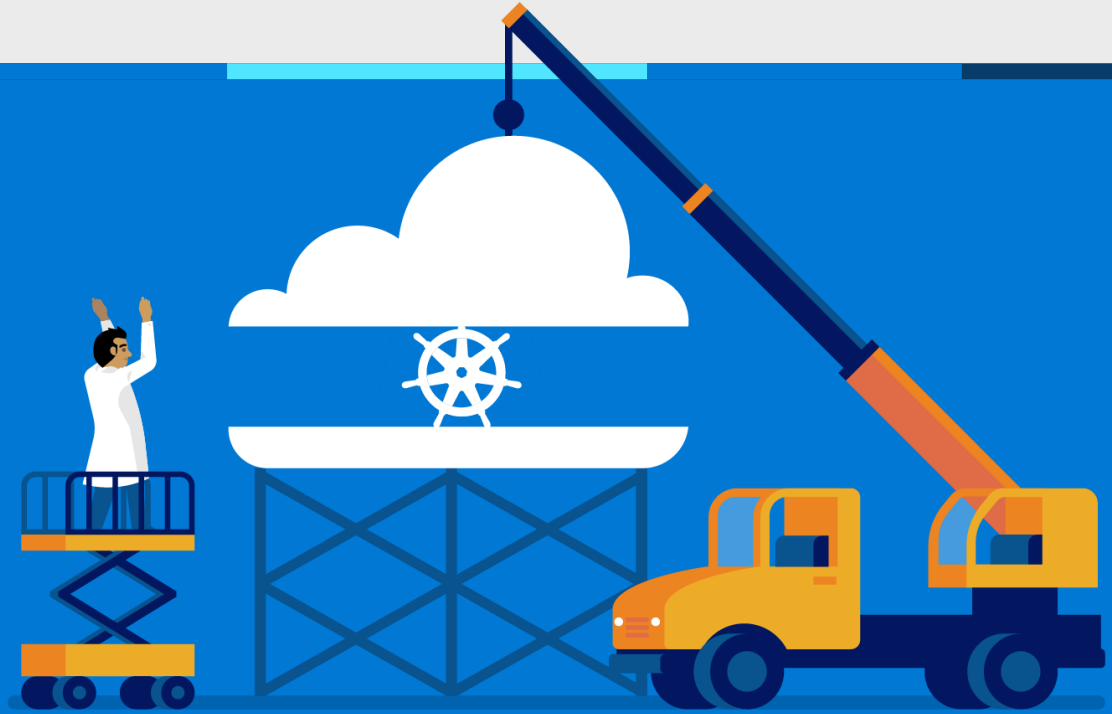
Microsoft Azure

# Kubernetes on Azure

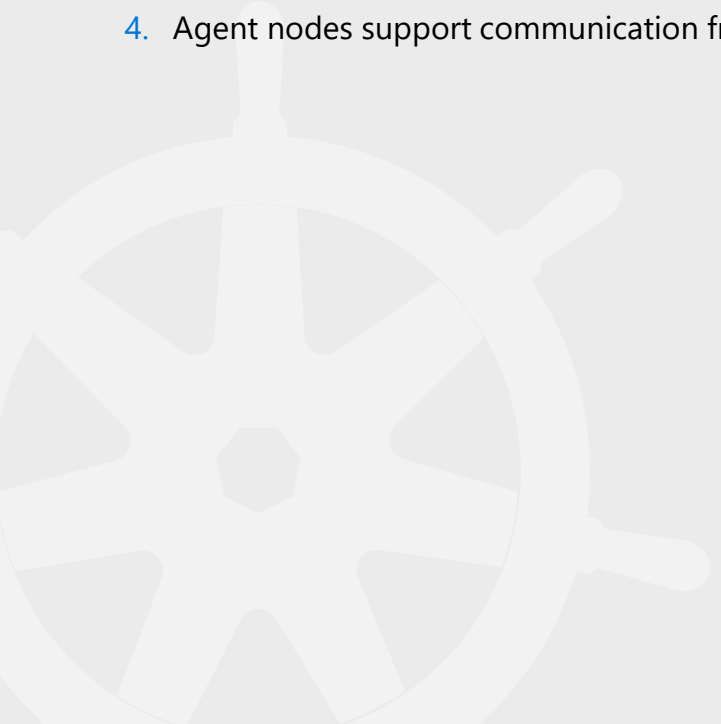
Learn about Kubernetes benefits, challenges, and enhancements made possible from a managed platform. Get the most out of Azure Kubernetes Service (AKS) with top scenarios, Azure capabilities, and tools.

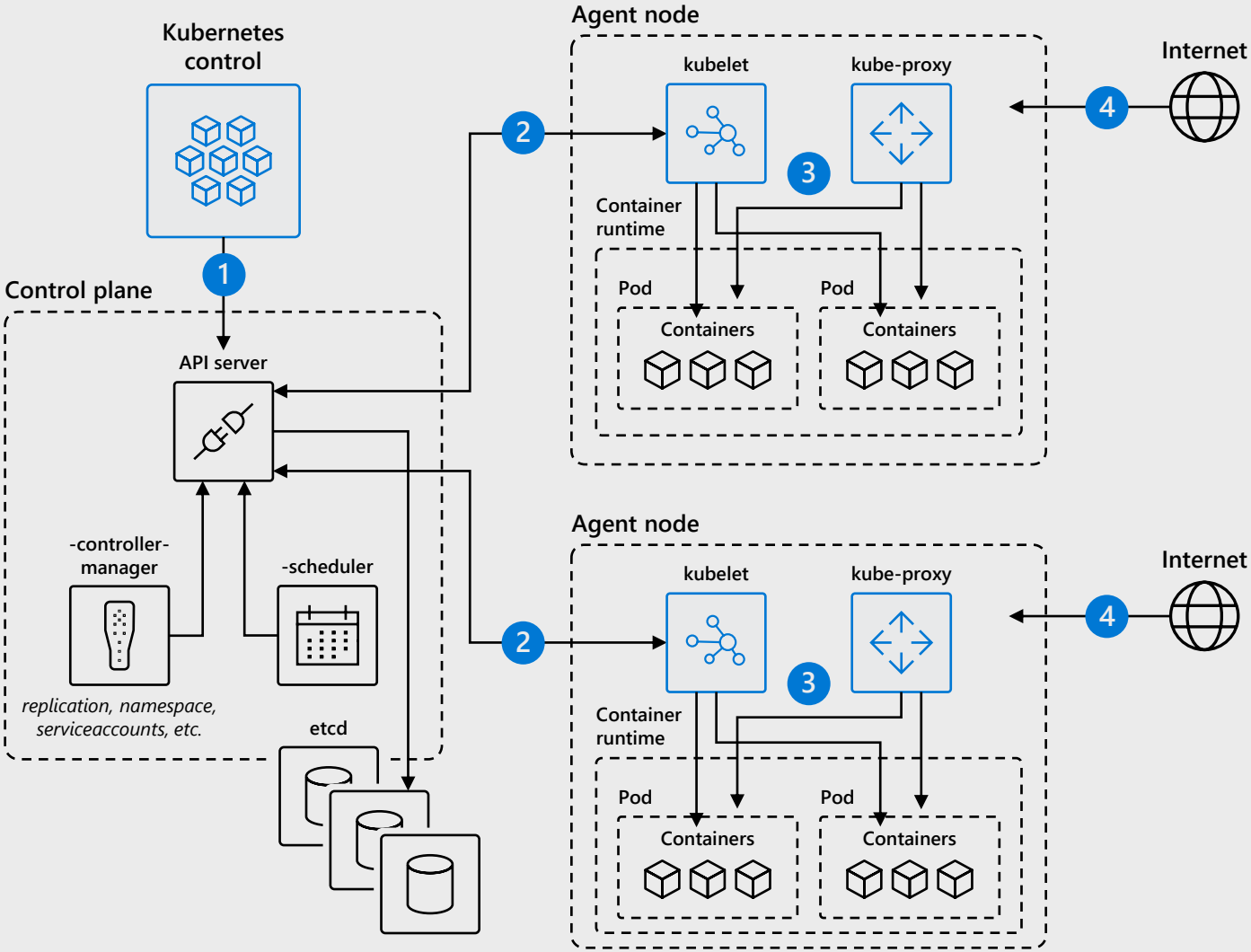


Published
October 2021

# How Kubernetes works

1. Kubernetes users communicate with API server and apply desired state
2. Control plane actively enforce desired state on agent nodes
3. Agent nodes support communication between containers
4. Agent nodes support communication from the Internet

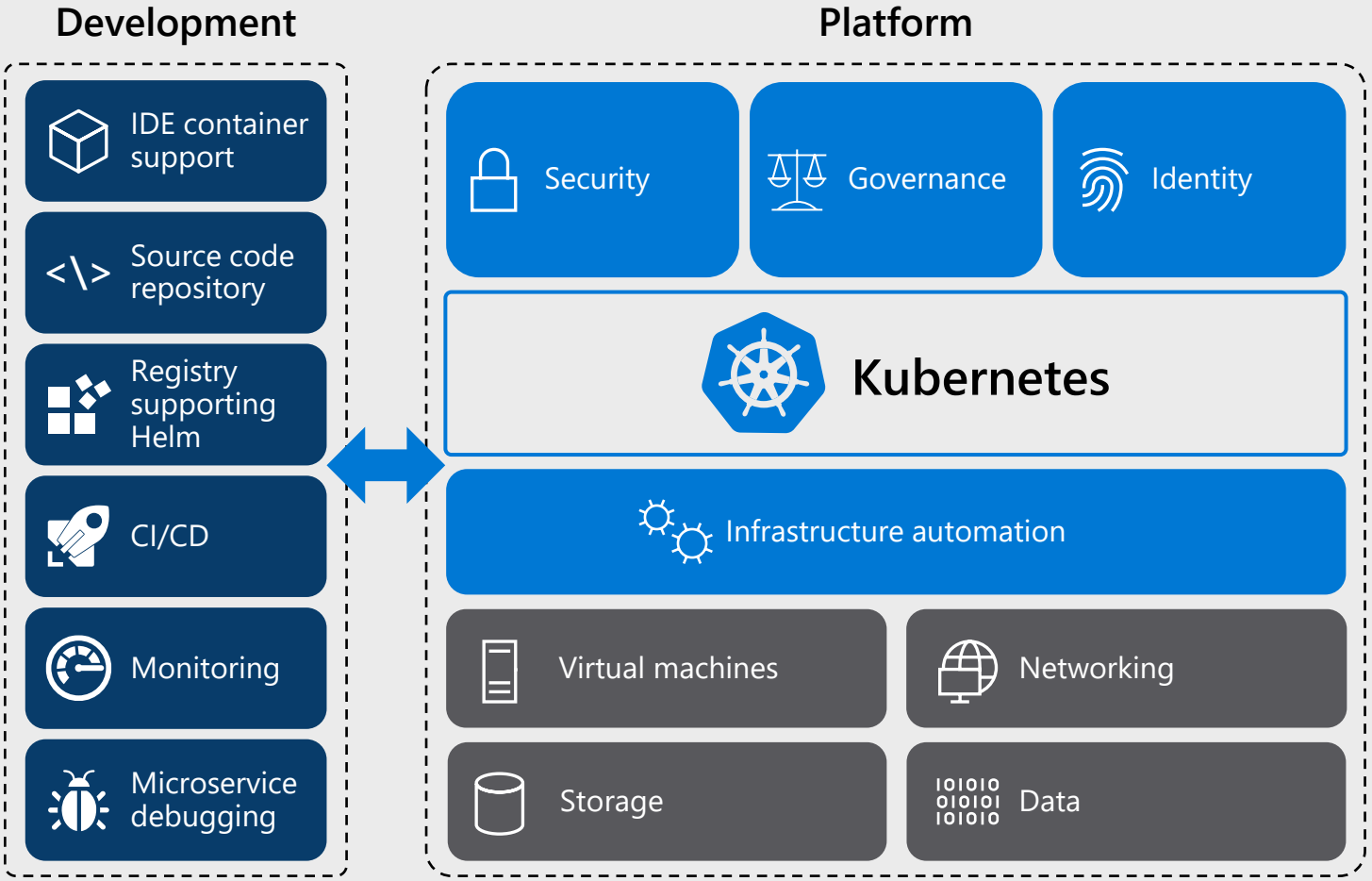



Kubernetes control
Control plane
1
API server
-controller-manager
-scheduler
replication, namespace, serviceaccounts, etc.
etcd
2
Agent node
kubelet
kube-proxy
3
Container runtime
Pod
Containers
Pod
Containers
4
Internet
Agent node
kubelet
kube-proxy
3
Container runtime
Pod
Containers
Pod
Containers
4
Internet

# But 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**

Learn more at
**aka.ms/k8slearning**


Development
IDE container support
Source code repository
Registry supporting Helm
CI/CD
Monitoring
Microservice debugging
Platform
Security
Governance
Identity
Kubernetes
Infrastructure automation
Virtual machines
Networking
Storage
Data

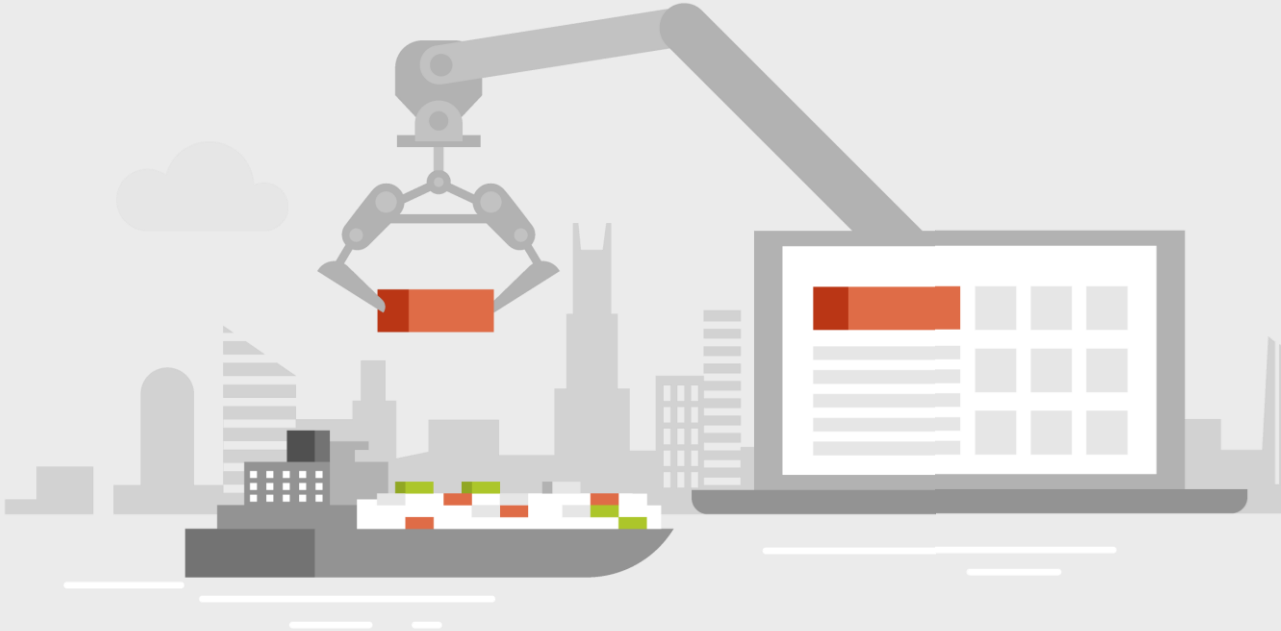
# Azure Kubernetes Service (AKS) momentum



## The fastest growing compute service on Azure

# Trusted by thousands of customers

# Infrastructure automation



Kubernetes gives you the knobs to schedule and deploy containers across clusters, scale to your desired state, and manage the Kubernetes lifecycle to keep your apps up and running.

