Containers for IT Ops

Stijn Callebaut

Kurt Van Hoecke

Stijn Callebaut
Kurt Van Hoeke

@Stijnca

@Bunkco

Agenda

- Containers?
- Container orchestrators!
- Deploy
- Deploy an app
- Monitor all things!

Why do deployments hurt?

Mostly manual

Complex applications

Way to many versions

No documentation

‘throw over the fence politics’

- it worked on my computer!
- Not my problem!

Fight the pain

Automate the process

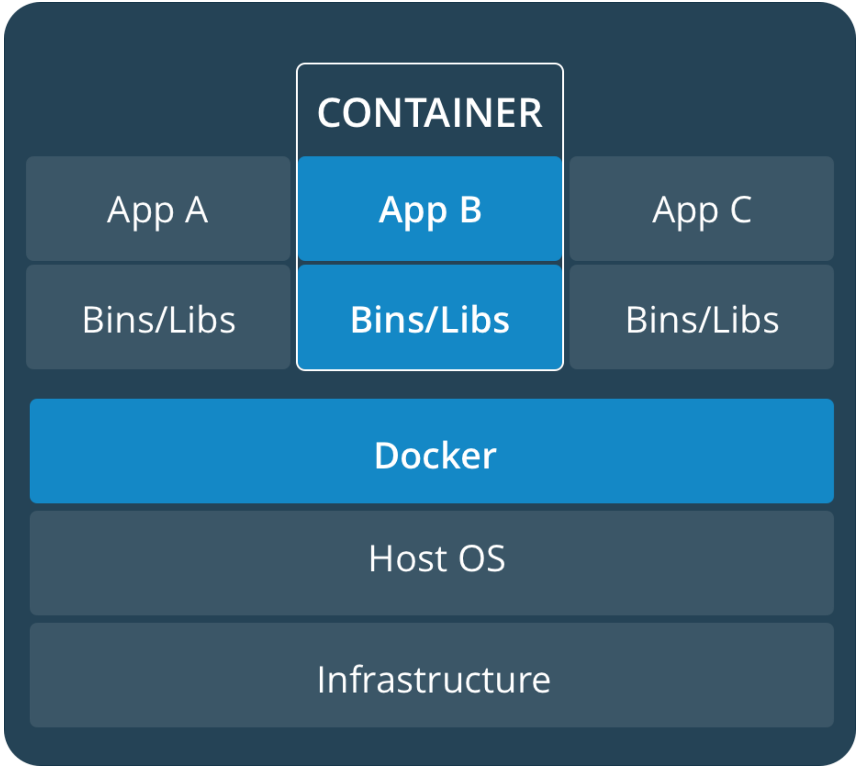
Divide the monolith into small services

Deploy small and fast

Encapsulate

What is a container

A container image is a **lightweight, stand-alone, executable package** of a piece of software that **includes everything needed to run it**: code, runtime, system tools, system libraries, settings. Available for both **Linux** and **Windows** based apps, containerized software will always run the same, regardless of the environment. Containers **isolate software from its surroundings**, for example differences between development and staging environments and help reduce conflicts between teams running different software on the same infrastructure.



