



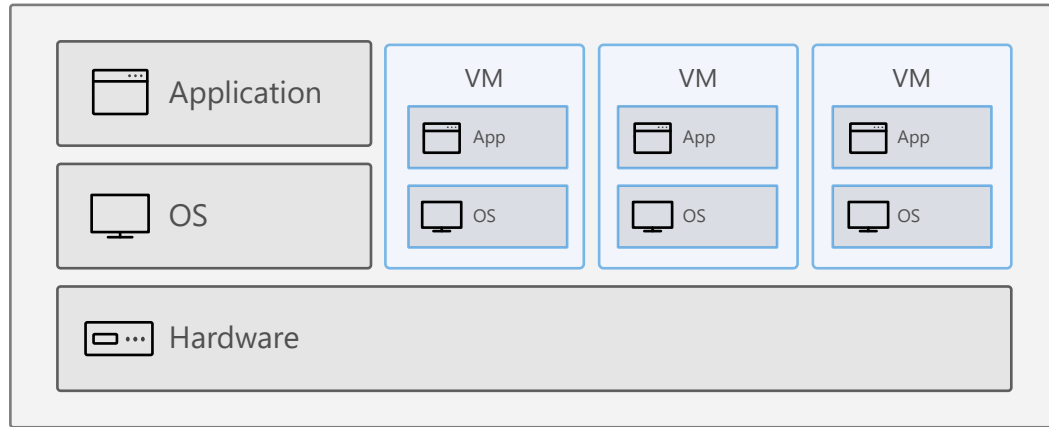
# Azure Kubernetes Service

Jay Kumar

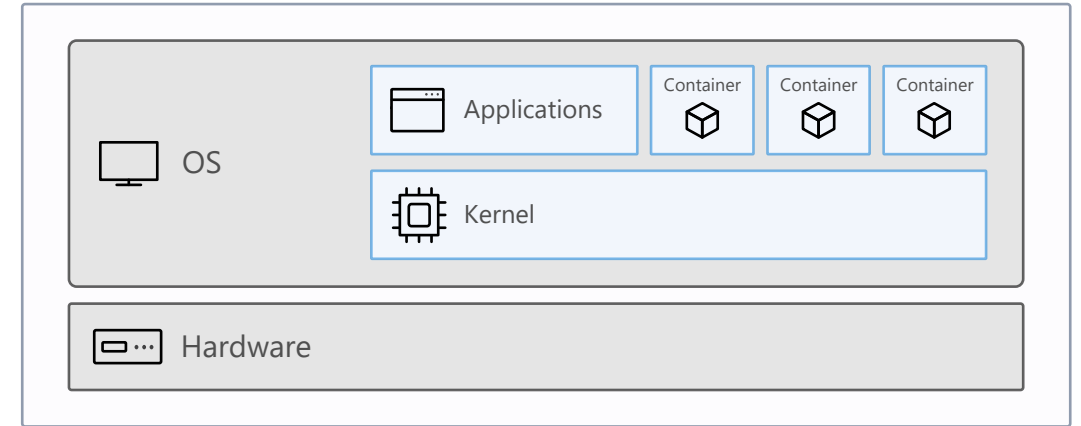
Sr. Cloud Solution Architect, Customer Success

# What is a **container**?

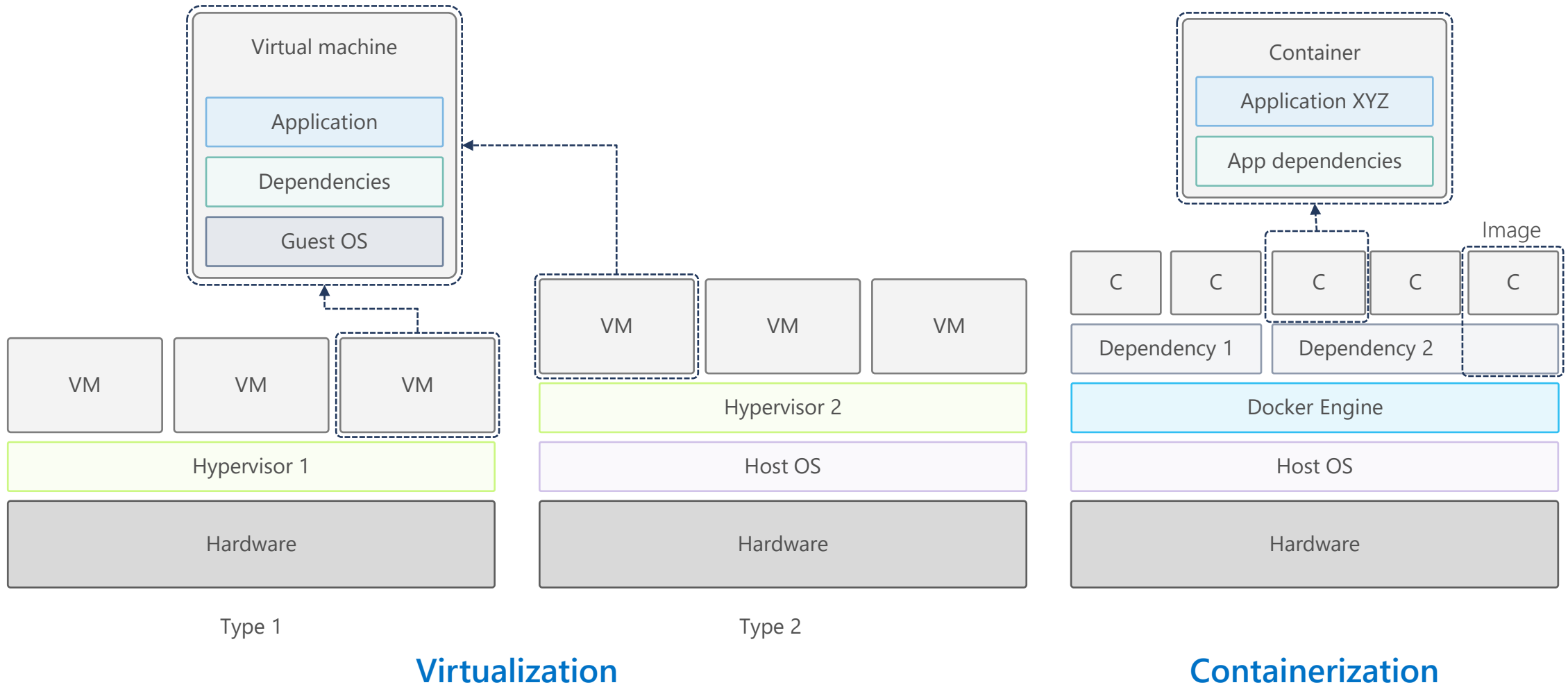
Traditional virtual machines = hardware virtualization



