PROGRESSIVE DELIVERY IN KUBERNETES

Carlos Sanchez / csanchez.org / @csanchez

Watch online at carlossg.github.io/presentations

Principal Scientist

Adobe Experience Manager Cloud Service

Long time OSS contributor at Jenkins, Apache Maven, Puppet,…

PROGRESSIVE DELIVERY

Progressive Delivery, a History.... Condensed

By Adam Zimman - August 6, 2018

RedMonk

the developer-focused industry analyst firm

JAMES GOVERNOR'S MONKCHIPS

Towards Progressive Delivery

By James Governor | @monkchips | August 6, 2018

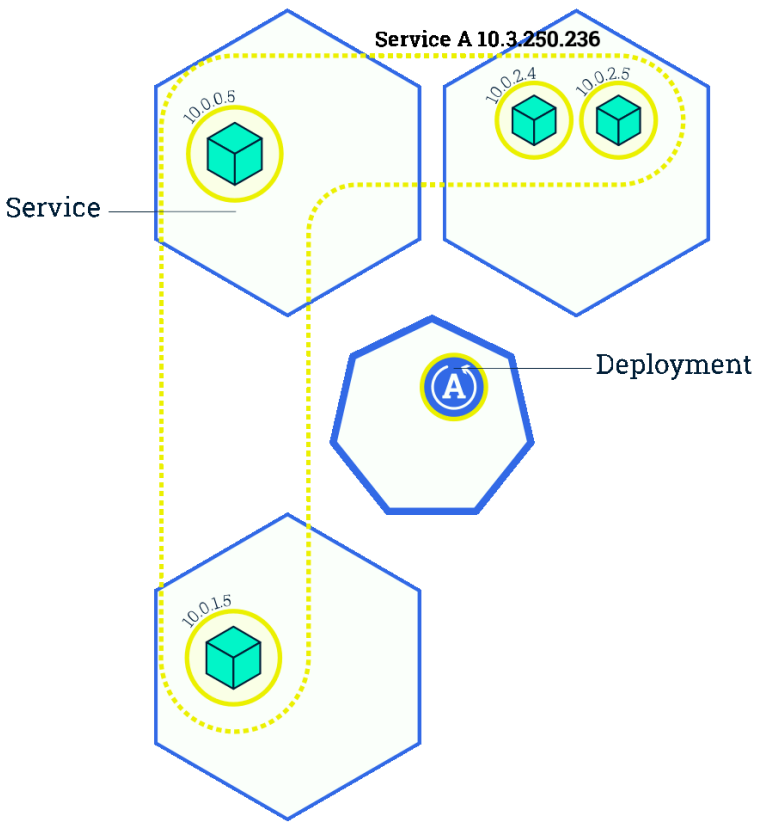
Progressive Delivery is a term that includes deployment strategies that try to avoid the pitfalls of all-or-nothing deployment strategies

New versions being deployed do not replace existing versions but run in parallel for an amount of time receiving live production traffic, and are evaluated in terms of correctness and performance before the rollout is considered successful.

