



18 września 2018 WARSZAWA

Czy Kubernetes jest nowym Linuxem? Czyli efektywne budowanie Twojej platformy z Kubernetes.

Wojciech Barczyński, SMACC.io | Hypatos.ai, wojciech.barczynski@smacc.io

PARTNERZY STRATEGICZNI



WOJCIECH BARCZYŃSKI

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



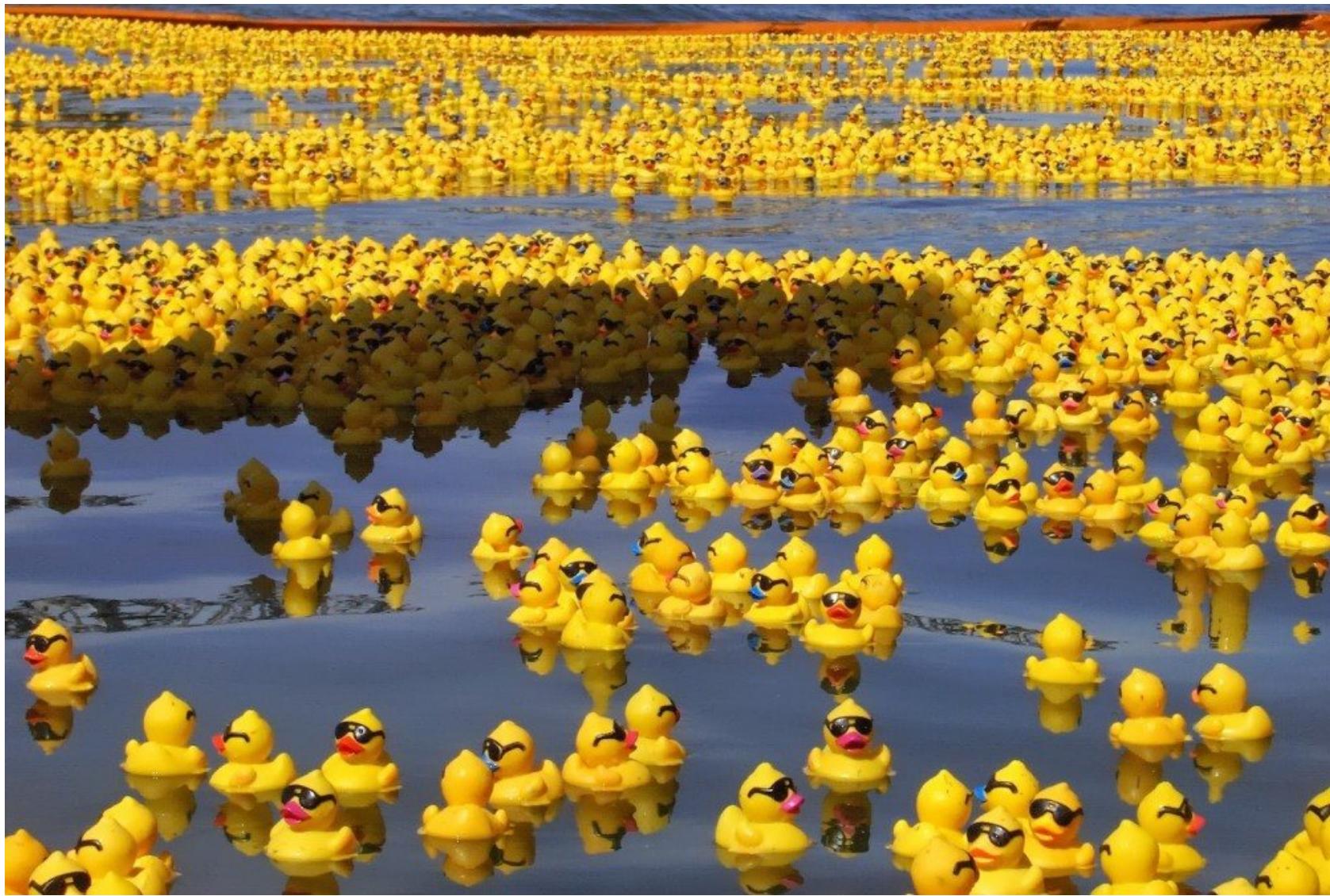
O MNIE

- Machine Learning FinTech - SaaS i produkt API
- Before:
 - 1 z 10 Indonesian mobile e-commerce (Rocket Internet)
- 3.5 roku Openstack (1000+ nodes, 21 DC)
- Before SAP R&D
- I do not like INFRA :D

PROBLEM Z INFRASTRUKTURĄ

- Administracja jest trudna i kosztowna
- Virtualne Maszyny, ansible, salt, etc.
- Za dużo ruchomych części
- Nie kończąca się standaryzacja

MIKROSERWISY AAA!



PROBLEM Z INFRASTRUKTURĄ

- Chmura nie taka tania - \$\$\$

KUBERNETES



WYOBRAŹ SOBIE

- nie musisz myśleć o IaaS
- nigdy nie musisz konfigurować maszyny
- mnie dyskusji o CI / CD ...
- DC jako czarna skrzynka

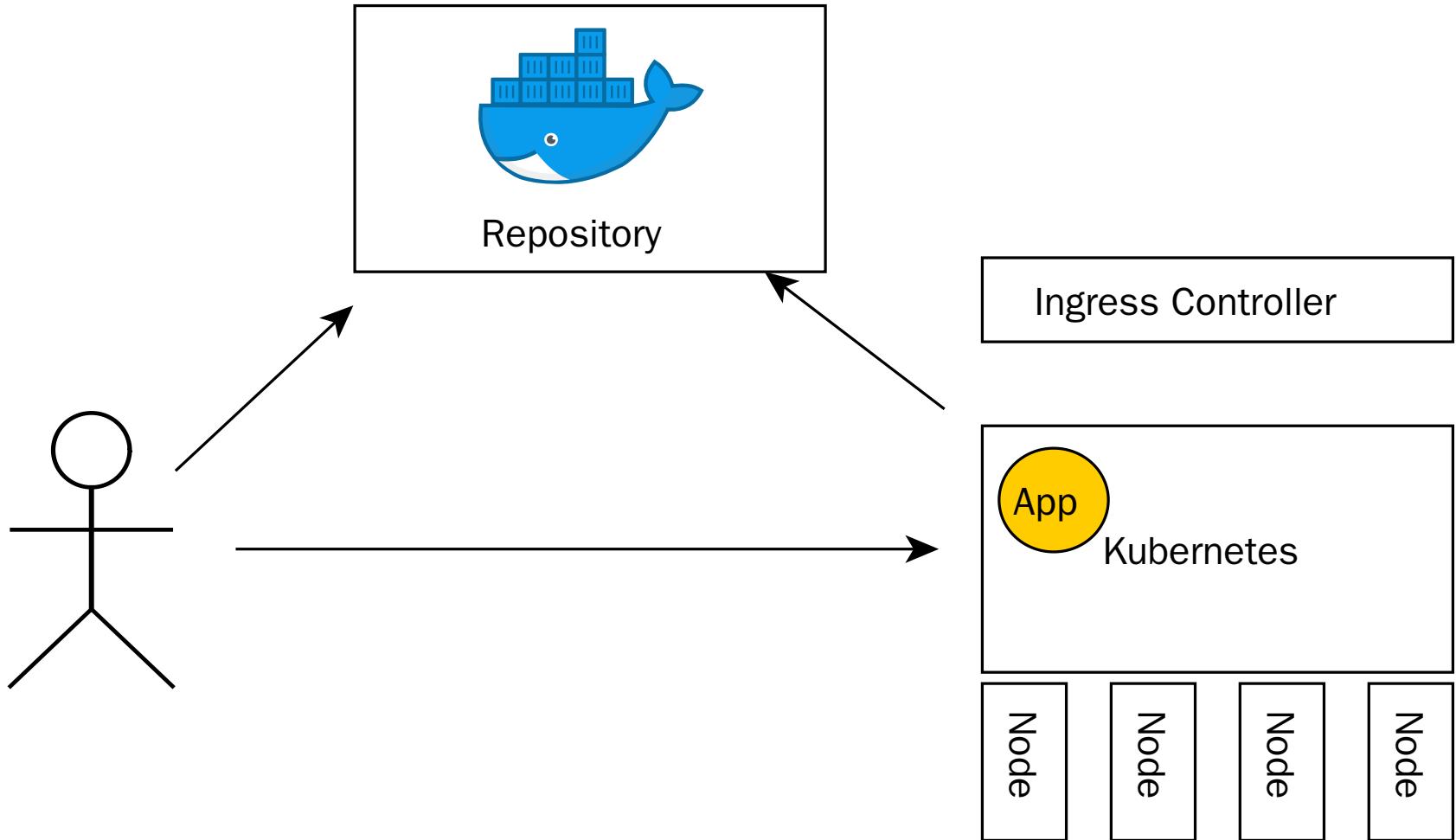
CO TO JEST KUBERNETES?

