

# (Ab)Using GitLab CI

A Poor Man's GitHub Actions

Trix Taiclet  
RIPE NCC  
05 Nov 2025

# **What is CI?**

(enterprise university edition)

Yes, these are real slides.

# Continuous integration, delivery, and deployment



I hope so...

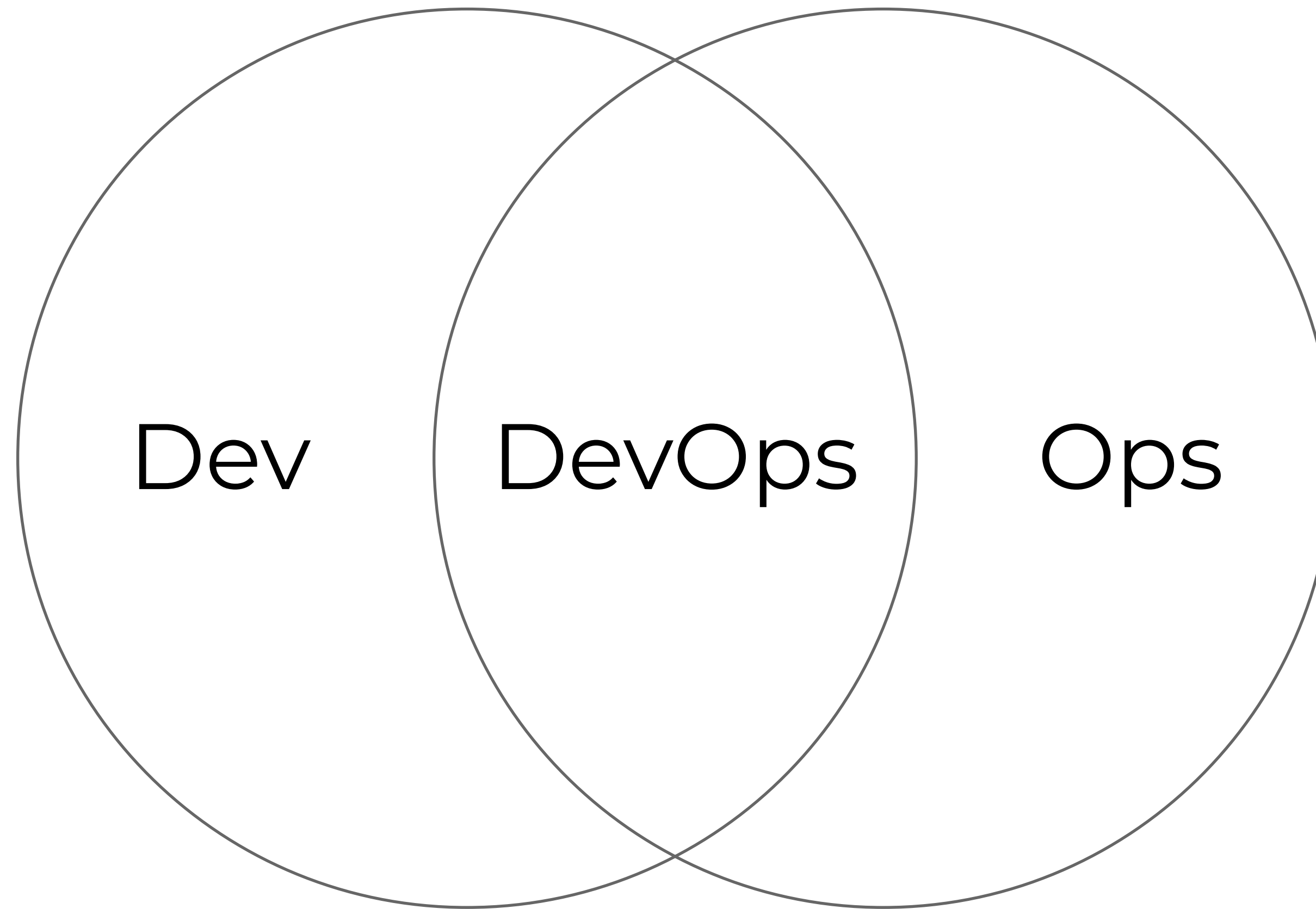
Which one is it?

**Nobody knows!**

# DevOps

---

**Thank you for this insight...**





# My reaction



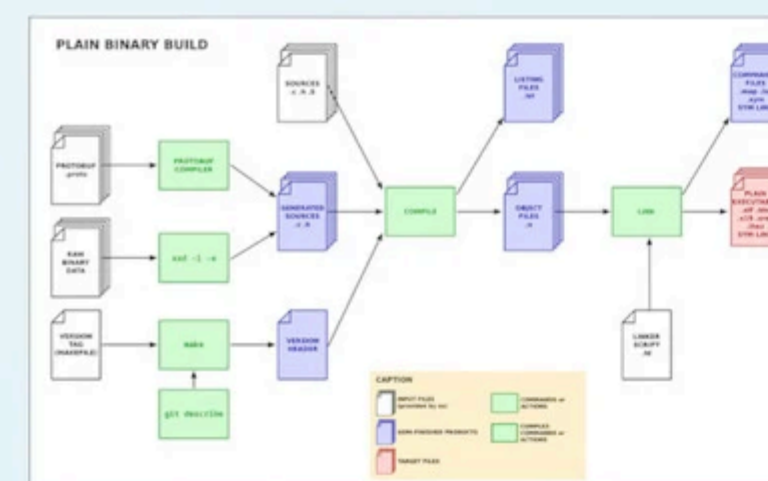
This is meaningless...

