

PROGRESSIVE DELIVERY IN KUBERNETES



Carlos Sanchez / csanchez.org

[@csanchez](#) / @csanchez@fosstodon.org

Principal Scientist

Adobe Experience Manager Cloud Service

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



PROGRESSIVE DELIVERY



[Home](#) [Feature Management](#) [DevOps](#) [Continuous Delivery](#) [To Be Continuous](#)



[Home](#) > [Continuous Delivery](#)

Progressive Delivery, a History.... Condensed

By Adam Zimman - August 6, 2018

👁 4326



the developer-focused industry analyst firm

[Videos](#)

[Research](#)

[Events](#)

[About](#)

[Team](#)

[Services](#)

[Clients](#)

[Contact](#)

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.

Continuous Delivery is hard

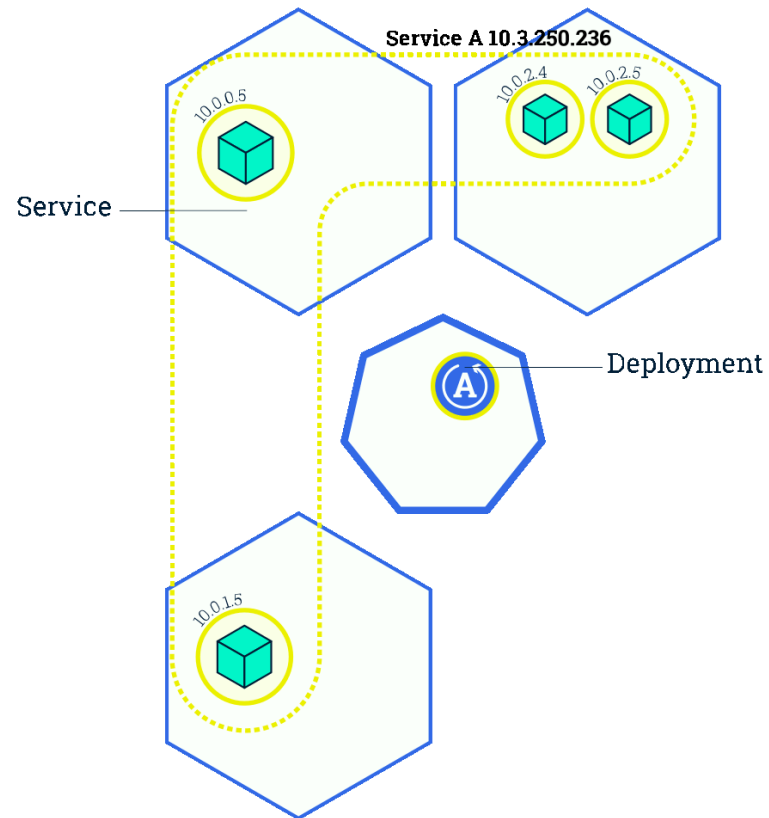
Progressive Delivery makes Continuous Delivery easier
to adopt