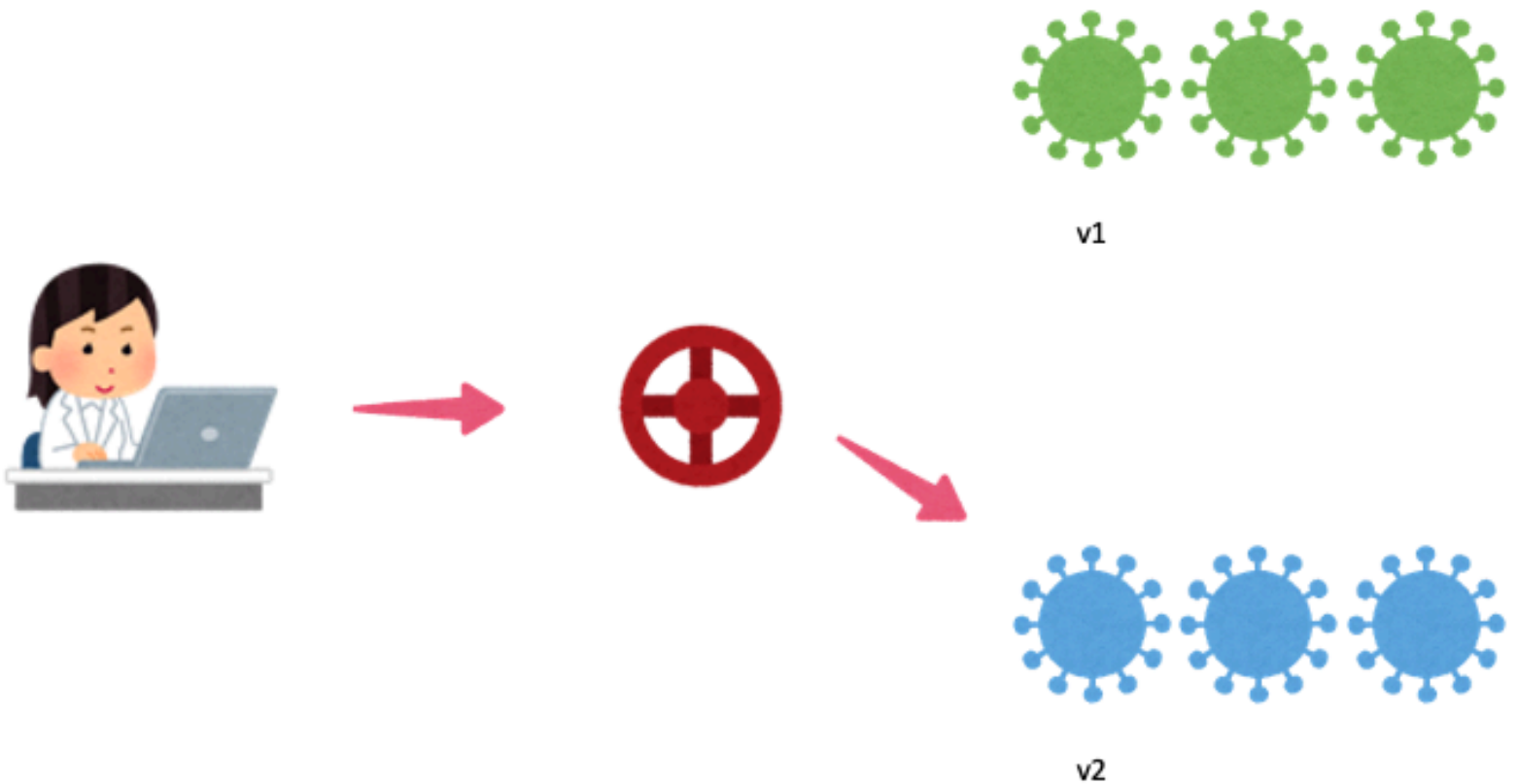
- Avoiding downtime
- Limit the blast radius
- Shorter time from idea to production

