less so for binary files



- .md
- .txt
- .html
- .svg

```
<?xml encoding="UTF-8" ?>
<svg
    width="105mm"
    height="148mm"
```



- .png
- .pdf
- .docx
- .mp4

```
01101000 01100001 01110110
01100101 00100000 01100001
01101110 00100000 11110000
10011111 10100101 10011010
```

Markdown

Markdown is a markup language

[what is markdown]

[aaron swartz]

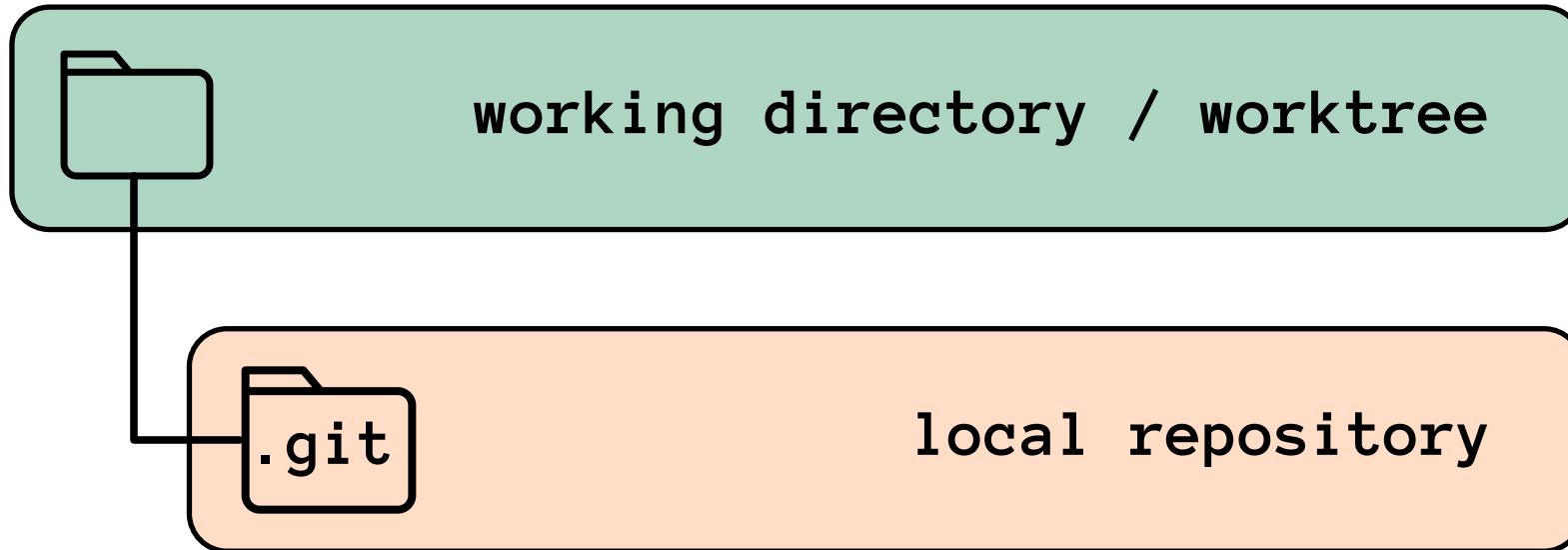
[john gruber]



How do you use Git?

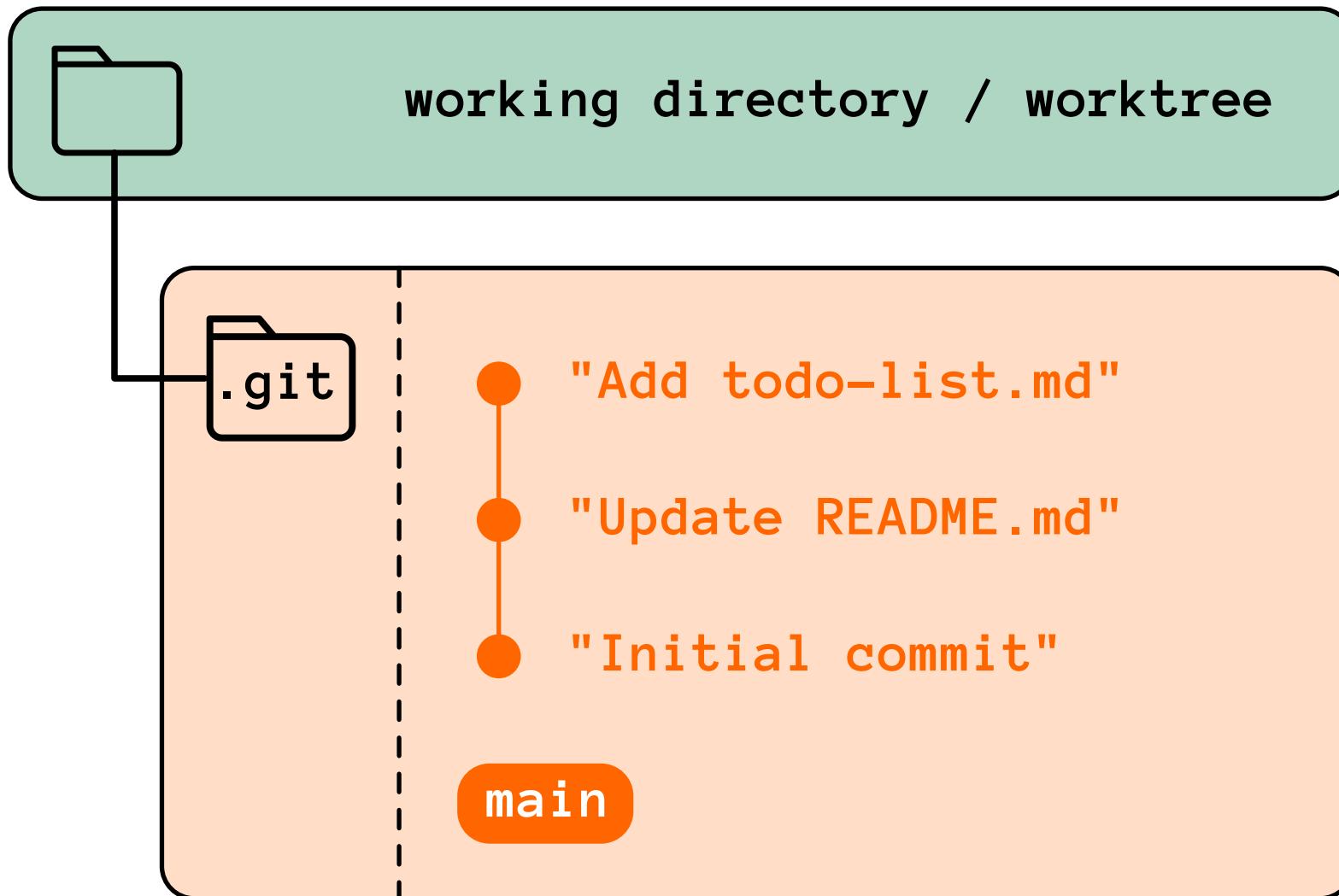
Init will turn a folder into a local git repository, creating a hidden **.git** folder