# STOP DOING BUILD SYSTEMS

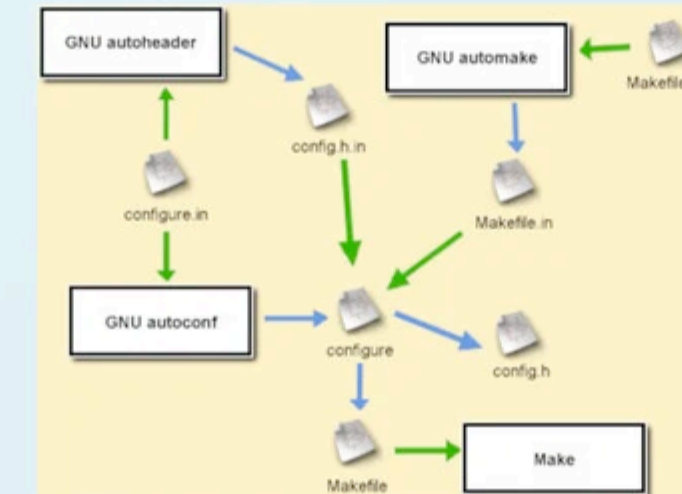
- YEARS of COMPILATION and yet NO REAL-WORLD USE FOUND for anything more complicated than build.sh
- Wanted cross-platform support anyway, for a laugh? We had a tool for that: It was called "build.bat"
- "Something in this pile of spaghetti is enabling -Werror and I can't figure out how to disable it" - Statements dreamed up by the utterly Deranged

LOOK at what DevOps Engineers have been demanding your Respect for all this time, with all the source code and toolchains we built for them

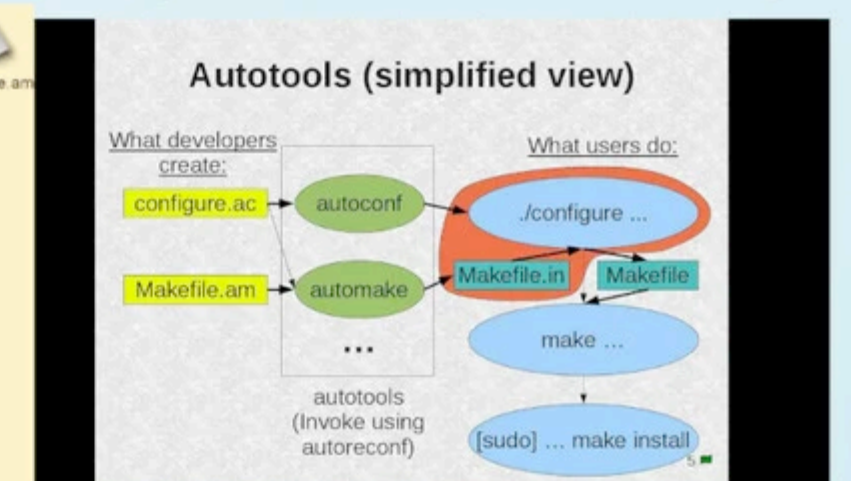
(These are REAL Build Systems, made by REAL Developers):



?????



???????



????????????????????

"The CI build pipeline is broken again"  
They have played us for absolute fools

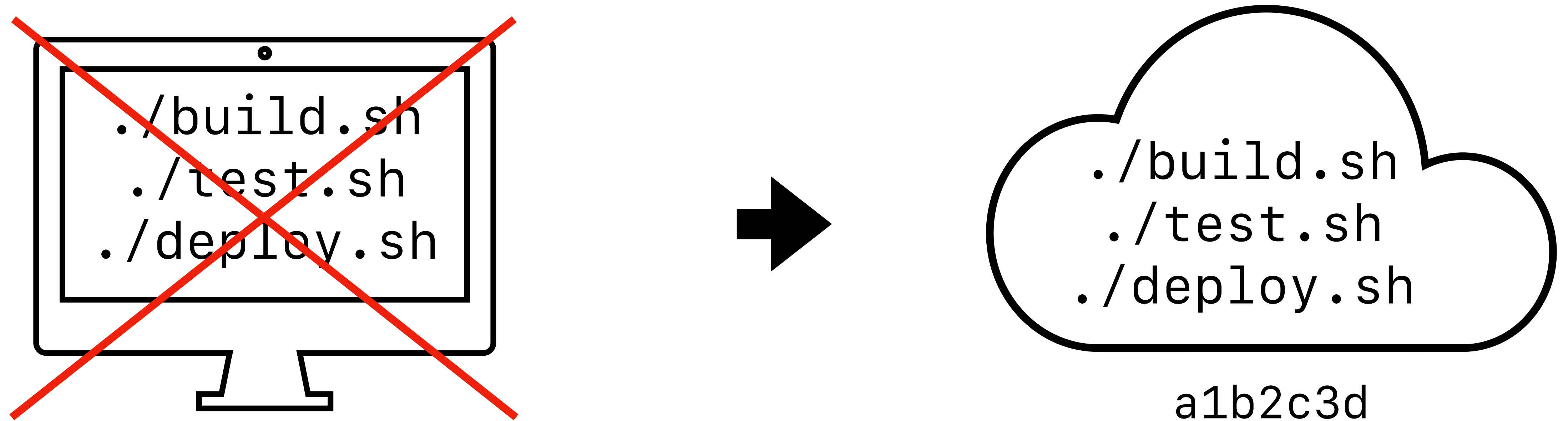
# My Background

- Lots of tinkering with "**GitHub Actions**"
  - personal/open source projects
- **Intern @ RIPE DB**, Jun - Nov 2024
  - Java & Linux/FreeBSD tinkering
- **(Re-)joined RIPE Atlas**, Jan 2025
  - first time dealing with **GitLab CI**



