Kubernetes

overhype or the next big thing?

Adam Gołąb

- Full-Stack JS Developer @ Brainhub
- #TeamReact
- #DevOps

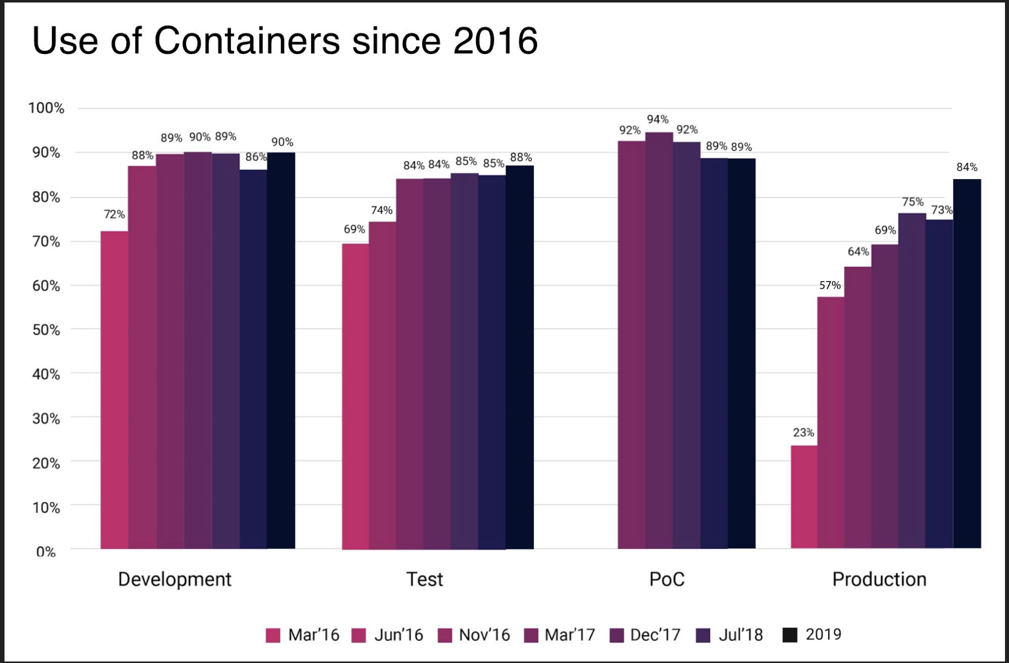
adam-golab

@adam_devops

The presentation

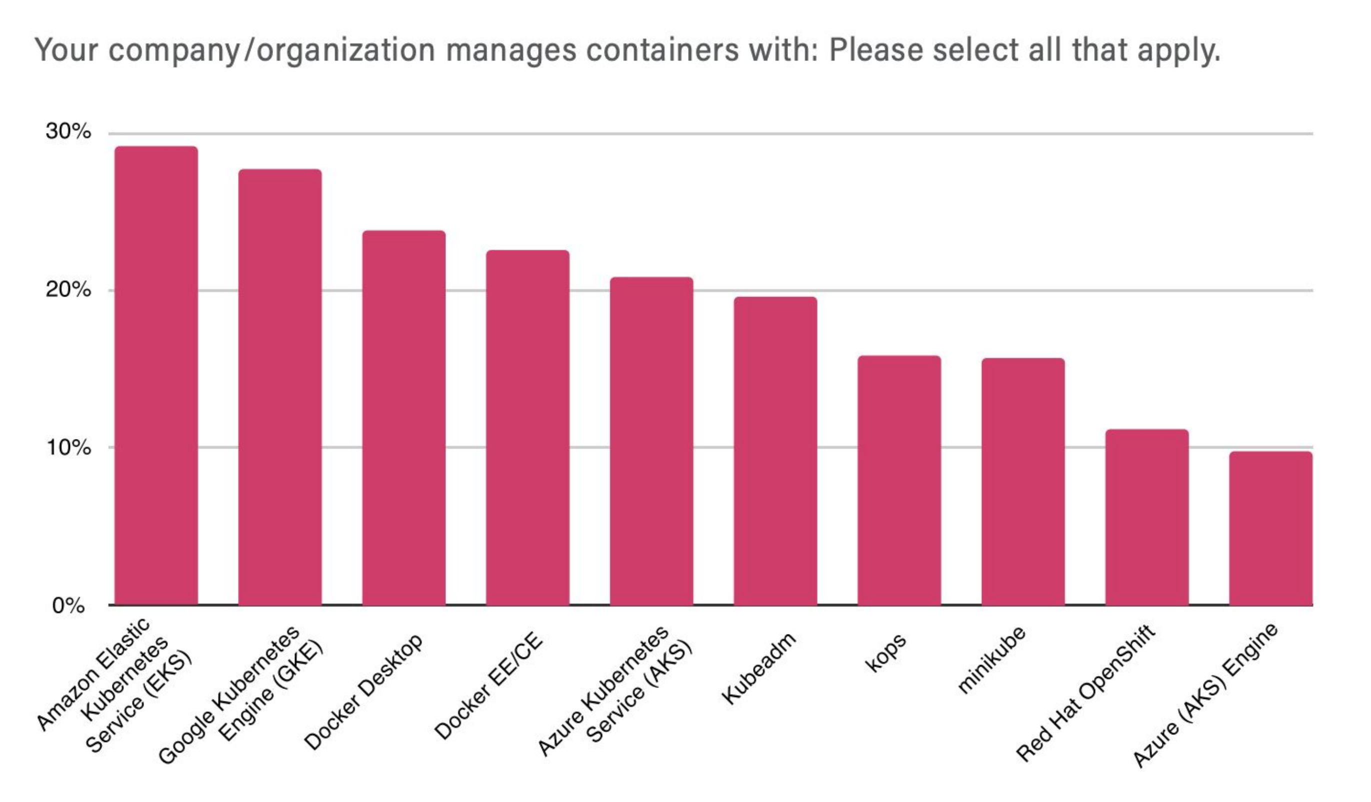
- What hype? 📈
- Some benefits 💰
- Kubernetes from the ground up 🧩
- Live demo 🔥
- Deploy to production 😂