git init



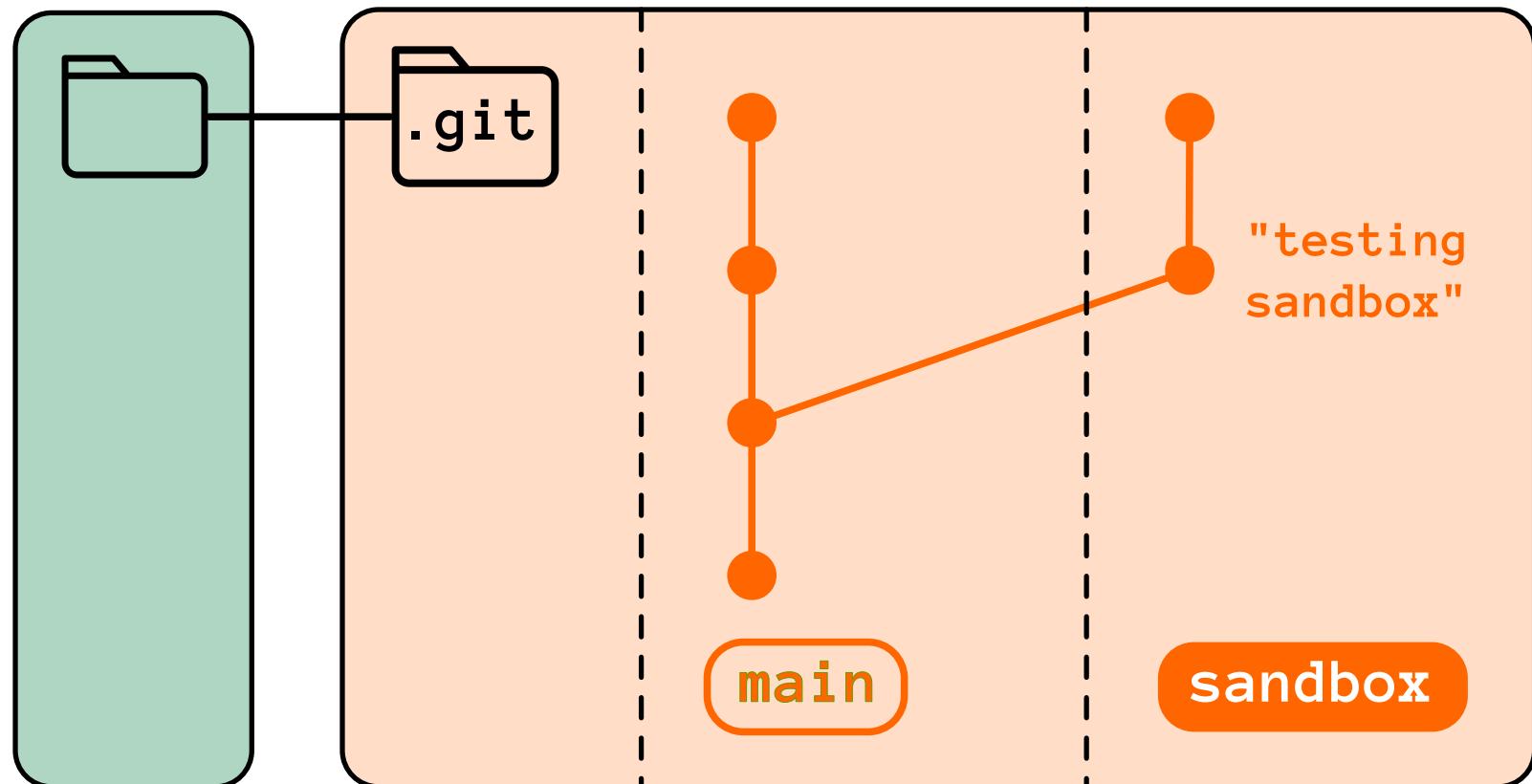
Commits are like save points,
that store versions of your files

git commit



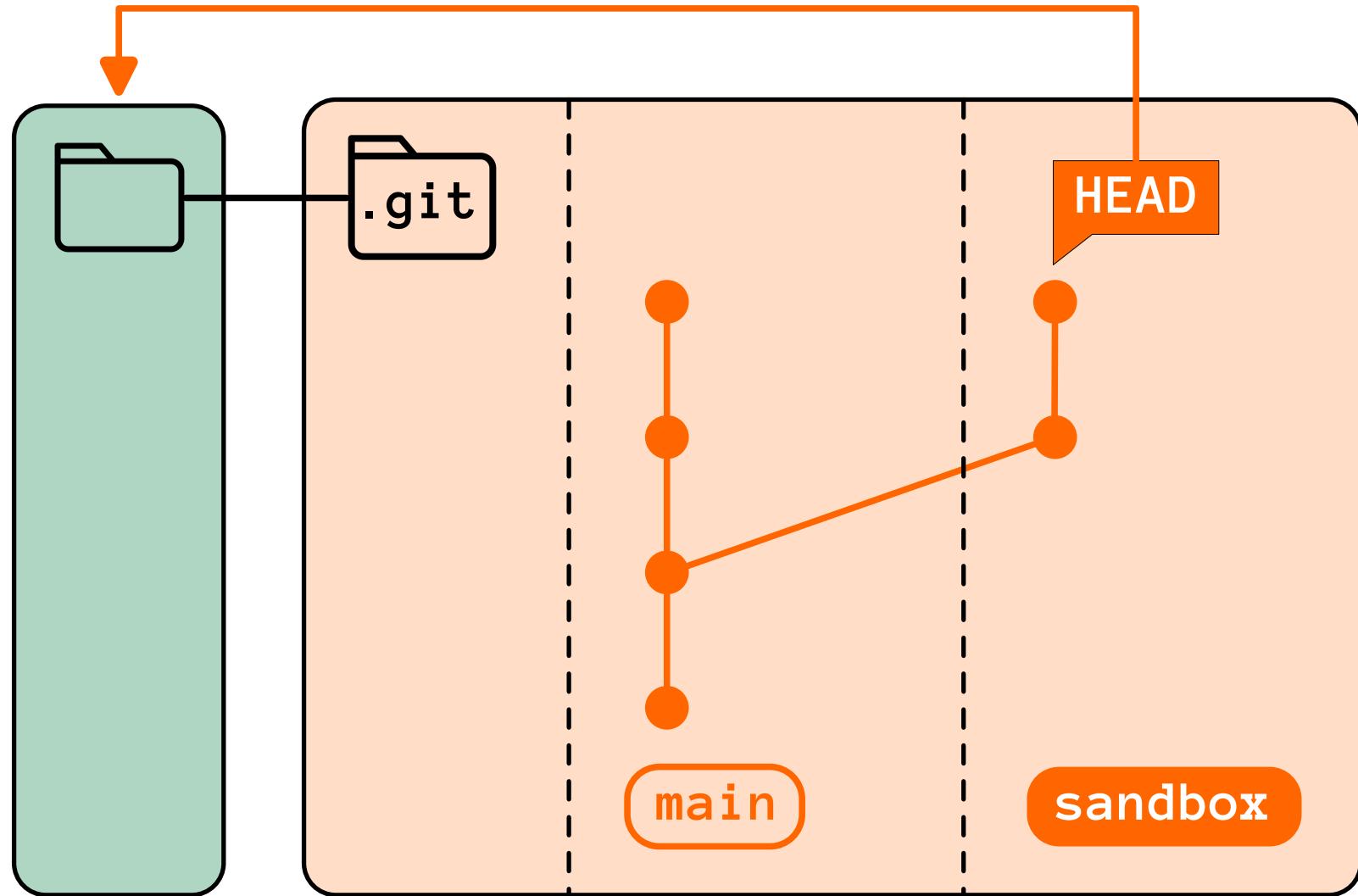
A **Branch** is an alternate version of your files which you can work on separately from the originals

