Microsoft® Most Valuable Professional

LIVE STREAMING

Live Webinar

CI/CD Build Test and Deploy to Kubernetes Cluster on Azure Cloud

May 2, 2020, 12:00 BST

Mohamed Radwan

Principal DevOps Consultant

Blog: mohamedradwan.com

Group (A)

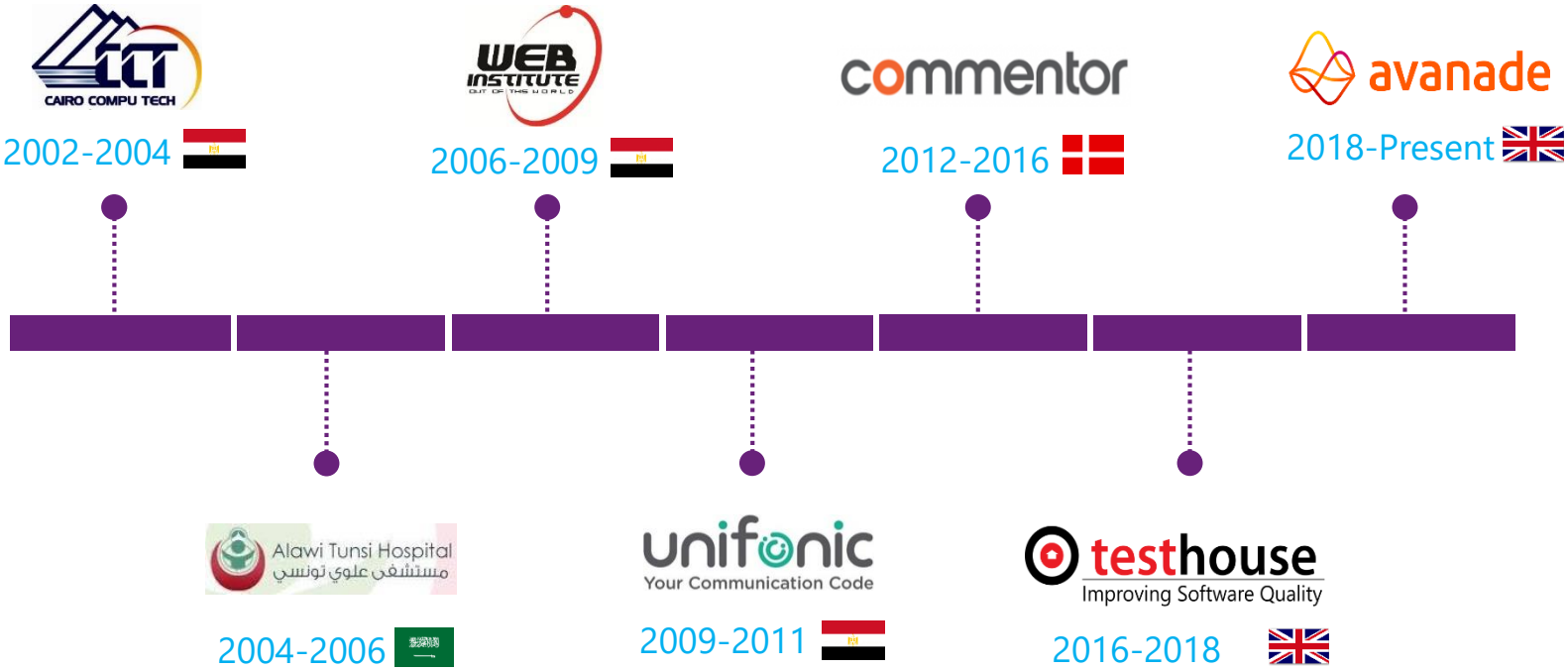
Mohamed Radwan
Principal DevOps

Principal Cloud DevOps Consultant

With 17+ Years of Experience, Helped 50+ companies around the globe

Blog: mohamedradwan.com
Twitter: @mradwan06

Work History & Locations

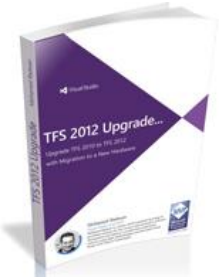


Guides, Open Source & Extensions

Migrate to VSTS

Agile Testing

DevOps Principles



Migrate TFS 2012

Upgrade TFS 2013

Upgrade TFS 2010

Integration Testing Framework

TFS Counter

Videos and Posts

YouTube

300+
VIDEOS

22.000+
VIEWS/MONTH

1.200.000+
VIEWS

5.000+
SUBSCRIBERS

10.000+
LIKS

WordPress

400+
BLOG POSTS

18.000+
VISIT/MONTH

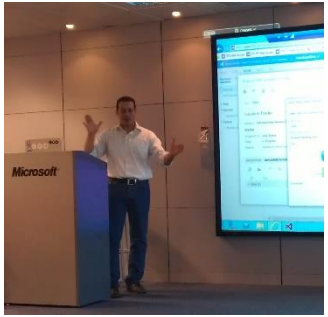
800.000+
TOTAL VIEWS

14.000+
UNIQUE
VISITS/MONTH

300+
REGISTE
R USERS

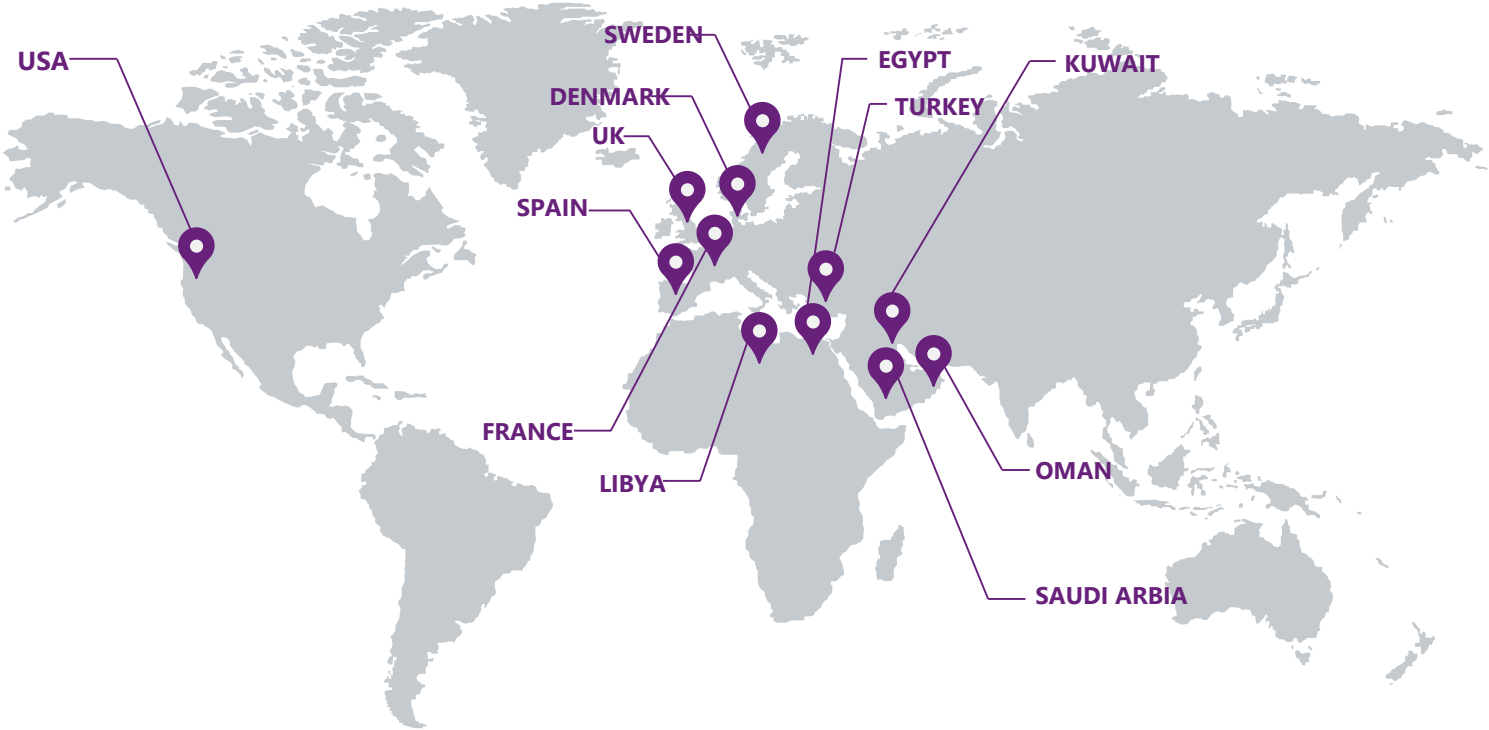
Events, Sessions & Workshops

100+ Sessions
5000+ Attendees



Global Experience

12+ Countries



Projects | Session/Events | Workshops | Training