PROGRESSIVE DELIVERY TECHNIQUES

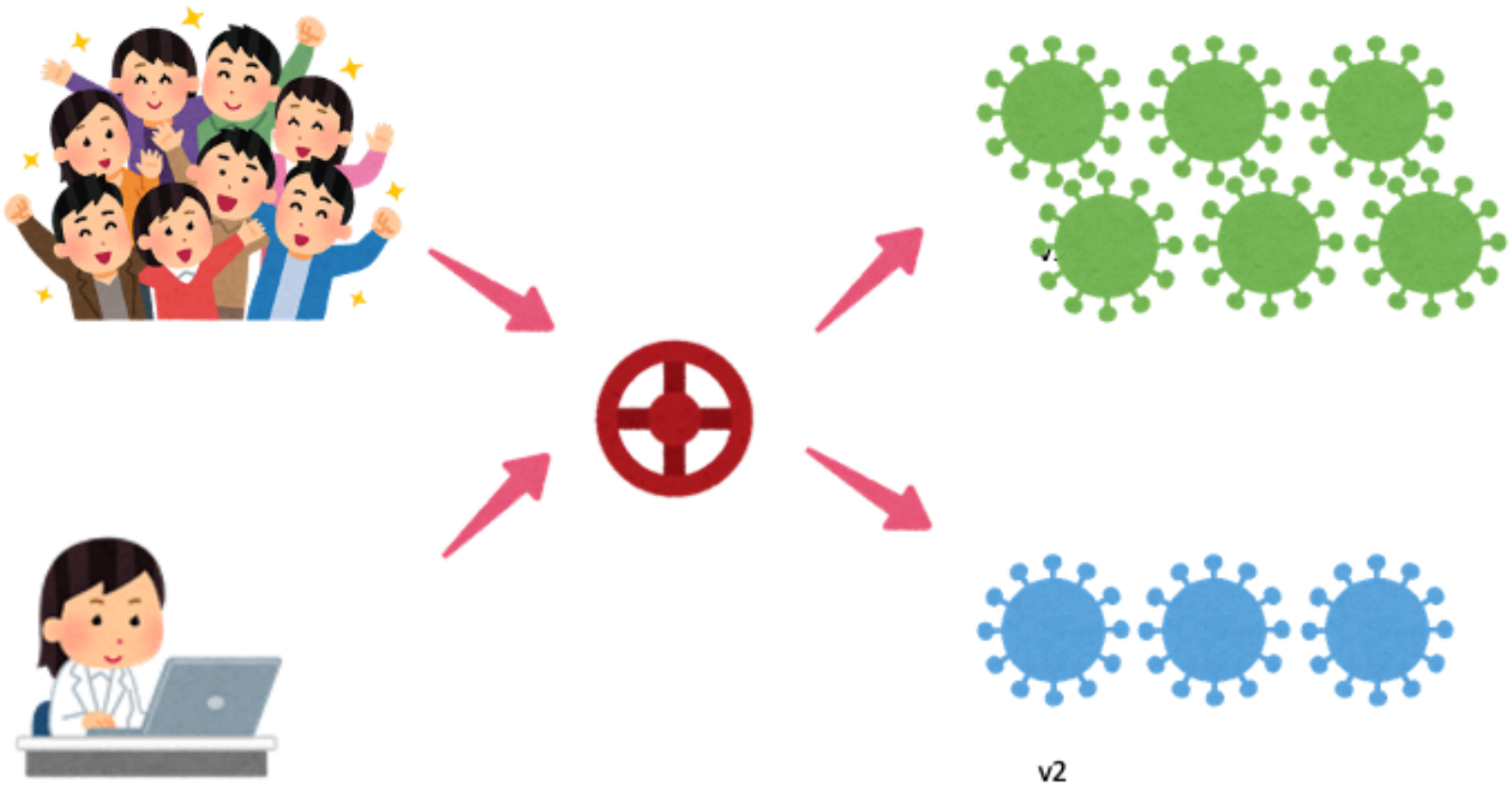
ROLLING UPDATES



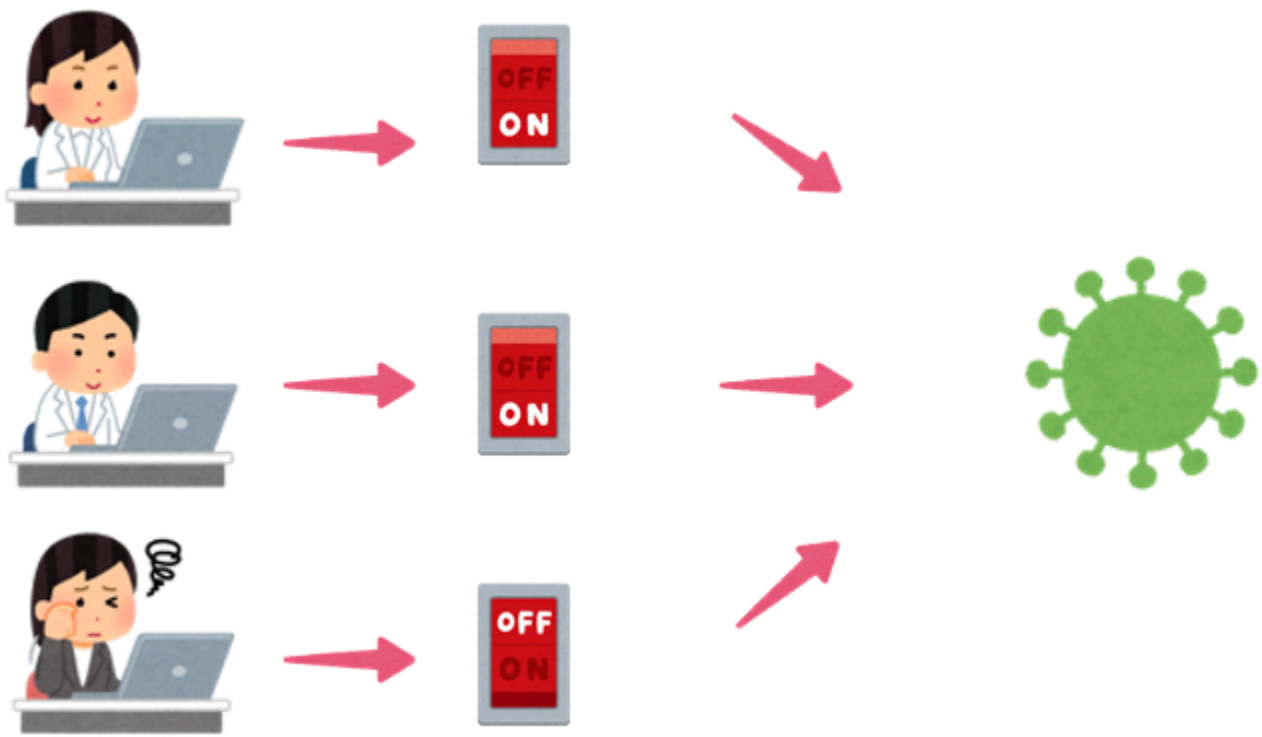
BLUE-GREEN DEPLOYMENT



CANARY DEPLOYMENT



FEATURE FLAGS



MONITORING IS THE NEW TESTING