git branch



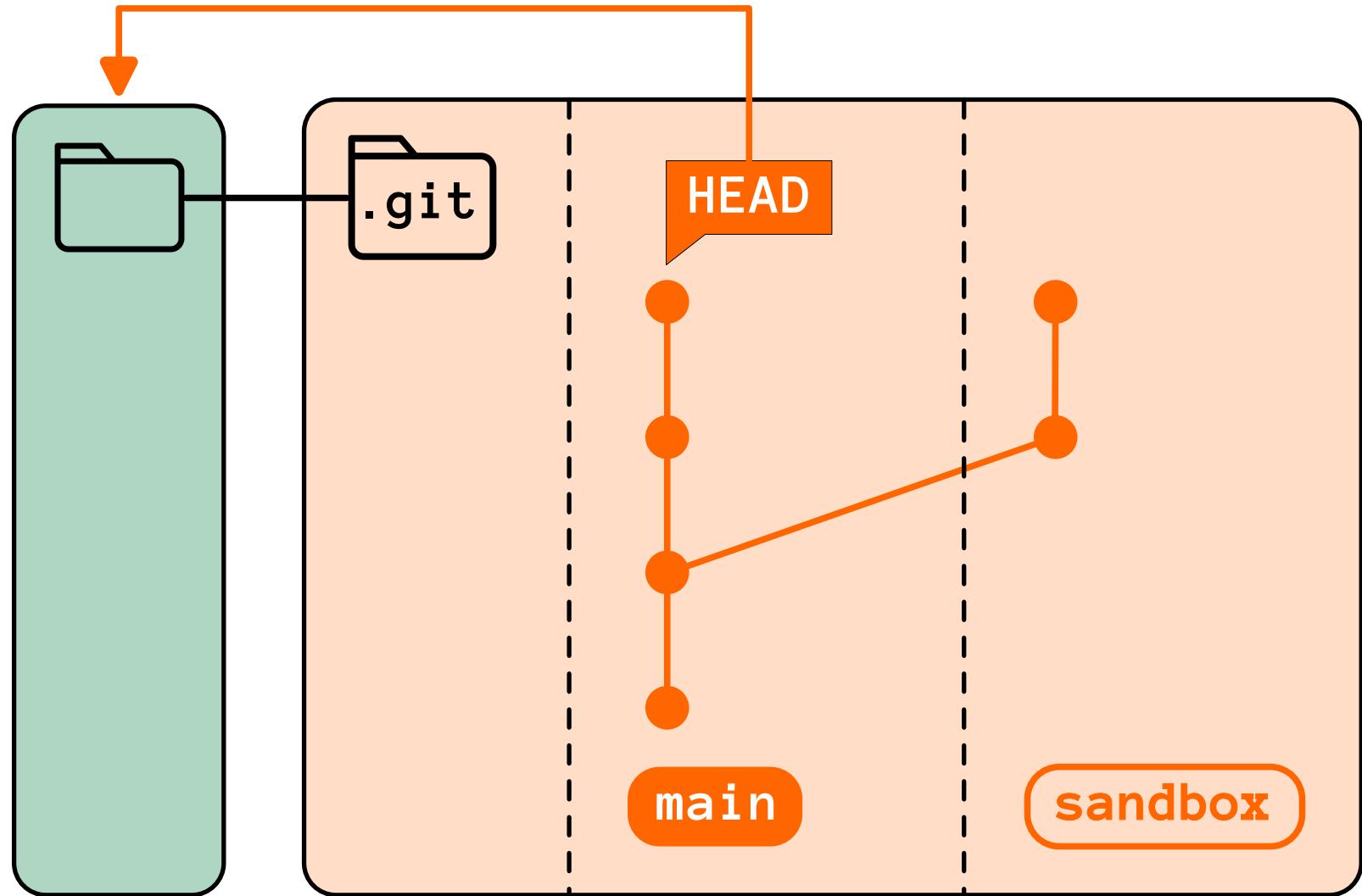
Checkout is used to switch between branches

