Certified Kubernetes Administrator

Azure Kubernetes Service

# Learning Objectives

By the end of this lesson, you will be able to:

- Understand Azure Kubernetes Service and its components

- Create and deploy Kubernetes workloads in an AKS cluster

- Perform Pod scheduling and rollout update for workloads

- Demonstrate persistent storage creation on the AKS cluster

Kubernetes in Cloud Environment

Kubernetes and Cloud Environment

Most cloud service providers have integrated Kubernetes as a part of the platform for their clients. Some of the most important and influential providers are:

Amazon **01**

02 Google

03 Microsoft

04 OpenStack
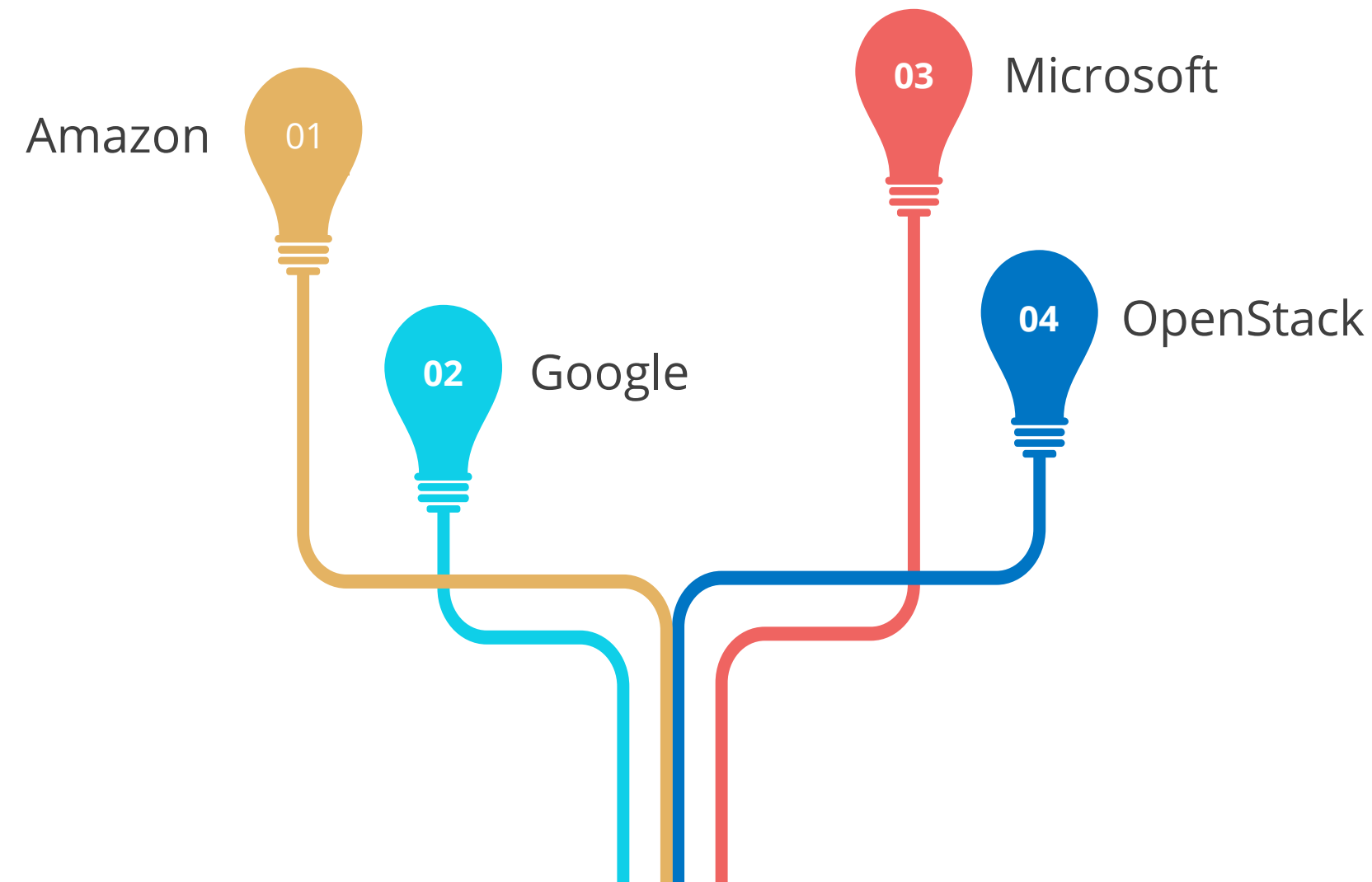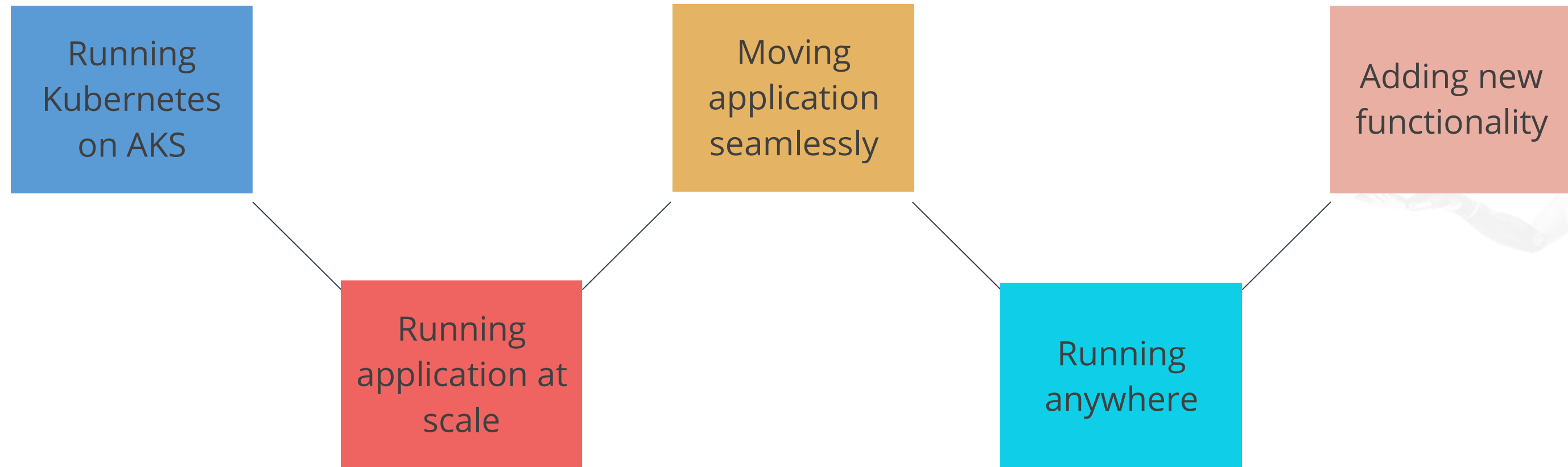
Azure Cloud Platform

- Azure is a cloud computing service created by Microsoft for building, testing, deploying, and managing applications.

- As part of the cloud solution to clients, Microsoft provides special integration of Kubernetes with the Azure Cloud Platform.

Azure Features

Kubernetes is an open-source project that can be used to run containerized applications anywhere, without needing to change operational tooling. Azure-specific features of Kubernetes include:
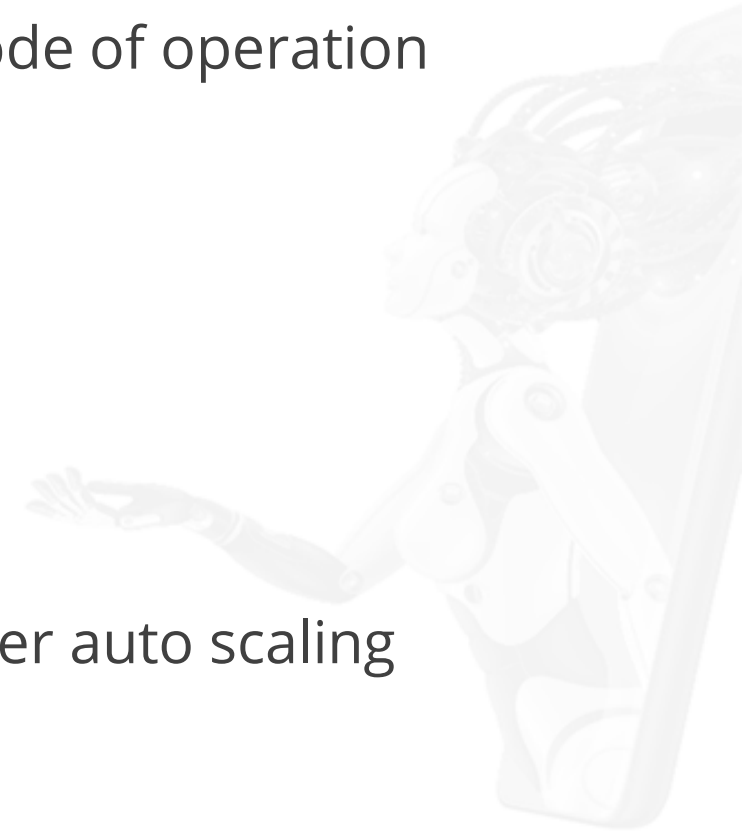
Running Kubernetes on AKS

