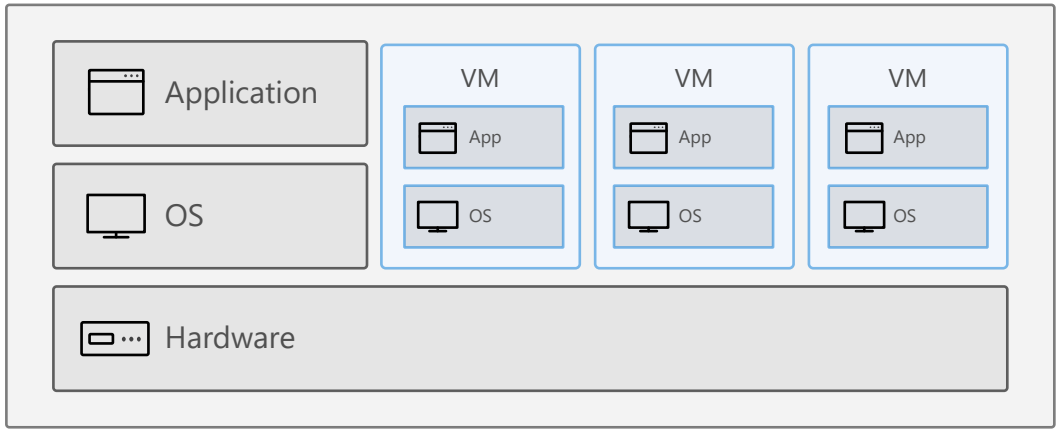
Microsoft

Azure Kubernetes Service

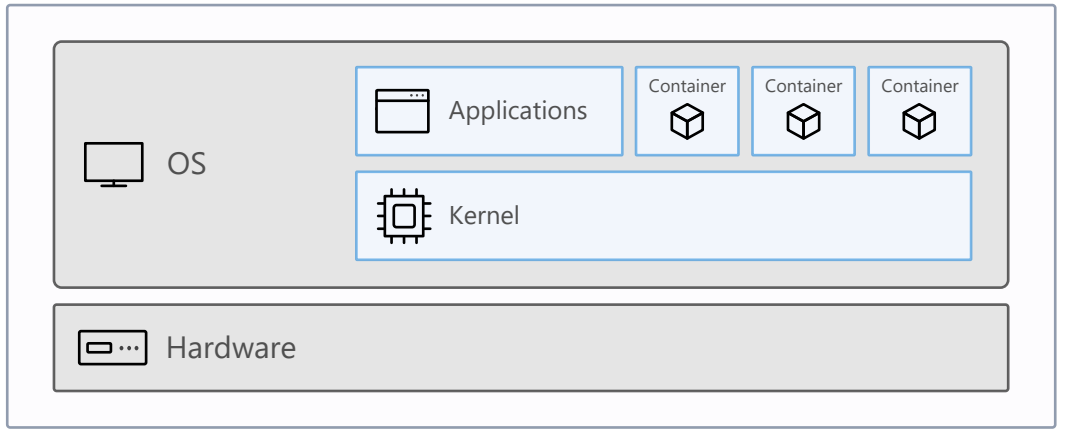
Jay Kumar
Sr. Cloud Solution Architect, Customer Success

What is a container?

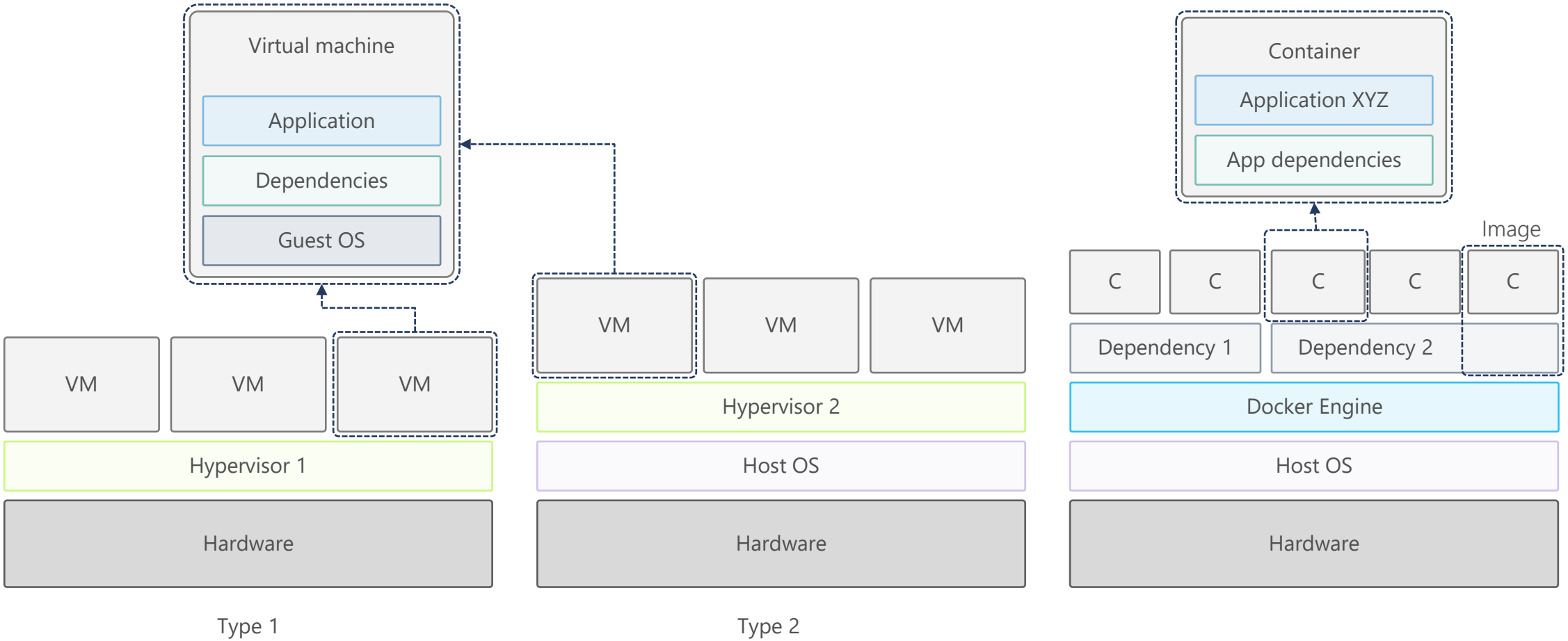
Traditional virtual machines = hardware virtualization



Containers = operating system virtualization



Virtualization versus containerization?