git checkout



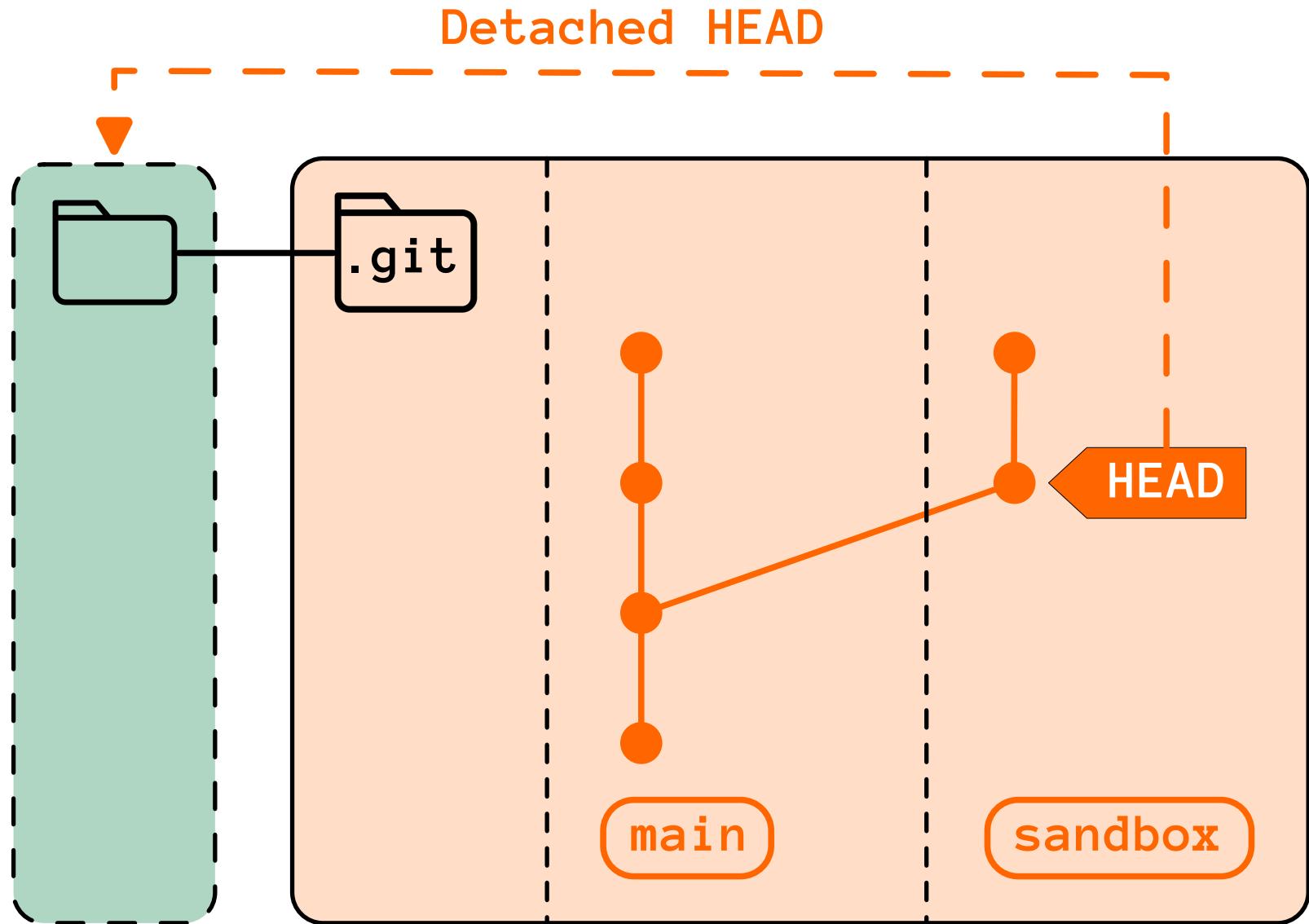
Checkout is used to switch between branches

