

Intro to GitLab

DevOps on a Shell Script Budget

Mac Justice

**Sr. Support Specialist
Synapse Product Development**

SYNAPSE



git

GI+olite

Git Host Wish List

- Web GUI
- Issue tracking
- Inexpensive

Git Host Wish List

- Web GUI
- Issue tracking
- Inexpensive



GitLab

GitLab Administration

GitLab Administration

Self Hosted Tiers

Edition	Price	Support	Features
Community	Free	Nope	All major functionality
Enterprise Starter	\$40/user/year	Next Day	More permissions controls
Enterprise Premium	\$200/user/year	4 Hour	High Availability, other advanced features

GitLab Administration

Easy Set Up

- "Omnibus" package for Linux
- Docker Container
- Pre-built VMs (Amazon EC2/LightSail, Digital Ocean)

GitLab Administration

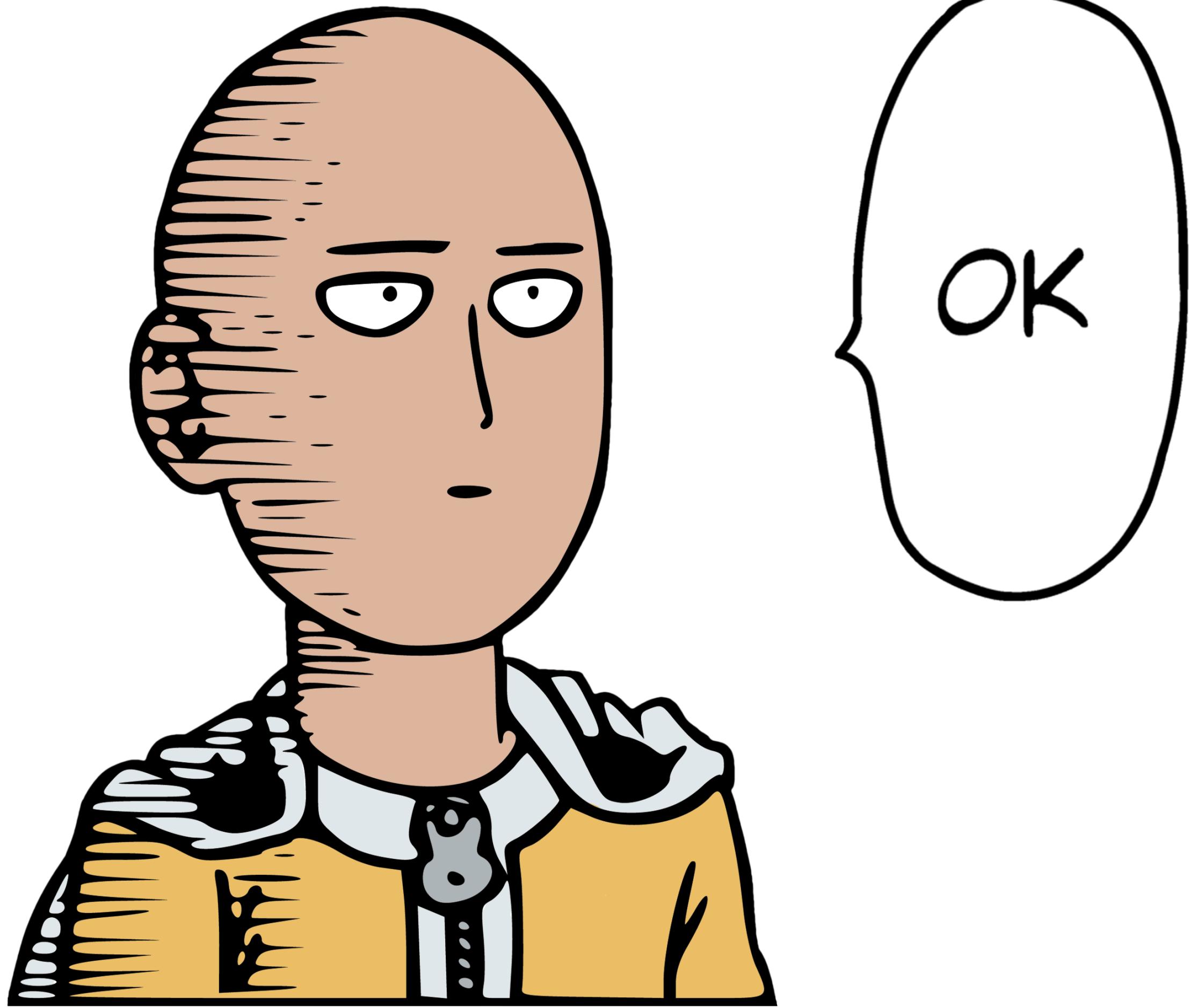
