GOTO
Guide

LET US HELP YOU

Ask questions through **the app**

★★★★★

also remember to rate session

THANK YOU!

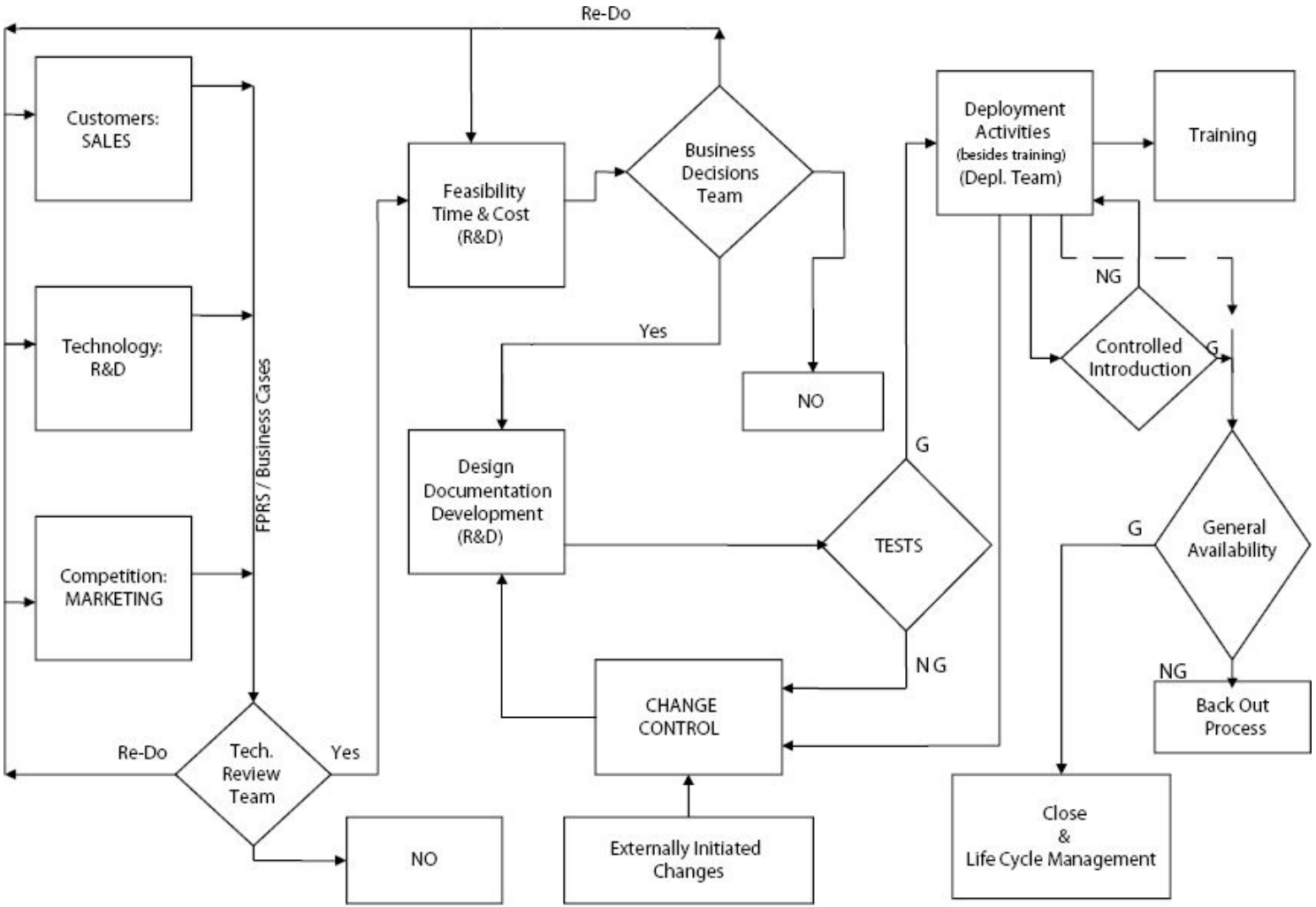
#GOTOams

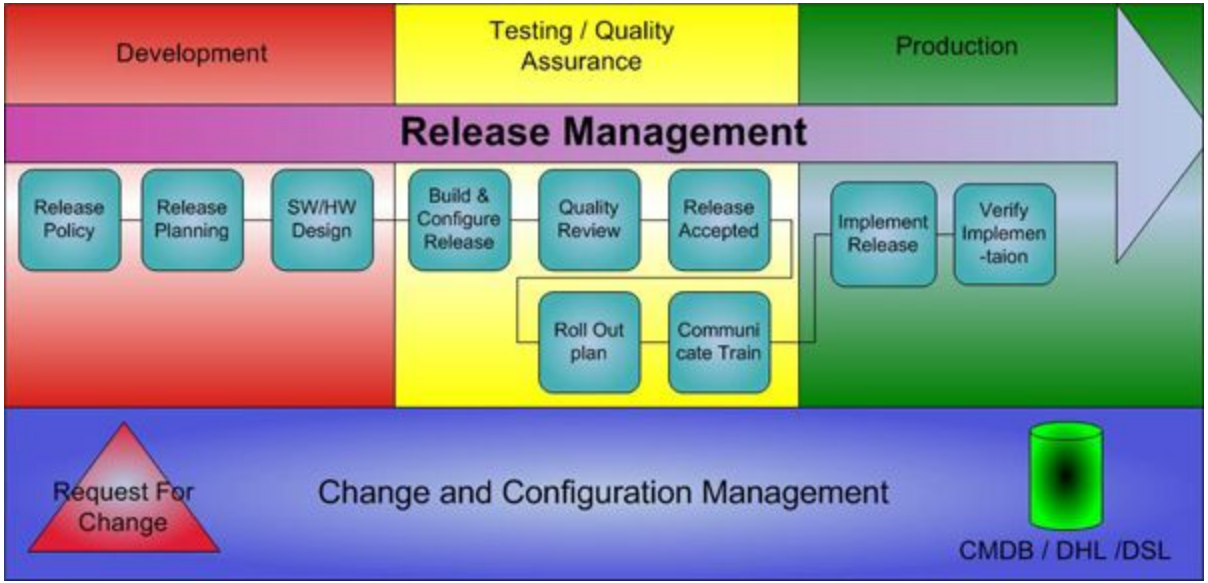
Cloud-Native Progressive Delivery

Matt Turner

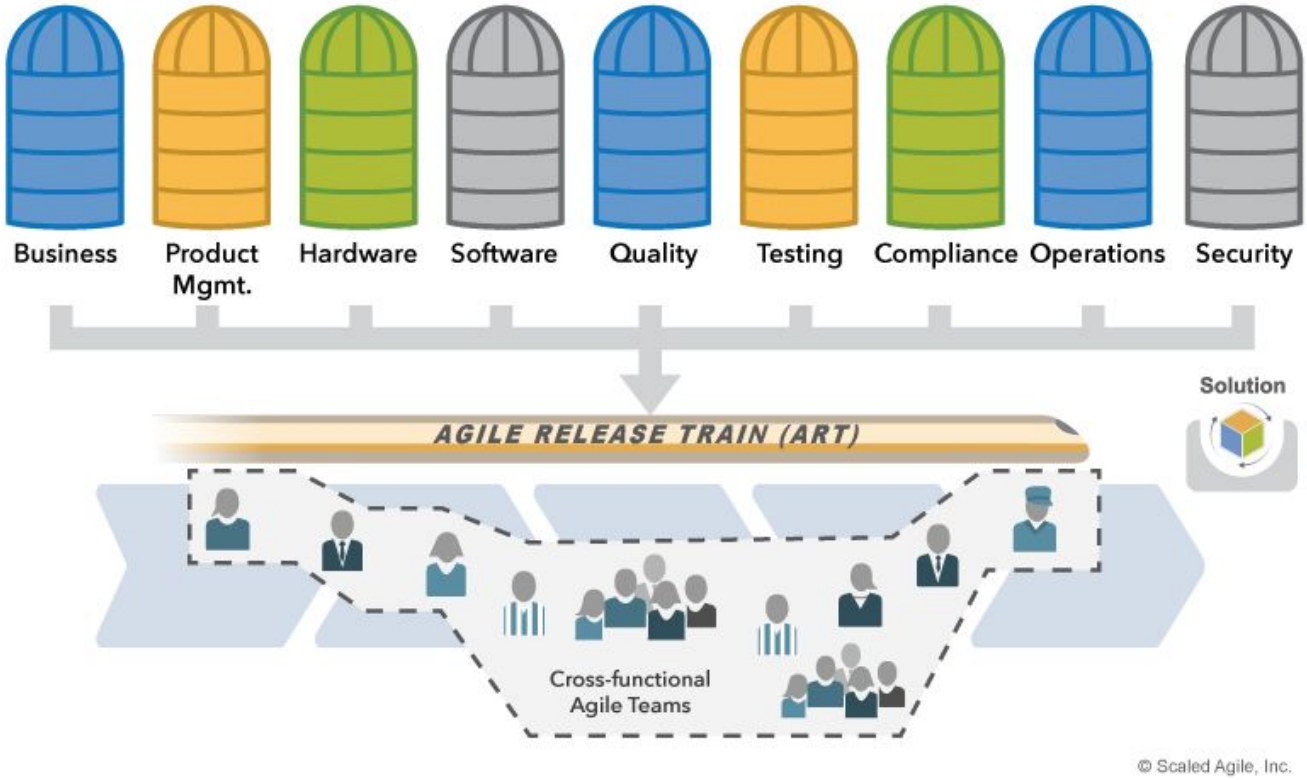
GOTO, Amsterdam Netherlands | June 2022

@mt165 | mt165.co.uk


Re-Do
Customers: SALES
Technology: R&D
Competition: MARKETING
FPRS / Business Cases
Feasibility Time & Cost (R&D)
Business Decisions Team
Yes
NO
Design Documentation Development (R&D)
TESTS
G
NG
CHANGE CONTROL
Externally Initiated Changes
Tech. Review Team
Re-Do
Yes
NO
Deployment Activities (besides training) (Depl. Team)
Training
NG
Controlled Introduction
G
G
General Availability
NG
Back Out Process
Close & Life Cycle Management

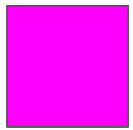

Development
Testing / Quality Assurance
Production
Release Management
Release Policy
Release Planning
SW/HW Design
Build & Configure Release
Quality Review
Release Accepted
Roll Out plan
Communi cate Train
Implement Release
Verify Implemen -taion
Request For Change
Change and Configuration Management
CMDB / DHL /DSL


SOFTWARE DEVELOPMENT LIFE CYCLE
Phases, Models, Process and Methodologies
© www.SoftwareTestingHelp.com

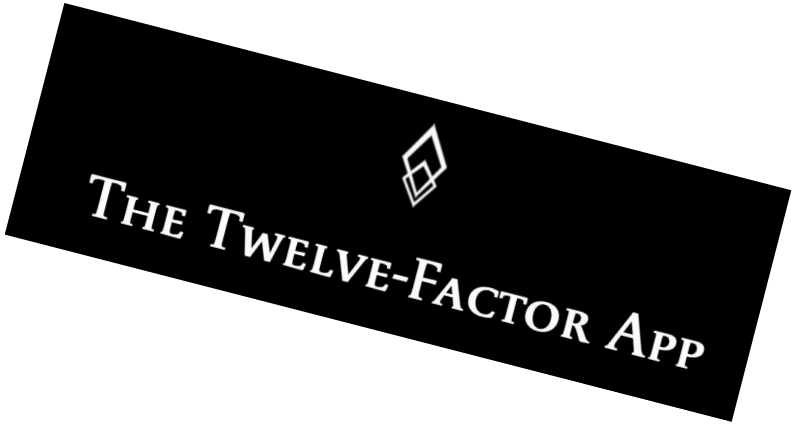

Business
Product Mgmt.
Hardware
Software
Quality
Testing
Compliance
Operations
Security
Solution
AGILE RELEASE TRAIN (ART)
Cross-functional Agile Teams
© Scaled Agile, Inc.

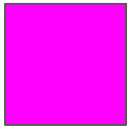
How Does Cloud-Native Enable This?