Running application at scale

Moving application seamlessly

Running anywhere

Adding new functionality

Key Features of Kubernetes on Azure

Enhanced Workload and network security

Automated mode of operation

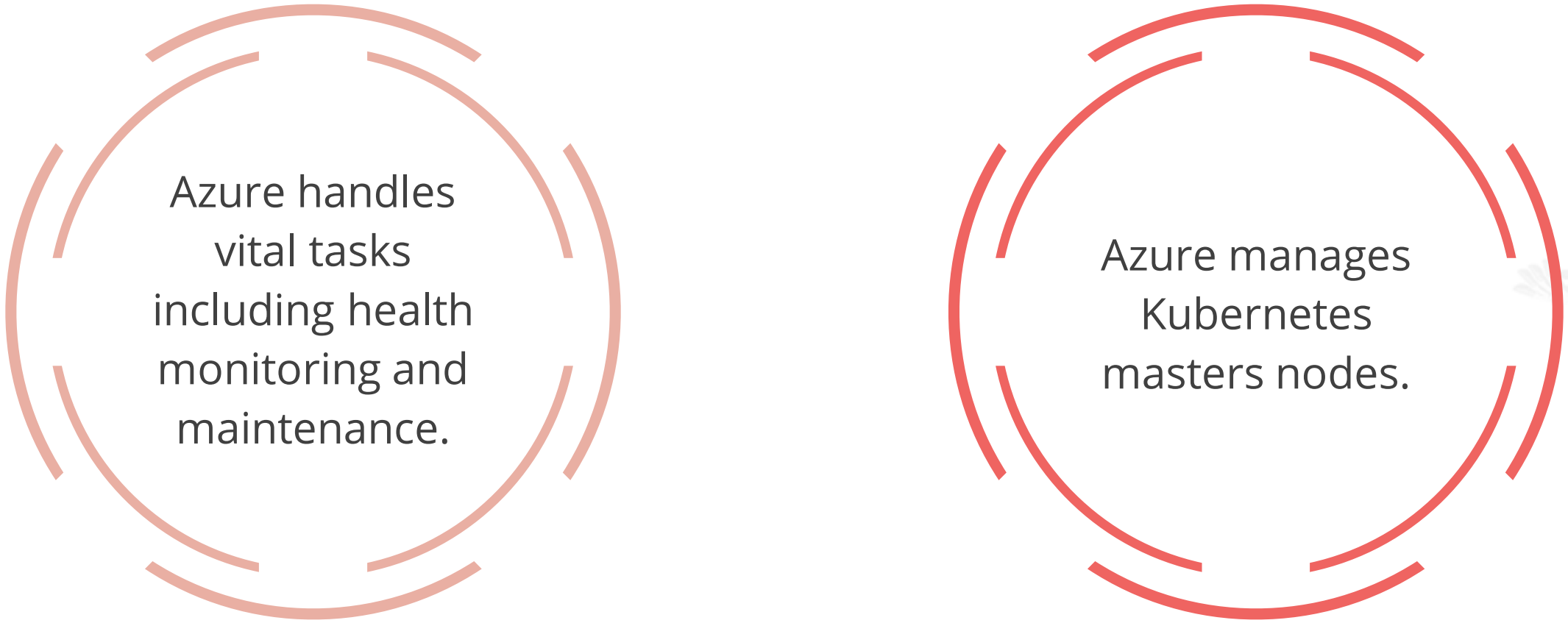High Performance Kubernetes applications

Pod and cluster auto scaling

Azure Kubernetes Service

Azure Kubernetes Service (AKS) manages hosted Kubernetes environment and makes it easy to deploy and manage containerized applications in Azure. The AKS environment is enabled with features such as automated updates, self-healing, and easy scaling.
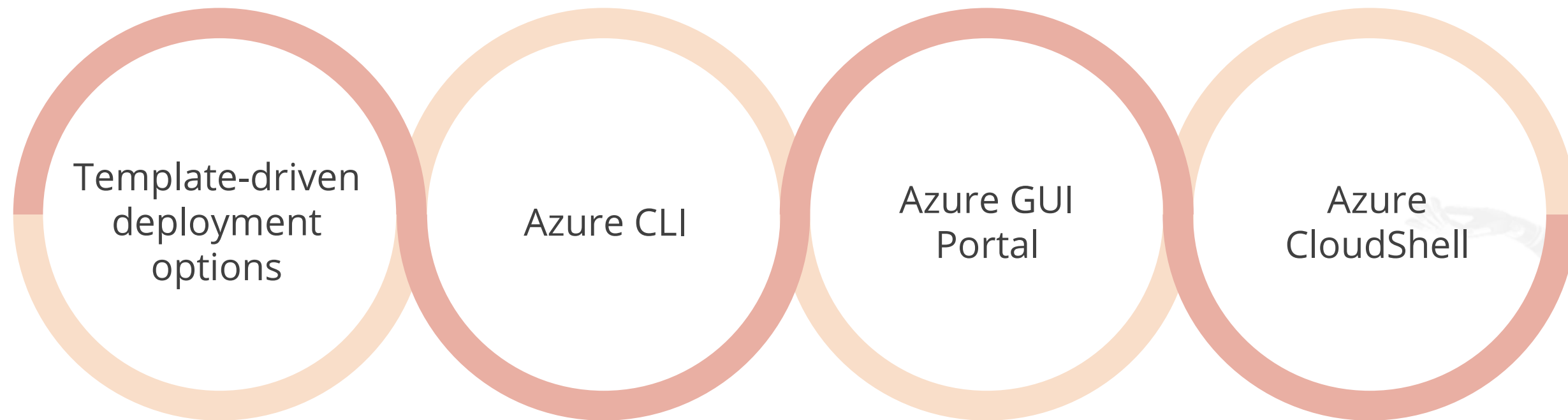
Azure handles vital tasks including health monitoring and maintenance.

Azure manages Kubernetes masters nodes.

AKS Components

AKS provides the following components as a part of the AKS, which help in managing the Kubernetes instance on the Azure instance.