# What's GitLab CI?

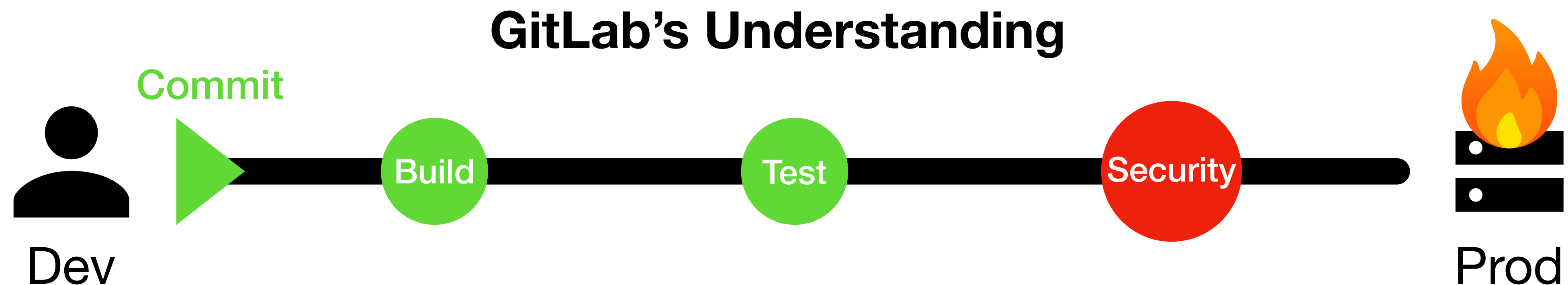
The CI framework provided by a software forge, GitLab



# Get started with GitLab CI/CD

## Step 1: Configure your pipeline

To use GitLab CI/CD, you start with a `.gitlab-ci.yml` file at the root of your project. This file specifies the stages, jobs, and scripts to be executed during your CI/CD pipeline. It is a YAML file with its own custom syntax.



Docker-based →

./build.sh

./test.sh

./deploy.sh

```
.gitlab-ci.yml

image: gradle:alpine

before_script:
  - GRADLE_USER_HOME="$(pwd)/.gradle"
  - export GRADLE_USER_HOME

build:
  stage: build
  script: gradle --build-cache assemble

test:
  stage: test
  script: gradle check

deploy:
  stage: deploy
  script: echo "Define your deployment script!"
```

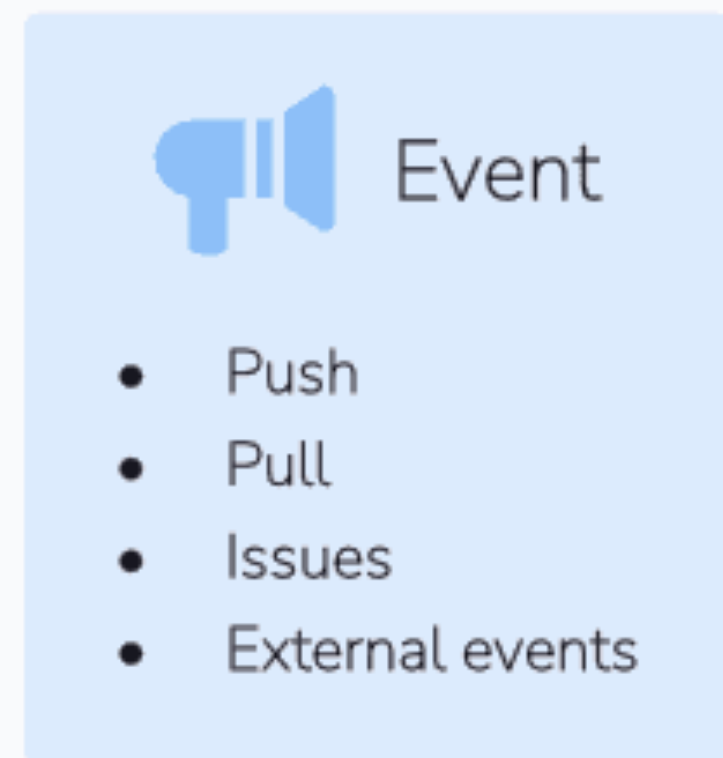
GitLab Runner provides the following executors:

- [SSH](#)
- [Shell](#)
- [Parallels](#)
- [VirtualBox](#)
- [Docker](#)
- [Docker Autoscaler](#)
- [Docker Machine \(auto-scaling\)](#)
- [Kubernetes](#)
- [Instance](#)
- [Custom](#)