How Container are Launched ?

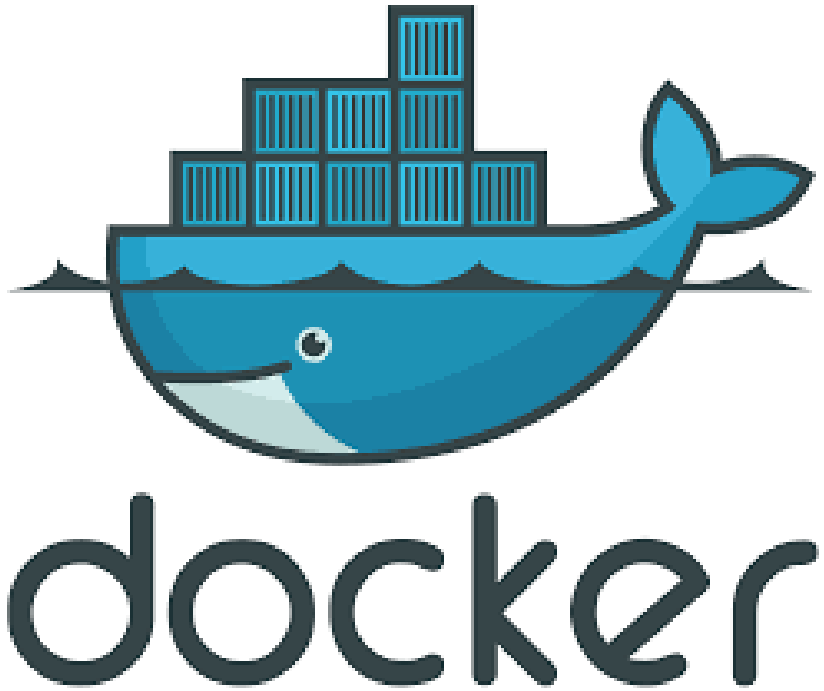
- A container is launched by running an image. An **image** is an executable package that includes everything needed to run an application--the code, a runtime, libraries, environment variables, and configuration files.
- A **container** is a runtime instance of an image--what the image becomes in memory when executed (that is, an image with state, or a user process).

Docker

Docker is a Open-Source computer program that performs operating-system-level virtualization, also known as "containerization". It was first released in 2013 and is developed by Docker, Inc.

Docker is used to run software packages called "containers". In a typical example use case, one container runs a web server and web application, while a second container runs a database server that is used by the web application.

- Launched in March 2013
- Over 37 billion+ Downloads
- Over 3,5million+ Docker-ized applications
- 100+ Case Studies worldwide



https://en.wikipedia.org/wiki/Docker_(software)

The benefits of using containers

Agility

Ship apps faster

Portability

Easily move workloads

Density

Achieve resource efficiency

Rapid scale

Scale easily to meet demand

Any OS

Linux

Windows

Anywhere

On-premises

Cloud

Any language

Java

.Net

Python

Node

Any app

Monolith

Microservice

Containers in Azure

App Service

Deploy web apps or APIs using containers in a PaaS environment

Service Fabric

Modernize .NET applications to microservices using Windows Server containers

Kubernetes Service

Scale and orchestrate Linux containers using Kubernetes

Container Instance

Elastically burst from your Azure Kubernetes Service (AKS) cluster

Ecosystem

Bring your Partner solutions that run great on Azure

Azure Container Registry

Docker Hub

Choice of developer tools and clients

Microsoft

Container Challenges


VM Host
Container
Application
App code
Dependencies
Environmental Variables
Files
Libraries
Drivers
Things can get complicated very quickly

Container Management at Scale

Cluster Management: deploy and manage cluster resources

Scheduling: where containers run

Lifecycle and Health: keep containers running despite failure

Naming and Discovery: where are my containers

Load Balancing: evenly distribute traffic

At the end of the day we need something to help us with all the orchestration..
An orchestrator!

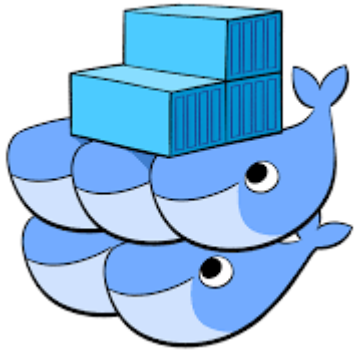
Scaling make se… contain… elastic i… number

…container images

…containers and cluster

…data …ers

Available Orchestrators



- Docker Swarm
- Apache Mesos
- Nomad (from HashiCorp)
- Rancher
- Service Fabric
- …

MESOS

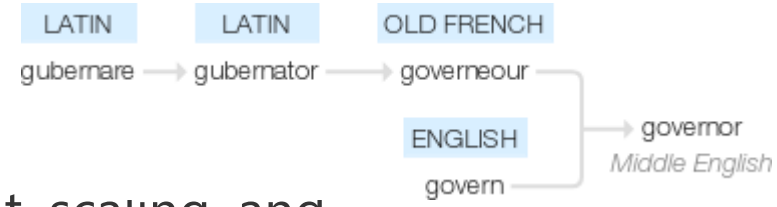
HashiCorp
Nomad

RANCHER

Kubernetes

Microsoft Azure
Service Fabric

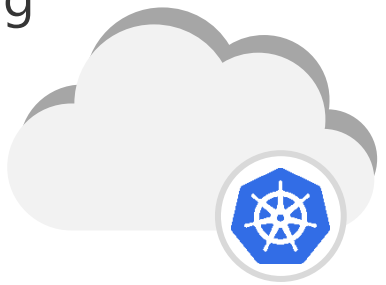
What is Kubernetes (k8s)?