Know when users are experiencing issues **in production**

React to the issues **automatically**

@DEVOPS_BORAT

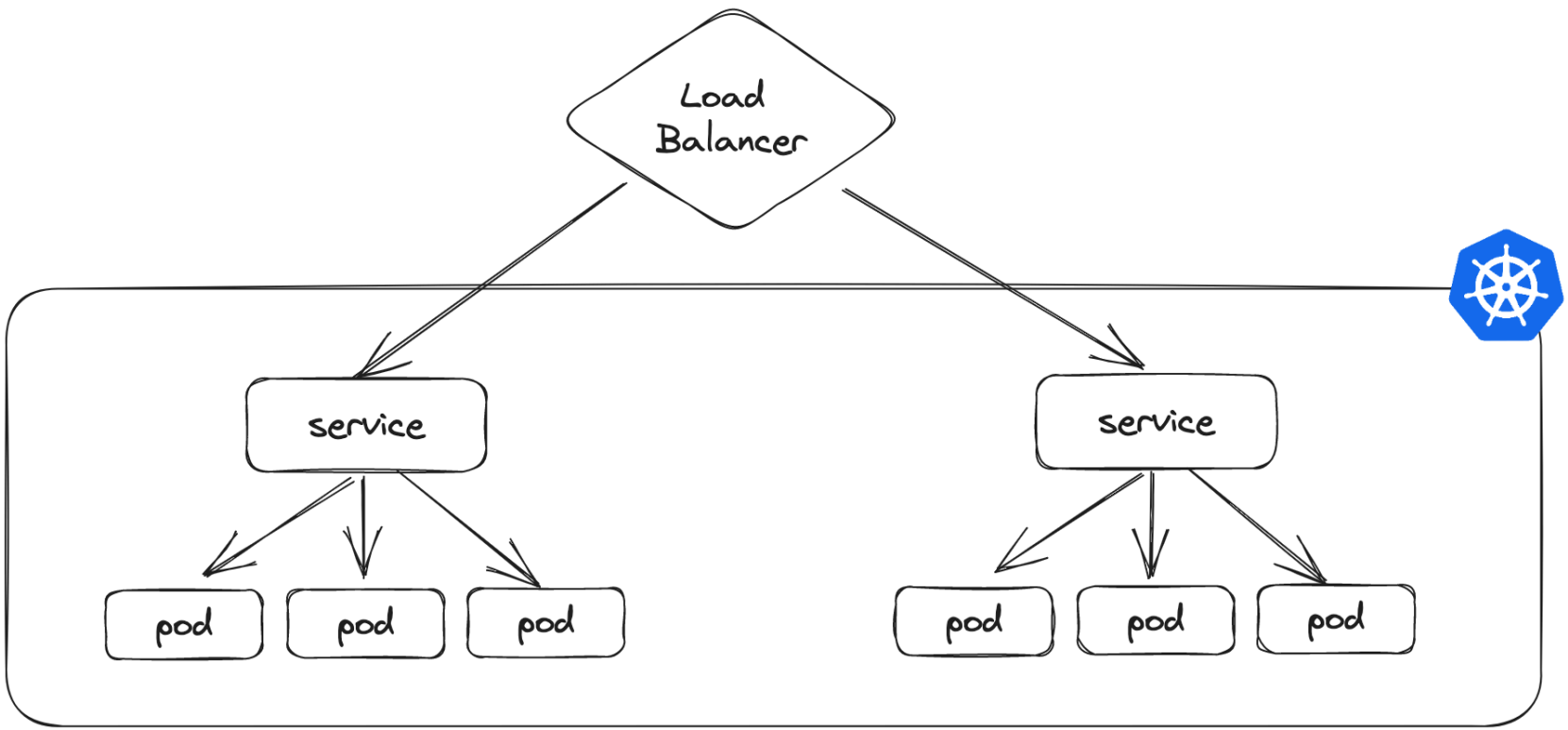
DevOps Borat

To make error is human. To propagate error to all server in automatic way is #devops.