Authentication & Authorization

- Built-in
- LDAP
- OAuth & SAML

GitLab Administration

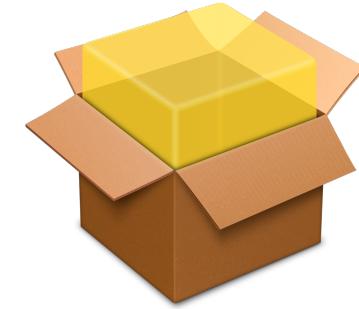
Integrations



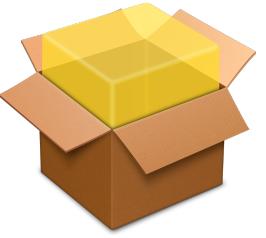


Git LFS

Git +

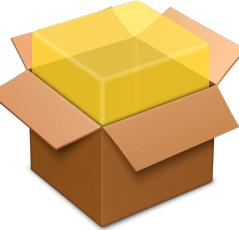


Git LFS +



=



Git LFS +   **=** 
(LFS = Large File Storage)

Git LFS Example

Local Installation

```
brew install git-lfs  
git lfs install
```

Git LFS Example

Repo Setup

```
cd munki-repo  
git lfs track "*.pkg"  
git add .gitattributes  
git commit -m "Added LFS tracking for PKGs"
```

Git LFS Example

There's no step 3!

```
git add pkgs/SweetApp.pkg  
git commit -m "Added SweetApp"  
git push origin master
```

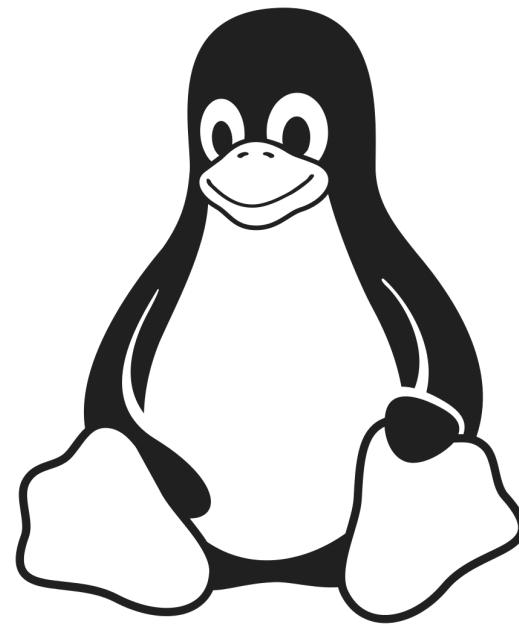
GitLab CI

What is CI?