What hype?

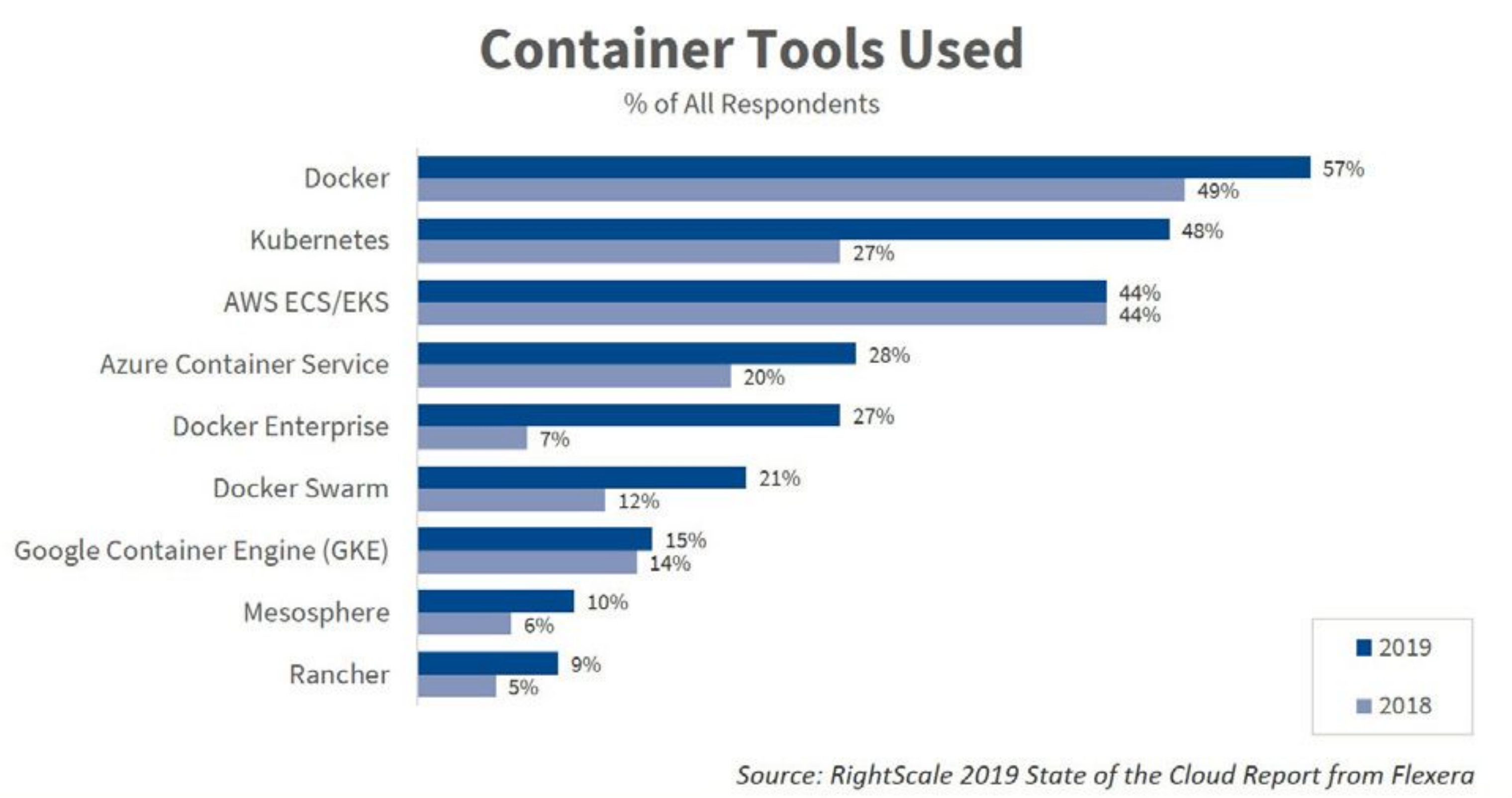


Source: 2019 CNCF Survey

What hype?