**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).

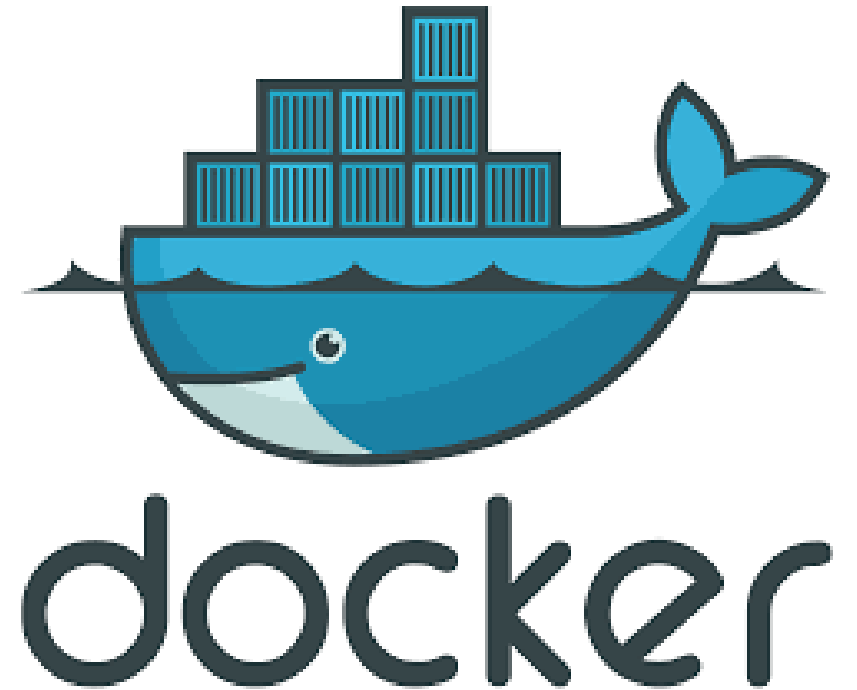
# Docker

**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.

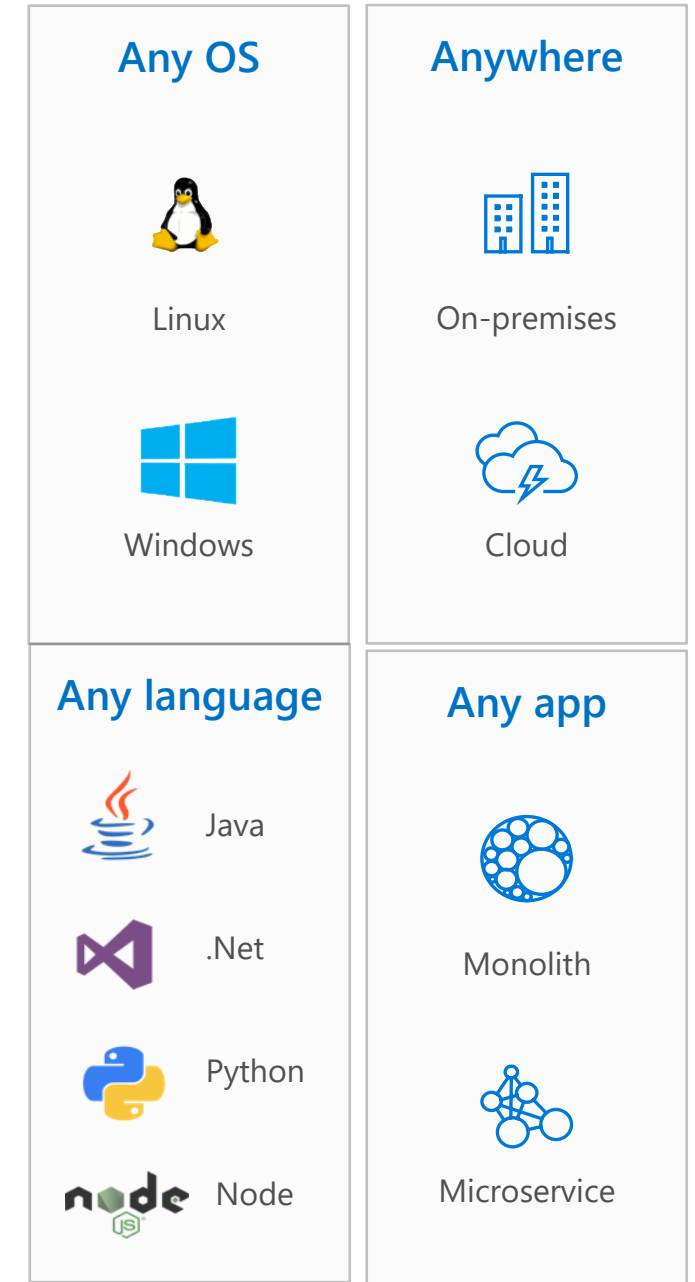
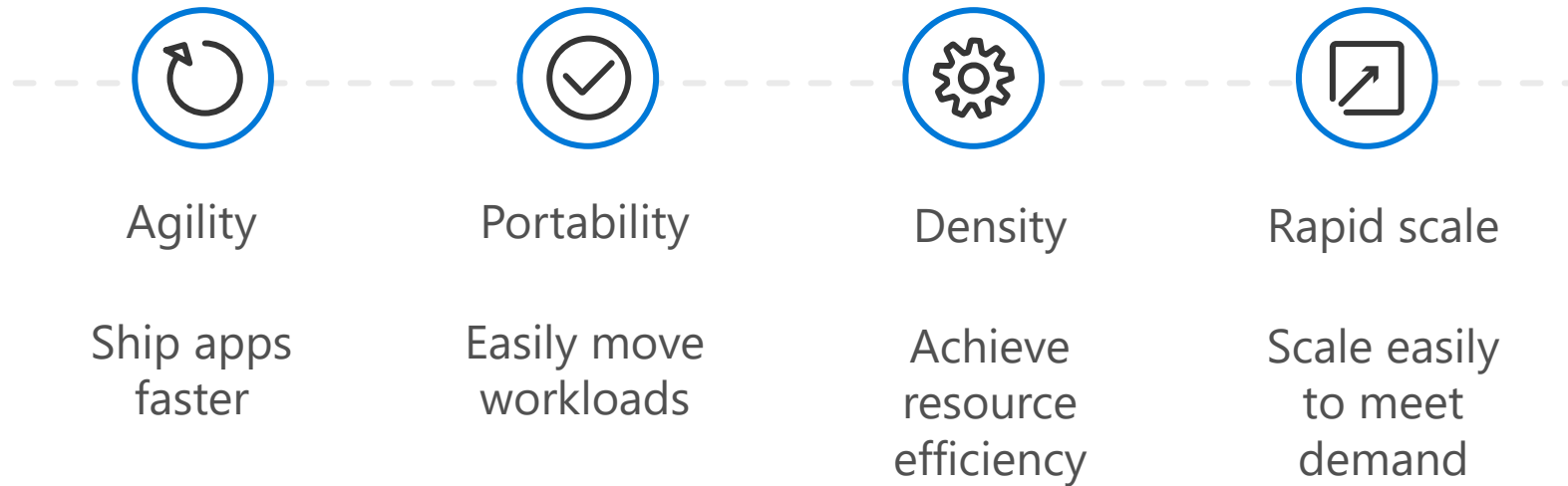
Docker is used to run software packages called "containers". In a typical example use case, one container runs a web server and web application, while a second container runs a database server that is used by the web application.

