The State of Containers for your DevOps journey

Build Cloud Native Apps

Agile Tour Montréal 2018

Mathieu Benoit

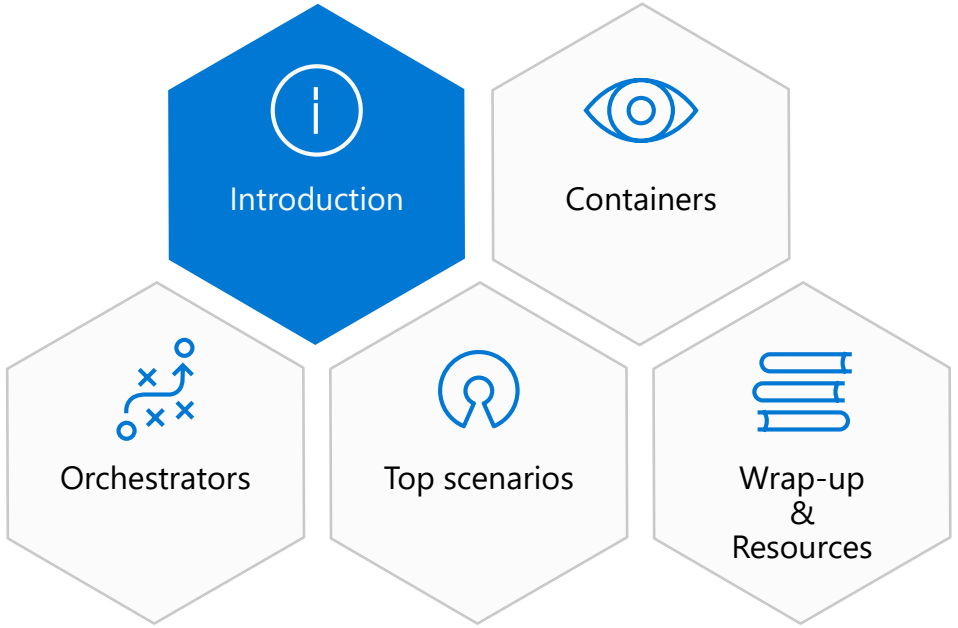
Cloud Solution Architect
Microsoft Canada

https://aka.ms/mabenoit

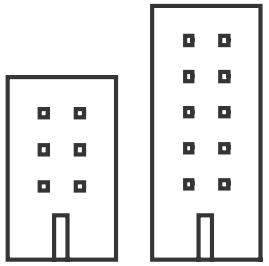
Objectives

- Docker, improves packaging any apps, any languages
- CaaS, runs quickly any Containers by abstracting the infra
- Kubernetes, simplifies the orchestration of your Containers
- Helm, unifies and automates your Kubernetes deployments
- CI/CD pipeline, delivers your code to production
- Cloud, accelerates your modernization journey
- Open Source, is everywhere and is leading innovation

Introduction



Widening divide between business and IT

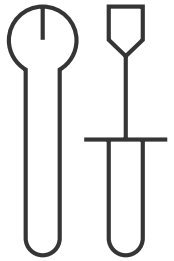


Business needs

Rapid innovation to transform products

Close the gap from data to decision

Connect with customers
and empower employees



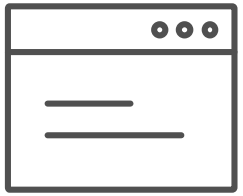
IT challenges

72% of IT budgets are dedicated towards maintenance (‘keeping the lights on’).

Only half of all decision makers got help from technologists with their analysis needs

Demand for mobile apps
>5x the capacity of IT

What we hear from **developers**



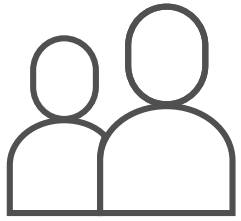
I need to create applications at a competitive rate without worrying about IT

New applications run smoothly on my machine but malfunction on traditional IT servers

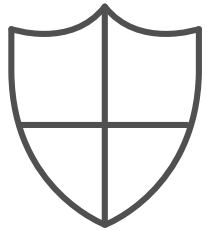
My productivity and application innovation become suspended when I have to wait on IT

What we hear from **IT**

I need to manage servers and maintain compliance with little disruption

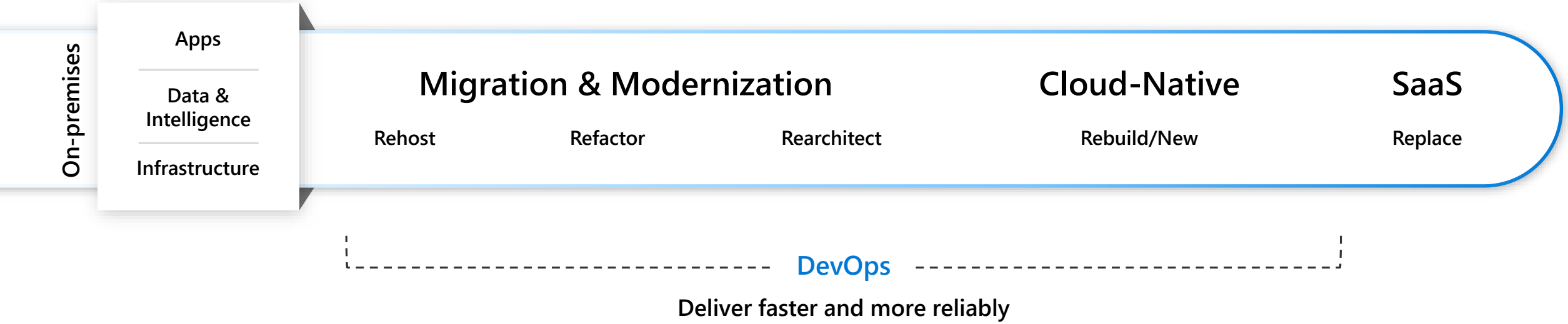


I'm unsure of how to integrate unfamiliar applications, and I require help from developers

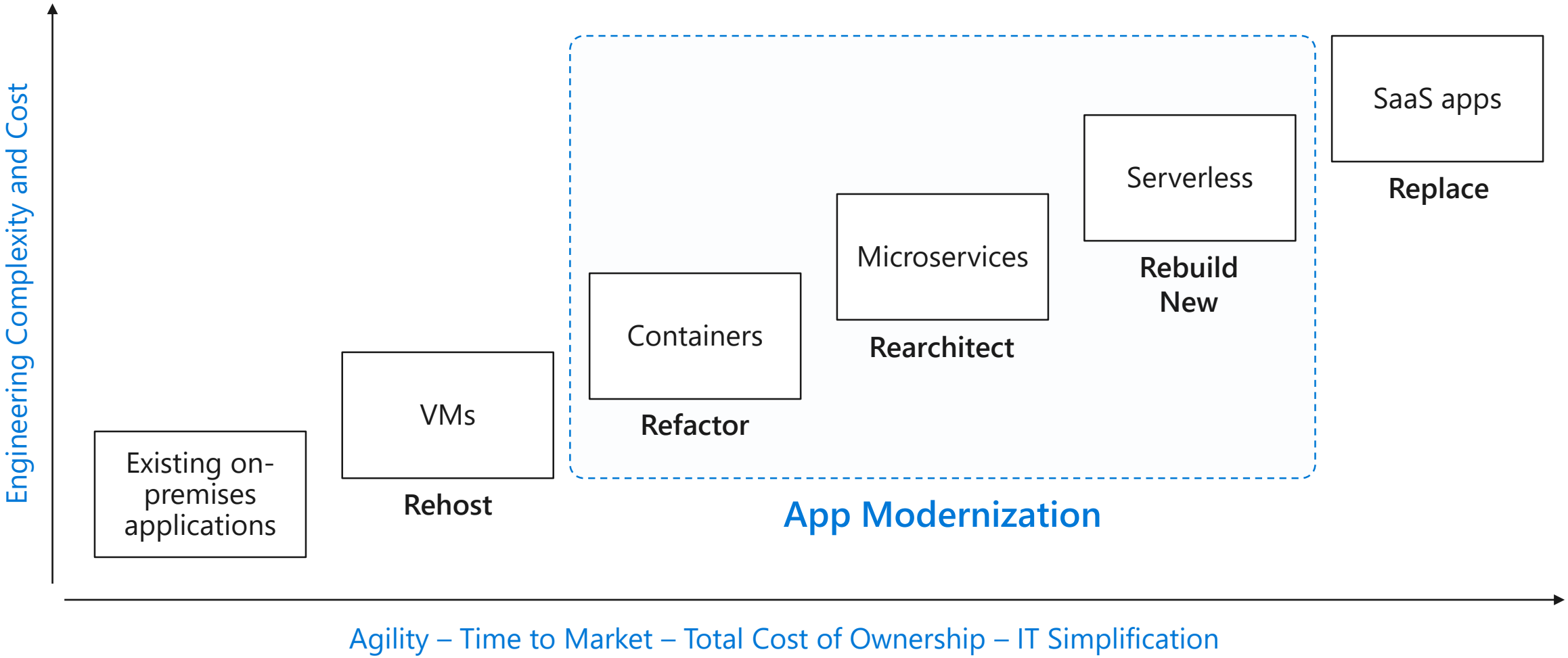


I'm unable to focus on both server protection and application compliance

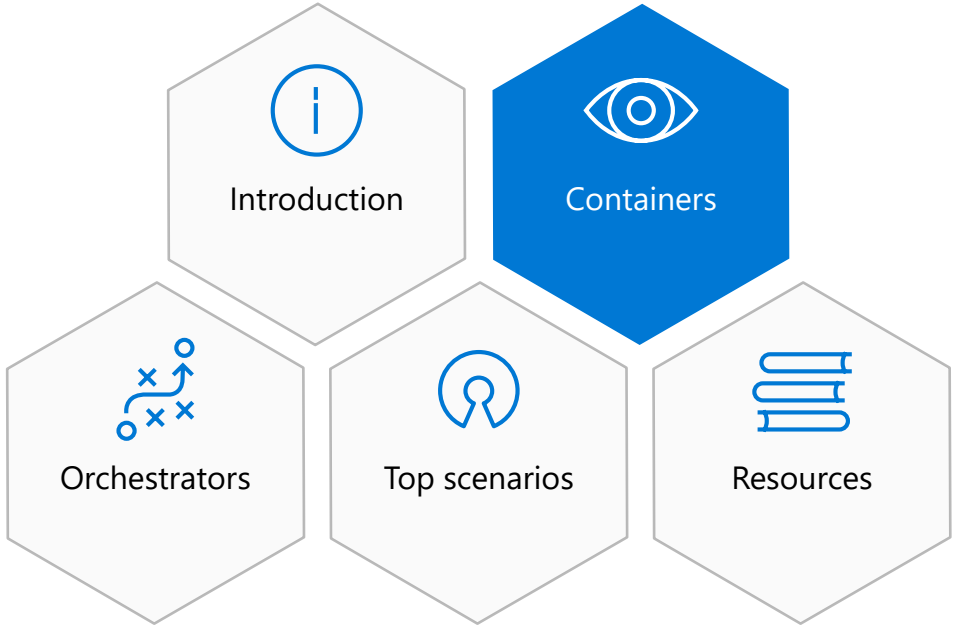
The **journey** to the cloud



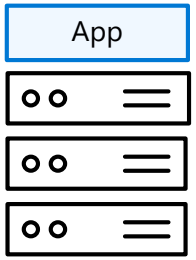
Cloud app **continuum**



Containers - Docker



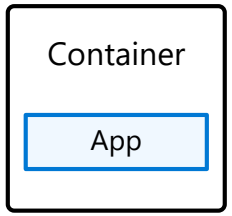
What is a **container**?