Agenda

Agenda

- Quick Overview about Kubernetes
- Understanding Pod in Kubernetes
- Understanding Node in Kubernetes
- Quick Introduction to Azure Kubernetes Service (AKS)
- Kubernetes With and Without AKS
- Quick Introduction Azure Container Registry
- End-to-End CI/CD for Kubernetes on Azure
- **Demo**
- Create Azure DevOps project with a sample web app
- Install Azure Pipelines Kubernetes extension

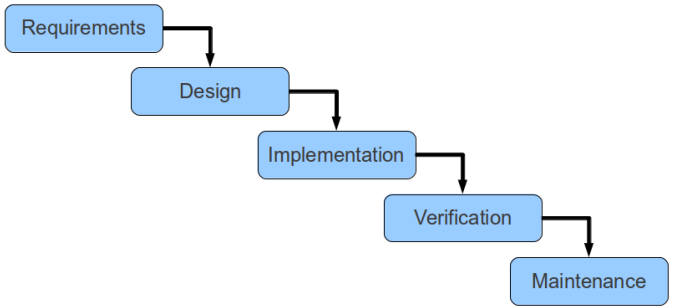
Continue on (P2)

Agenda (2)

- Create Azure Container Registry
- Create Azure Kubernetes Service
- Create Azure SQL Server and Azure SQL DB
- Configure Azure Build Pipelines to get image, build image then push image
- Configure Azure Release Pipelines to deploy DB and docker image to AKS
- Run the build and release pipelines and examine the web app
- Navigate to Kubernetes web dashboard for Kubernetes clusters

What is Kubernetes?

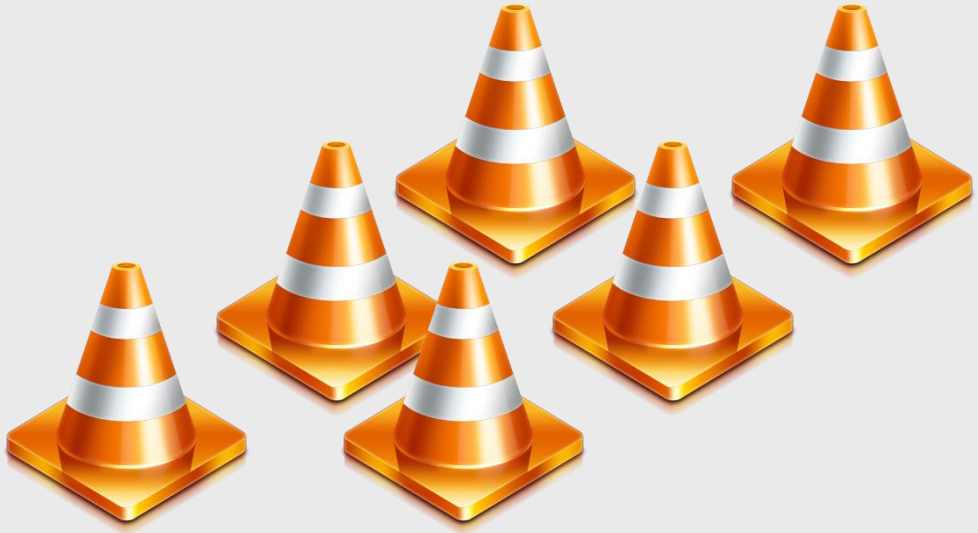
The story begins…..

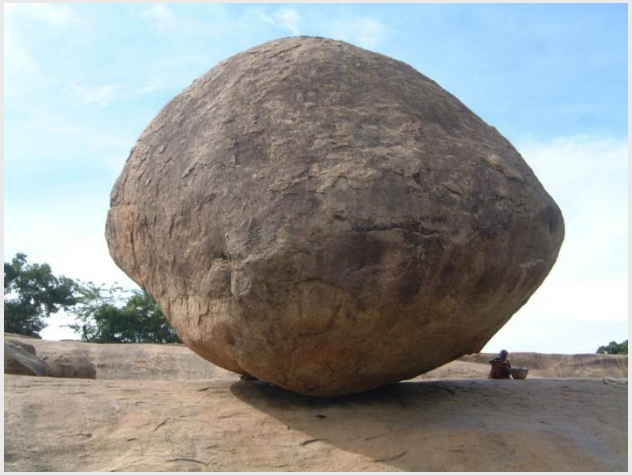


Waterfall

Agile/DevOps

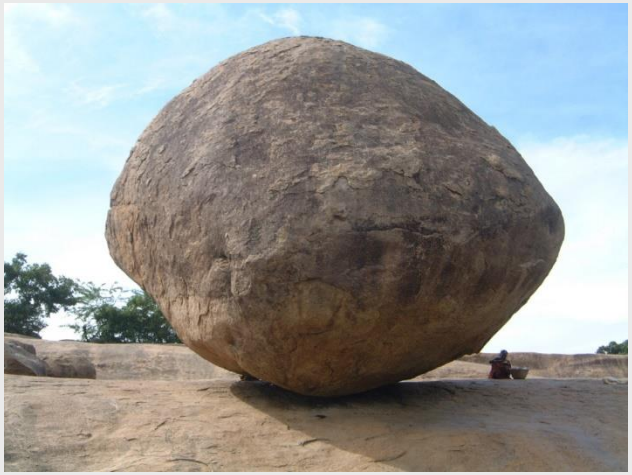
Obstacles and challenges

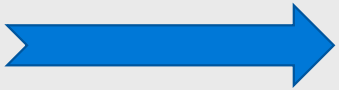




Monolithic Application

The Solution Was…….