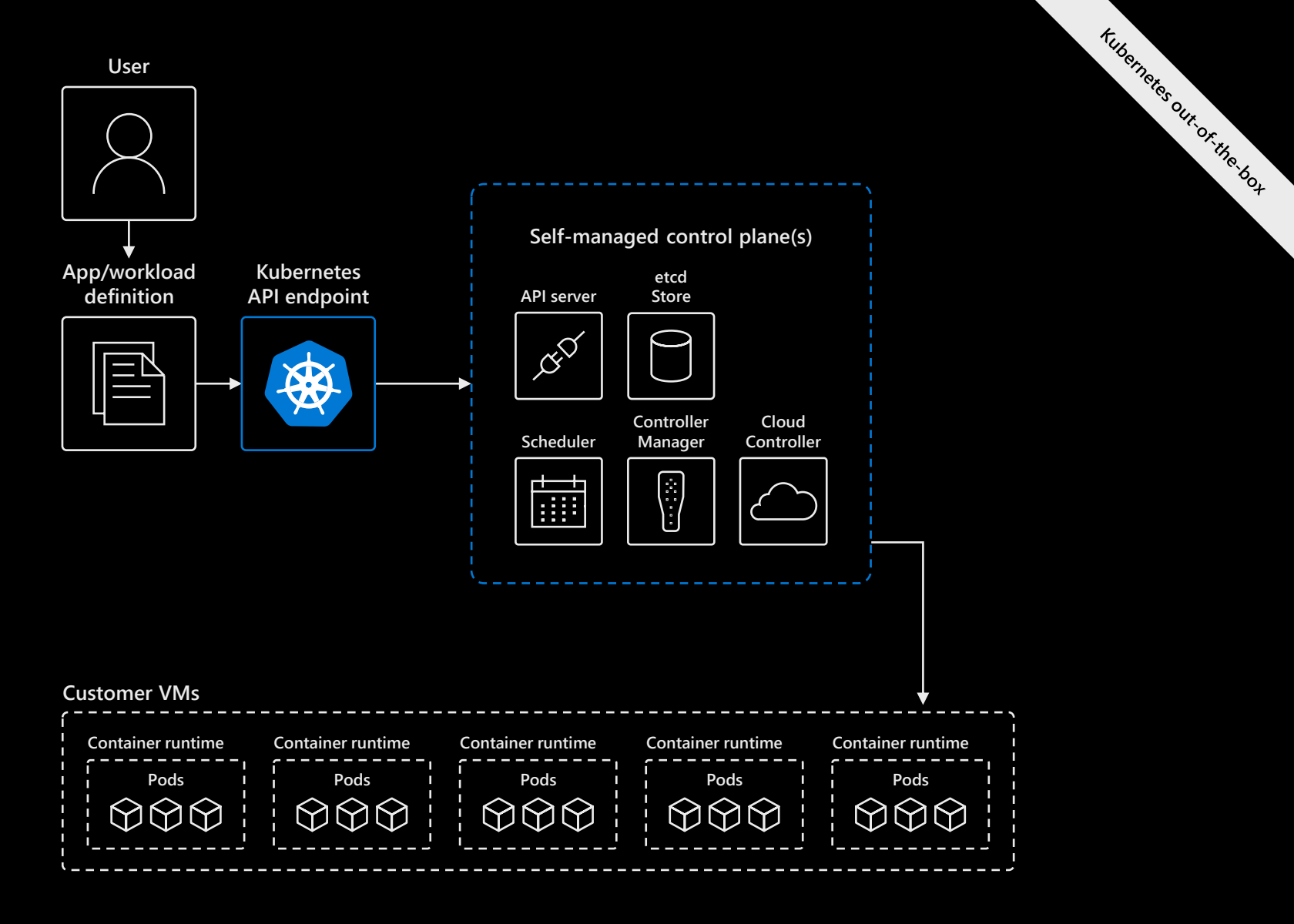
As your applications move to production, they often span multiple containers, deployed across a cluster of servers—increasing the complexity of operating the knobs and taking up time you could be spending delivering value to your customers.

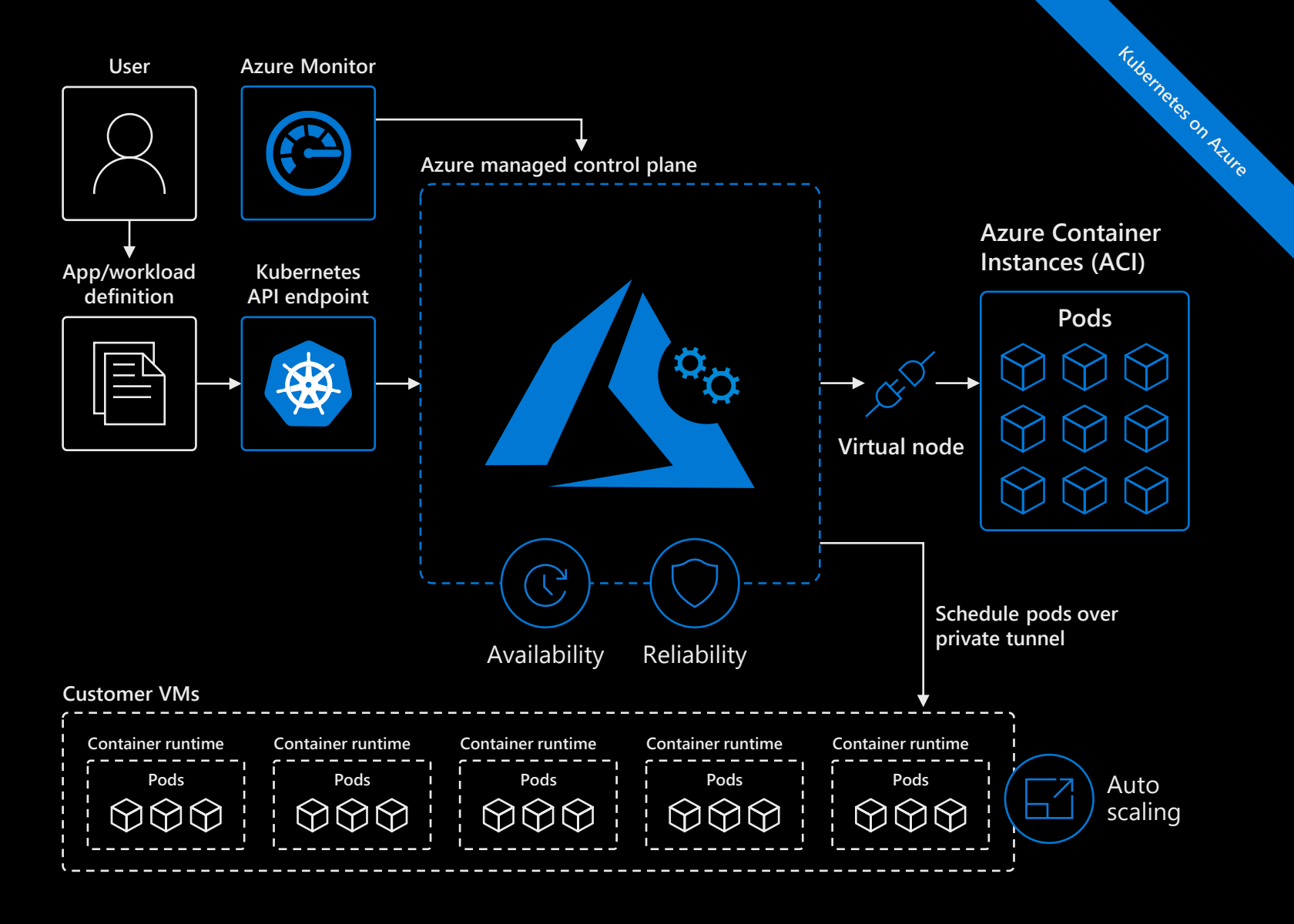
A fully managed Kubernetes service, like Azure Kubernetes Service (AKS), **automates provisioning, upgrading, monitoring, and scaling for compute resources**.

Kubernetes on Azure

# Manage Kubernetes with ease

- Automated provisioning, upgrades, and patches
- High reliability and availability
- Serverless scaling
- API server monitoring
- Delivered at no charge


Kubernetes out-of-the-box
User
App/workload definition
Kubernetes API endpoint
Self-managed control plane(s)
API server
etcd Store
Scheduler
Controller Manager
Cloud Controller
Customer VMs
Container runtime
Pods
Container runtime
Pods
Container runtime
Pods
Container runtime
Pods
Container runtime
Pods


Kubernetes on Azure
User
Azure Monitor
Azure managed control plane
App/workload definition
Kubernetes API endpoint
Azure Container Instances (ACI)
Pods
Virtual node
Availability
Reliability
Schedule pods over private tunnel
Customer VMs
Container runtime
Pods
Container runtime
Pods
Container runtime
Pods
Container runtime
Pods
Container runtime
Pods
Auto scaling

“Thanks to AKS, we can now spin up new demo environments in 10 minutes instead of 24 hours.  Moving Docushare Flex from virtual machines to containers in Azure allows us to provision environments faster, empowering our sales and partner network.”

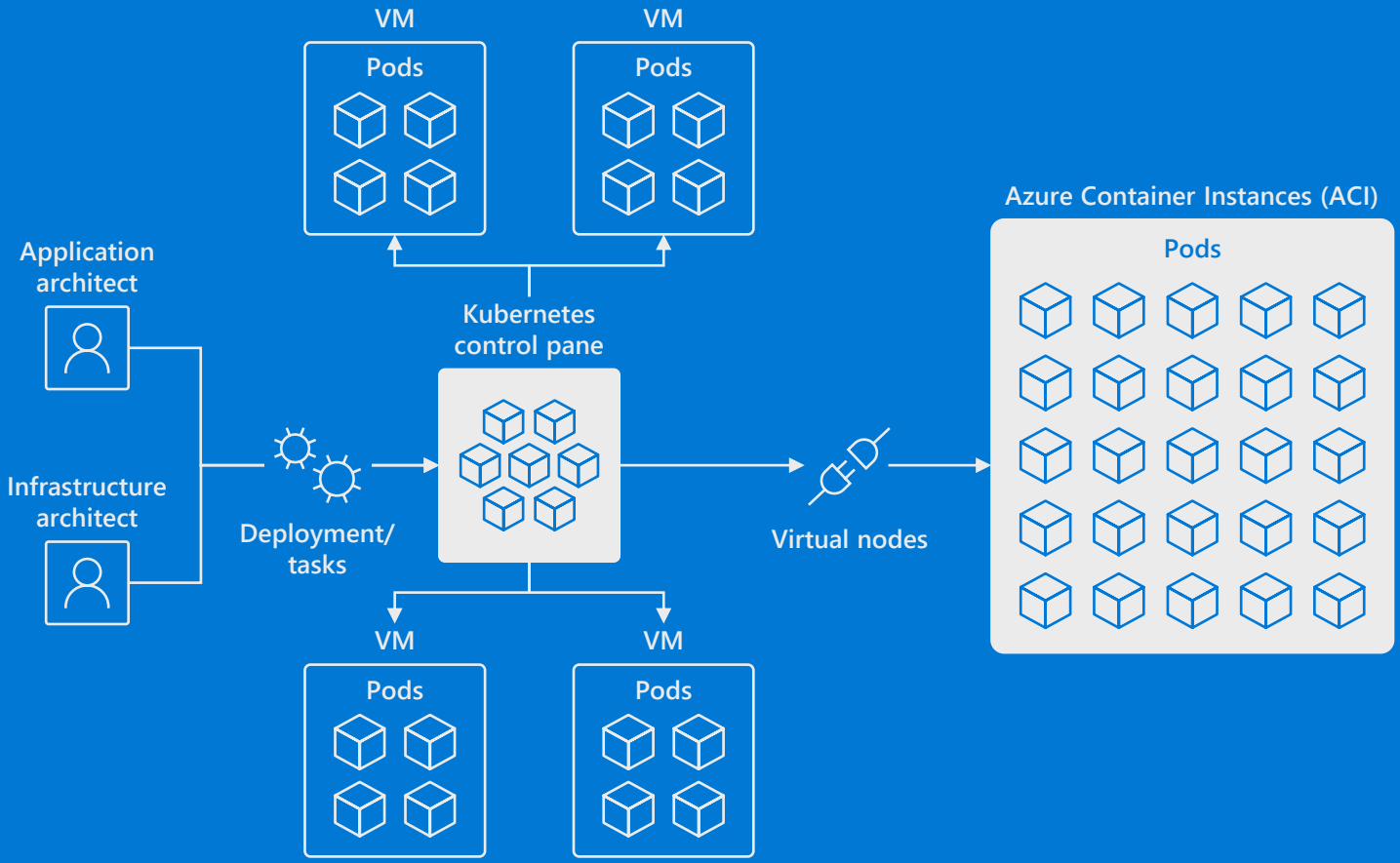
— Robert Bingham, Director of DocuShare Cloud Operation
Xerox