Virtual machines

Virtualize the hardware

VMs as units of scaling

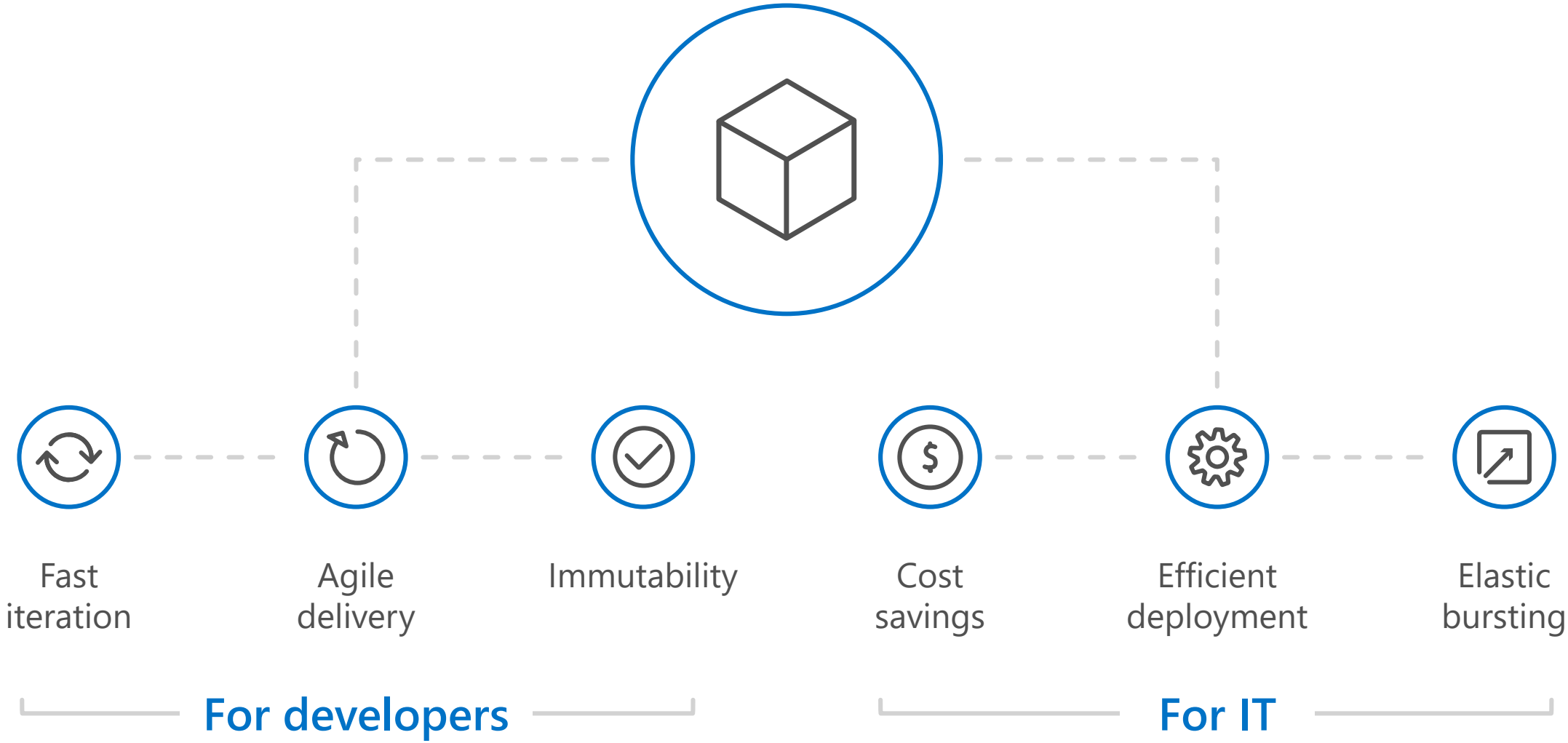


Containers

Virtualize the operating system

Applications as units of scaling

The container **advantage**



The benefits of using containers

Any OS

Linux

Windows

Anywhere

On-premises

Cloud

Any app

Monolith

Microservice

Any language

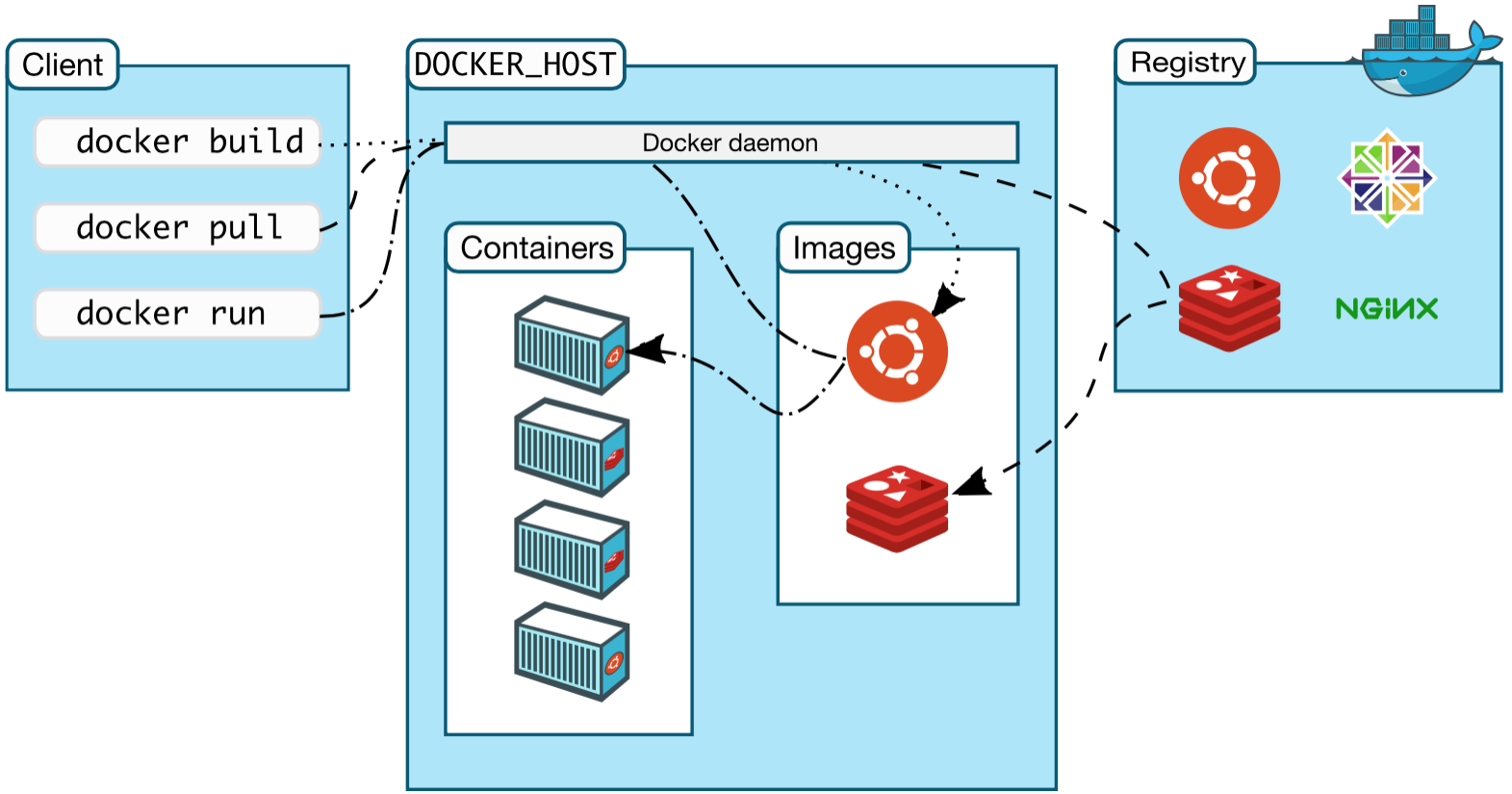
Java

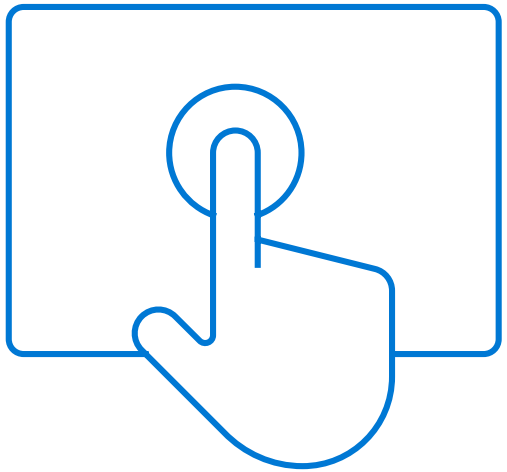
.Net

Python

Node

Docker Architecture & Workflow

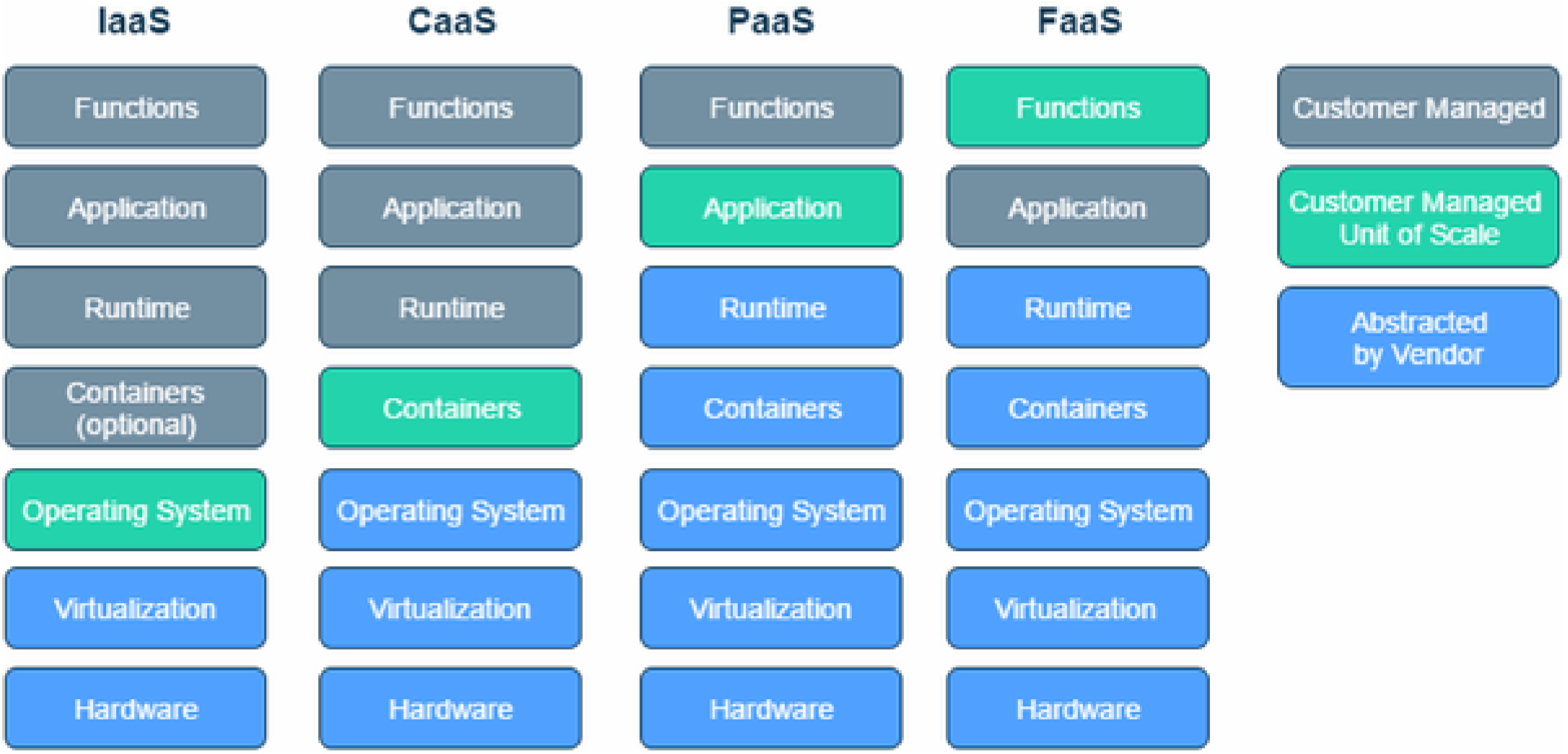




Demo

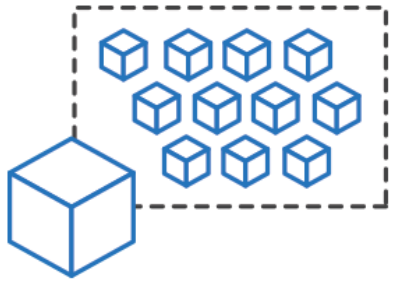
Get started with Docker

Container-as-a-Service (CaaS)

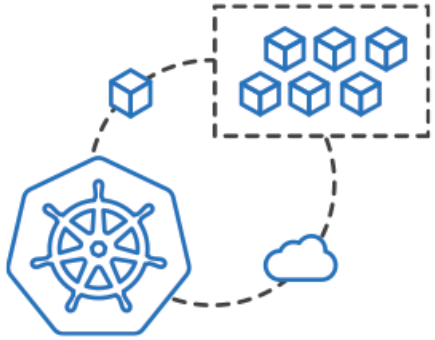


Azure Container Instances (ACI)

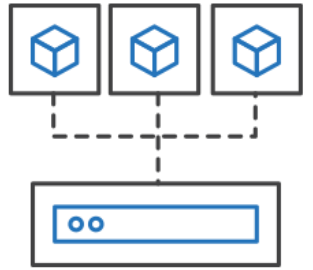
Easily run containers on Azure without managing servers



Run containers without managing servers



Increase agility with containers on demand