Monolithic Application

Break it down into small pieces

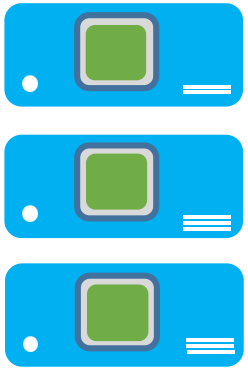
Monolithic Approach VS. Microservices

Monolithic Approach

(Web app or large service) usually has most of its functionality within a single process

App 1

And scales by cloning the whole app on multiple servers/VMs/containers.



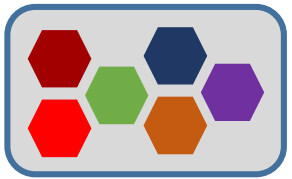
Deploy it to server/VM/container.

VS.

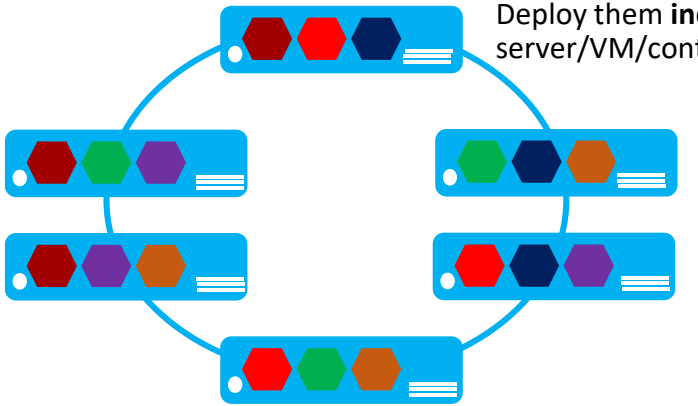
Microservices Approach

Segregates functionality into small independent services.

App 1

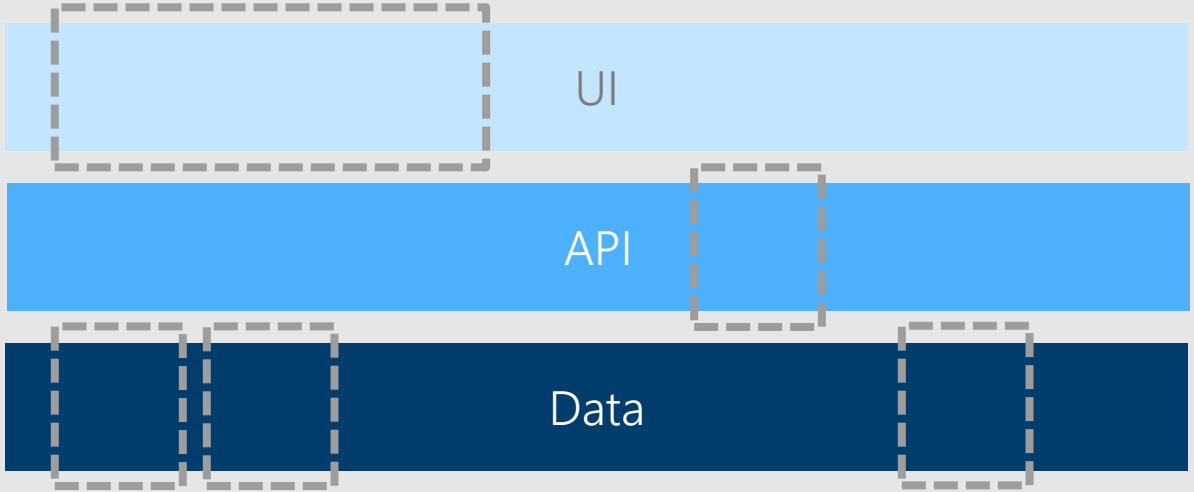


And scales out by **deploying independently** and replicating these services across servers/VMs/containers.

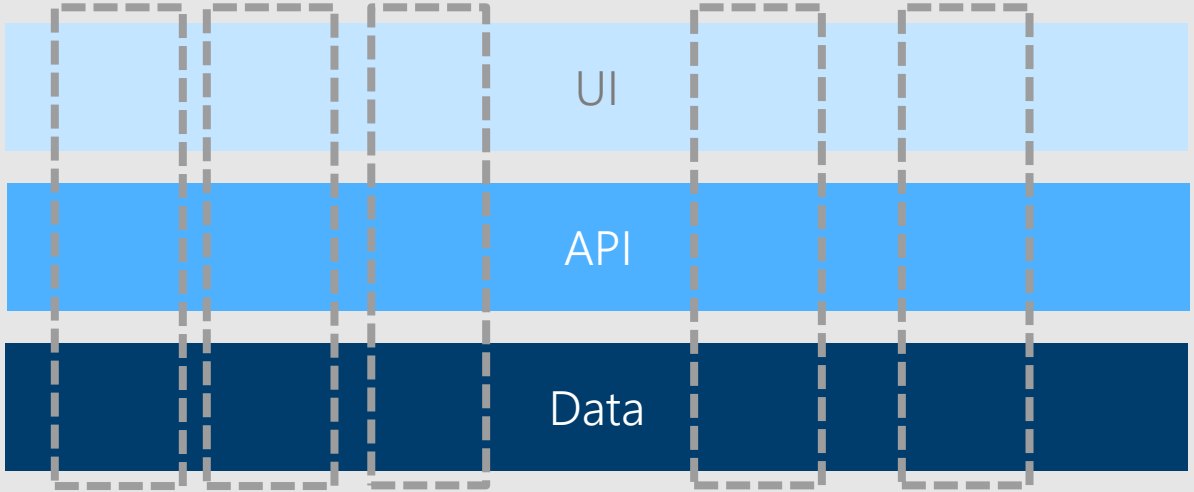


Deploy them **independently** to server/VM/container.