Technology Overview

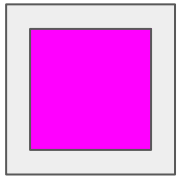


12 Factor Apps

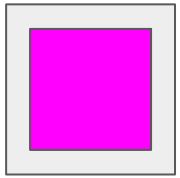




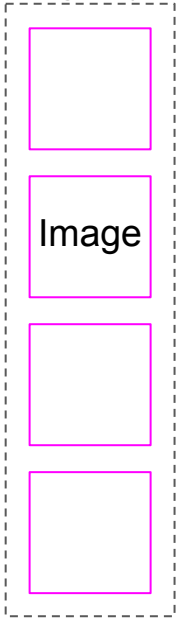
Docker



Docker

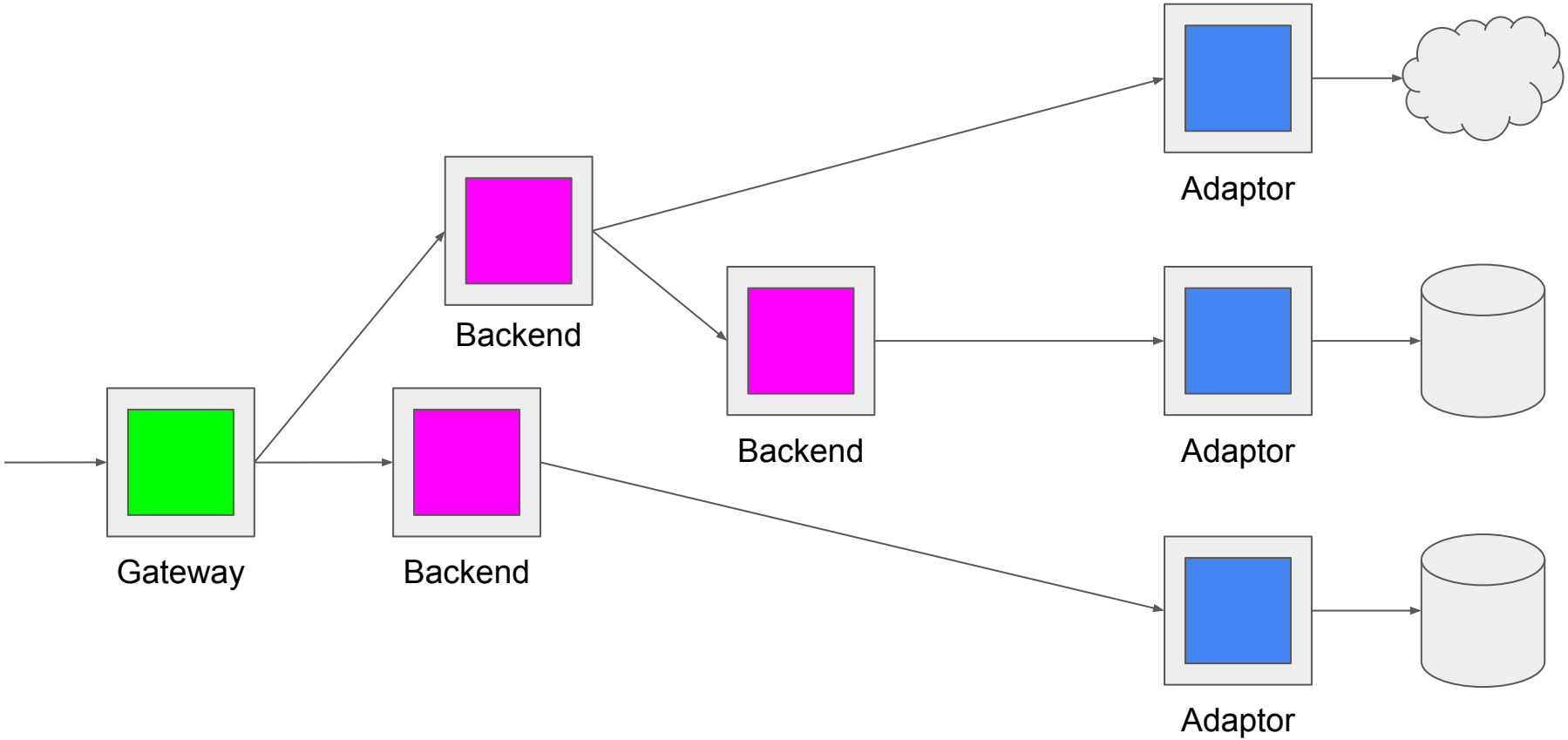


Registry

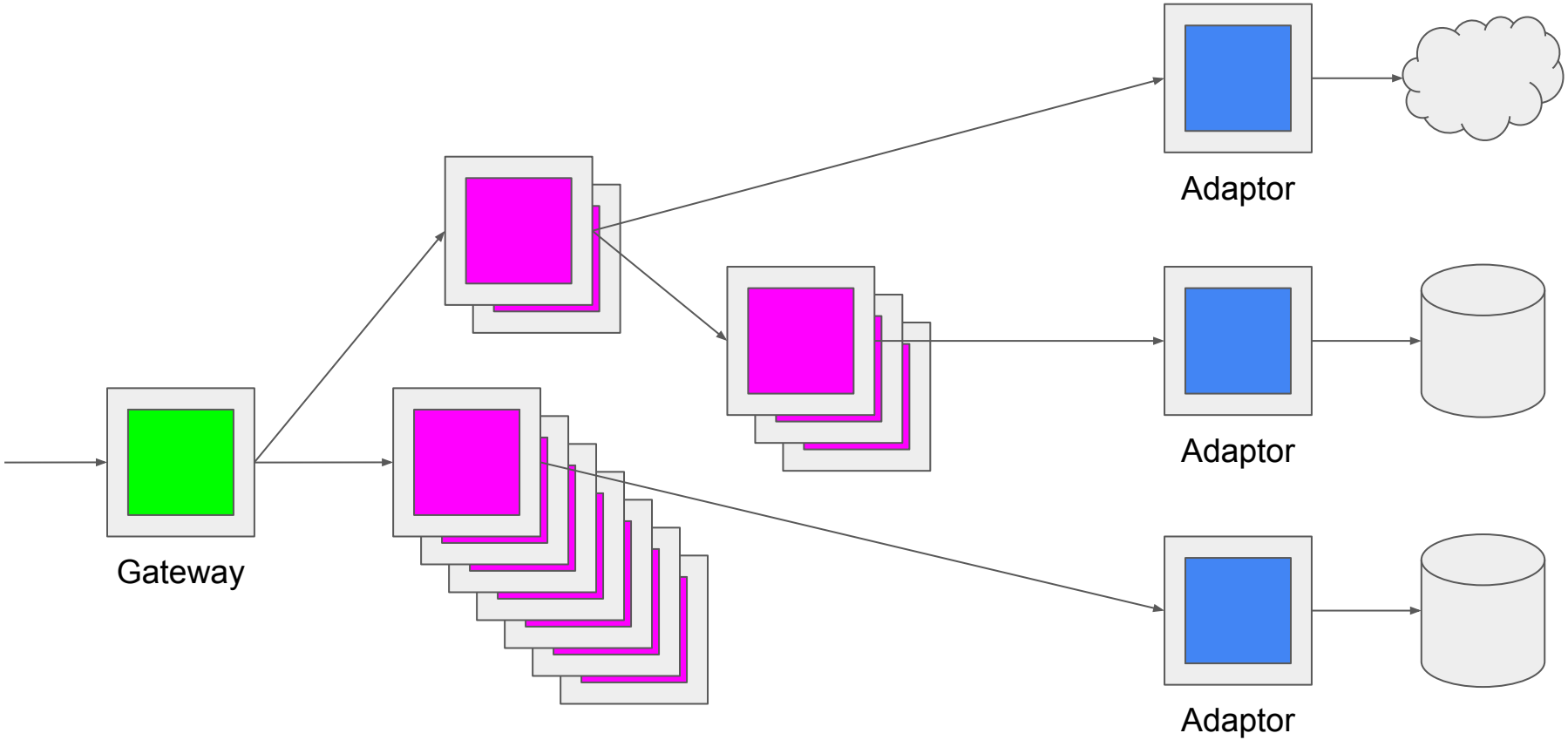


V. Strictly separate build and run

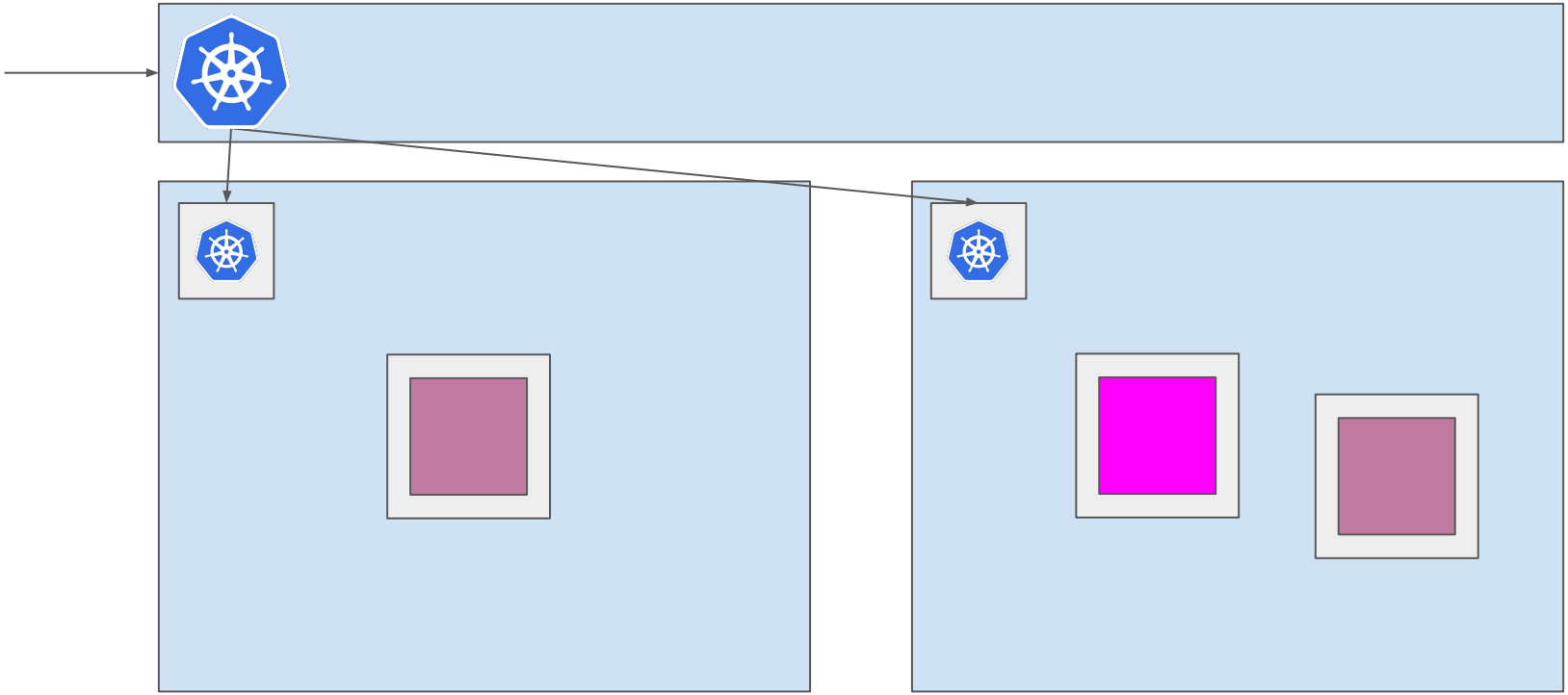
Microservices



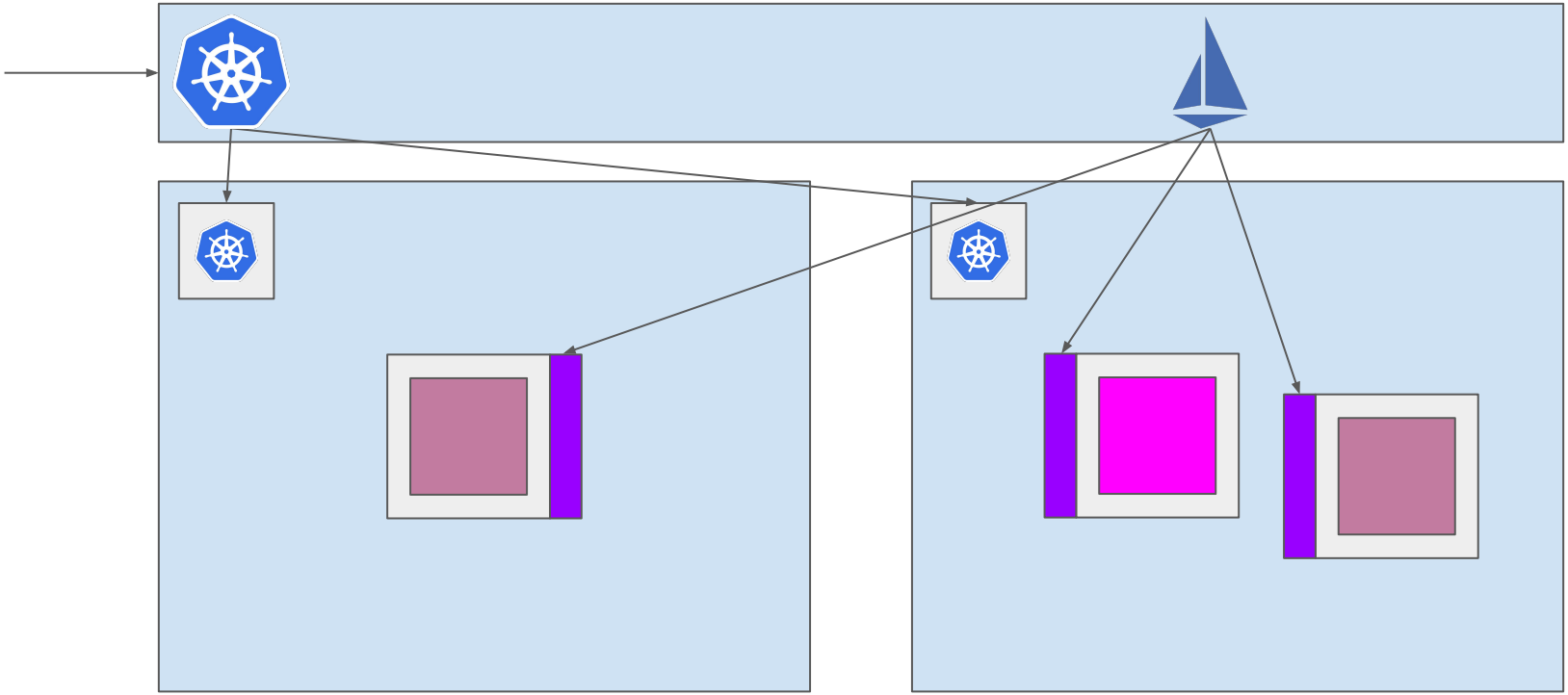
Microservices



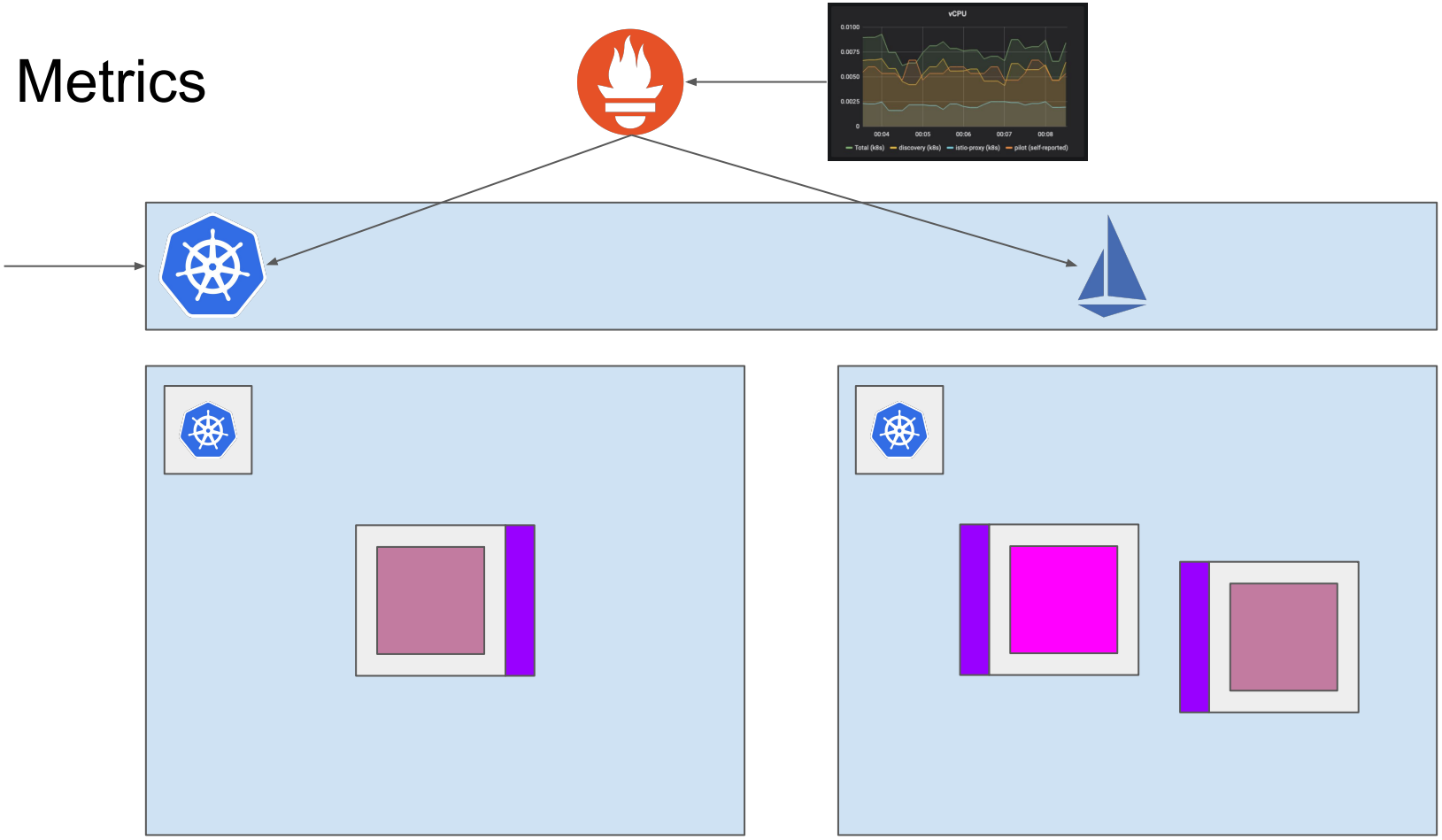
Kubernetes



Service Mesh



Metrics


vCPU
0.0100
0.0075
0.0050
0.0025
0
00:04
00:05
00:06
00:07
00:08
Total (k8s)
discovery (k8s)
istio-proxy (k8s)
pilot (self-reported)

Metrics

“RED”

- Rate - requests / second
- Errors - errors (%)
- Duration - latency of responses

Service Levels

- SLA - Service Level Agreement - broad statement of what's on offer, reads like a contract
- SLO - Service Level Objective - measurable, quantified target for availability, performance, etc. Eg error rate %, latency ms.
- SLI - Service Level Indicator - how will we measure the service level? How are we measuring things? Where? How are we aggregating them?

Current Deployment Practices

Continuous Integration

- These days actually means Continuous Build
- Original meaning still relevant and coming later

Deployment

- Taking a software package and running it

Continuous Deployment

- Deploying every time there’s a new build

Progressive Delivery

Release

n. Exposing a piece of software to *users*

— Matt

Continuous Release

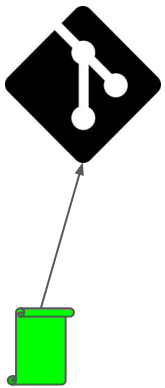
- Exposing users to every new Deployment
- => Exposing them to every new Build

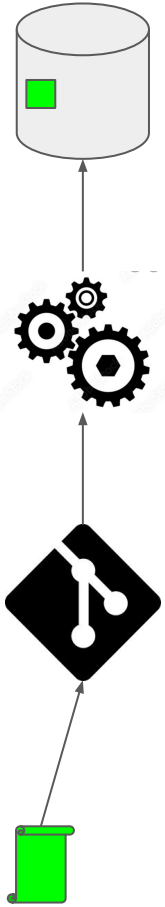
Deployment == Release?

Deployment ≠ Release!

We Have the Technology!