Secure applications with hypervisor isolation

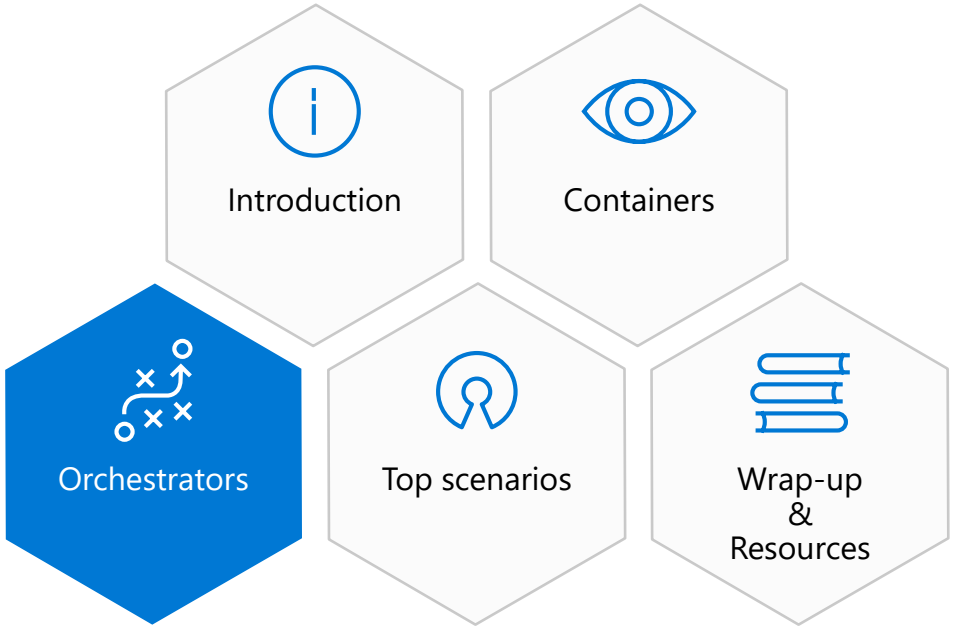
Azure Container Instances (ACI)

ACI in action

```
$ az container create -n mycontainer -i microsoft/aci-helloworld -g myResourceGroup --os-type
Linux --ip-address public
  "ipAddress": {
    "ip": "52.168.86.133",
    "ports": [...]
  },
  "location": "eastus",
  "name": "mycontainer",
  "osType": "Linux",
  "provisioningState": "Succeeded",

$ curl 52.168.86.133
<html>
<head>
  <title>Welcome to Azure Container Instances!</title>
</head>
```

Orchestrators - Kubernetes



The elements of **containers orchestration**

Scheduling

Affinity/anti-affinity

Health monitoring

Failover

Scaling

Networking

Service discovery

Coordinated app upgrades

Kubernetes: the industry leading orchestrator

Portable

Public, private, hybrid, multi-cloud

Extensible

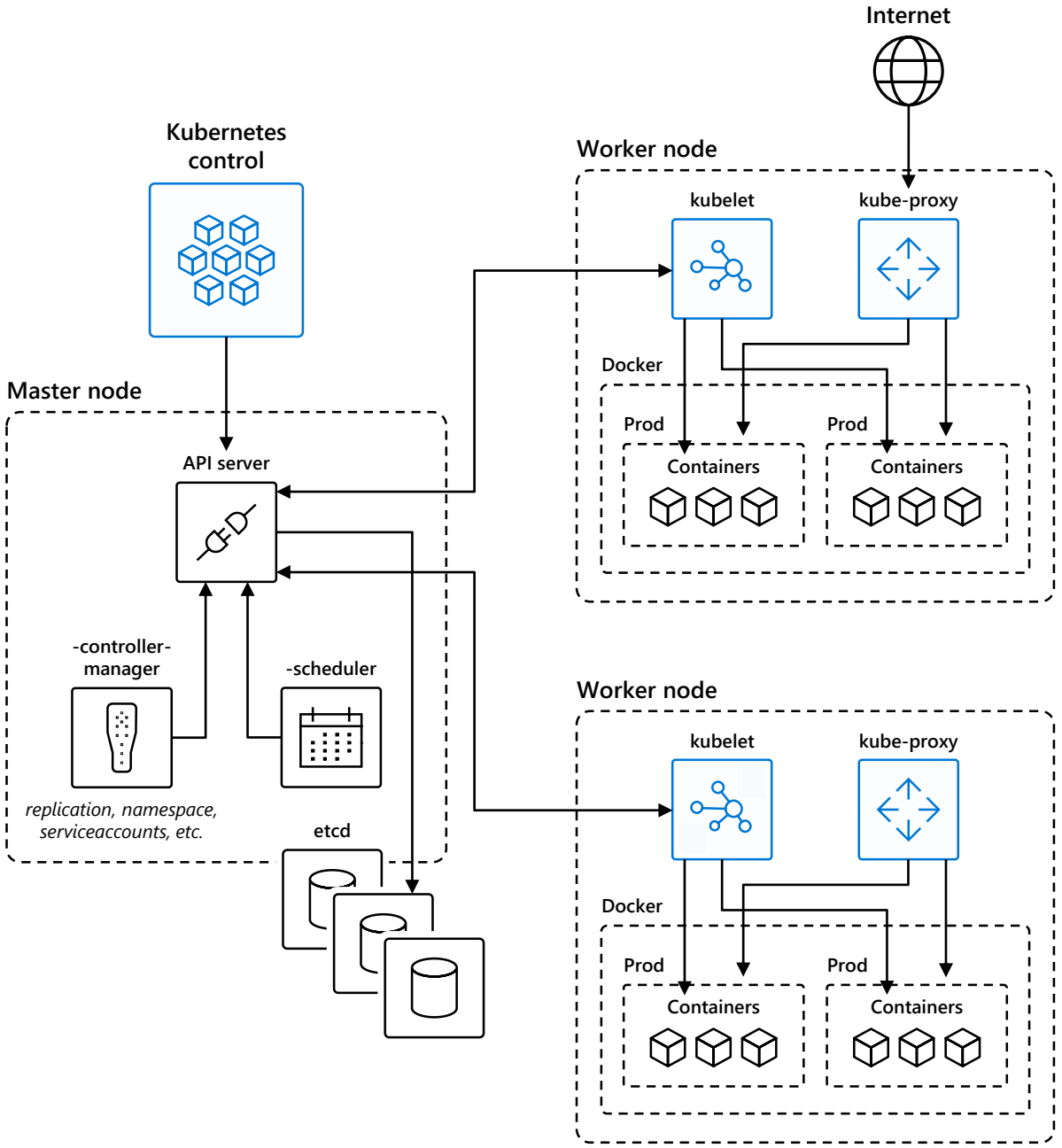
Modular, pluggable, hookable, composable