Template-driven deployment options

Azure CLI
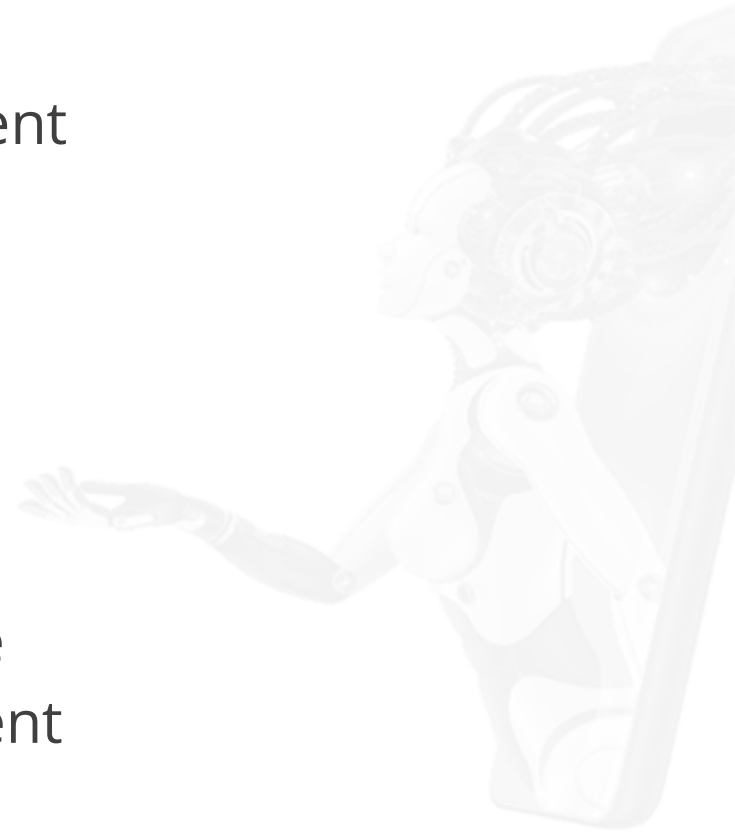