reduces the risk associated with Continuous Delivery

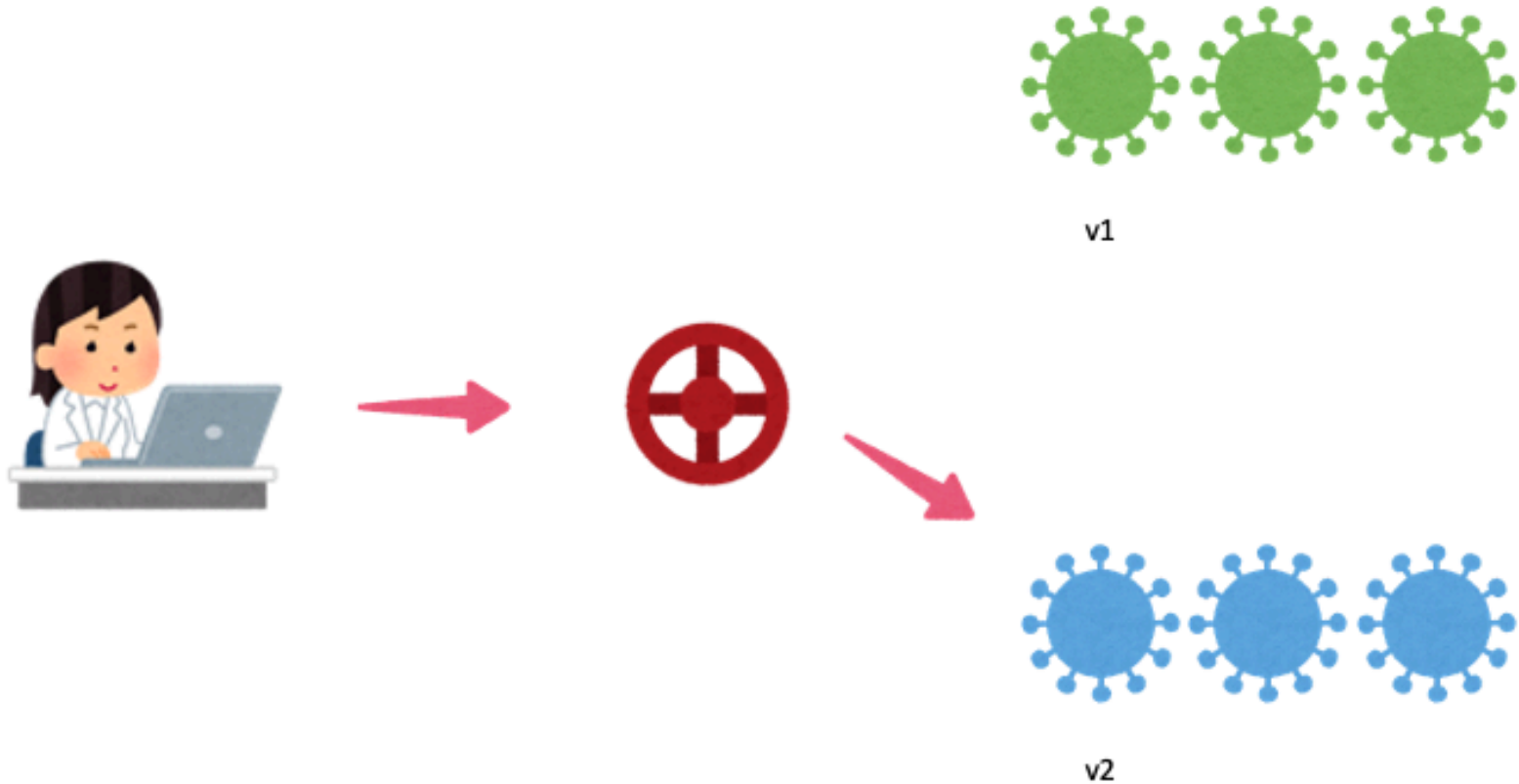
- Avoiding downtime
- Limit the blast radius
- Shorter time from idea to production