# Virtual nodes

- Elastically provision capacity in seconds
- No infrastructure to manage
- Built on open-sourced Virtual Kubelet technology, a sandbox project from CNCF

Learn more at
aka.ms/aksbook/virtualnode

Capability


VM
Pods
VM
Pods
Azure Container Instances (ACI)
Pods
Application architect
Kubernetes control pane
Infrastructure architect
Deployment/ tasks
Virtual nodes
VM
Pods
VM
Pods

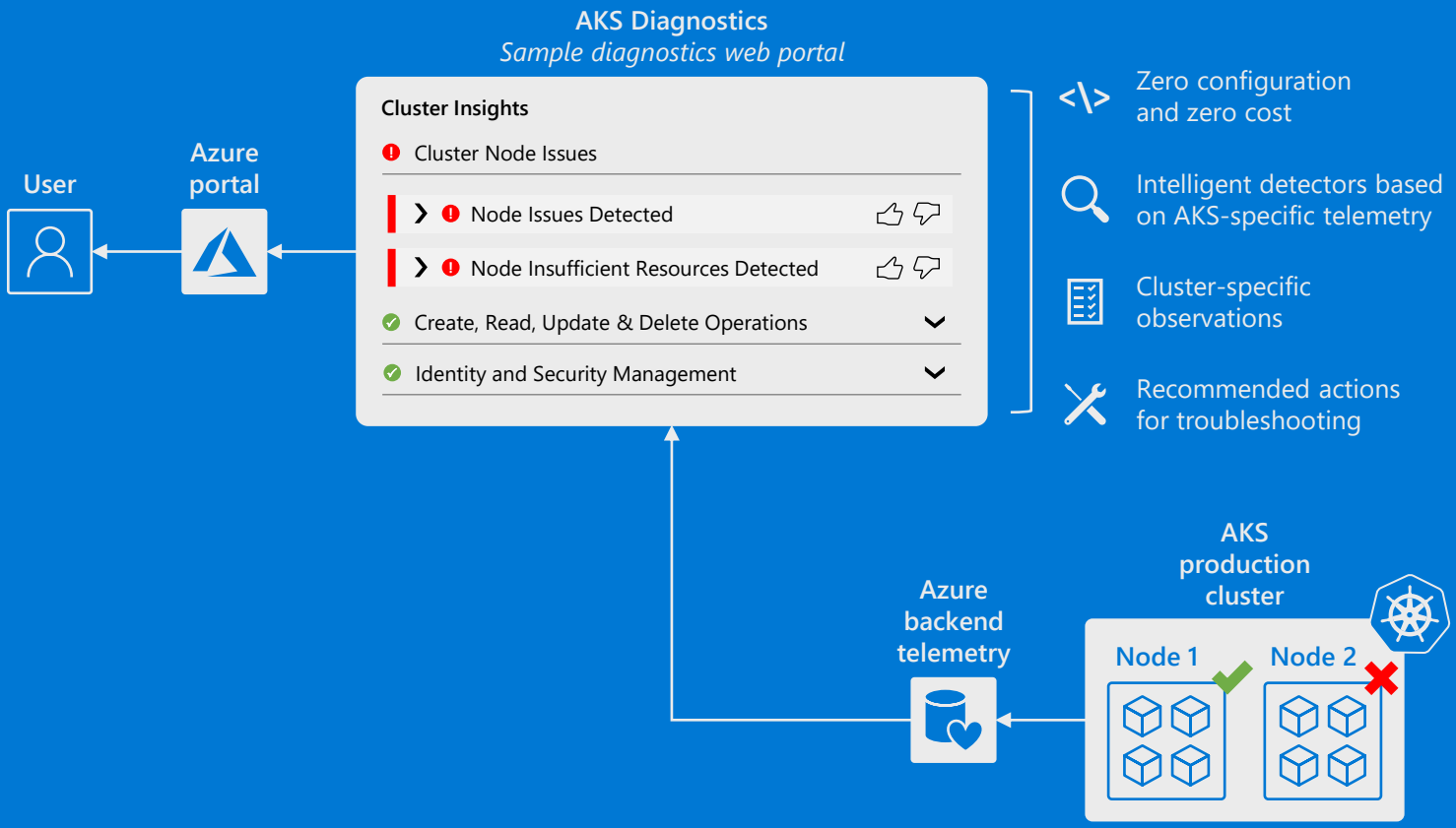
Capability

# Diagnostics

- An out-of-the-box guided and interactive experience that helps you diagnose and solve potential issues with your AKS cluster
- Quickly pinpoint cluster issues using the detectors in AKS diagnostics to analyze backend telemetry on node issues, CRUD operations, identity and security management, and more
- Get recommendations for next steps and potential solutions
- With red, orange, and green status icons to indicate whether there is a pertinent issue, it's simple to click through each detector and expand "insights" via Genie with observations and recommended actions

**To get started, navigate to your AKS cluster in the Azure Portal and select the "Diagnose and solve problems" tab**

Learn more at
**aka.ms/aksbook/aksdiagnostics**


AKS Diagnostics
Sample diagnostics web portal
Cluster Insights
Cluster Node Issues
Node Issues Detected
Node Insufficient Resources Detected
Create, Read, Update & Delete Operations
Identity and Security Management
User
Azure portal
Zero configuration and zero cost
Intelligent detectors based on AKS-specific telemetry
Cluster-specific observations
Recommended actions for troubleshooting
Azure backend telemetry
AKS production cluster
Node 1
Node 2

# End-to-end developer experience

Kubernetes API itself doesn't include development tools. To run an application in a Kubernetes cluster, a developer may use a code editor to write code and perhaps a source code control repository to manage it; a Docker client to help with containerization; Helm for packaging; and kubectl, or a YAML configuration to deploy containers to Kubernetes.

In a real-world scenario, the picture becomes much more complicated. As containers, environments, and the teams that work with them multiply, release frequency increases—along with developmental and operational complexity. For example, the need to merge code effectively with the option to rollback; testing the application in a way that mimics the production environment but doesn't impact the production environment; and, quickly identifying and addressing any issues without downtime. The last thing you want to have on top of this complexity is a fragmented tool chain.

A managed Kubernetes platform designed for developers can **integrate seamlessly with your favorite IDE, CI/CD, and monitoring tools and automate these workflows to support your Kubernetes app development**. An IDE that directly supports Kubernetes deployment can help you set up the most complex microservices development environment and connect with your private container registry. With built-in CI/CD and a pre-configured deployment strategy, you can accelerate the move from code to container to Kubernetes cluster in minutes by automating those tasks. Finally, a complete view from container health monitoring to centralized logging can be auto-configured with your developer portal to prevent resource bottlenecks, trace malicious requests, and keep your Kubernetes applications healthy.

Kubernetes on Azure

# Accelerate containerized development

## Develop

- Native containers and Kubernetes support in IDE
- Remote debugging and iteration for multi-containers
- Effective code merge
- Automatic containerization

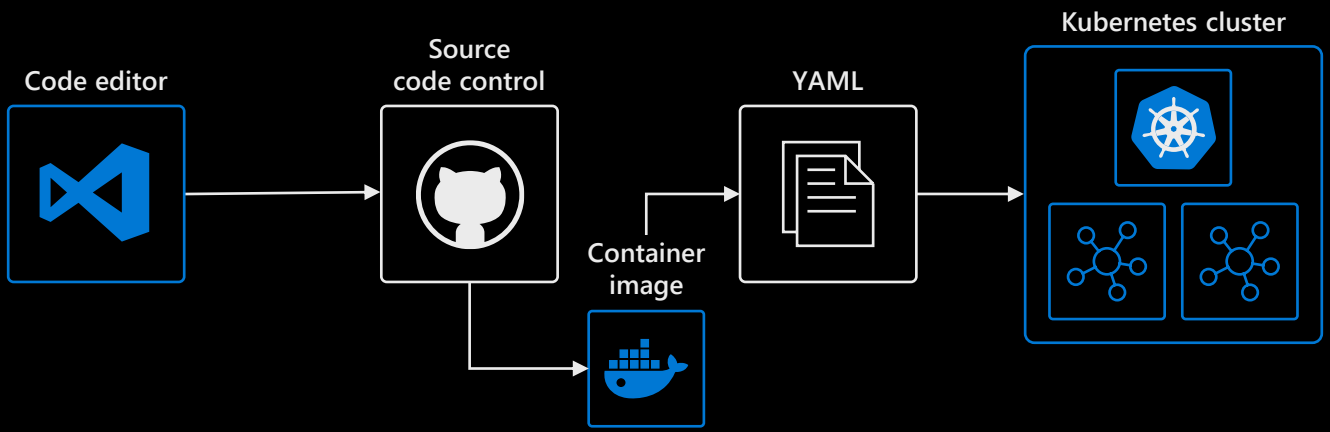
## Deliver

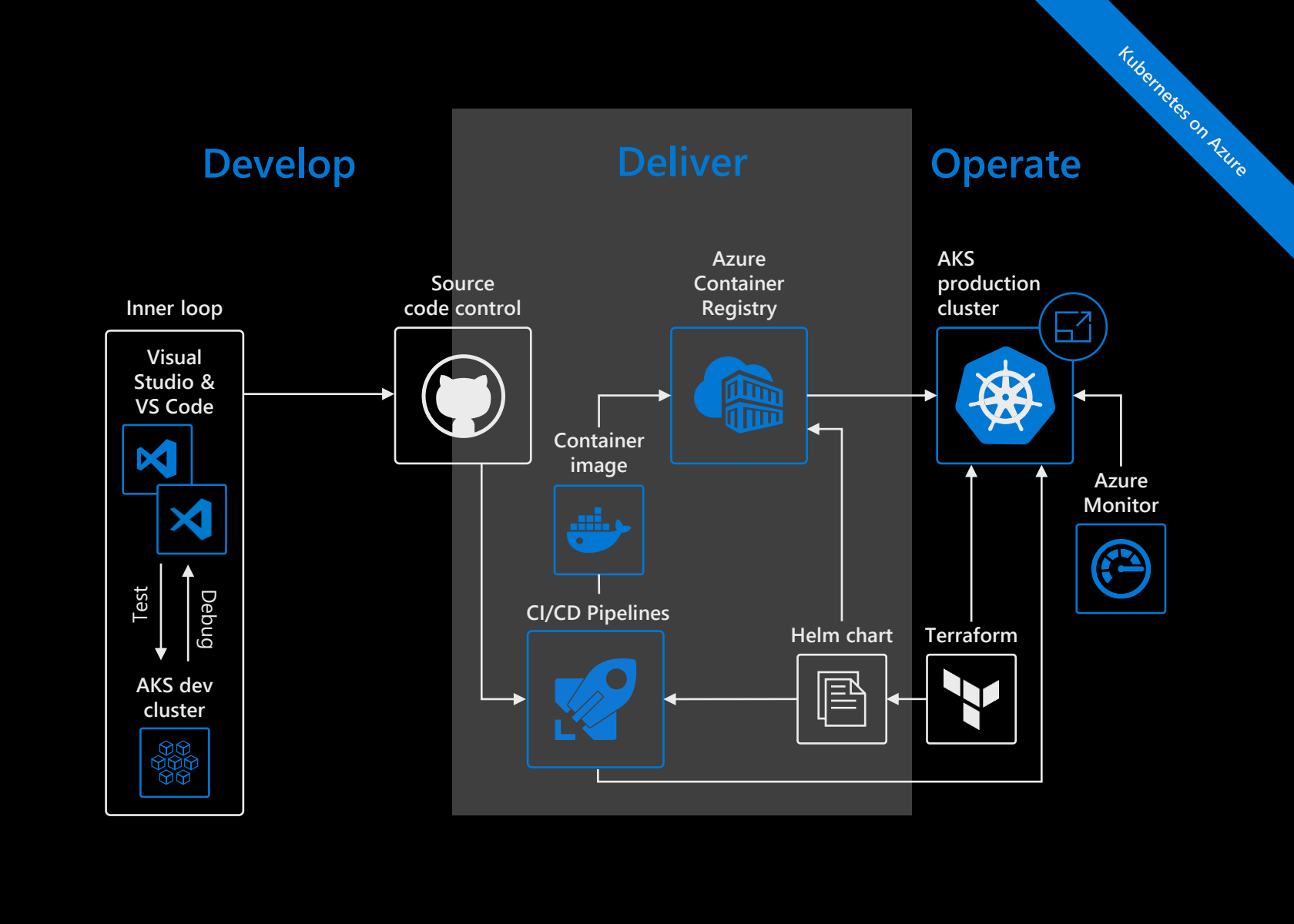
- CI/CD pipeline with automated tasks in a few clicks
- Pre-configured canary deployment strategy
- In-depth build and delivery process review and integration testing
- Private registry with both container image and Helm chart management

## Operate

- Out-of-box control plane telemetry, log aggregation, and container health
- Declarative resource management
- Auto-scaling

Kubernetes out-of-the-box




Kubernetes on Azure
Develop
Deliver
Operate
Inner loop
Visual Studio & VS Code
Test
Debug
AKS dev cluster
Source code control
Container image
CI/CD Pipelines
Azure Container Registry
Helm chart
Terraform
AKS production cluster
Azure Monitor

“We are building our own new applications using microservices—and AKS is our choice for orchestrating their workloads.”

— Ståle Heitmann, Chief Technology Officer
Hafslund Nett

Hafslund Nett

Capability

# Bridge to Kubernetes

Client-side tool integrated into Visual Studio and VS Code allows you to run and debug microservice code **on your development workstation** while connected to your Kubernetes cluster with the rest of your application or services.