Your company/organization manages containers with: Please select all that apply.
30%
20%
10%
0%
Amazon Elastic Kubernetes Service (EKS)
Google Kubernetes Engine (GKE)
Docker Desktop
Docker EE/CE
Azure Kubernetes Service (AKS)
Kubeadm
kops
minikube
Red Hat OpenShift
Azure (AKS) Engine

What hype?



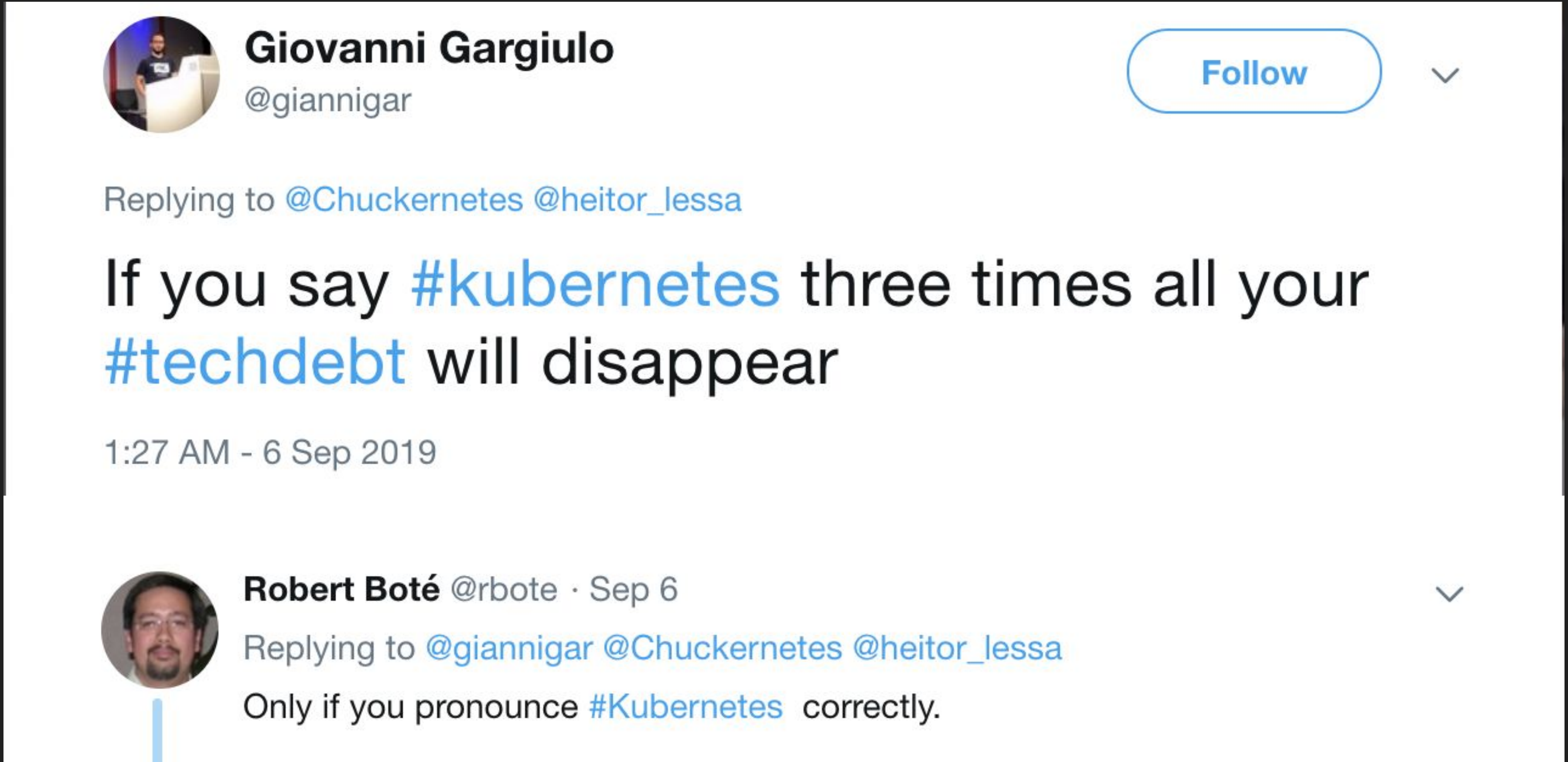
Source: RightScale 2019 State of the Cloud Report from Flexera

It’s everywhere

- Google Kubernetes Engine (GKE)
- Amazon Elastic Kubernetes Service (EKS)
- Azure Kubernetes Service (AKS)
- Digital Ocean
- Alibaba Cloud Container Service for Kubernetes (ACK)
- OVH
- Many more
- Self hosted

Benefits?