PROGRESSIVE DELIVERY TECHNIQUES

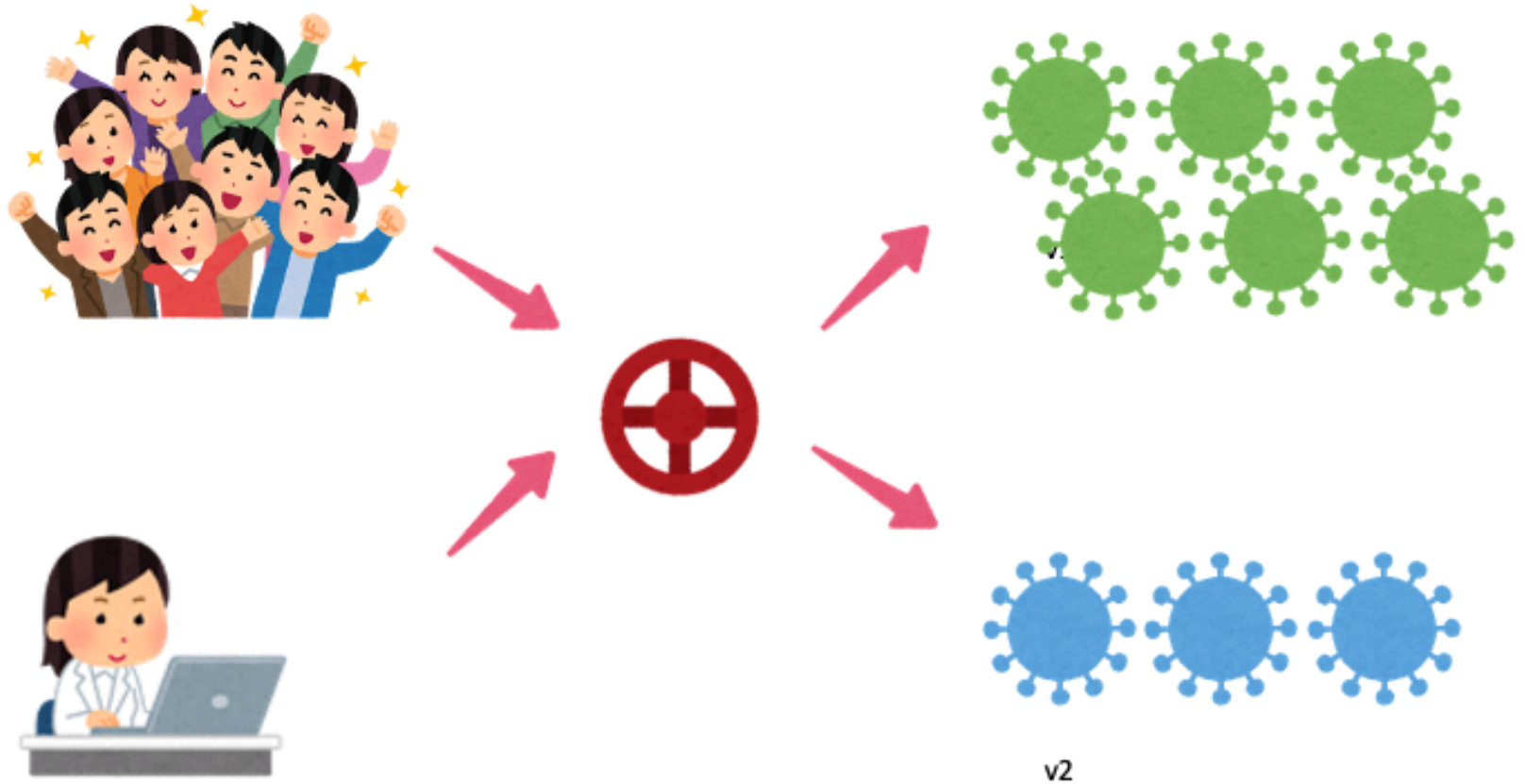
ROLLING UPDATES



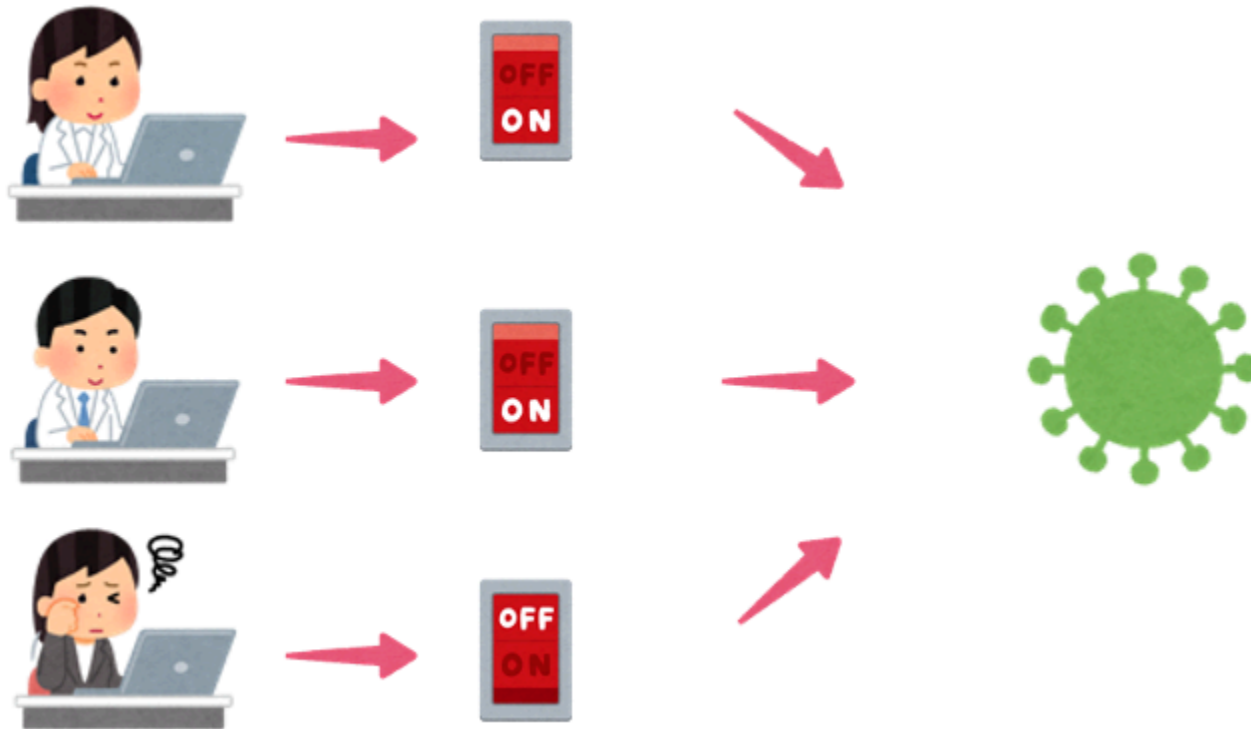
BLUE-GREEN DEPLOYMENT



CANARY DEPLOYMENT



FEATURE FLAGS



MONITORING IS THE NEW TESTING

