

ZERO DOWNTIME DEPLOYMENT STRATEGIES WITH K8S



+ HOW TO PREPARE YOUR SERVICE

Wojciech Barczynski - SMACC.io | Hypatos.ai
Listopad 2018

WOJCIECH BARCZYŃSKI

- Lead Software Engineer
& System Engineer
- Interests:
working software
- Hobby:
teaching software
engineering and
programming



STORY

Go + Kubernetes

- **SMACC** - Fintech / ML - [10.2017- ...]
- **Lyke** - Mobile Fashion app - [12.2016, 07.2017]

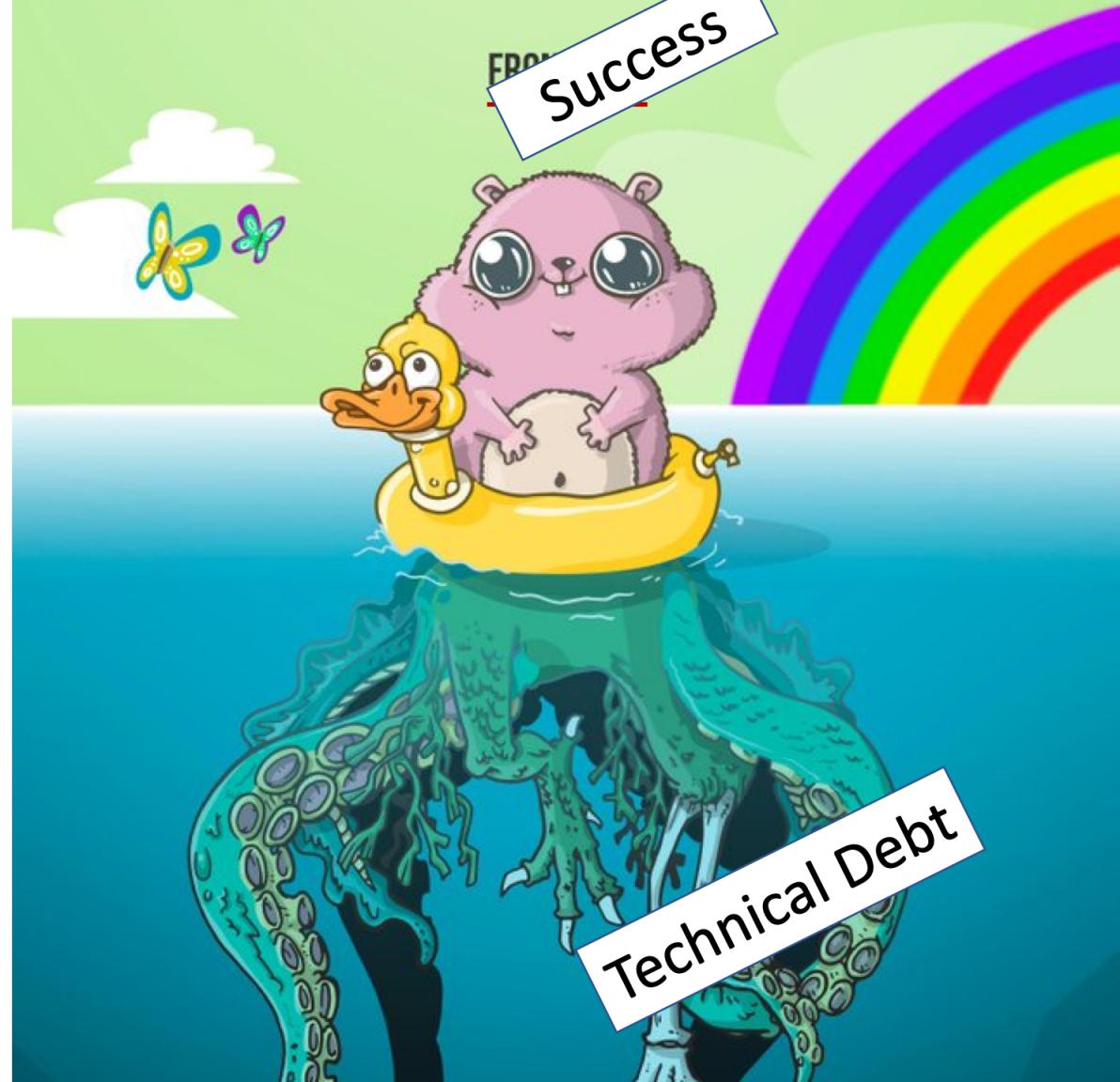
Slow delivery

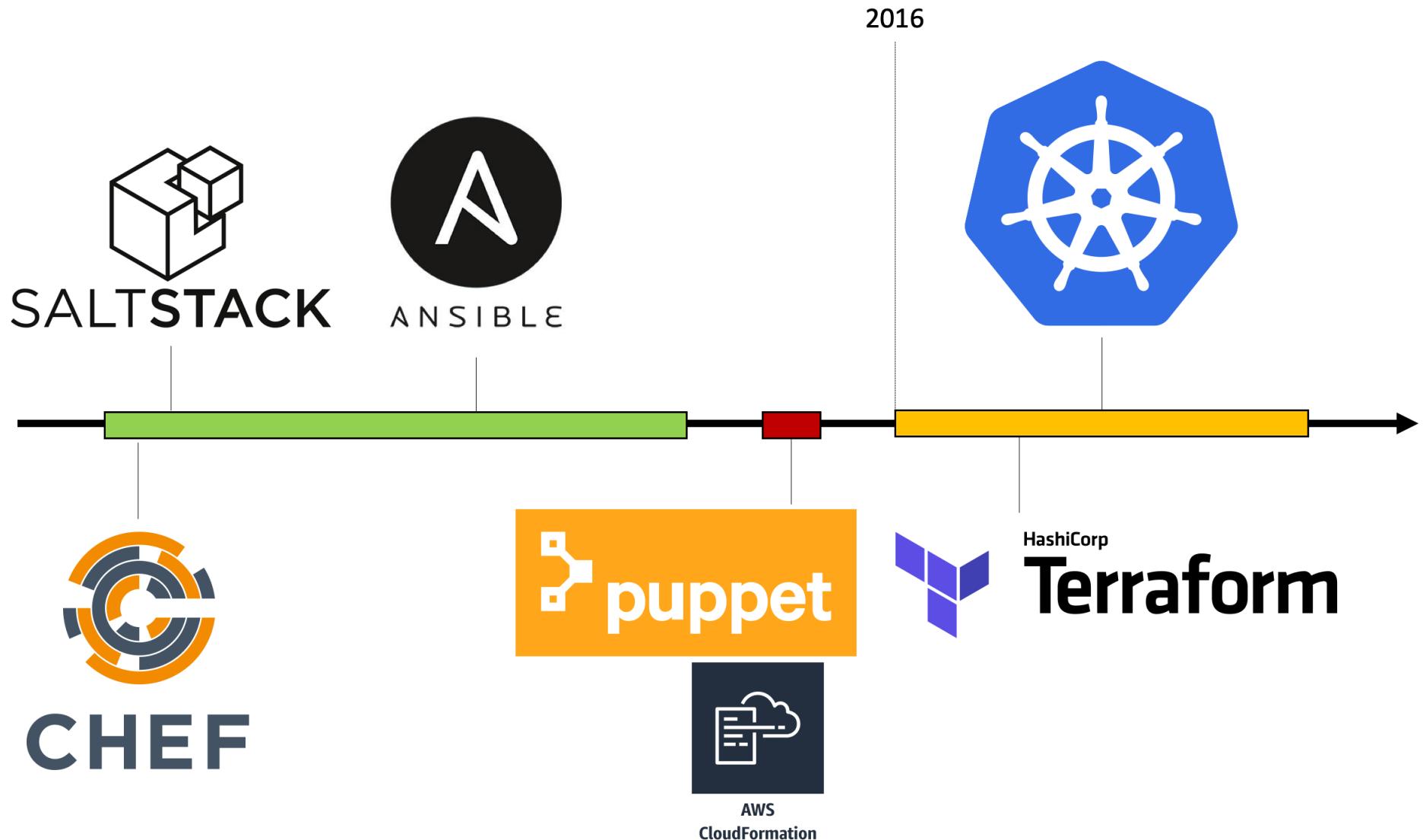
Continuous Deployment?