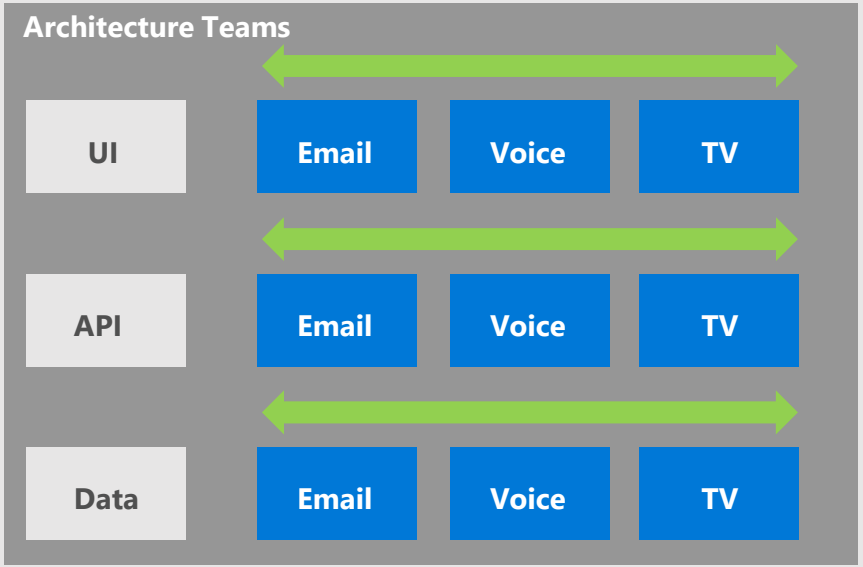
Instead of Horizontal…

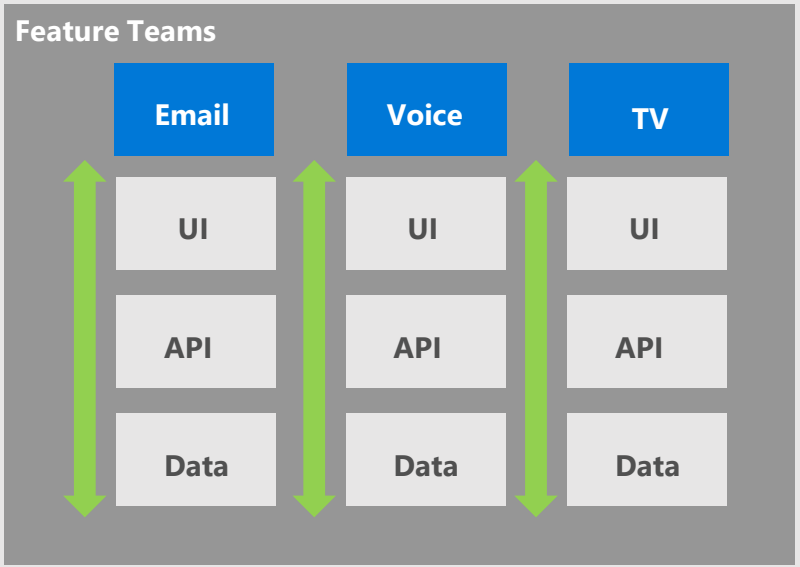


We went for Vertical

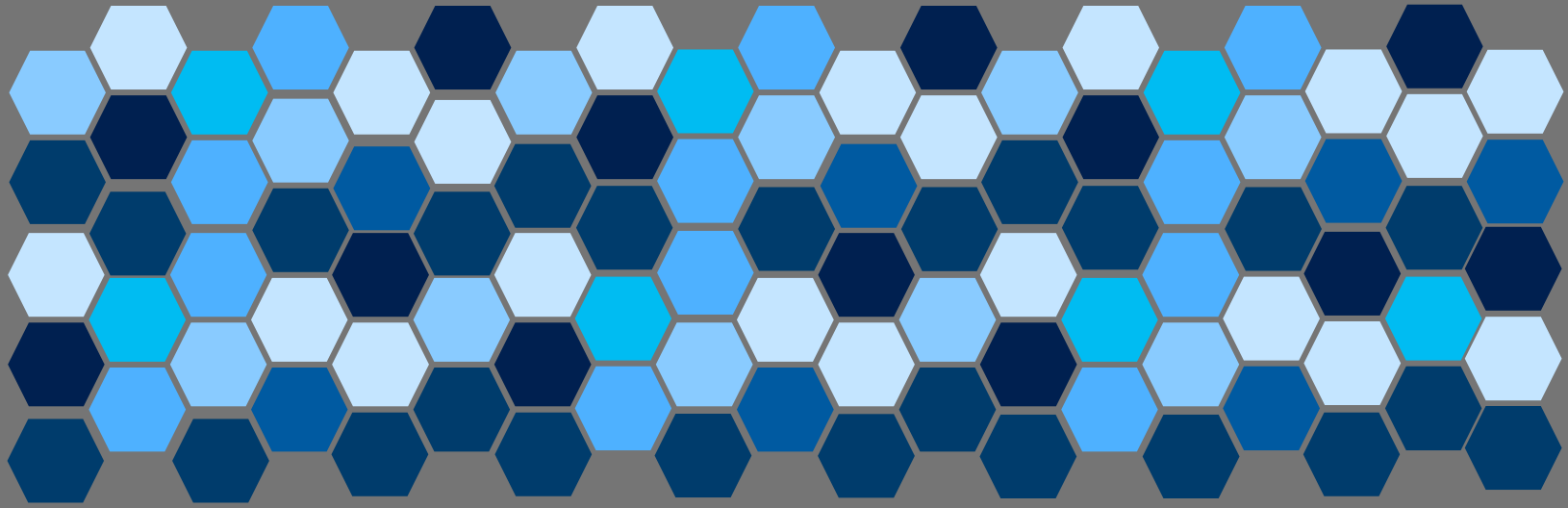


Horizontal team vs. Vertical team





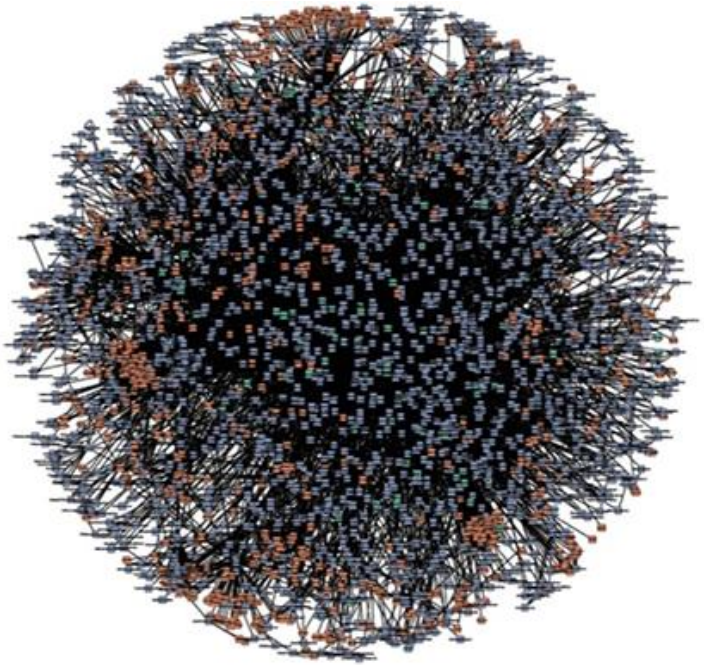
This leads to……



We may end up with huge bunch of Microservices

Is there any real example?