Know when users are experiencing issues in
production

React to the issues **automatically**

Progressive Delivery requires a good amount of
metrics



@DEVOPS_BORAT

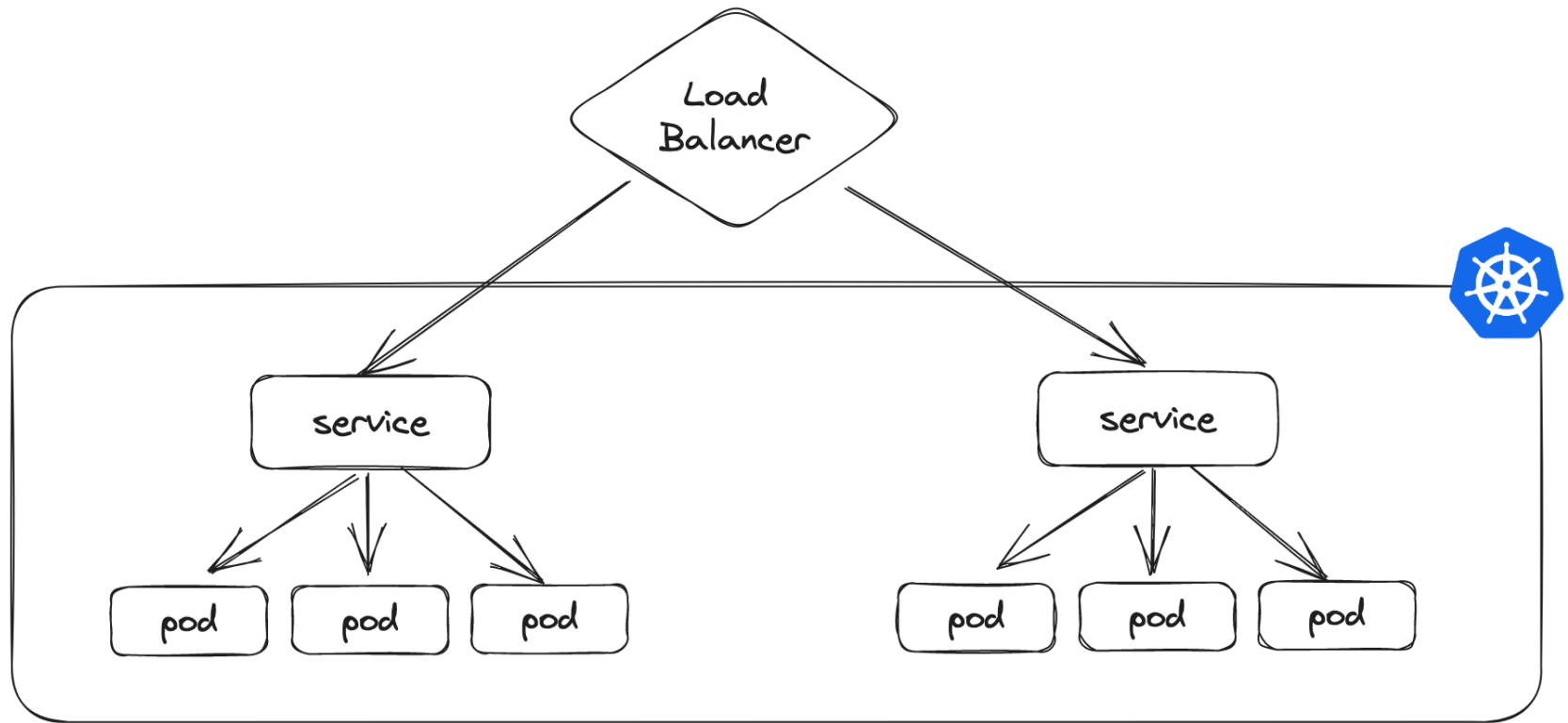