CONTAINER

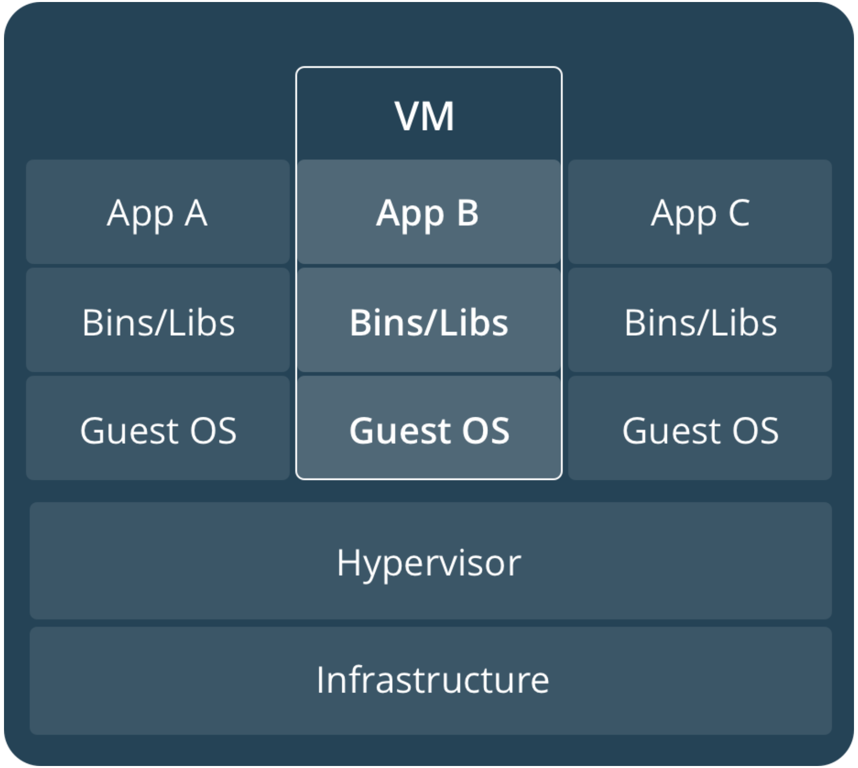
App A | App B | App C

Bins/Libs | Bins/Libs | Bins/Libs

Docker

Host OS

Infrastructure



VM

App A | App B | App C

Bins/Libs | Bins/Libs | Bins/Libs

Guest OS | Guest OS | Guest OS

Hypervisor

Infrastructure

Containers and images

Containers are the running micro applications – ~~like a vm~~ like an App-V package instance

Images include all the requirements for running the container. – ~~Like a sysprepped image \ template~~ an App-V package

This is all dev related, why do I care?

Knowing is half the battle!

This is all dev related, why do I care?

Developers:

take care of the contents of the container

IT Operations:

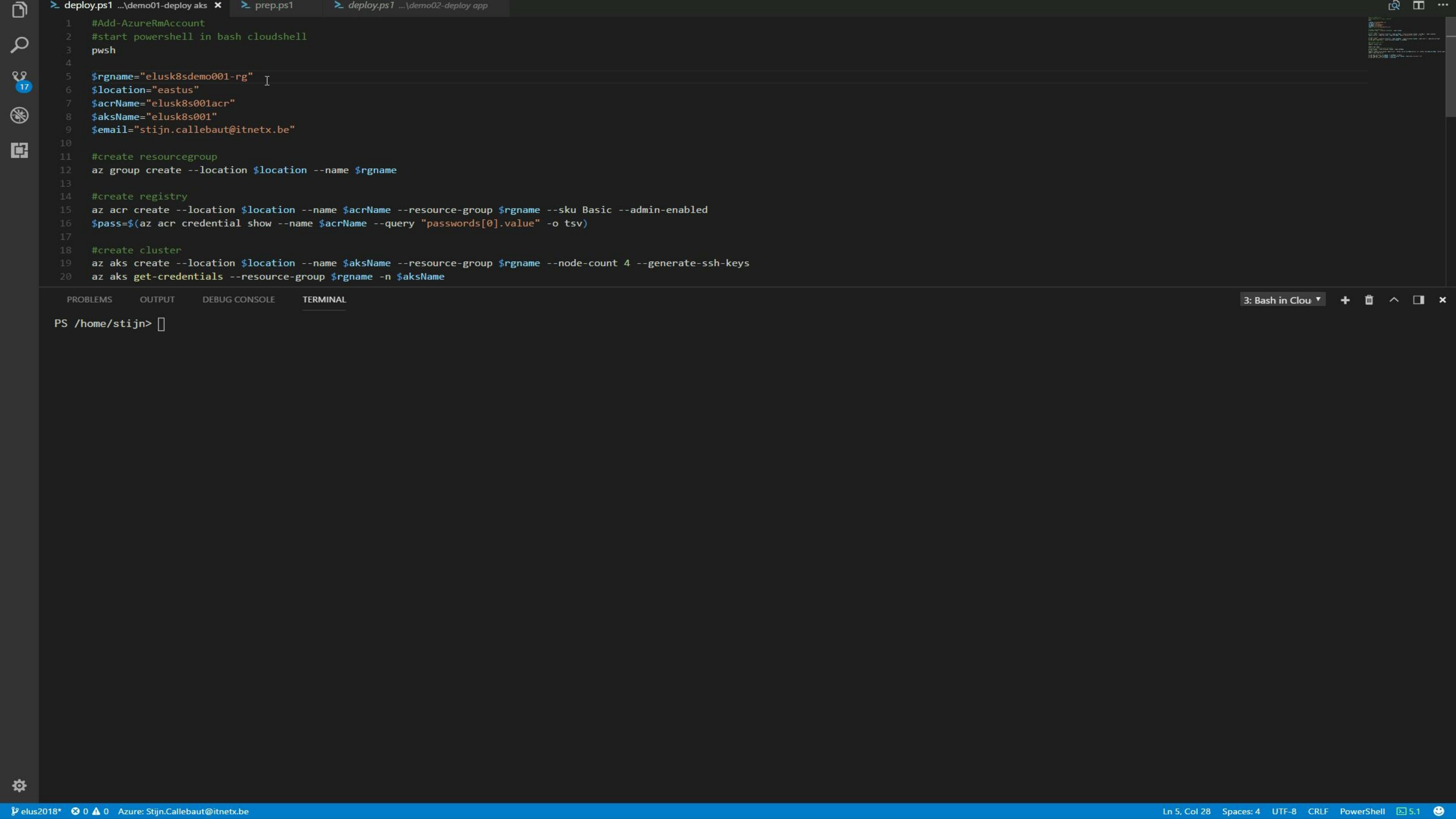
takes care of the operations of the container

Container ~~managers~~ orchestrators

system for automating deployment, scaling, and management of containerized applications.

DEMO

Deploy k8s

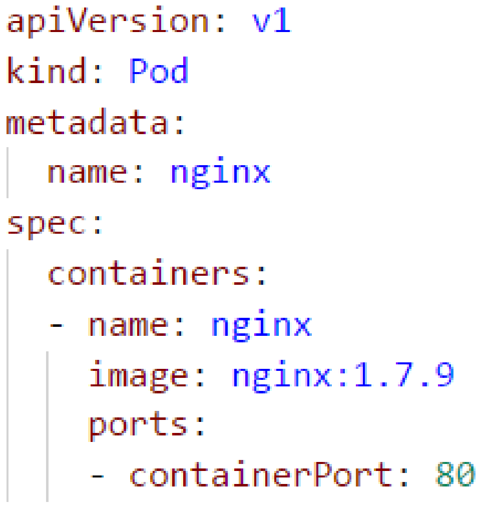


```
#Add-AzureRmAccount
#start powershell in bash cloudshell
pwsh

$rgname="elusk8sdemo001-rg"
$location="eastus"
$acrName="elusk8s001acr"
$aksName="elusk8s001"
$email="stijn.callebaut@itnetx.be"

#create resourcegroup
az group create --location $location --name $rgname

#create registry
az acr create --location $location --name $acrName --resource-group $rgname --sku Basic --admin-enabled
$pass=$(az acr credential show --name $acrName --query "passwords[0].value" -o tsv)

#create cluster
az aks create --location $location --name $aksName --resource-group $rgname --node-count 4 --generate-ssh-keys
az aks get-credentials --resource-group $rgname -n $aksName
```