- **Kubernetes** is "an open-source software for automating deployment, scaling, and management of containerized applications".
- **Kubernetes**, in Greek κυβερνήτης, means the Helmsman, or pilot of the ship.
- Keeping with the maritime theme of **Docker** containers, **Kubernetes** is the pilot of a ship of containers.

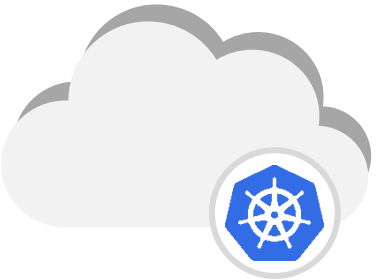
History

- Originally designed by Google and is now maintained by the Cloud Native Computing Foundation (CNCF).
- Google still actively involved
- Kubernetes v1.0 was released on July, 2015 by Joe Beda, Brendan Burns and Craig McLuckie
- Most discussed repo in GitHub last year.
- Over 1,700 authors and releases every three month
- To learn more about the ideas behind Kubernetes: read the Large-scale cluster management at Google with Borg paper

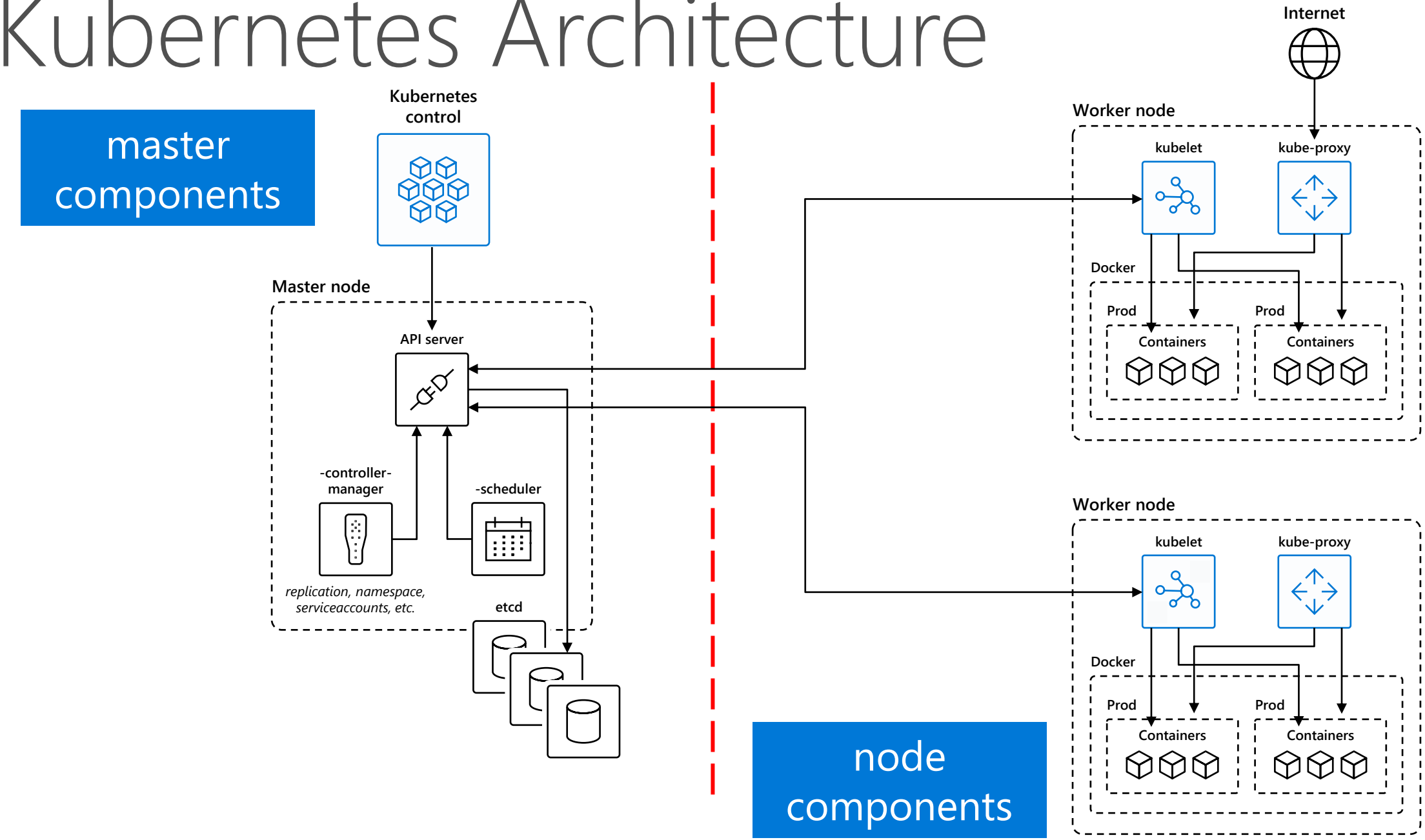


Kubernetes Features