- Container management
- Zorientowany na serwisy i aplikacje
- Prosty w użyciu*
- Uniezależnia nas od IaaS

KUBERNETES

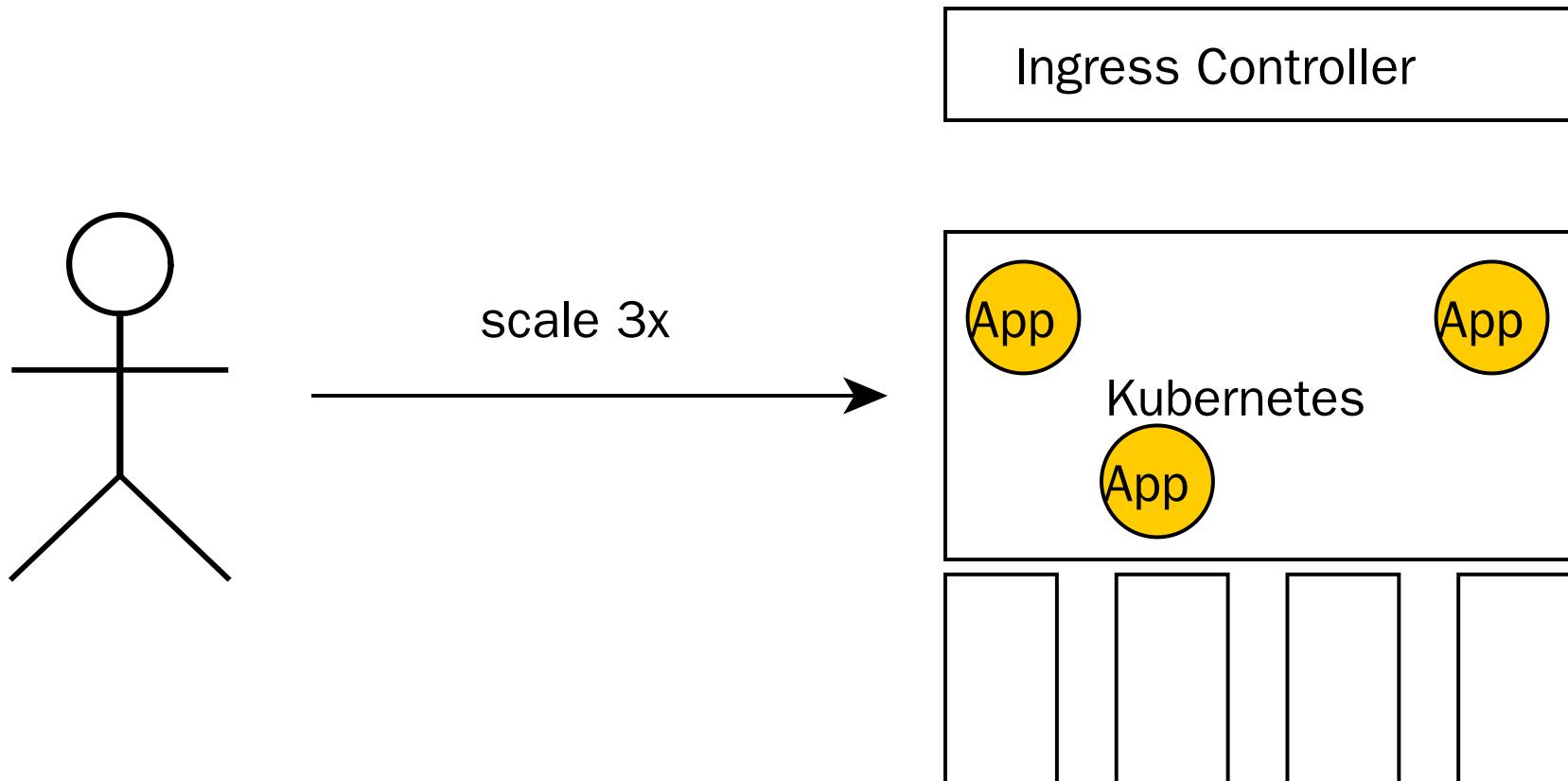
- Wszystko dla 12factory apps
- Service discovery i meta-data support

KUBERNETES



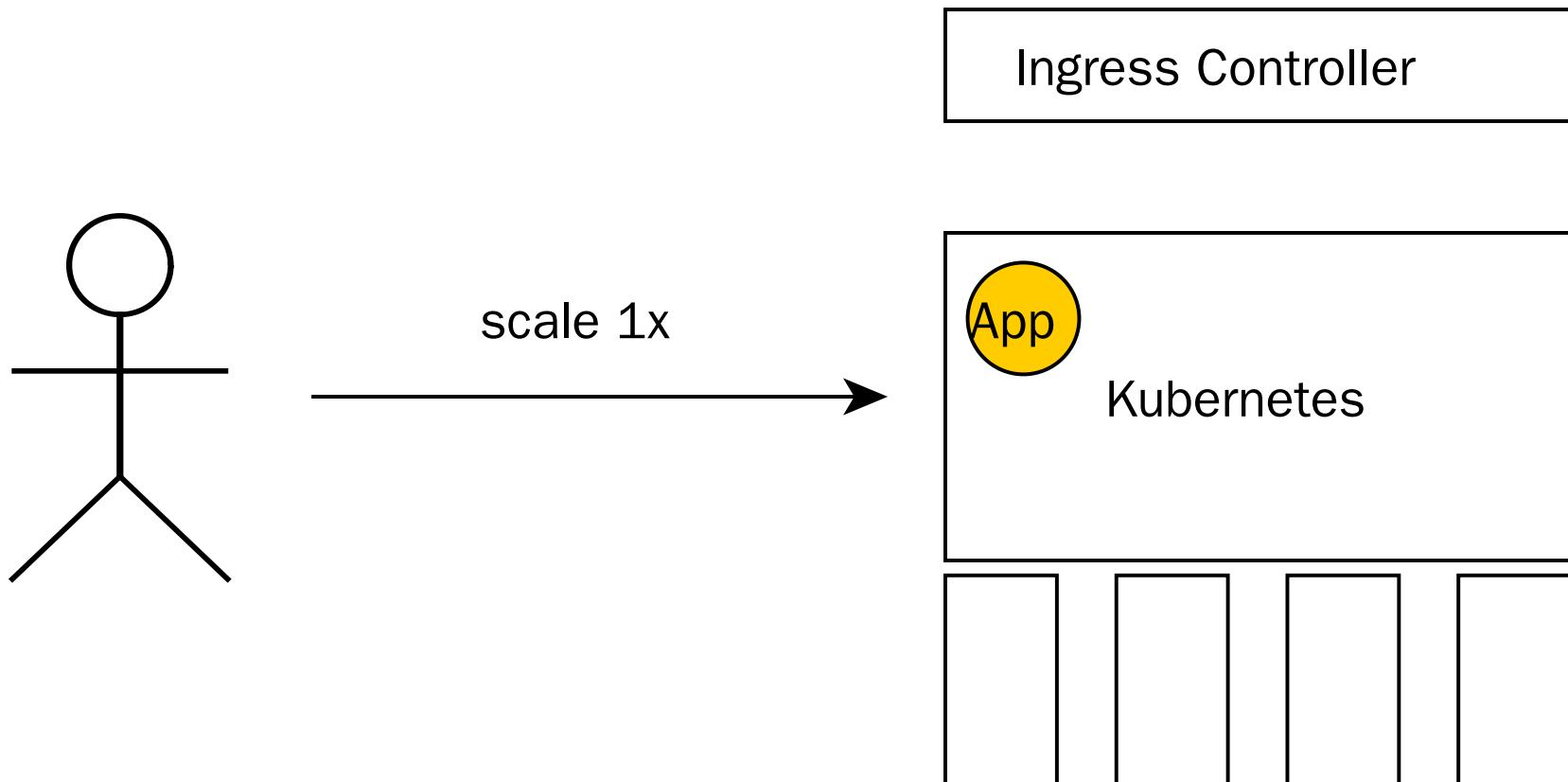
make docker_push; kubectl create -f app-srv-dpl.yaml

SKALOWNIE!



