

[JF Martin - Unsplash](#)

git (intro)

Adapted from materials from Dr. Carrier

Imagine...

I ask you to do a group coding project

Imagine...

I ask you to do a group coding project

How do you manage the code?

Imagine...

I ask you to do a group coding project

How do you manage the code?



Imagine...

I ask you to do a group coding project

How do you manage the code?



Issues with these techniques



Issues with these techniques

- How do you merge?
- Multiple editors at once?
- History of files?
- How to integrate into workflow?
- Lose the flash drive? Oops!



[Vector Portal](#)

Standard solution: `git`

Standard solution: `git`



Standard solution: `git`

- Version control software



Standard solution: `git`

- Version control software
- Free and [open source](#)



Version control systems

Software that tracks all changes to a “repository” of files

Version control systems

Software that tracks all changes to a “repository” of files

Benefits:

Version control systems

Software that tracks all changes to a “repository” of files

Benefits:

- Full history
 - Rollbacks, traceability

Version control systems

Software that tracks all changes to a “repository” of files

Benefits:

- Full history
 - Rollbacks, traceability
- Working concurrently

Version control systems

Software that tracks all changes to a “repository” of files

Benefits:

- Full history
 - Rollbacks, traceability
- Working concurrently
- Merging

Version control systems

Software that tracks all changes to a “repository” of files

Benefits:

- Full history
 - Rollbacks, traceability
- Working concurrently
- Merging
- No files ever fully deleted

Distributed version control

Distributed version control

- Each user has a local copy of *all* files, including full history

Distributed version control

- Each user has a local copy of *all* files, including full history
- Allows for working offline

Distributed version control

- Each user has a local copy of *all* files, including full history
- Allows for working offline
- *Should* make merging easier

Some common version control systems

Distributed



mercurial

Centralized



PERFORCE

git vs GitHub



git



git vs GitHub



- Software
- Handles the version control
- Operates locally

git vs GitHub



- Software
- Handles the version control
- Operates locally
- Website / server
- Stores git repositories
- Additional features
 - Public sites
 - GitHub actions

git vs GitHub



- Software
- Handles the version control
- Operates locally
- Website / server
- Stores git repositories
- Additional features
 - Public sites
 - GitHub actions

Both are free!

A few notes

- git has *many* moving parts
- You *will* mess up
 - I still do
 - Most mistakes are easily recoverable
- Don't worry about all the details
 - Feel comfortable with the big picture
 - Look things up when you need to know / get curious!

What we're building toward...

Working
Directory

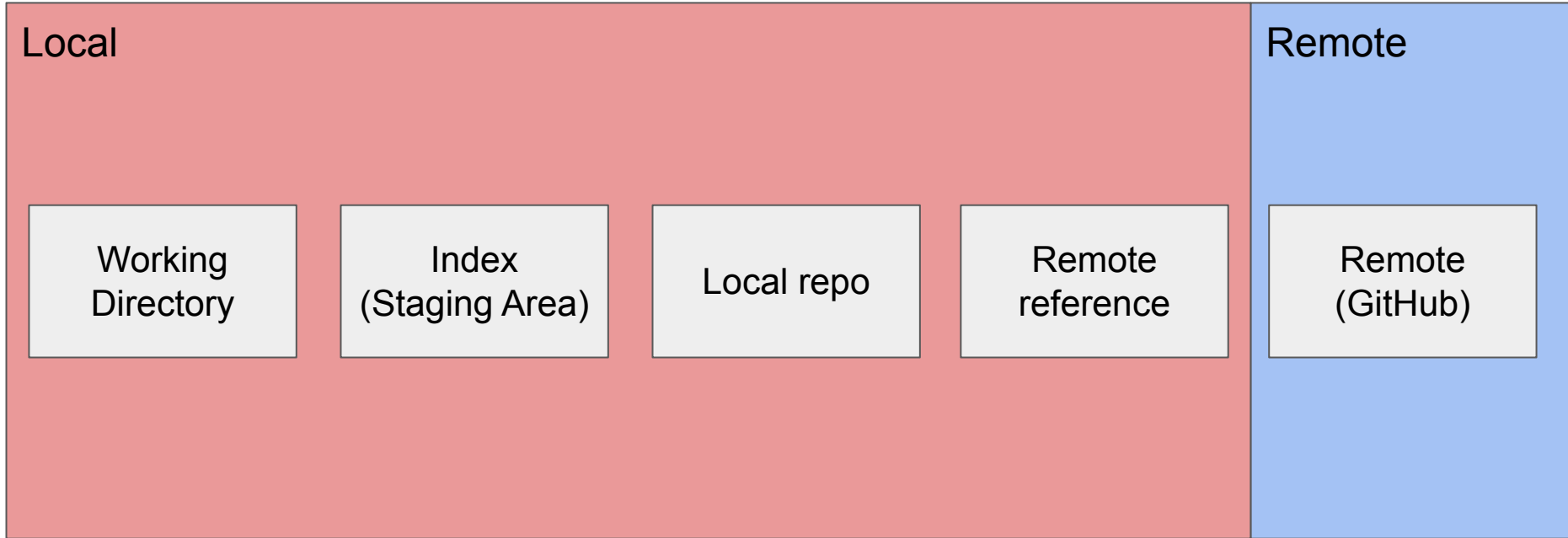
Index
(Staging Area)