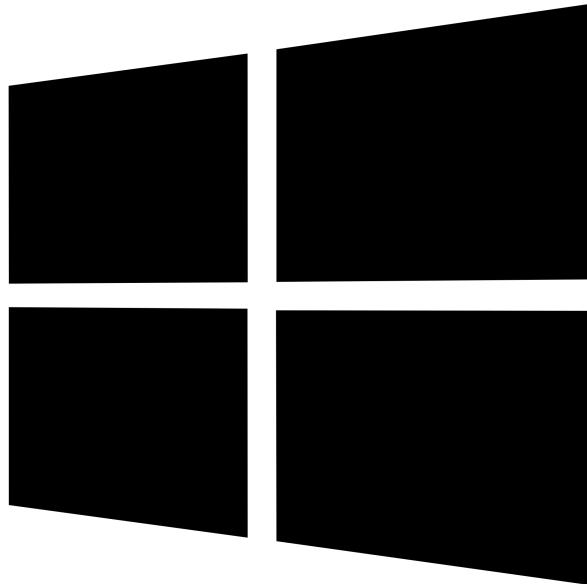
Continuous Integration

1. Push code to GitLab  
2. Do something with that code  
3. Report results  

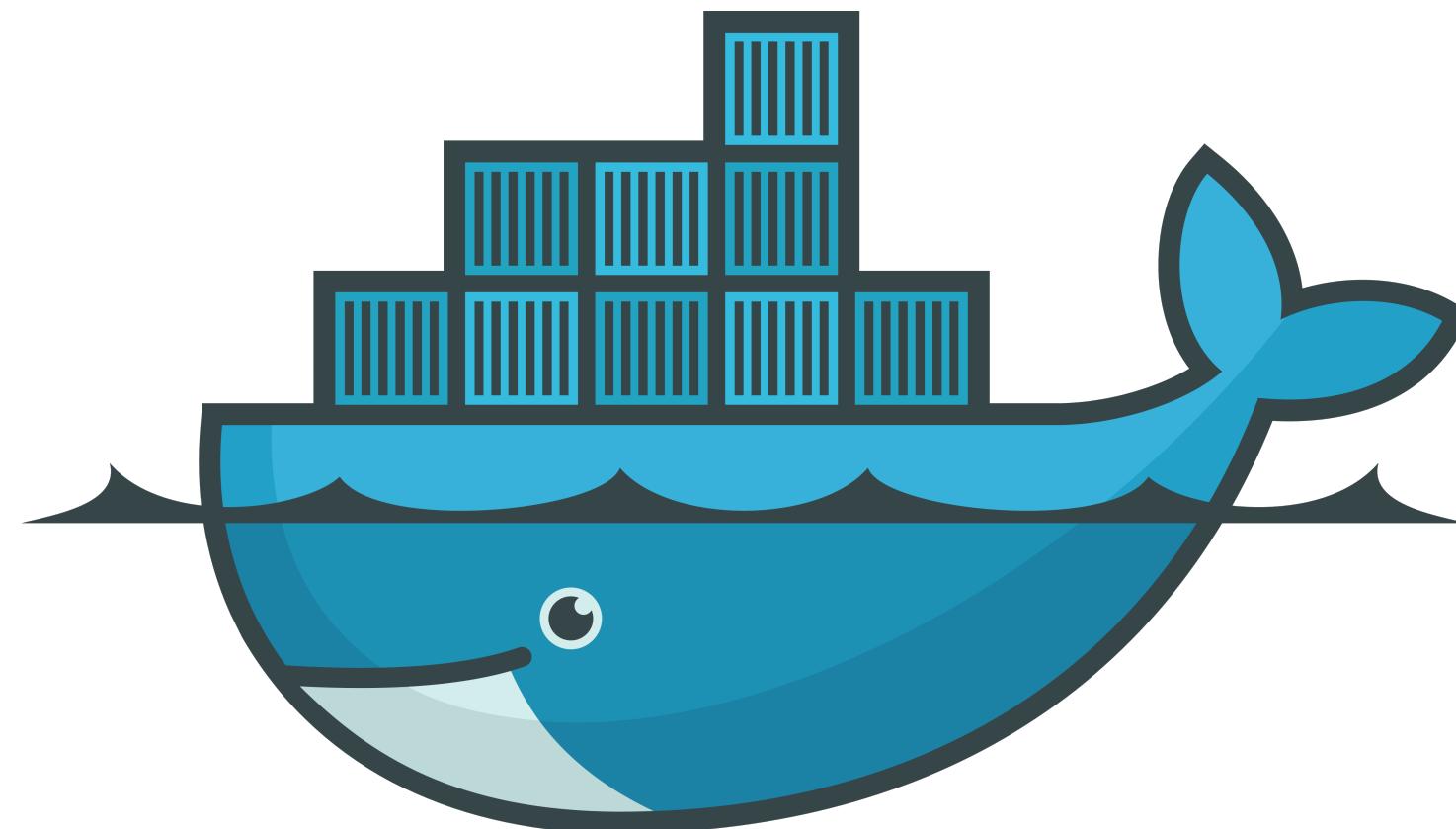
CI Runners



macOS



CI Runners



docker

ci Autoscaling Mode

CI Configuration

.gitlab-ci.yml

```
validate:
  stage: test
  script: lint_roller.sh

roll_out:
  stage: deploy
  only: master
  script:
    - ./sound_klaxons_and_flash_lights.py
    - rsync build/* user@remote_server:/deploy/path/
```

.gitlab-ci.yml, annotated

```
validate: # first job name
  stage: test # All 'test' stage jobs run before 'deploy' stage
  script: lint_roller.sh # Run this script

deploy: # second job name
  stage: deploy # Start only when all 'test' stage jobs complete
  only: master # Only run on Master branch
  script:
    - ./sound_klaxons_and_flash_lights.py # Call script in repo
    - rsync build/* user@remote_server:/deploy/path/ # inline command
```

So what about Macs?