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

[https://en.wikipedia.org/wiki/Docker\\_\(software\)](https://en.wikipedia.org/wiki/Docker_(software))



# The benefits of using containers



# 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 -----

# Container Challenges





# 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 set of containers elastic in number

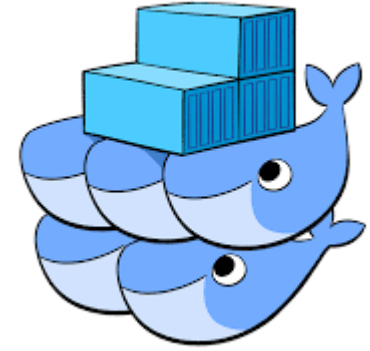
container images

containers and cluster

# Available Orchestrators

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

Kubernetes



MESOS



HashiCorp  
**Nomad**



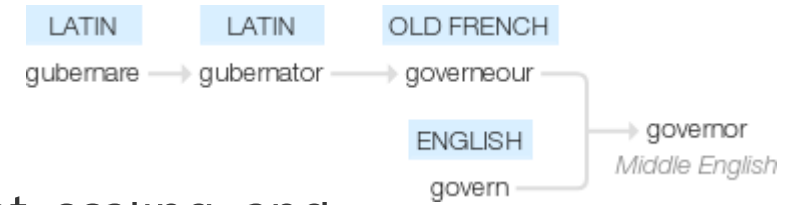
**RANCHER**



Microsoft Azure  
Service Fabric



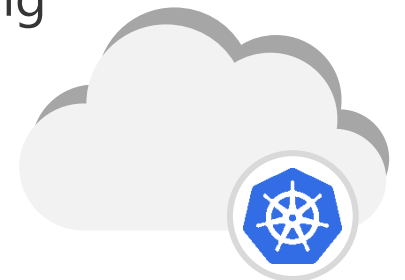
# What is Kubernetes (k8s)?



- **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.