Microservices Real Examples



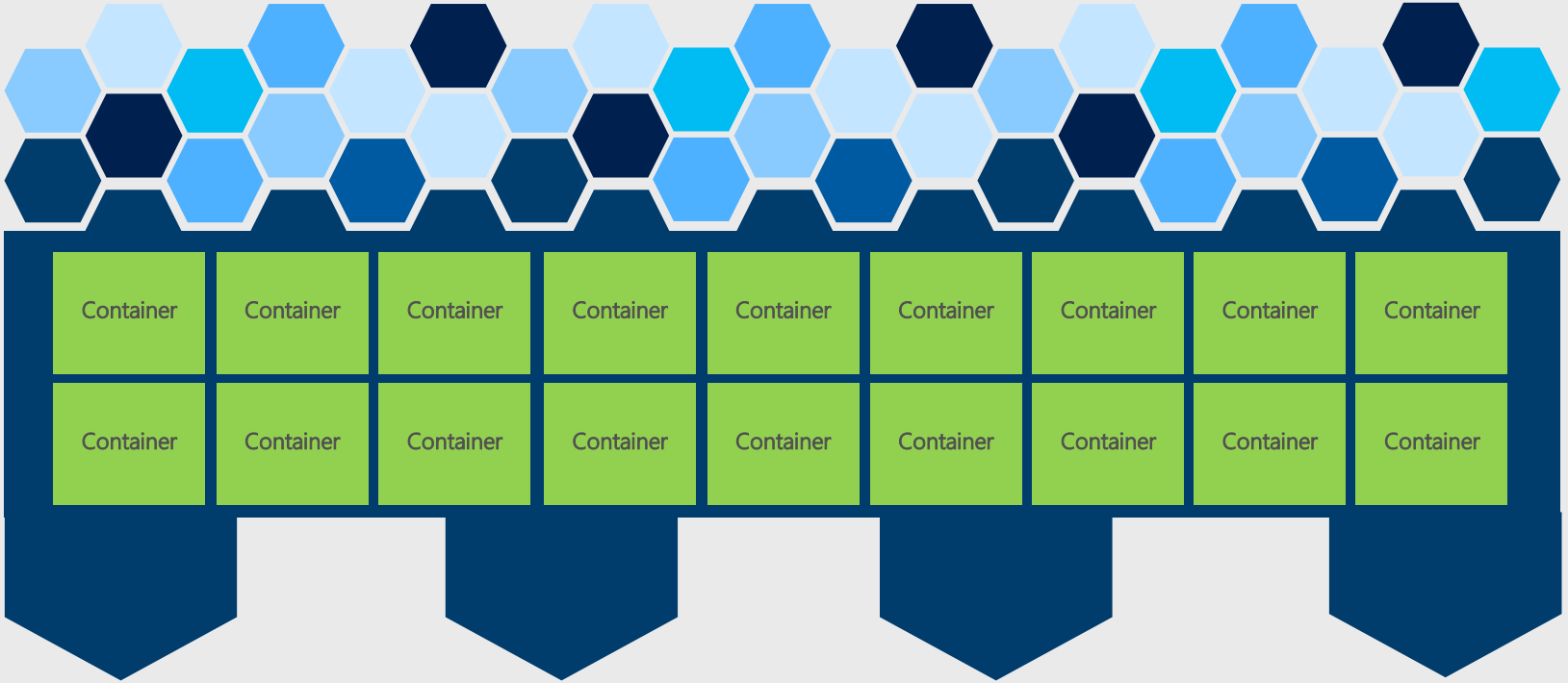
amazon.com

NETFLIX

Microservices

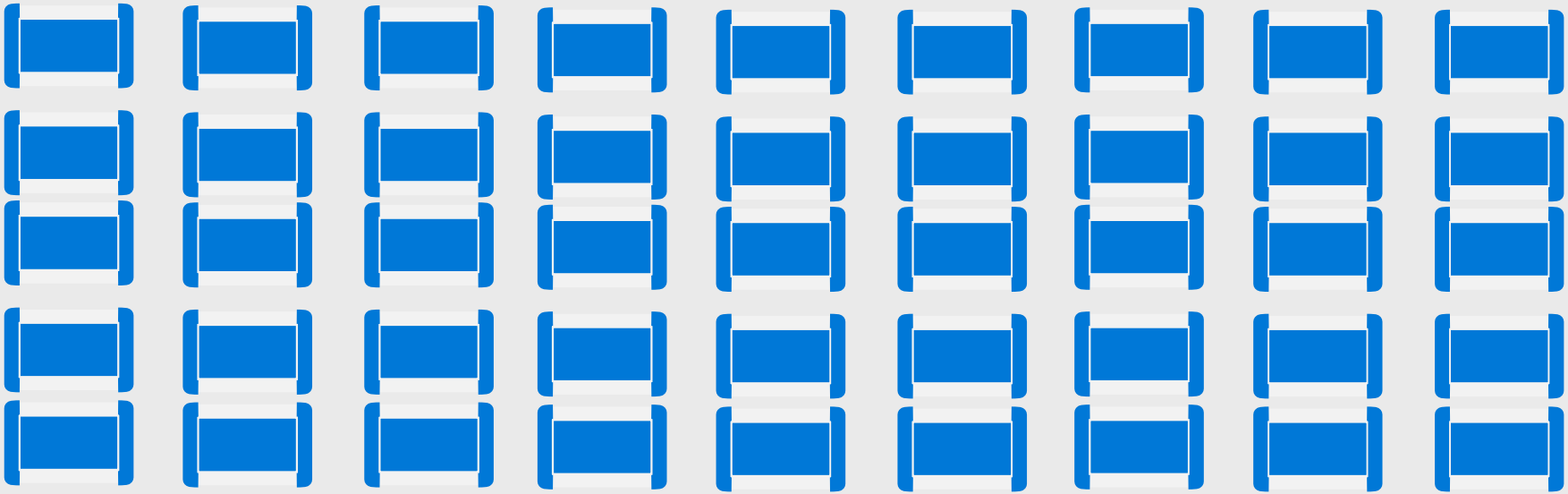
Hosting Microservices on VMs

What is the alternative host for Microservices?