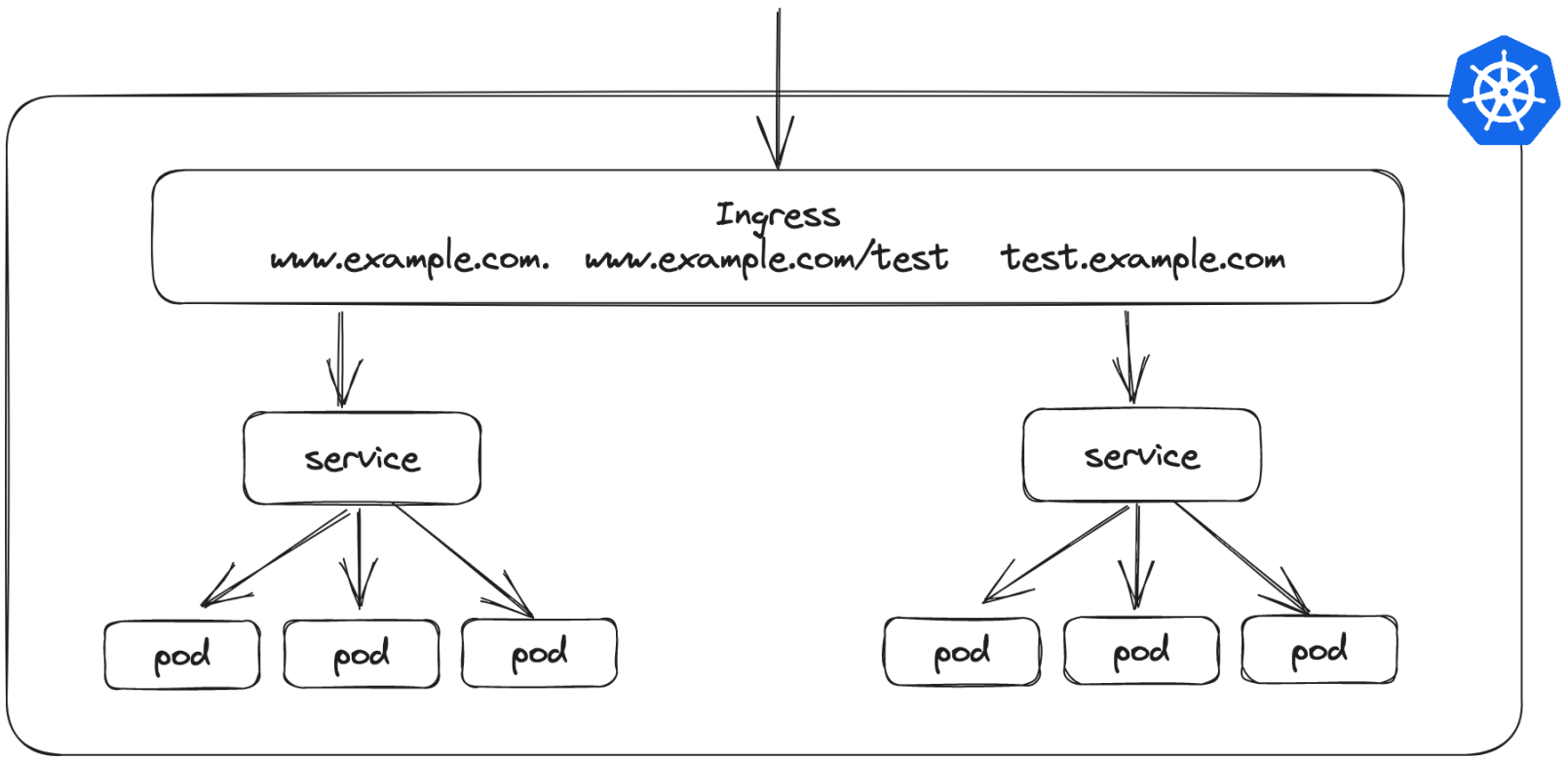
If you haven't automatically destroyed something by mistake, you are not automating enough

PROGRESSIVE DELIVERY IN KUBERNETES

KUBERNETES SERVICE ARCHITECTURE