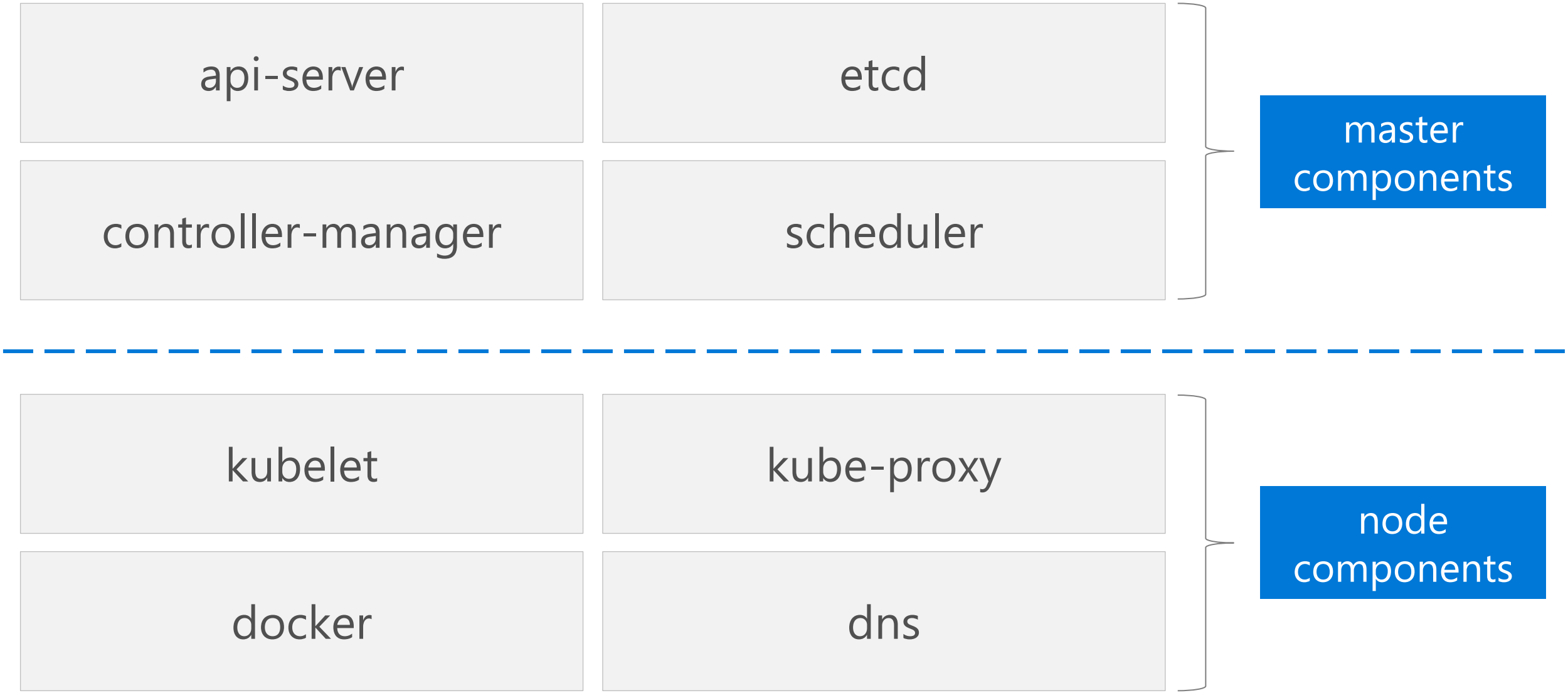
- Self-Healing
- Horizontal Scaling
- Automated rollouts and rollbacks
- Service Discovery and Load Balancing
- Automatic bin packing
- Storage orchestration
- Secret and configuration management



Kubernetes Architecture



Recap – K8s Components



Kubernetes – The Hard Way

- Installing the Client Tools
- Provisioning Compute Resources
- Provisioning the CA and Generating TLS Certificates
- Generating Kubernetes Configuration Files for Authentication
- Generating the Data Encryption Config and Key
- Bootstrapping the etcd Cluster
- Bootstrapping the Kubernetes Control Plane
- Bootstrapping the Kubernetes Worker Nodes
- Configuring kubectl for Remote Access
- Provisioning Pod Network Routes
- Deploying the DNS Cluster Add-on
- Smoke Test

Reference: https://github.com/kelseyhightower/kubernetes-the-hard-way

Microsoft

Kubernetes in Azure (AKS)