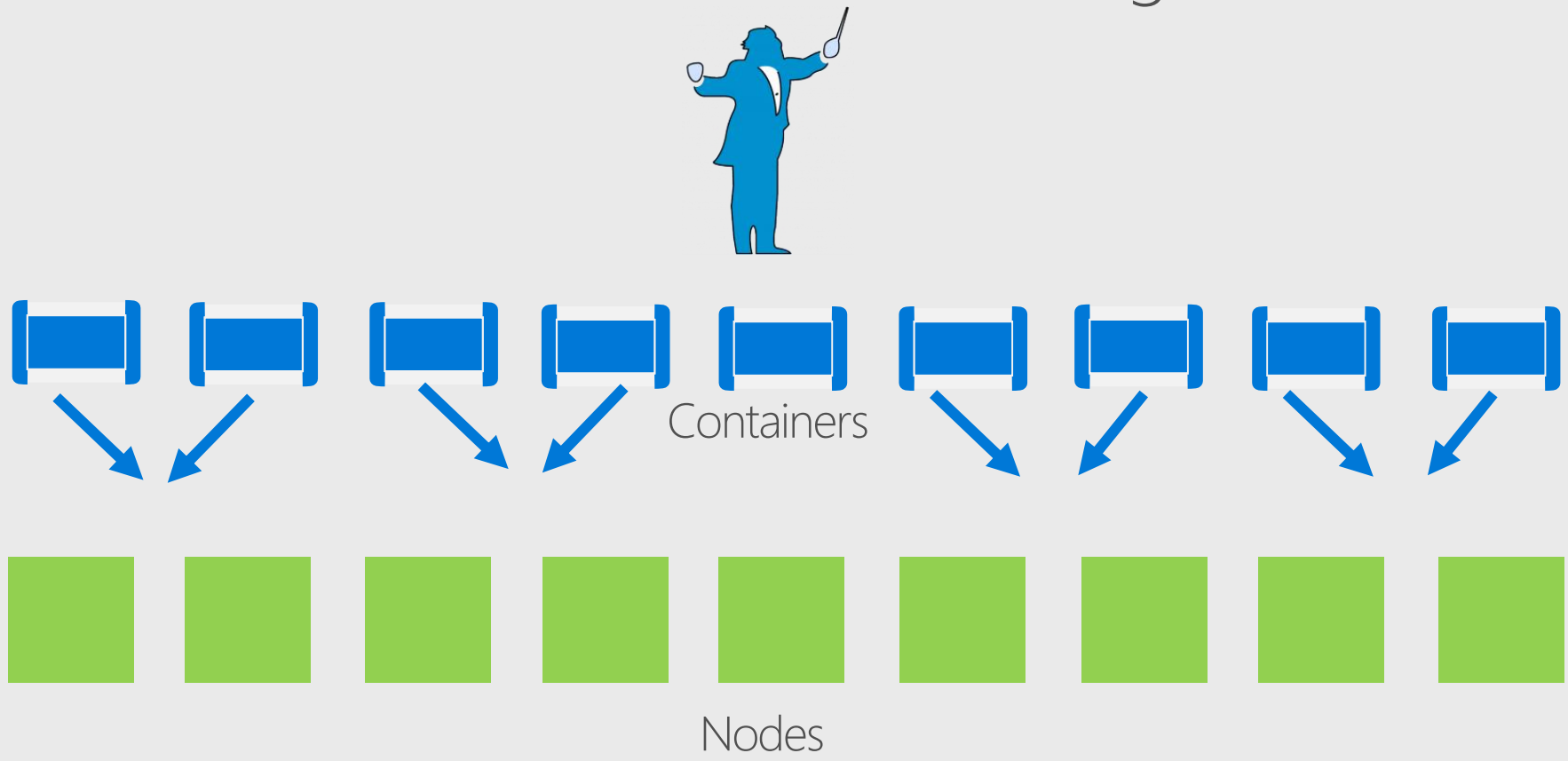


Containers and Orchestration

Container Orchestrator and Clustering?



Container Orchestrator and Clustering



Kubernetes

Deploy your applications quickly and predictably

Scale your applications on the fly

Roll out new features easily

Limit hardware usage to required resources only

Kubernetes

Portable

Public, private, hybrid, multi-cloud

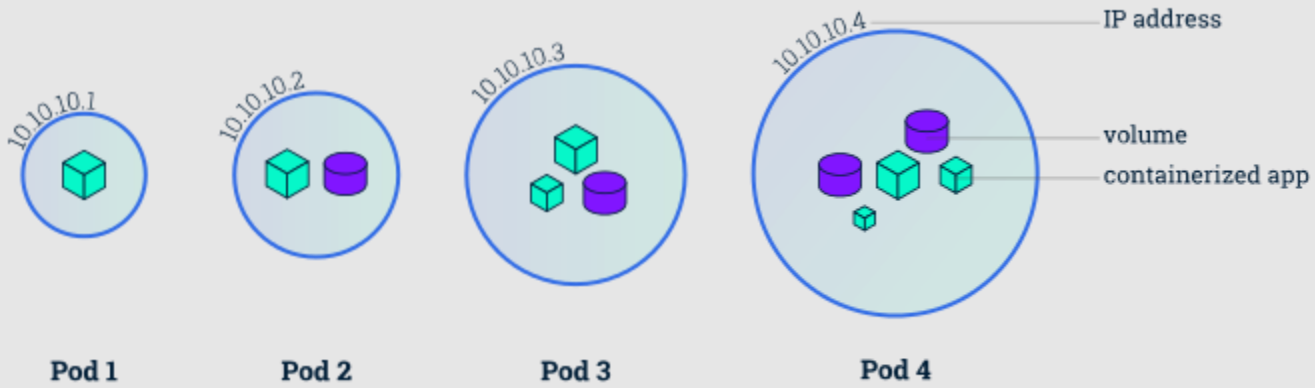
Extensible

Modular, pluggable, hookable, composable

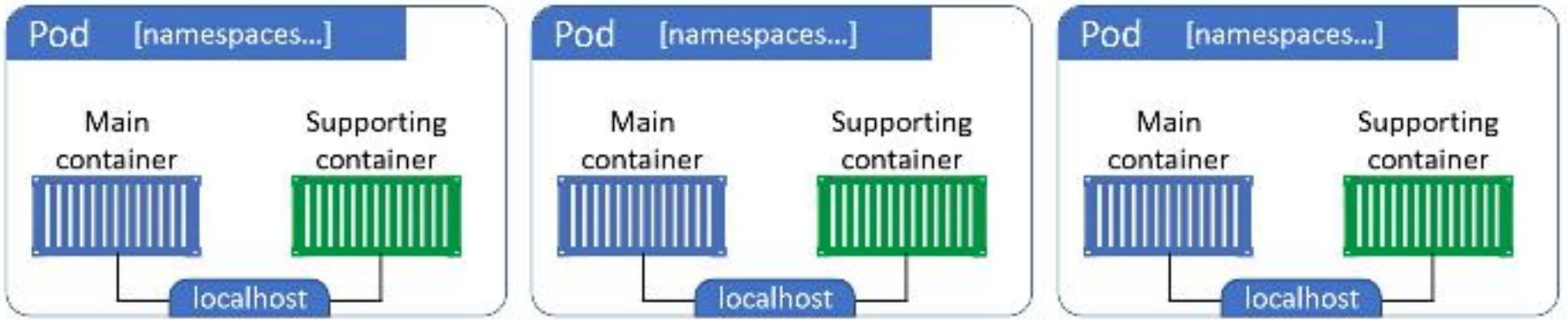
Self-healing

Auto-placement, auto-restart, auto-replication, auto-scaling

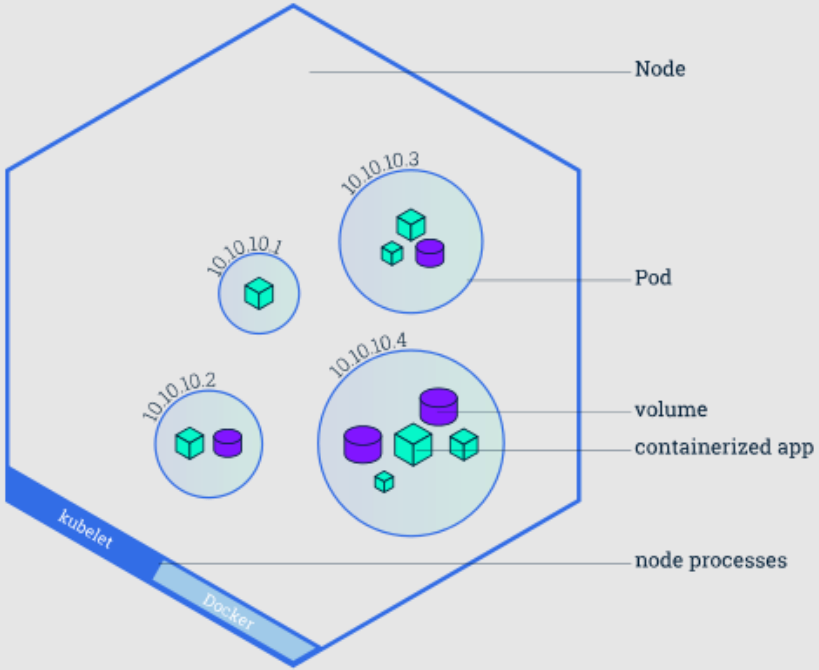
Pod


10.10.10.1
10.10.10.2
10.10.10.3
10.10.10.4
IP address
volume
containerized app
Pod 1
Pod 2
Pod 3
Pod 4