KUBERNETES INGRESS ARCHITECTURE



KUBERNETES INGRESS

Ingress controllers:

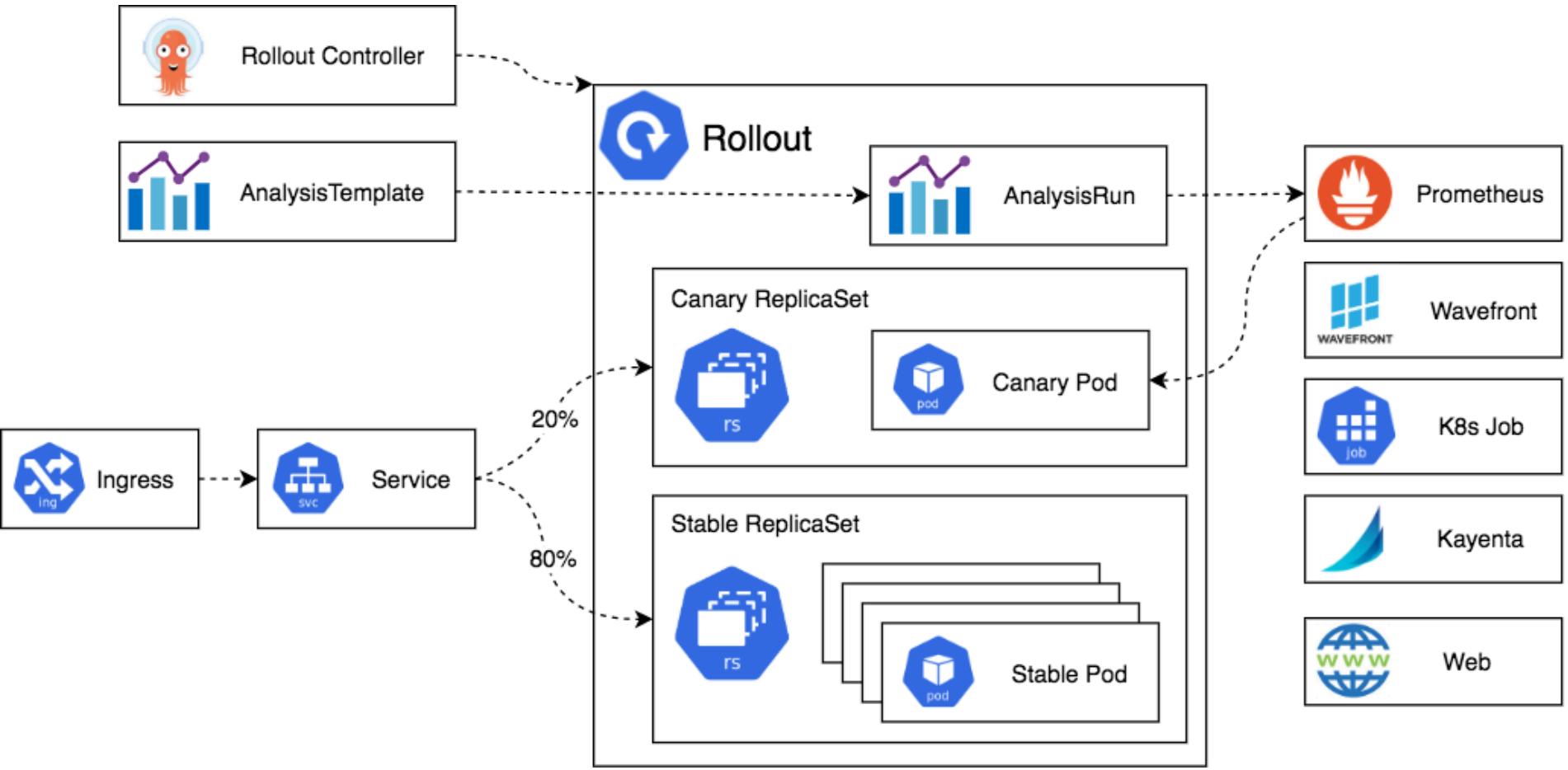
- AWS
- GCE
- nginx
- Ambassador
- Istio Ingress
- Traefik
- HAProxy
- ...