Self-healing

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

Kubernetes 101

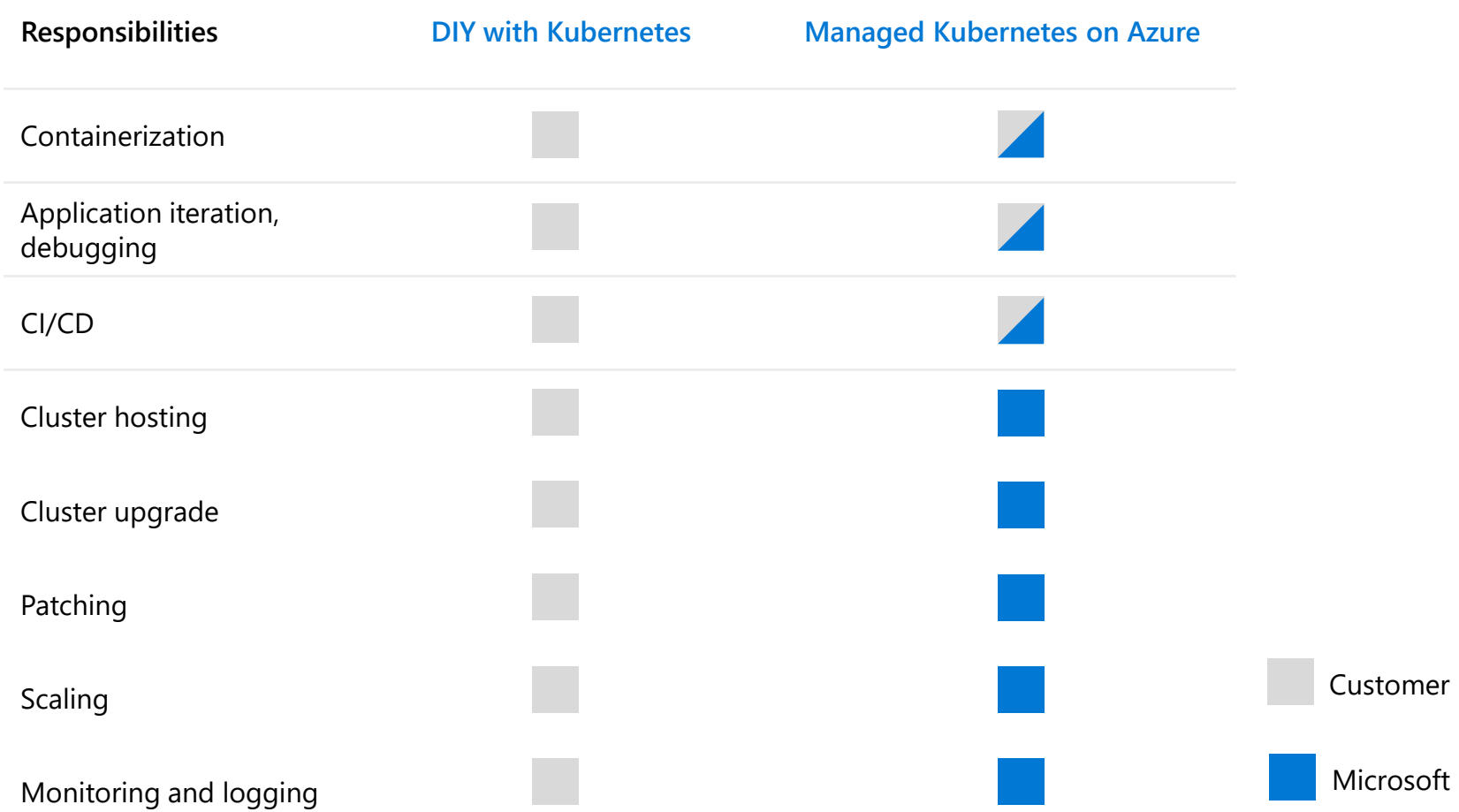
1. Kubernetes users communicate with API server and apply desired state
2. Master nodes actively enforce desired state on worker nodes
3. Worker nodes support communication between containers
4. Worker nodes support communication from the Internet



From infrastructure to **innovation**

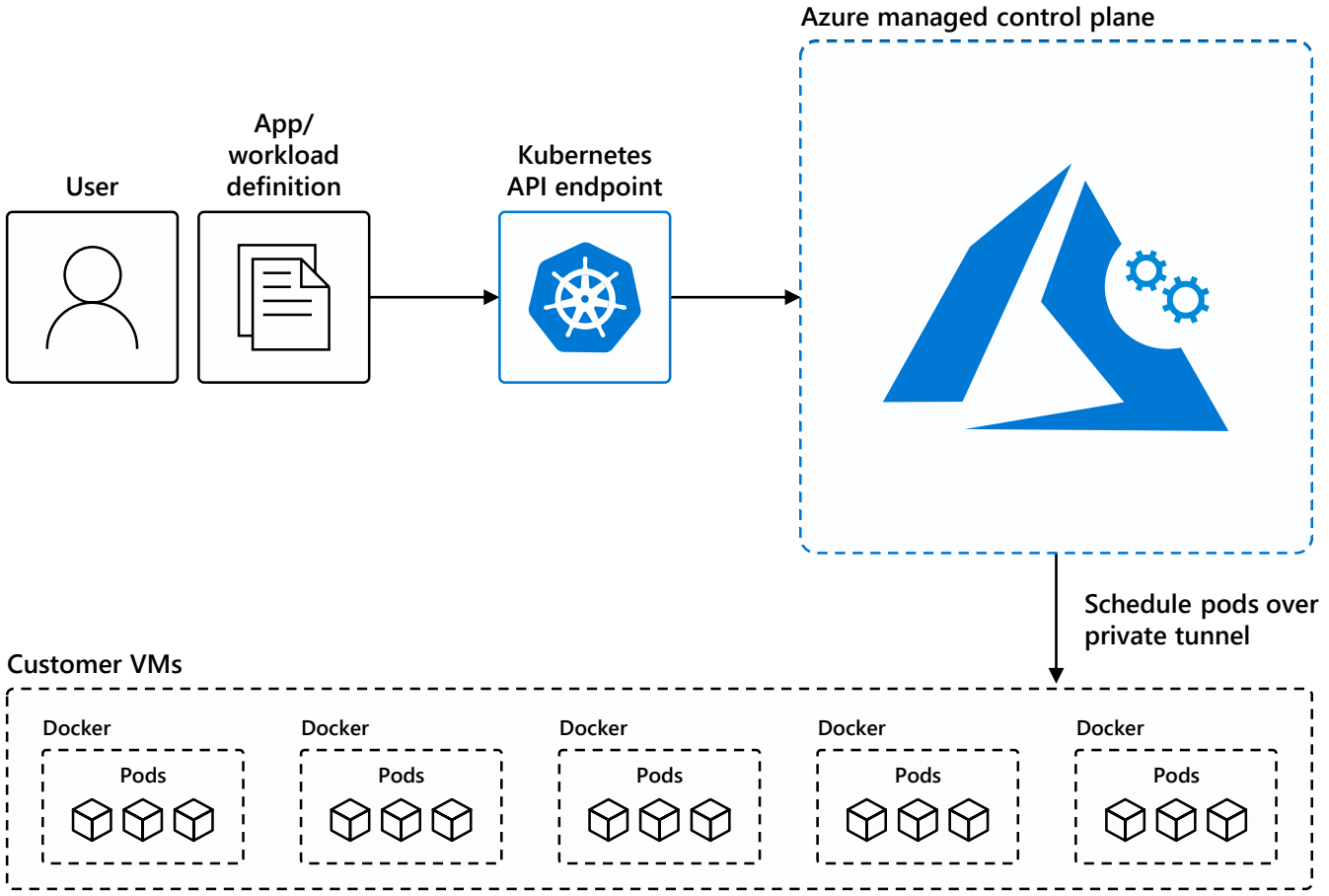
Managed Kubernetes empowers you to do more