### Simplify microservice development

- Eliminate the need to manually source, configure, and compile external dependencies

### Develop applications faster

- Sidestep operational complexities of building and deploying code into the cluster to test and debug

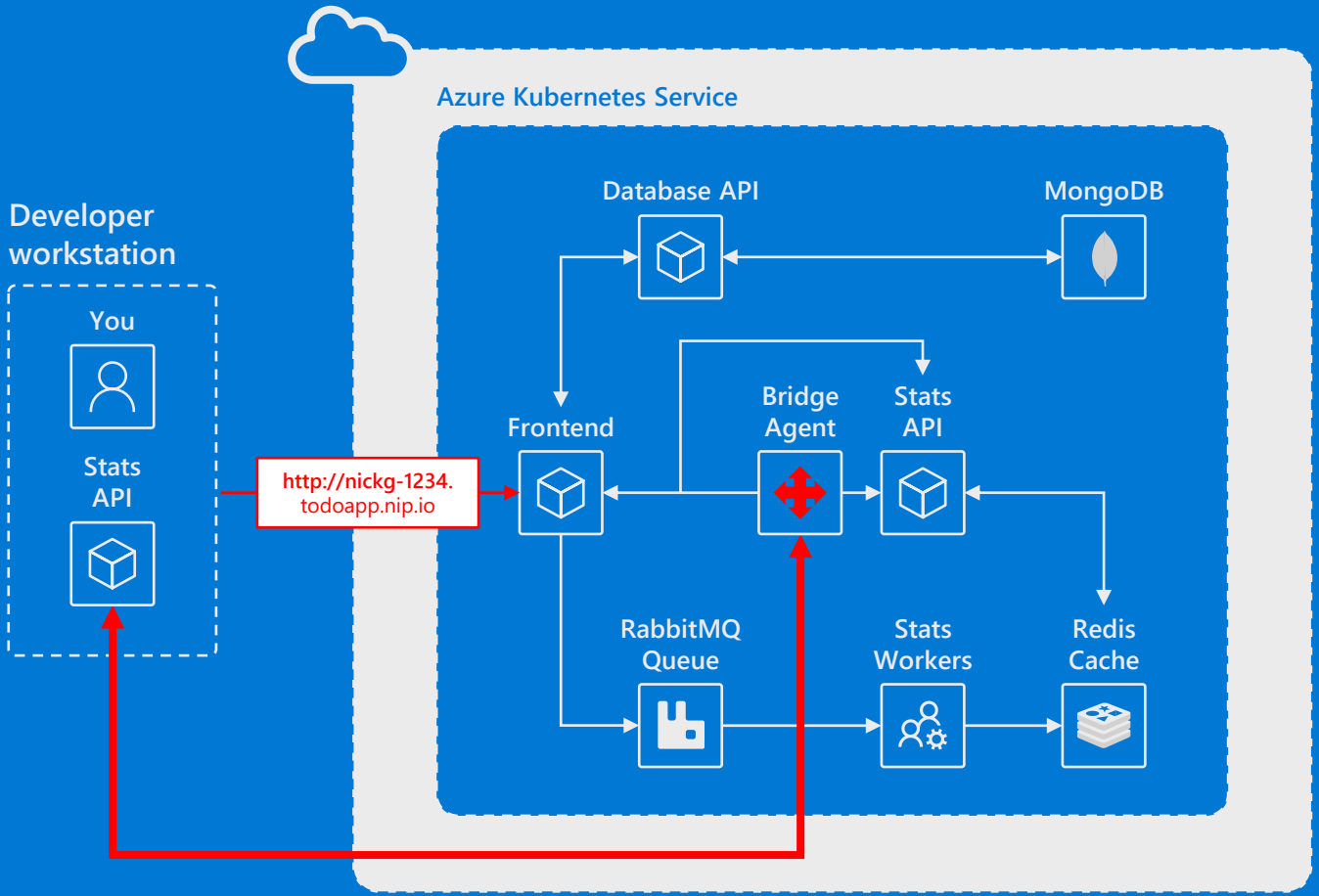
### Debug and test end-to-end

- Route traffic from the cluster to development workstations and back seamlessly

### Work in isolation in a shared development environment

- Work in a private "sandbox" environment by routing specific traffic locally

Learn more at
**aka.ms/bridge-to-k8s-vs-quickstart**
**aks.ms/bridge-to-k8s-vsc-quickstart**

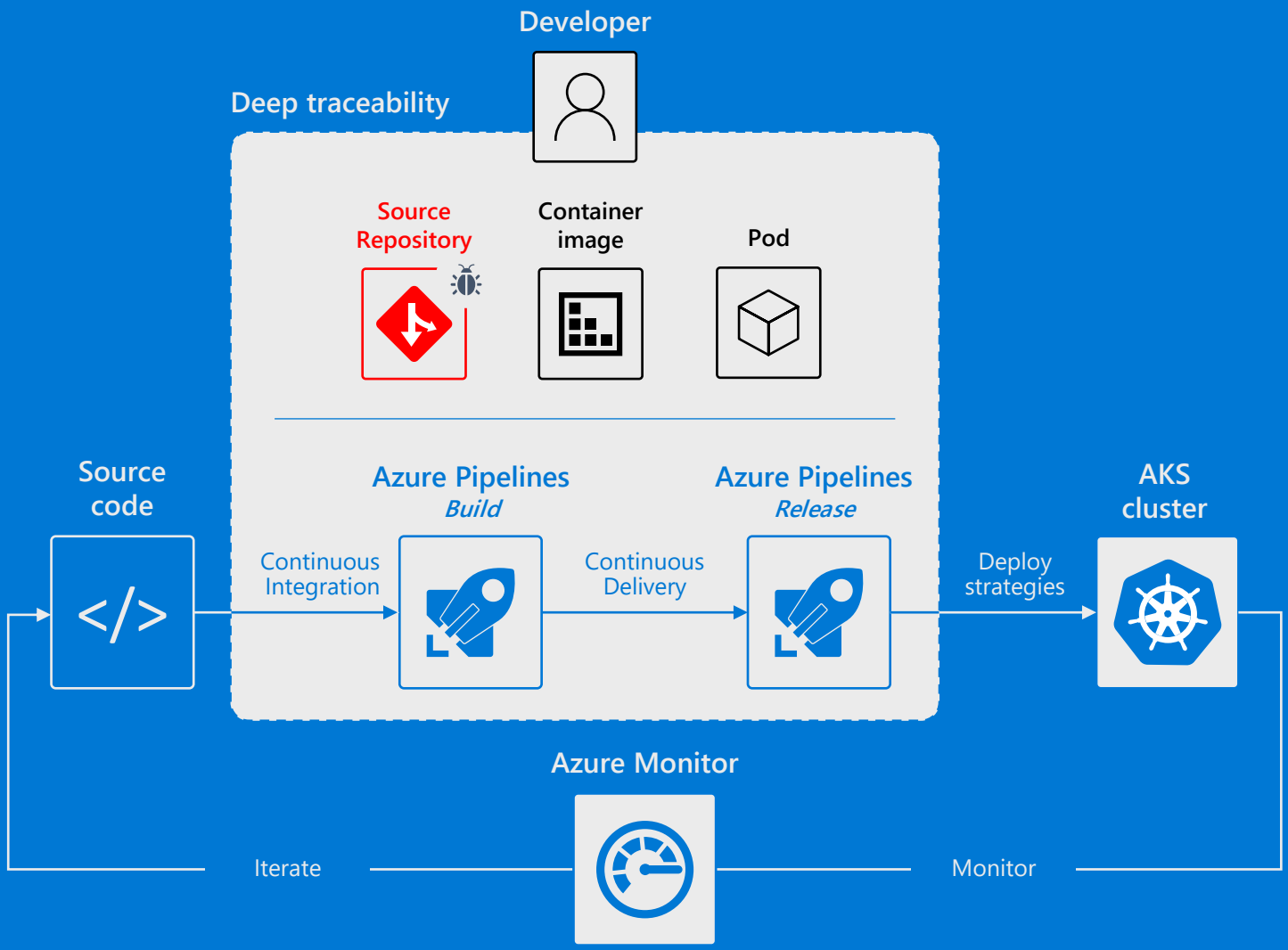

Azure Kubernetes Service
Database API
MongoDB
Developer workstation
You
Stats API
http://nickg-1234.todoapp.nip.io
Frontend
Bridge Agent
Stats API
RabbitMQ Queue
Stats Workers
Redis Cache

Capability

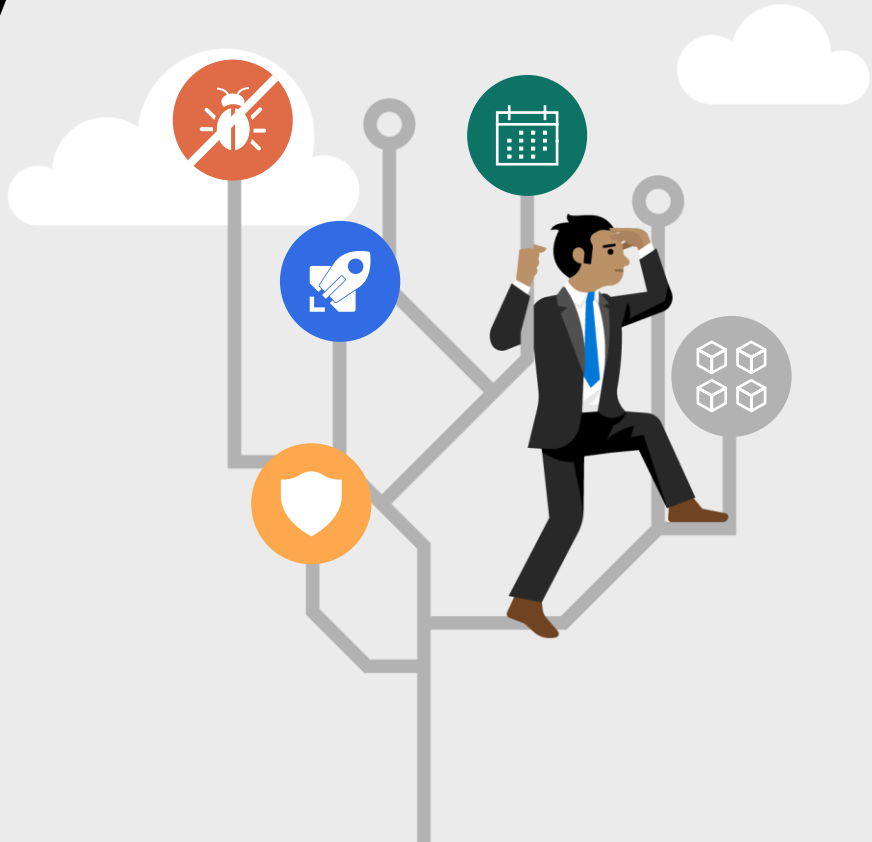
# Azure Pipelines for AKS

- Add a full CI/CD pipeline to your AKS cluster with automated routine tasks and multiple deployment strategies—all set up in just a few clicks
- Detect failures early and optimize your pipelines in a heartbeat with deep traceability into your deployments and source code

Learn more at
**aka.ms/aksbook/pipelines**


Developer
Deep traceability
Source Repository
Container image
Pod
Source code
Azure Pipelines
Build
Azure Pipelines
Release
AKS cluster
Continuous Integration
Continuous Delivery
Deploy strategies
Azure Monitor
Iterate
Monitor

# Balancing agility and security



Kubernetes applications are distributed by nature. Building and securing distributed applications is inherently difficult and must be carefully implemented.

To secure Kubernetes, it is not sufficient to just secure specifics such as the API server, kubectl, and access to the network. You need holistic, end–to-end security that spans from how you build your application all the way to how you run your application in the production environment.

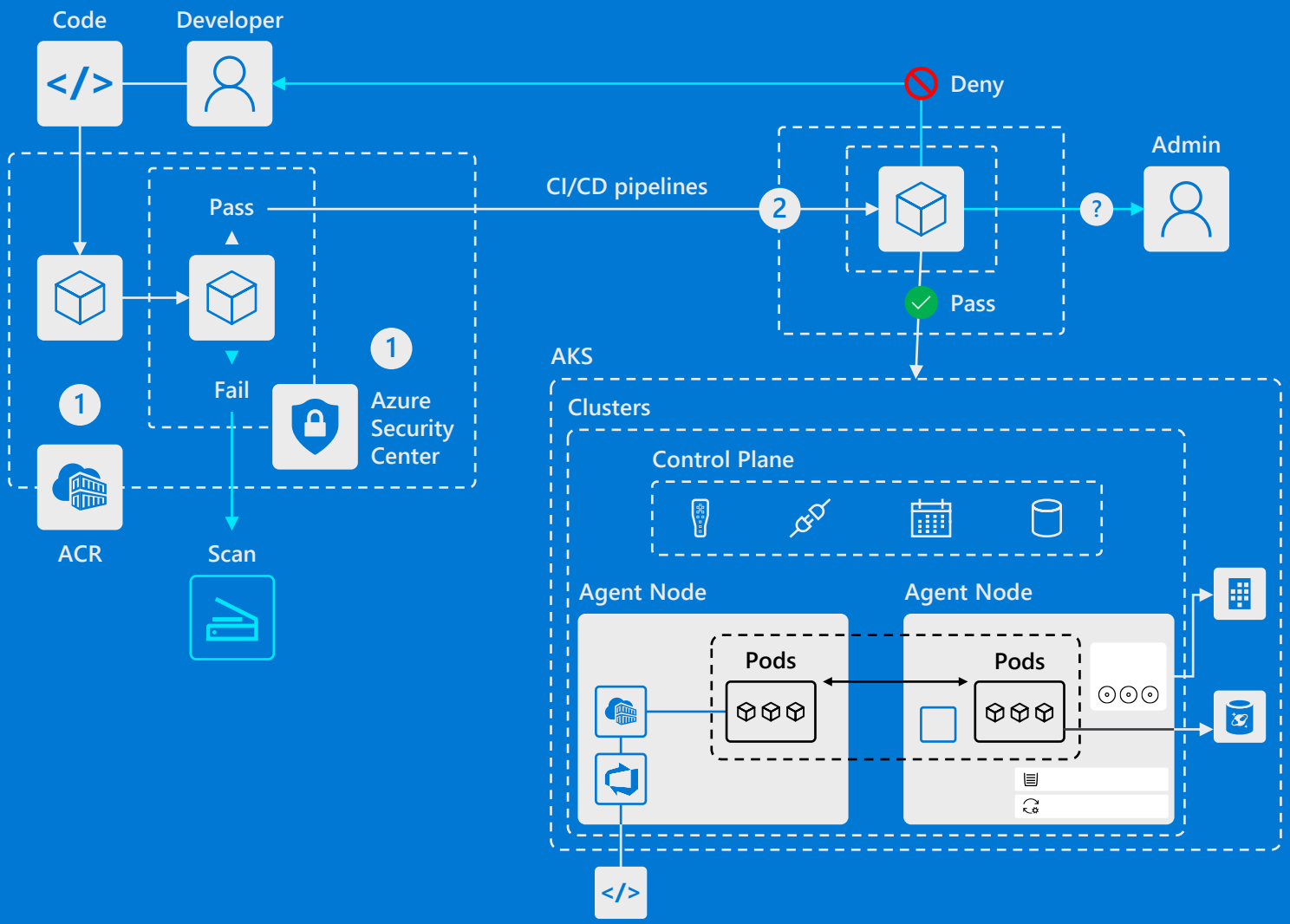
You want to build on a **secure, enterprise-grade platform that encompasses build security, registry security, cluster security, node security, and application security**.