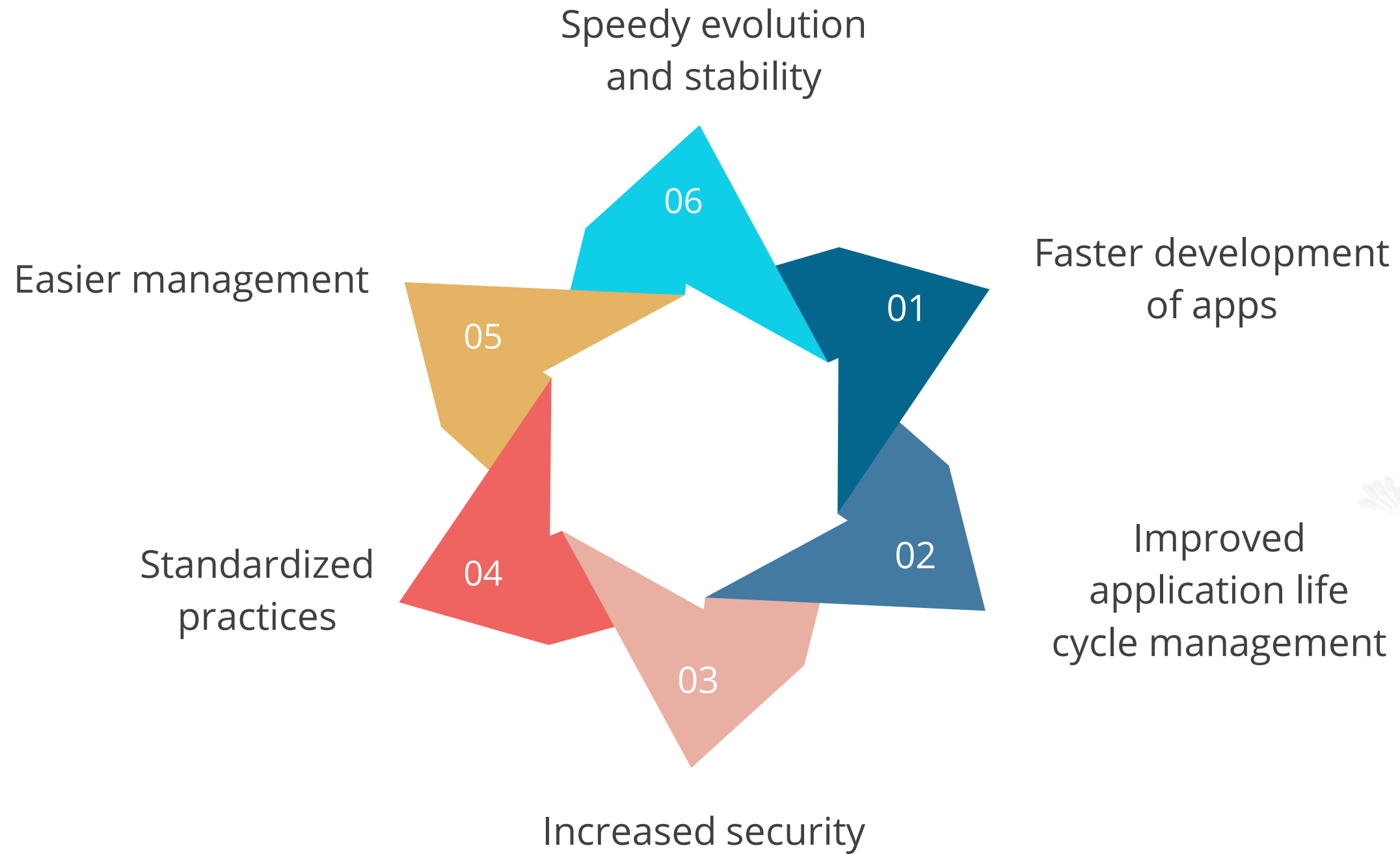Azure GUI Portal
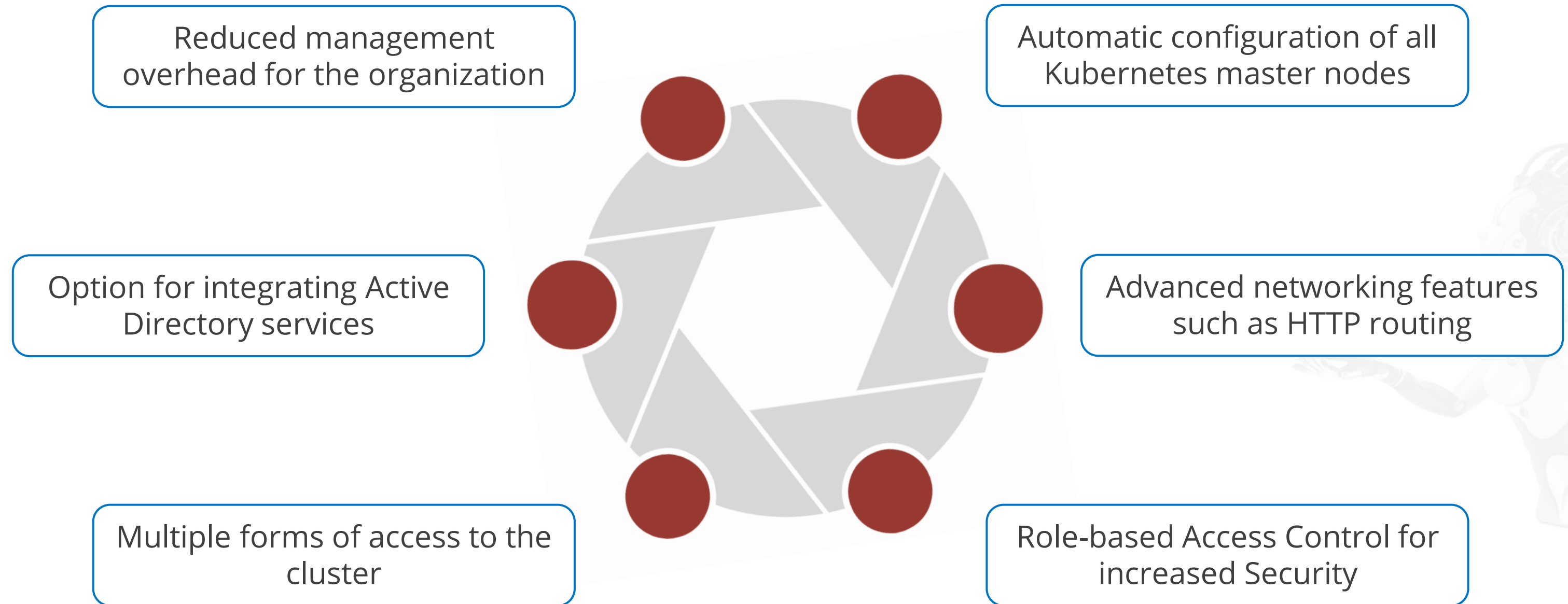
Azure CloudShell