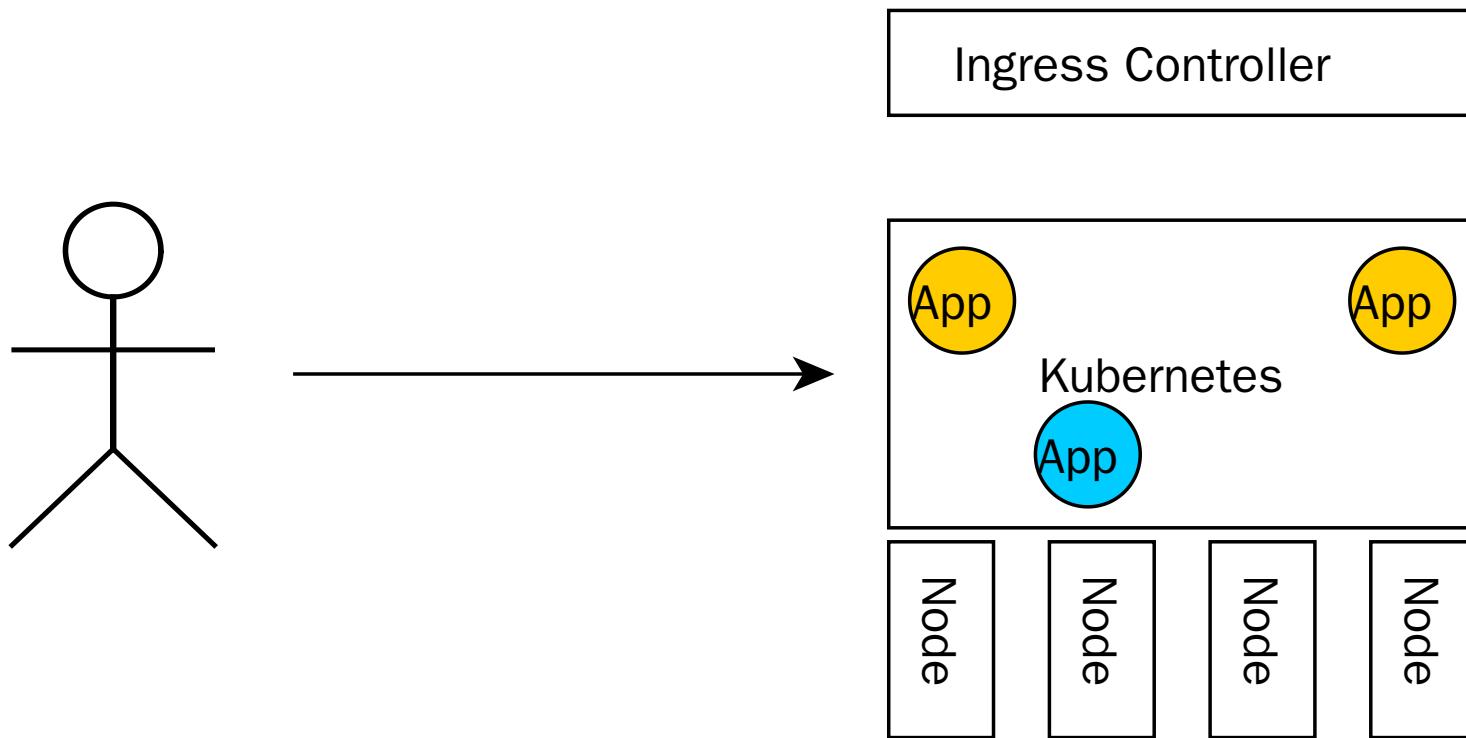
`kubectl --replicas=3 -f app-srv-dpl.yaml`

SKALOWANIE!



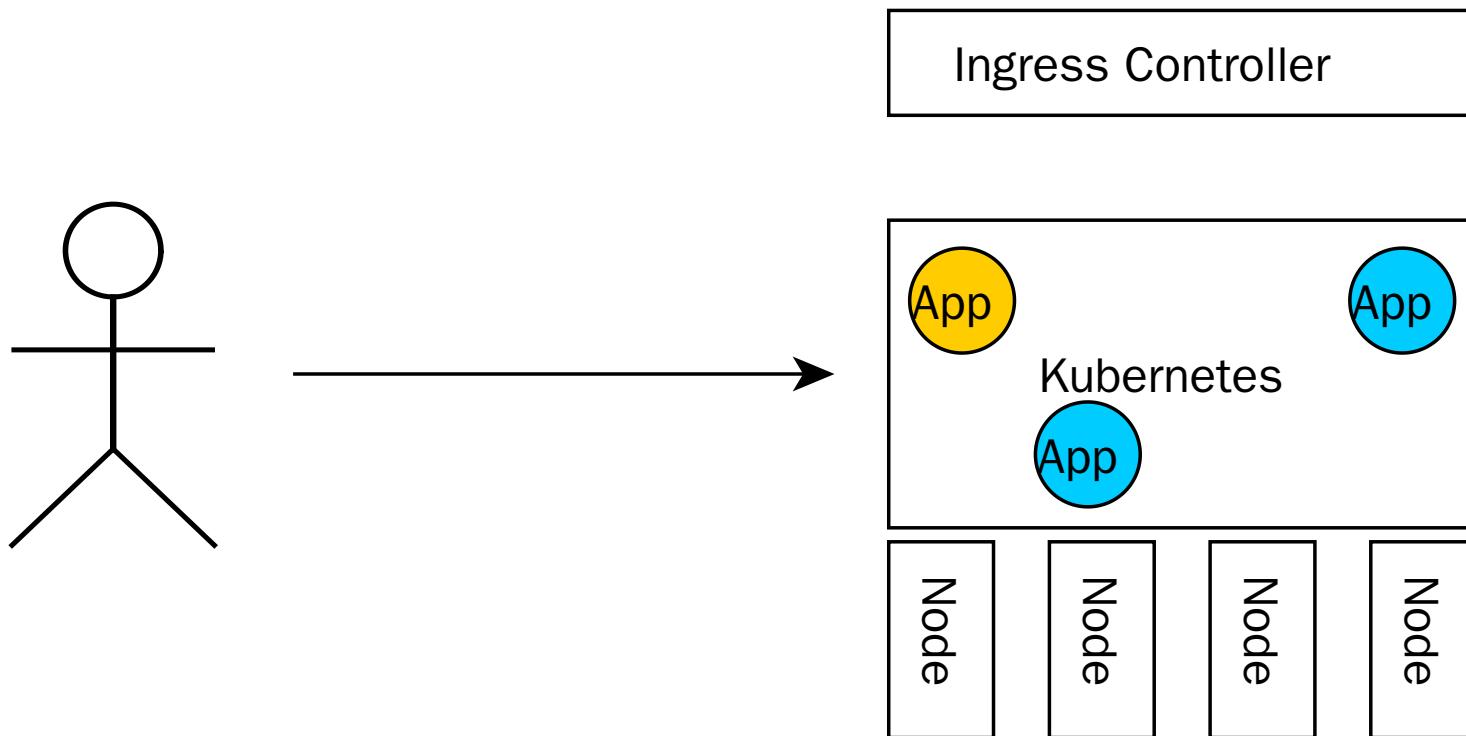
`kubectl --replicas=1 -f app-srv-dpl.yaml`

ROLLING UPDATES!

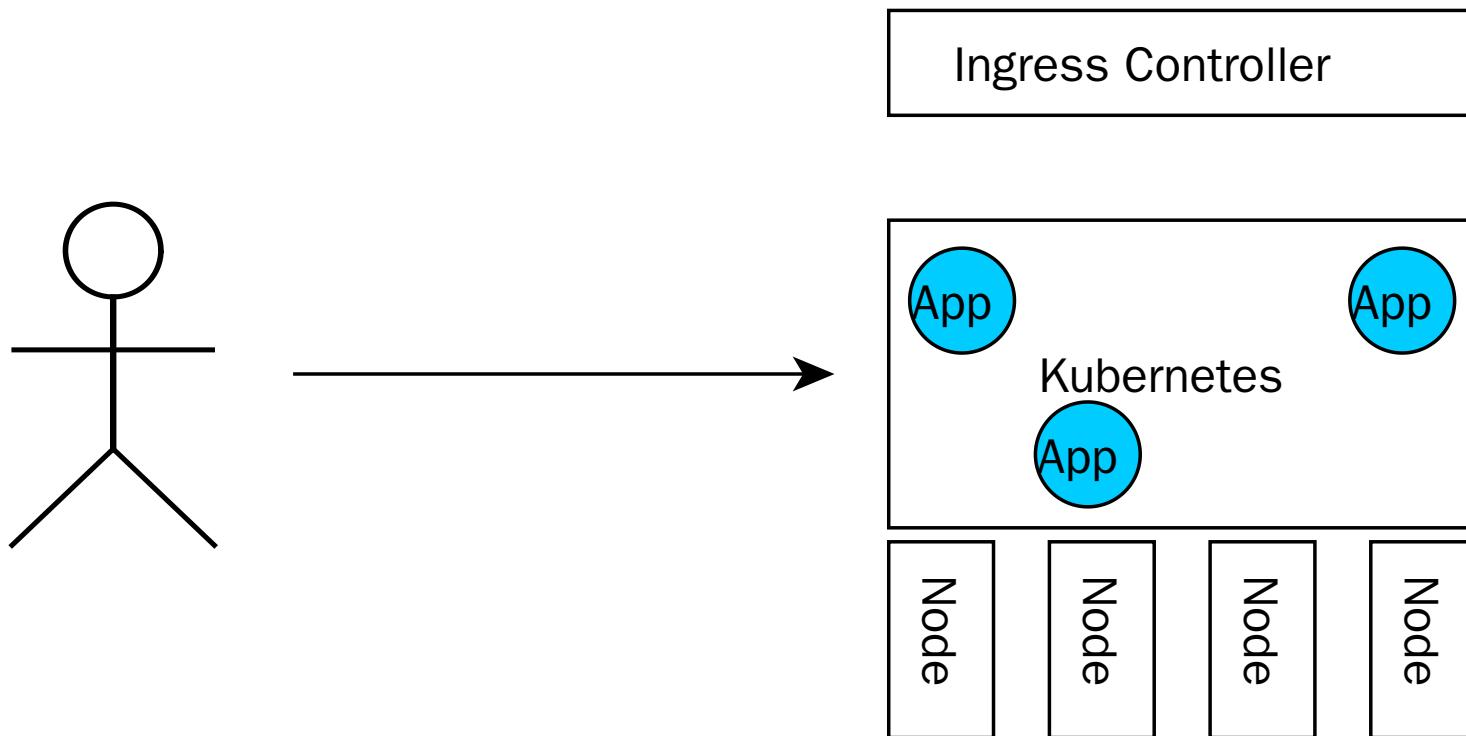


```
kubectl set image deployment/app app=app:v2.0.0
```