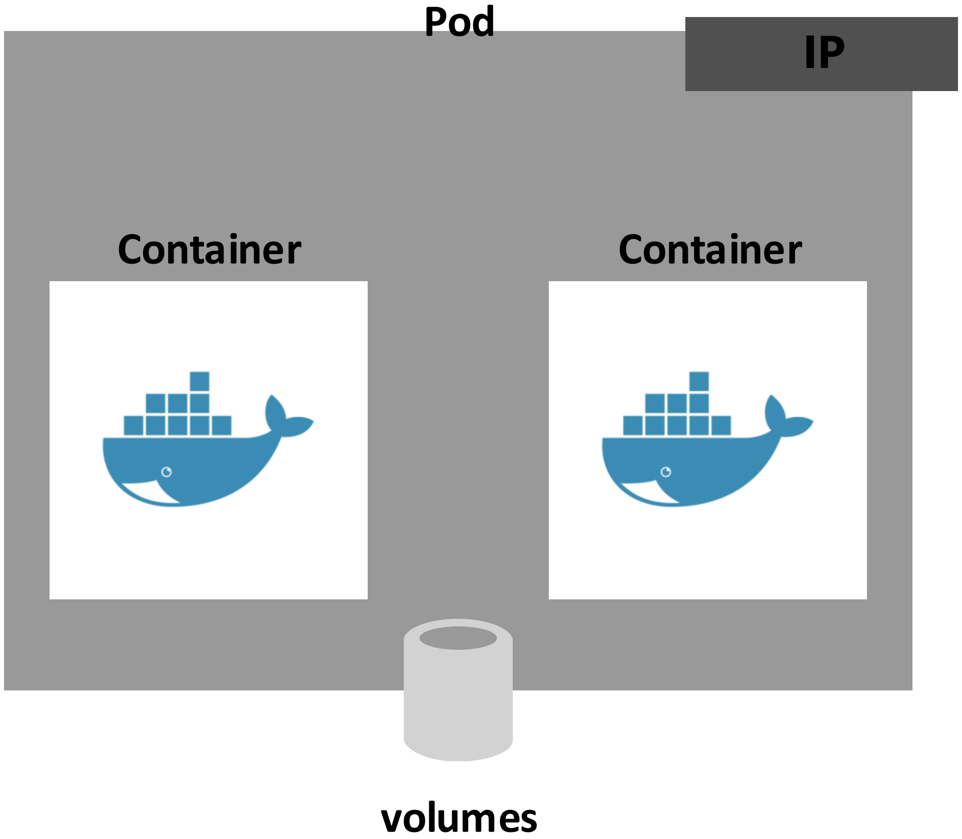
```
PS /home/stijn>
```

Introducing kubernetes (k8s)

Pods represent the smallest deployable artifact in k8s. It is a collection of containers running in the same execution environment

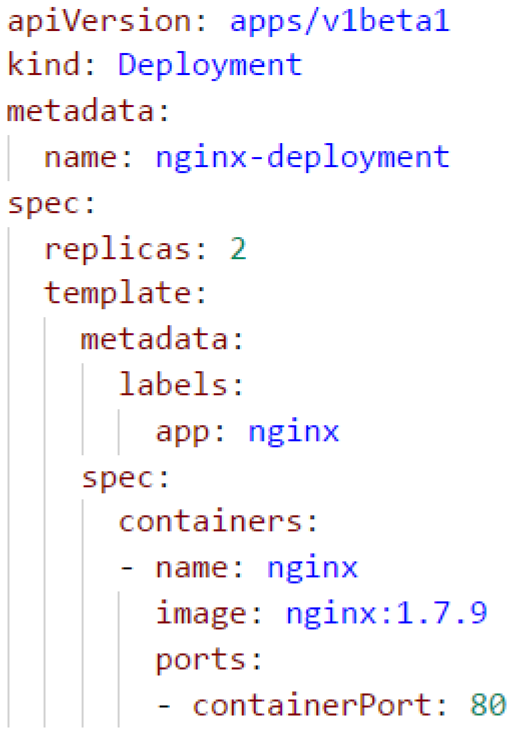


```yaml
apiVersion: v1
kind: Pod
metadata:
  name: nginx
spec:
  containers:
  - name: nginx
    image: nginx:1.7.9
    ports:
    - containerPort: 80
```