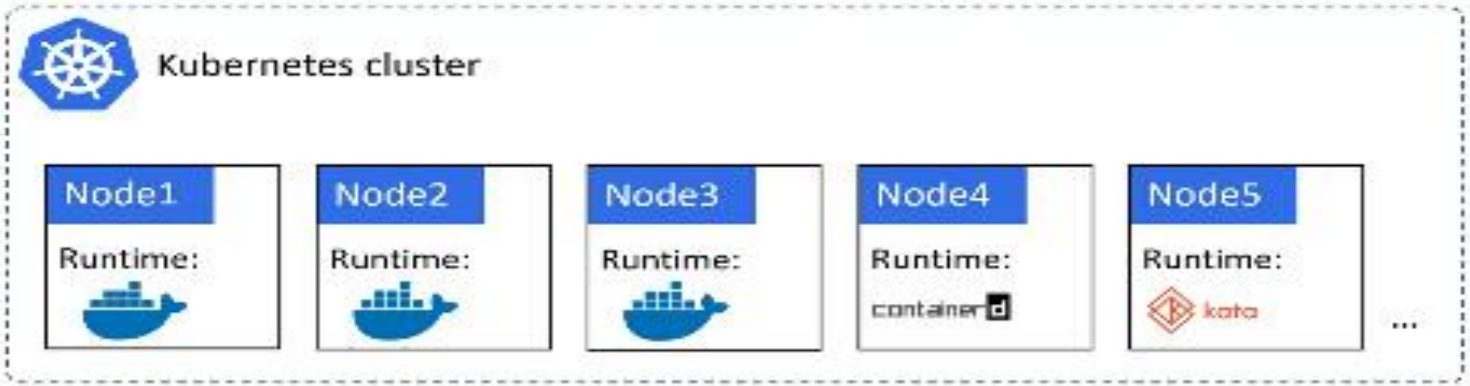
Containers, Pods and Scheduling



Node


Node
10.10.10.3
10.10.10.1
Pod
10.10.10.4
10.10.10.2
volume
containerized app
kubelet
Docker
node processes

Kubernetes Operates on a Cluster of Nodes



Azure Kubernetes Service (AKS)

Fully managed Kubernetes orchestration service

Auto patching, auto scaling, auto updates

Use the full Kubernetes ecosystem (100% upstream)

Deeply integrated with Azure Dev Tools and services

Azure Kubernetes Service (AKS)

Simplify the deployment, management, and operations of Kubernetes

Deploy and manage Kubernetes with ease

Scale and run applications with confidence

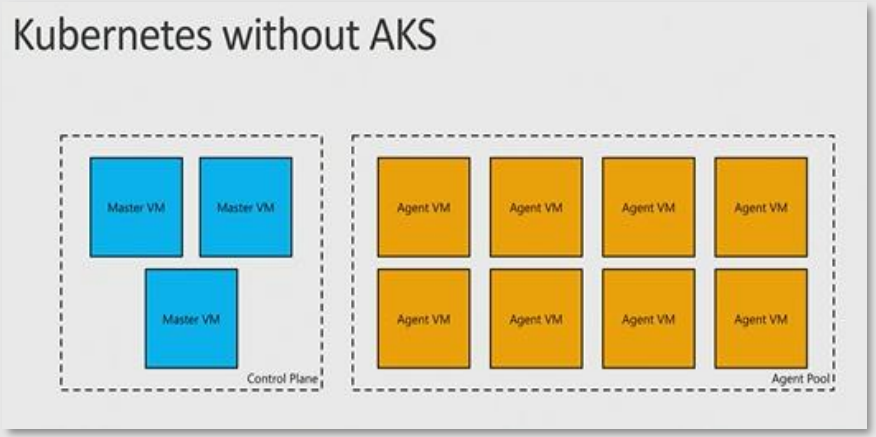
Secure your Kubernetes environment

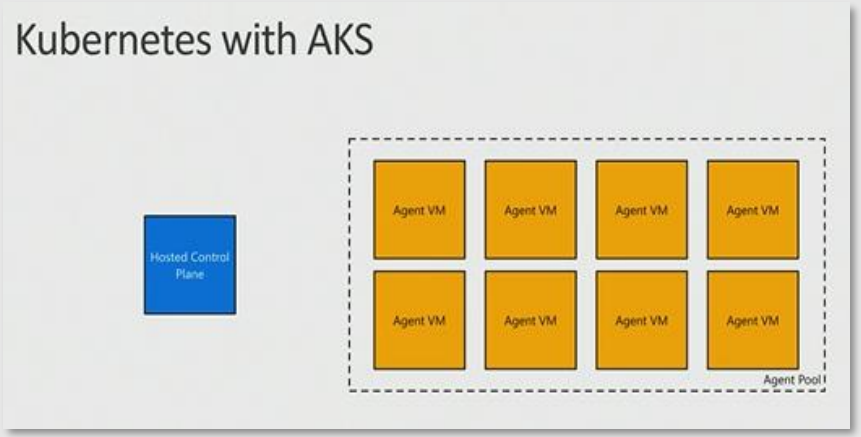
Accelerate containerized application development

Work how you want with open-source tools & APIs

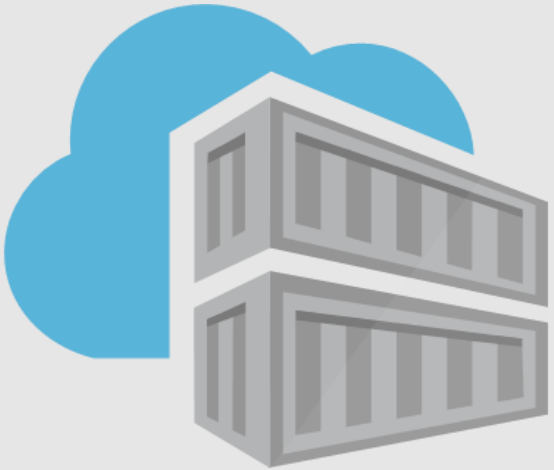
Set up CI/CD in a few clicks

Kubernetes With and Without AKS





Azure Container Registry



Azure Container Registry

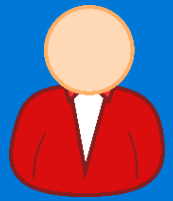
Azure DevOps and Azure Kubernetes Service