DevOps Borat

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

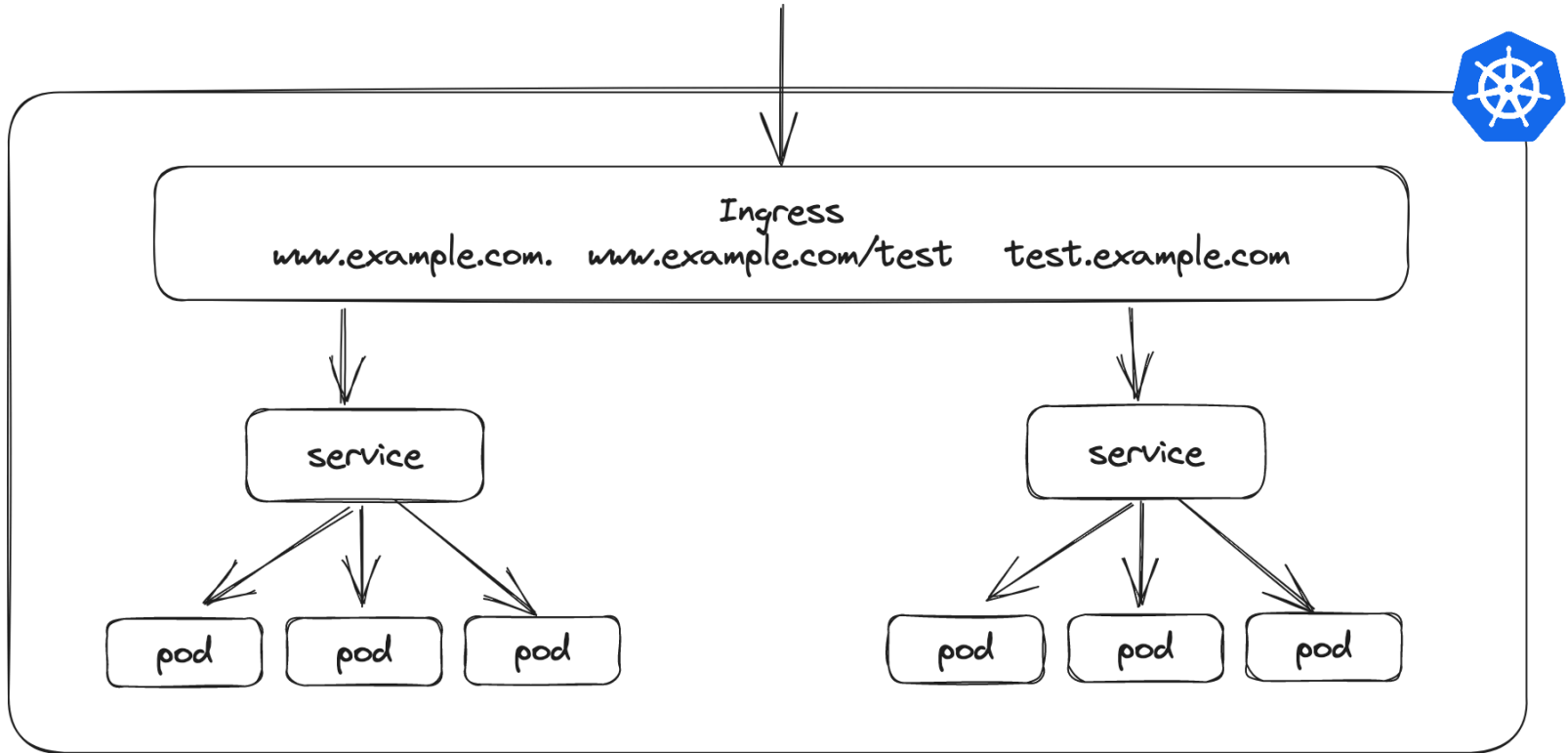
*If you haven't automatically destroyed
something by mistake, you are not
automating enough*