- Vendor independence *(kind of)*
- Stateless units (Pods) *(functional programming)*
- Scalability
- High Availability - Health checks and self-healing
- Declarative model
- Automated rollouts and rollbacks *(green-blue deployments)*
- Infrastructure as Code (GitOps)
- Serverless
- **Keeps consistent architecture**



Giovanni Gargiulo
@giannigar

Replying to @Chuckernetes @heitor_lessa

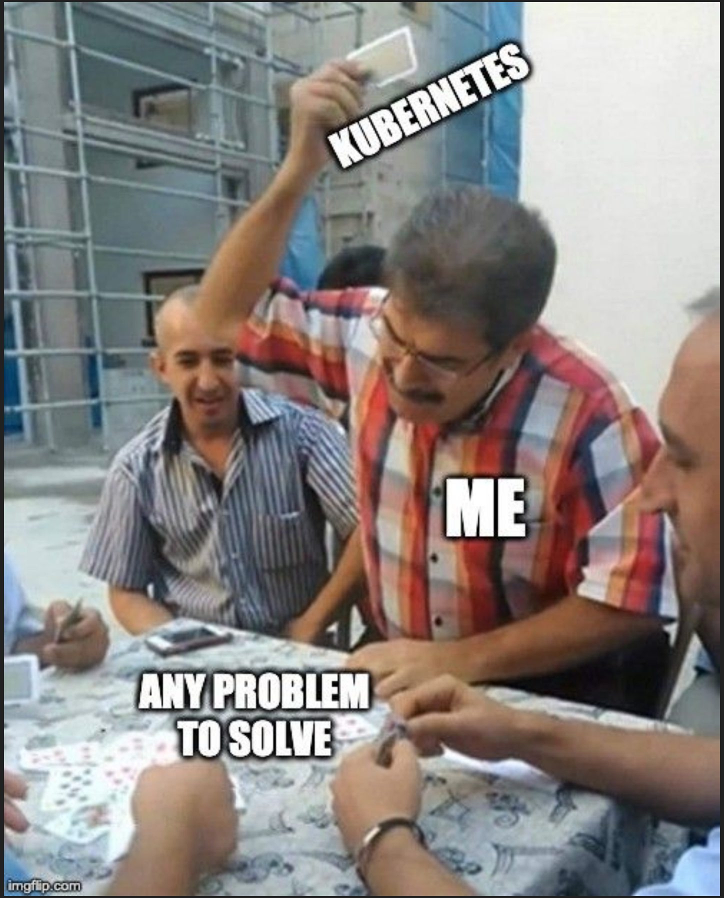
If you say #kubernetes three times all your #techdebt will disappear