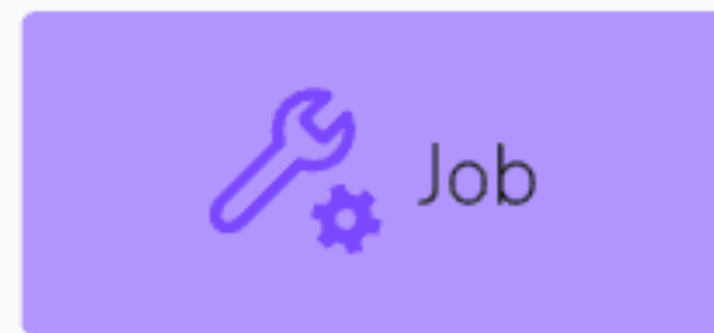
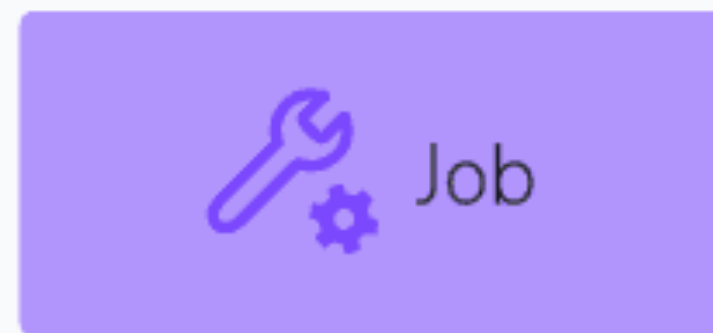
“Docker only”  
...or is it?

# My understanding of “CI”

- **Scripting** on the **Git** remote
  - user is **authenticated**
  - environment is **controlled**
  - **execution proof** through **logs**



= `.gitlab-ci.yml`  
= 1 pipeline  
(You can have many of these)



VM =

Shell Command  
/  
Shared “Step” =  
(e.g. Docker  
or “Action”)

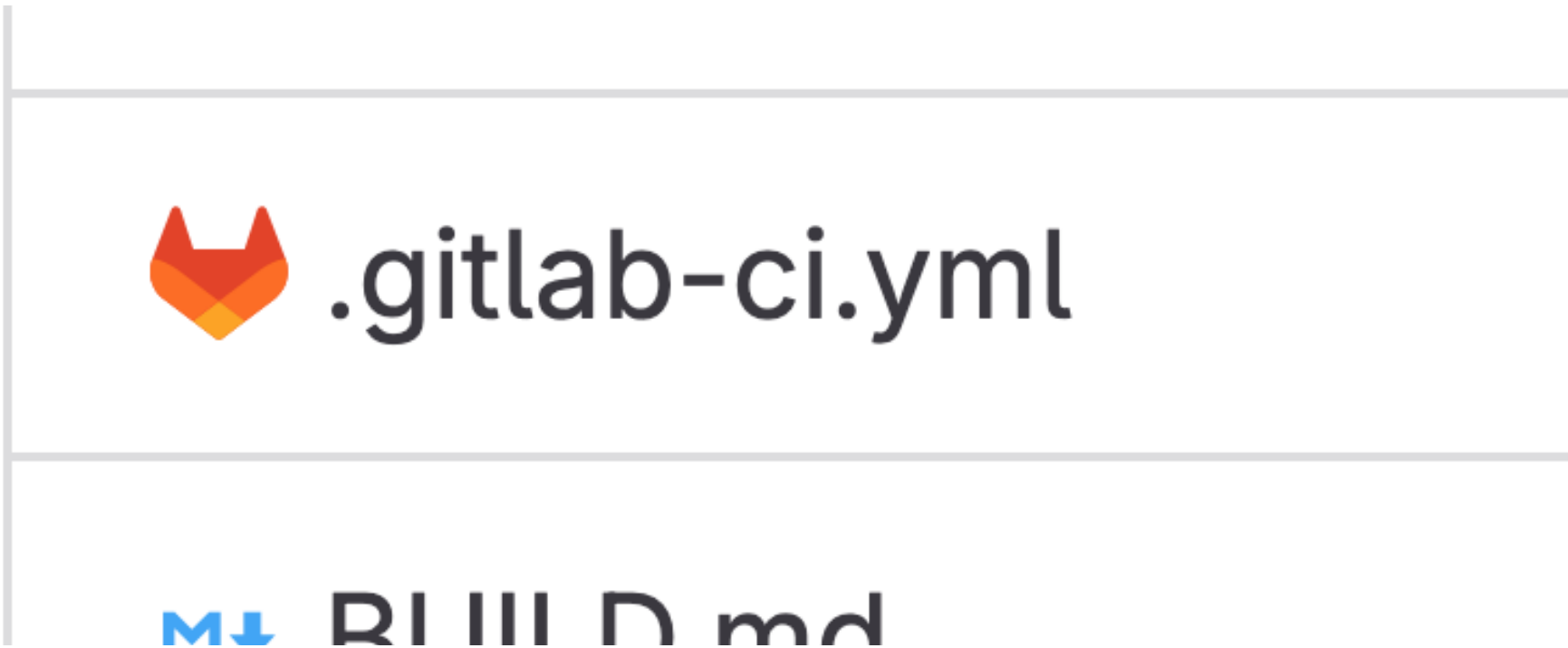
# What?