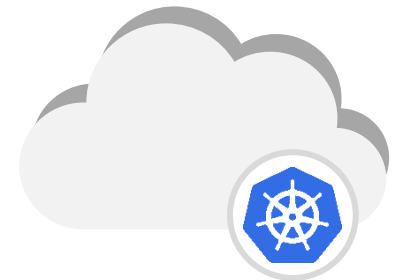
## History

- 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

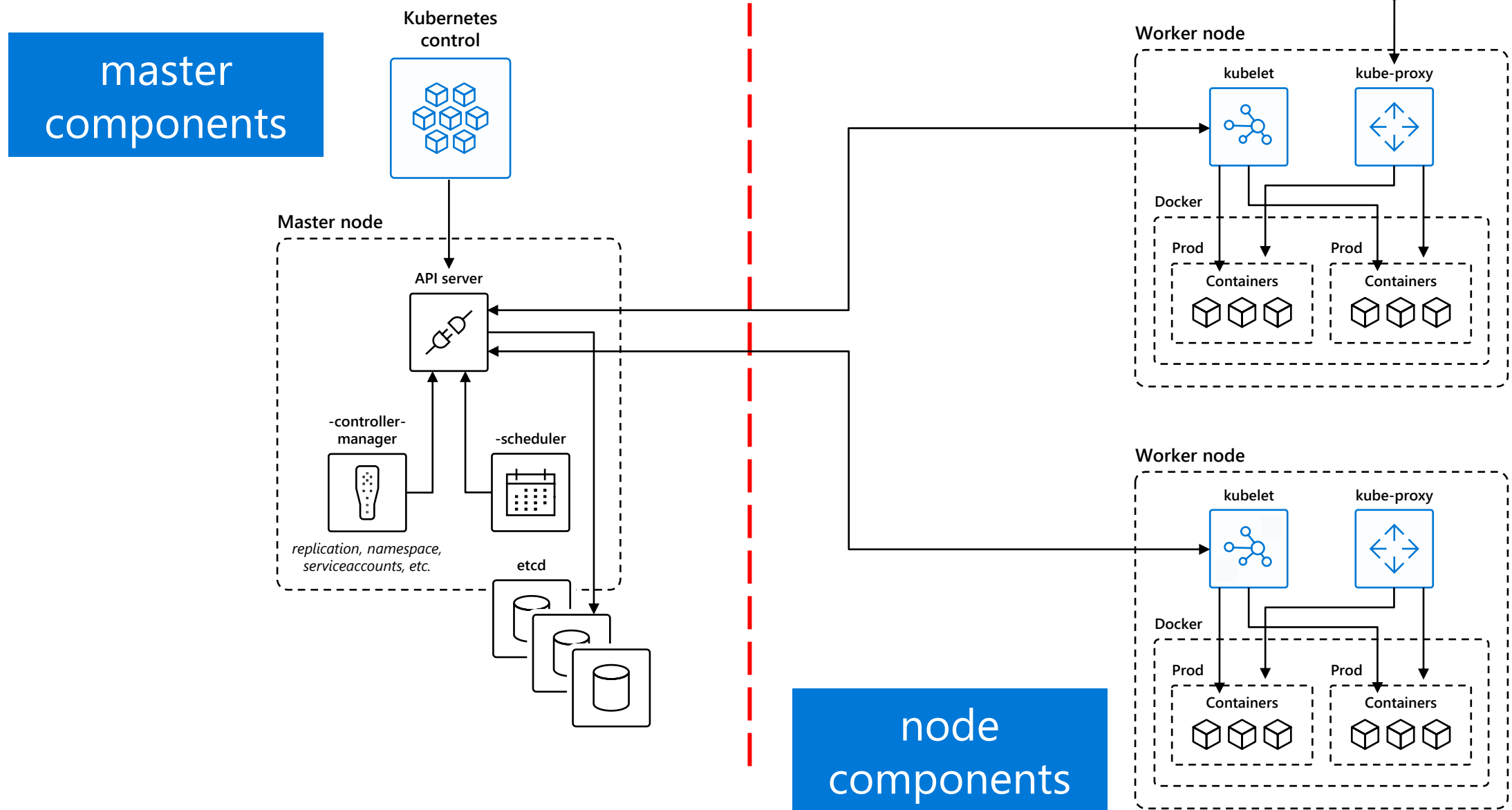


# Kubernetes Features

- 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

api-server

etcd

controller-manager

scheduler

master  
components

kubelet

kube-proxy

docker

dns

node  