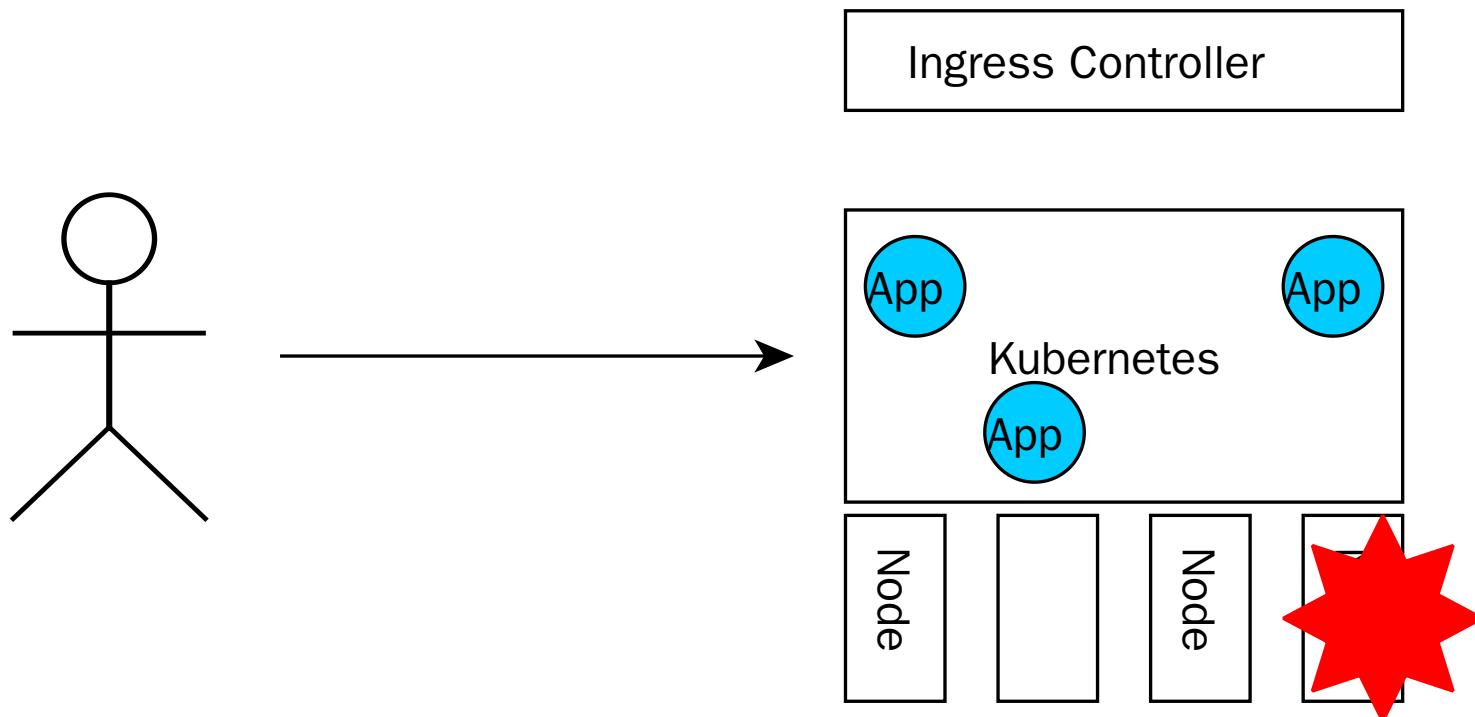
ROLLING UPDATES!



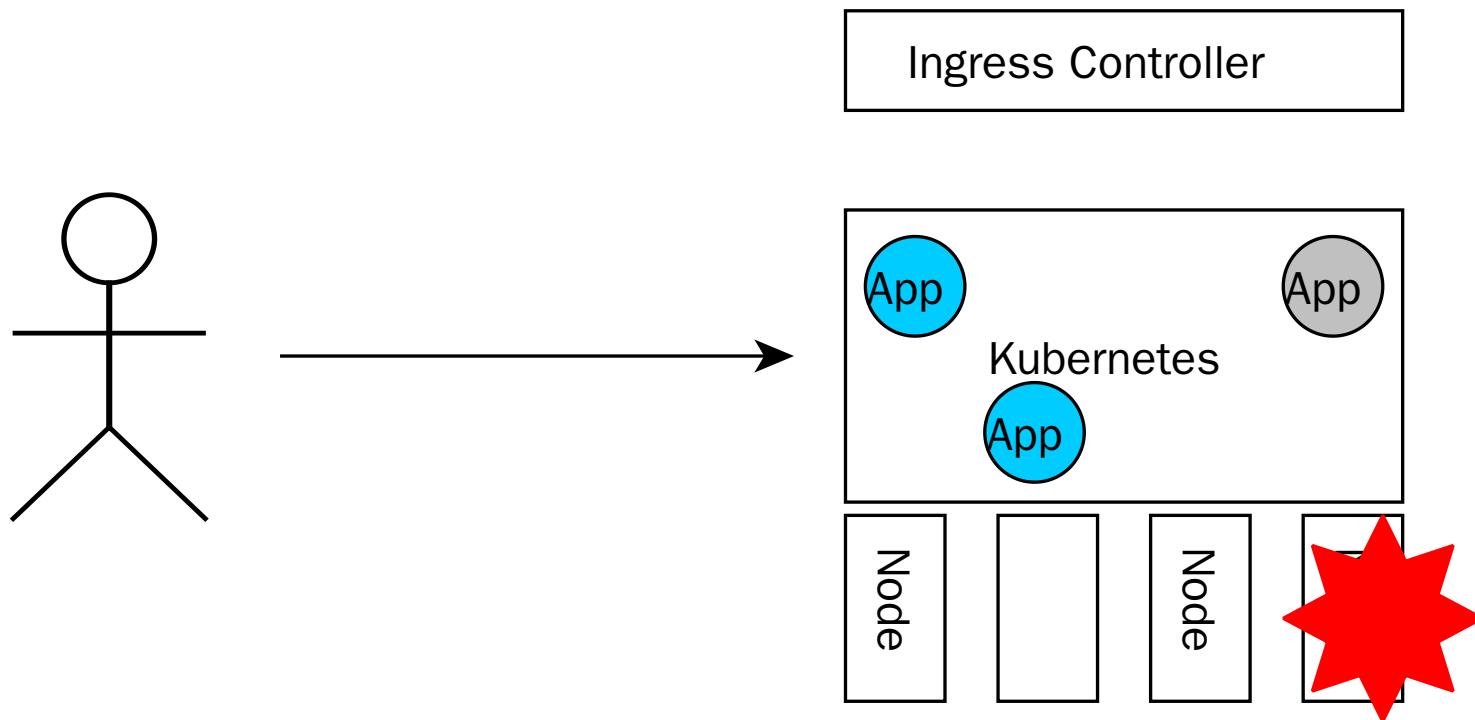
ROLLING UPDATES!