1:27 AM - 6 Sep 2019

Robert Boté @rbote · Sep 6

Replying to @giannigar @Chuckernetes @heitor_lessa

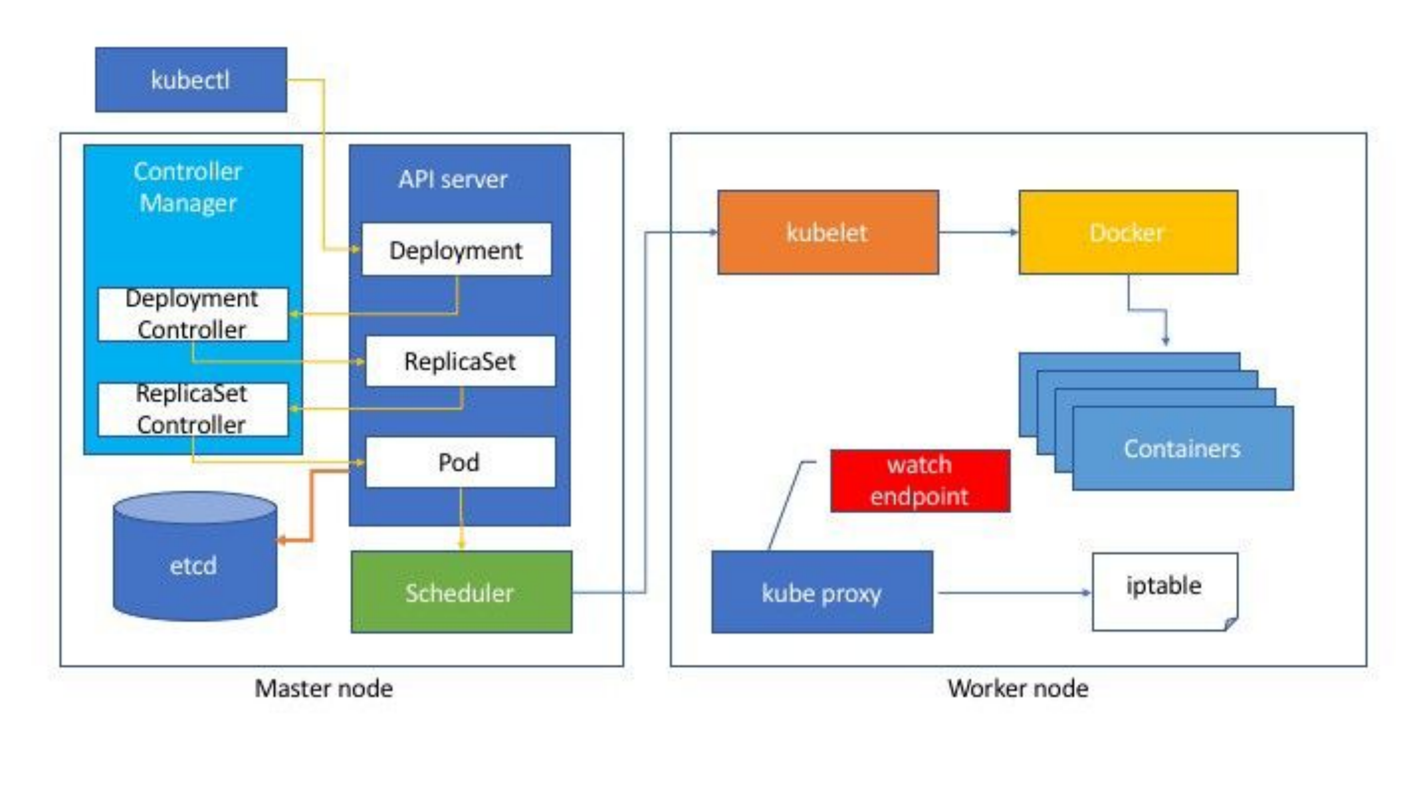
Only if you pronounce #Kubernetes correctly.


KUBERNETES
ME
ANY PROBLEM TO SOLVE
imgflip.com

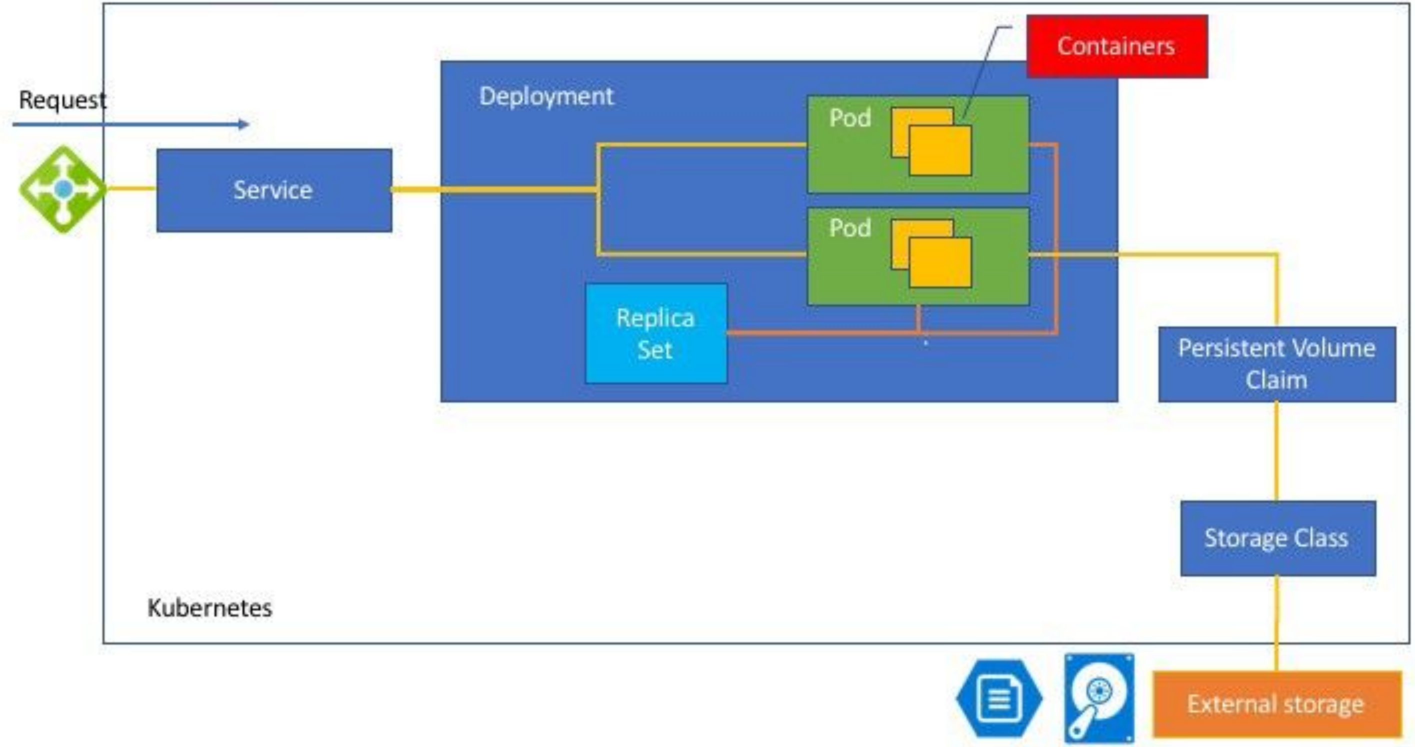
What exactly is Kubernetes (k8s)?