# Build security

1. **Image vulnerability scanning**
   Azure Container Registry (ACR) only deploy images validated by Azure Security Center

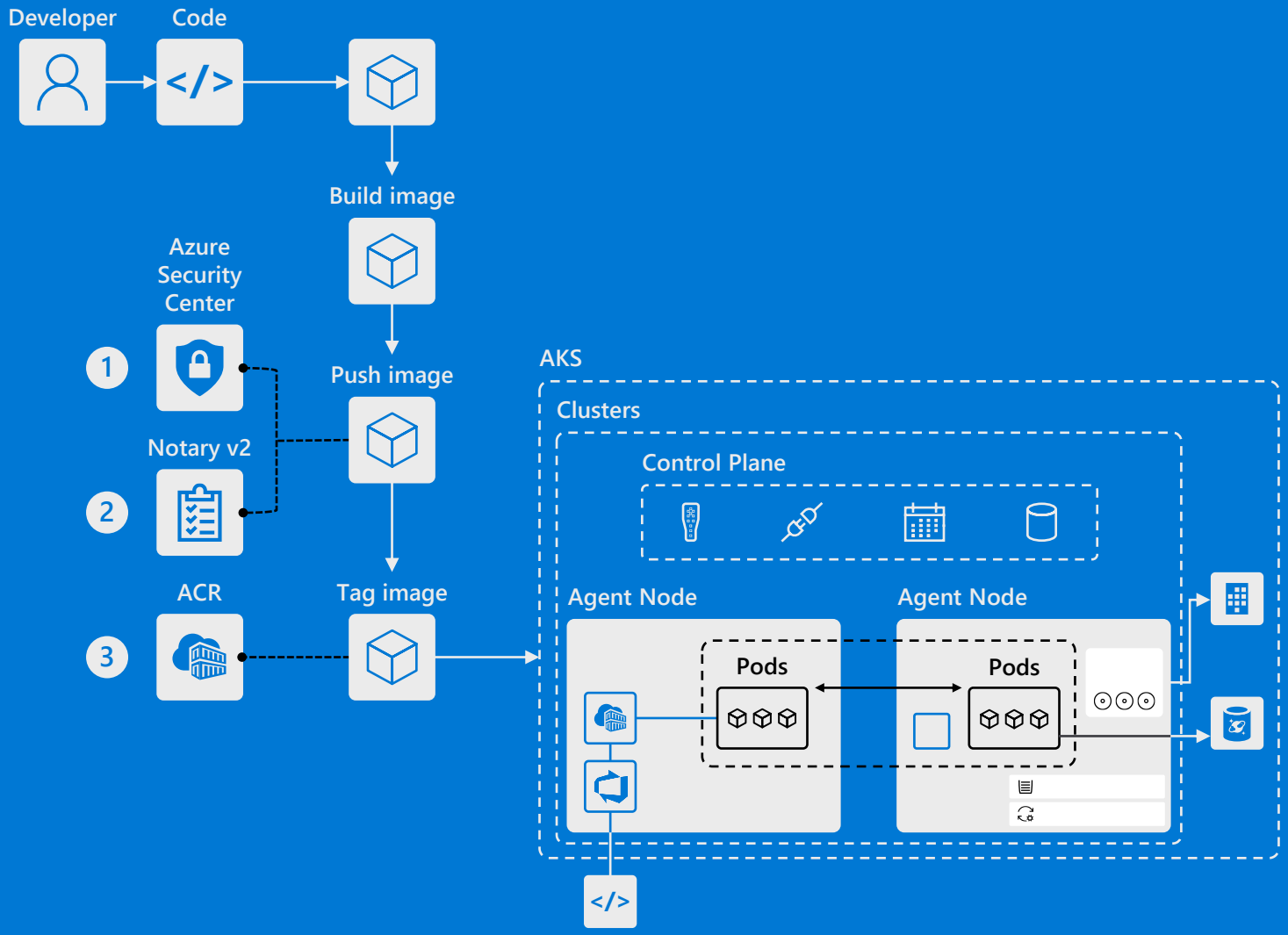
   Get actionable recommendations from Azure Security Center to help mitigate risks

2. **Compliance check**
   Leverage compliance policies on build to ensure applications meet governance standards before they are deployed (enabled through third-party solutions)


Code
Developer
Deny
Admin
CI/CD pipelines
2
?
Pass
Pass
Fail
1
1
Azure
Security
Center
ACR
Scan
AKS
Clusters
Control Plane
Agent Node
Agent Node
Pods
Pods

# Registry security

1. **Vulnerability assessment**
   Gain vulnerability information on images in the registry by integrating your pipeline with Azure Container Registry (ACR) and your registry with Azure Security Center
2. **Notary v2**
   Leverage Notary v2 to sign and ensure images through your pipeline are secure
3. **Private registry**
   Safeguard your IP and implement additional security measures for your images through ACR


Developer
Code
Build image
Azure Security Center
1
Push image
Notary v2
2
ACR
3
Tag image
AKS
Clusters
Control Plane
Agent Node
Agent Node
Pods
Pods

# Cluster security: resources

1. **Authentication and authorization**
   Kubernetes applications can access pods and other cloud resources that use managed Azure Active Directory (AAD) as an identity provider

   Make decisions and enforce organizational policies that keep AKS resource secure using AAD conditional access

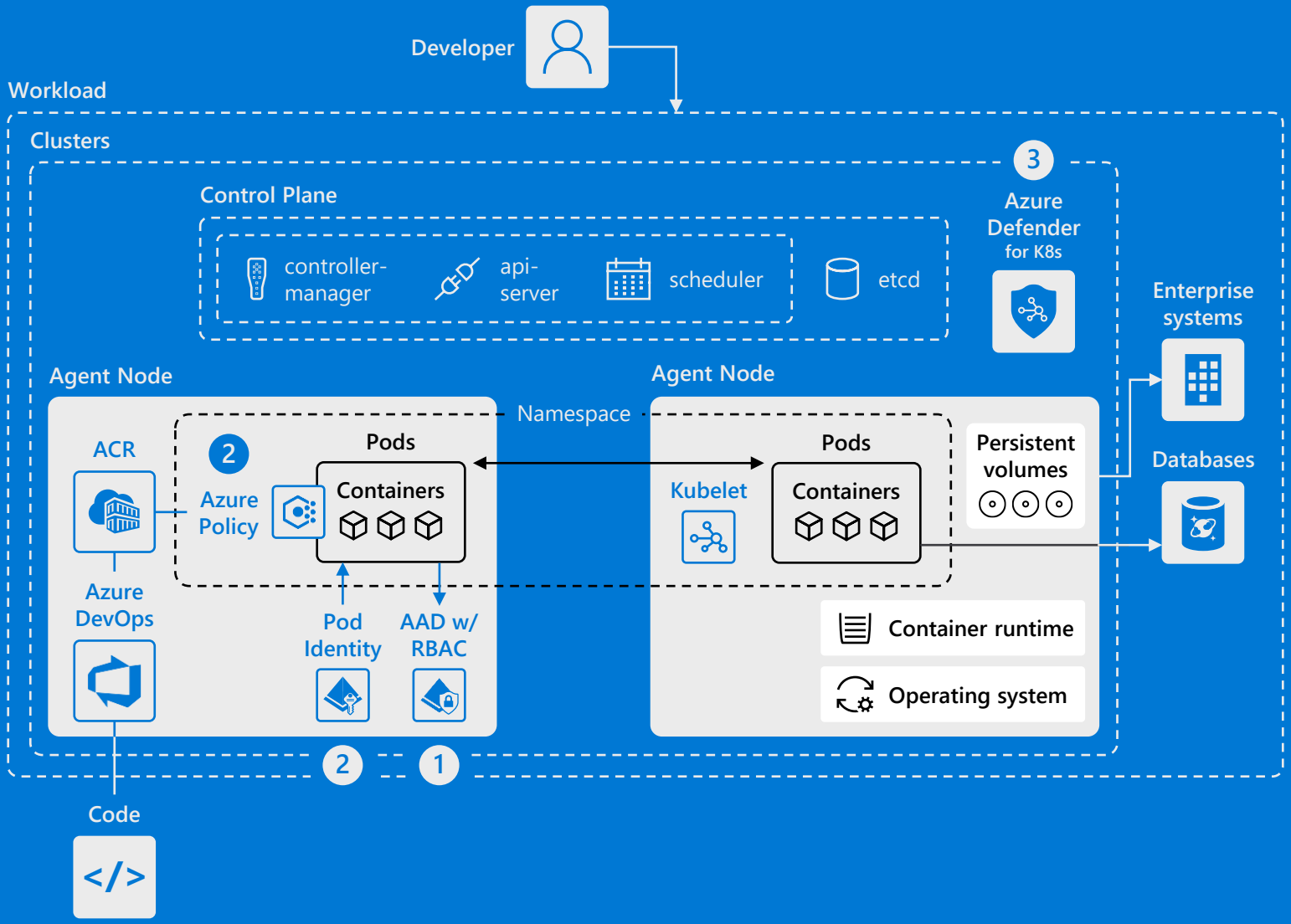
   AKS allows for just-in-time access rules to limit the threat of compromised accounts and other internal activities

   Enable AAD and Kubernetes RBAC to leverage conditional access capabilities and assign users, or specific groups of users, to create resources, view logs, etc.

2. **Pod security**
   Azure Policy with Gatekeeper and Open Policy Agent helps you consistently enforce pod security context and configurations across multiple clusters

   Ensure secure communications between AKS and other Azure entities/services with Pod Identity

3. **Azure Defender for Kubernetes**
   Investigate suspicious API requests, collect statistics, and create monitoring alerts—all from Kubernetes API audit logging through AKS

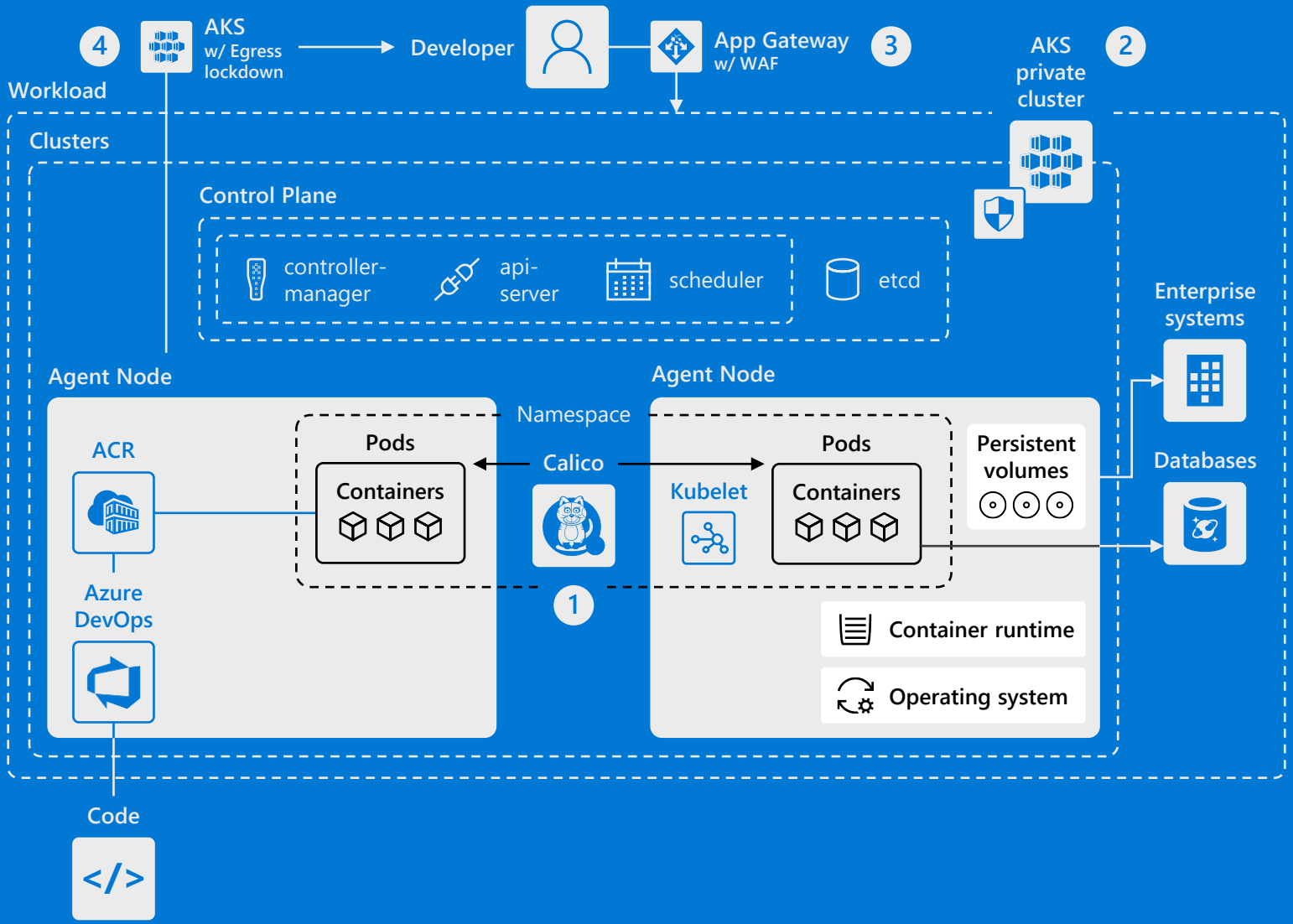

Developer
Workload
Clusters
3
Control Plane
controller-manager
api-server
scheduler
etcd
Azure Defender for K8s
Enterprise systems
Databases
Agent Node
Namespace
Agent Node
ACR
2
Pods
Azure Policy
Containers
Pods
Kubelet
Containers
Persistent volumes
Azure DevOps
Pod Identity
AAD w/ RBAC
Container runtime
Operating system
2
1
Code
</>