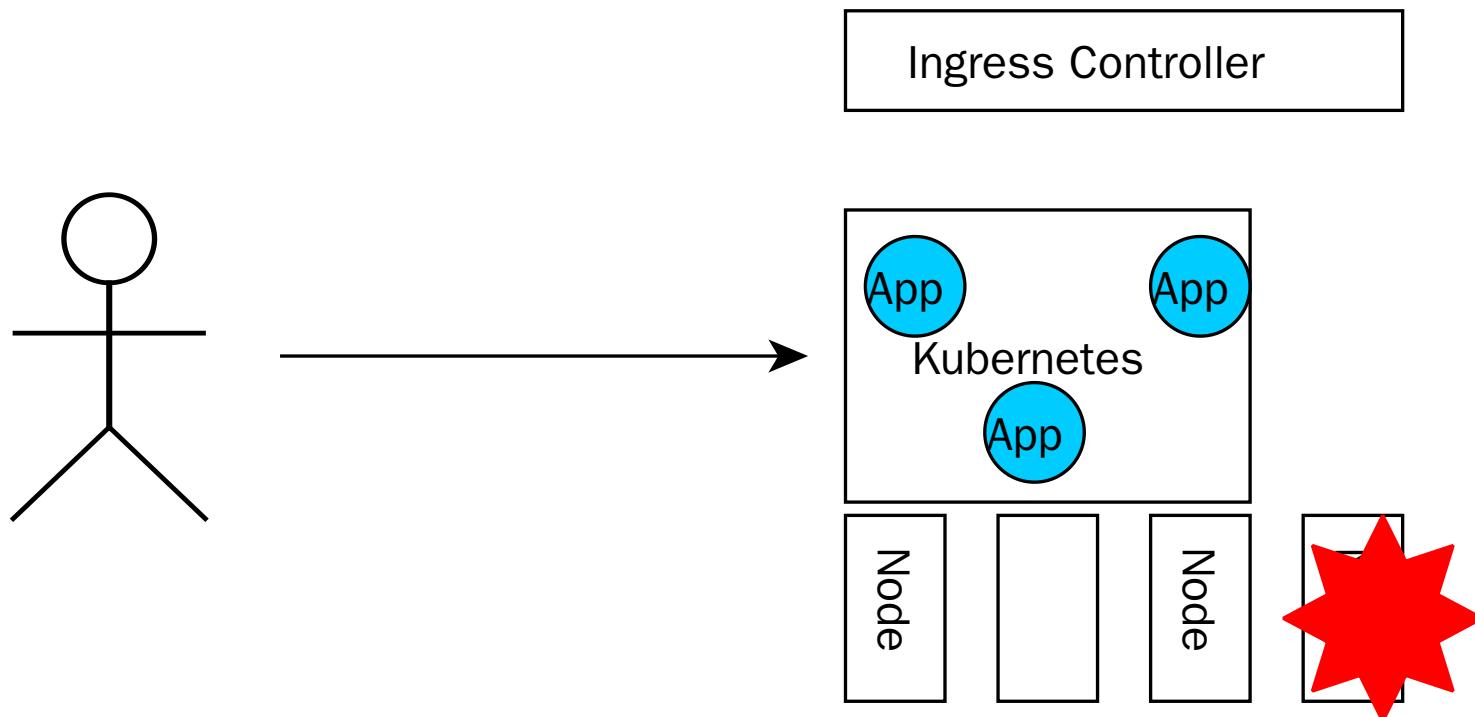
ODPORNOŚĆ



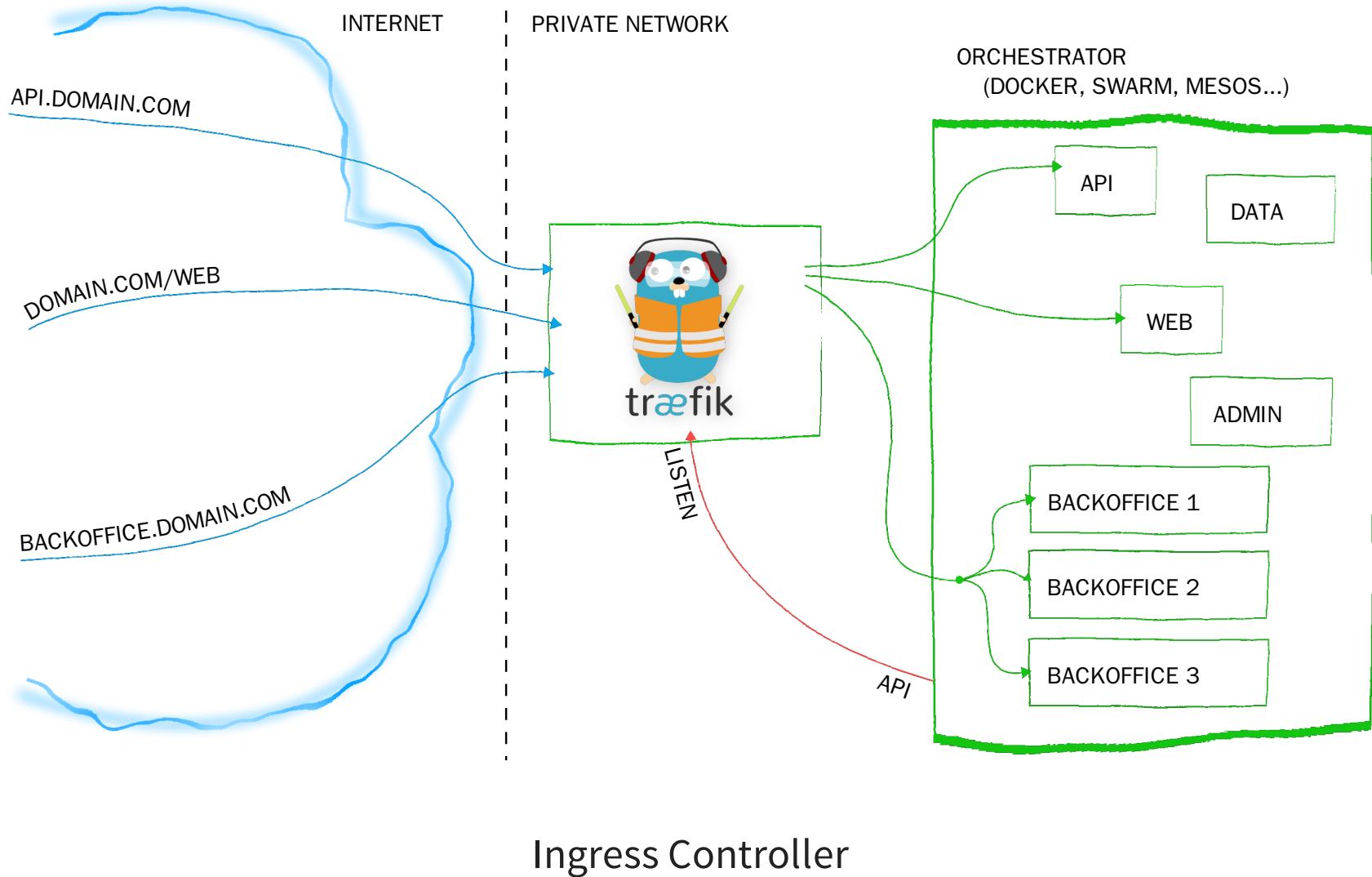
ODPORNOŚĆ



ODPORNOŚĆ



HOW GET USER REQUESTS?



INGRESS

Pattern

api.smacc.io/v1/users

Target App Service

users-v1

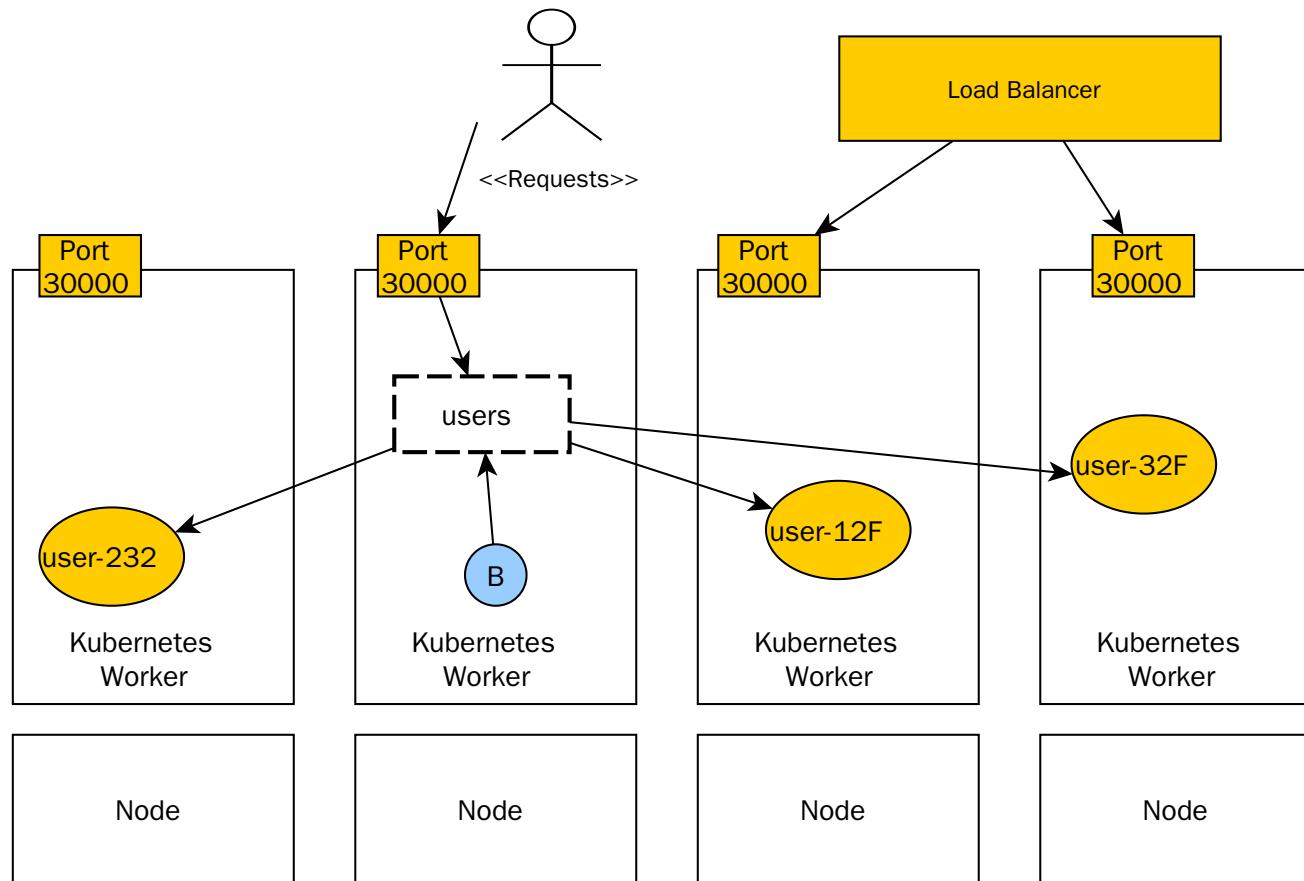
api.smacc.io/v2/users

users-v2

smacc.io

web

LOAD BALANCING



SERVICE DISCOVERY

- nazwy wstrzykiwane w DNS:

```
curl http://users/list
```

- etykiety:

```
name=value
```

- annotacje:

```
prometheus.io/scrape: "true"
```

SERVICE DISCOVERY

- luźno powiązane komponenty
- auto-wiring z logigem i monitoringiem

PLUG-AND-PLAY

- traefik / Nginx / Haproxy / Envoy
- monitoring: prometheus
- logging
- audit checks
- ...