git checkout



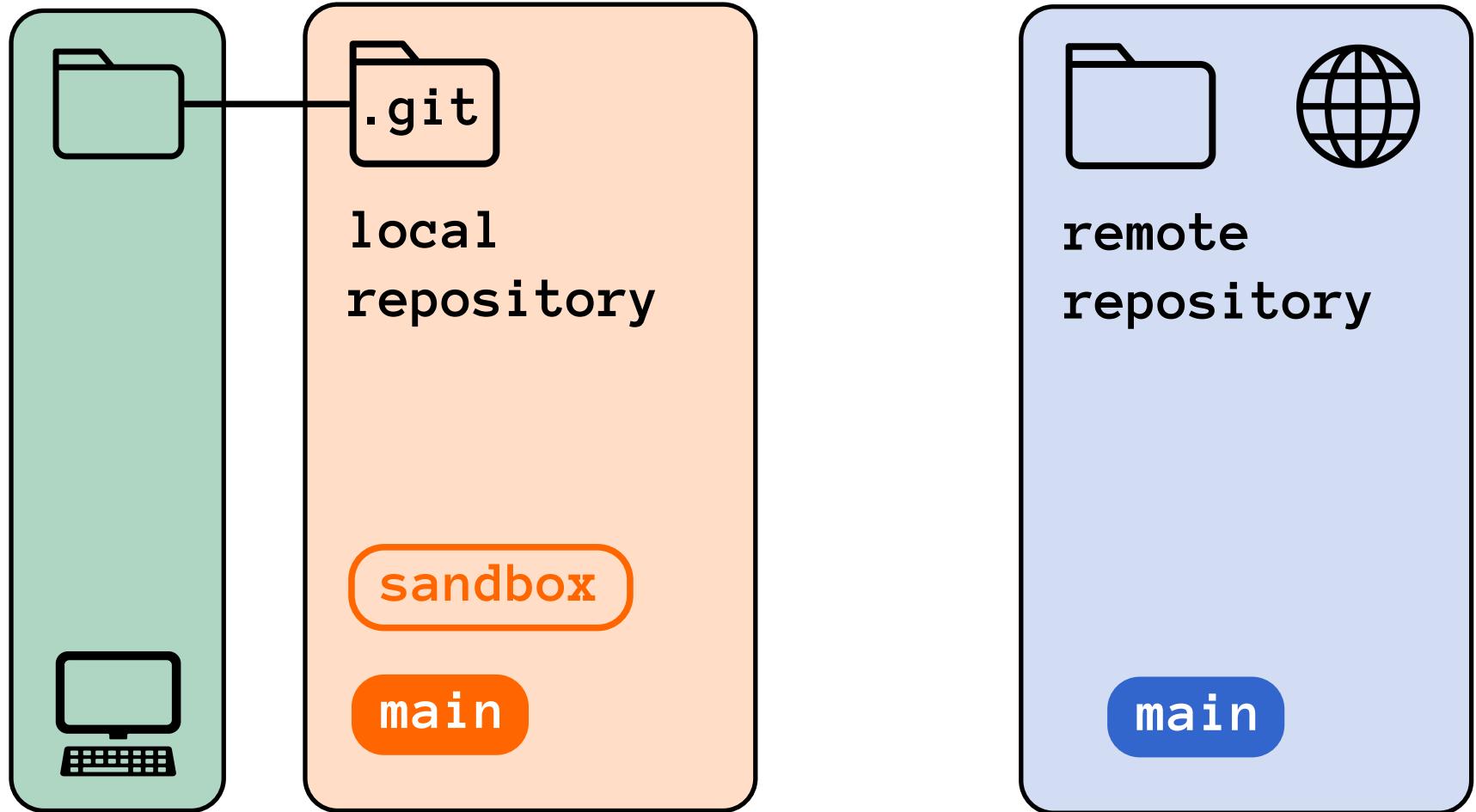
git checkout

If you checkout a previous commit
in the history, it will "detach" the **HEAD**



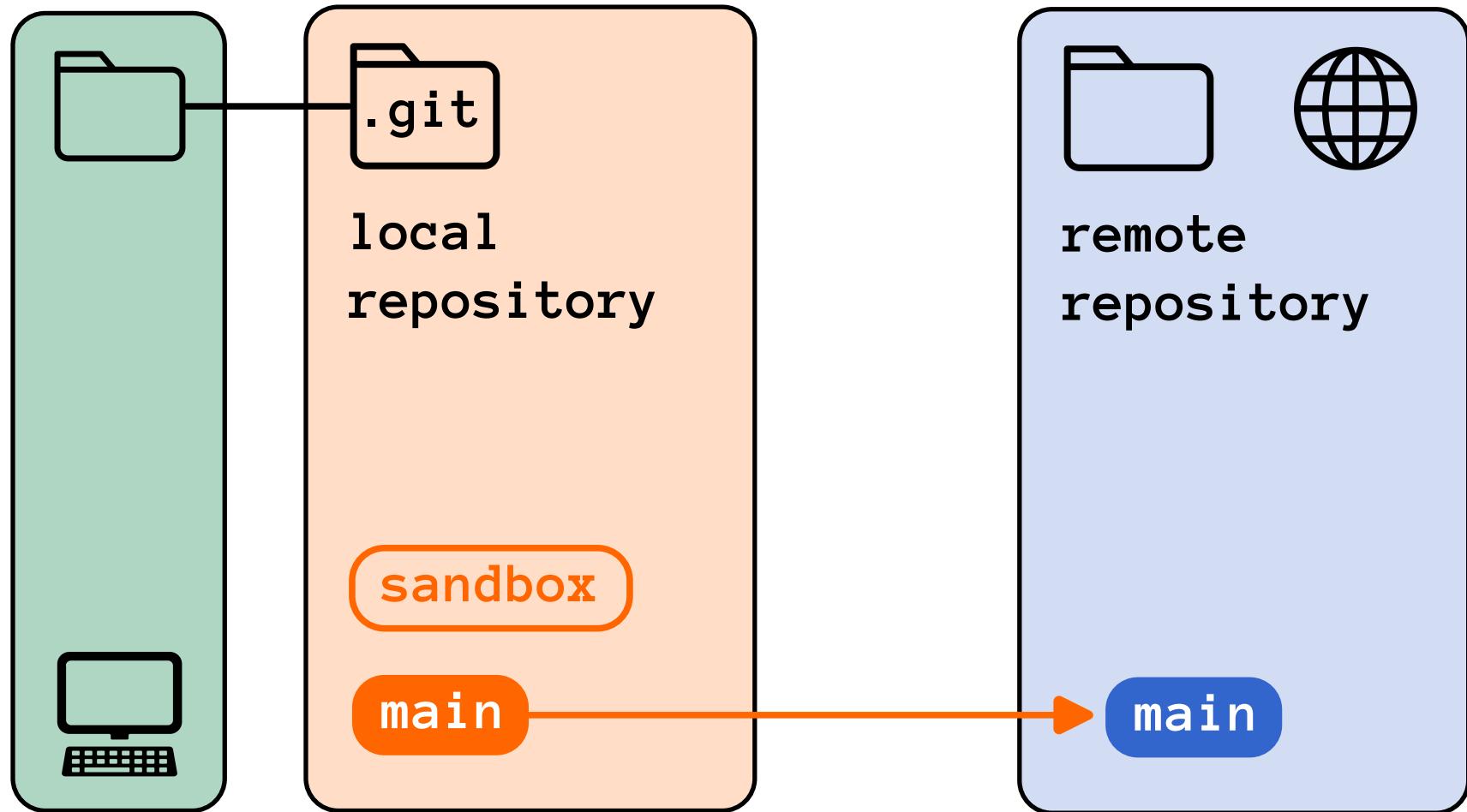
A **Remote** is a copy of the repository which is somewhere else, like on an online git hosting platform

git remote



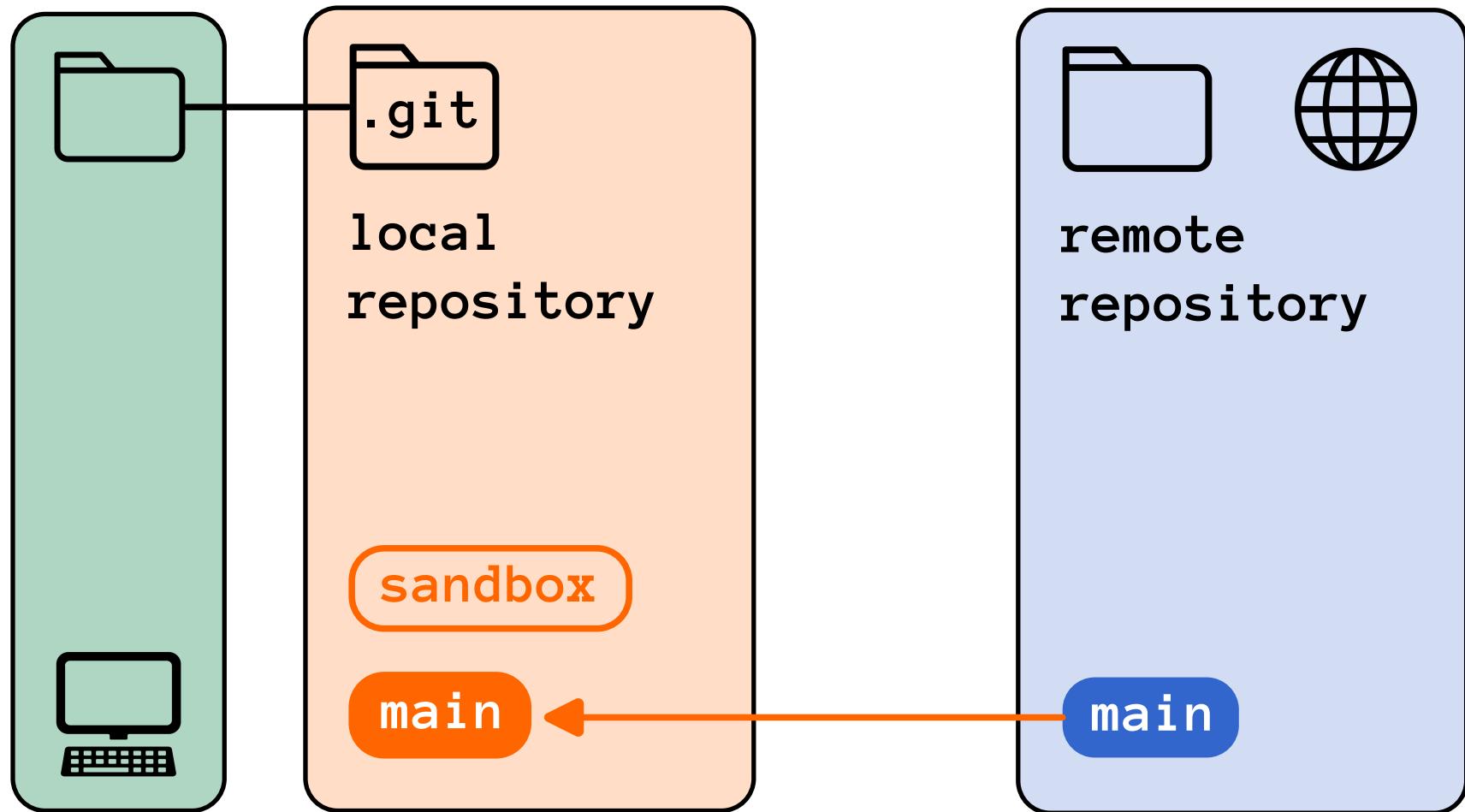
Push will copy new commits
from your **local** repository
to a **remote** repository
(usually) only for your current branch