1. Trigger (n)
  - Something happens
2. Workflow (1)
  - “Build & Test”
3. Job (n)
  - “Build for Ubuntu”
  - `runs-on: ubuntu-24.04`
4. Steps (n)
  - `dpkg-buildpackage`

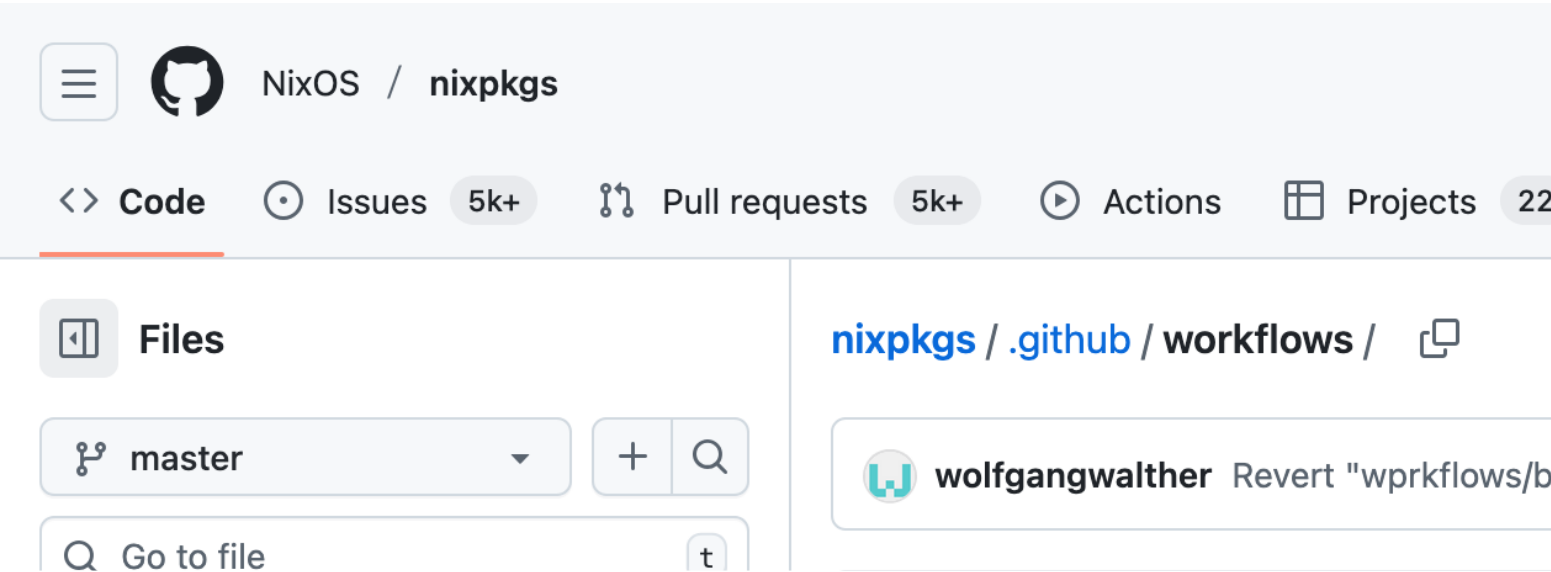


# Difference

1 GitLab CI pipeline (max)



triggered on commits



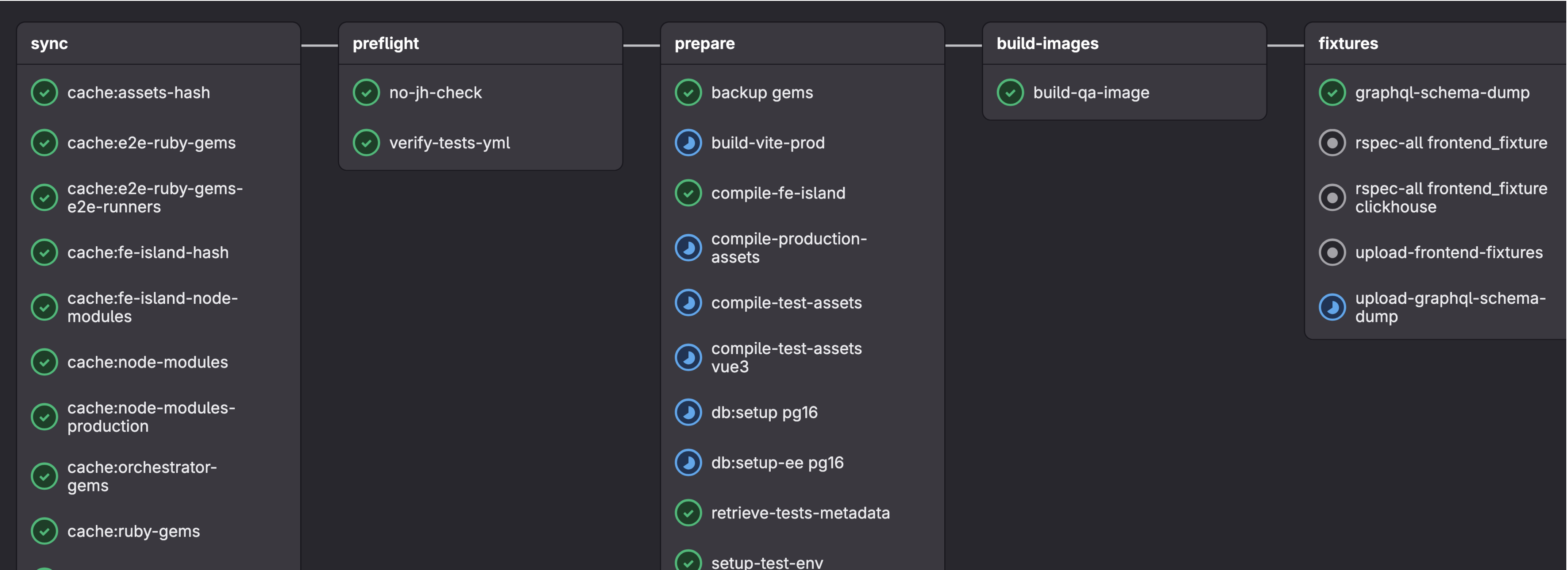
18 different workflows  
triggered on *anything*