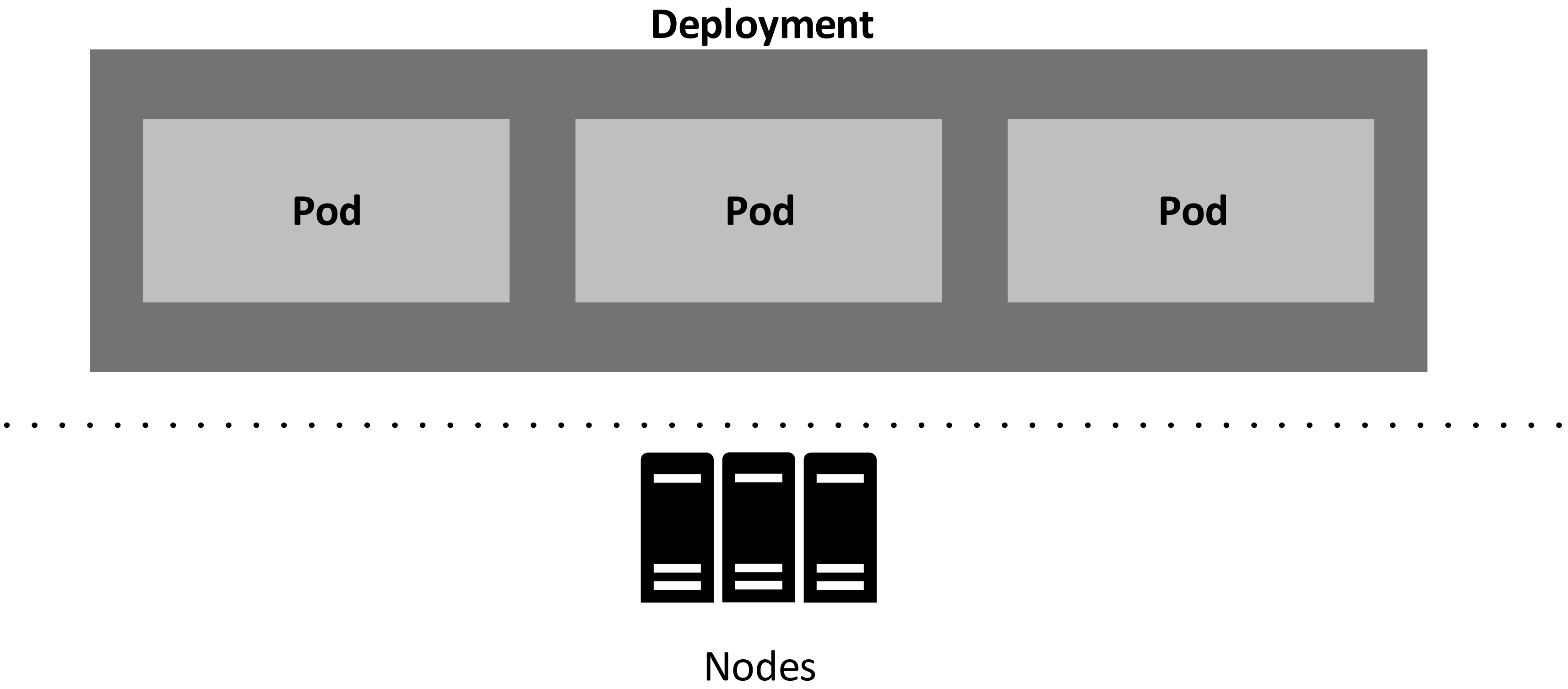

Pod
IP
Container
Container
volumes

a **replicaset** defines a single scalable, self-healing state of a pod.