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>





# Kubernetes in Azure (AKS)

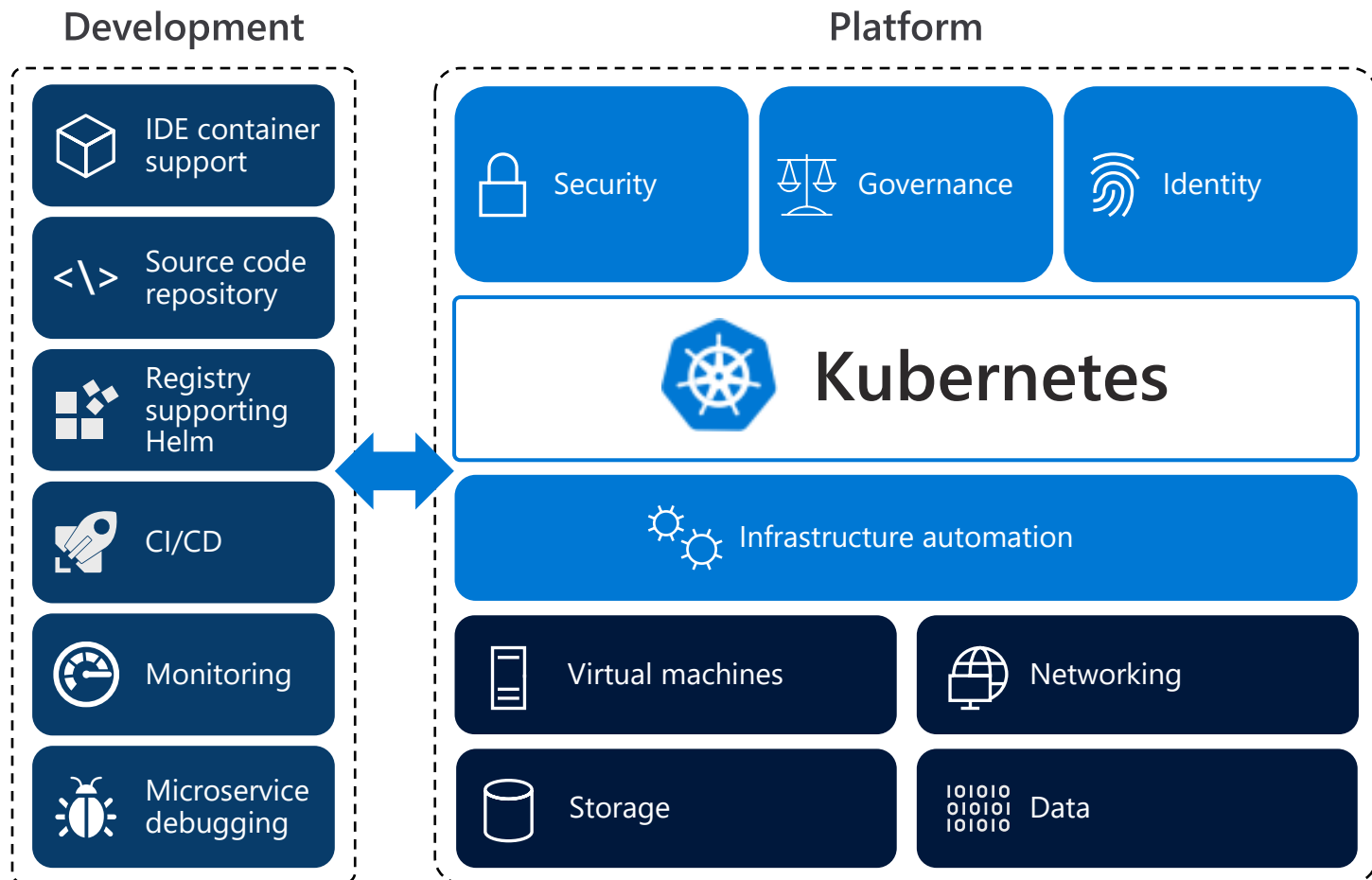
Managed k8s service

# Kubernetes on its own is **not enough**

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

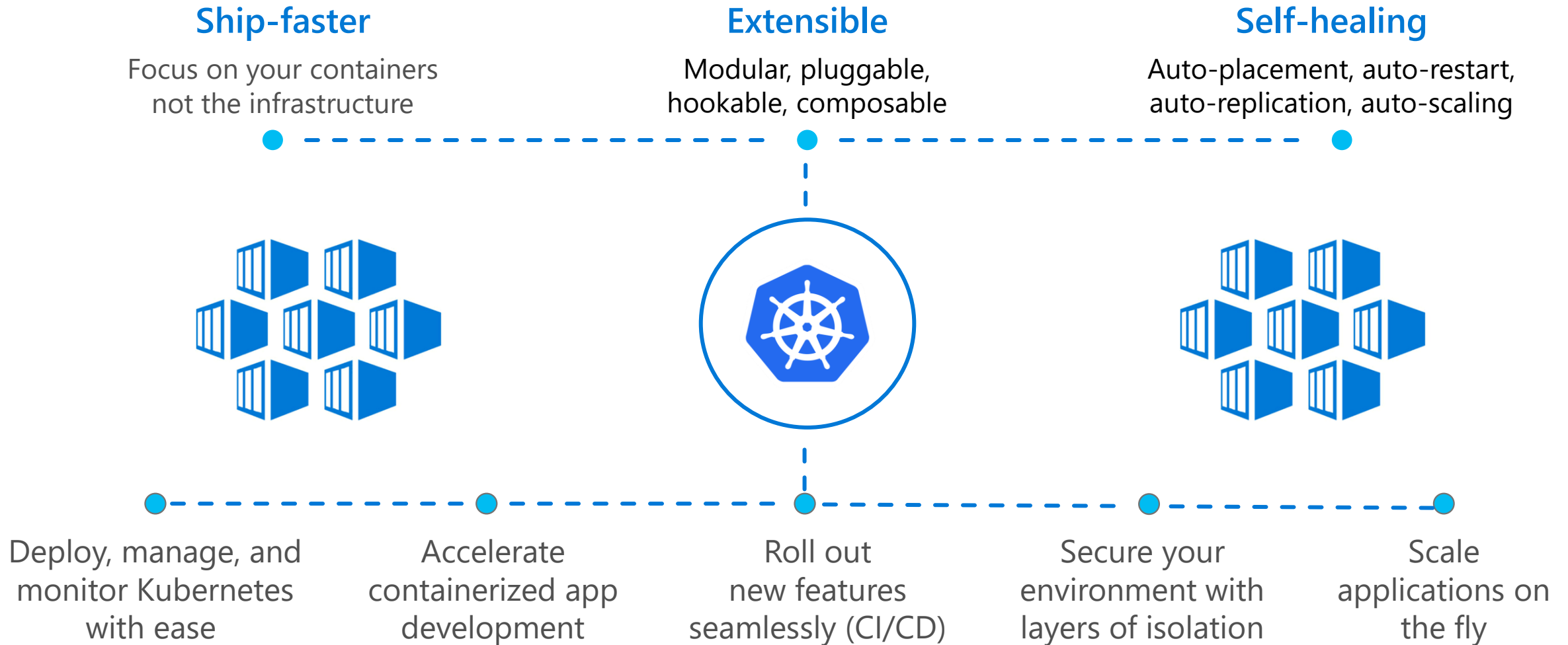
Unlock the agility for containerized applications using:

- **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**



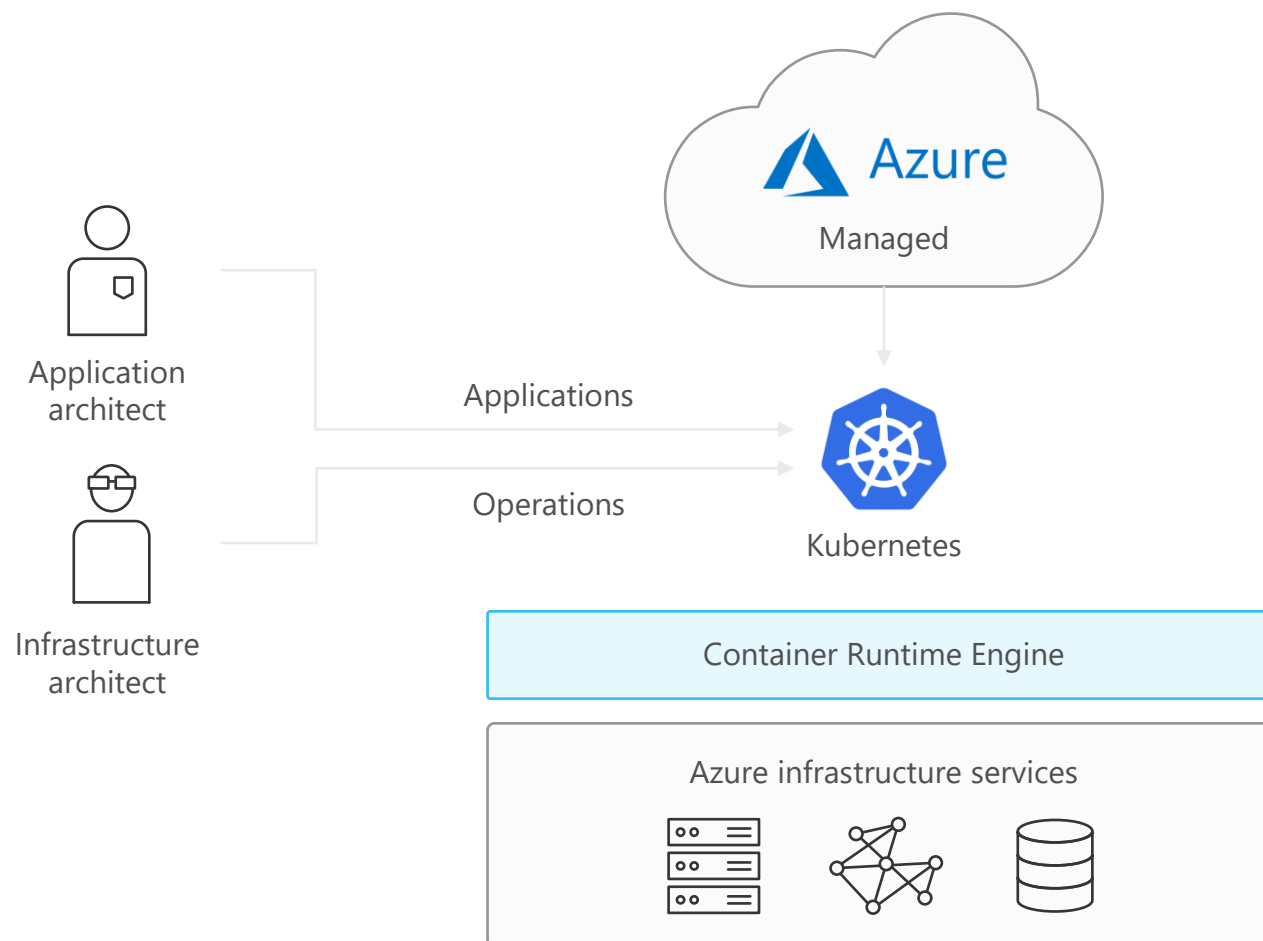
# Azure Kubernetes Service

Simplify the deployment, management, and operations of Kubernetes

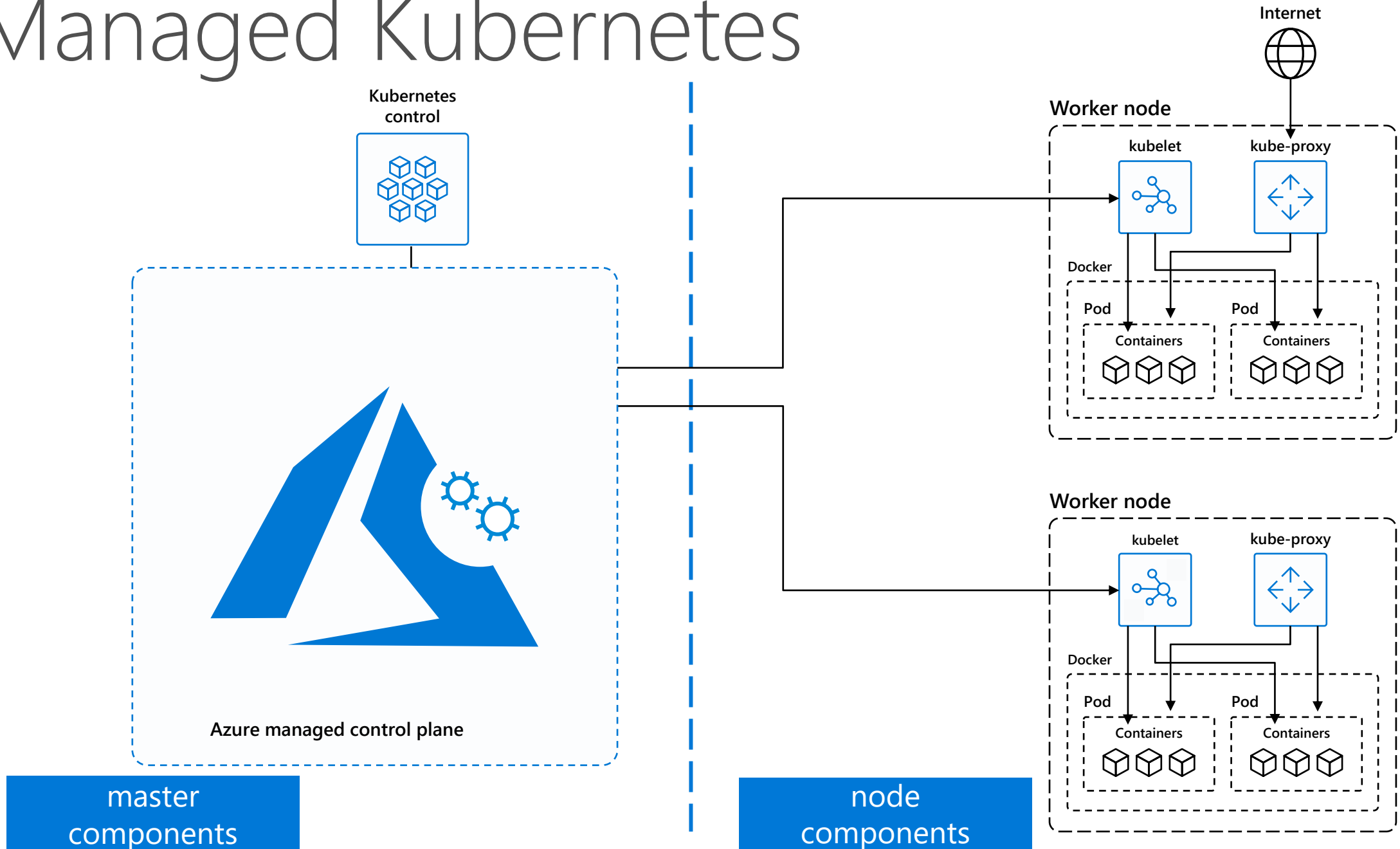


# Azure Kubernetes Service (AKS)

A fully managed Kubernetes cluster

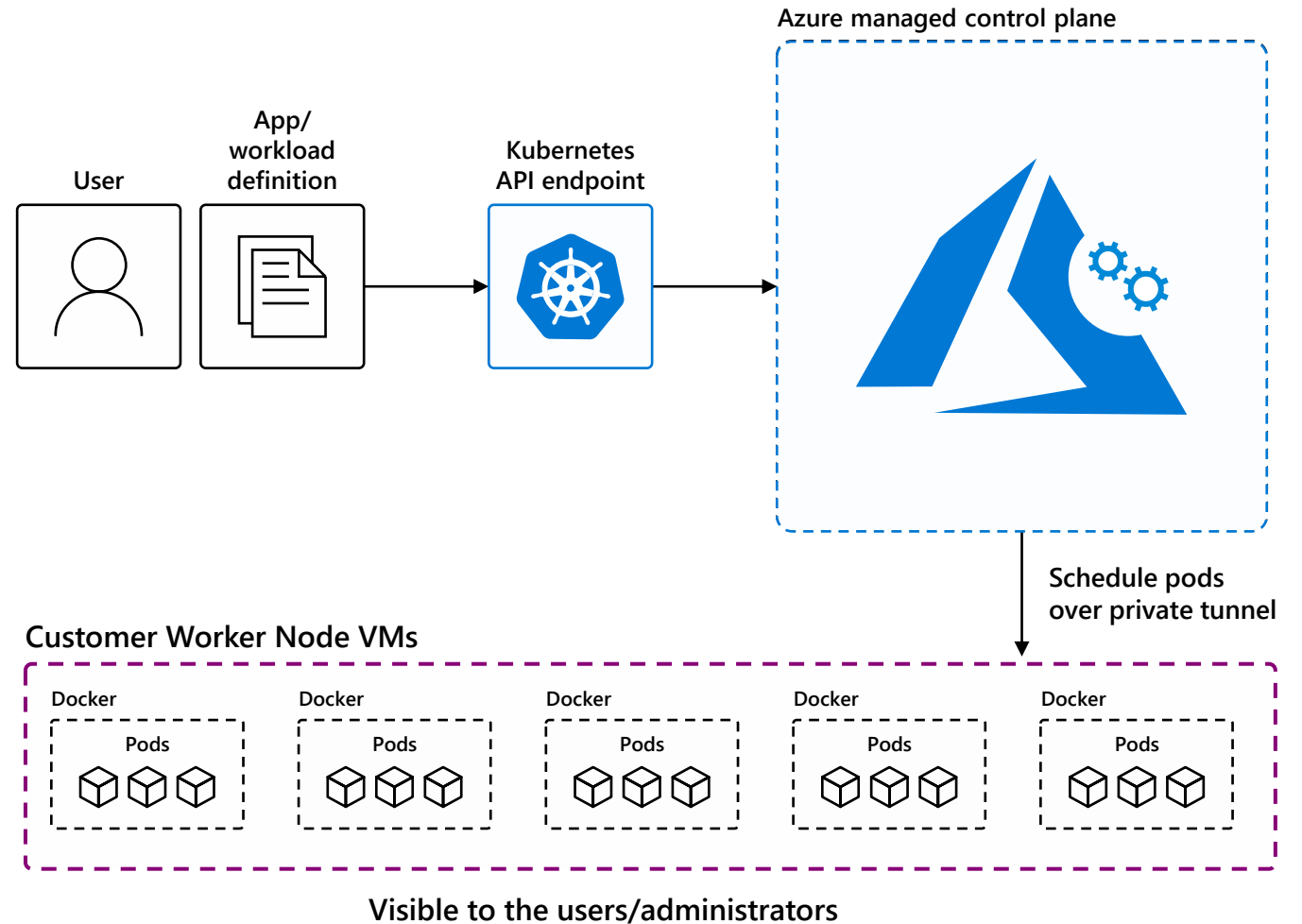


# Managed Kubernetes






# 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	<code>az aks create</code>
Upgrade a cluster	Upgrade your master nodes Cordon/drain and upgrade worker nodes individually	<code>az aks upgrade</code>
Scale a cluster	Provision new VMs Install system components Register nodes with API server	<code>az aks scale</code>

# 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/kubect1 ..
```

```
> 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

Create a resource

All services

FAVORITES

Dashboard

All resources

Resource groups

App Services

SQL databases

SQL data warehouses

Azure Cosmos DB

Virtual machines

Load balancers

Storage accounts

Virtual networks

Azure Active Directory

Monitor

Advisor

Security Center

Cost Management + Billing

Help + support

Container services

Kubernetes services

Report a bug

Search resources, services, and docs

Home > New > Create Kubernetes cluster

Create Kubernetes cluster

Basics Networking Monitoring Tags Review + create

Azure Kubernetes Service (AKS) manages your hosted Kubernetes environment, making it quick and easy to deploy and manage containerized applications without container orchestration expertise. It also eliminates the burden of ongoing operations and maintenance by provisioning, upgrading, and scaling resources on demand, without taking your applications offline. [Learn more about Azure Kubernetes Service](#)

PROJECT DETAILS

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

\* Subscription ⓘ Gabe Internal

\* Resource group ⓘ ☒ Create new ☐ Use existing  
my-resource-group ✓

CLUSTER DETAILS

\* Kubernetes cluster name ⓘ my-cluster ✓

\* Region ⓘ East US

\* Kubernetes version ⓘ 1.9.6

\* DNS name prefix ⓘ my-cluster ✓

AUTHENTICATION

\* Service principal ⓘ (new) default service principal  
[Config my service principal](#)

SCALE

The number and size of nodes in your cluster. For production workloads, at least 3 nodes are recommended for resiliency. For development or test workloads, only one node is required. You will not be able to change the node size after cluster creation, but you will be able to change the number of nodes in your cluster after creation. [Learn more about scaling in Azure Kubernetes Service](#)