Cloud Interoperability

Cloud interoperability evolves apace with its platforms. Kubernetes and Azure can be combined to create cloud-native applications.

Benefits of Leveraging AKS

Speedy evolution and stability — 06
Faster development of apps — 01
Improved application life cycle management — 02
Increased security — 03
Standardized practices — 04
Easier management — 05

Advantages of AKS over Local Deployment

Reduced management overhead for the organization

Automatic configuration of all Kubernetes master nodes

Option for integrating Active Directory services

Advanced networking features such as HTTP routing

Multiple forms of access to the cluster
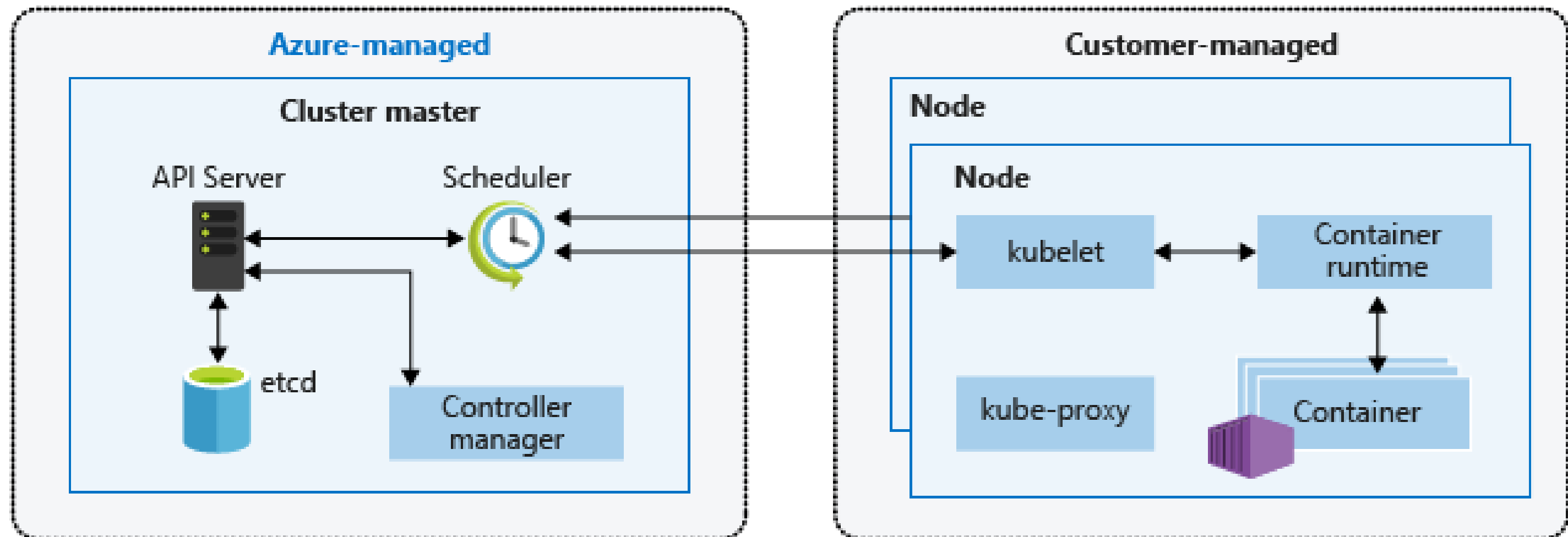
Role-based Access Control for increased Security

AKS Service Components

AKS Cluster Creation

You can either create your cluster in the Azure portal or use the Azure CLI. When you create the cluster, you can use the resource manager templates to automate cluster creation.

Namespaces in AKS

Kubernetes resources, such as pods and deployments, are logically grouped into a namespace to divide an AKS cluster and restrict create, view, or manage access to resources.
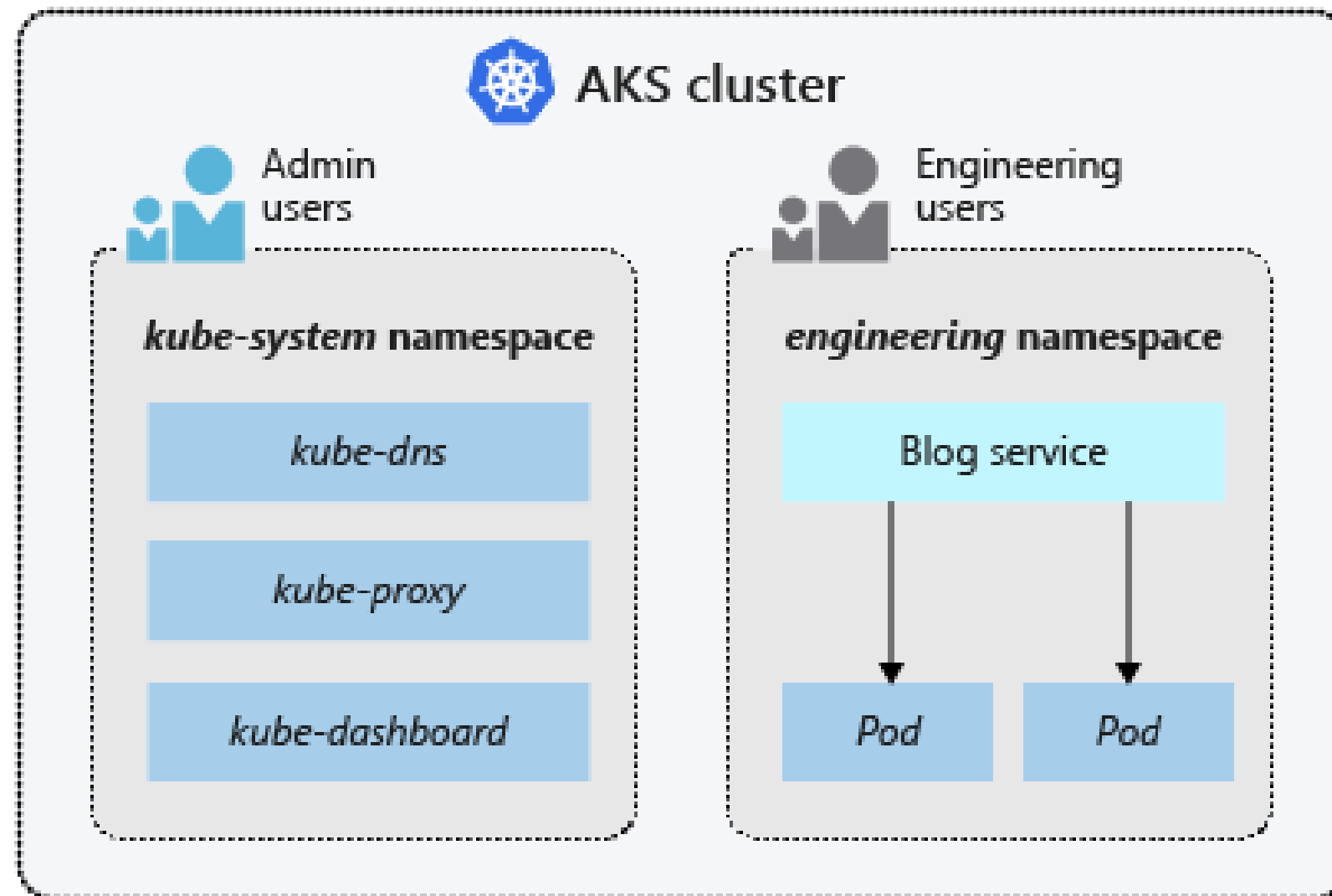
Creating Workloads