Deployments manage replicasets and amongst others. It describes the desired state



```yaml
apiVersion: apps/v1beta1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  replicas: 2
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:1.7.9
        ports:
        - containerPort: 80
```



Deployment

Pod

Pod

Pod

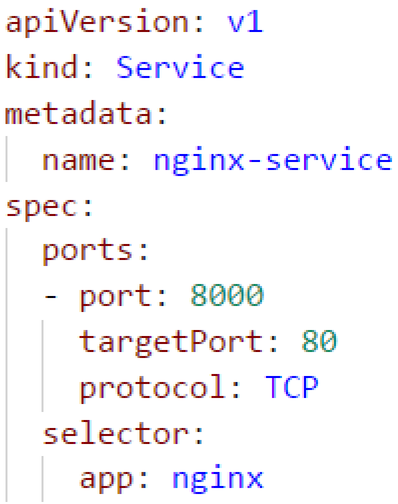
Nodes

Services is an abstraction which defines a logical set of pods\deployments and a policy by which to access them

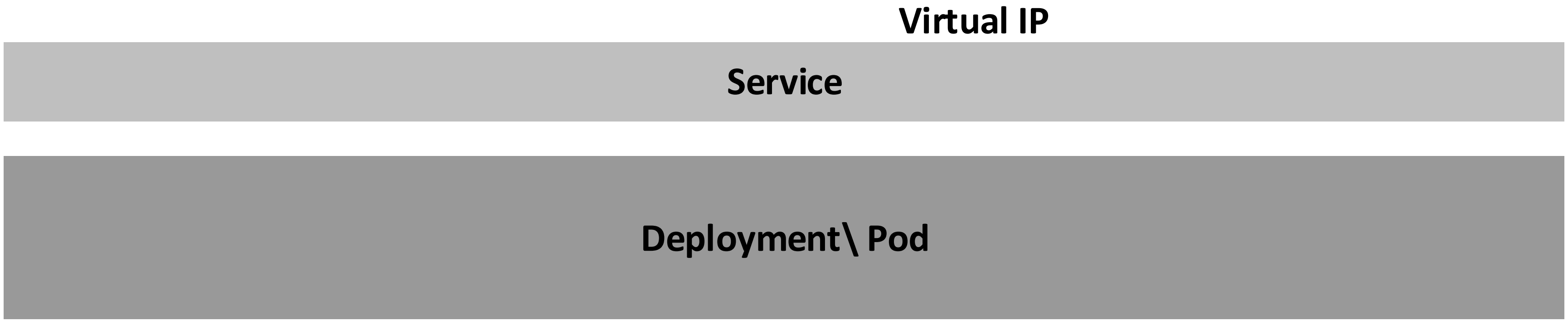
Ingress: An API object that manages external access to the services in a cluster, typically HTTP.

Namespaces: Namespaces are a way to divide cluster resources

Volumes: a way to share files between containers and to preserve files between restarts



```yaml
apiVersion: v1
kind: Service
metadata:
  name: nginx-service
spec:
  ports:
  - port: 8000
    targetPort: 80
    protocol: TCP
  selector:
    app: nginx
```



Virtual IP

Service

Deployment\ Pod

DEMO

Deploy our application

Monitoring

Prometheus

Datadog

Elasticsearch + kibana

Grafana (influxdb – heapster)

‘fluentd’ and OMS

DEMO

K8s and OMS

Recap

Containers are here to stay

Declarative syntax end-to-end

Faster deployments

Stateless and statefull

PaaS integrations

Monitoring

Containers on Azure

Azure web apps

Azure Batch

Azure Container Service \ ACS-Engine

Azure Kontainer Service (AKS)

Azure Container Instances

Azure Service Fabric

Useful information

Kubernetes up and running

Kubernetes the hard way

Kubernetes.io

The illustrated guide to kubernetes (video)

Containers on Azure

Useful information

Kubernetes Azure interest group

Docker file reference

OMS container solution

Session demo's and examples

Questions?