Build

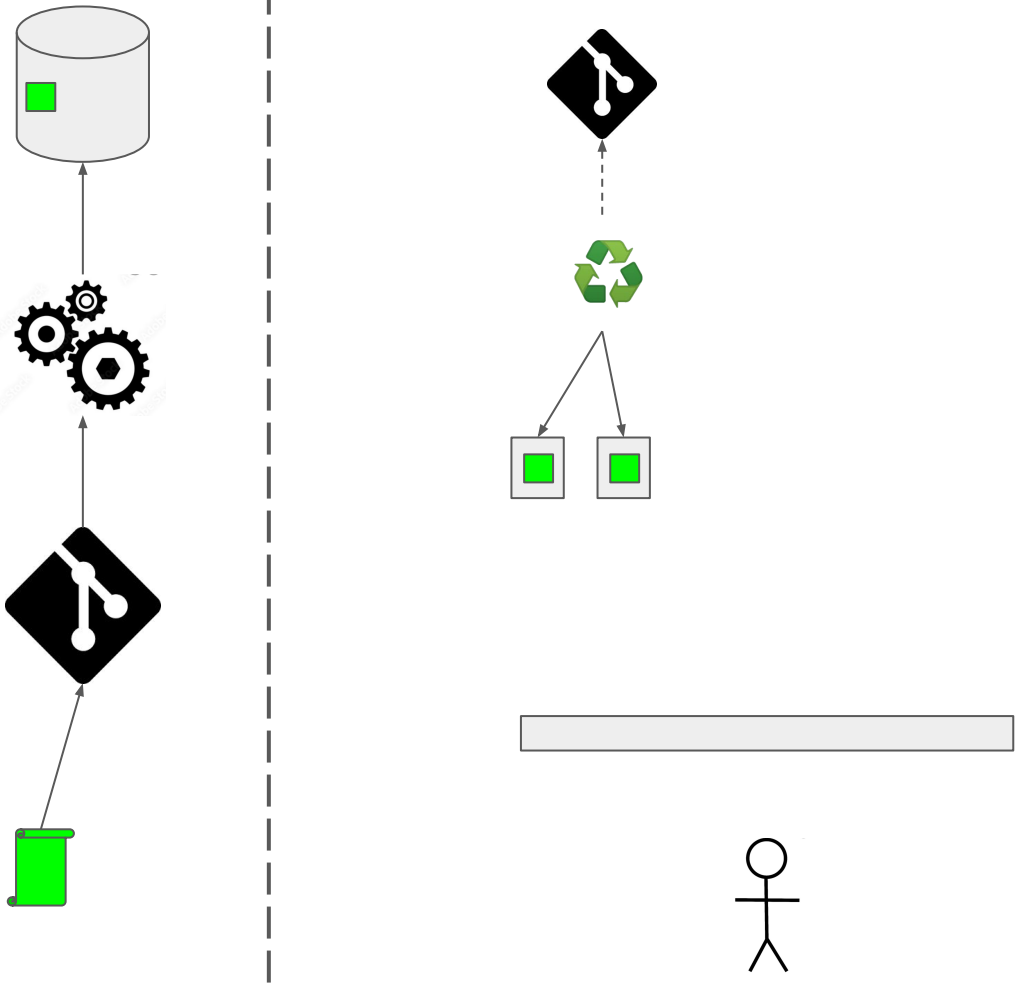


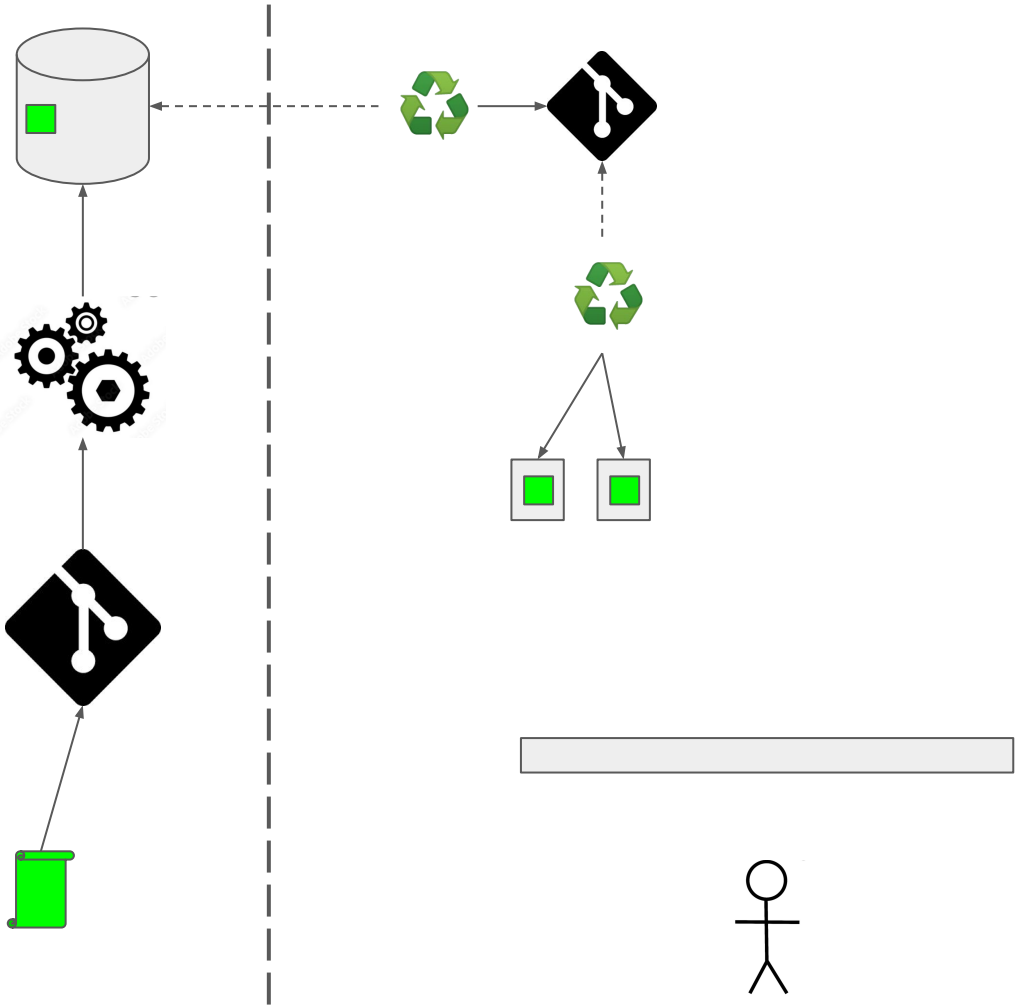


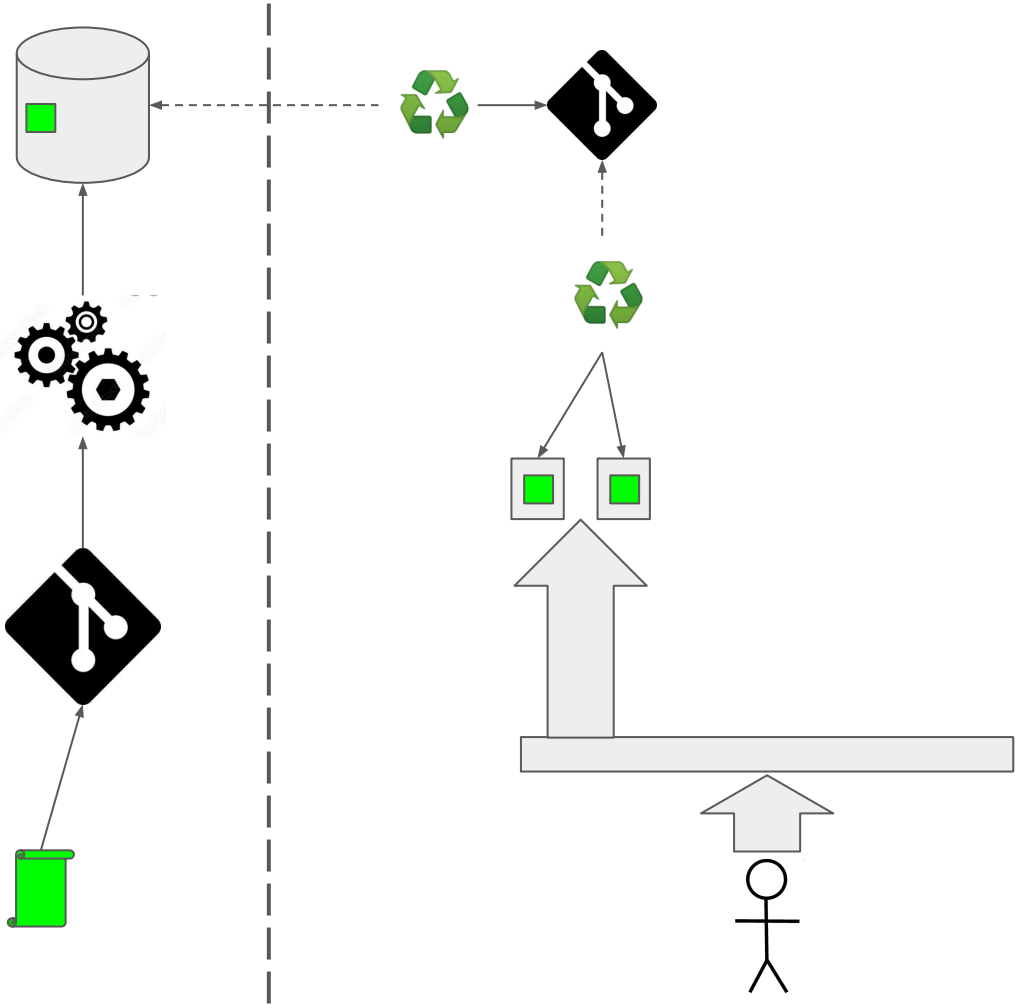
Contract

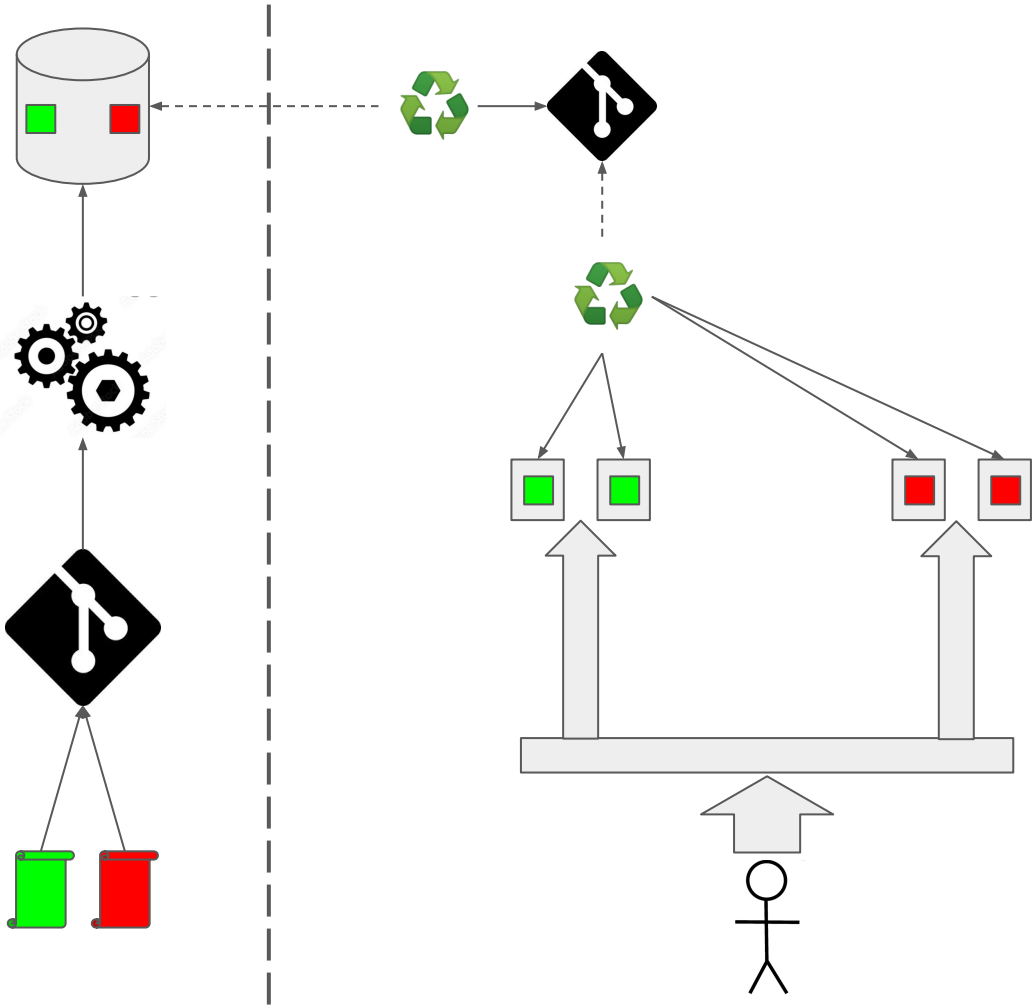
- Triggered by a new commit to main
- Produces a new container image and push to the registry
- Bottom of the testing pyramid: Linting, Compilation, Unit Testing