git push



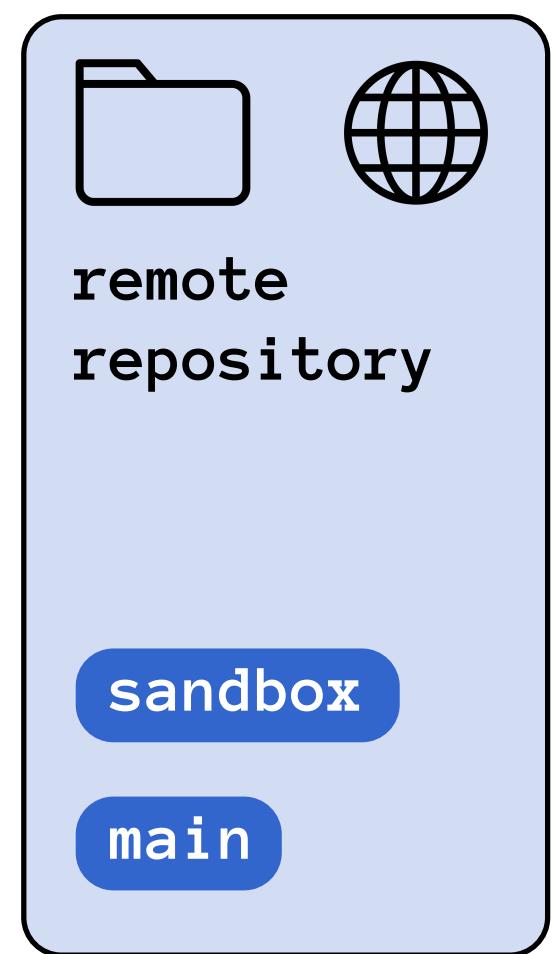
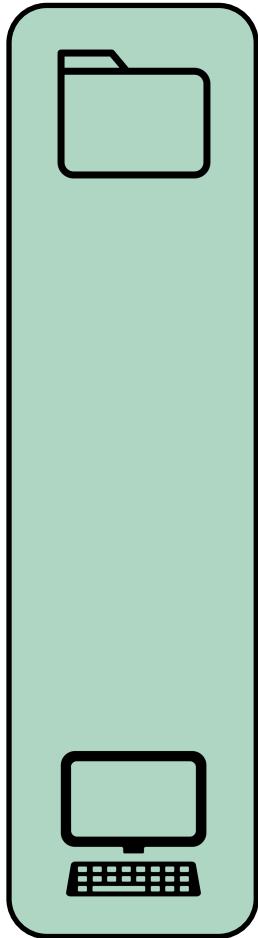
Pull will copy new commits
from the remote repository
to your local repository
(usually) only for your current branch

git pull



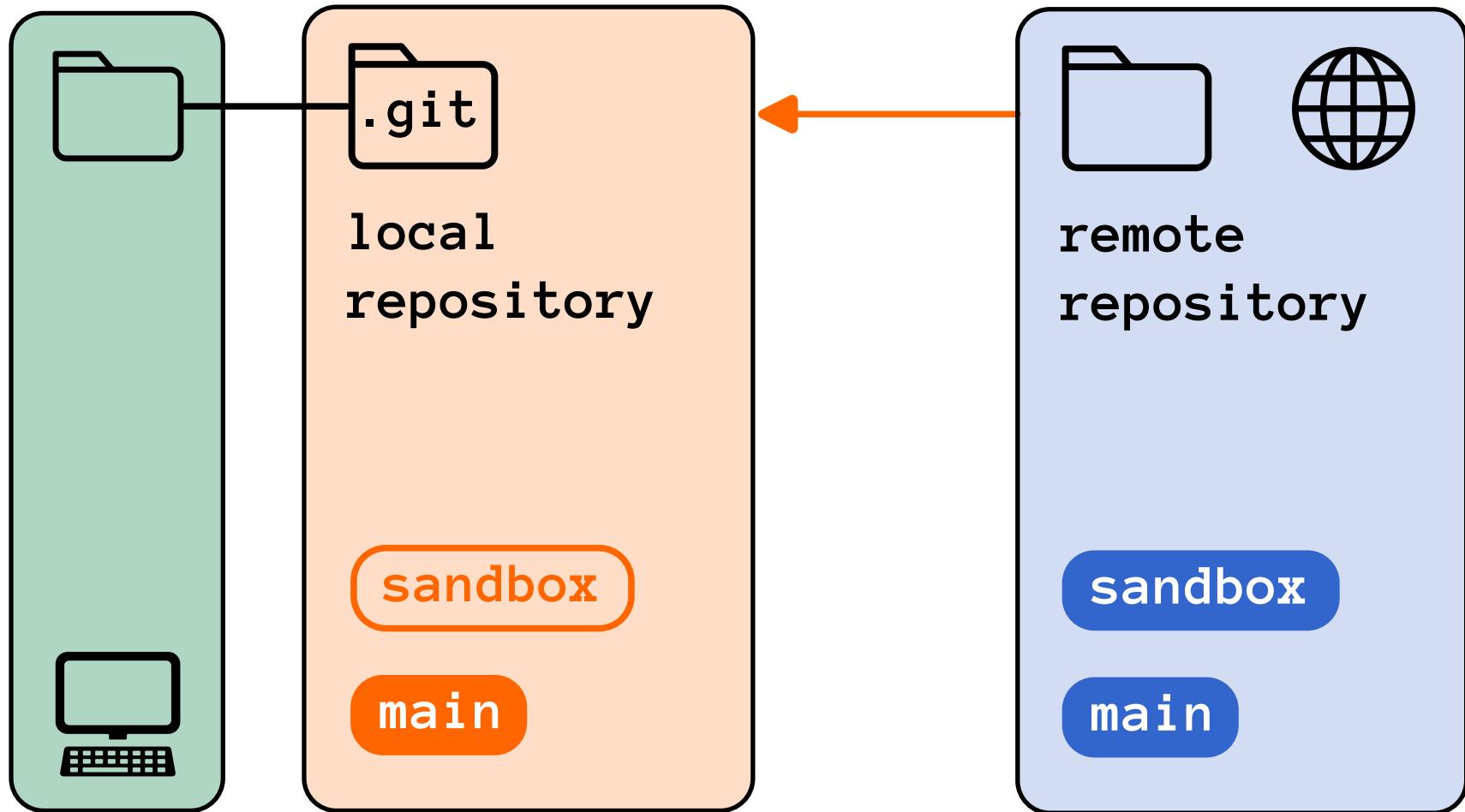
Clone will make a new local copy
of a remote repository