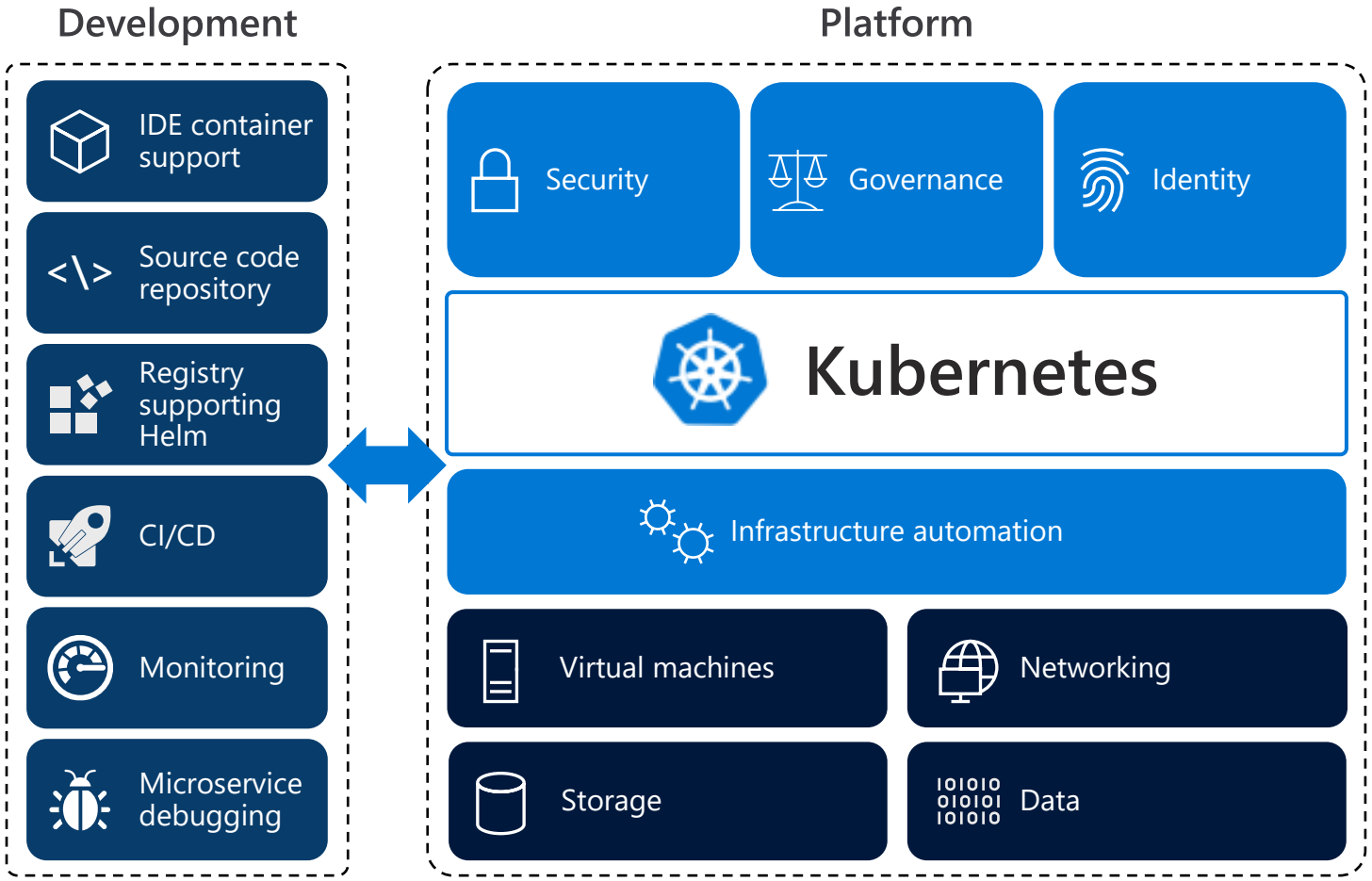
Managed k8s service

Kubernetes on its own is not enough

Save time from infrastructure management and roll out updates faster without compromising security

Unlock the agility for containerized applications using:

- **Infrastructure automation** that simplifies provisioning, patching, and upgrading
- Tools for **containerized app development and CI/CD workflows**
- Services that support **security, governance, and identity and access management**



Azure Kubernetes Service

Simplify the deployment, management, and operations of Kubernetes

Ship-faster

Focus on your containers not the infrastructure

Extensible

Modular, pluggable, hookable, composable

Self-healing

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

Deploy, manage, and monitor Kubernetes with ease

Accelerate containerized app development

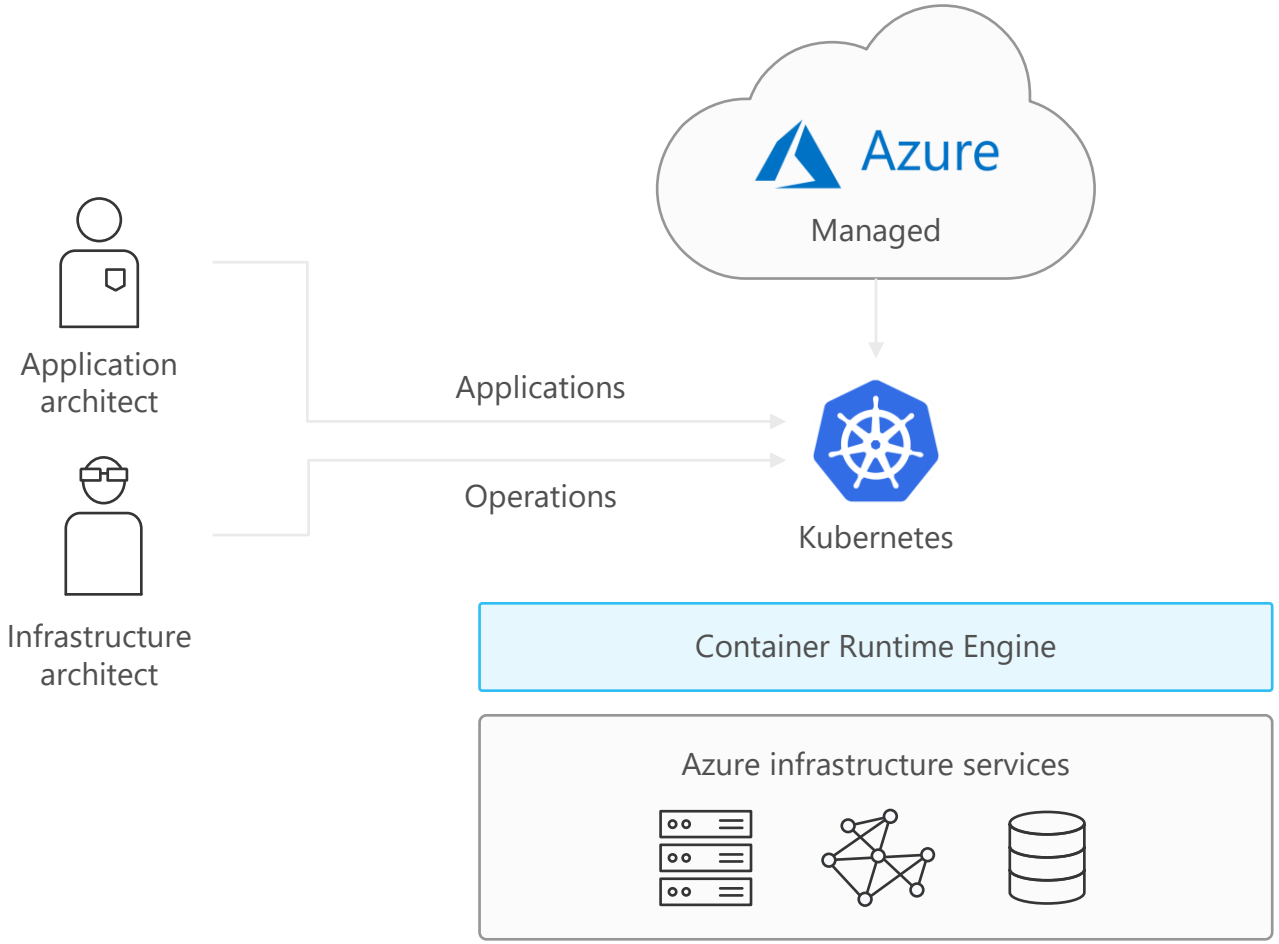
Roll out new features seamlessly (CI/CD)