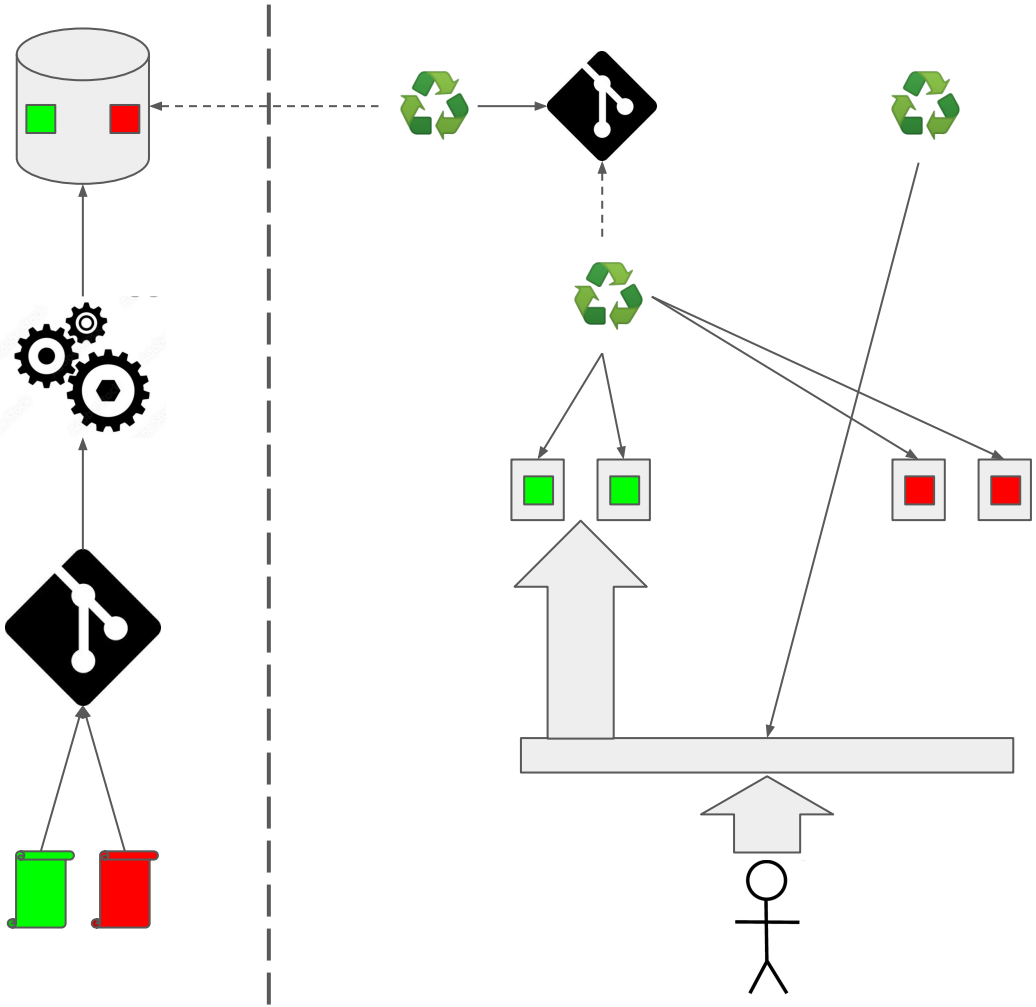
Deploy





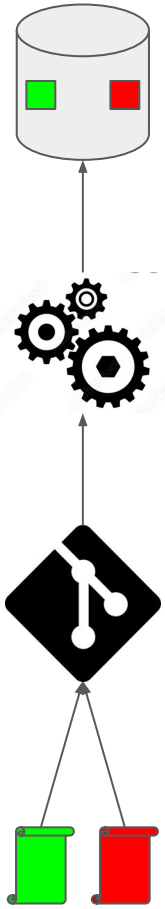


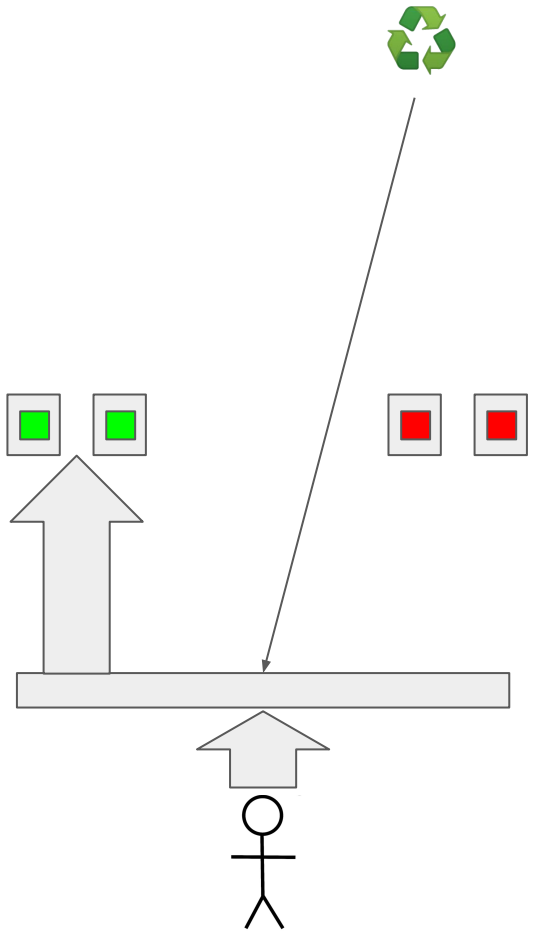


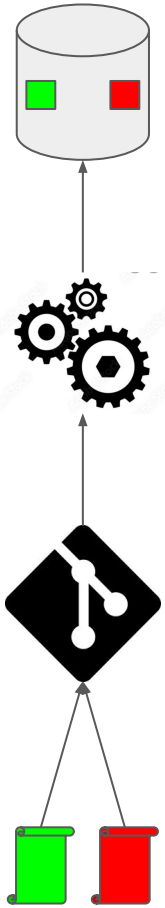


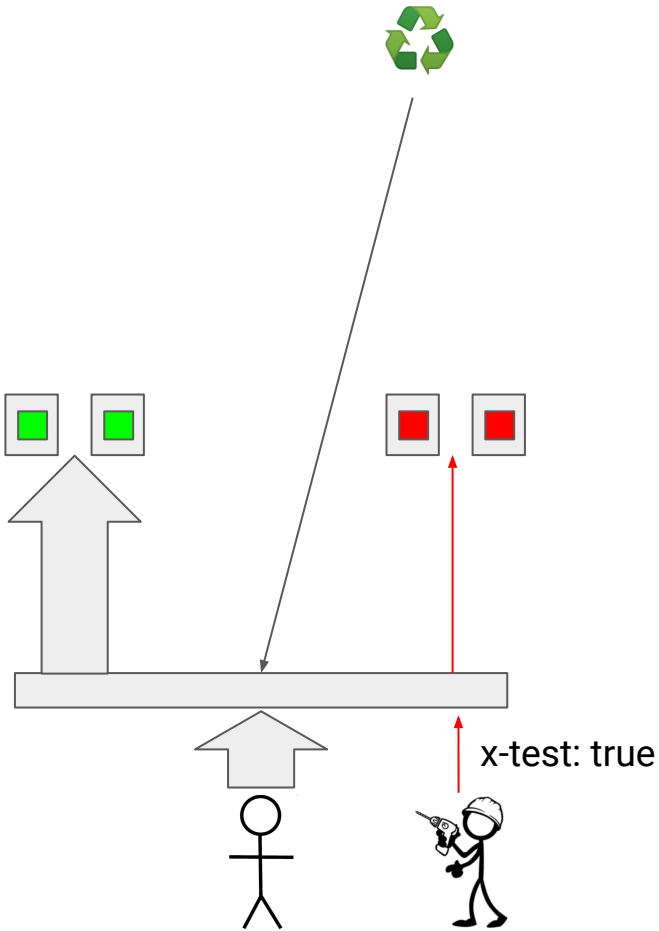
Contract

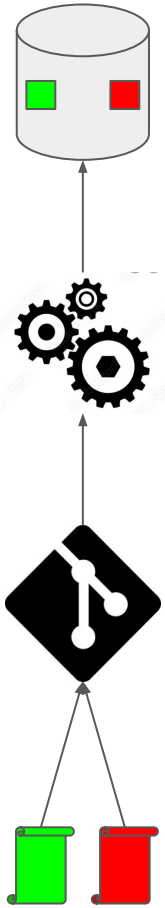
- Triggered by a new image appearing
- Deploys to prod cluster, prod namespace
- No user traffic