NAJLEPSZE

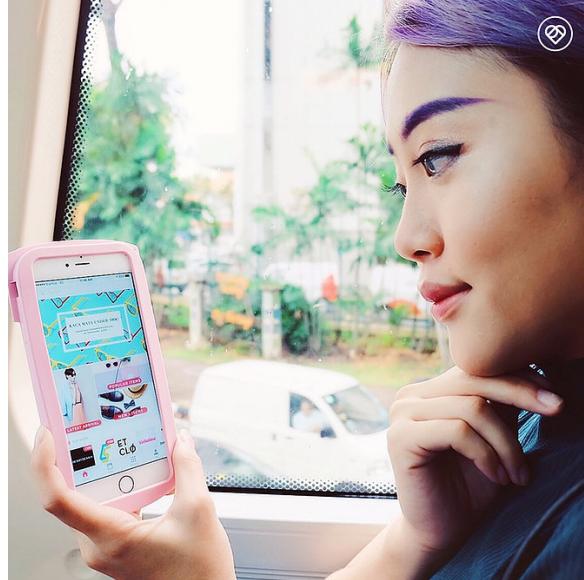
Wszystko w GIT:

- pliki k8s yaml
- integracja z monitoring, alarming, ... - anotacje
- integracja z ingress-controller
- ...
- Devs mogą, *prawie* zapomnieć o infrastrukturze

DevOps Culture Dream!

LYKE

- 50k+ użytk
- 2M downloads
- Top 10 Fashion Apps w Google Play Store
- Wszystko testowane A/B
- Szybki rosnący i data-driven



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

LYKE

- 50+ VMs - Amazon, 1 VM - 1 App, bezczynne
- Puppet, semi-manualny proces deploymentu
- Strach
- Zapomniene komponenty
- Problemy z wydajnością

ROZWIAZANIE

1. Uprościć infrastruktue
2. Wprowadzić Continuous Deployment
3. Zmienić organizację pracy

patrz: Conway's law

UPROŚCIĆ

1. Kubernetes z Google Kubernetes Engine
2. Terraform dla wszystkiego innego i AWSa

PLUG-AND-PLAY

1. Prometheus, AlertManager i Grafana
2. Elasticsearch-Fluentd-Kibana
3. Google Identity-Aware-Proxy do ochrony paneli narzędzi
4. 3rd party: statuscake i opsgenie

W krótkim czasie mamy już działający system!
Skraca czas nauki nowej technologii.

TRAVISCI

1. Tests
2. Zbudowanie docker
3. Deploy docker do Google Container Registry
4. Deploy docker to k8s
5. No other config applied
6. Repo independent

GIT REPO

```
| - tools
|   | - kube-service.yaml
|   \ - kube-deployment.yaml
|
| - Dockerfile
| - VERSION
\ - Makefile
```

Makefile

```
SERVICE_NAME=v-connector
GCP_DOCKER_REGISTRY=eu.gcr.io
test: test_short test_integration

run_local: # !!!!

docker_build: docker_push

kube_create_config:

kube_apply:

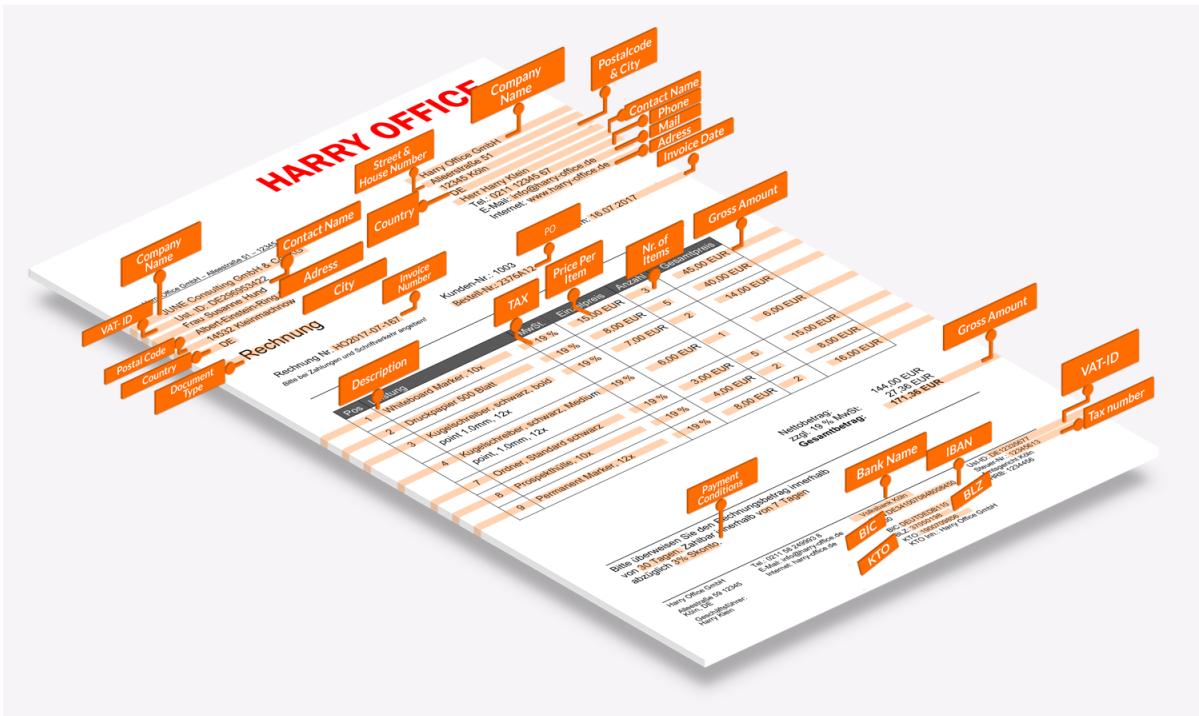
kube_deploy:
```

Copy&Paste from the project to project

SMACC

Hypatos

SMACC.io



Klienci

Deutsche Bank



BMB
巴三巴
GRUPPE

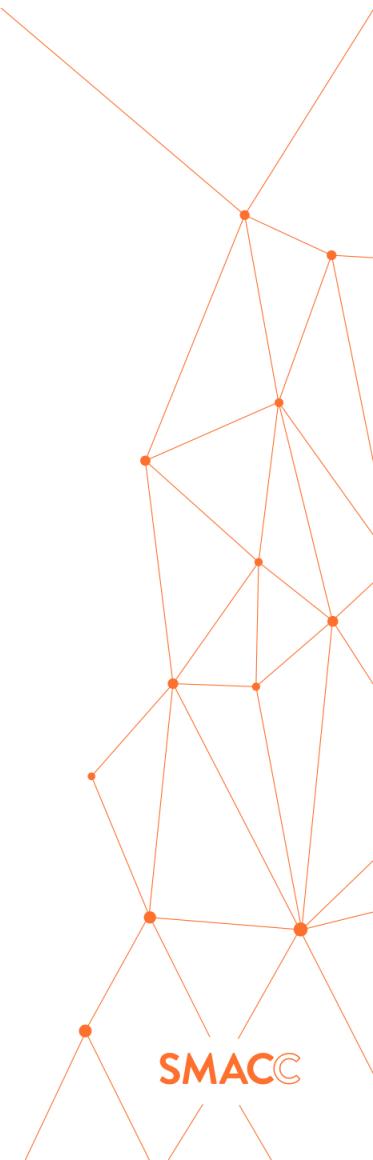


Deloitte.



accenture

McKinsey&Company



SMACC

SMACC

- Komponenty na VM w AWS, próby z AWS ECS :/
- Self-hosted K8S na ProfitBricks
- Microsoft ScaleUp, welcome Azure
- Luckily - AKS