git clone



Clone will make a new local copy
of a remote repository

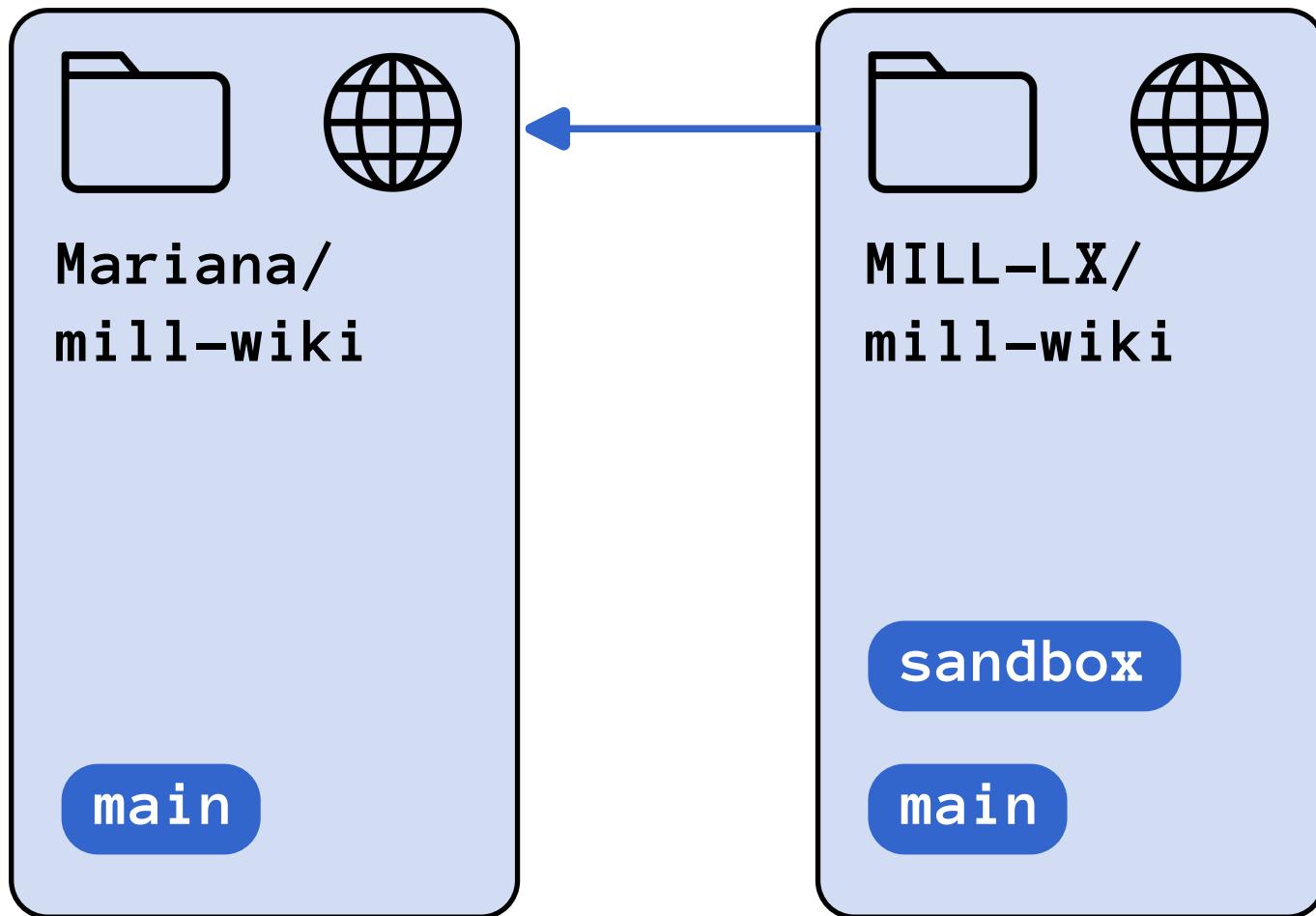
git clone



Fork will copy a remote repository
from somebody else's account into your account