Focus on your containers and code, not the plumbing of them



How managed Kubernetes on Azure works

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

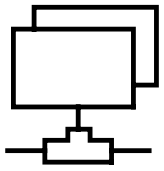


Azure makes Kubernetes easy

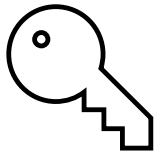
Deploy and manage Kubernetes with ease

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

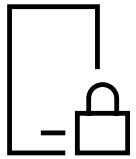
Secure your Kubernetes environment



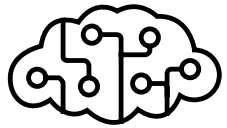
Control access through AAD and RBAC



Safeguard keys and secrets with Key Vault



Secure network communications with VNET and CNI



Compliant Kubernetes service with certifications covering SOC, HIPAA, and PCI

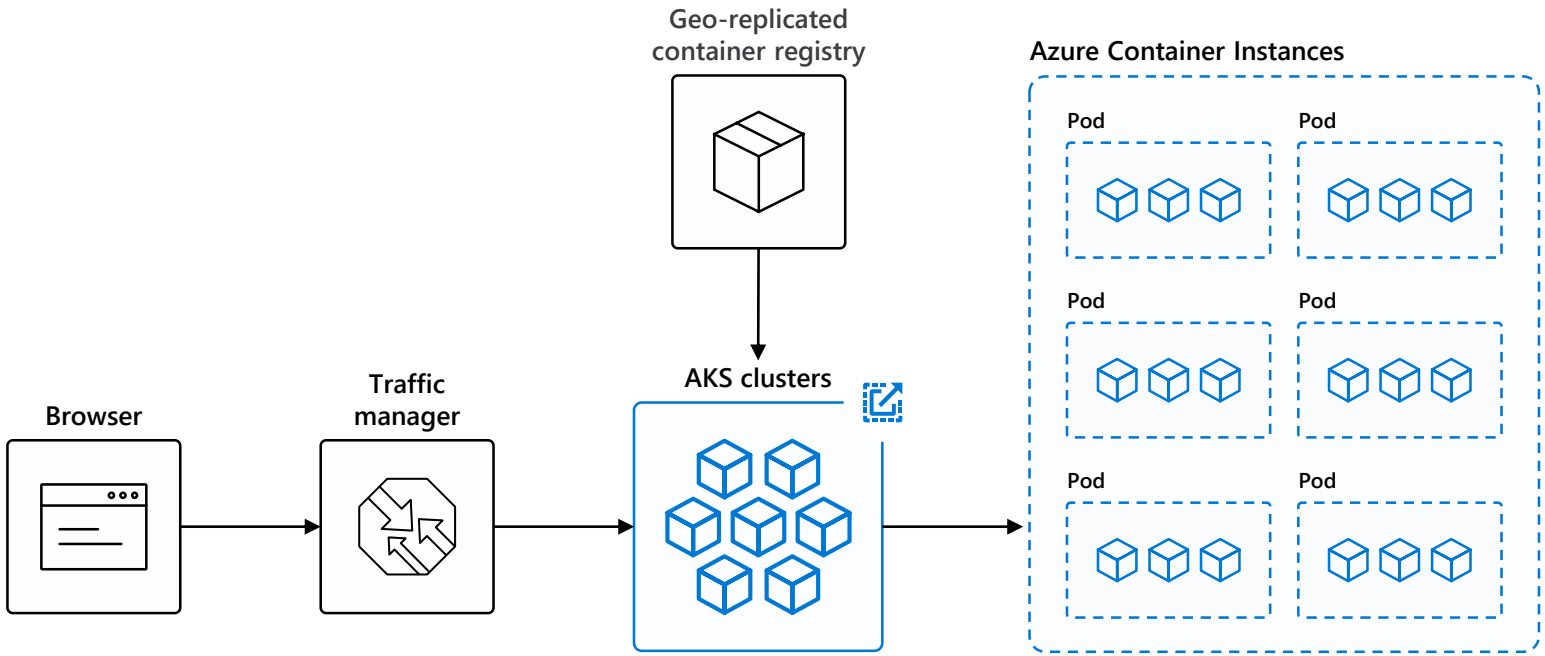
Scale and run with confidence

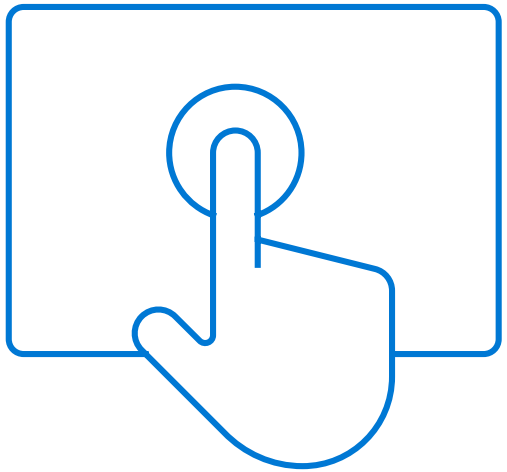
Built-in
auto scaling

Global
data center

Elastically burst
using ACI

Geo-replicated
container registry

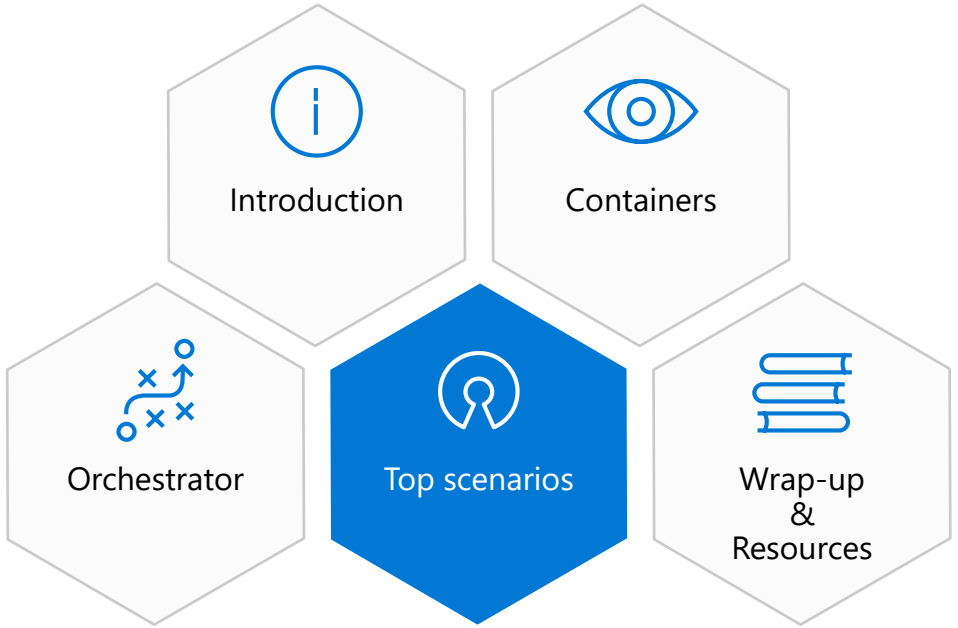




Demo

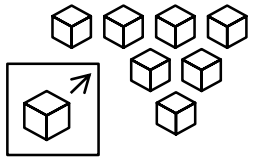
Get started easily with AKS through the Azure portal and the Azure CLI

Open source culture



Top scenarios for Containers and Kubernetes

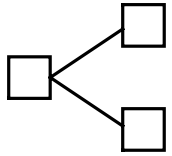
Lift and shift to containers



Cost saving

without refactoring your app

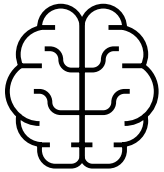
Microservices



Agility

Faster application development

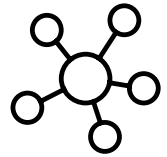
Machine learning



Performance

Low latency processing

IoT

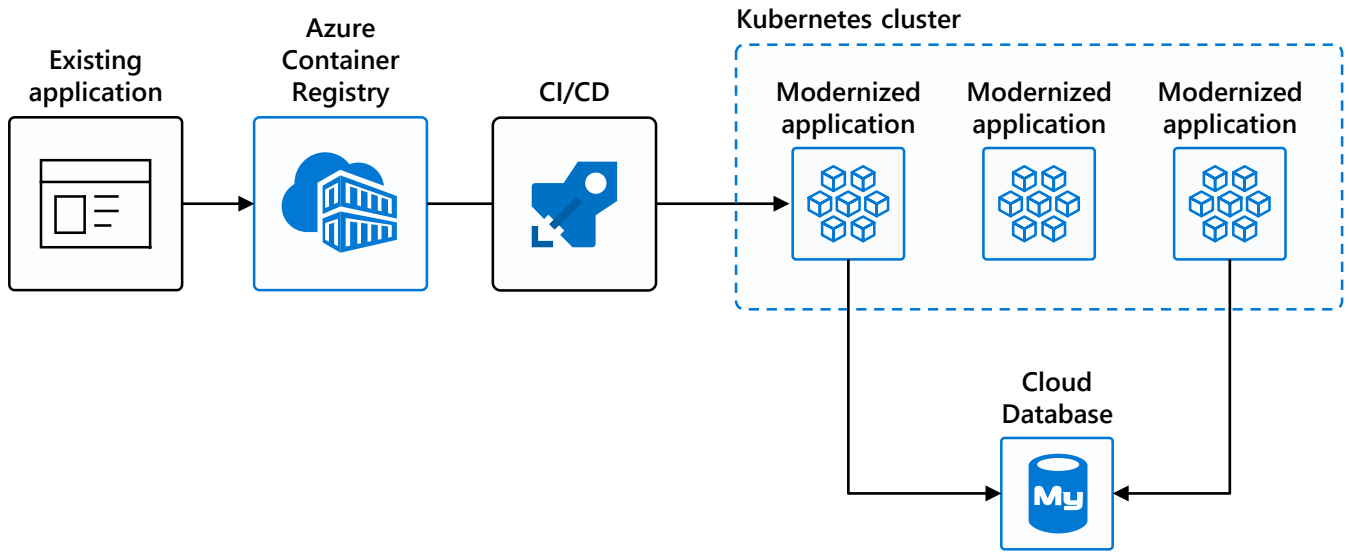


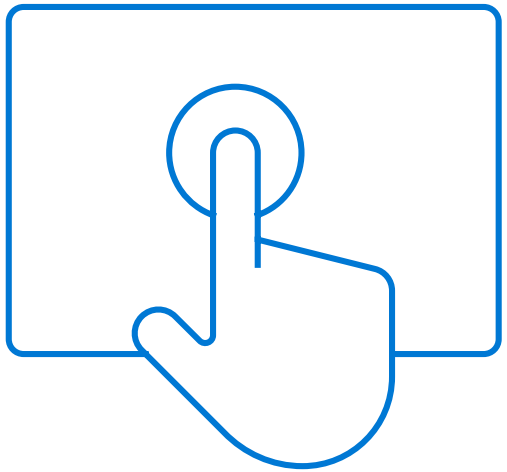
Portability

Build once, run anywhere

App modernization without code changes

- Speed application deployments by using container technology
- Defend against infrastructure failures with container orchestration
- Increase agility with continuous integration and continuous delivery