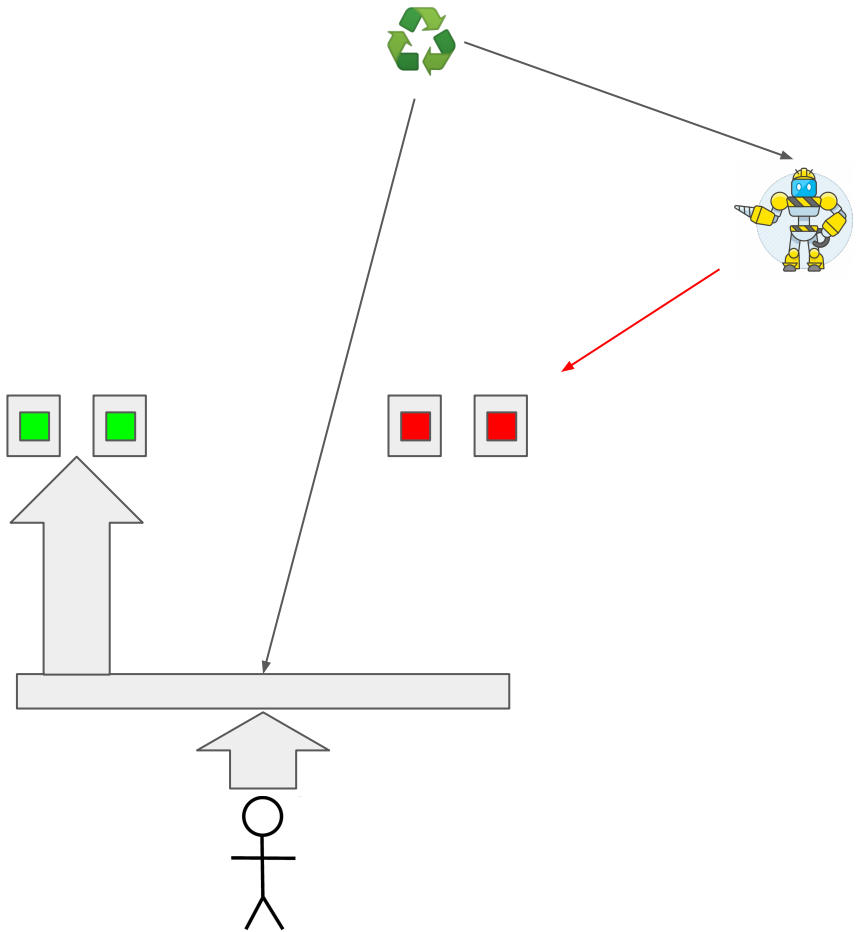


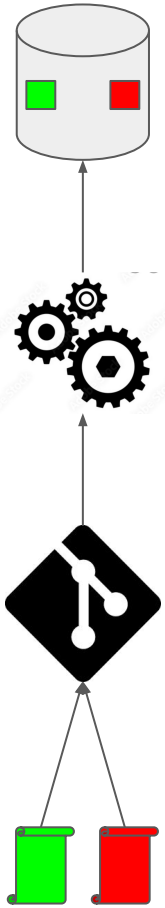


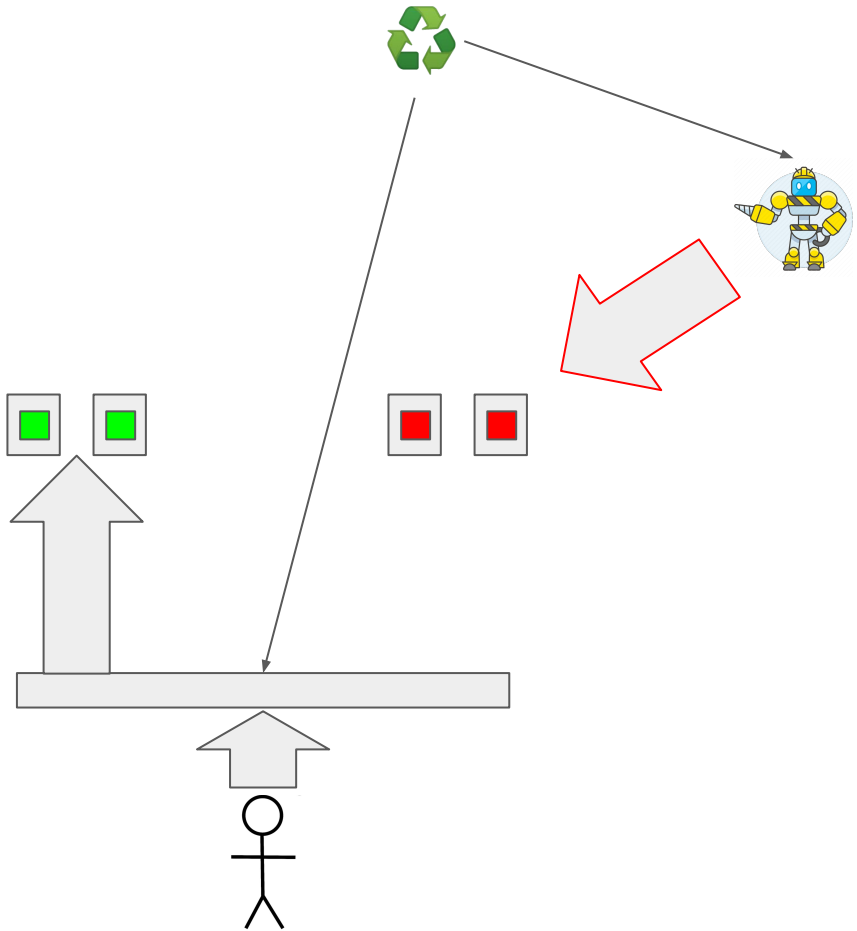



x-test: true





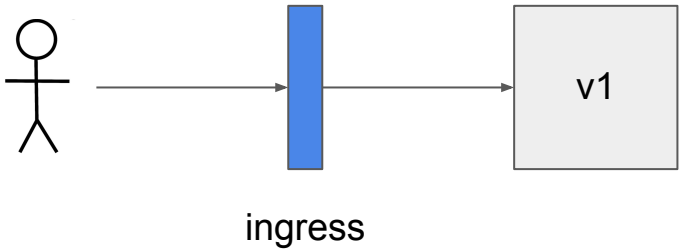




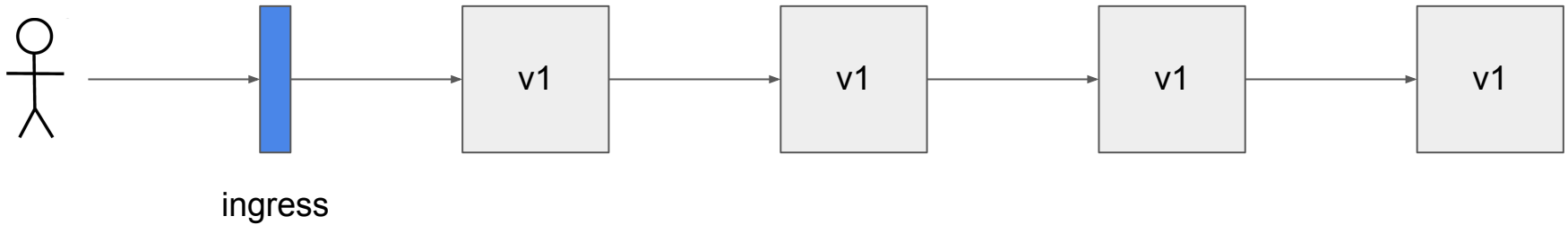
Stage: Running in Prod, but Isolated from Users

- Does it even start?
- Available for manual testing
- Automated integration testing
- Automated end-to-end testing
- Automated non-functional testing
    - Failed if performance isn’t within SLO