# Cluster security: network

1. **Pod-to-pod communication paths**
   Secure communication paths between namespaces and nodes and get better controls with a user-defined network policy
2. **AKS private clusters**
   Help ensure network traffic between your API server and node pools remains on the private network with internal IP addresses for the control plane or API server
3. **Threat and intrusion protection**
   Protect your web applications from common exploits and vulnerabilities using Application Gateway with Azure Web Application Firewall (WAF)

   Extend App Gateway to serve as the ingress (inbound) traffic load-balancer for Kubernetes pods within an AKS cluster too
4. **Outbound traffic restriction**
   Use Egress lockdown to prevent malicious cluster activities from originating and restrict egress (outbound) traffic for cluster nodes to a limited port

   Deny access to harmful resources from outside the network


4
AKS
w/ Egress
lockdown
Developer
App Gateway
w/ WAF
3
AKS
private
cluster
2
Workload
Clusters
Control Plane
controller-
manager
api-
server
scheduler
etcd
Enterprise
systems
Agent Node
Agent Node
Namespace
ACR
Pods
Containers
Calico
Pods
Kubelet
Containers
Persistent
volumes
Databases
Azure
DevOps
1
Container runtime
Operating system
Code
</>

# Node security

1. **Encryption of persistent volumes**
   Using host-based encryption, data stored on the VMs for your AKS agent nodes is encrypted at rest and flows as encrypted to the storage service

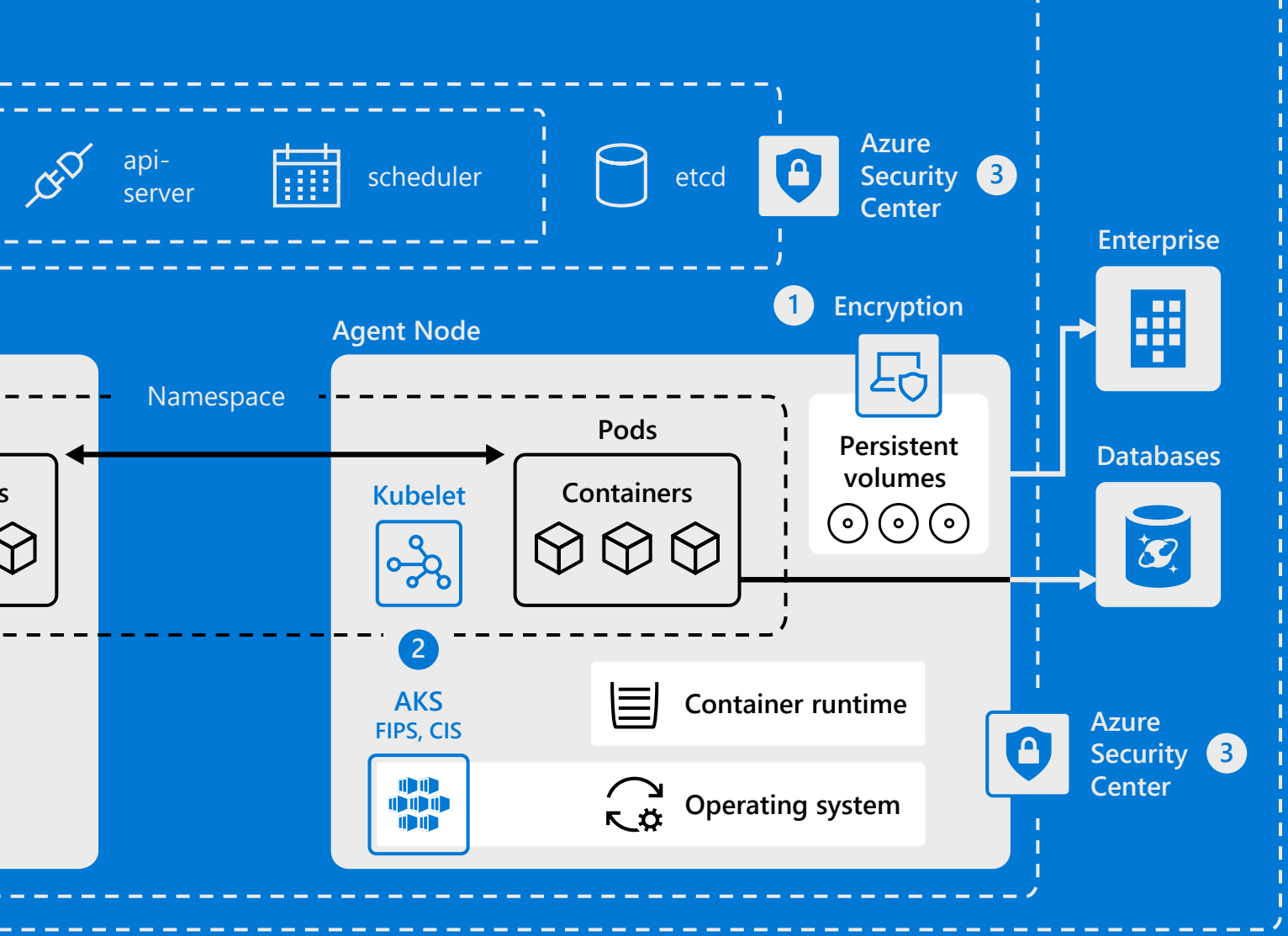
   This means temp disks, data disks, and OS cache are all encrypted at rest with either platform- or customer-managed keys

2. **Operating system security**
   Hardened OS image with automated OS patching

   **Federal Information Processing Standards (FIPS)**: achieve FedRAMP compliance required for meeting mandated security and computing standards

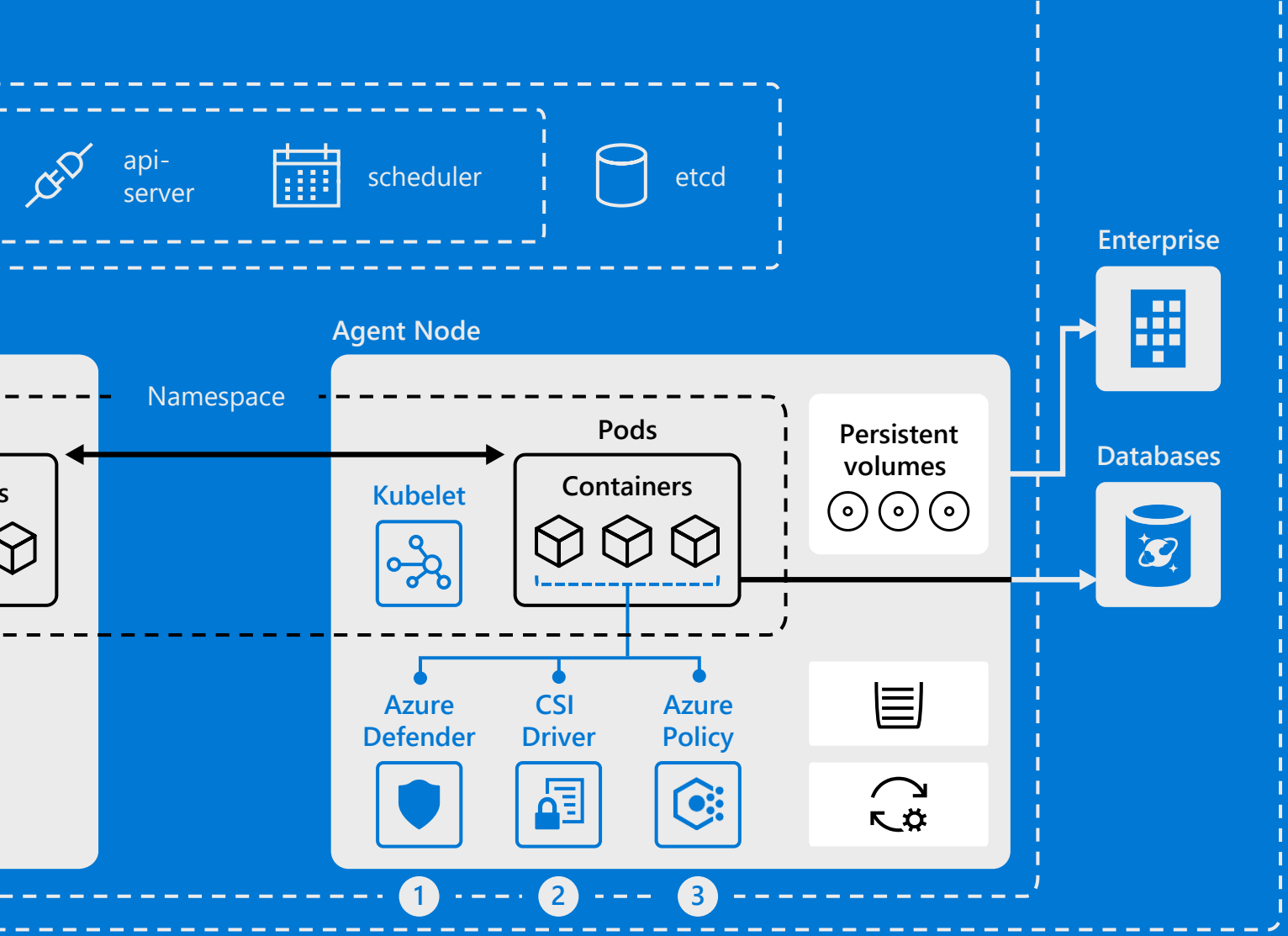
   **Center of Internet Security (CIS)**: Use the CIS AKS Security Benchmark V1.0.0 checklists and benchmark tools to validate cluster compliance

3. **Threat protection**
   Get automated threat detection and best practice recommendations for Kubernetes clusters and nodes from Azure Security Center

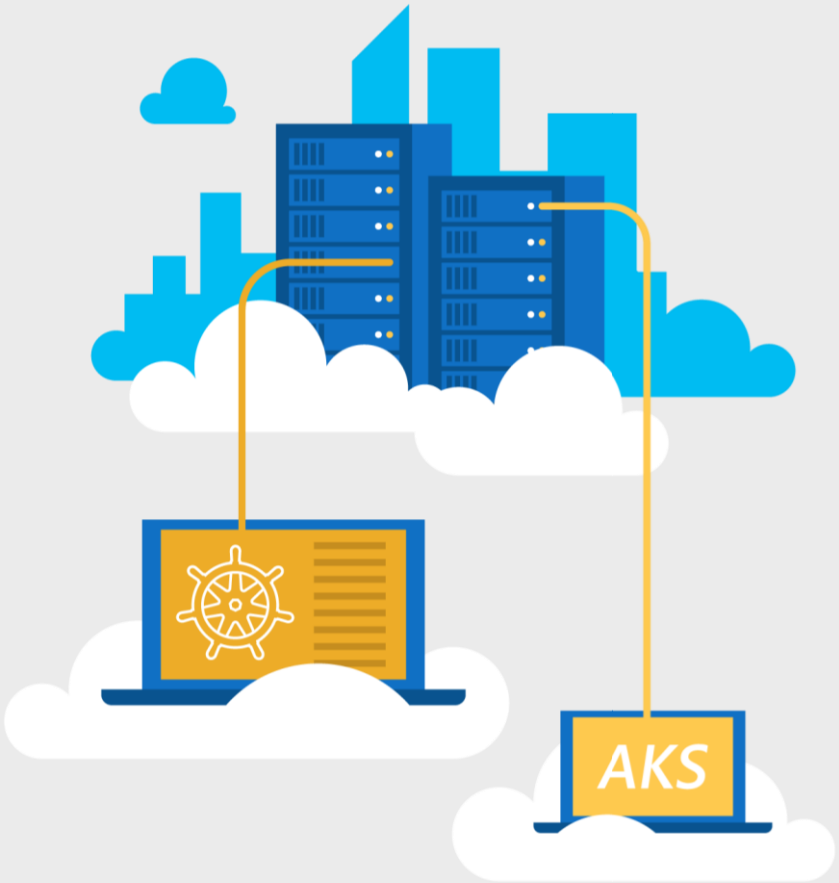

api-
server
scheduler
etcd
Azure
Security
Center
3
Enterprise
1
Encryption
Agent Node
Namespace
Pods
Persistent
volumes
Databases
Kubelet
Containers
2
AKS
FIPS, CIS
Container runtime
Operating system
Azure
Security
Center
3

# Application security

1. **Application security**
   Perform continual scanning of images running on your AKS nodes with the Azure Defender add-on
2. **Secrets management**
   Simplify secure access to secrets. Mount multiple secrets, keys, and certs to your pod as a CSI volume using the secret store Container Storage Interface (CSI) driver
3. **Azure Policy**
   Apply at-scale enforcements and safeguards on your clusters in a centralized, consistent manner


api-
server
scheduler
etcd
Enterprise
Agent Node
Namespace
Pods
Persistent
volumes
Databases
Kubelet
Containers
Azure
Defender
CSI
Driver
Azure
Policy
1
2
3

# Kubernetes anywhere



Many enterprise organizations choose a hybrid cloud approach so they can take advantage of cloud benefits while keeping certain workloads on premises. With the rise of edge computing, companies are also starting to build new apps that collect and process data at the edge of the network close to users and devices.

As Kubernetes continues to gain momentum, you are further faced with sprawling assets across heterogenous Kubernetes environments, compounding complexity to deploy, manage, and secure these resources and applications.

As a result, you need to govern and operate disparate environments across on-premises datacenters, branch offices, multiple clouds, and the edge and ensure integrated security across the entire organization. Ideally, you may want to **use familiar enterprise-grade tools to operate seamlessly across environments**.