Local repo

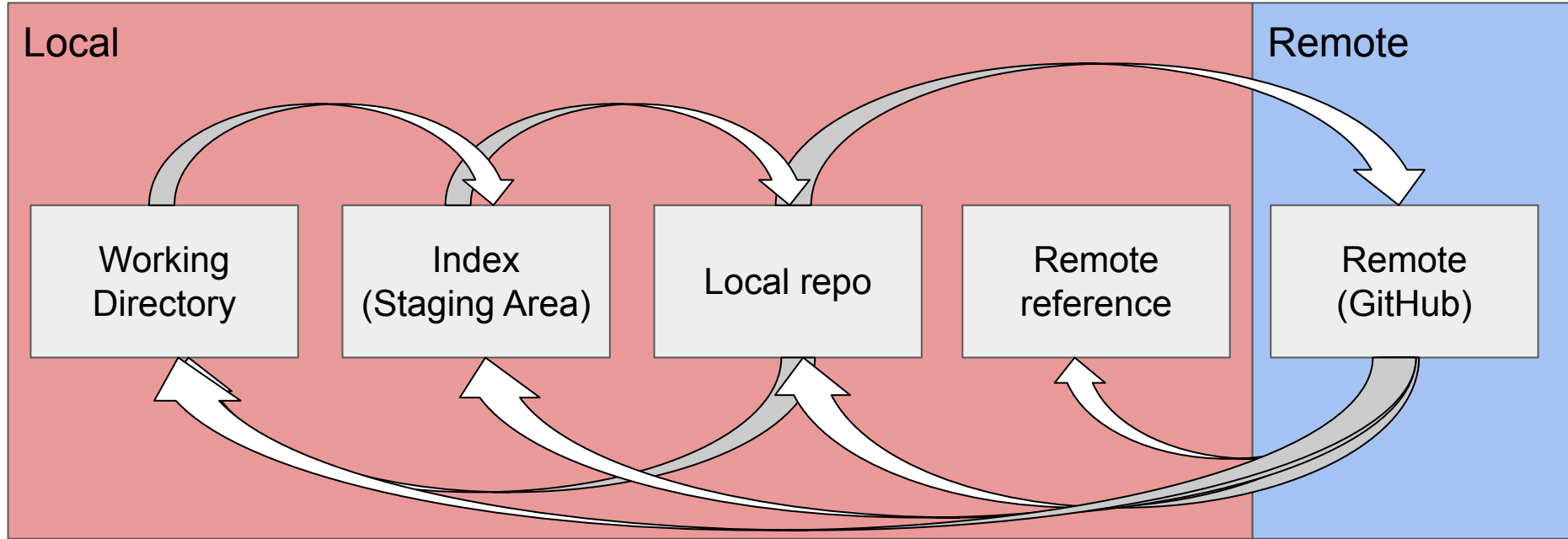
Remote
reference

Remote
(GitHub)

What we're building toward...



What we're building toward...



What I want you to do

- Install git
 - Likely already installed on Linux, Mac, and WSL
 - <https://github.com/git-guides/install-git>
 - We're *NOT* using GitHub desktop
 - To test, run `git version` (Should be 2. something)
- Create a GitHub account (I recommend [GitHub Education](#), too)
 - I will ask for it soon
- Configure git

```
git config --global user.name "Your Name"
```

```
git config --global user.email name@example.com
```