Fear

Frustration

XX% Idle Machines





Black (Blue) Box

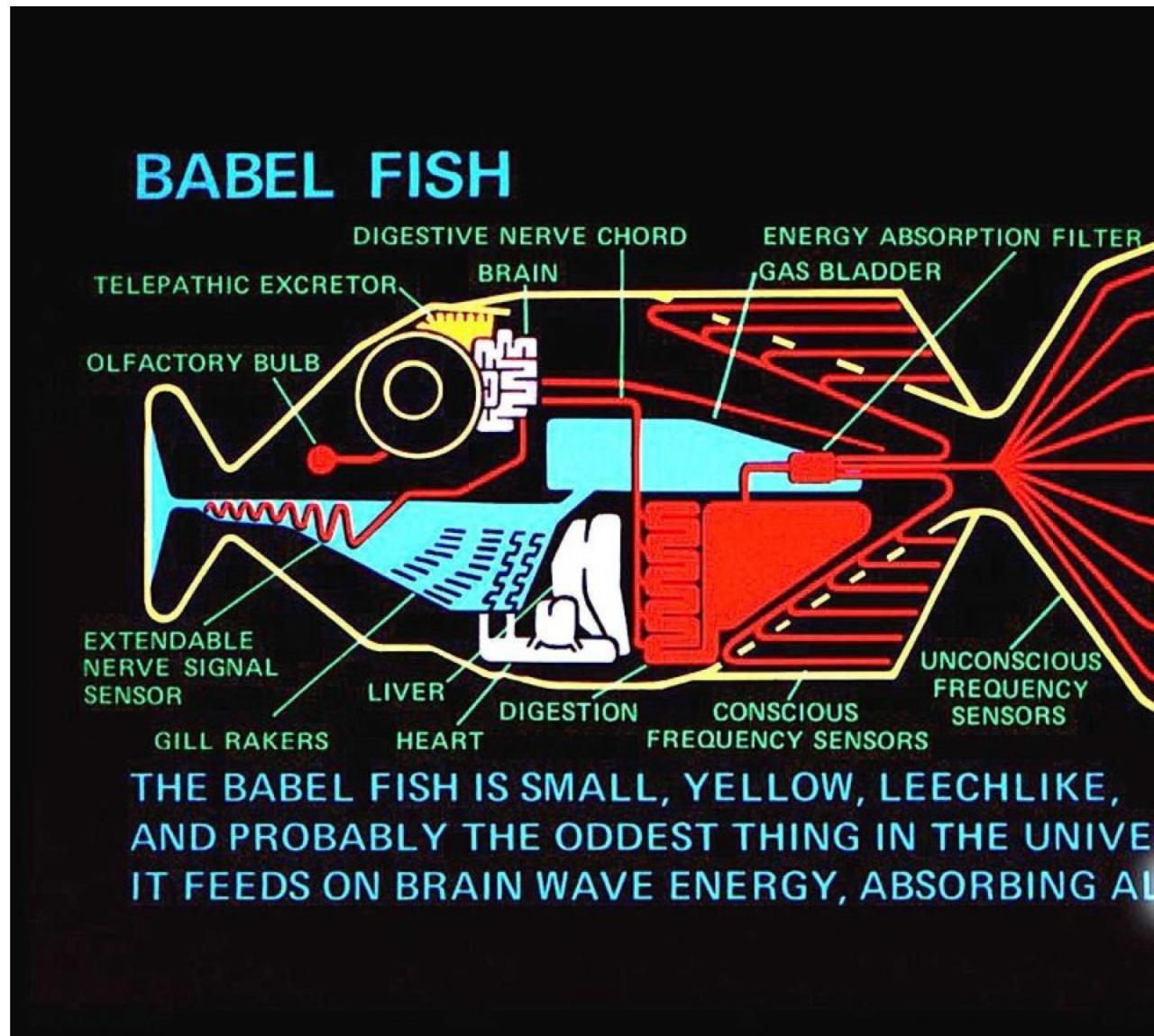
Infrastructure (almost) invisible

Easy* Continuous Deployment



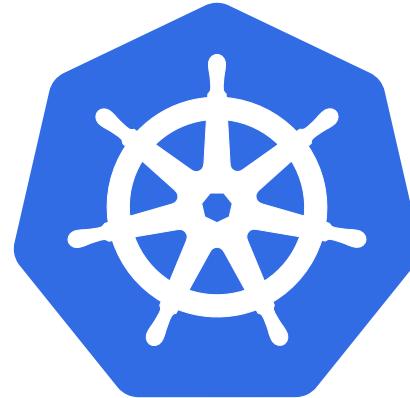
[https://en.wikipedia.org/wiki/File:Dr_Who_\(316350537\).jpg](https://en.wikipedia.org/wiki/File:Dr_Who_(316350537).jpg)