AKS supports the Docker image format that means that you can use any development environment to create a workload, package the workload as a container and deploy the container as a Kubernetes pod.
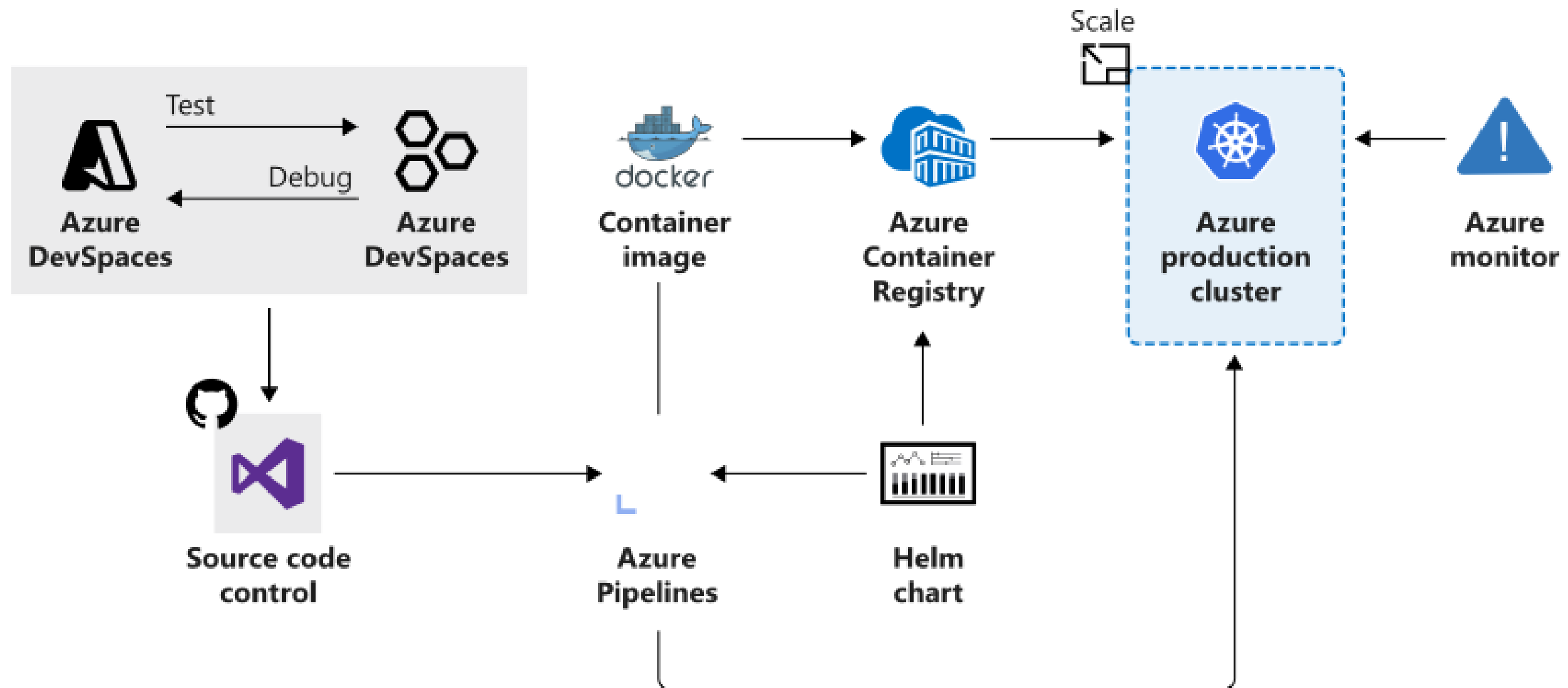
Creating and Deploying Pods