- README.md
- backport.yml
- bot.yml
- build.yml
- check.yml
- comment.yml
- edited.yml
- eval.yml
- lint.yml
- merge-group.yml
- periodic-merge-24h.yml
- periodic-merge-6h.yml
- periodic-merge.yml
- pull-request-target.yml
- review.yml
- reviewed.yml
- reviewers.yml
- teams.yml
- test.yml
- ISSUE\_TEMPLATE.md
- PULL\_REQUEST\_TEMPLATE.md
- STALE-BOT.md

- backport.yml
- bot.yml
- build.yml
- check.yml
- comment.yml
- edited.yml
- eval.yml
- lint.yml
- merge-group.yml
- periodic-merge-24h.yml
- periodic-merge-6h.yml
- periodic-merge.yml
- pull-request-target.yml
- review.yml
- reviewed.yml
- reviewers.yml
- teams.yml
- test.yml

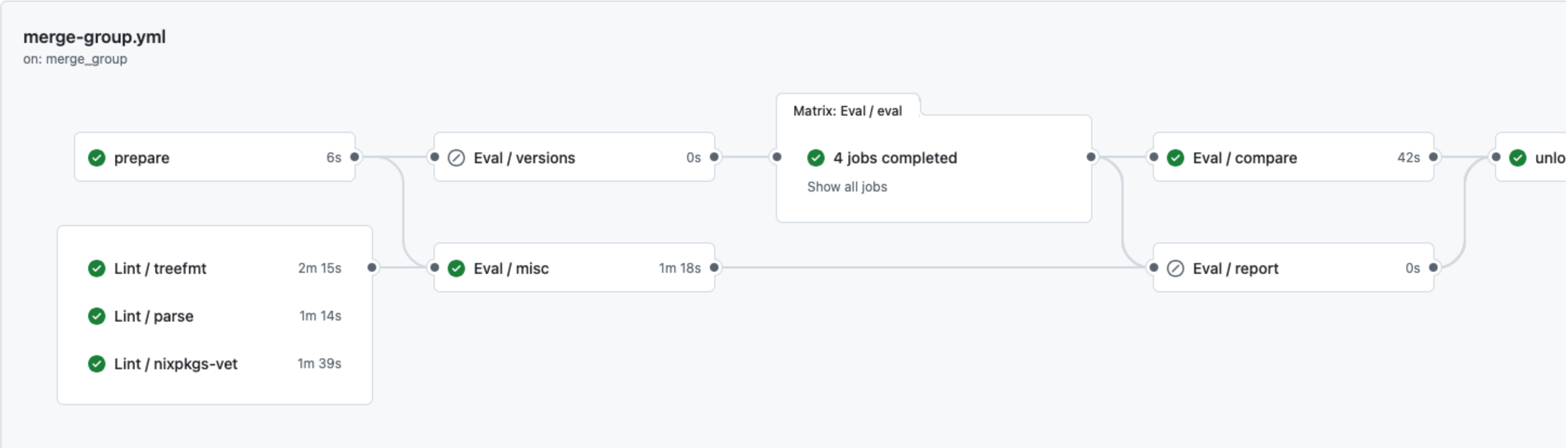
# GitLab

## The only Pipeline



# GitHub

## One pipeline (of many)



# “Get to the point”

- **kaniko**
- Building containers  
*in* containers  
*fast*  
*without dind*  
*without socket mount*

The screenshot shows the GitHub repository page for `GoogleContainerTools / kaniko`. The repository is archived, with a yellow banner stating: "This repository was archived by the owner on Jun 3, 2025. It is now read-only." The repository has 707 issues, 54 pull requests, and 15.7k stars. The main branch is selected. The file list shows a commit by `seenkey` titled "Add archive n..." with a commit hash of `236ba56` from 6 months ago. The file list includes `.github`, `cmd`, `deploy`, and `docs`. The right sidebar shows the repository's description: "Build Container Images In Kubernetes", and tags for `docker`, `kubernetes`, `containers`, and `developer-tools`. The sidebar also lists the README, Apache-2.0 license, and Code of conduct.

GoogleContainerTools / kaniko

Code Issues 707 Pull requests 54 Discussions Actions

This repository was archived by the owner on Jun 3, 2025. It is now read-only.

kaniko Public archive Watch 144 Fork 1.5k Star 15.7k

main Go to file Code

seenkey Add archive n... 236ba56 · 6 months ago

.github	fix: --body-file fl...	7 months ago
cmd	warmer validate ...	last year
deploy	chore: upgrade ...	6 months ago
docs	Update design...	2 years ago