Azure Pipelines

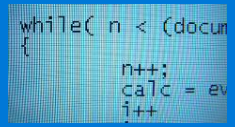
Azure Container Registry

Azure Kubernetes

End-to-End Delivery Activities



Dev



Code

Build

Unit Test

Package

Provision

Deploy

Deploy

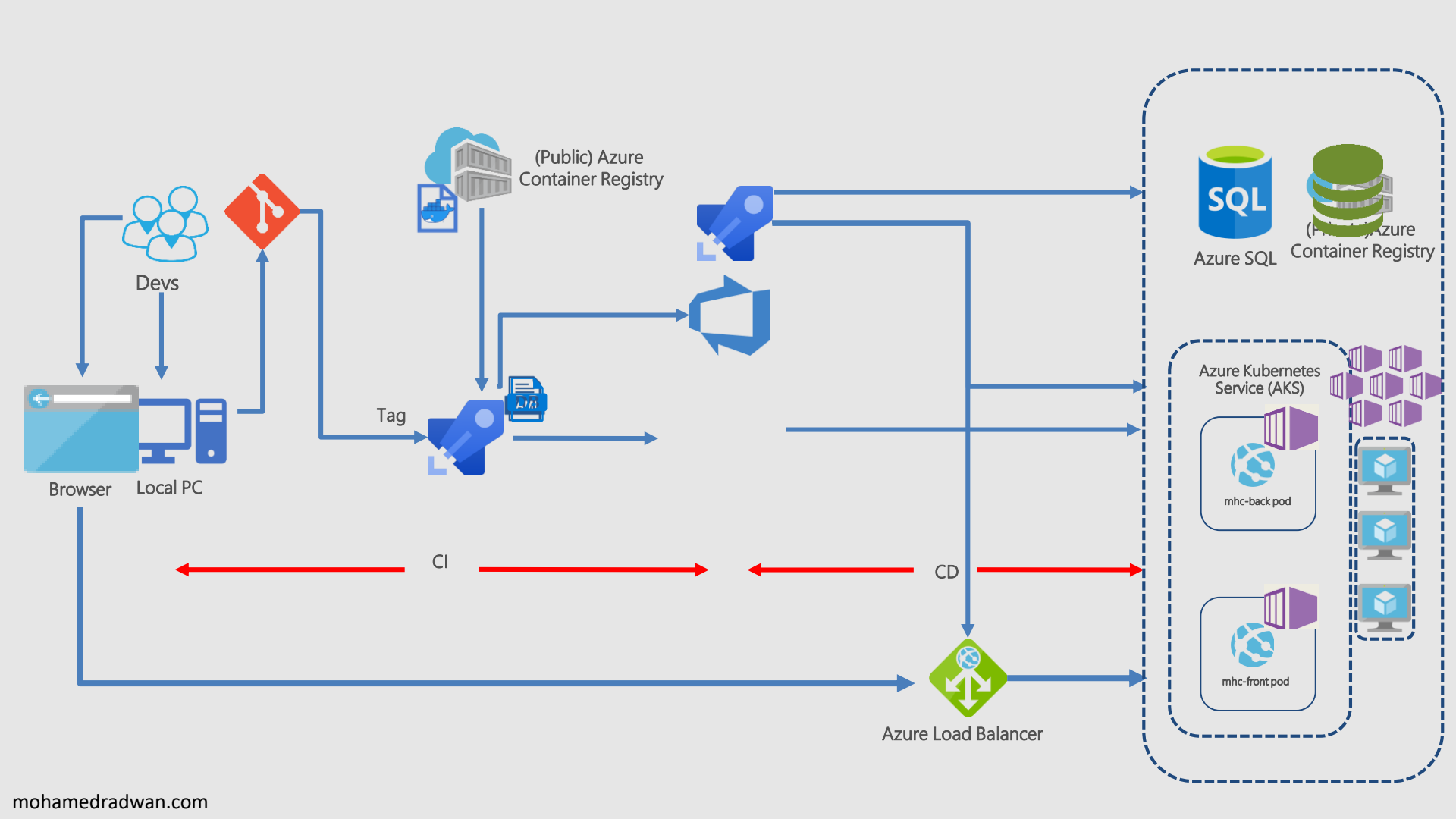
QA

Test

Ops

The End-to-End Workflow Demo

Azure resources	Description
 Azure Container Registry	Used to store the Docker images privately
 AKS	Docker images are deployed to Pods running inside AKS
 Azure SQL Server	SQL Server on Azure to host database


(Public) Azure
Container Registry
Devs
Tag
Browser
Local PC
CI
CD
Azure SQL
Azure
Container Registry
Azure Kubernetes
Service (AKS)
mhc-back pod
mhc-front pod
Azure Load Balancer

Demo

Commands used in the video:

```
version=$(az aks get-versions -l "West Europe" --query 'orchestrators[-1].orchestratorVersion' -o tsv)

az group create --name AKSRG --location "West Europe"

az aks create --resource-group AKSRG --name aksmohamedradwan --enable-addons monitoring --kubernetes-version $version --generate-ssh-keys --location "West Europe"

az acr create --resource-group AKSRG --name acrmohamedradwan --sku Standard --location "West Europe"

CLIENT_ID=$(az aks show --resource-group AKSRG --name aksmohamedradwan --query "servicePrincipalProfile.clientId" --output tsv)

ACR_ID=$(az acr show --name acrmohamedradwan --resource-group AKSRG --query "id" --output tsv)

az role assignment create --assignee $CLIENT_ID --role acrpull --scope $ACR_ID

az sql server create -l "West Europe" -g AKSRG -n sqlmohamedradwan -u sqladmin -p P2ssw0rd1234


az sql db create -g AKSRG -s sqlmohamedradwan -n mhcdb --service-objective S0


az aks get-credentials --resource-group AKSRG --name aksmohamedradwan

kubectl get pods

kubectl get service mhc-front --watch
 az aks install-cli

set PATH=%PATH%;C:\Users\mohamed.radwan\.azure-kubectl (for the current session) or copy it to Env vars

kubectl create clusterrolebinding kubernetes-dashboard -n kube-system --clusterrole=cluster-admin --serviceaccount=kube-system:kubernetes-dashboard

kubectl delete clusterrolebinding kubernetes-dashboard -n kube-system

az aks browse --resource-group AKSRG --name aksmohamedradwan


az aks scale --resource-group AKSRG --name aksmohamedradwan --node-count 3

az aks upgrade --resource-group AKSRG --name aksmohamedradwan --kubernetes-version 1.14.0
```

Your support really needed and appreciated!

Like

Share

Comment

Subscribe

Activate Notification

Survey

Upcoming Session Alert

Microsoft® Most Valuable Professional

DevOps Thoughts

PLAN CODE BUILD TEST PACKAGE DEPLOY CONFIGURE MONITOR

LIVE STREAMING

DevOps Open Source and Open Q&A

May 9, 2020 12:00 BST

Mohamed Radwan
Principal DevOps Consultant
Blog: mohamedradwan.com

Group (B)

LIVE
STREAMING

Free Live Webinar

CI/CD Pipelines as Code to Build Test and Deploy to Azure Cloud Web App & SQL

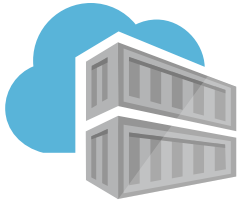
Saturday May 16, 2020
12:00 to 13:00 BST

Mohamed Radwan
Principal DevOps Consultant
Blog: mohamedradwan.com

LIVE
STREAMING

Live Webinar

Q&A



Mohamed Radwan

Principal DevOps Consultant

Blog: mohamedradwan.com

Group (A)

MVP Microsoft® Most Valuable Professional