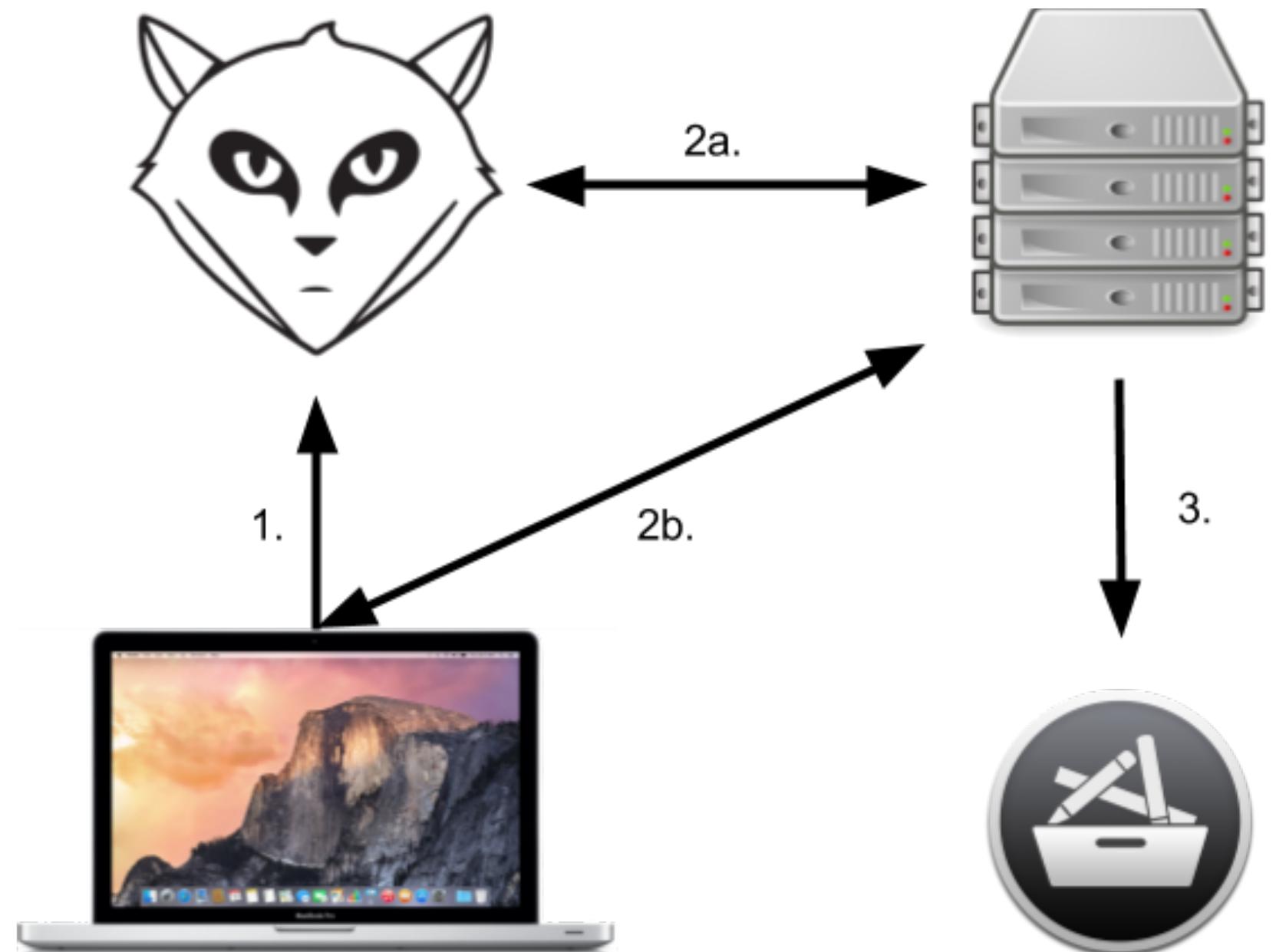


Munki in Git

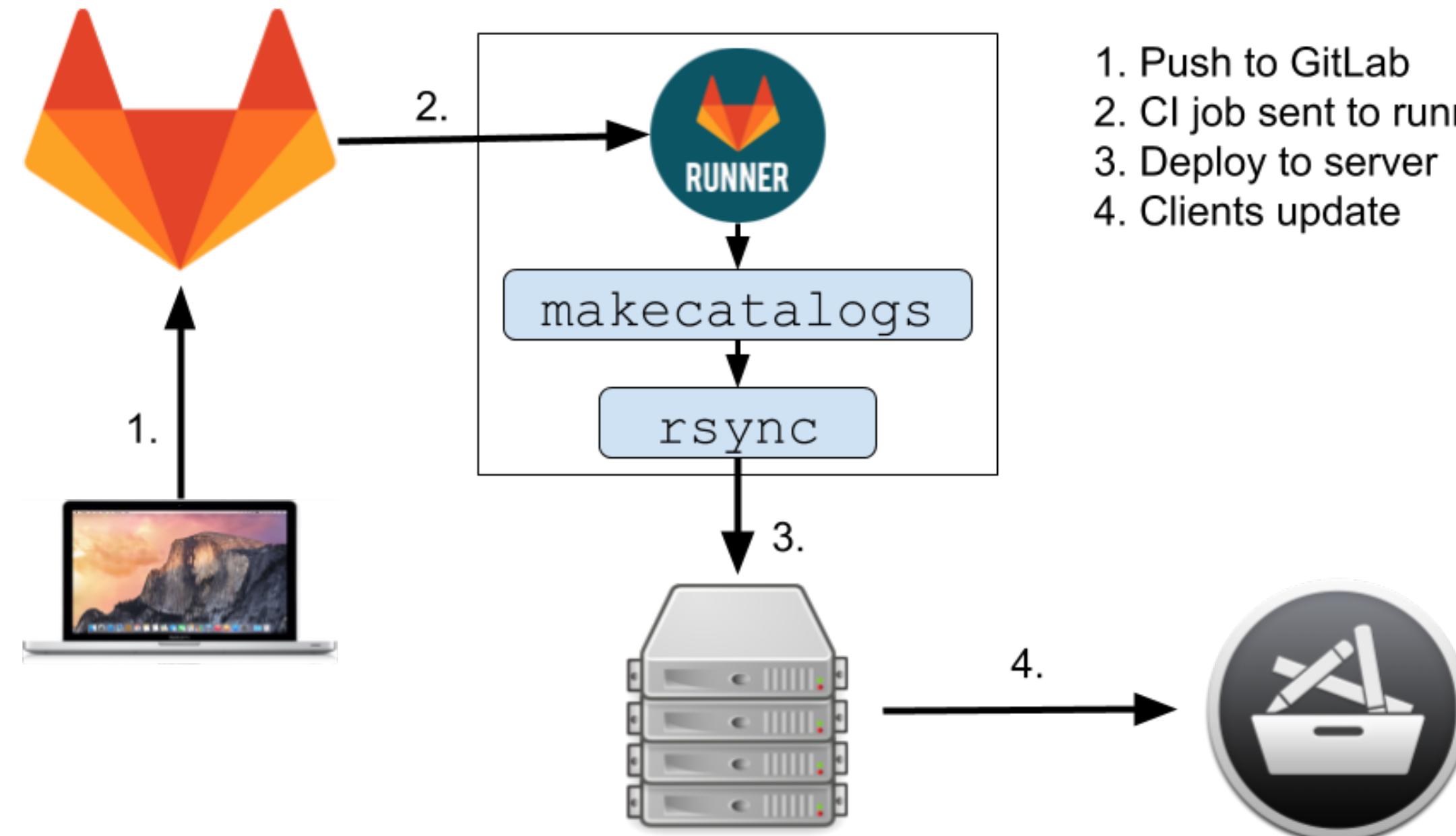
- History
- Collaboration
- Experimentation



The Old Way (No LFS, No CI)