PROGRESSIVE DELIVERY IN KUBERNETES

KUBERNETES SERVICE ARCHITECTURE



KUBERNETES INGRESS ARCHITECTURE

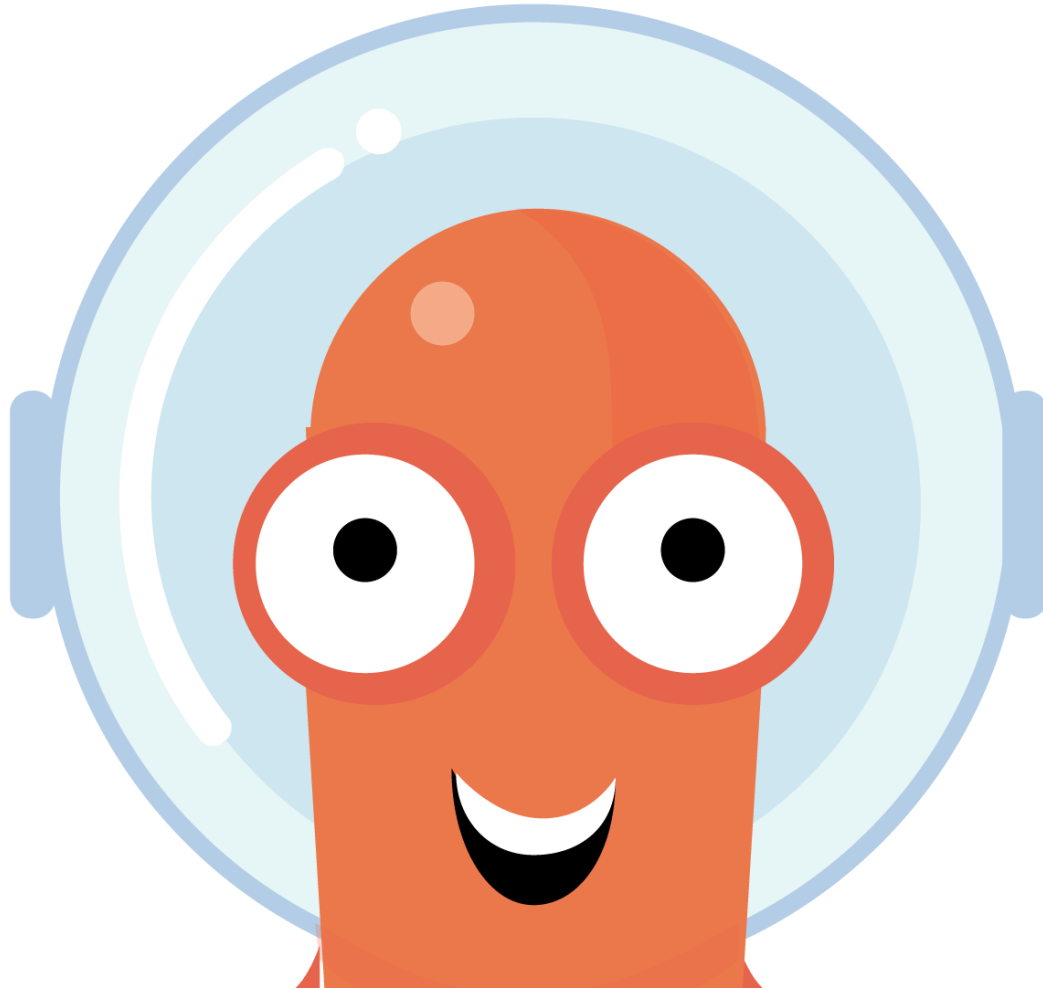


KUBERNETES INGRESS

Ingress controllers:

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

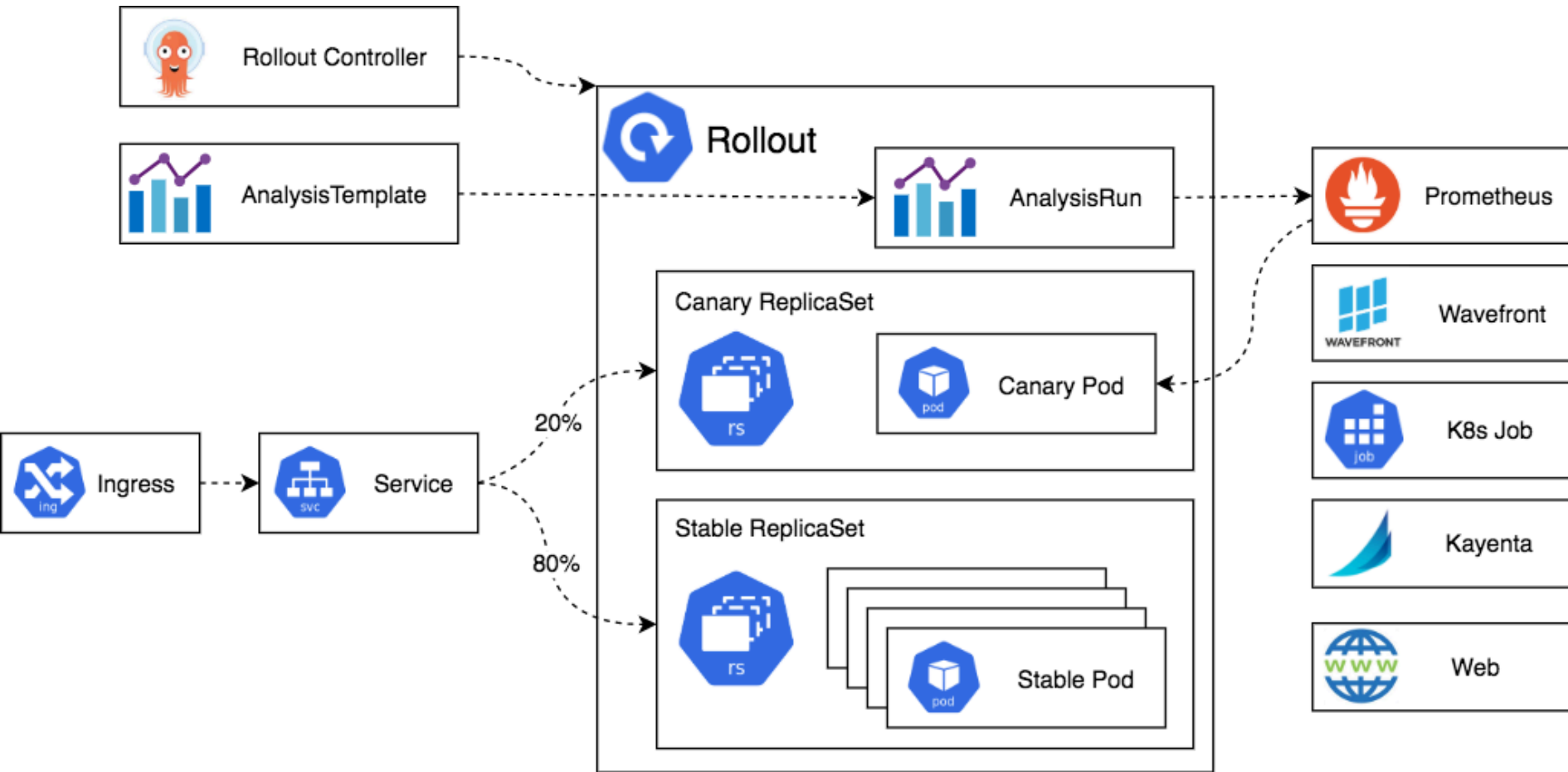
ARGO ROLLOUTS

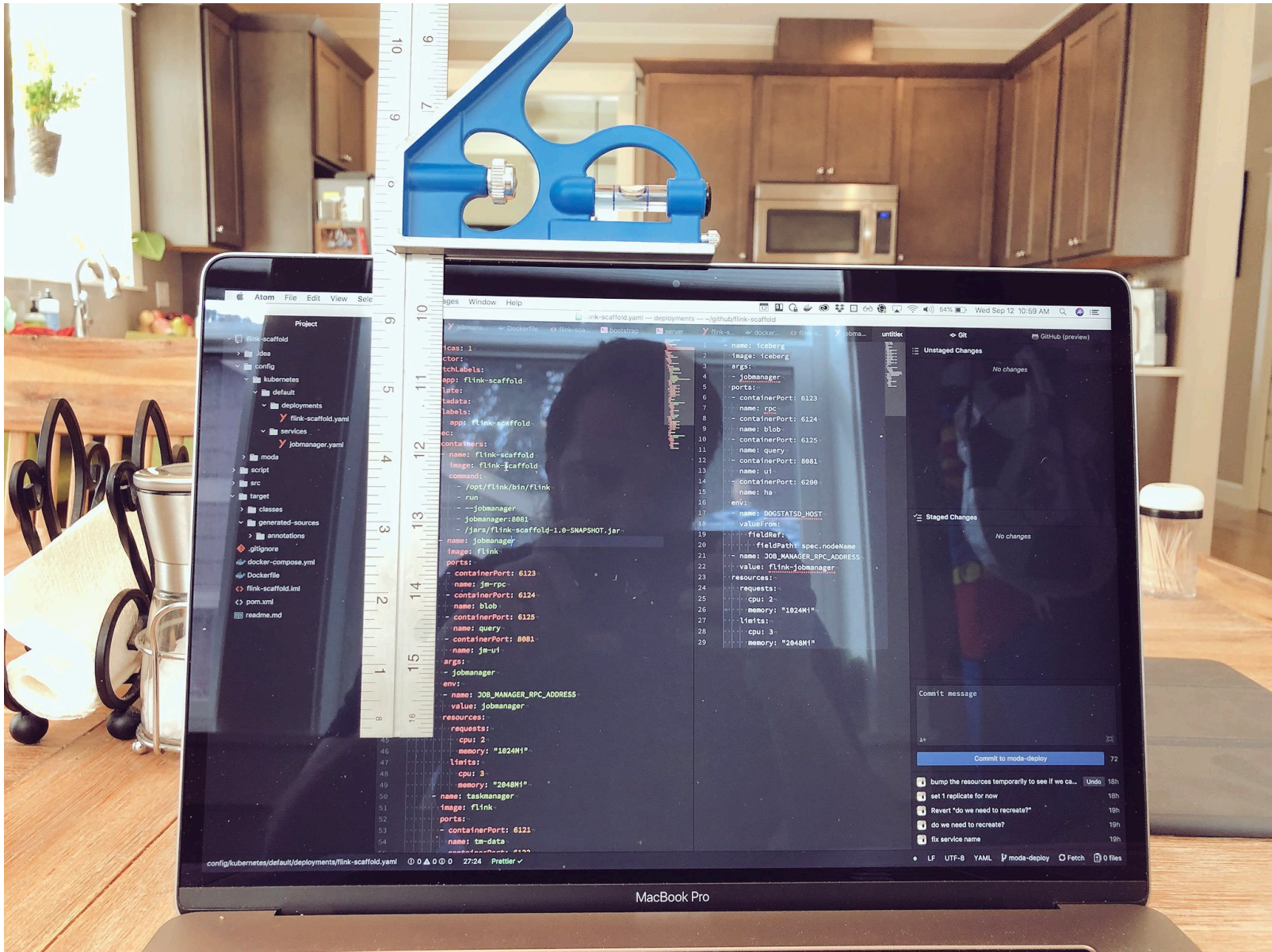


ARGO ROLLOUTS

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

ARGO ROLLOUTS





Atom File Edit View Settings

Project
flink-scaffold
├── .idea
├── config
├── kubernetes
│ ├── default
│ │ ├── deployments
│ │ │ └── flink-scaffold.yaml
│ │ └── services
│ │ └── jobmanager.yaml
├── mode
├── script
├── src
├── target
├── classes
├── generated-sources
├── annotations
├── .gitignore
├── docker-compose.yml
├── Dockerfile
├── flink-scaffold.ini
├── pom.xml
└── readme.md