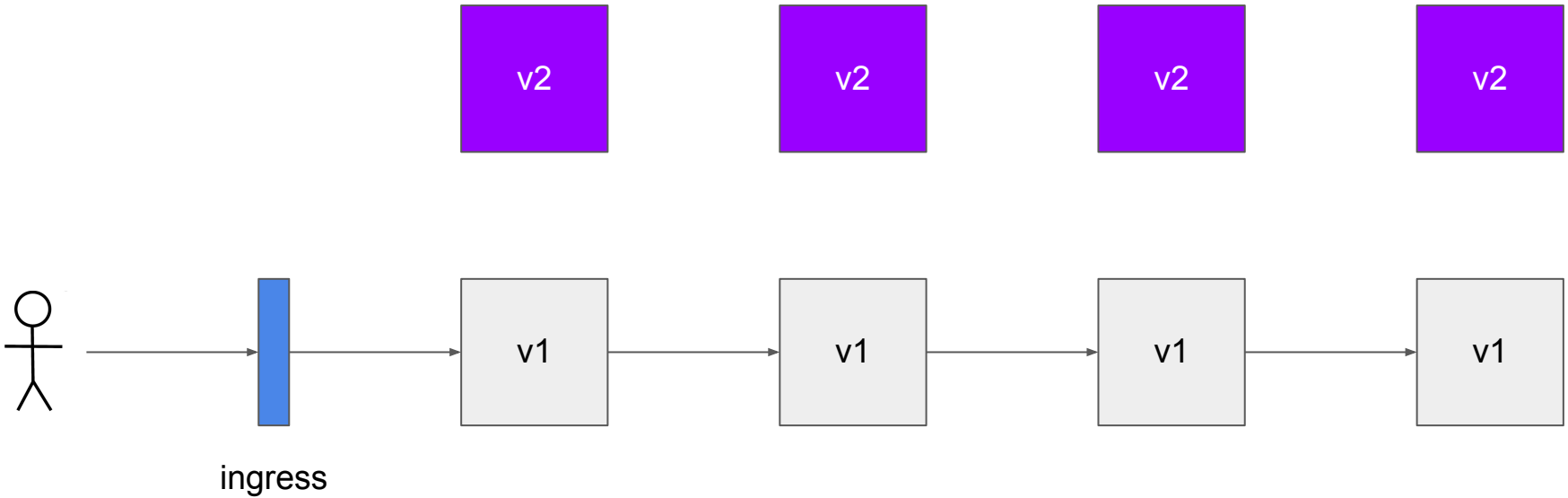
Per-Service Overrides



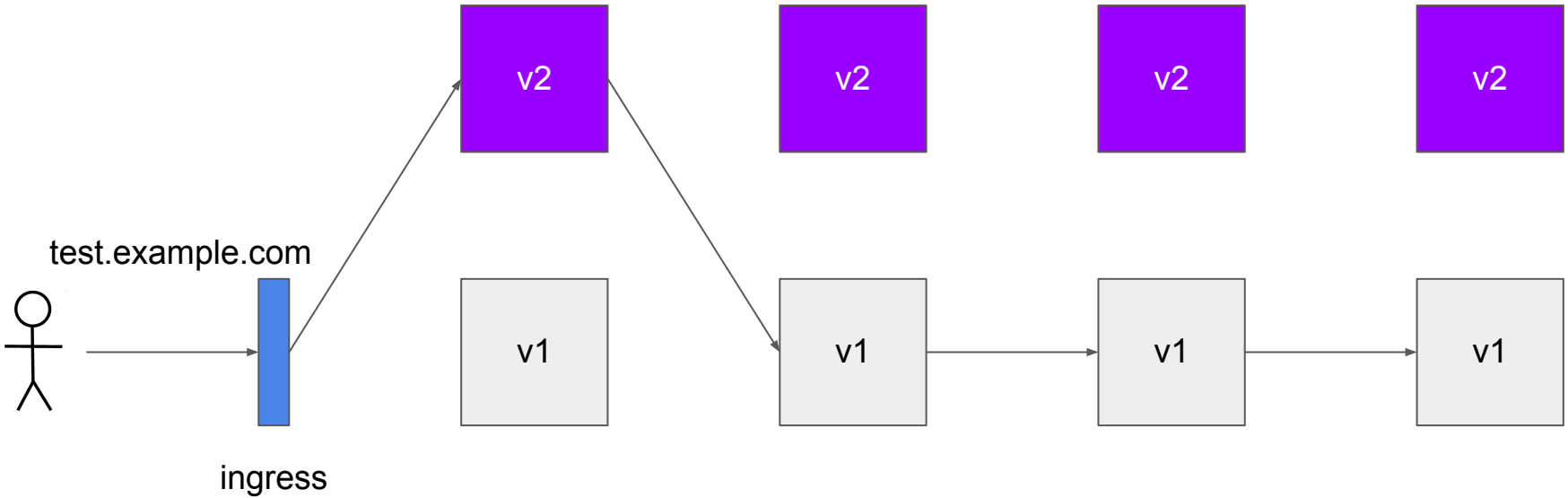
Per-Service Overrides



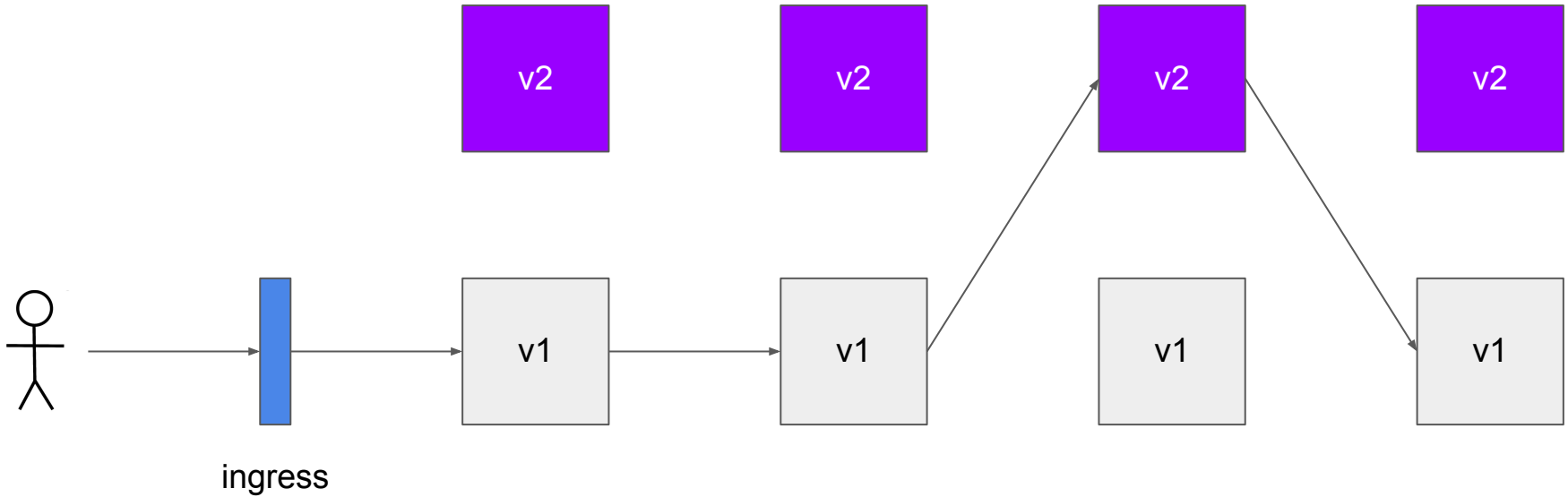
Per-Service Overrides



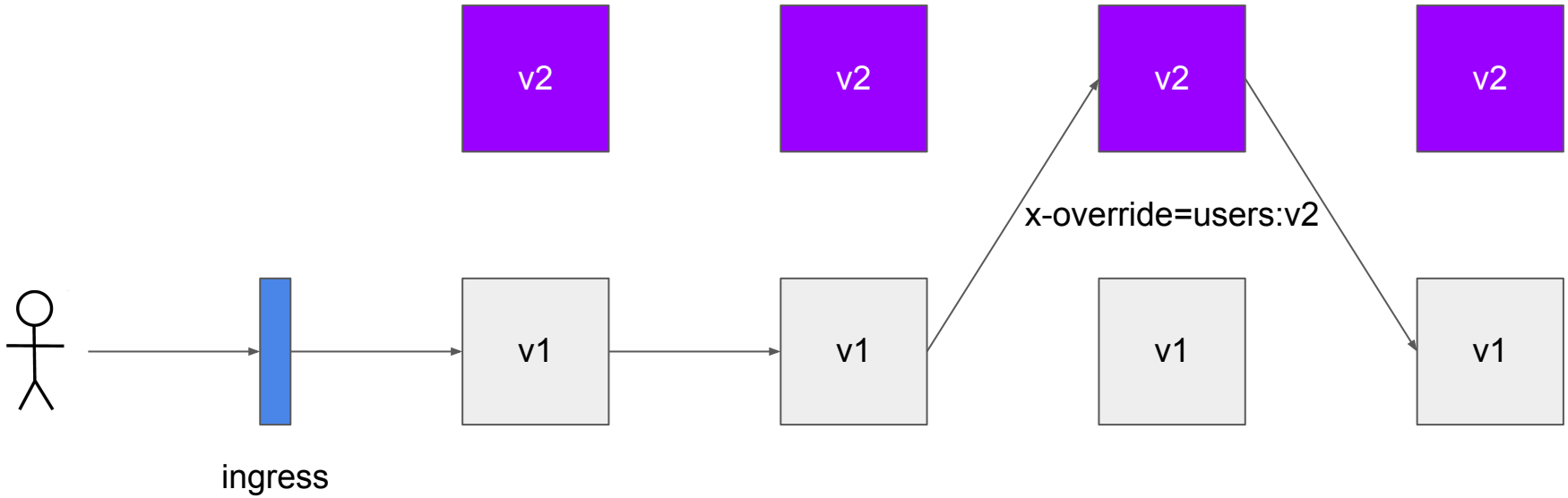
Per-Service Overrides



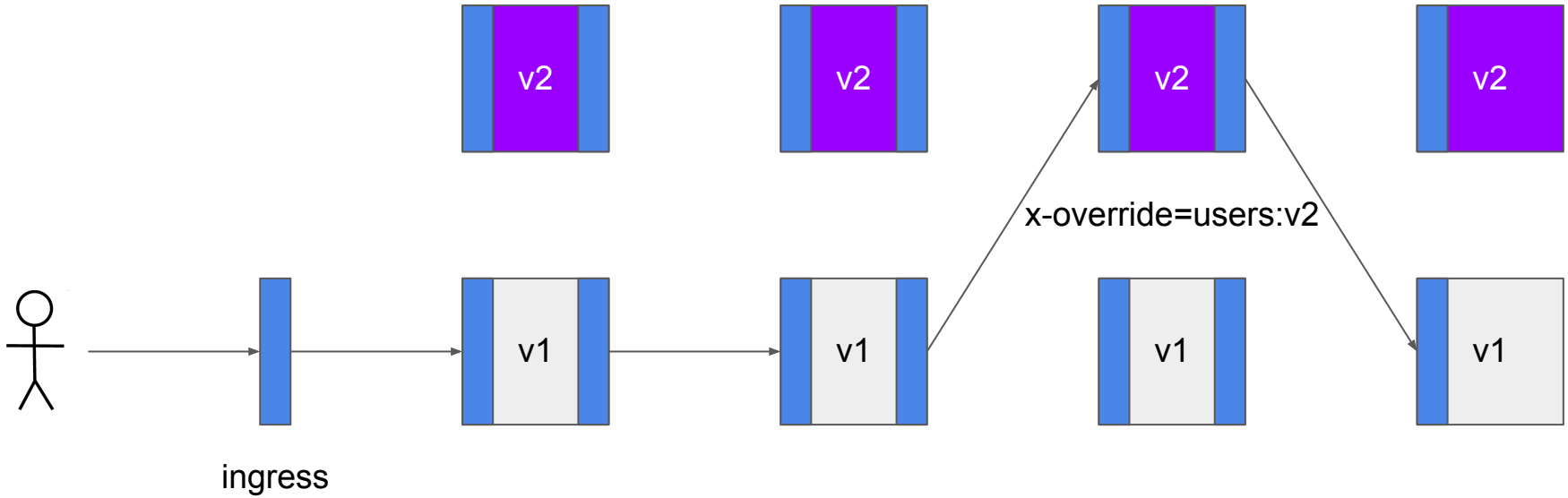
Per-Service Overrides



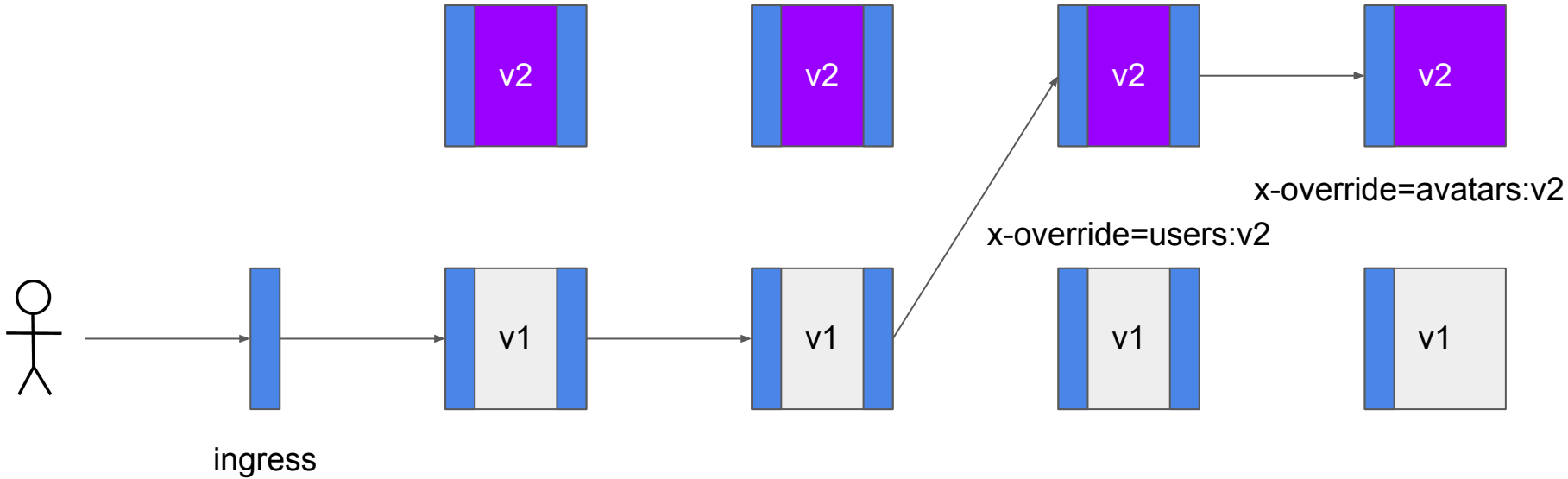
Per-Service Overrides



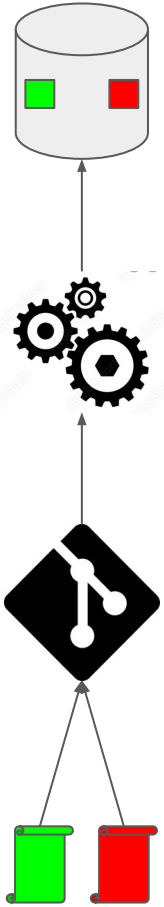
Per-Service Overrides

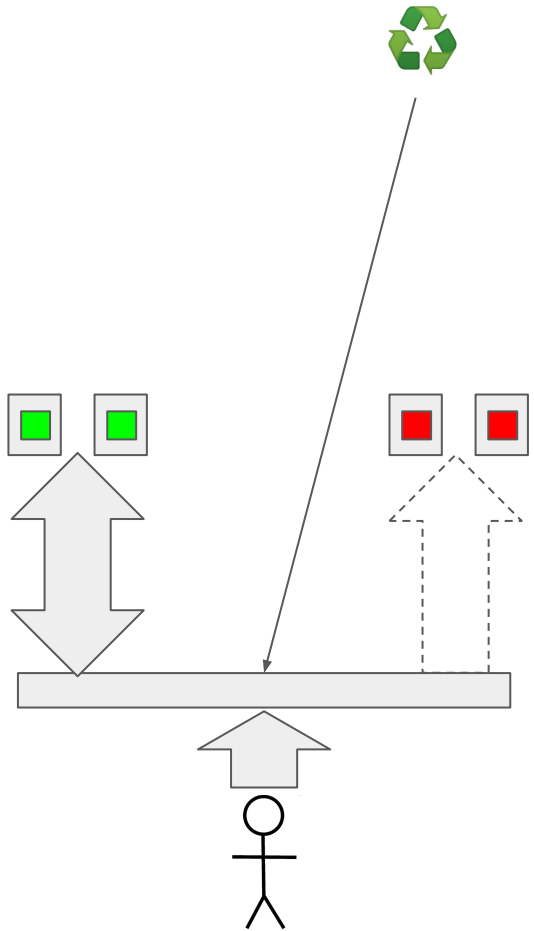


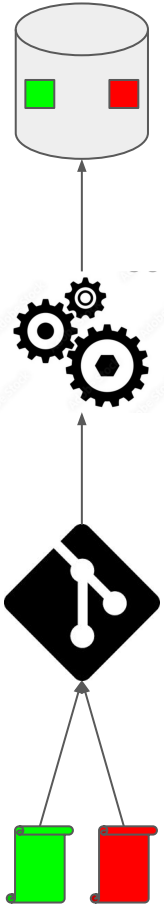
Per-Service Overrides