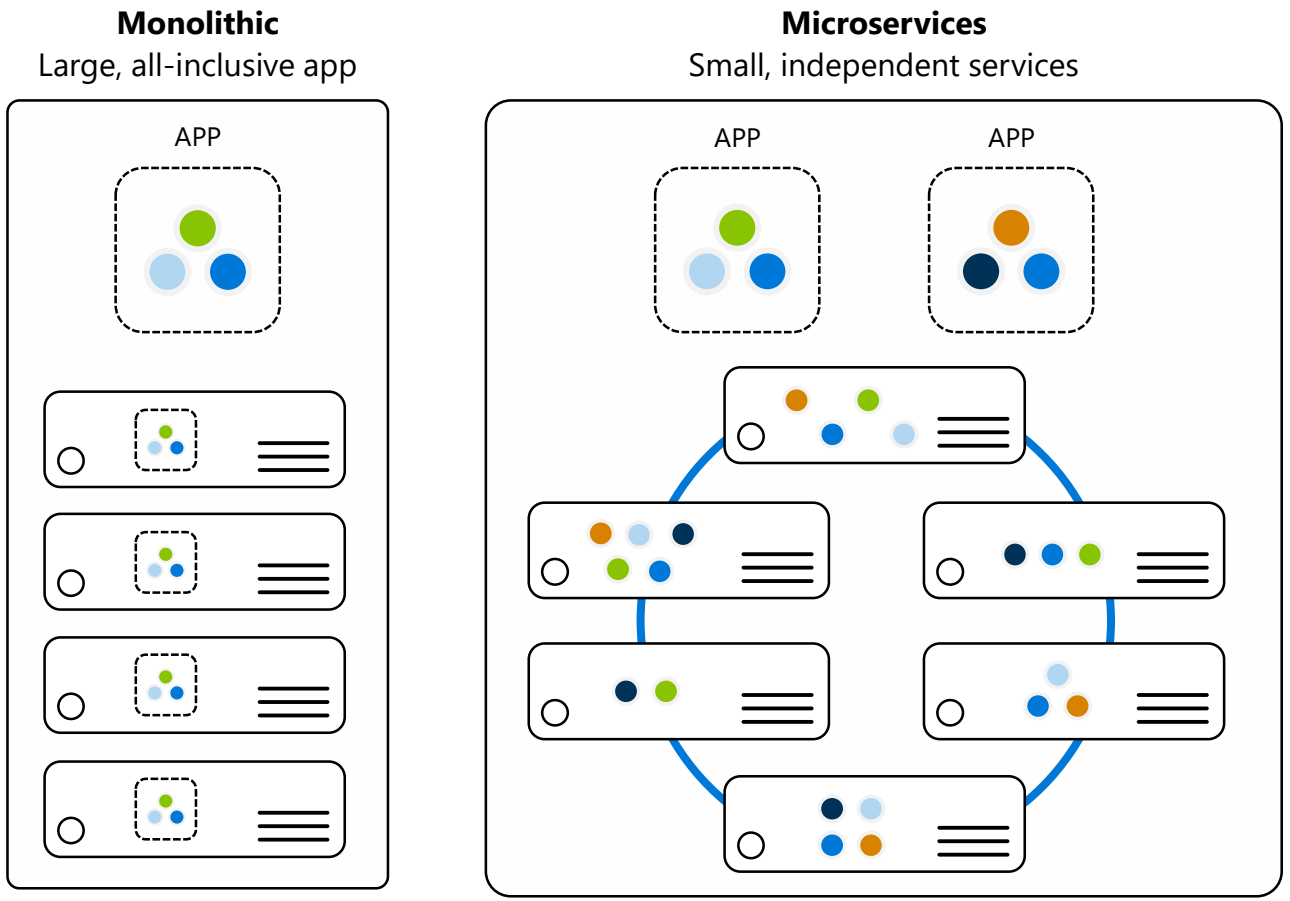
Demo

Set up CI/CD in simple steps with Azure DevOps Projects

Get started with Azure Container Registry (ACR) and Helm

Microservices: for faster app development

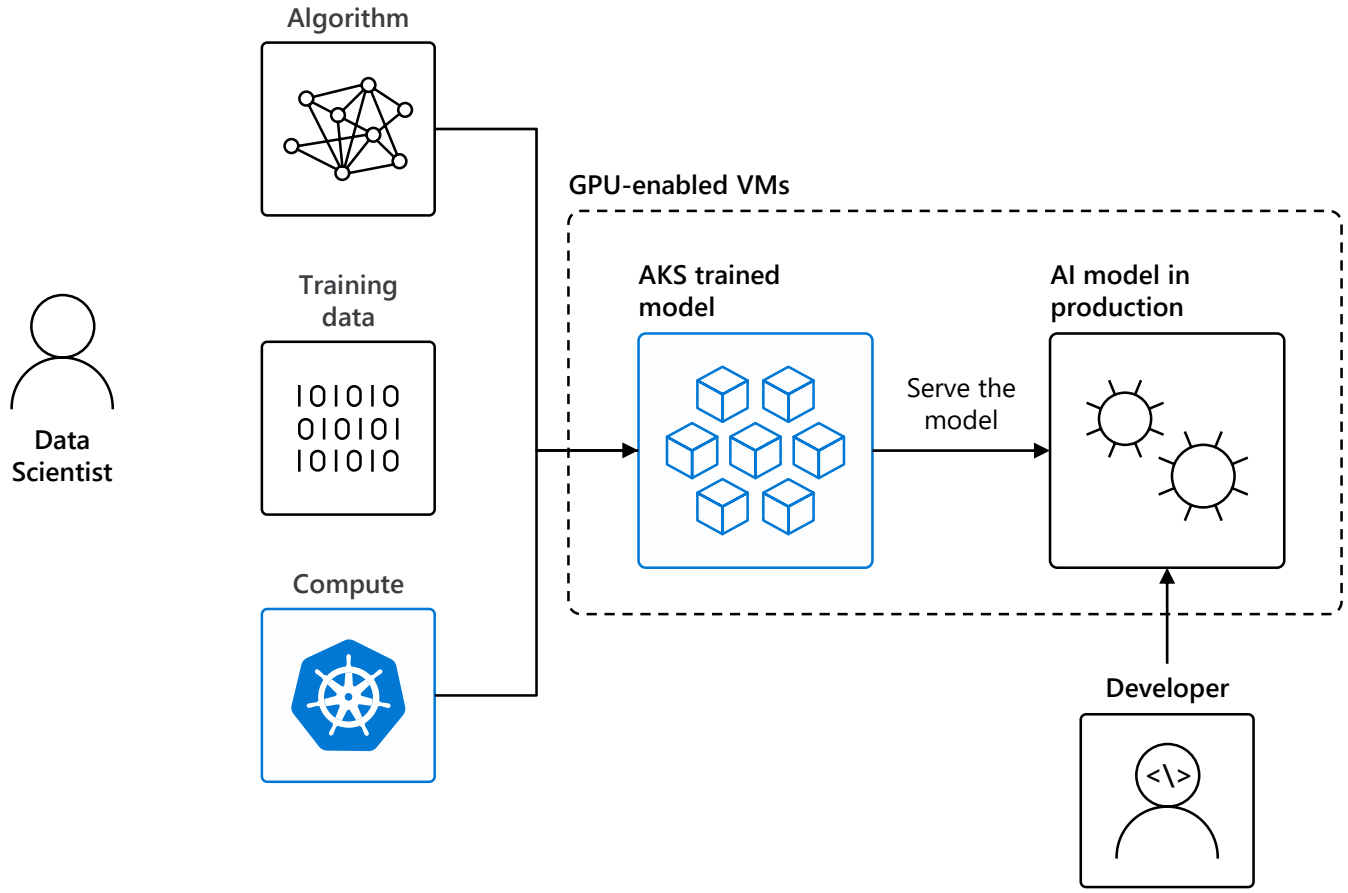
- Independent deployments
- Improved scale and resource utilization per service
- Smaller, focused teams



Data science in a box

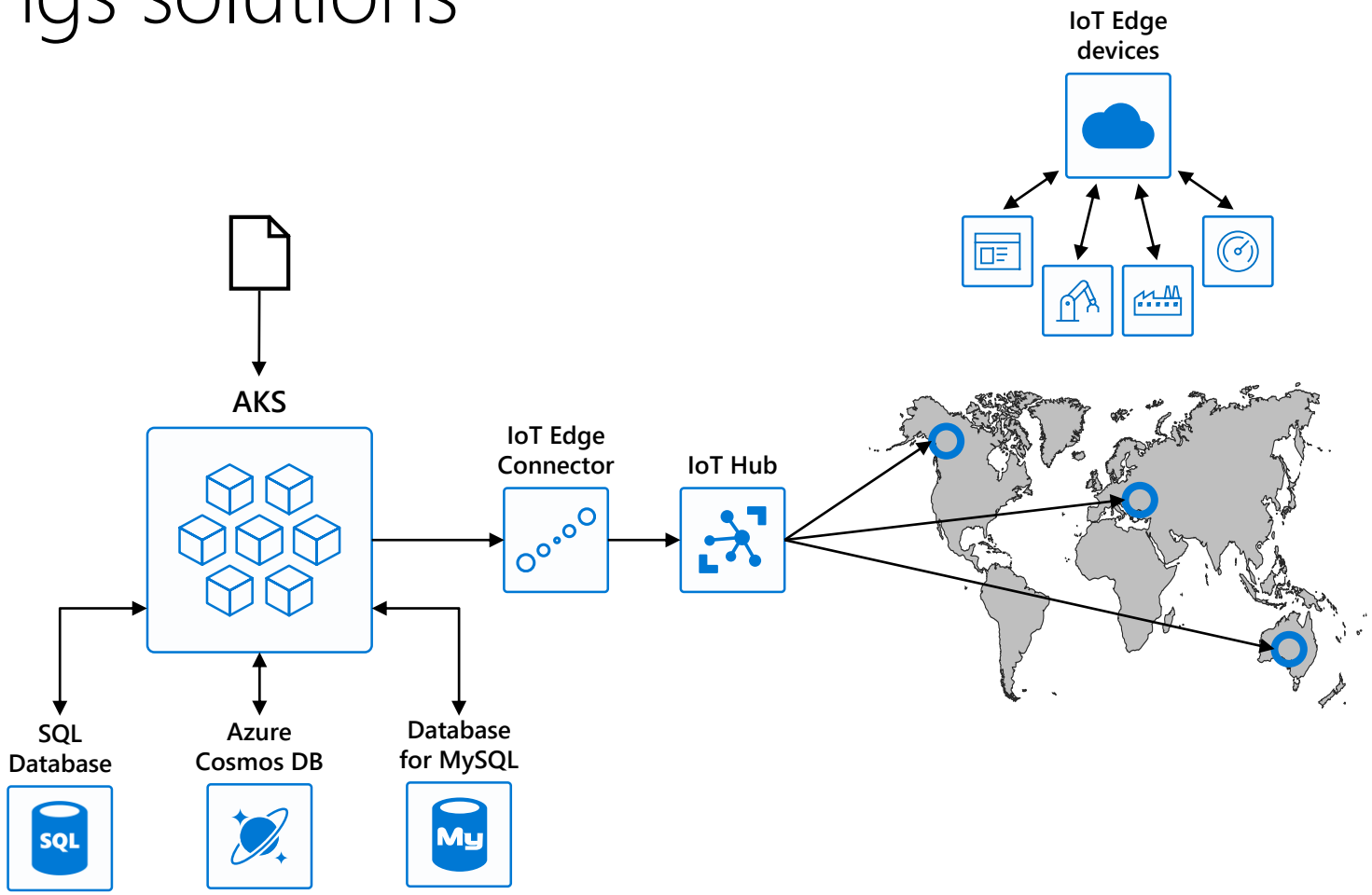
- Quick deployment and high availability
- Low latency data processing
- Consistent environment across test, control and production

https://github.com/Azure/kubeflow-labs

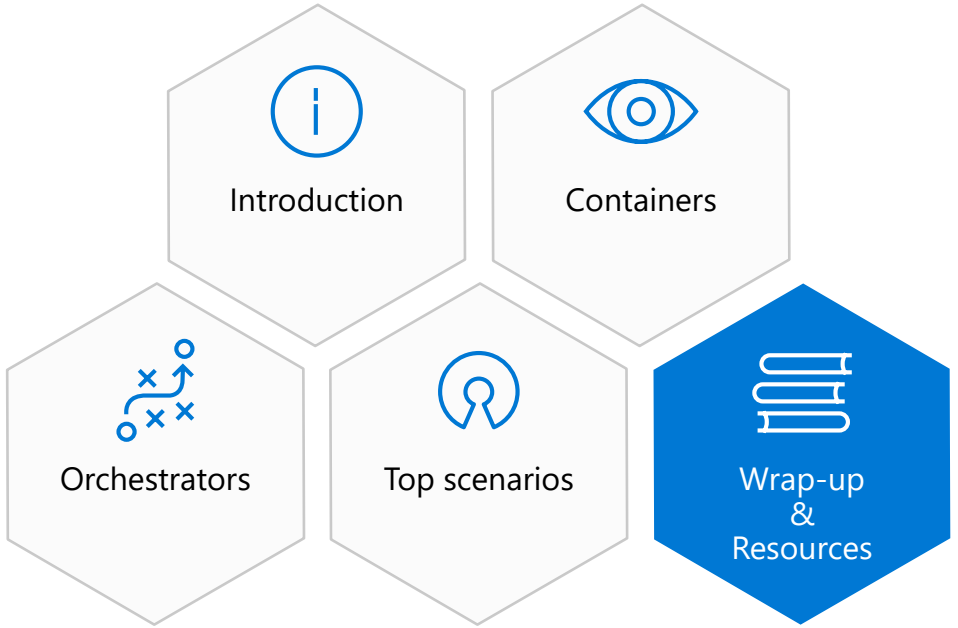


Scalable Internet of Things solutions

- Portable code, runs anywhere
- Elastic scalability and manageability
- Quick deployment and high availability