config/kubernetes/default/deployments/flink-scaffold.yaml

Window Help

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: flink-scaffold
  labels:
    app: flink-scaffold
spec:
  replicas: 1
  selector:
    matchLabels:
      app: flink-scaffold
  template:
    metadata:
      labels:
        app: flink-scaffold
    spec:
      containers:
        - name: flink-scaffold
          image: flink-scaffold
          command:
            - /opt/flink/bin/flink
            - run
            - --jobmanager
            - jobmanager:8081
            - /jars/flink-scaffold-1.0-SNAPSHOT.jar
          name: flink
          ports:
            - containerPort: 6123
              name: job-rpc
            - containerPort: 6124
              name: blob
            - containerPort: 6125
              name: query
            - containerPort: 8081
              name: job-ui
          args:
            - jobmanager
          env:
            - name: JOB_MANAGER_RPC_ADDRESS
              value: jobmanager
          resources:
            requests:
              cpu: 2
              memory: "1024Mi"
            limits:
              cpu: 3
              memory: "2048Mi"
          name: taskmanager
          image: flink
          ports:
            - containerPort: 6121
              name: tm-data
```

```
1  name: iceberg
2  args:
3  - jobmanager
4  ports:
5  - containerPort: 6123
6  name: rpc
7  containerPort: 6124
8  name: blob
9  containerPort: 6125
10 name: query
11 containerPort: 8081
12 name: ui
13 containerPort: 6200
14 name: ha
15 env:
16   name: DOGSTATSD_HOST
17   valueFrom:
18     fieldRef:
19       fieldPath: spec.nodeName
20   name: JOB_MANAGER_RPC_ADDRESS
21   value: flink-jobmanager
22 resources:
23   requests:
24     cpu: 2
25     memory: "1024Mi"
26   limits:
27     cpu: 3
28     memory: "2048Mi"
```

untracked changes

Unstaged Changes
No changes

Staged Changes
No changes

Commit message

Commit to mode-deploy 72

☐ bump the resources temporarily to see if we ca... Undo 10h
☐ set 1 replicate for now 10h
☐ Revert "do we need to recreate?" 10h
☐ do we need to recreate? 10h
☐ fix service name 10h

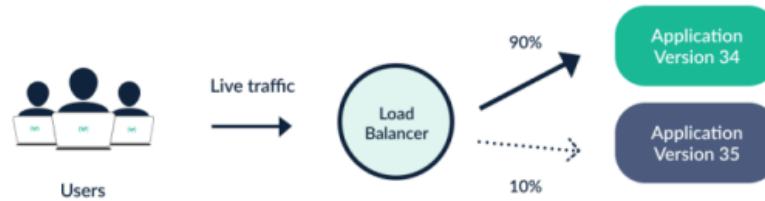
LF UTF-8 YAML mode-deploy Fetch 0 files

MacBook Pro

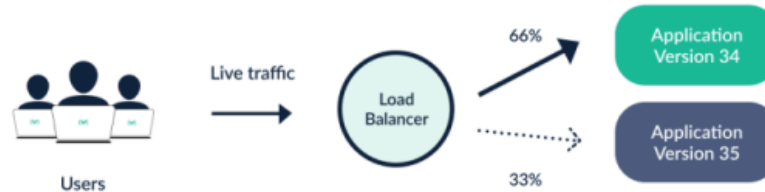
1 INITIAL VERSION



2 NEW VERSION USED BY 10% OF USERS



3 NEW VERSION USED BY 33% OF USERS



4 NEW VERSION USED BY ALL USERS





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

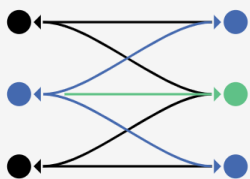
Set Weight: 60%

Pause: 10s



Istio

Connect, secure, control, and observe services.



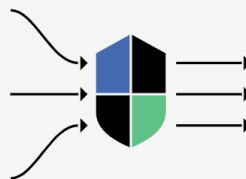
Connect

Intelligently control the flow of traffic and API calls between services, conduct a range of tests, and upgrade gradually with red/black deployments.



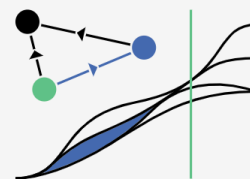
Secure

Automatically secure your services through managed authentication, authorization, and encryption of communication between services.



Control

Apply policies and ensure that they're enforced, and that resources are fairly distributed among consumers.



Observe

See what's happening with rich automatic tracing, monitoring, and logging of all your services.

PROMETHEUS



A systems monitoring and alerting toolkit

csanchez.org