Secure your environment with layers of isolation

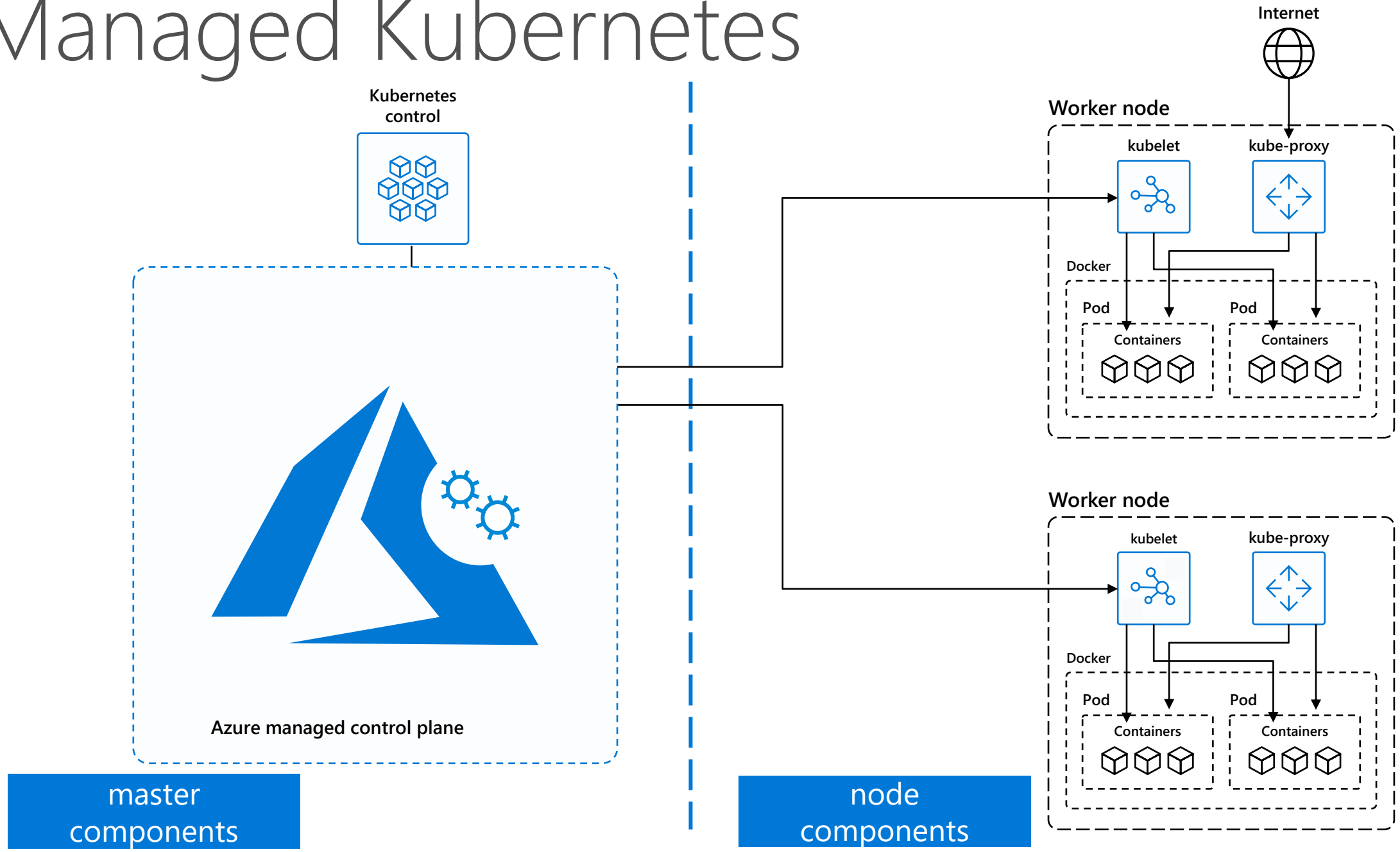
Scale applications on the fly

Azure Kubernetes Service (AKS)