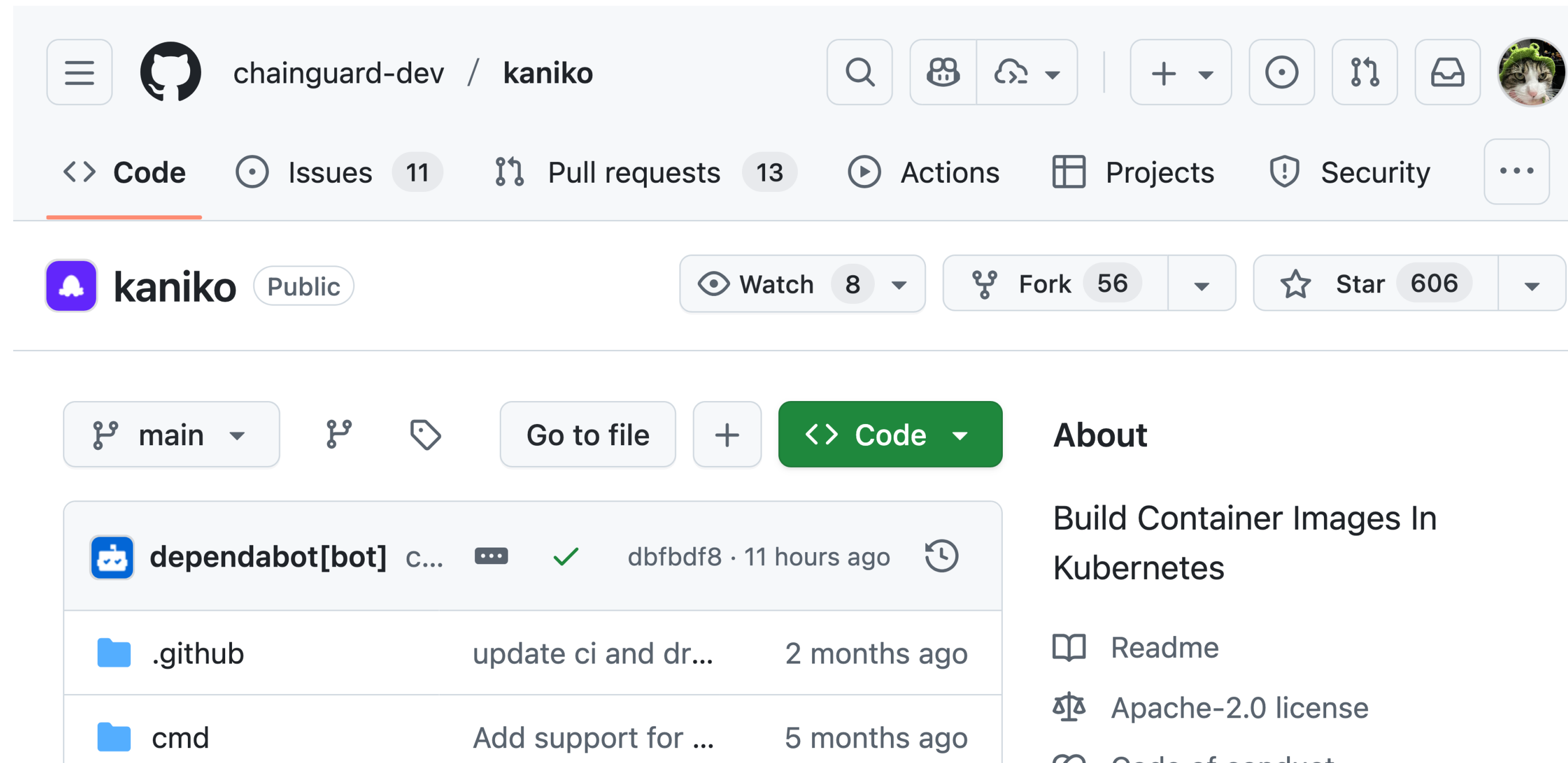
About

Build Container Images In Kubernetes

docker kubernetes containers developer-tools

Readme Apache-2.0 license Code of conduct

# chainguard to the rescue?



*Users must build these artifacts themselves or  
use the **Chainguard** container images as  
customer*


# How to mirror

- The repo lives on GitHub, and **needs to be mirrored** to GitLab


Trix Taiclet / kaniko / Repository settings

▼ Mirroring repositories

Set up your project to automatically push and/or pull changes to/from another r

Mirrored repositories  1

Repository	Direction
<a href="https://github.com/chainguard-dev/kaniko.git">https://github.com/chainguard-dev/kaniko.git</a> <div>Specific branches</div>	Pull

 Trix Taiclet / kaniko / Repository

main ▼

kaniko


kaniko

+ ▼

Find file

Code ▼

⋮


 kaniko

chore(deps): bump github.com/aws/aws-s...  
dependabot[bot] authored 12 hours ago and GitHub com...






⋮

Unverified

dbfbdf82



History

Name	Last commit	Last update
 .github	update ci and drop integratio...	1 month ago
 cmd	Add support for --annotation...	4 months ago
 deploy	chore(deps): bump golang fr...	2 months ago
 docs	Update source repo URL (#93)	3 months ago
 examples	Update image reference in e...	3 months ago

# How to CI on mirror?

- We now have a mirror. How do we build?
- It has GitHub Actions (kinda... theirs-only), but **no GitLab CI**
- What I'd do on GitHub Actions:
  - Create a new workflow file
  - Make it trigger automatically (e.g. schedule)
  - **no use of mirroring** needed



# Turning GitLab CI into GitHub Actions (1)

- “One `.gitlab-ci.yml` per project directory” ?!
- Do I need to create a new repo?
- You can have multiple root directories!
- It's called **orphan branches**  
-> you only mirror main

Trix Taiclet / kaniko / Repository

build kaniko

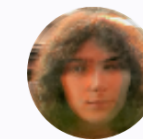
kaniko

+ ▾

Find file

Code ▾

⋮



Update `.gitlab-ci.yml` file

Trix Taiclet authored 3 weeks ago



2e9fbd87



History

Name

Last commit

Last update



`.gitlab-ci.yml`

Update `.gitlab-ci.yml` ...

3 weeks ago

# Turning GitLab CI into GitHub Actions (2)

- How to trigger the build branch pipeline?