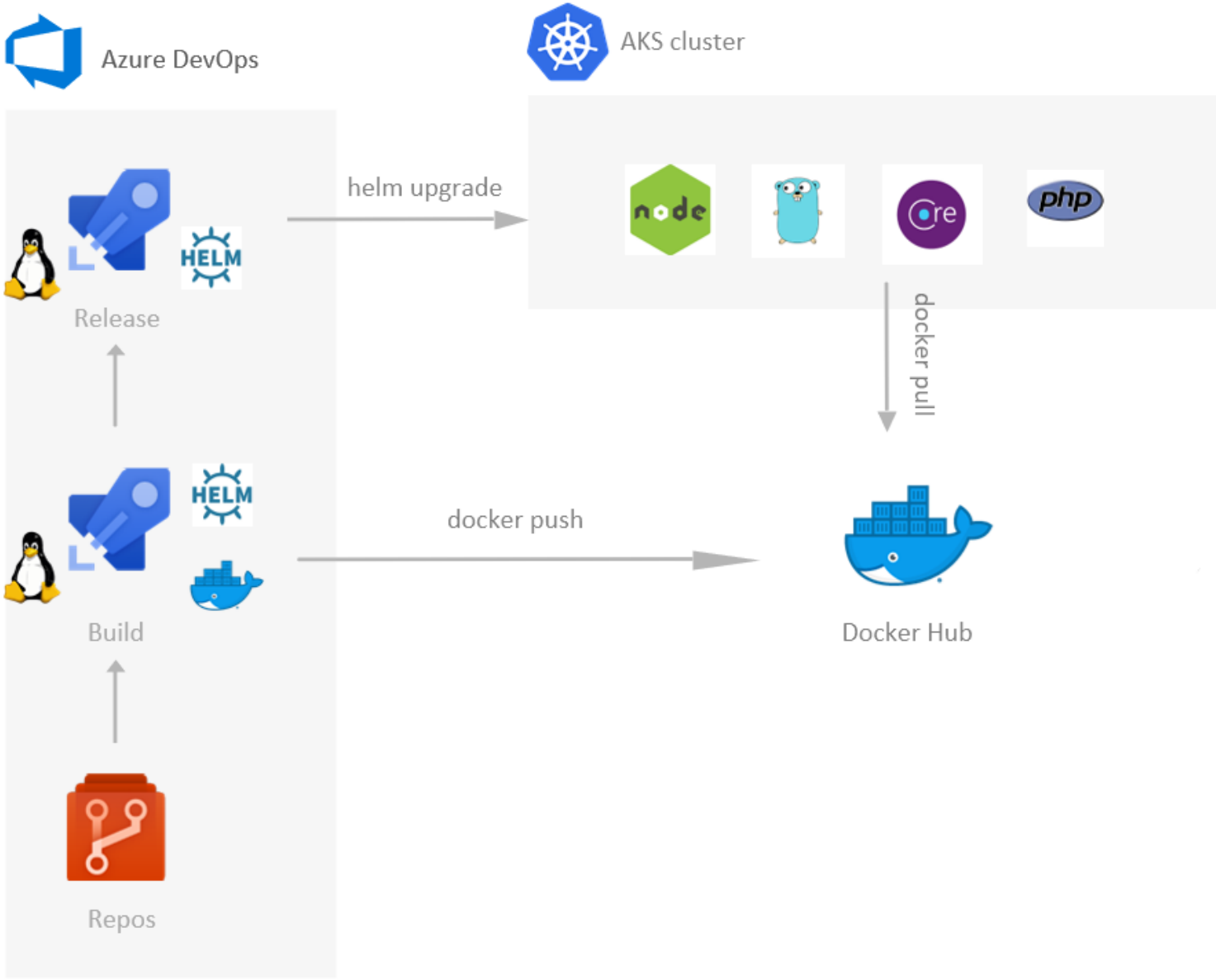
Resources


Introduction
Containers
Orchestrators
Top scenarios
Wrap-up
&
Resources

Objectives

- Docker, improves packaging any apps in any languages
- CaaS, runs quickly any Containers by abstracting the infra
- Kubernetes, simplifies the orchestration of your Containers
- Helm, unifies your Kubernetes deployments
- CI/CD pipeline, delivers rapidly your code to production
- Cloud, accelerates your modernization journey
- Open Source, is everywhere and is leading innovation

The generic Containers Workflow

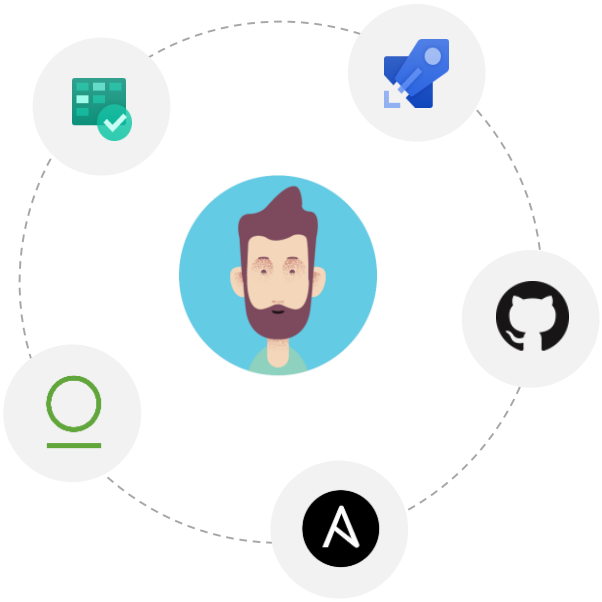


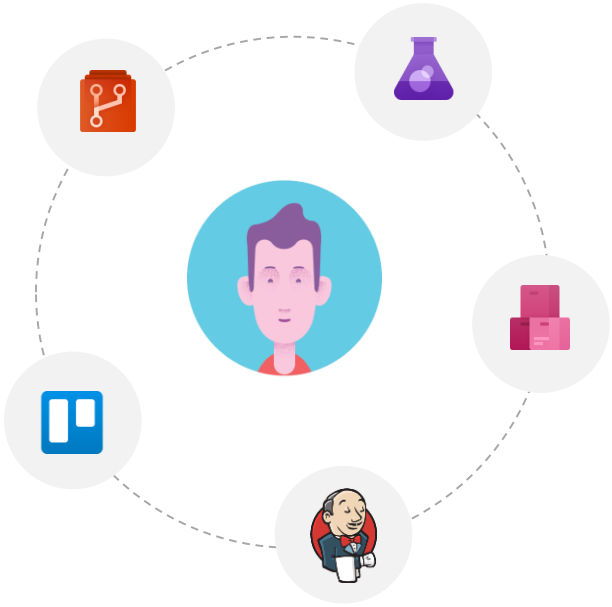
AKS: Work how you want with opensource tools

	Development	DevOps	Monitoring	Networking	Storage	Security
Take advantage of services and tools in the Kubernetes ecosystem	HELM DRAFT	Jenkins Terraform BRIGADE JFrog CODESHIP HASHICORP	Prometheus fluentd Grafana OPENTRACING DATADOG JAEGER	CNI Networking TIGERA	MAPR portworx	Twistlock aqua heptio RBAC
…or… Leverage growing Azure support	VS Code	Azure DevOps ARM	Azure Monitor	Azure VNET	Azure Storage	Azure Container Registry AAD Key Vault