Common
Language
Artifacts
Platform

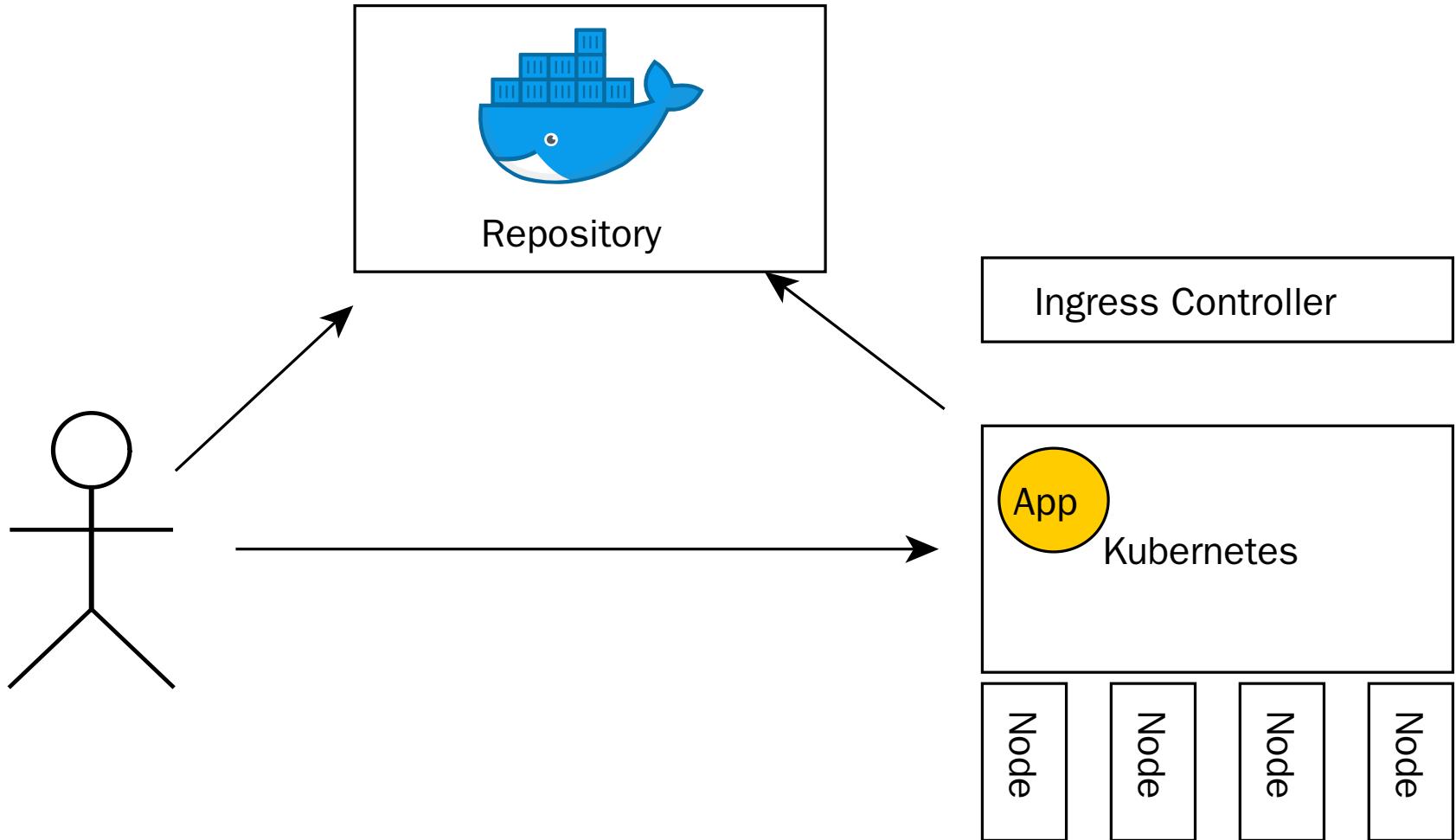


KUBERNETES

- Learn-as-you-go
- Battery for 12factor apps
- ...app must be smarter

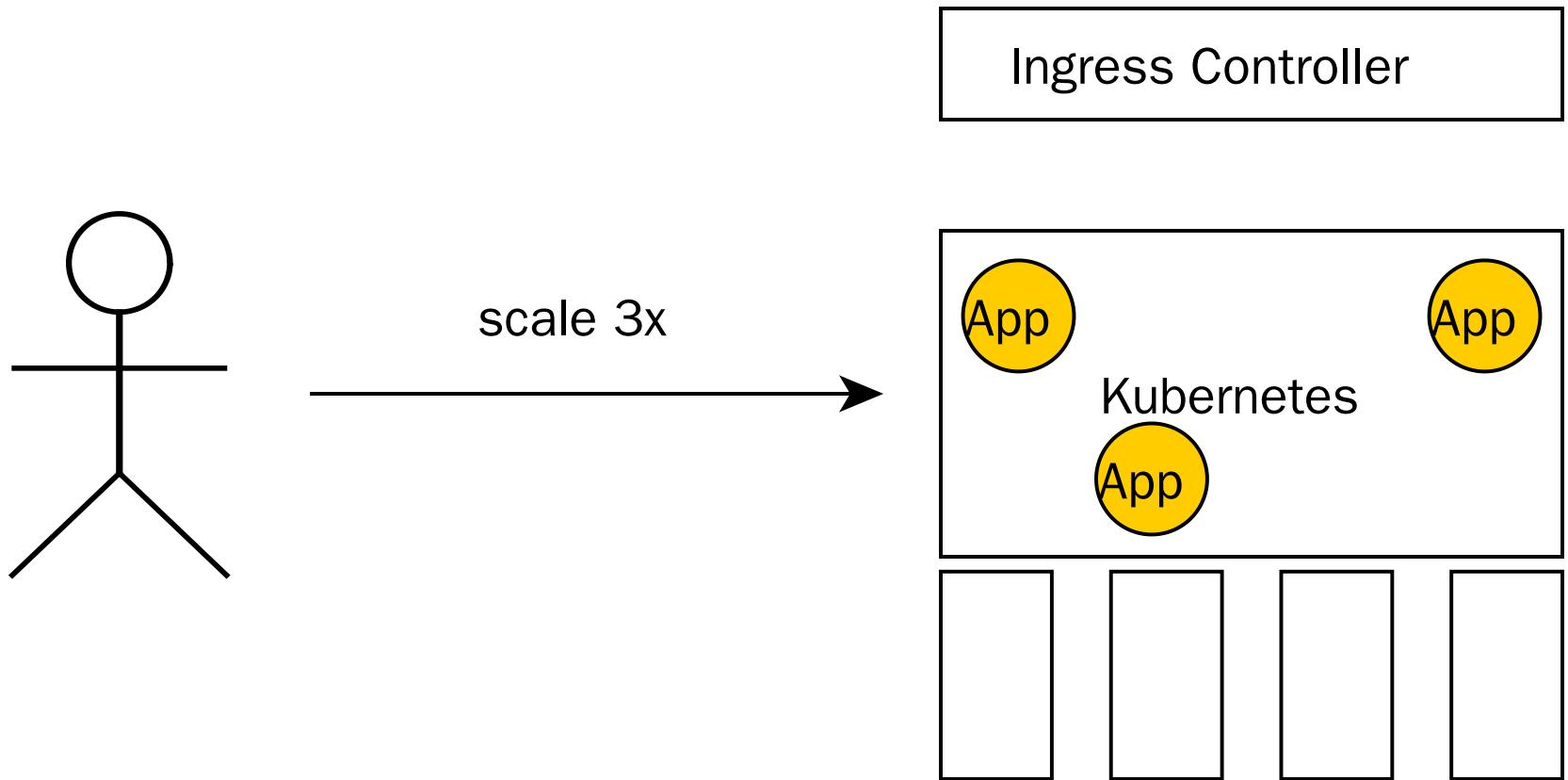


KUBERNETES



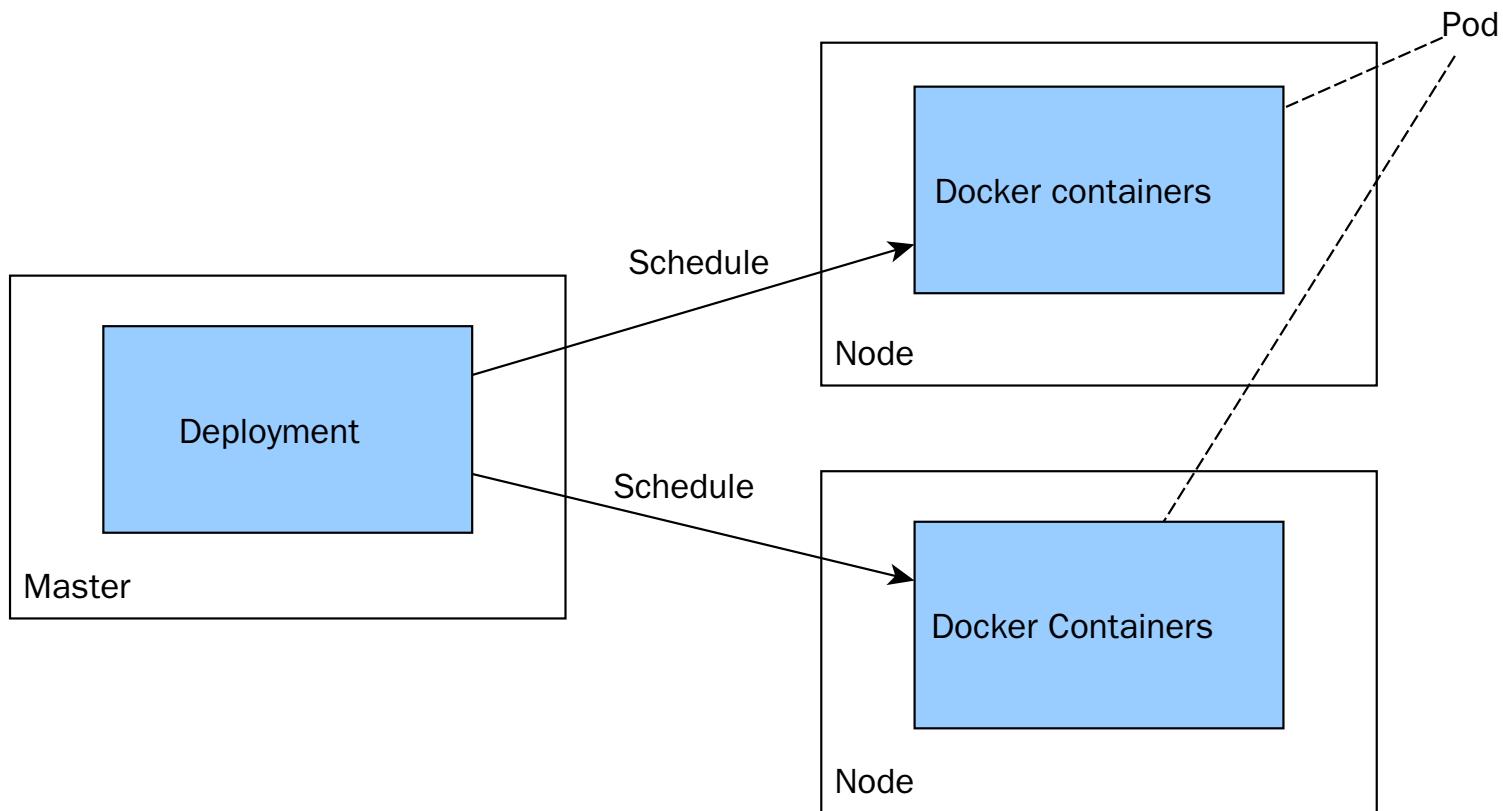
make docker_push; kubectl create -f app-srv-dpl.yaml