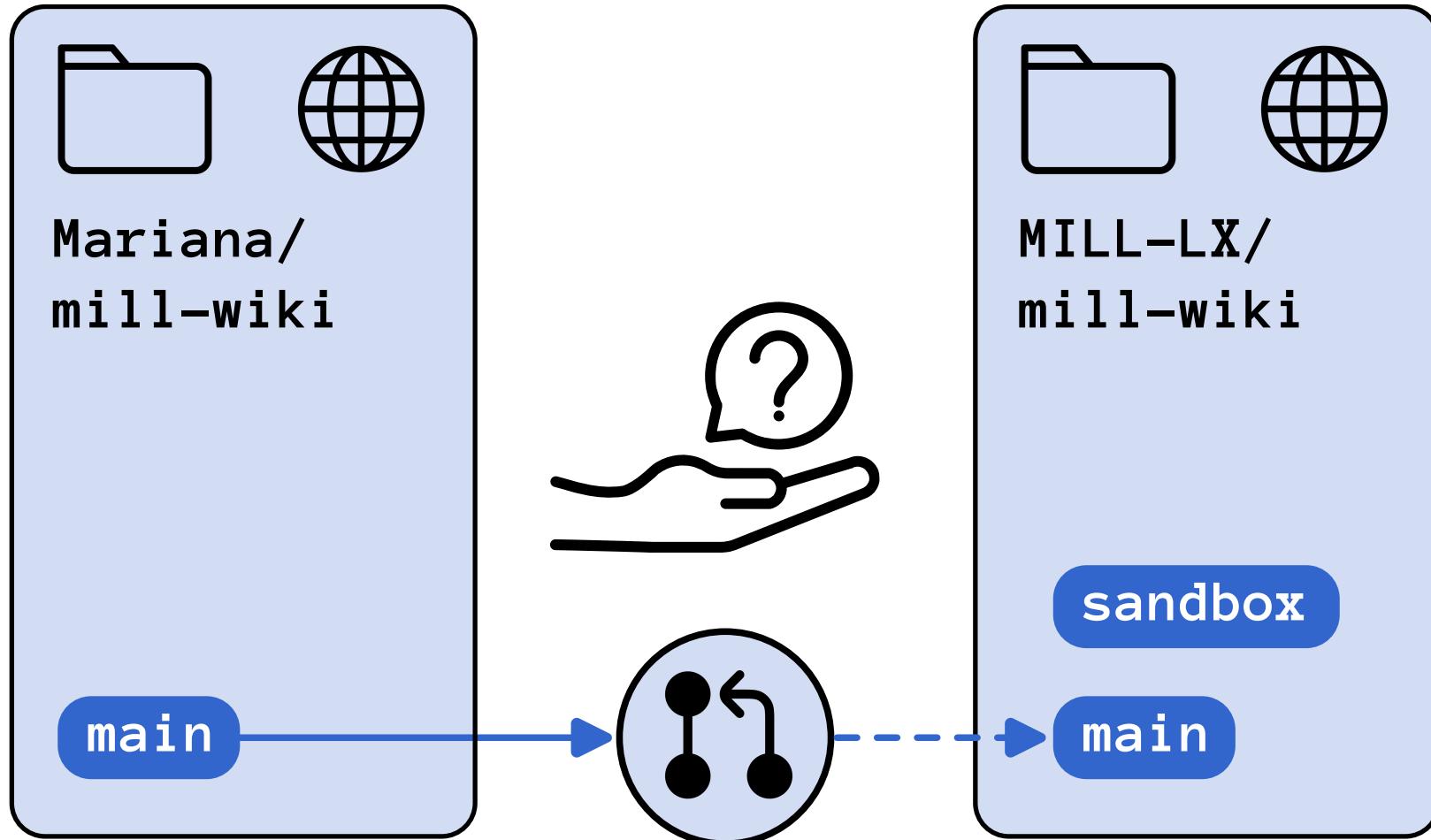
fork

You can choose to copy only the default branch

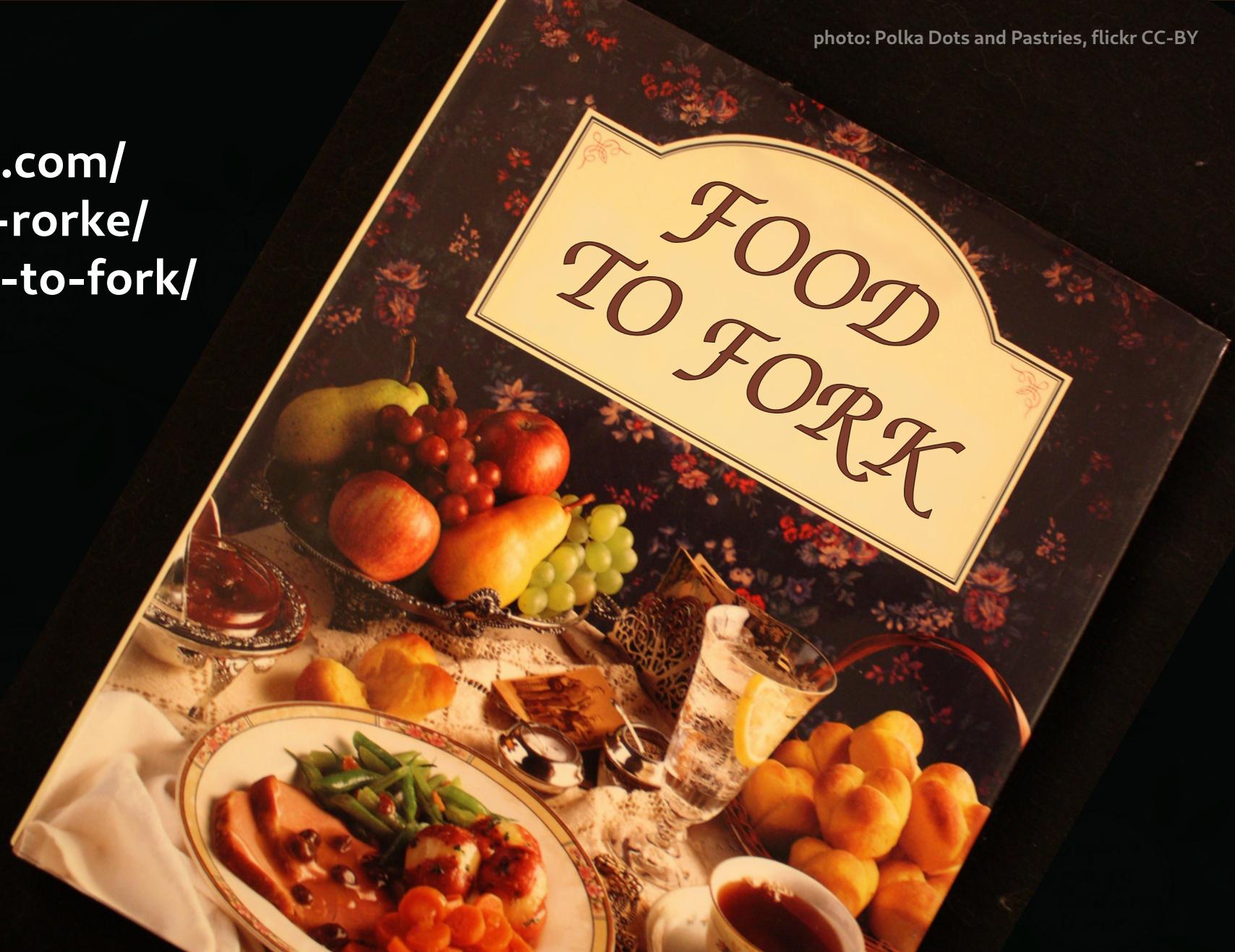


A **Pull Request** lets you propose changes from your remote repository to the original you created a fork from

pull request



[github.com/
tiago-rorke/
food-to-fork/](https://github.com/tiago-rorke/food-to-fork/)



Try it out, by sharing a recipe

merge / fetch



GitHub Pages

Make your own recipe webpage

[thank you]

[further resources]

Now you know how to use Git!