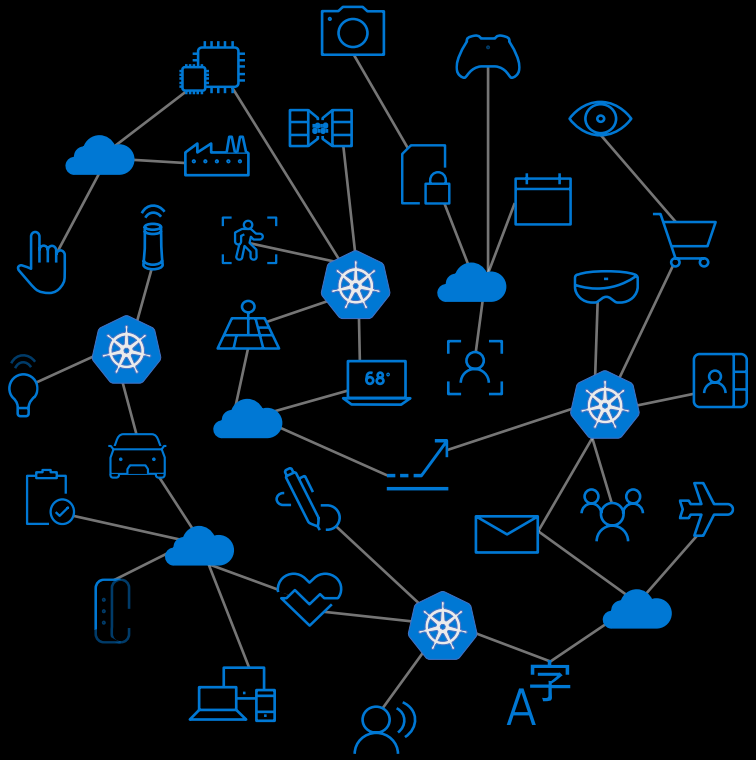
Azure Arc can help you achieve that. It extends Azure management to any infrastructure including VMs, Kubernetes, and more for unified management, governance, and control across on-premises, multi-cloud, and edge.

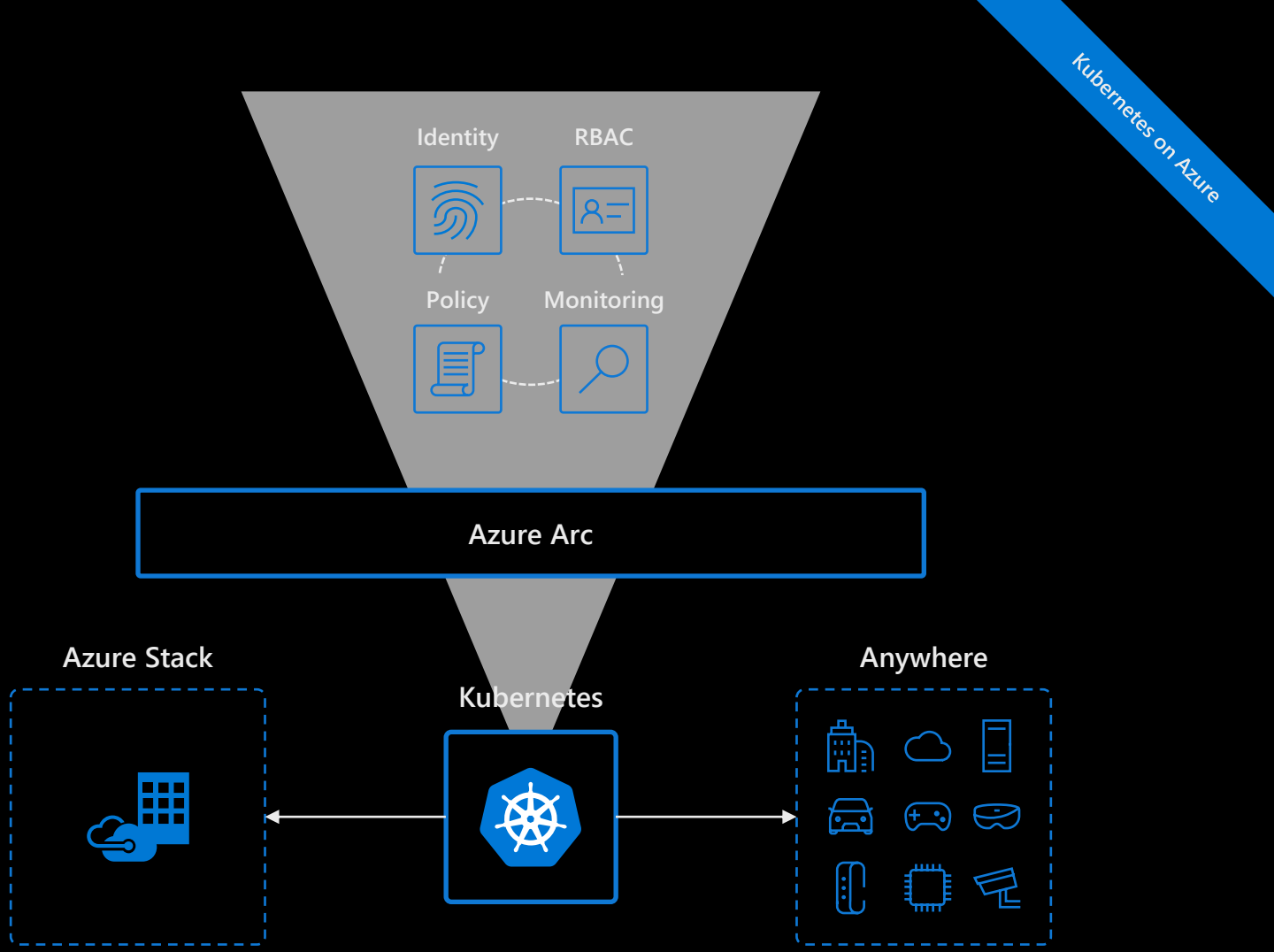
Kubernetes on Azure

# Operate seamlessly across clouds, data centers, and edge

- Central inventory and monitoring of the sprawling assets running anywhere **from on-premises to edge**
- Consistently apply policies and role-based access controls (RBACs) for **universal governance**
- Deploy Kubernetes resources to all clusters using a **GitOps**-based workflow
- Available for Kubernetes or any infrastructure

Kubernetes out-of-the-box



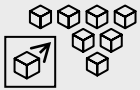

Kubernetes on Azure
Identity
RBAC
Policy
Monitoring
Azure Arc
Azure Stack
Kubernetes
Anywhere

"AKS allows us to deploy and run containers very fast, without dealing with the burden of allocating VMs, storage, and configuring networking. Moreover, changing decisions about deployment parameters is quick and easy."

— Giuseppe Zicari, Cloud Architect
Eni SpA

# Top AKS scenarios

**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

**DevSecOps**

**Security**

Deliver code faster and securely at scale

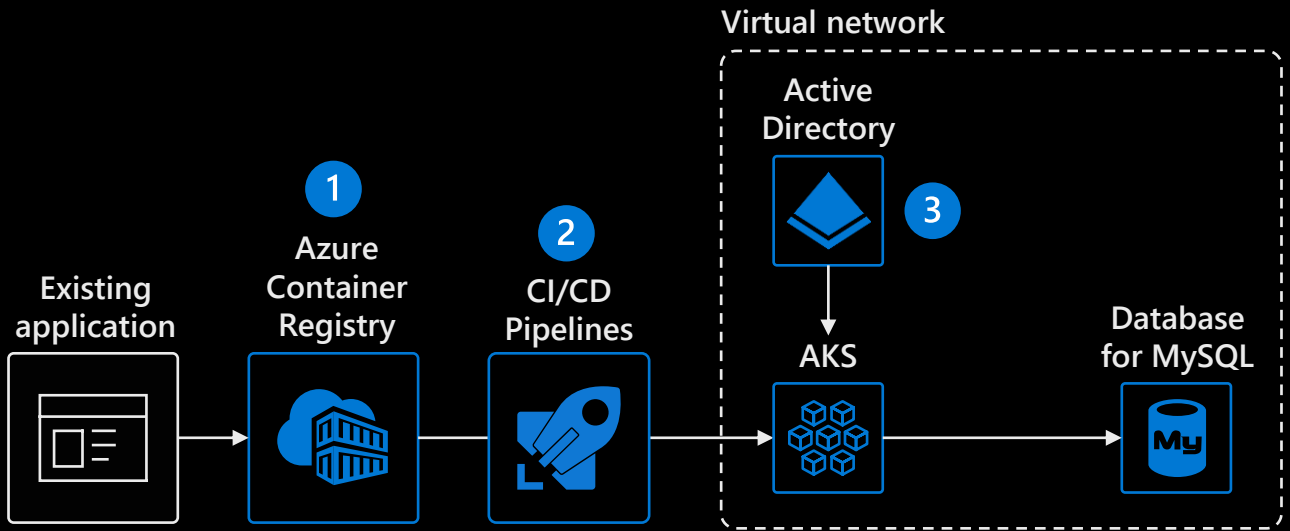
Lift and shift to containers

# App modernization without code changes

## Capabilities

1. Use **Azure Container Registry** to store container images and Helm charts for your modernized applications, replicated globally for low-latency image serving
2. Integrate AKS with **GitHub Actions** or other Kubernetes ecosystem tooling to enable continuous integration/continuous delivery (CI/CD)
3. Enhance security with **Azure Active Directory** and RBAC to control access to AKS resources

Learn more at
**aka.ms/aksbook/liftandshift**


Virtual network
Active Directory
3
1
Azure Container Registry
2
CI/CD Pipelines
Existing application
AKS
Database for MySQL

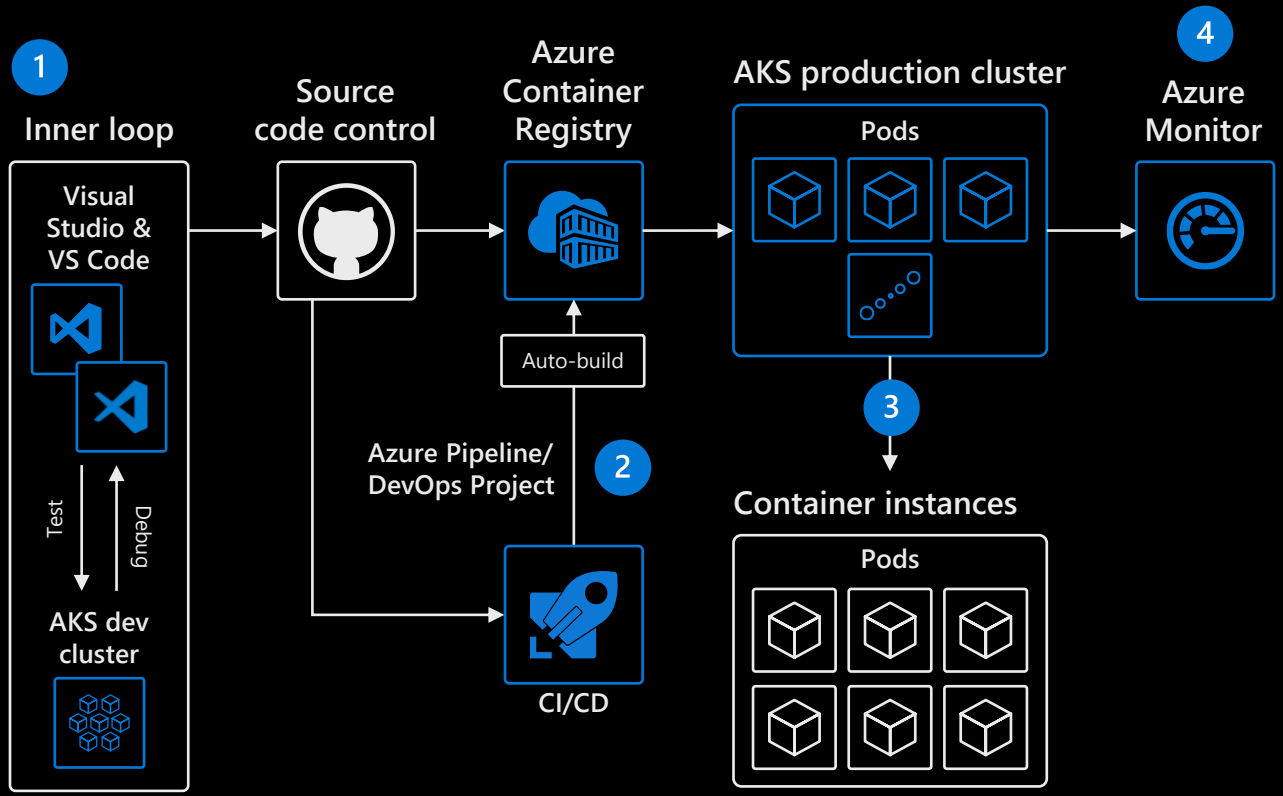
Microservices

# Microservices for faster app development

## Capabilities

1. Use **Bridge to Kubernetes** with Visual Studio or VS Code to run and debug microservice code on your development workstation, while connected to your Kubernetes cluster with the rest of your application or services
2. **Azure DevOps** has native integration with Helm and helps simplifying continuous integration/continuous delivery (CI/CD)
3. **Virtual nodes**—a Virtual Kubelet implementation—allows fast scaling of services for unpredictable traffic.
4. **Azure Monitor** provides a single pane of glass for monitoring app telemetry, cluster-to-container level health analytics.