SCALE UP! SCALE DOWN!



```
kubectl --replicas=3 -f app-srv-dpl.yaml
```

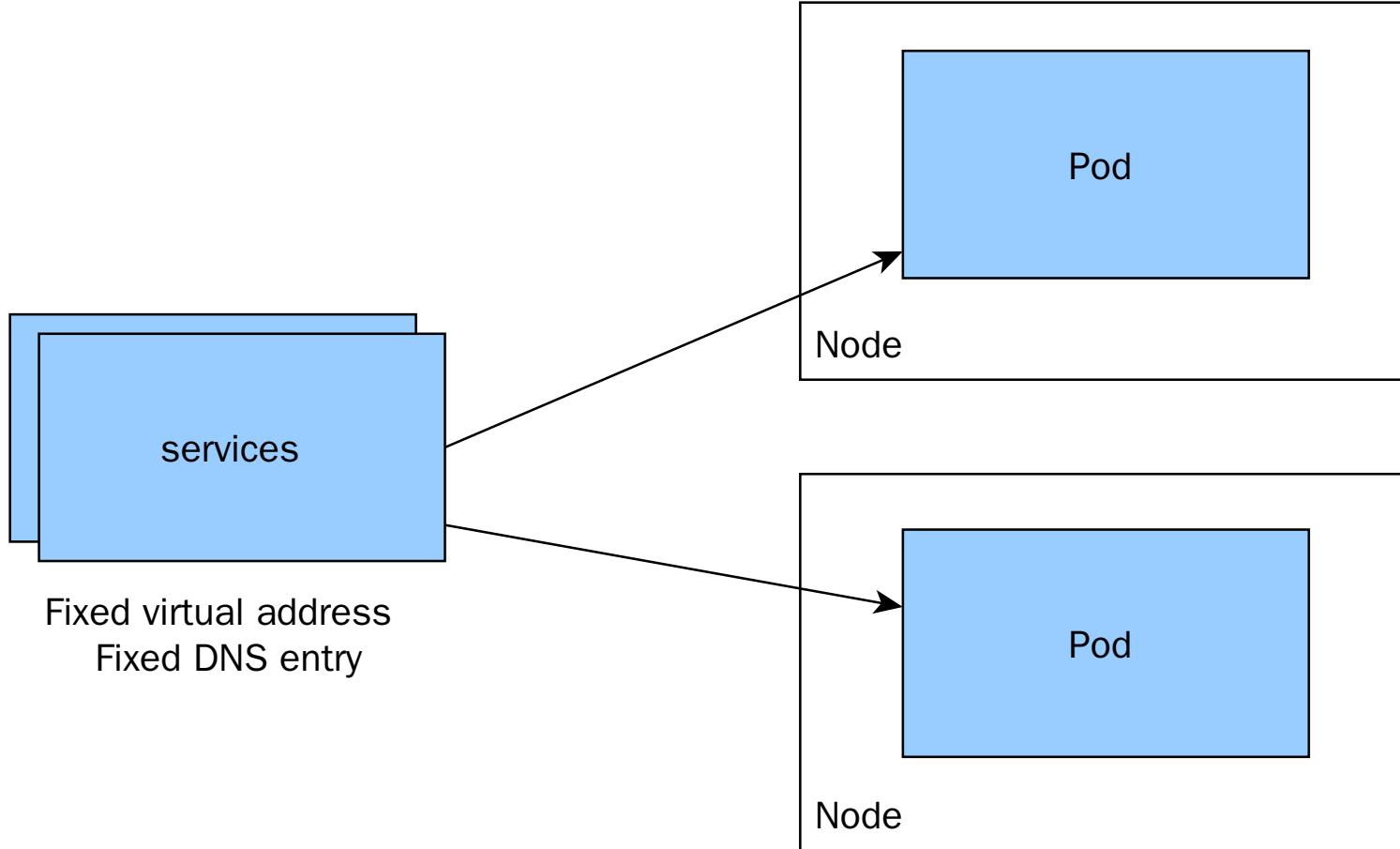
DEPLOYMENT AND PODS



DEPLOYMENT.YML

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: demo-api
  labels:
    app: demo-api
spec:
  replicas: 3
  strategy:
    type: Recreate
  selector:
    matchLabels:
      app: demo-api
template:
  metadata:
```

SERVICE AND PODS



Service matches pods based on labels

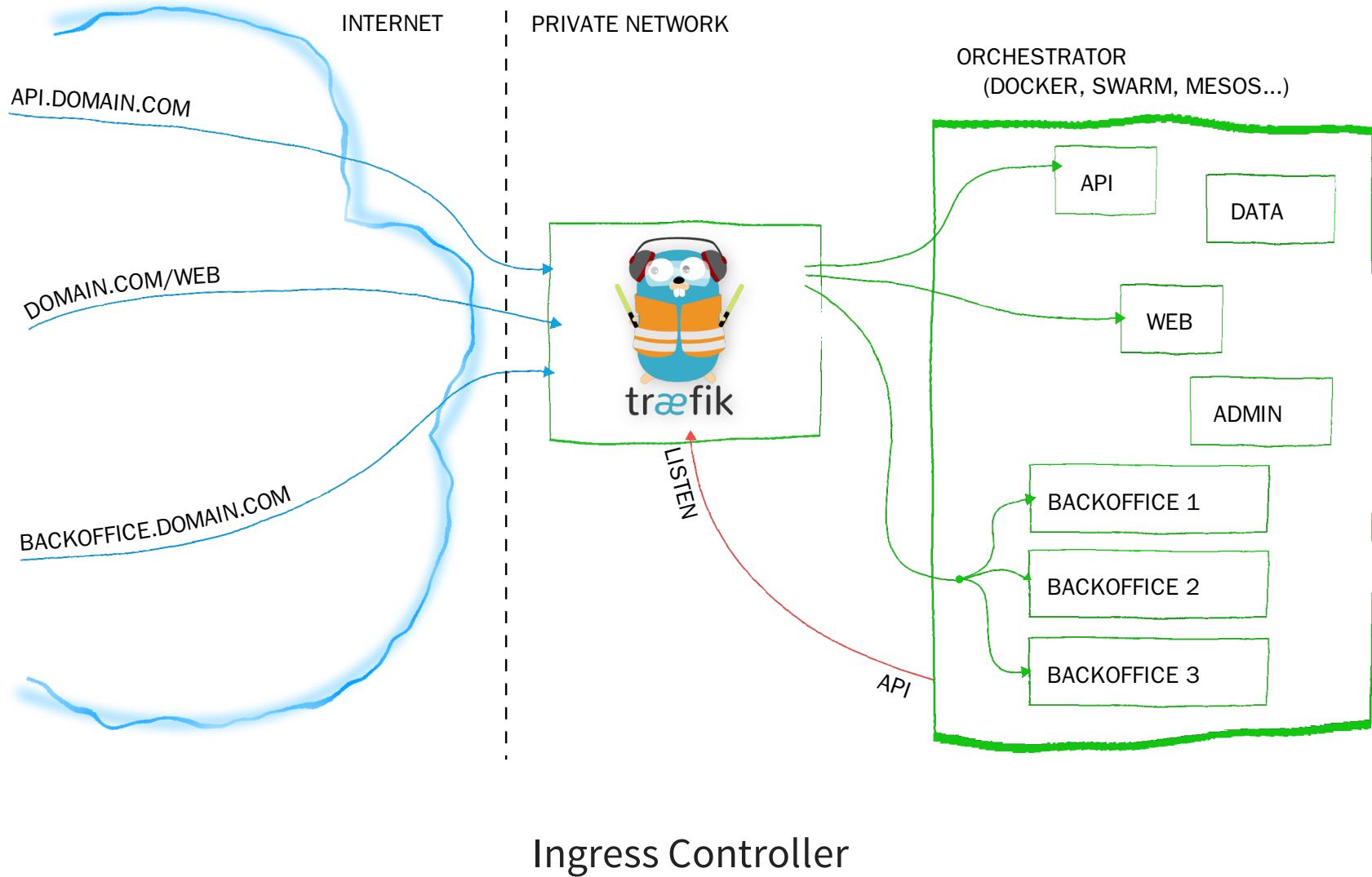
SERVICE.YML

```
apiVersion: v1
kind: Service
metadata:
  name: demo-api
spec:
  ports:
  - port: 80
    protocol: TCP
  selector:
    app: demo-api
  type: LoadBalancer
```

BASIC CONCEPTS

Name	Purpose	
Service	Interface	Entry point (Service Name)
Deployment	Factory	How many pods, which pods
Pod	Implementation	1+ docker running

HOW GET USER REQUESTS?



INGRESS

Pattern

api.smacc.io/v1/users

Target App Service

users-v1

api.smacc.io/v2/users

users-v2

smacc.io

web

SERVICE DISCOVERY AND LABELING

- names in DNS:

curl http://users/list

- labels:

name=value

- annotations:

prometheus.io/scrape: "true"

DEPLOYMENT STRATEGIES

STRATEGIES

We will see:

- Replace (downtime visible)
- Rolling updates
- Blue Green
- Canary

OTHER

We will not cover:

- Feature toggles
- A/B like
- Shadow deployment

FIRST THE HOMEWORK

Need to support:

- liveness - am I dead?
- readiness - can I serve requests?

KUBE LIVENESS PROBE

```
livenessProbe:  
  httpGet:  
    path: /model  
    port: 8000  
    httpHeaders:  
      - name: X-Custom-Header  
        value: Awesome  
  initialDelaySeconds: 600  
  periodSeconds: 5  
  timeoutSeconds: 18  
  successThreshold: 1  
  failureThreshold: 3
```

LIVENESS PROBE

- our pod gets restarted
- too many restarts -> CrashLoop

KUBE READINESS PROBE

```
readinessProbe:  
  exec:  
    command:  
    - cat  
      - /tmp/healthy  
  initialDelaySeconds: 5  
  periodSeconds: 5
```

YOUR APP SHOULD ON STOP

1. we get SIGTERM signal

YOUR APP SHOULD ON STOP

1. we get SIGTERM signal
2. app gives 500 on readinessProbe

YOUR APP SHOULD ON STOP

1. we get SIGTERM signal
2. app gives 500 on readinessProbe
3. k8s does not send more requests

YOUR APP SHOULD ON STOP

1. we get SIGTERM signal
2. app gives 500 on readinessProbe
3. k8s does not send more requests
4. app shuts down gracefully

YOUR APP SHOULD ON STOP

1. we get SIGTERM signal
2. app gives 500 on readinessProbe
3. k8s does not send more requests
4. app shuts down gracefully
5. kubernetes forces kill if 30s limit exceeded

ALWAYS

Implement readiness for:

- ML Model-based components
- slow starting time

DEMO SERVICE IMPLEMENTATION

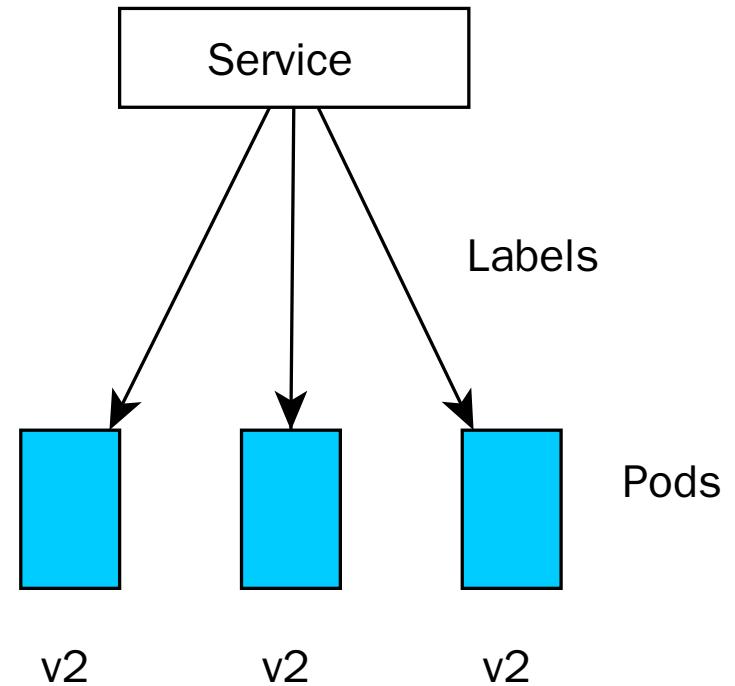
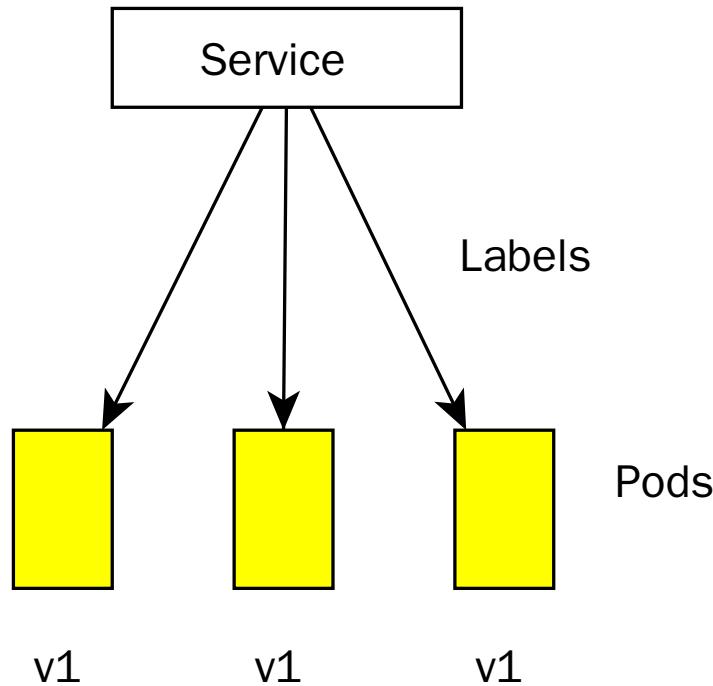
- graceful shutdown
- demo service

GRACEFUL SHUTDOWN

From [missy](#):