KUBERNETES Z AKS

- Niezależność od Azure
- Nasz OnPrem = Nasz OnCloud
- Konsolidacja naszych micro-servicesów
- Plug and play, np., monitoring z prometheusem

UPROŚCIĆ

- az aks CLI do k8s - README.rst
- Terraform dla wszystkiego innego
- 1Password i gopass.pw

Terraform też do AWSa

NOWE DOŚWIADCZENIE

- deweloperzy nie lubią wszystkiego poza kodowaniem
- migracja z PB do AKS była trudna

ROZWIAZANIE

- uproszczenie procesu dla dev
- wszystko do CD jako c&p
- ukryć generacje plików k8s
- deployment na *tag*

Similar to the [Kelsey Hightower approach](#)

Makefile

```
| - tools
|   | - Makefile
|   | - kube-service.yaml
|   \- kube-deployment.yaml
|
| - Dockerfile
\- Makefile
```

CONTINUOUS DEPLOYMENT

- Github
- TravisCI
- hub.docker.com
- AKS

PROCESS

1. git tag and push

PROCESS

2. tag ➔ deployment do stagingu
3. tag ➔ Pull-Request do brancha produkcji
4. merge Pull-Request ➔ deployment do produkcji

KUBERNETES

- Pure, generated, kubernetes config
- 2x kubernetes operators

NASTĘPNE KROKI

- Acceptance tests na każdy deployment
- Skalowanie trenowania naszych modeli ML
- Budowa toola do deploymentu w oparciu o [missy](#)
- Evaluacja Istio

K8S - Linux

- Kubernetes nie jest "srebną kulą" ale dość blisko
- Wspólne środowisko dla onPrem i onCloud
- Upraszczza procesy, uwalnia zasoby
- Plug-and-play

K8S - Linux

- The biggest asset - the API
- Z service discovery, doskonała platforma do integracji
- Z kubevirt - alternatywa dla Openstacka

DZIĘKUJĘ. PYTANIA?

ps. We are hiring.

```
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)]
```



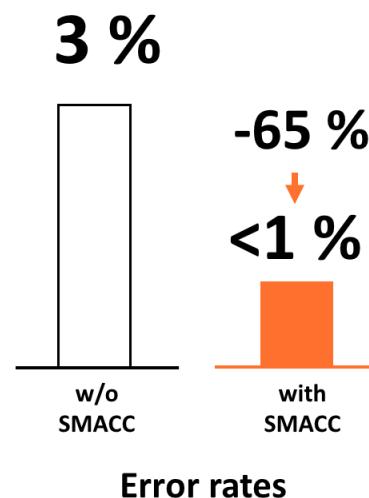
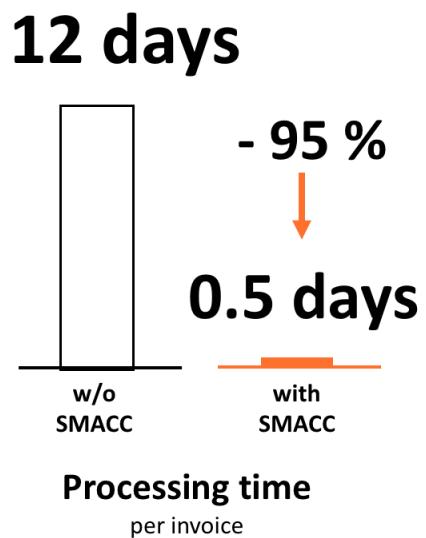
HIRING

- Senior Polyglot Software Engineers
- Experienced System Engineers
- Front-end Engineers
- 1 Data-Driven Product Manager

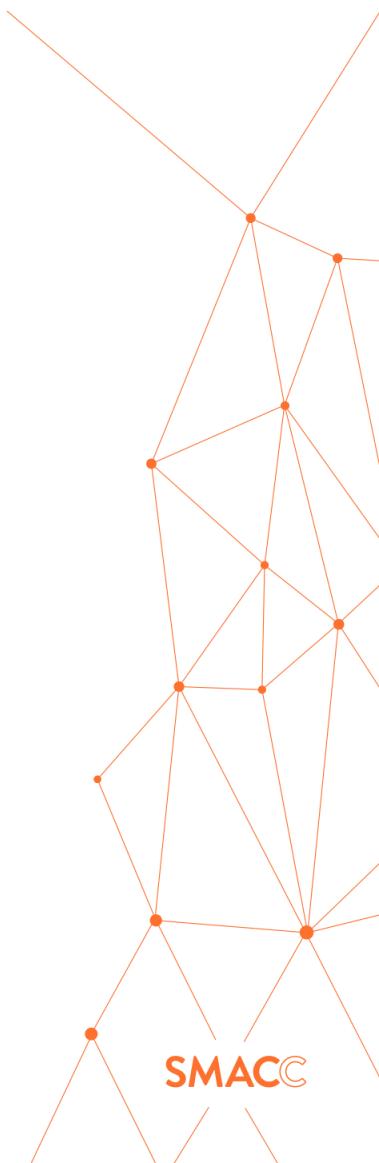
Apply: hello-warsaw@smacc.io,
Questions? wojciech.barczynski@smacc.io, [FB](#) or [LI](#)

We will teach you Go if needed. No k8s or ML, we will take care of that.