- **Kubernetes** (*κυβερνήτης*, Greek for "governor", "helmsman" or "captain")
- open-source container-orchestration system for automating application deployment, scaling, and management
- originally designed by **Google** *(Google Borg)*
- maintained by the **Cloud Native Computing Foundation**.
- Initial release: **7 June 2014**

Architecture



Kubernetes Objects



Request

Service

Deployment

Containers

Pod

Pod

Replica Set

Persistent Volume Claim

Storage Class

Kubernetes

External storage

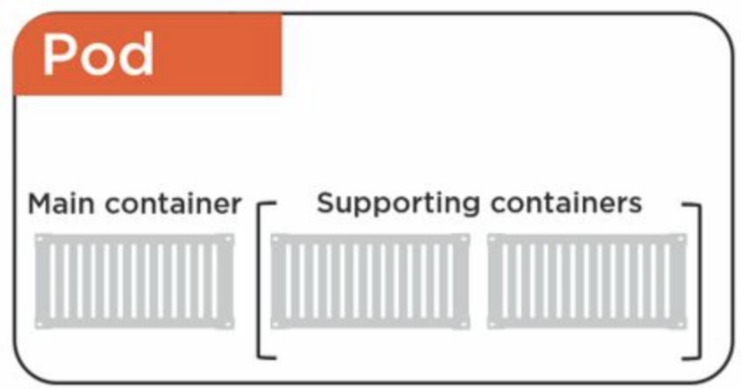
Pod

One or more docker containers

Share IPC namespace, shared memory, volumes, network stack (internal IP address)

Minimum unit of scaling in Kubernetes

Mortal (not healed but recreated when die)



Manifest

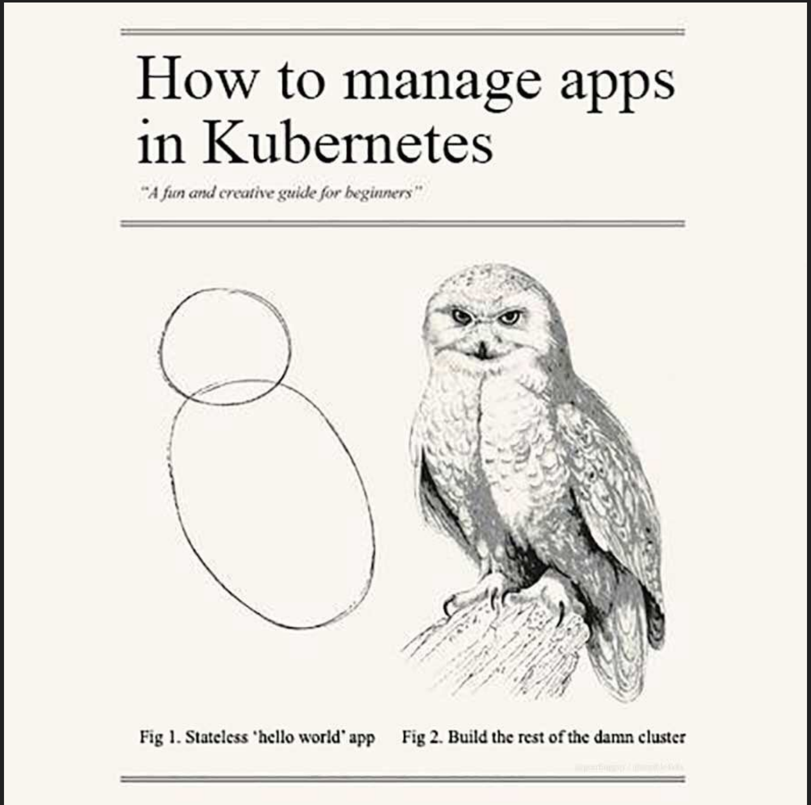
```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: <APPLICATION NAME>
spec:
  replicas: 1
  template:
    metadata:
      labels:
        app: <APPLICATION NAME>
    spec:
      containers:
      - name: <APPLICATION NAME>
        image: <DOCKER IMAGE>
        ports:
        - containerPort: 80
---
apiVersion: v1
kind: Service
metadata:
  name: <APPLICATION NAME>
spec:
  ports:
  - port: 80
  - targetPort: 80
  selector:
    app: <APPLICATION NAME>
```