```
func (s *Service) prepareShutdown(h Server) {
    signal.Notify(s.Stop, os.Interrupt, syscall.SIGTERM)
    <-s.Stop
    s.StatusNotReady()
    shutdown(h)
}
```

DEMO - RECREATE



DEMO - RECREATE

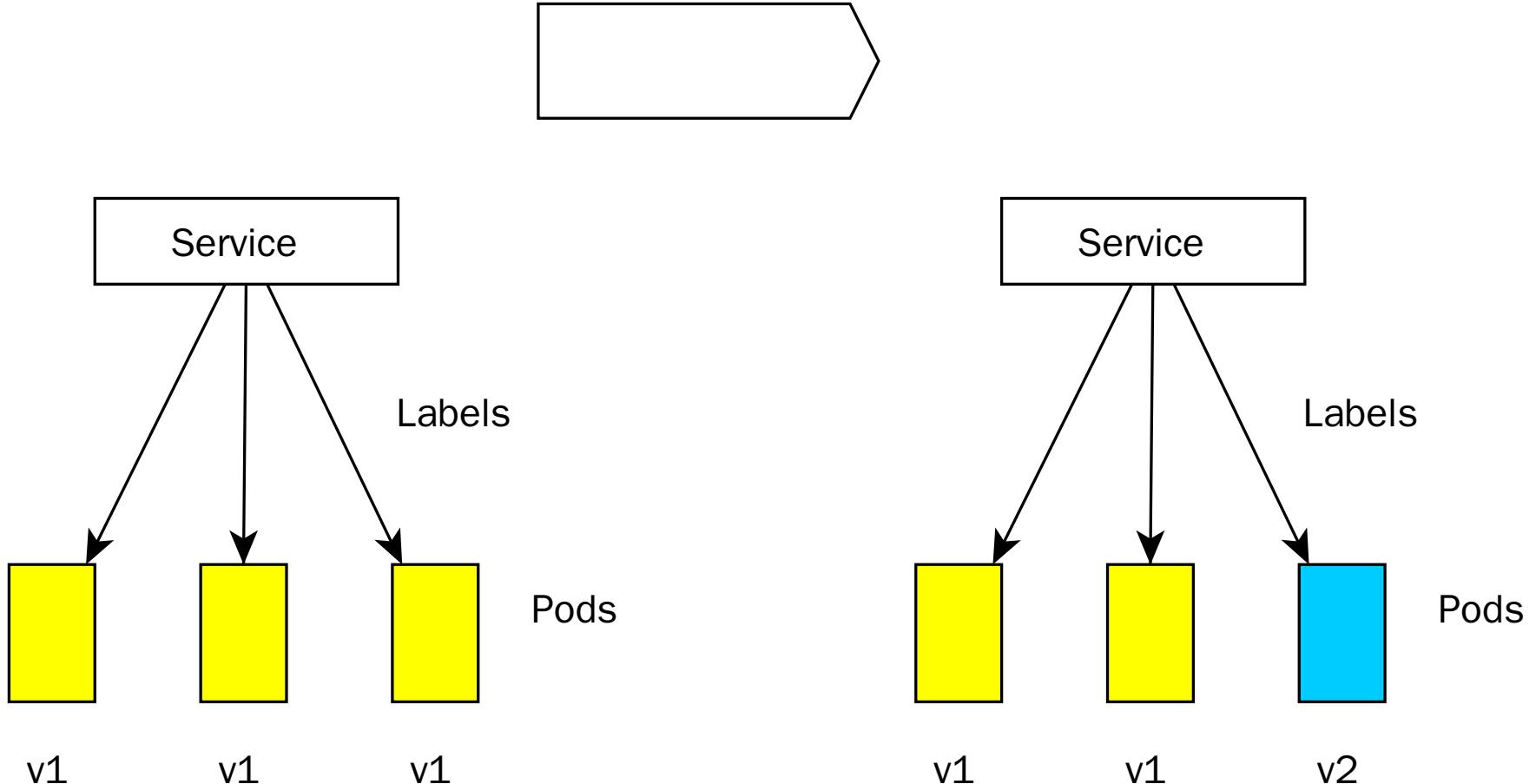
```
spec:  
  replicas: 3  
  strategy:  
    type: Recreate
```

```
kubectl set image deployment/demo-api \  
  app=wojciech11/api-status:2.0.0
```

DEMO - RECREATE

- quick
- downtime visible

DEMO - ROLLING UPDATES



DEMO - ROLLING UPDATES

```
strategy:  
  type: RollingUpdate  
  rollingUpdate:  
    maxSurge: 2  
    maxUnavailable: 0
```

[docs](#)

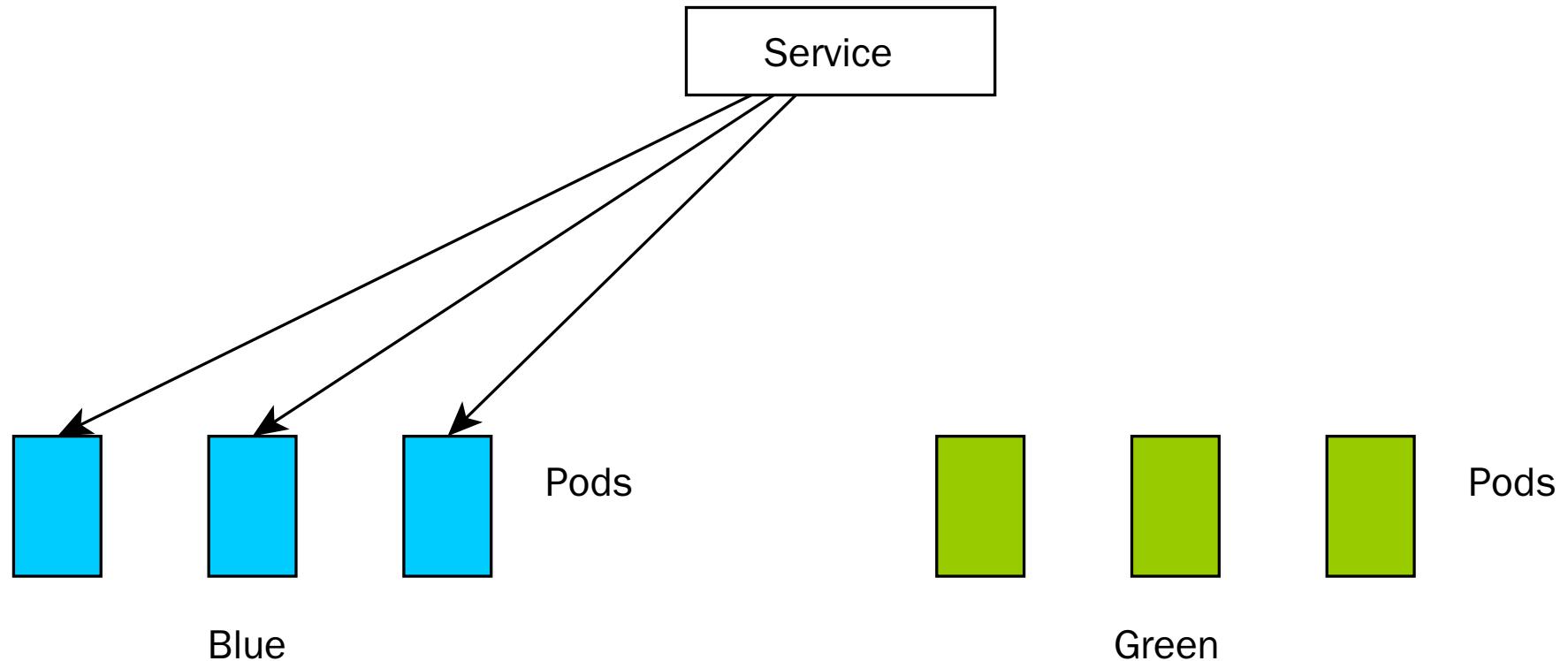
DEMO - ROLLING UPDATES