Learn more at
**aka.ms/aksbook/microservices**


1
Inner loop
Visual Studio & VS Code
Test
Debug
AKS dev cluster
Source code control
Azure Container Registry
Auto-build
Azure Pipeline/ DevOps Project
2
CI/CD
AKS production cluster
Pods
3
Container instances
Pods
4
Azure Monitor

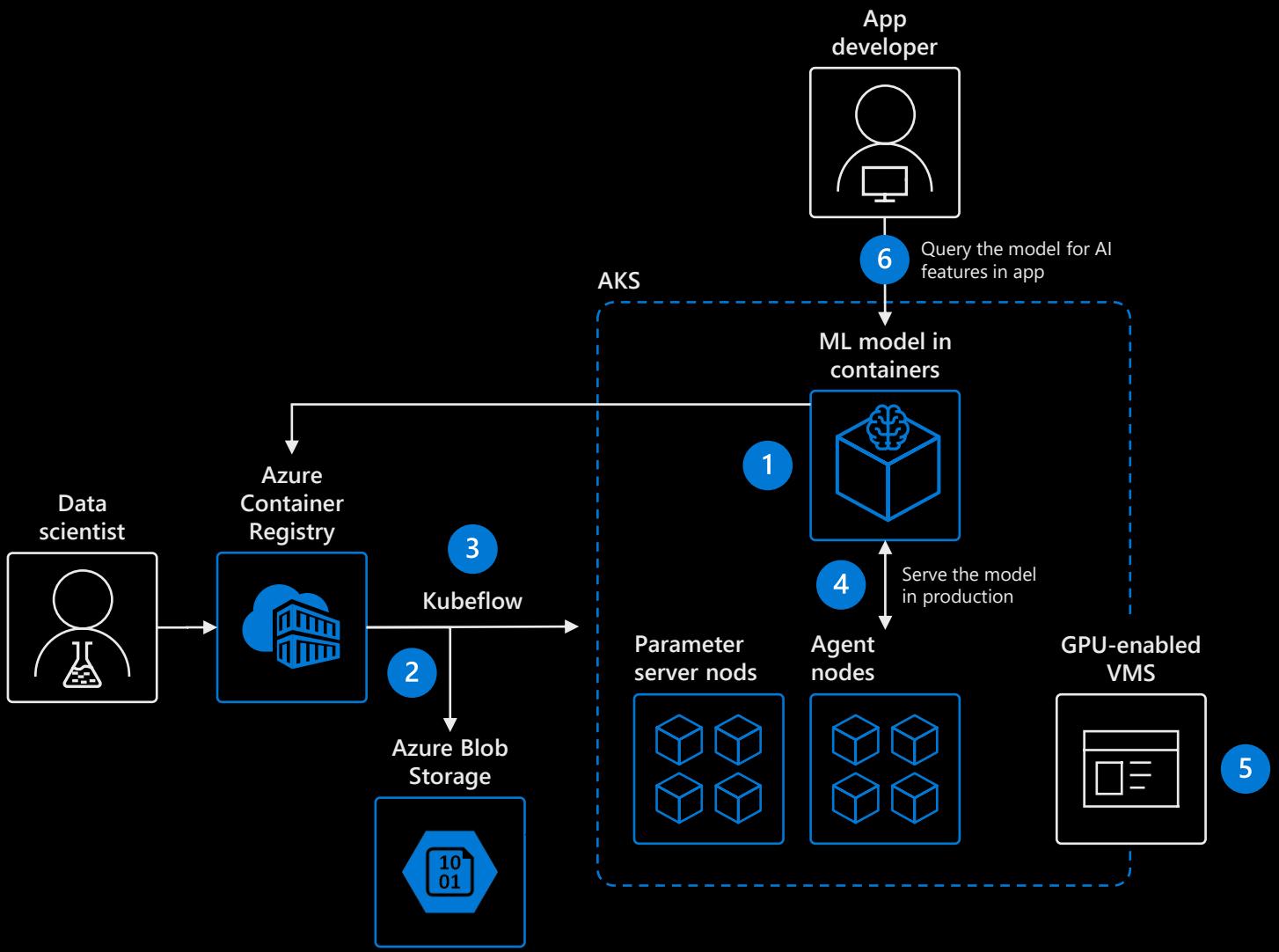
Machine learning

# Data scientist in a box

## Capabilities

1. Package ML model into a container and publish to **Azure Container Registry**
2. **Azure Blob Storage** hosts training data sets and trained model
3. Use **Kubeflow** to deploy training job to AKS; distributed training job to AKS includes Parameter servers and Agent nodes
4. Serve production model using **Kubeflow**, promoting a consistent environment across test, control, and production
5. AKS supports **GPU-enabled VM**
6. Developer can build features querying the model running in AKS cluster

Learn more at
**aka.ms/aksbook/ml**


App developer
6
Query the model for AI features in app
AKS
ML model in containers
1
Azure Container Registry
Data scientist
3
Kubeflow
2
Azure Blob Storage
4
Serve the model in production
Parameter server nods
Agent nodes
GPU-enabled VMS
5

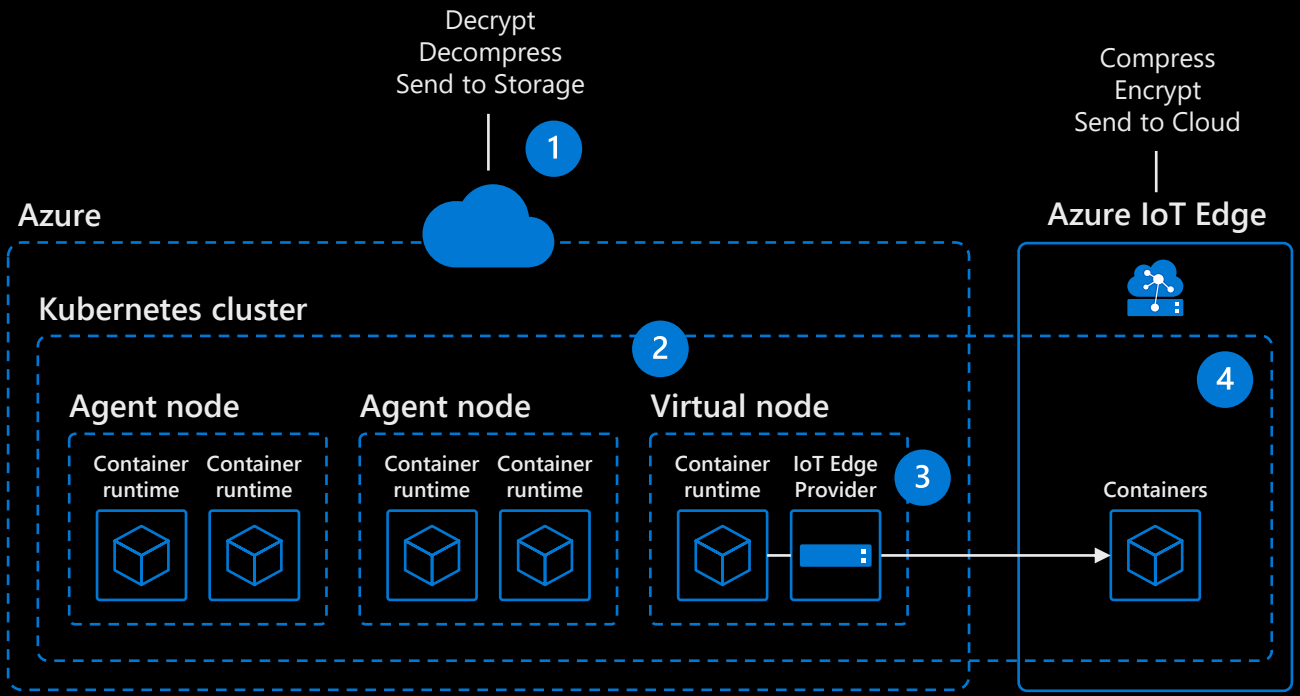
# Scalable Internet of Things solutions

## Capabilities

1. **Azure IoT Edge** encrypts data and send to Azure, which then decrypts the data and sends to storage
2. **Virtual node**, an implementation of Virtual Kubelet, serves as the translator between cloud and Edge
3. **IoT Edge Provider in virtual node** redirects containers to IoT Edge and extend AKS cluster to target millions of edge devices
4. Consistent update, management, and monitoring as one unit in AKS using single pod definition

Learn more at
**aka.ms/aksbook/iot**

IoT


Decrypt
Decompress
Send to Storage
1
Compress
Encrypt
Send to Cloud
Azure
Azure IoT Edge
Kubernetes cluster
2
4
Agent node
Agent node
Virtual node
Container runtime
Container runtime
Container runtime
Container runtime
Container runtime
IoT Edge Provider
3
Containers

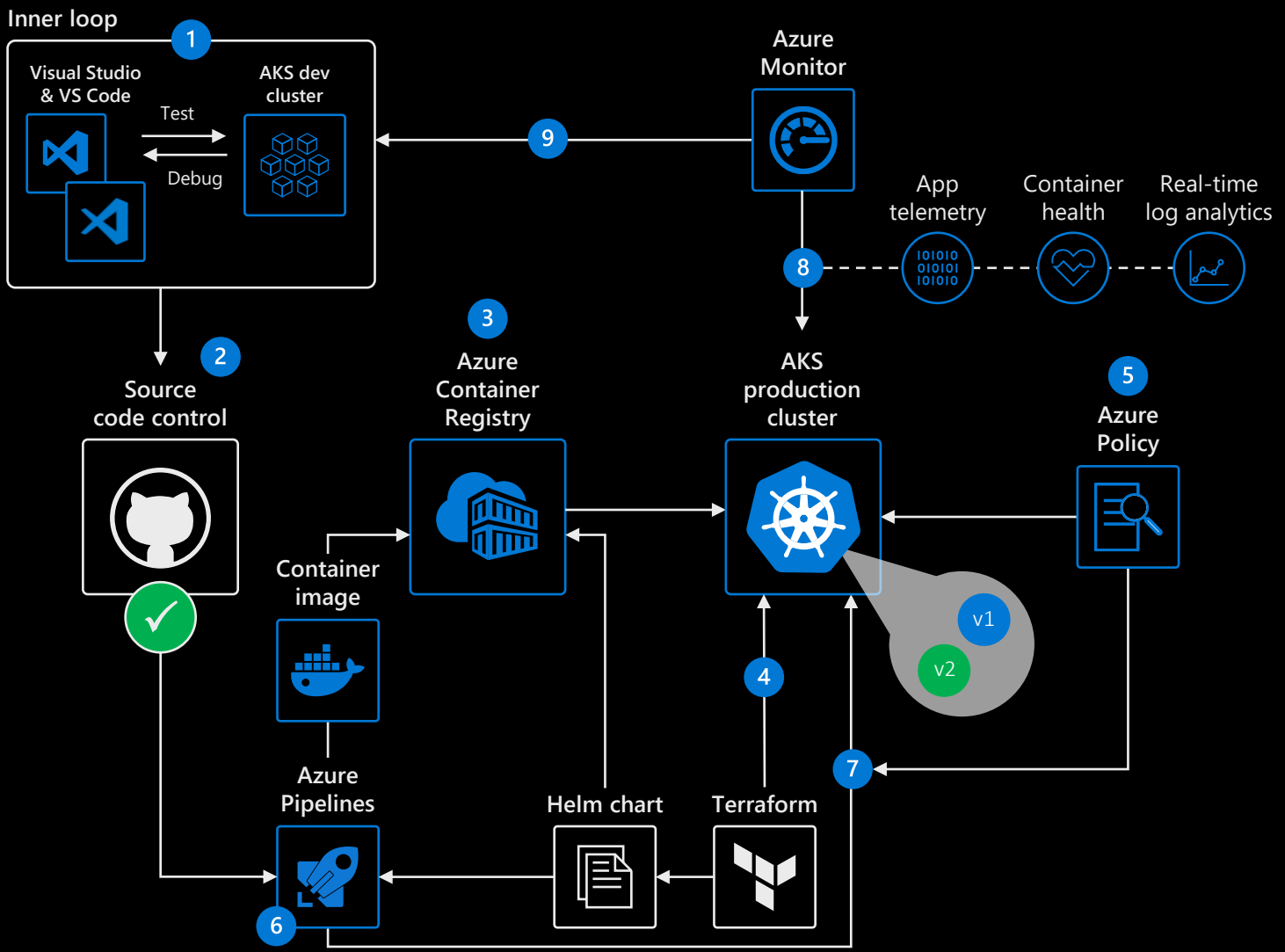
# DevSecOps

## Capabilities

1. Developers rapidly iterate, test, and debug different parts of an application together in the same Kubernetes cluster
2. Code is merged into a GitHub repository, after which automated builds and tests are run by Azure Pipelines
3. Container image is registered in Azure Container Registry
4. Kubernetes clusters are provisioned using tools like Terraform; Helm charts, installed by Terraform, define the desired state of app resources and configurations
5. Operators enforce policies to govern deployments to the AKS cluster
6. Release pipeline automatically executes pre-defined deployment strategy with each code change
7. Policy enforcement and auditing is added to CI/CD pipeline using Azure Policy
8. App telemetry, container health monitoring, and real-time log analytics are obtained using Azure Monitor
9. Insights used to address issues and fed into next sprint plans

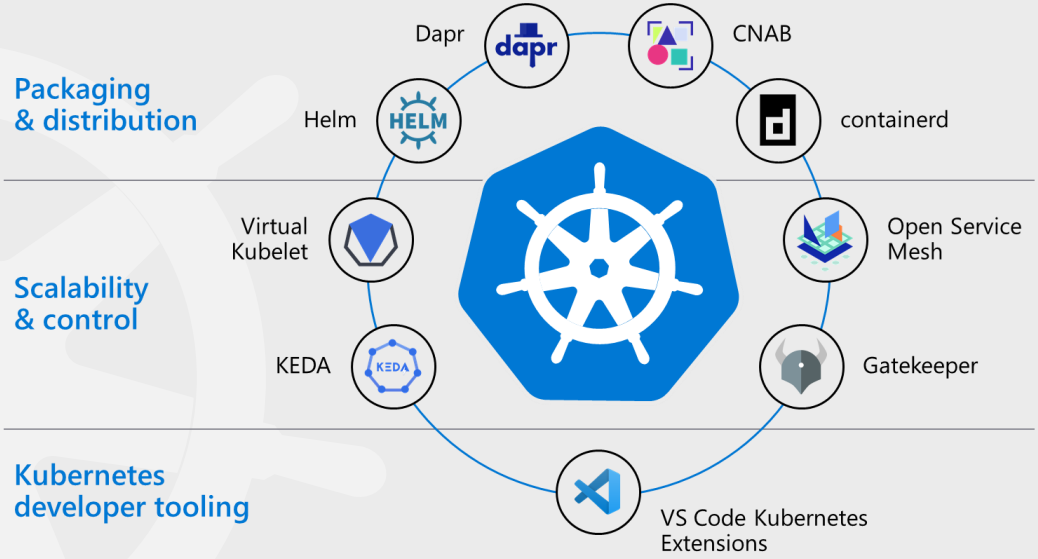
Learn more at
aka.ms/aksbook/devsecops

DevSecOps


Inner loop
1
Visual Studio & VS Code
Test
Debug
AKS dev cluster
Azure Monitor
9
App telemetry
Container health
Real-time log analytics
8
2
Source code control
3
Azure Container Registry
AKS production cluster
5
Azure Policy
Container image
v1
v2
4
7
Azure Pipelines
Helm chart
Terraform
6

# Microsoft's contributions to the community

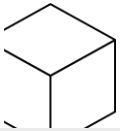
Microsoft brings expertise from working with diverse customers to the Kubernetes community, giving developers access to the latest Microsoft learnings and technologies, and making Kubernetes enterprise-friendly and easier to use.



# Best support for your needs

**Learning path**
aka.ms/LearnKubernetes

**What is Kubernetes**
aka.ms/aks/k8sLearning



**Case studies**
aka.ms/aks/casestudy

**Azure Kubernetes**
aka.ms/aks/page

**Hear from experts**
aka.ms/aks/videos

**Try for free**
aka.ms/aks/trial

Microsoft Azure

Published
October 2021