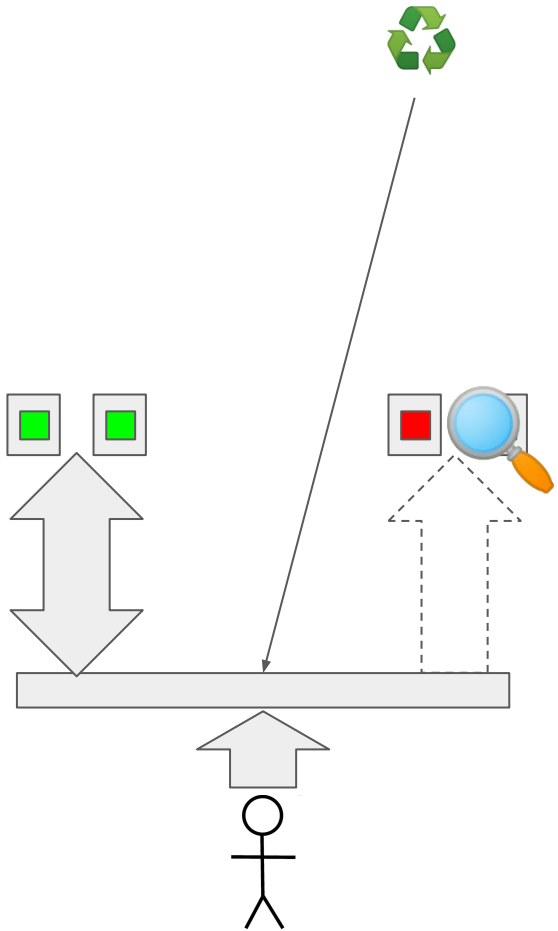


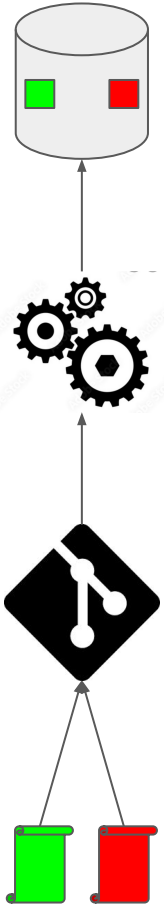
 It’s fiddly 

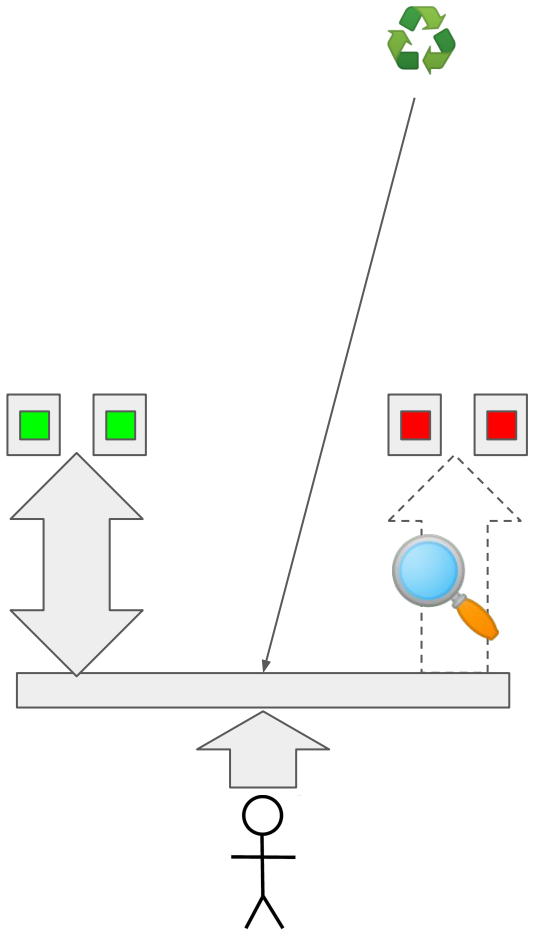


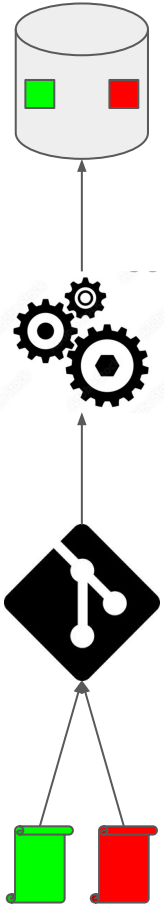


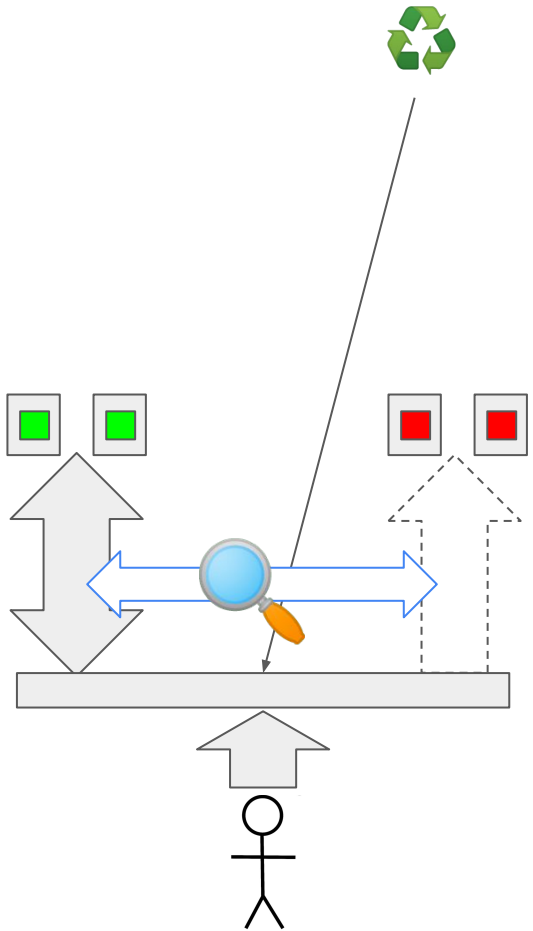








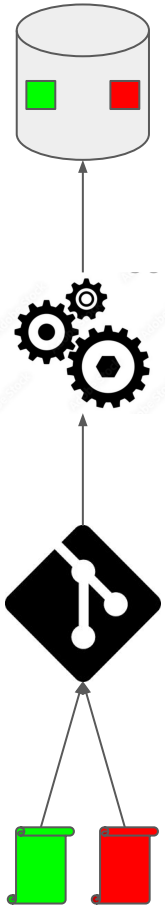


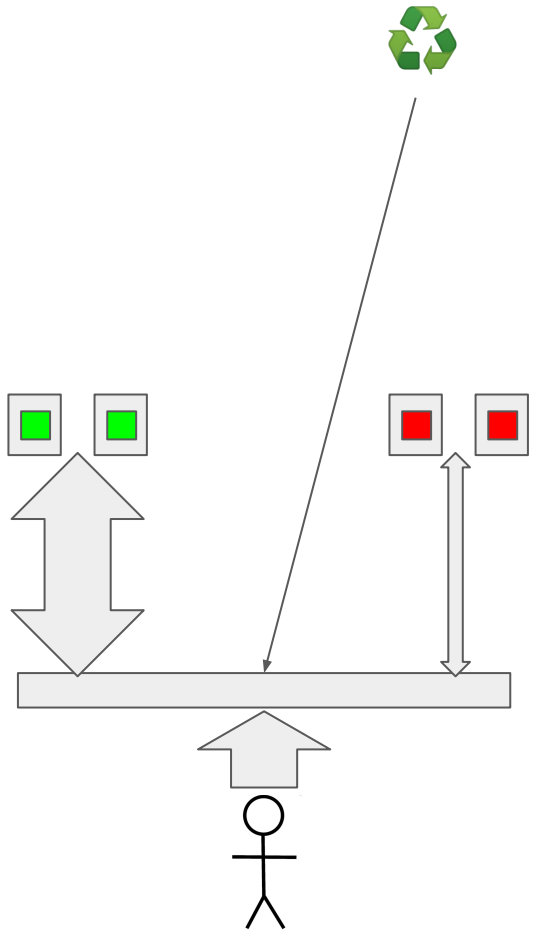


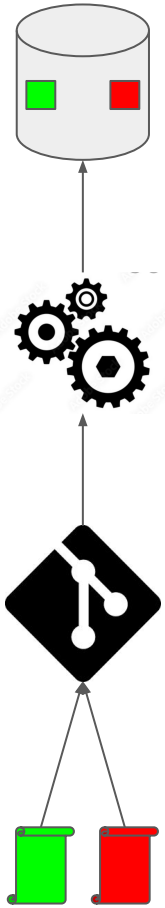
Stage: Running in Prod, Invisible to Users

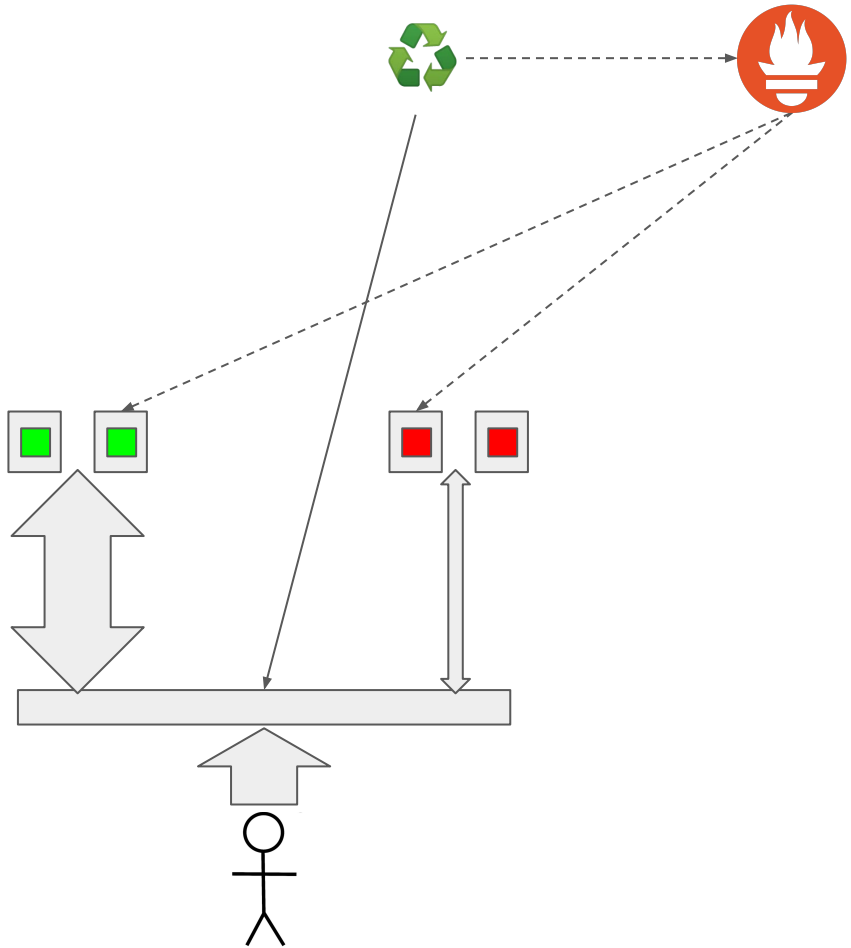
- Gets a mirror of user traffic, but responses dropped
- What’s its Service Level? - crash rate, error rate, performance
- Compare results, if helpful