A fully managed Kubernetes cluster

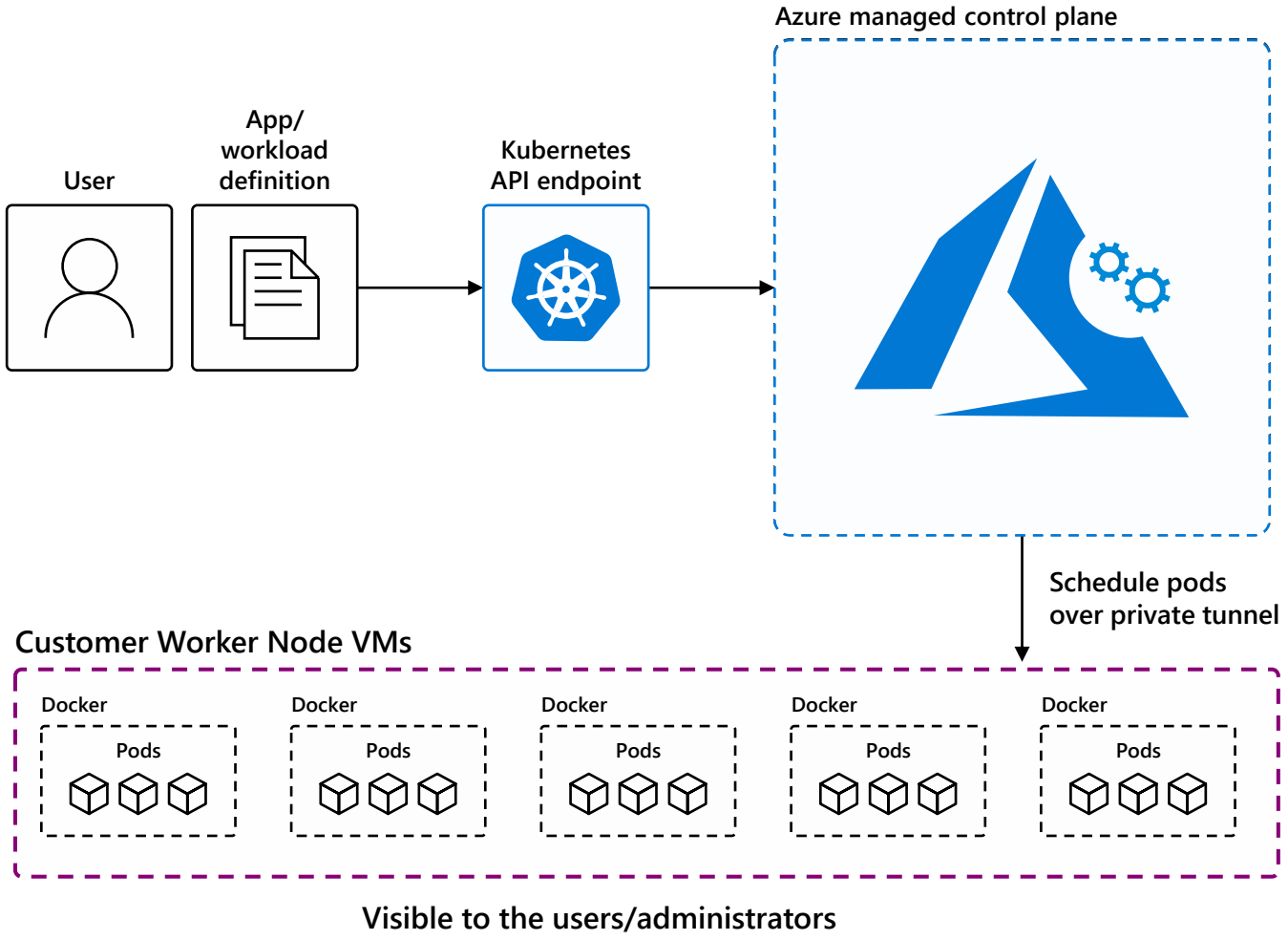


Managed Kubernetes



How managed Kubernetes on Azure works

- Automated upgrades, patches
- High reliability, availability
- Easy, secure cluster scaling
- Self-healing
- API server monitoring
- **At no charge**
- Auto Scaling of Nodes



Azure makes Kubernetes easy

Deploy and manage Kubernetes with ease

Task	The old way	With Azure
Create a cluster	Provision network and VMs Install dozens of system components including etcd Create and install certificates Register agent nodes with control plane	az aks create
Upgrade a cluster	Upgrade your master nodes Cordon/drain and upgrade worker nodes individually	az aks upgrade
Scale a cluster	Provision new VMs Install system components Register nodes with API server	az aks scale

Azure makes Kubernetes easy

Get started easily

```
> az aks create -g myResourceGroup -n myCluster --generate-ssh-keys
\ Running ..

> az aks install-cli
Downloading client to /usr/local/bin/kubectl ..

> az aks get-credentials -g myResourceGroup -n myCluster
Merged "myCluster" as current context ..

> kubectl get nodes
NAME                          STATUS    AGE       VERSION
aks-mycluster-36851231-0      Ready     4m        v1.8.1
aks-mycluster-36851231-1      Ready     4m        v1.8.1
aks-mycluster-36851231-2      Ready     4m        v1.8.1
```

Azure makes Kubernetes easy

Manage an AKS cluster

```
> az aks list –o table
Name                Location    ResourceGroup     KubernetesRelease  ProvisioningState
------------------  ----------  --------------  -------------------  ------------------
myCluster           westus2     myResourceGroup                1.7.7  Succeeded
```

```
➢ az aks upgrade -g myResourceGroup -n myCluster –-kubernetes-version
  1.8.1

   \ Running ..
```

```
➢ az aks scale -g myResourceGroup -n myCluster --agent-count 10

   \ Running ..
```