🔥 Demo 🔥

github.com/adam-golab/kubernetes-overhyped

It works!

Deploy to production



How to manage apps in Kubernetes

"A fun and creative guide for beginners"

Fig 1. Stateless 'hello world' app

Fig 2. Build the rest of the damn cluster

k3s

Lightweight Kubernetes. k3s is a fully compliant production-grade Kubernetes distribution with some changes.

Minimum System Requirements:

- Linux 3.10+
- 512 MB of ram per server
- 75 MB of ram per node
- 200 MB of disk space
- x86_64, ARMv7, ARM64

k3s

Removes:

- Legacy and non-default features
- Alpha features
- In-tree cloud providers
- In-tree storage drivers
- Docker (optional)

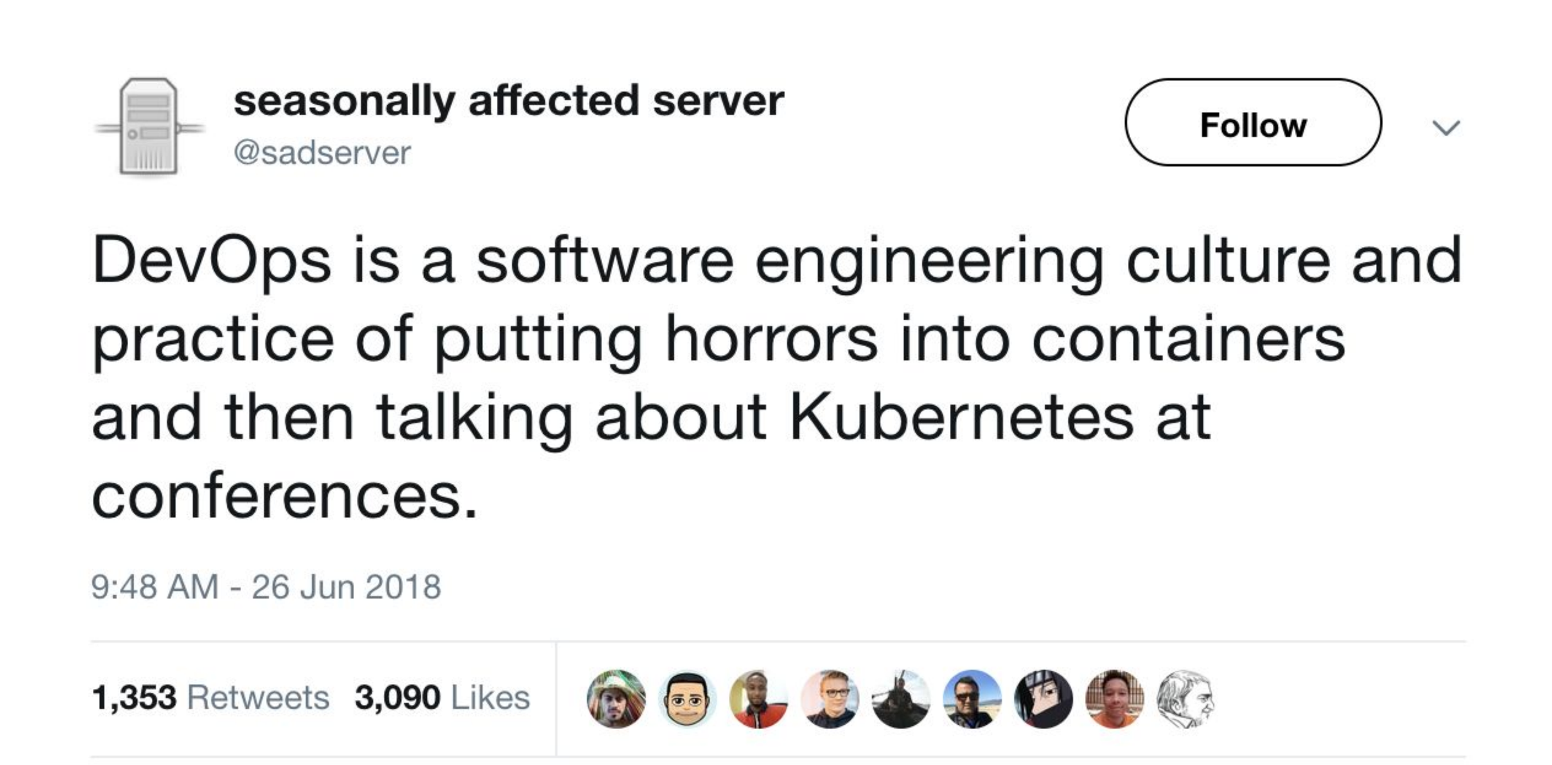
Adds:

- Simplified installation
- SQLite3, Postgres support in addition to etcd
- TLS management
- Automatic Manifest and Helm Chart management
- containerd, CoreDNS, Flannel

Deploy to production. When?

When you:

- manage a lot of microservices (definitely not created for monoliths)
- need to scale applications (services)
- can provide correct infrastructure
- have budget and time for maintaining it
- can switch to infrastructure as code approach
- need to avoid vendor lock-in
- are ready to educate team how to use it (introduce DevOps culture)

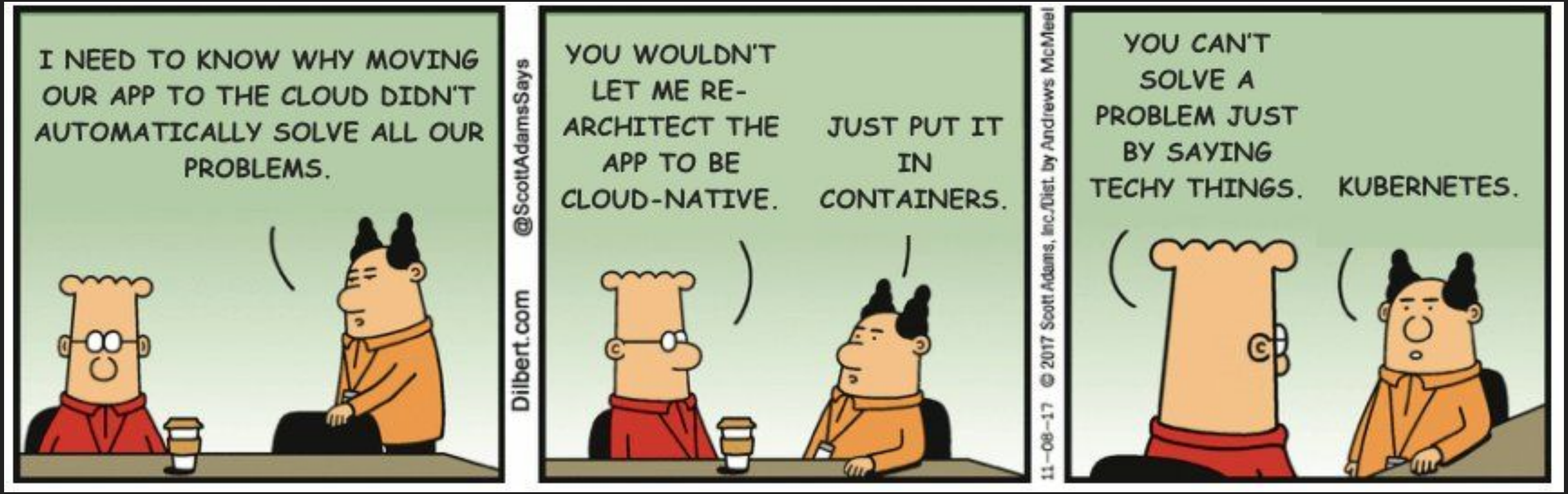


seasonally affected server
@sadserver

DevOps is a software engineering culture and practice of putting horrors into containers and then talking about Kubernetes at conferences.

9:48 AM - 26 Jun 2018

1,353 Retweets **3,090** Likes


I NEED TO KNOW WHY MOVING OUR APP TO THE CLOUD DIDN'T AUTOMATICALLY SOLVE ALL OUR PROBLEMS.
YOU WOULDN'T LET ME RE-ARCHITECT THE APP TO BE CLOUD-NATIVE.
JUST PUT IT IN CONTAINERS.
YOU CAN'T SOLVE A PROBLEM JUST BY SAYING TECHY THINGS.
KUBERNETES.
Dilbert.com @ScottAdamsSays
11-08-17 © 2017 Scott Adams, Inc./Dist. by Andrews McMeel

Useful links

Official documentation - https://kubernetes.io/docs/home/

CNCF Cloud Native Interactive Landscape - https://landscape.cncf.io/

Kubernetes Official Slack - https://slack.k8s.io/

Thanks

Questions?