Azure DevOps: Choose the tools and clouds you love

Azure DevOps lets developers choose the tools that are right for them

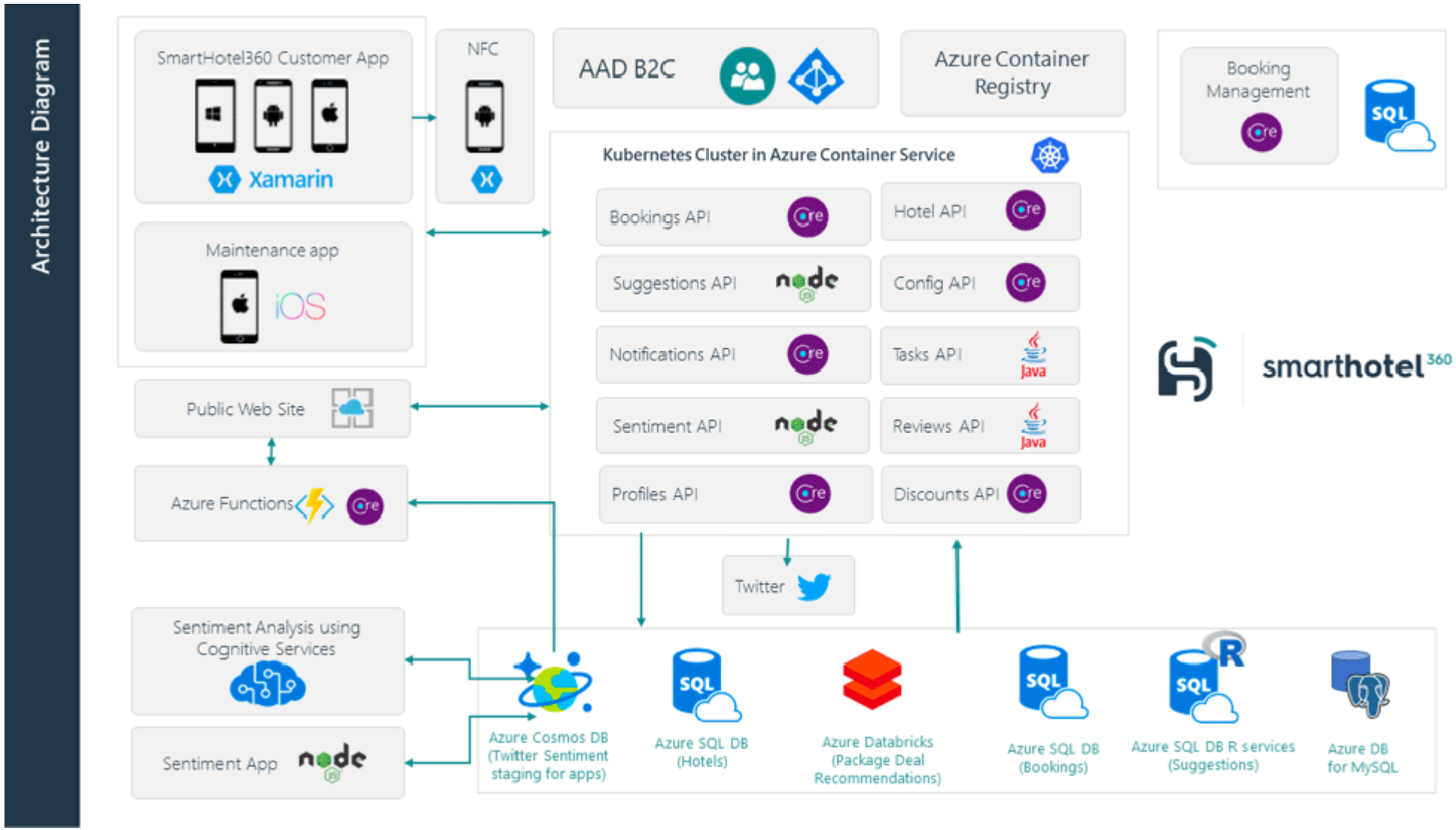




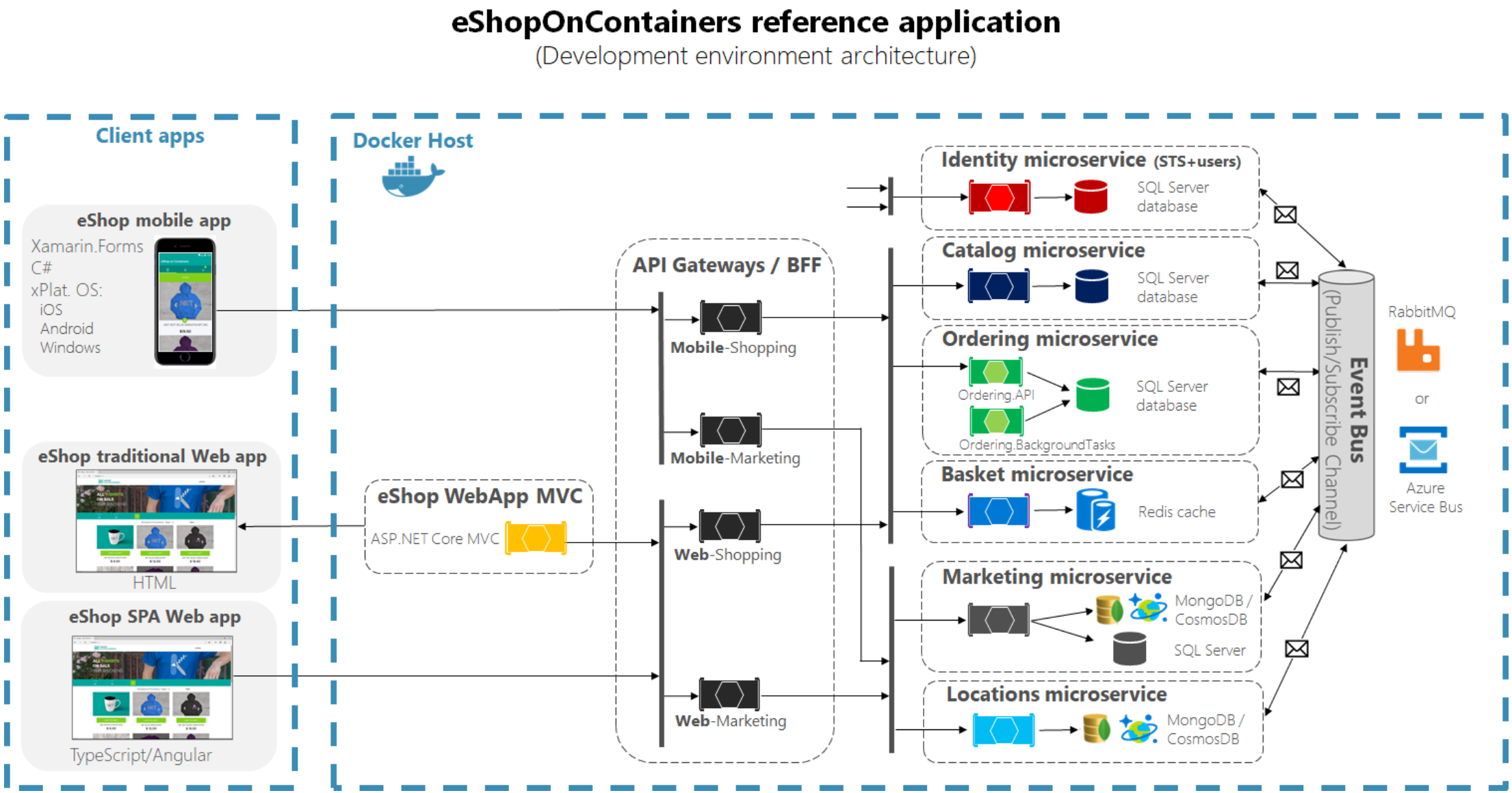
Mix and match to create workflows with tools from Microsoft, open source or your favorite 3rd party tools

Target any cloud, on-prem or both and deploy to the servers you need

SmartHotel 360



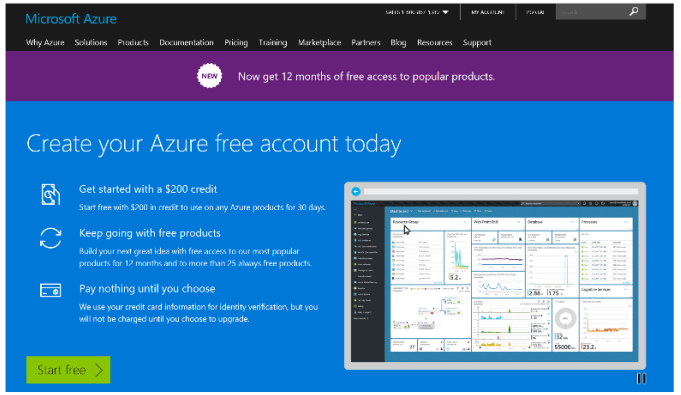
eShopOnContainers



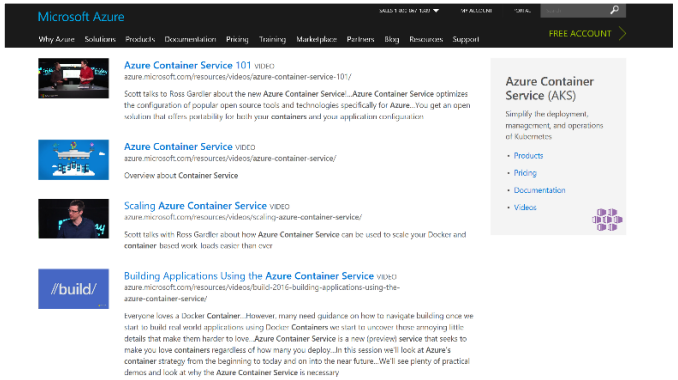
AKS resources

- Azure Kubernetes Service (AKS)
- Documentation resources
- Ebook for distributed systems
- Distributed system HoL
- AKS HoL

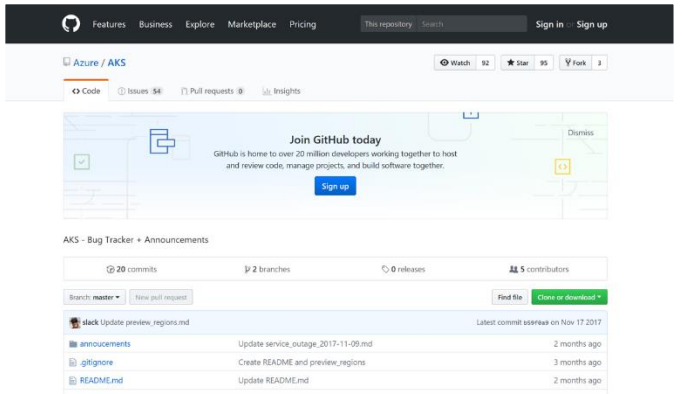
Sign up for a free Azure account



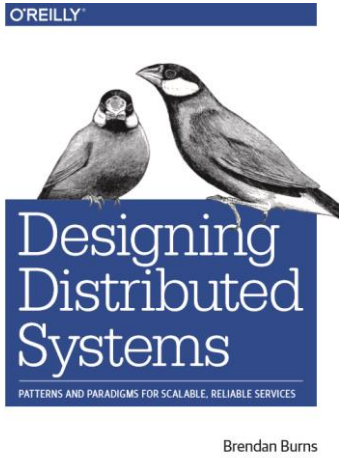
Check out the Azure container videos page

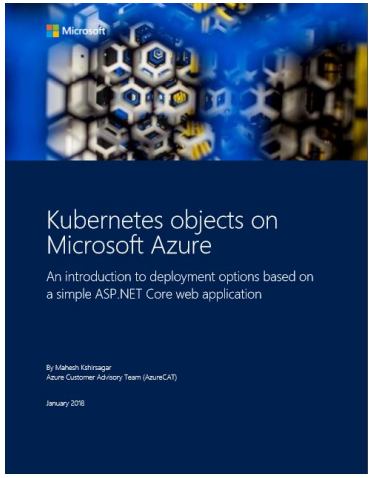


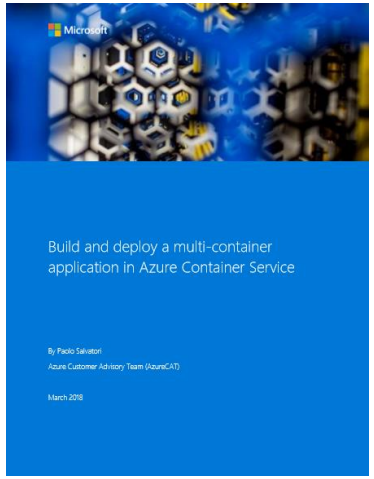
Get the code from GitHub

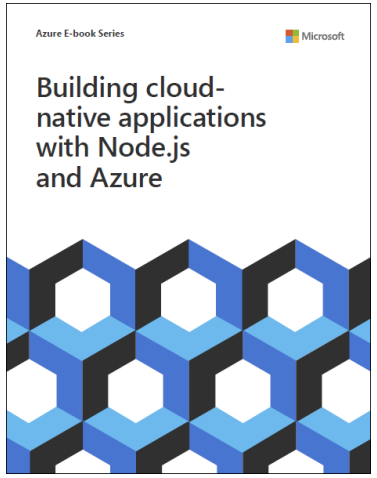


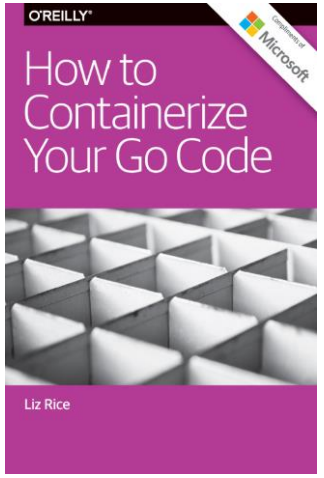
Download **free ebooks**











Containers at Microsoft Ignite 2018

Code	Session title
BRK2196	Architect your app modernization journey with containers on Microsoft Azure
BRK2030	Azure Service Fabric Overview and the Road Ahead
BRK2045	Fundamentals of Windows containers and Windows container-based web apps on Azure App Service
BRK2396	Fundamentals of Kubernetes on Microsoft Azure
BRK3344	Keep your PaaS and serverless apps healthy and happy
BRK2381	Azure Service Fabric Mesh: The Serverless Microservices Platform
BRK3208	Operational best practices for Azure Kubernetes Service
BRK3194	Deploying containerized and serverless apps using Terraform with Kubernetes (AKS) and Azure Functions
BRK2236	Take the next step with Windows Server container orchestration
BRK3190	Building resilient microservices with .NET Core and Azure Kubernetes Service (AKS)
BRK 3345	Best practices for Azure Service Fabric applications and clusters
BRK2392	Master tooling for containers
BRK2390	Bring your container or code to easily deploy to App Service on Linux

Merci ! Thank you! Any Questions?