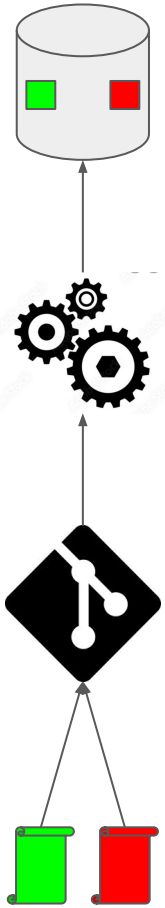
Release

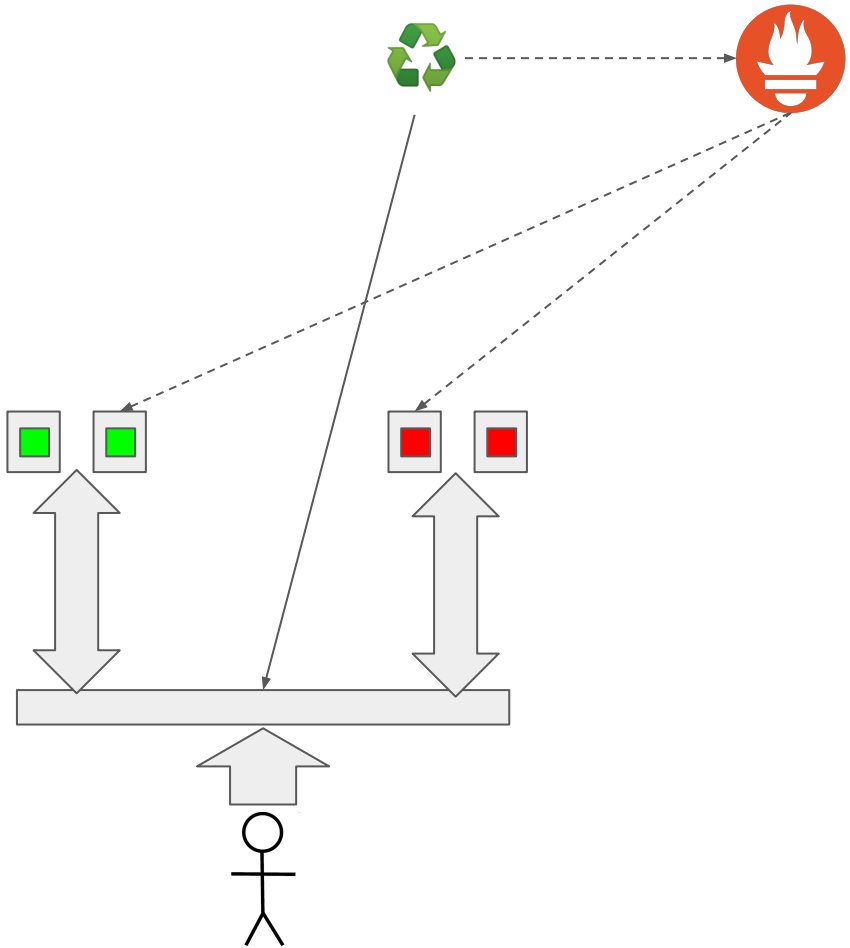


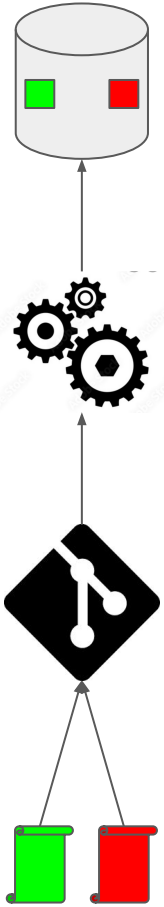


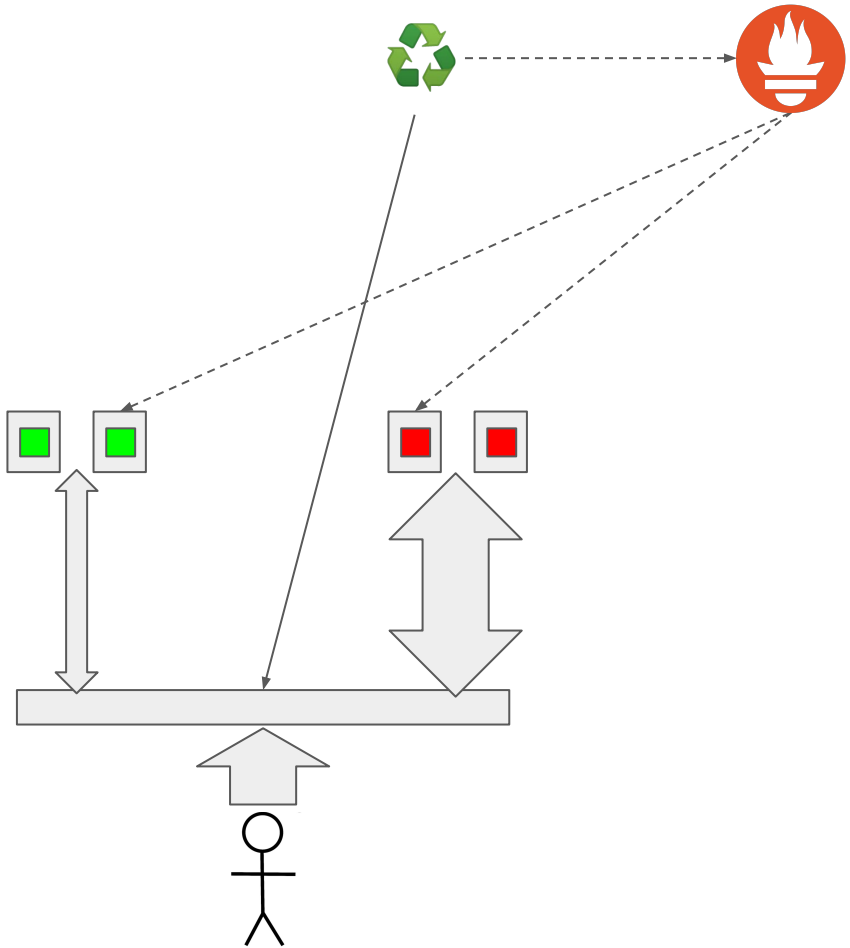


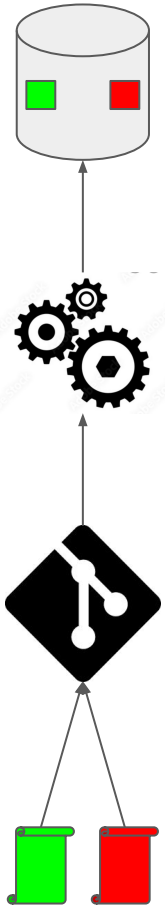


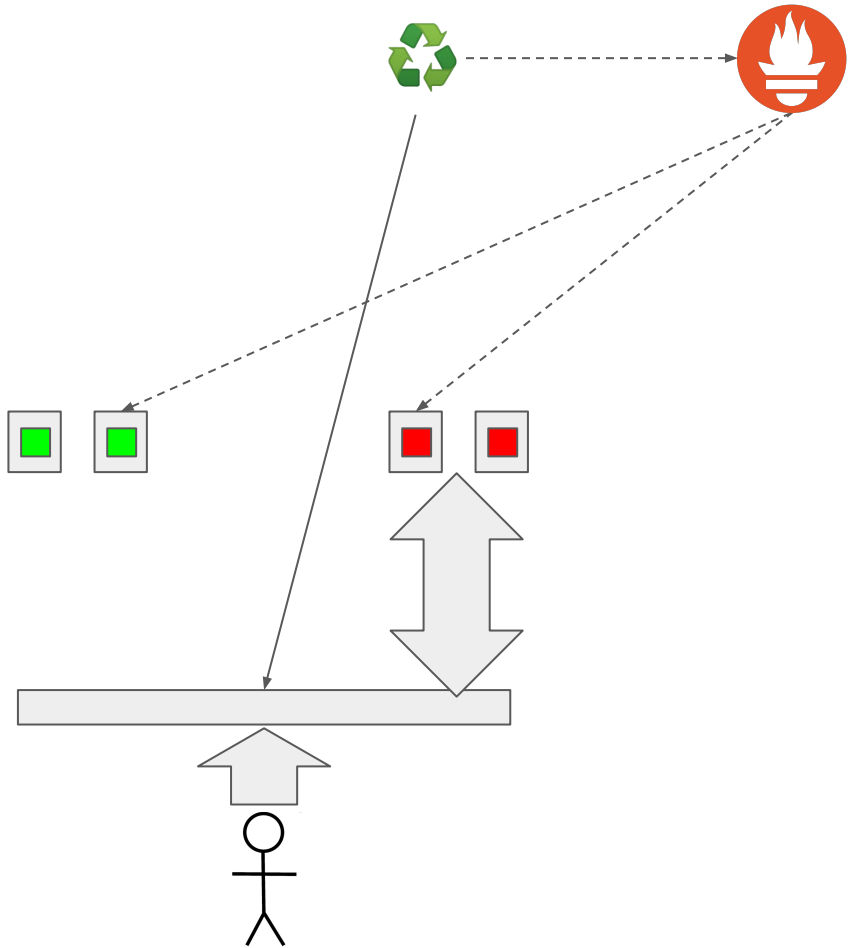




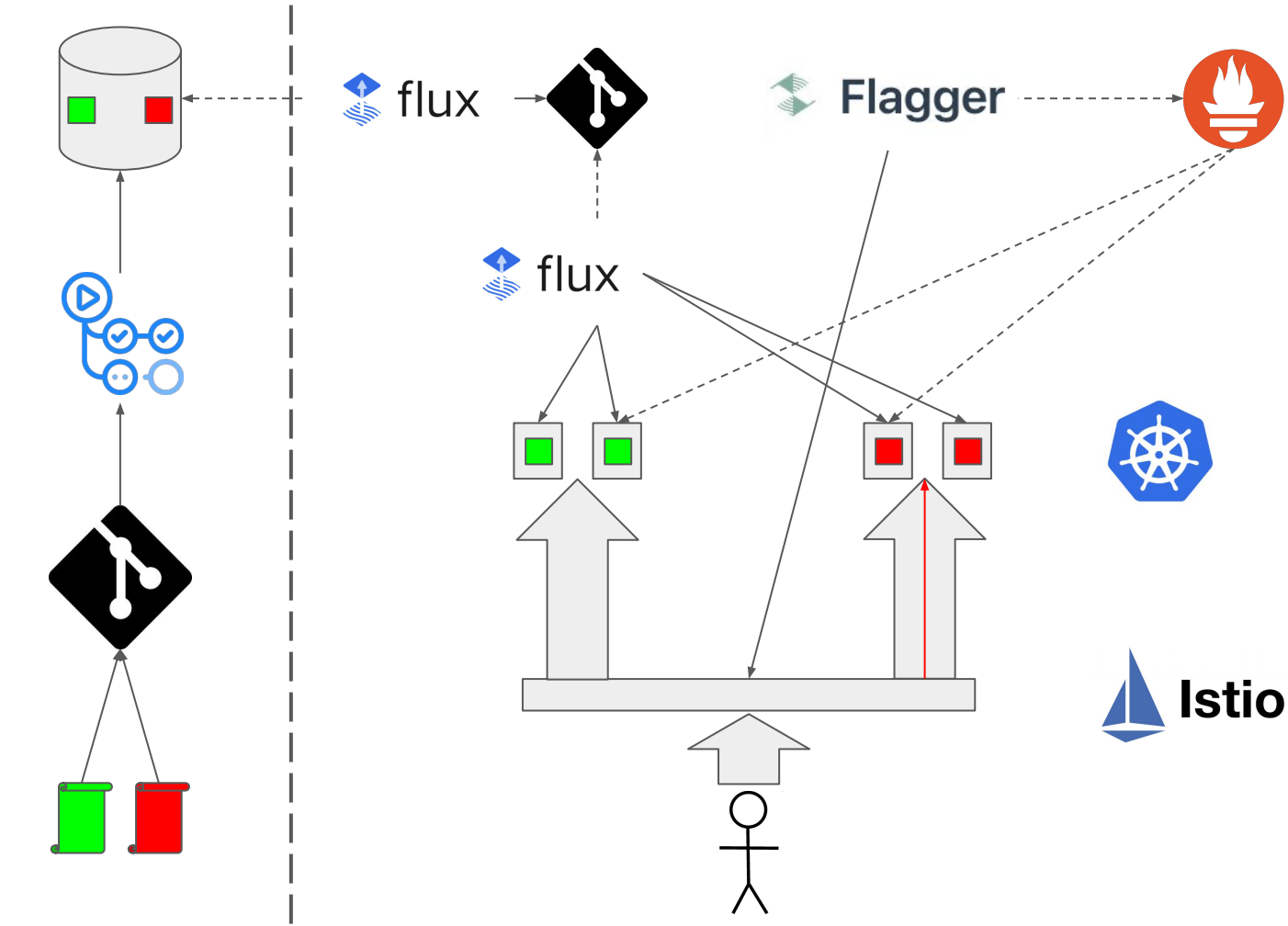


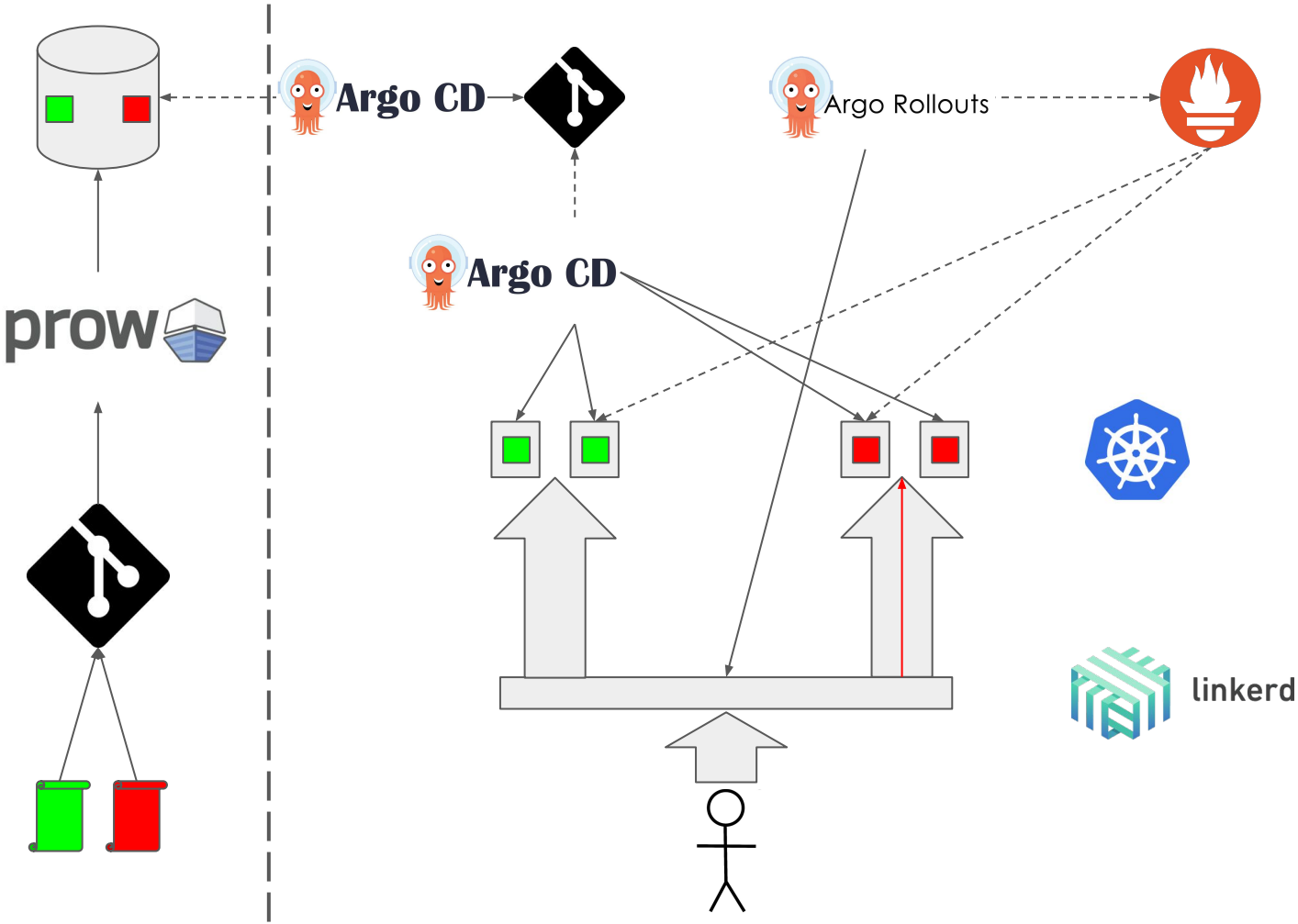






Demo Time!


flux
Flagger
flux
Istio


Argo CD
Argo Rollouts
Argo CD
prow
linkerd

Progressive Roll-Out

- Sends 1% of user traffic to new version
- Monitor all SLIs for a period of time
- If it’s within the SLOs, add 1% more traffic

Roll-back

- If it fails SLO at any point, all traffic sent back to the old version
- New version left running for inspection
- Alert raised

“I test in Prod”

- Charity Majors

GOTO
Guide

★★★★★

Remember to **rate this session**

THANK YOU!

Download on the App Store

GET IT ON Google Play

#GOTOams