```
kubectl set image deployment/demo-api  
  app=wojciech11/api-status:2.0.0
```

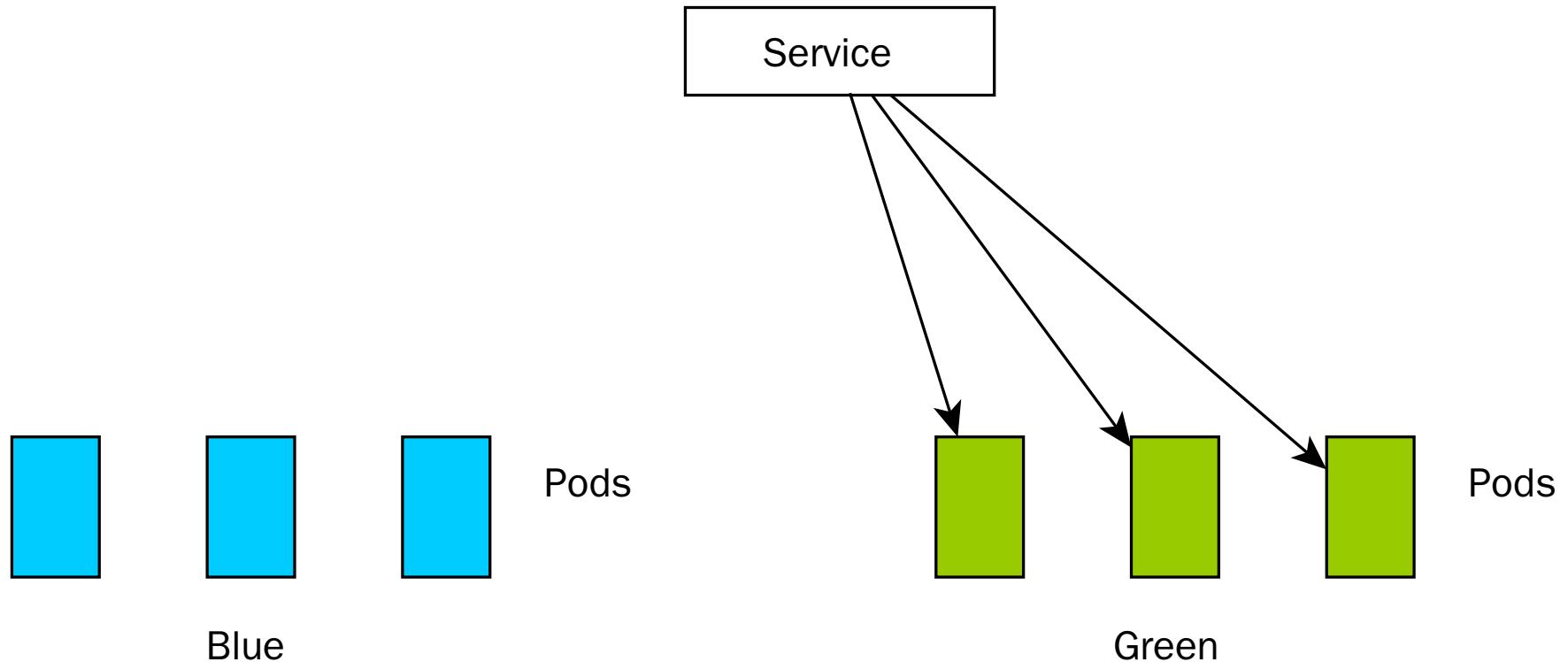
DEMO - ROLLING UPDATES

- the most popular

DEMO - GREEN/BLUE



DEMO - GREEN/BLUE



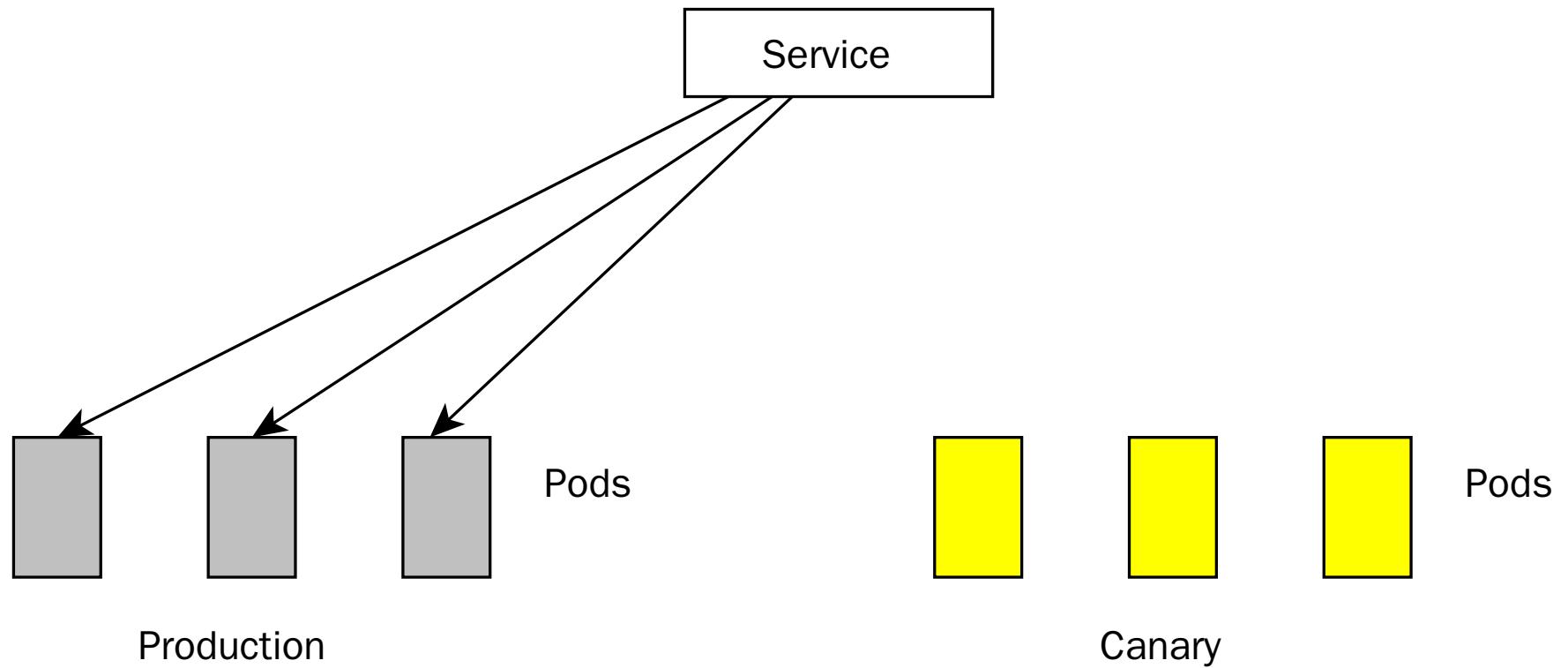
DEMO - GREEN/BLUE