Azure makes Kubernetes easy

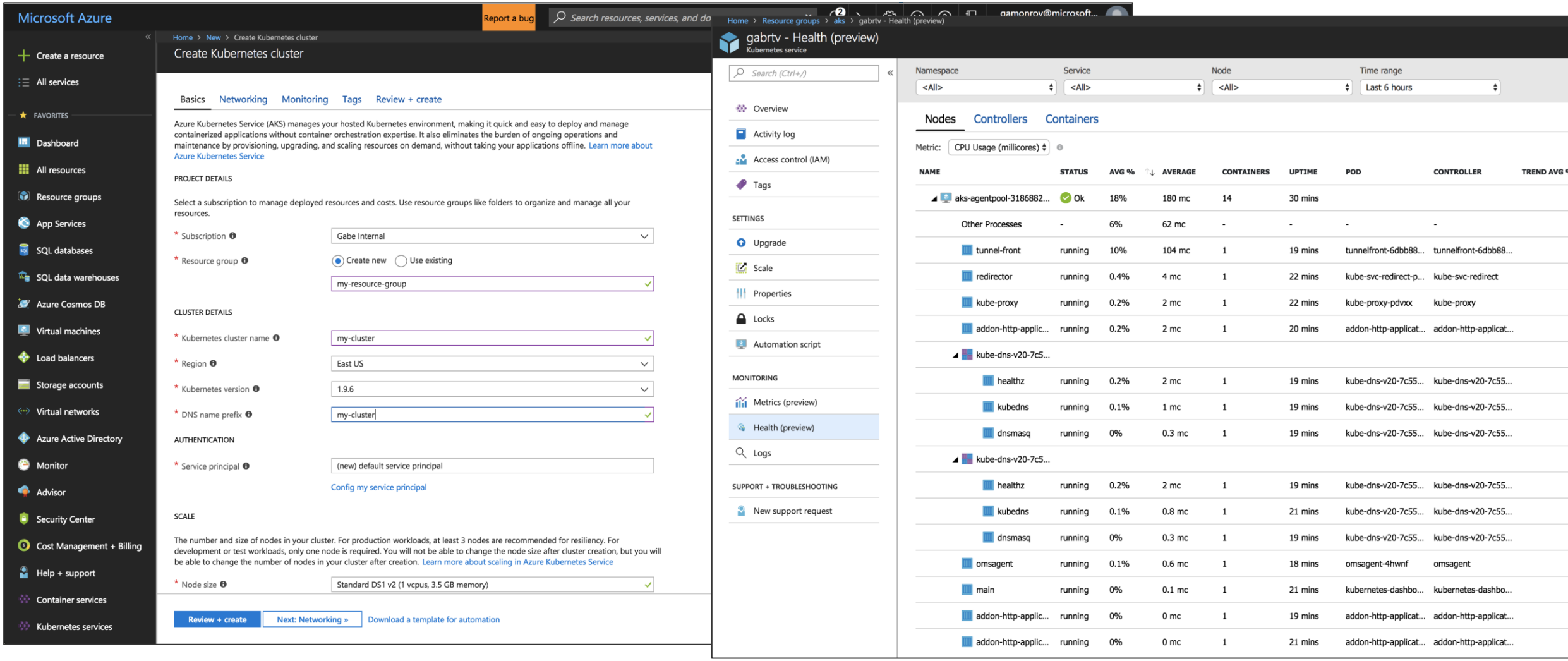
Cluster Upgrade

Upgrade to version 1.11.4

```
$ az aks upgrade --name myAKSCluster --resource-group myResourceGroup --
kubernetes-version 1.11.4
```

- The Kubernetes community releases minor versions roughly every three months
- AKS supports *4* minor versions of Kubernetes
  - The latest stable version upstream and the previous 3
- Each supported minor version also supports *2* stable patches.

AKS – Portal Experience

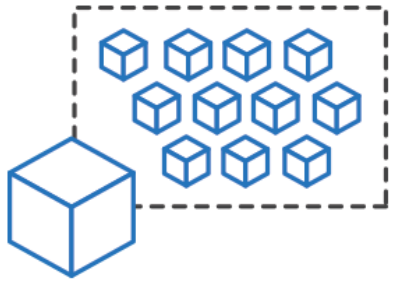


Microsoft

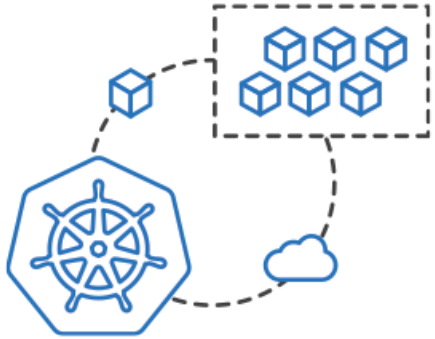
Azure Container Instances (ACI)

Azure Container Instances (ACI)

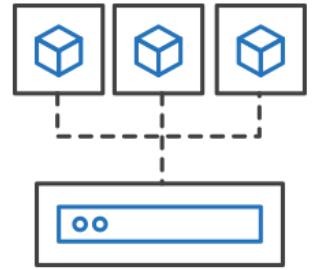
Easily run containers on Azure without managing servers



Run containers without managing servers



Increase agility with containers on demand



Secure applications with isolation

Release automation tools

Simplifying the Kubernetes experience

The package manager for Kubernetes

Streamlined Kubernetes development

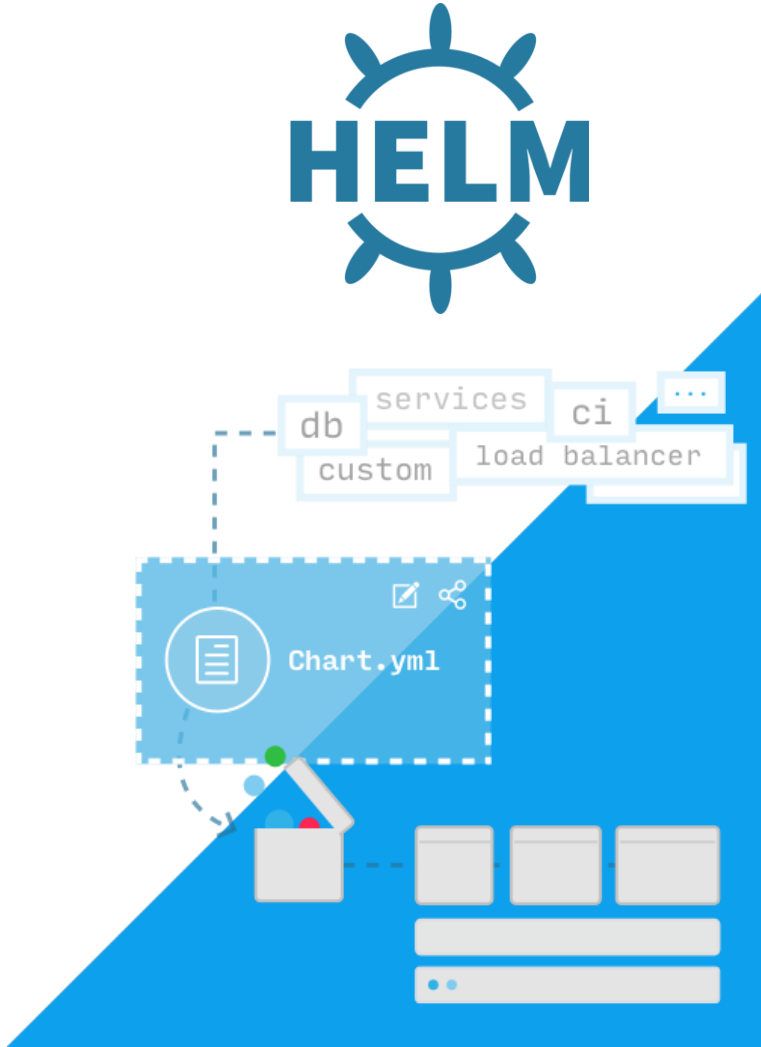
Event-driven scripting for Kubernetes

CLOUD NATIVE COMPUTING FOUNDATION

Helm

What is Helm?

- Helm helps you manage Kubernetes applications
- Helm Charts helps you define, install, and upgrade even the most complex Kubernetes application.



More on helm.sh

Looking for sample charts?

Visit the **Chart Directory** at **kubeapps.com** to explore and use community charts.

Resources

Where	What
AKS Workshop	MS Learn – AKS Workshop
k8s on Azure	Kubernetes on Azure
AKS	Azure Kubernetes Services
AKS on GitHub	AKS Issues and feature Tracking
AKS DevOps labs	Deploying multi-container apps in AKS

Microsoft

Thank You