Korzyści Naszego Rozwiazania



Quelle: PWC Study; SMACC customer results

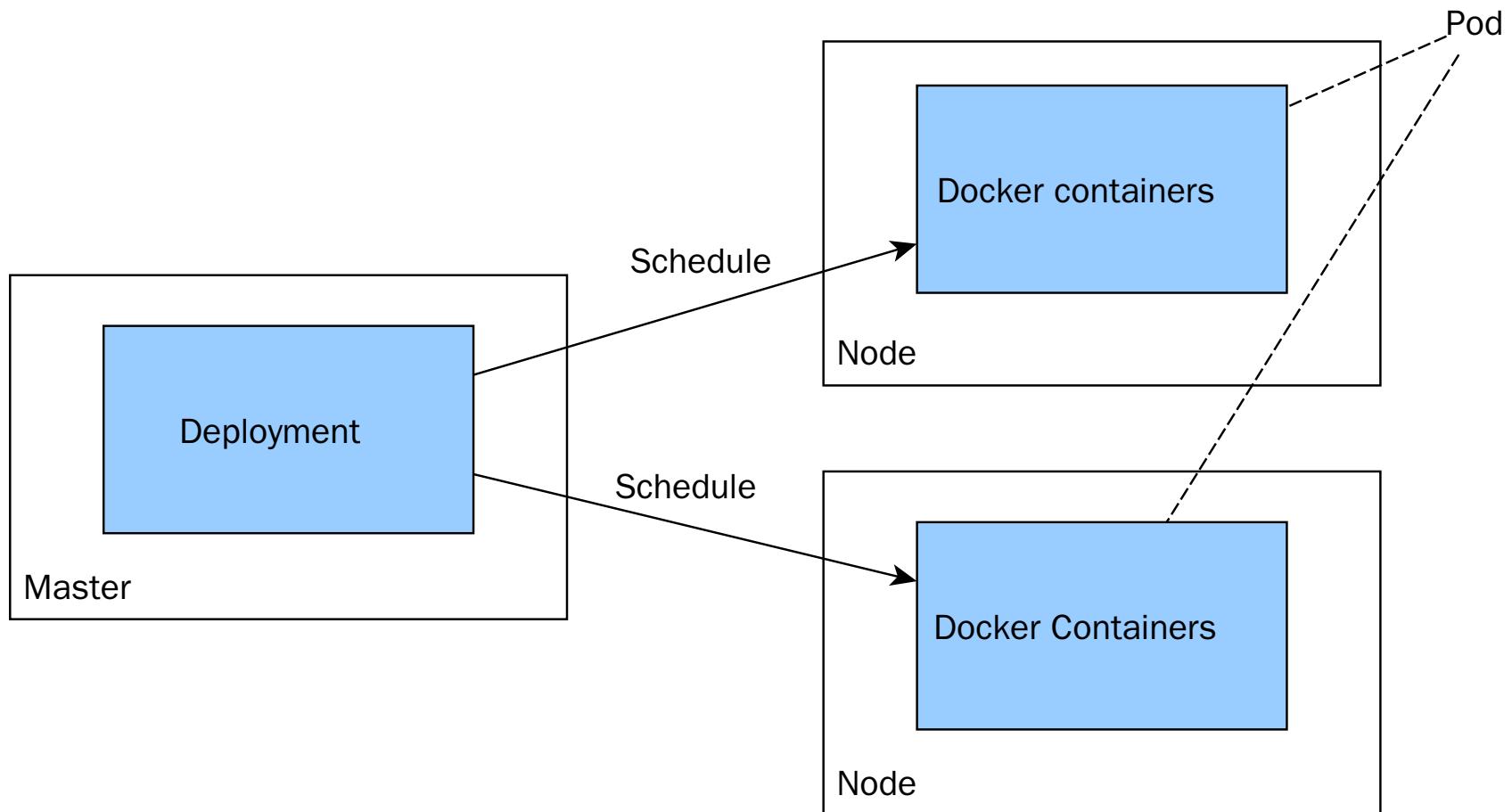


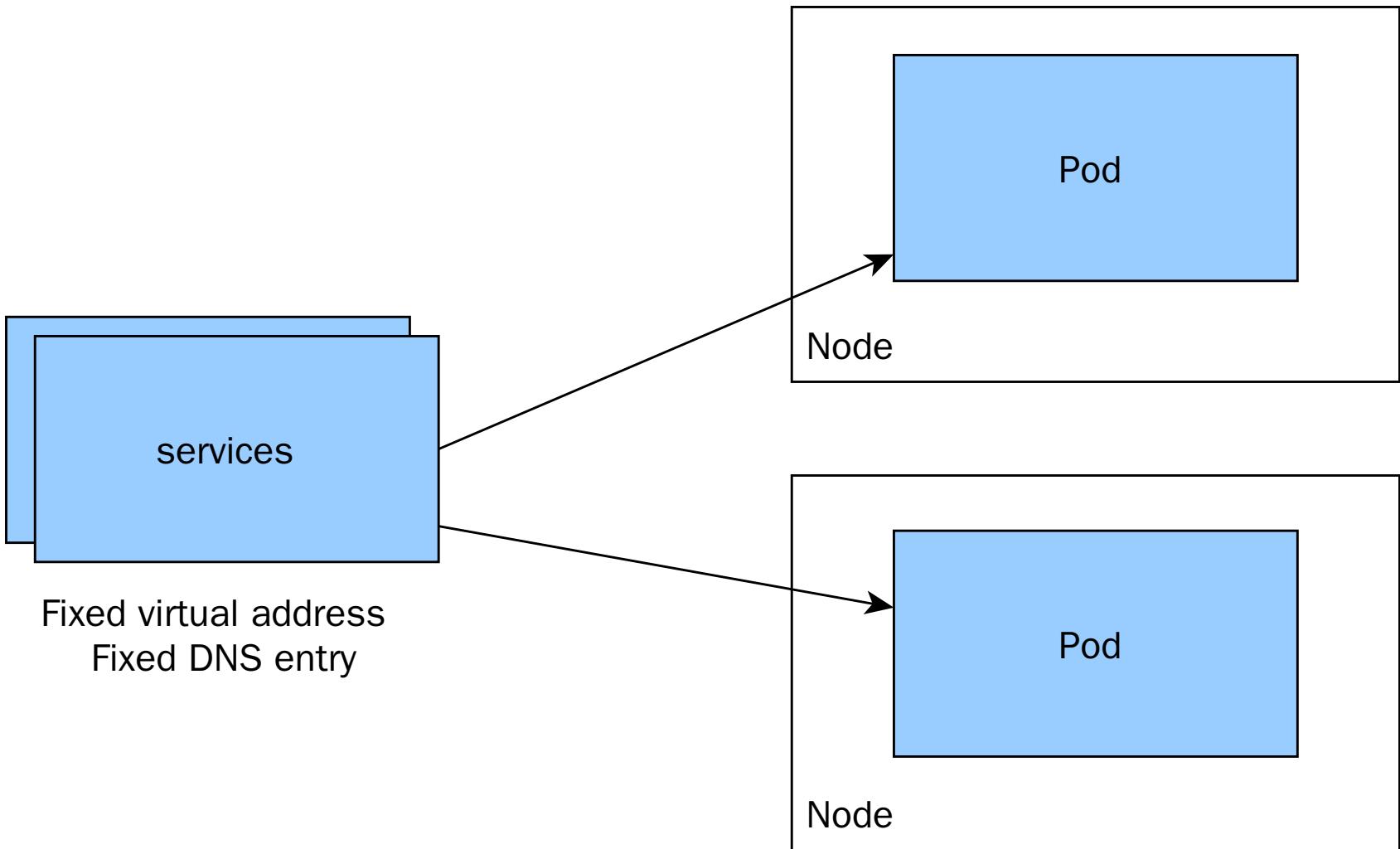
BACKUP SLIDES

```
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)]
```



KUBERNETES CONCEPTS



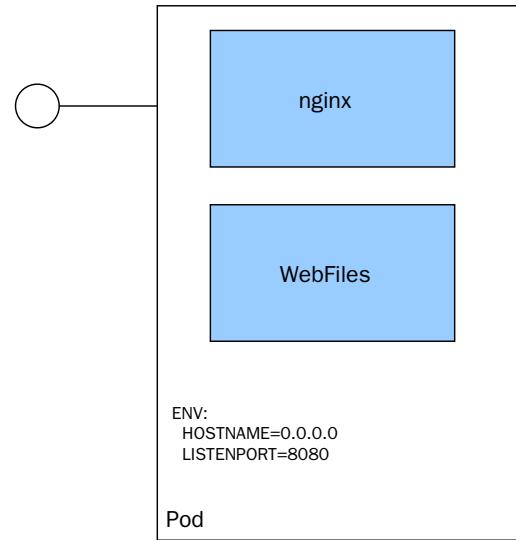


BASIC CONCEPTS

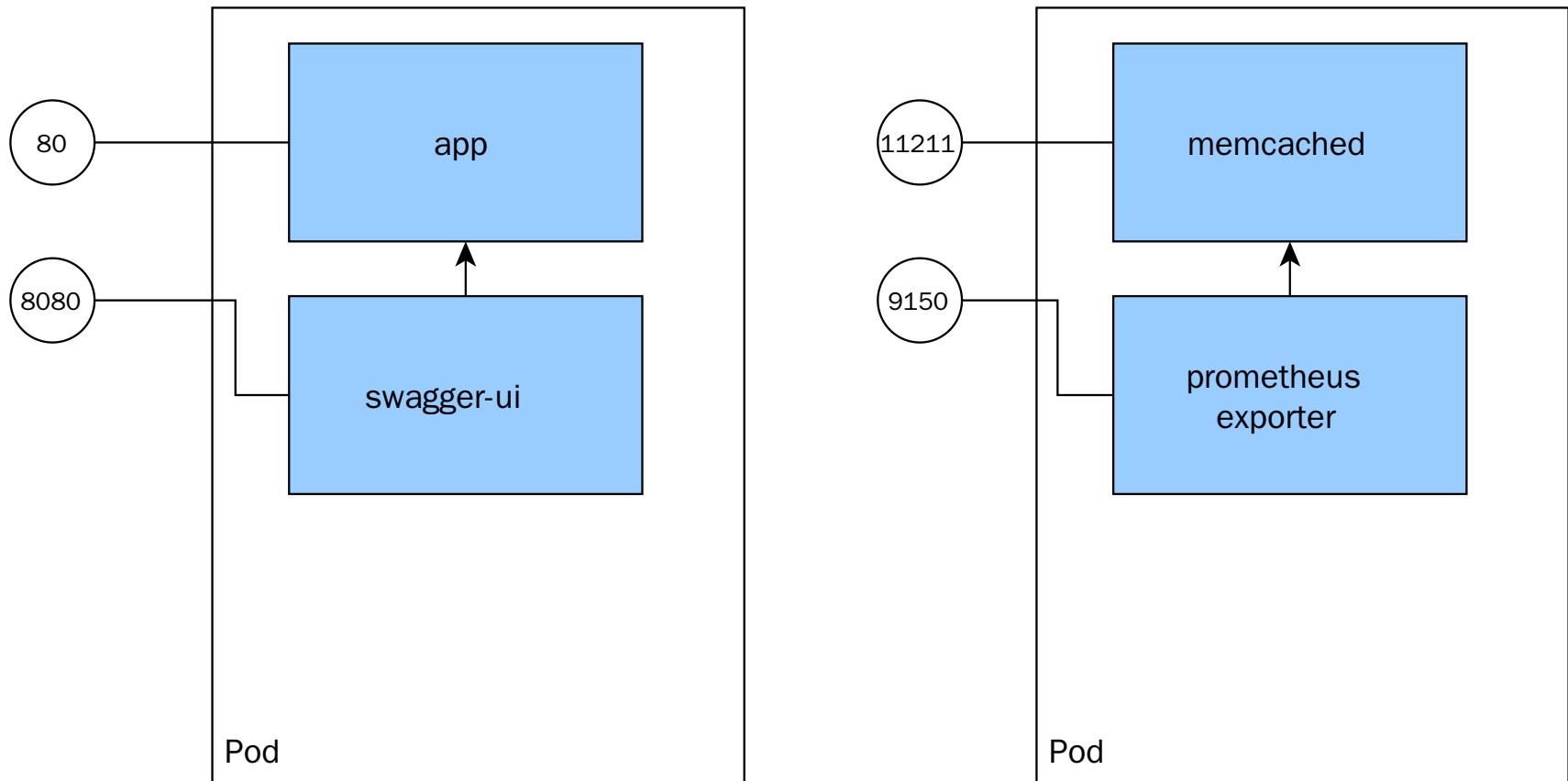
Name	Purpose	
Service	Interface	Entry point (Service Name)
Deployment	Factory	How many pods, which pods
Pod	Implementation	1+ docker running

PODS

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



SIDE-CARS



ROLLING RELEASE WITH DEPLOYMENTS