```
kubectl patch service api-status \
  -p '{"spec":{"selector":{"label": "green"}}}'
```

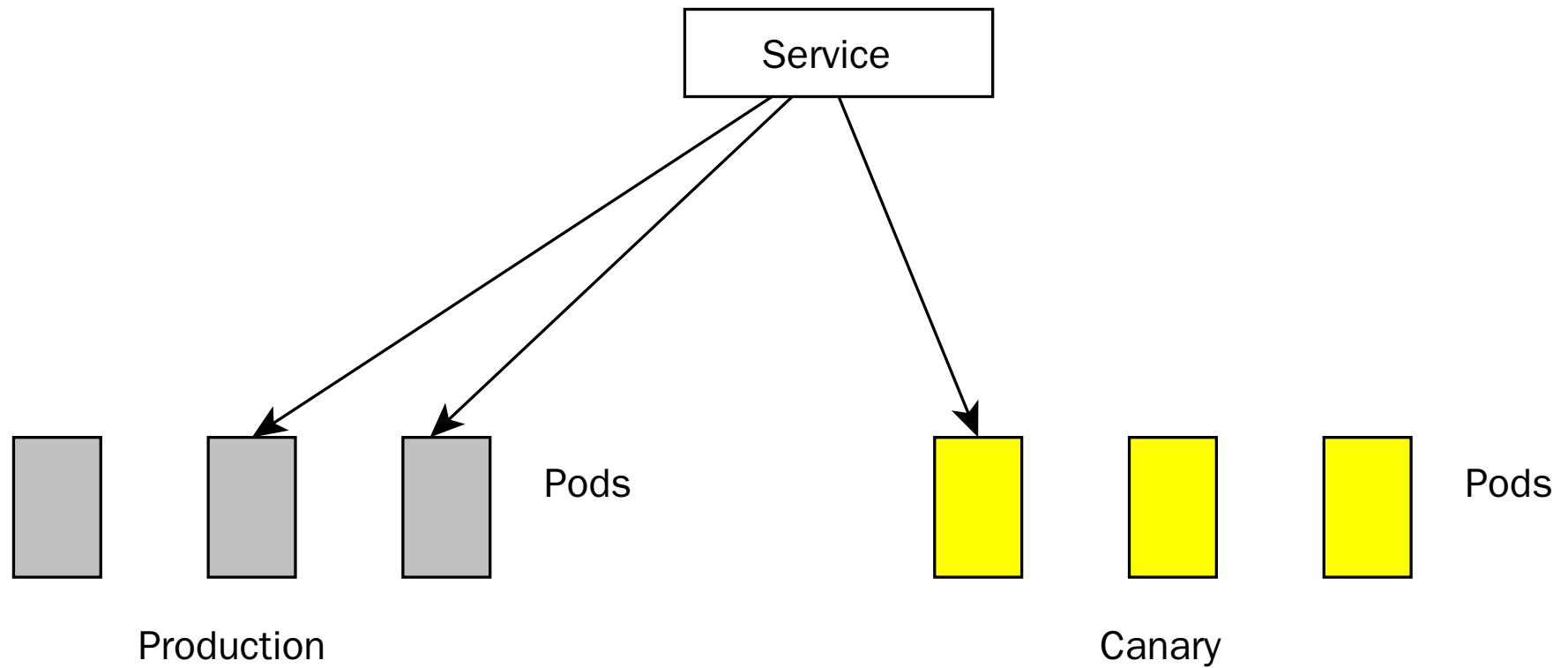
DEMO - GREEN/BLUE

- For big changes
- Less common
- Might be implemented with *Ingress*

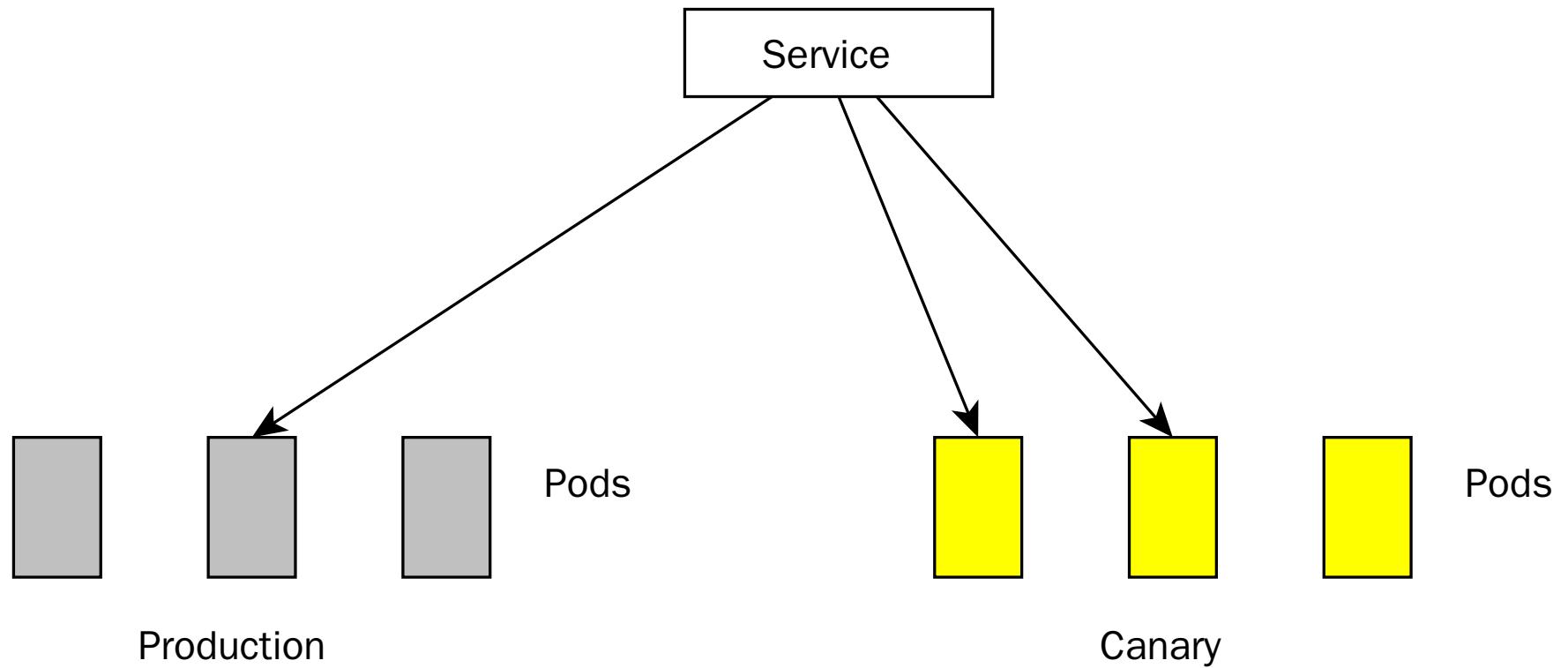
DEMO - CANARY



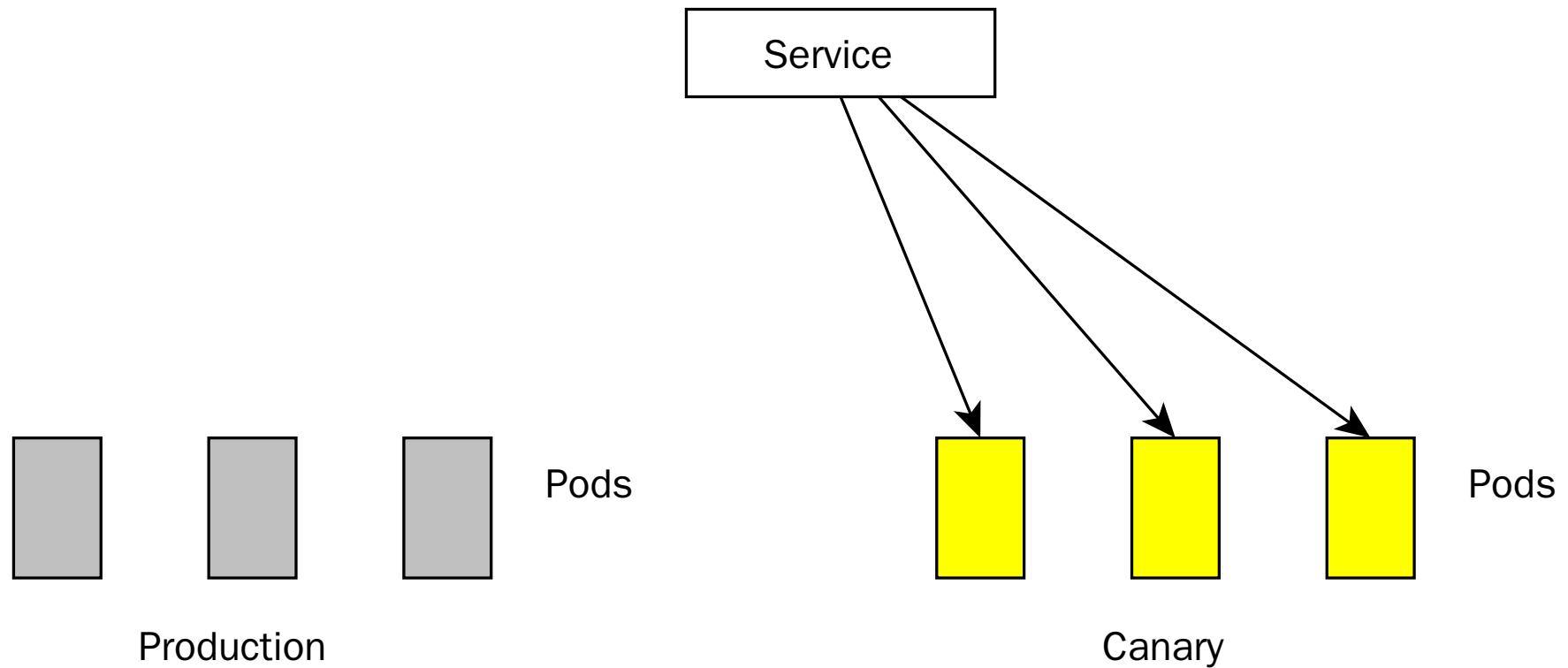
DEMO - CANARY



DEMO - CANARY



DEMO - CANARY



DEMO - CANARY

- manually

```
kubectl scale --replicas=3 deploy/api-status-nginx-blue
kubectl scale --replicas=1 deploy/api-status-nginx-green

# no errors, let's continue
kubectl scale --replicas=2 deploy/api-status-nginx-blue
kubectl scale --replicas=2 deploy/api-status-nginx-green
```

TRAEFIK

```
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  annotations:
    traefik.ingress.kubernetes.io/service-weights: |
      my-app: 99%
      my-app-canary: 1%
  name: my-app
spec:
  rules:
  - http:
    paths:
    - backend:
        serviceName: my-app
        servicePort: 80
```

traffic-splitting

ISTIO

```
apiVersion: networking.istio.io/v1alpha3
kind: VirtualService
metadata:
  name: helloworld
spec:
  hosts:
    - helloworld
  http:
    - route:
        - destination:
            host: helloworld
            subset: v1
            weight: 90
        - destination:
            host: helloworld
```

traffic shifting with Istio

SUMMARY

- learn-as-you go <3
- easy to implement complex strategies
- see `kubectl rollout` and `minReadySeconds` for more control

THANK YOU. QUESTIONS?

https://github.com/wojciech12/talk_zero_downtime_deployment_with_kubernetes