- API call  
w/ branch




Trix Taiclet / kaniko / CI/CD Settings

▼ Pipeline trigger tokens

Trigger a pipeline for a branch or tag by generating a trigger token and using it with an API call. The token impersonates a user's project access and permissions. [Learn more.](#)

Active pipeline trigger tokens 1

Reveal valuesAdd new token

Token	Description	Owner	Last Used	Expires	Actions
*****	Main push trigger		12 hours ago	Never	 

▼ View trigger token usage examples

These examples show common methods of triggering a pipeline with a pipeline trigger token. The URL and ID for this project is prefilled.

In each example, replace `TOKEN` with the trigger token you generated and replace `REF_NAME` with the branch or tag name.

Use cURL

```
curl -X POST \
  --fail \
  -F token=TOKEN \
  -F ref=REF_NAME \
  https://gitlab.example.com/api/v4/projects/1234/trigger/pipeline
```

specify branch in call



# Turning GitLab CI into GitHub Actions (3)

- How to trigger the build branch pipeline **on push/mirror?**

- **Webhook**  
to  
API call  
w/ branch

🏠 Trix Taiclet / kaniko / Webhook settings

🔍 Search page

## Webhooks

Use [webhooks](#) to send notifications to external applications in response to events in a GitLab project or group. Consider using an [integration](#) instead.

Webhooks 📌 1

Add new webhook

Build when push on main

https://gitlab.example.com/api/v4/projects/1234/ref/build/

trigger/pipeline?token={trigger\_token}

Test ▾

Edit

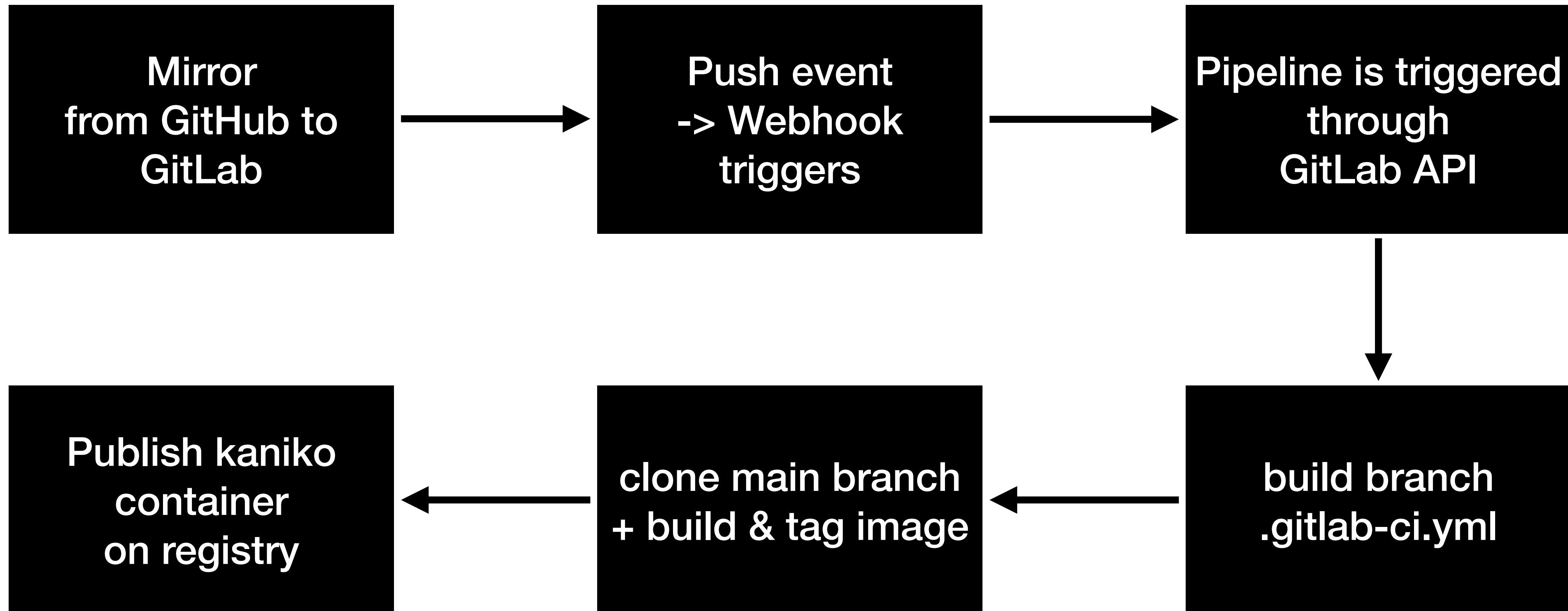
Delete

Push events

SSL Verification: enabled

this is the build branch, specifically

# Turning GitLab CI into GitHub Actions (4)



# Turning GitLab CI into GitHub Actions (5)



Trix Taiclet / kaniko / Container registry / 1

kaniko 



 5 tags  Cleanup will run in 1 hour  Created Oct 9, 2025 14:40

Filter results



Name ▾



☐ Select all



v1.25.3  ▾

15.00 MiB

Published 13 hours ago

Digest: ed610be



latest  ▾

15.00 MiB

Published 13 hours ago

Digest: ed610be



# Recap: What did I rant about?

- “CI” is meaningless -> it’s just **scripts** that embed **in your Git remote**
- You can have **many** CI pipelines *per repo* that trigger on anything you want
- You can do anything in zombo.com CI, anything at all
  - CI is purely **scripting in “the cloud”** (someone else’s computer)
- It’s your turn: **script your things** in “CI”



**Thank you!**

**Questions? Anger?**

[trix@c3l.lu](mailto:trix@c3l.lu)