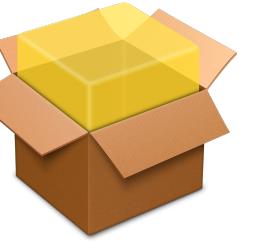
The New Way



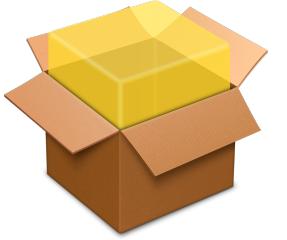




CONFIG

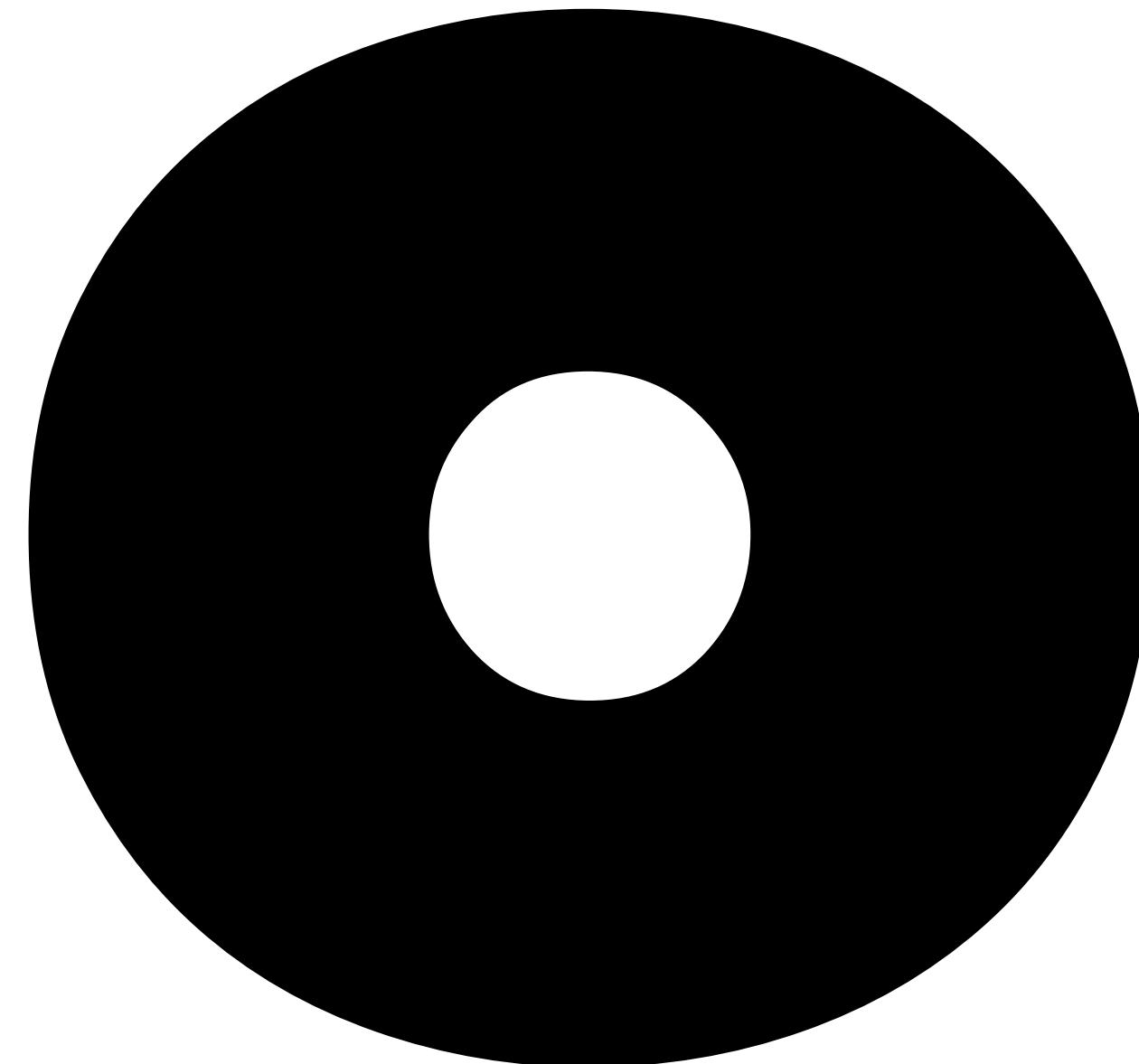
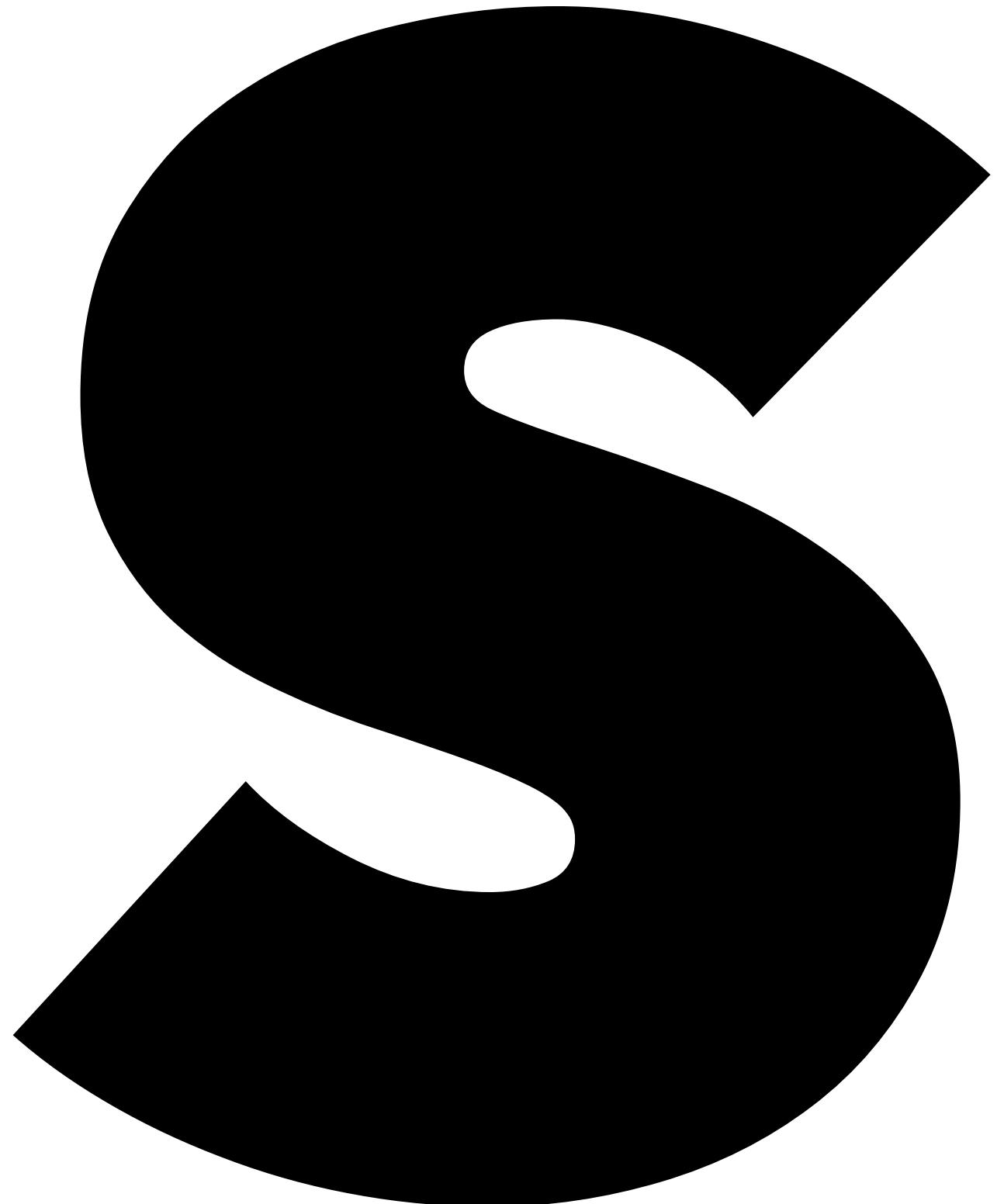


?



?





Automatic

Accountable

Collaborative

More Information

- This presentation: gitlab.com/macjustice/gitlab-presentation
- [GitLab](#)
- [GitLab CI](#)
- [Git LFS](#)
- MacDevOps:YVR 2016
 - [Tim Sutton on Jenkins CI](#)
 - [Wade Robson on Munki & S3](#)

Thanks!

Twitter: @macjustice
MacAdmins Slack: macjustice
Wherever: macjustice

Q & A