```
123 def distance_matrix(regions):  
124     """ Computes a distance matrix against a region list """  
125     tuples = [r.as_tuple() for r in regions]  
126     return cdist(tuples, tuples, region_distance)  
127  
128  
129 def clusterize(words, **kwargs):  
130     # TODO: write a cool docstring here  
131     db = DBSCAN(metric="precomputed", **kwargs)  
132     X = distance_matrix([Region.from_word(w) for w in words])  
133     labels = [int(l) for l in db.fit_predict(X)]
```



SMACC



Go



PYTORCH

TensorFlow™



amazon
web services™



Azure



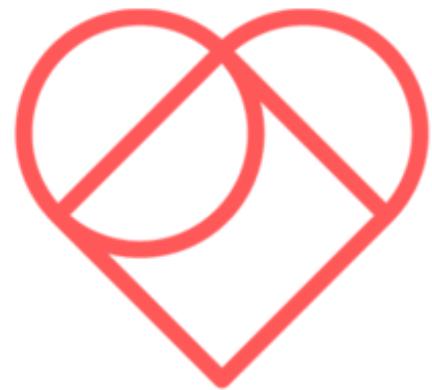
BACKUP SLIDES

```
123 def distance_matrix(regions):  
124     """ Computes a distance matrix against a region list """  
125     tuples = [r.as_tuple() for r in regions]  
126     return cdist(tuples, tuples, region_distance)  
127  
128  
129 def clusterize(words, **kwargs):  
130     # TODO: write a cool docstring here  
131     db = DBSCAN(metric="precomputed", **kwargs)  
132     X = distance_matrix([Region.from_word(w) for w in words])  
133     labels = [int(l) for l in db.fit_predict(X)]
```



STORY

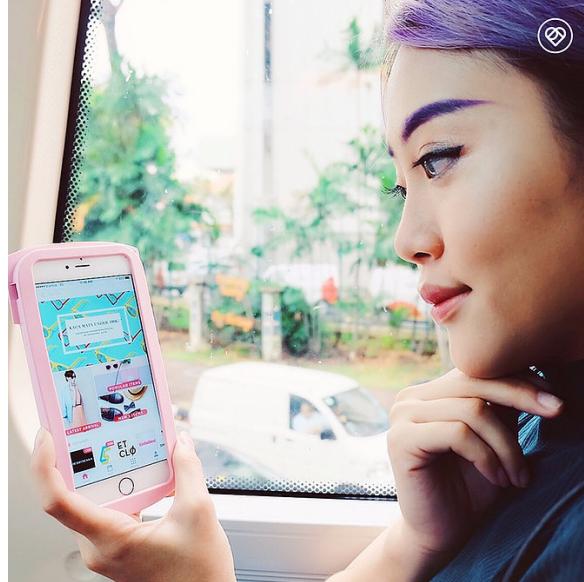
- Lyke - [12.2016 - 07.2017]
- SMACC - [10.2017 - present]



LYKE

LYKE

- E-commerce
- Mobile-only
- 50k+ users
- 2M downloads
- Top 10 Fashion Apps
w Google Play Store



<http://www.news.getlyke.com/single-post/2016/12/02/Introducing-the-New-Beautiful-LYKE>

Now JollyChic Indonesia

GOOD PARTS

- Fast Growth
- A/B Testing
- Data-driven
- Product Manager,
UI Designer,
Mobile Dev,
and tester - one
body



CHALLENGES

- 50+ VMs in Amazon, 1 VM - 1 App, idle machine
- Puppet, hilarious (manual) deployment process
- Fear
- Forgotten components
- sometimes performance issues

SMACC

Hypatos

SMACC

- Machine Learning FinTech
- SaaS and API platform
- From Enterprise (Deutsche Bank, AoK) to SME
- Well-known FinTech Startup in Germany



STORY

- Legacy on AWS, experiments with AWS ECS :/
- Self-hosted K8S on ProfitBricks
- Get to Microsoft ScaleUp, welcome Azure
- Luckily - Azure-Kubernetes-Service

DIFFERENCE



- Two teams in Berlin and Warsaw
- Me in Warsaw

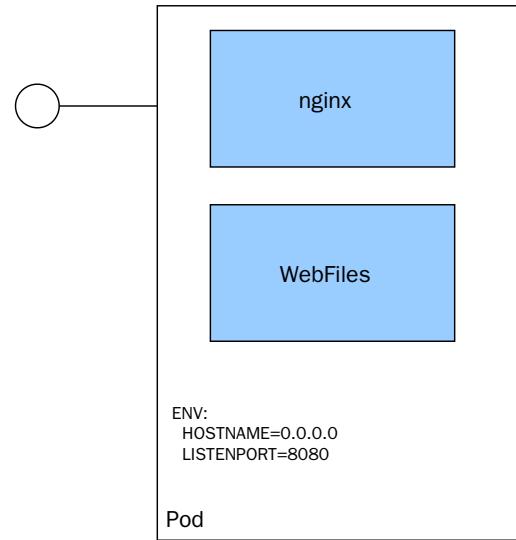
APPROACH

- Simplify, Simplify
- Hide K8S magic
- git tag driven Continuous Deployment

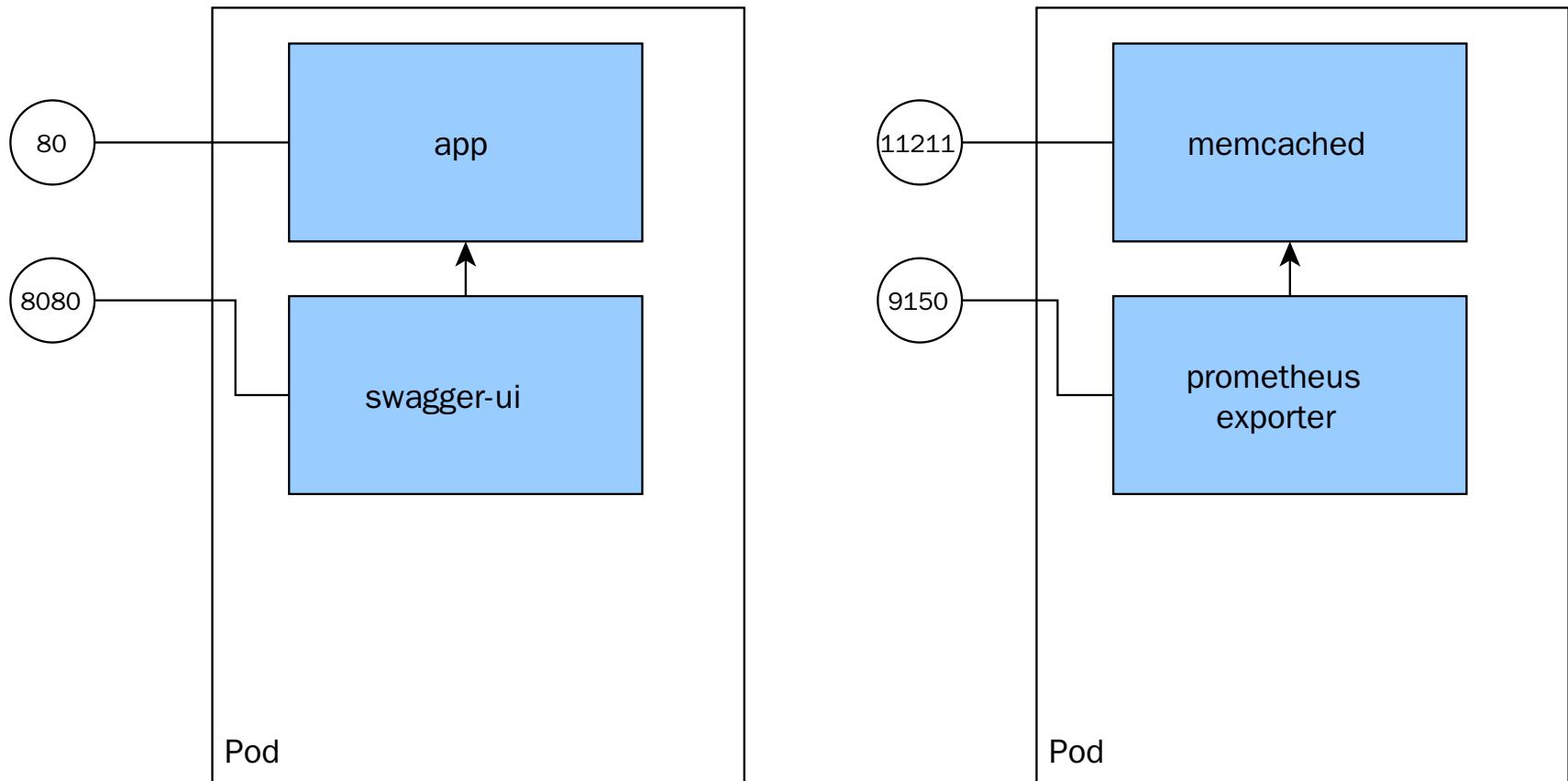
KUBERNETES CONCEPTS+

PODS

- See each other on localhost
- Live and die together
- Can expose multiple ports



SIDE-CARS



LOAD BALANCING