\* Node size ⓘ Standard DS1 v2 (1 vcpus, 3.5 GB memory) ✓

Review + create

Next: Networking »

Download a template for automation

gabrty - Health (preview)

Kubernetes service

Search (Ctrl+J)

Overview

Activity log

Access control (IAM)

Tags

SETTINGS

Upgrade

Scale

Properties

Locks

Automation script

MONITORING

Metrics (preview)

Health (preview)

Logs

SUPPORT + TROUBLESHOOTING

New support request

Namespace Service Node Time range

<All> <All> <All> Last 6 hours

Nodes Controllers Containers

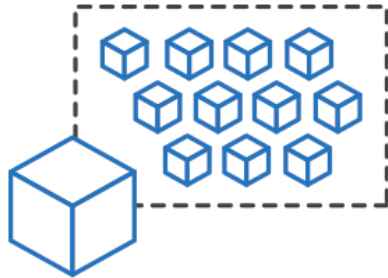
Metric: CPU Usage (millicores)

NAME	STATUS	AVG %	AVERAGE	CONTAINERS	UPTIME	POD	CONTROLLER	TREND AVG
aks-agentpool-3186882...	Ok	18%	180 mc	14	30 mins			
Other Processes	-	6%	62 mc	-	-	-	-	
tunnel-front	running	10%	104 mc	1	19 mins	tunnelfront-6dbb88...	tunnelfront-6dbb88...	
redirector	running	0.4%	4 mc	1	22 mins	kube-svc-redirect-p...	kube-svc-redirect	
kube-proxy	running	0.2%	2 mc	1	22 mins	kube-proxy-pdvxx	kube-proxy	
addon-http-applic...	running	0.2%	2 mc	1	20 mins	addon-http-applicat...	addon-http-applicat...	
kube-dns-v20-7c5...								
healthz	running	0.2%	2 mc	1	19 mins	kube-dns-v20-7c55...	kube-dns-v20-7c55...	
kubedns	running	0.1%	1 mc	1	19 mins	kube-dns-v20-7c55...	kube-dns-v20-7c55...	
dnsmasq	running	0%	0.3 mc	1	19 mins	kube-dns-v20-7c55...	kube-dns-v20-7c55...	
kube-dns-v20-7c5...								
healthz	running	0.2%	2 mc	1	19 mins	kube-dns-v20-7c55...	kube-dns-v20-7c55...	
kubedns	running	0.1%	0.8 mc	1	21 mins	kube-dns-v20-7c55...	kube-dns-v20-7c55...	
dnsmasq	running	0%	0.3 mc	1	19 mins	kube-dns-v20-7c55...	kube-dns-v20-7c55...	
omsagent	running	0.1%	0.6 mc	1	18 mins	omsagent-4hwnf	omsagent	
main	running	0%	0.1 mc	1	21 mins	kubernetes-dashbo...	kubernetes-dashbo...	
addon-http-applic...	running	0%	0 mc	1	19 mins	addon-http-applicat...	addon-http-applicat...	
addon-http-applic...	running	0%	0 mc	1	21 mins	addon-http-applicat...	addon-http-applicat...	

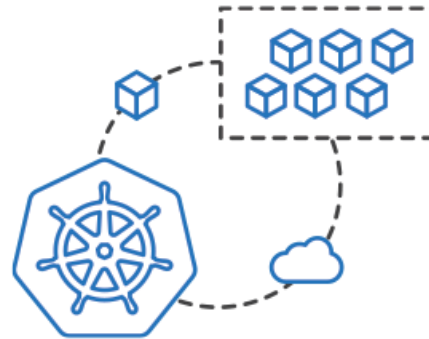
# Azure Container Instances (ACI)

# Azure Container Instances (ACI)

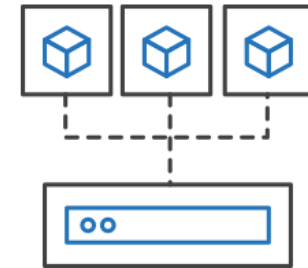
Easily run containers on Azure without managing servers



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



# Helm

What is Helm?

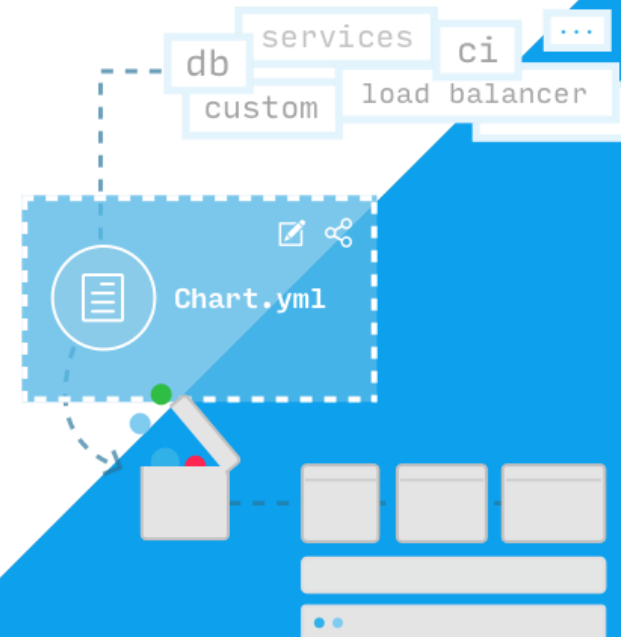
- Helm helps you manage Kubernetes applications
- Helm Charts helps you define, install, and upgrade even the most complex Kubernetes application.

More on [helm.sh](https://helm.sh)



Looking for sample charts?

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



# Resources

Where	What
AKS Workshop	<a href="#"><u>MS Learn – AKS Workshop</u></a>
k8s on Azure	<a href="#"><u>Kubernetes on Azure</u></a>
AKS	<a href="#"><u>Azure Kubernetes Services</u></a>
AKS on GitHub	<a href="#"><u>AKS Issues and feature Tracking</u></a>
AKS DevOps labs	<a href="#"><u>Deploying multi-container apps in AKS</u></a>





Thank You