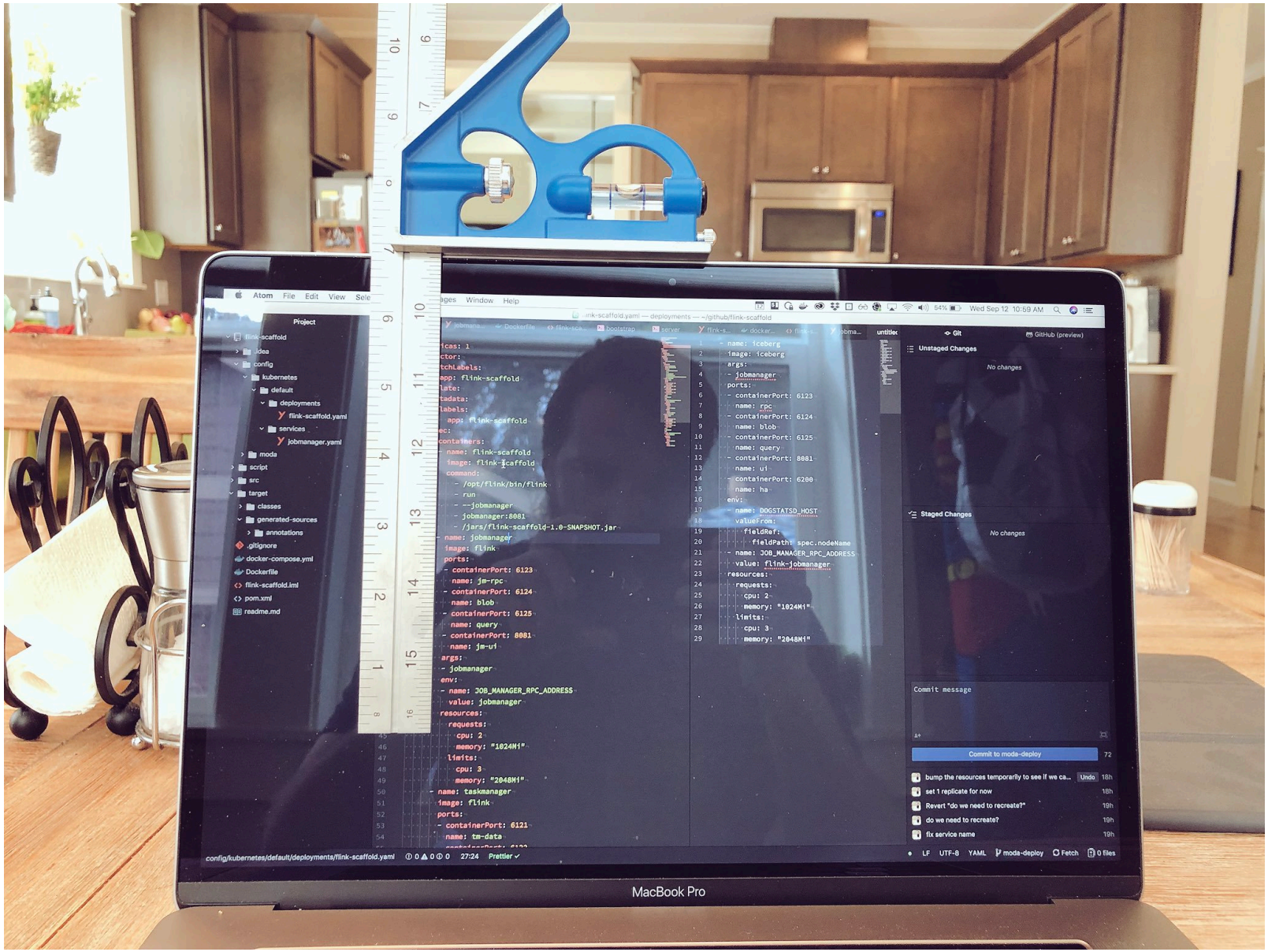
ARGO ROLLOUTS

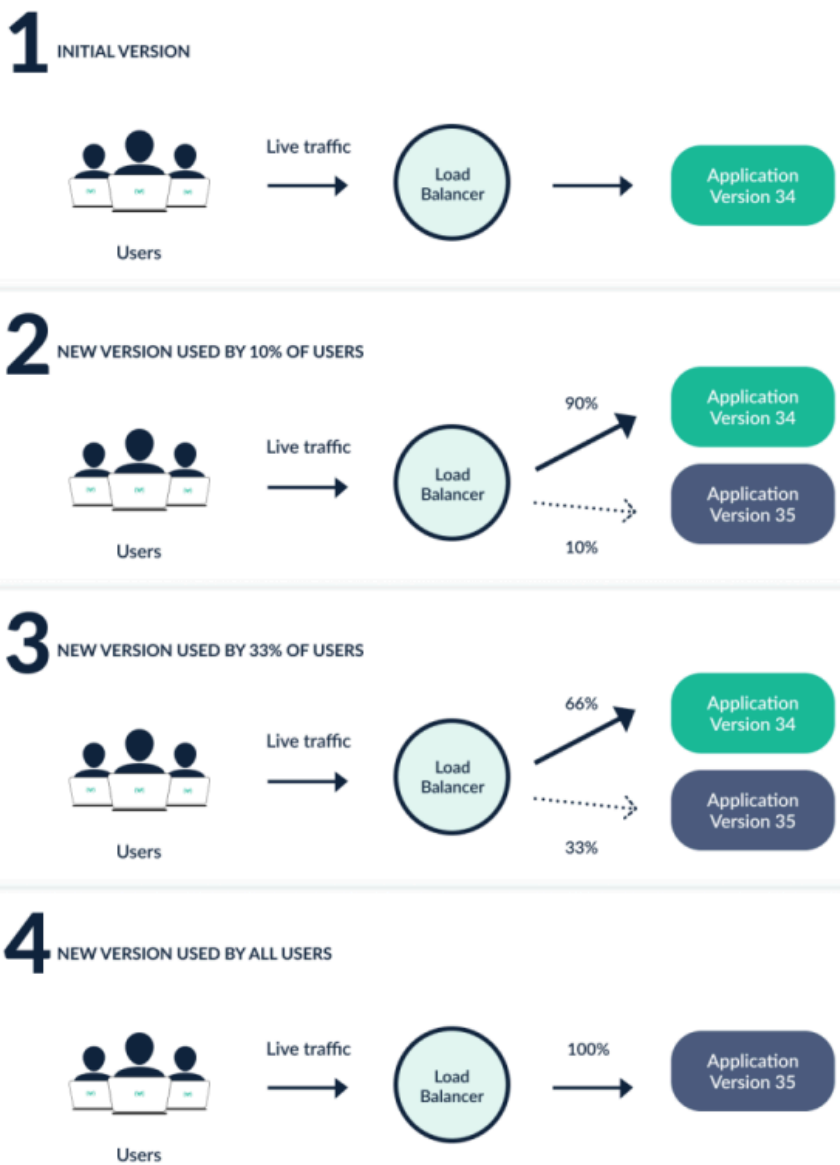
ARGO ROLLOUTS

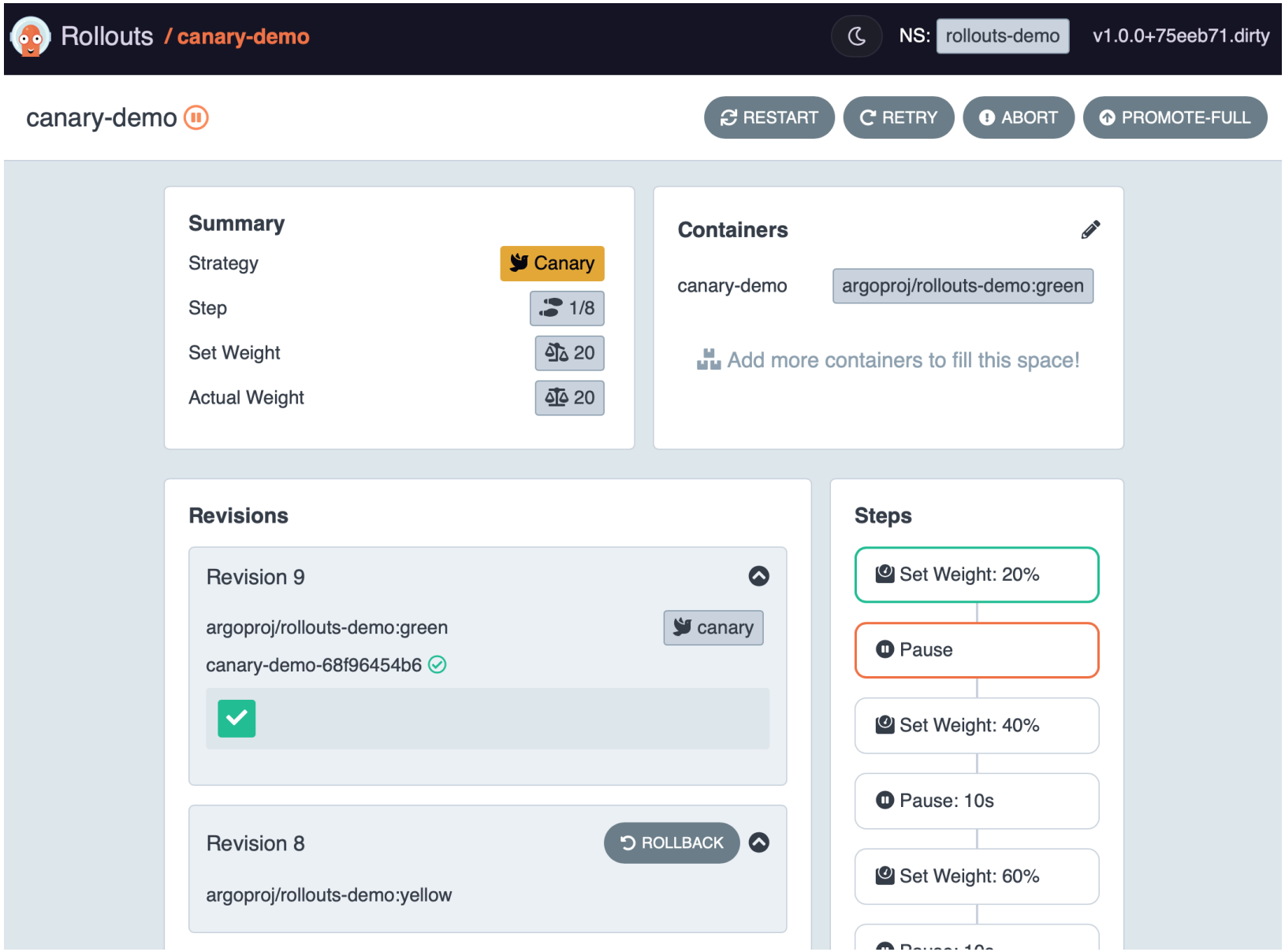
provides advanced deployment capabilities such as blue-green, canary, canary analysis, experimentation, and progressive delivery features to Kubernetes.

ARGO ROLLOUTS




Project
flink-scaffold
.idea
config
kubernetes
default
deployments
flink-scaffold.yaml
services
jobmanager.yaml
moda
script
src
target
classes
generated-sources
annotations
.gitignore
docker-compose.yml
Dockerfile
flink-scaffold.iml
pom.xml
readme.md
Unstaged Changes
No changes
Staged Changes
No changes
Commit message
Commit to moda-deploy
bump the resources temporarily to see if we ca...
Undo
set 1 replicate for now
Revert "do we need to recreate?"
do we need to recreate?
fix service name
MacBook Pro


1
INITIAL VERSION
Users
Live traffic
Load Balancer
Application Version 34
2
NEW VERSION USED BY 10% OF USERS
Users
Live traffic
Load Balancer
90%
10%
Application Version 34
Application Version 35
3
NEW VERSION USED BY 33% OF USERS
Users
Live traffic
Load Balancer
66%
33%
Application Version 34
Application Version 35
4
NEW VERSION USED BY ALL USERS
Users
Live traffic
Load Balancer
100%
Application Version 35


Rollouts / canary-demo
NS: rollouts-demo
v1.0.0+75eeb71.dirty
canary-demo
RESTART
RETRY
ABORT
PROMOTE-FULL
Summary
Strategy
Canary
Step
1/8
Set Weight
20
Actual Weight
20
Containers
canary-demo
argoproj/rollouts-demo:green
Add more containers to fill this space!
Revisions
Revision 9
argoproj/rollouts-demo:green
canary
canary-demo-68f96454b6
Revision 8
ROLLBACK
argoproj/rollouts-demo:yellow
Steps
Set Weight: 20%
Pause
Set Weight: 40%
Pause: 10s
Set Weight: 60%

csanchez.org

csanchez

carlossg