Creating and deploying pods in AKS is similar to the process in the local environment. These pods can then be controlled remotely using the AKS tools.
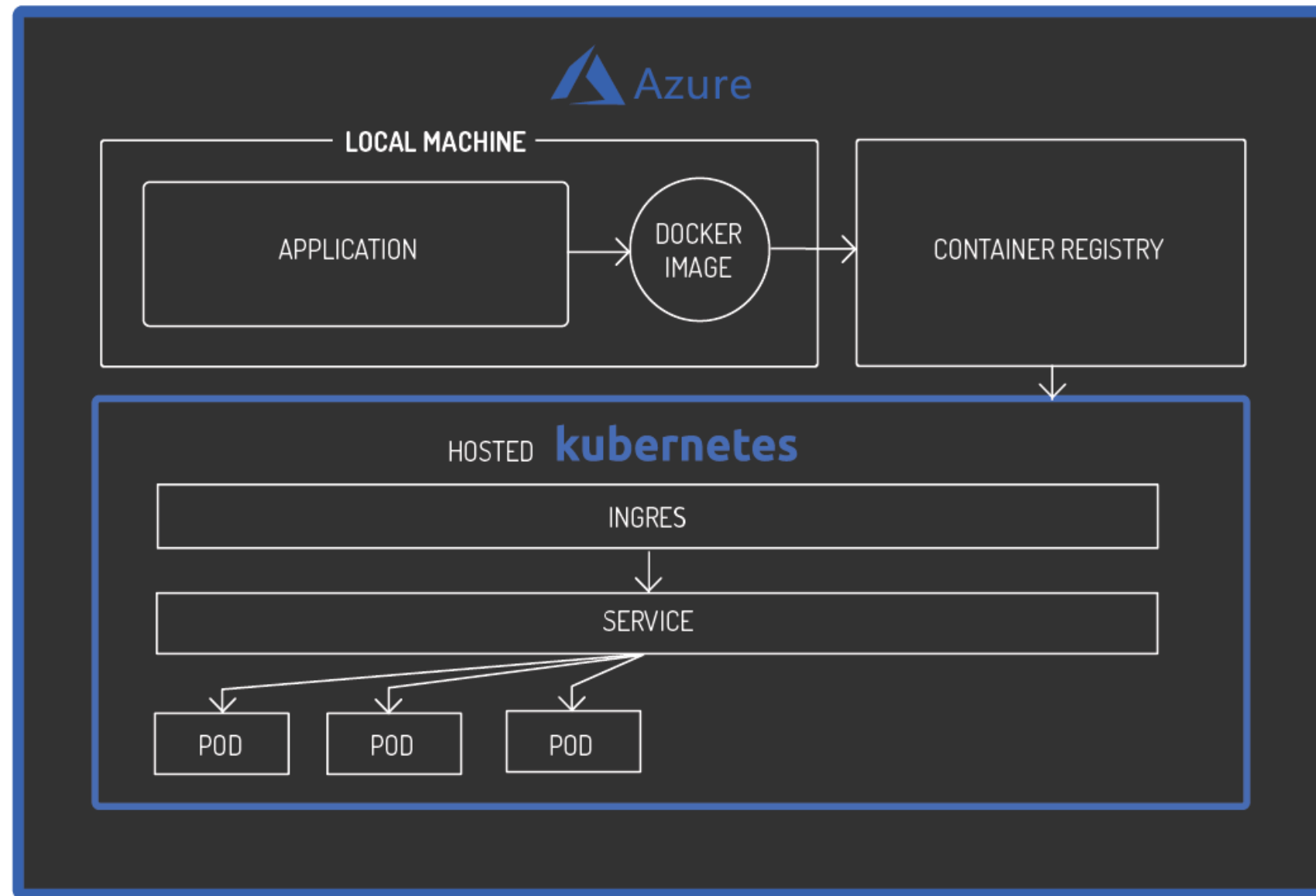
Scheduling Pods

- Kubernetes uses the concepts of taints and tolerances to schedule the pods and services.

- A taint is applied to a node to indicate only specific pods can be scheduled on them.

- A toleration is applied to a pod, allowing them to tolerate a node's taint.

Storage in AKS Cluster

- To implement a volume in a Kubernetes cluster, you need to define a PersistentVolumeClaim for a specific storage class.

- A storage class represents the characteristics of the underlying storage, such as performance or support for shared access.

- PersistentVolumeClaim includes information about the required access mode and volume size.

- Azure has its own storage volume types called Azure Disk and Azure files.

Create a Kubernetes Cluster using AKS

Problem Statement:

Set up the prerequisites and create a Kubernetes cluster using AKS.

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to Create a Kubernetes Cluster using AKS:

1. Setting up the prerequisites for configuring an AKS cluster

2. Creating a Kubernetes cluster using AKS service

Create Workloads in an AKS Cluster

Problem Statement:

Create workloads such as Pods and Deployments in the AKS cluster.

Assisted Practice: Guidelines

Steps to Create Workloads in an AKS Cluster:

1. Creating a namespace

2. Creating a pod

3. Creating a deployment

Problem Statement:

Rollout an update for a deployment.

Assisted Practice: Guidelines

Steps to Rolling out an Update for a Deployment:

1. Setting up a storage account resource

2. Creating a new file share for the storage account

3. Setting up Azure Cloud Shell

4. Creating a deployment and rolling out an update for it

Expose a Deployment as a Service

Problem Statement:

Expose a deployment as a service.

Assisted Practice: Guidelines

Steps to Expose a Deployment as a Service:

1. Connecting Azure Cloud Shell to the Kubernetes cluster

2. Exposing a deployment as a service

Create a Persistent Storage in an AKS Cluster

Problem Statement:

Create a Persistent Storage in an AKS Cluster.

Assisted Practice: Guidelines

Steps to Create a Persistent Storage in an AKS Cluster:

1. Creating a PersistentVolume

2. Creating a PersistentVolumeClaim bound to the PersistentVolume

Key Takeaways

◉ Azure is a cloud platform created by Microsoft for building, testing, deploying, and managing applications.

◉ Azure Kubernetes Service (AKS) manages the hosted Kubernetes environment and makes it simple to deploy and manage containerized applications in Azure.

◉ Clusters can be created using the Azure portal or Azure CLI.

◉ Kubernetes resources are logically grouped into a namespace to divide an AKS cluster and restrict create, view, or manage access to the resources.

simplilearn