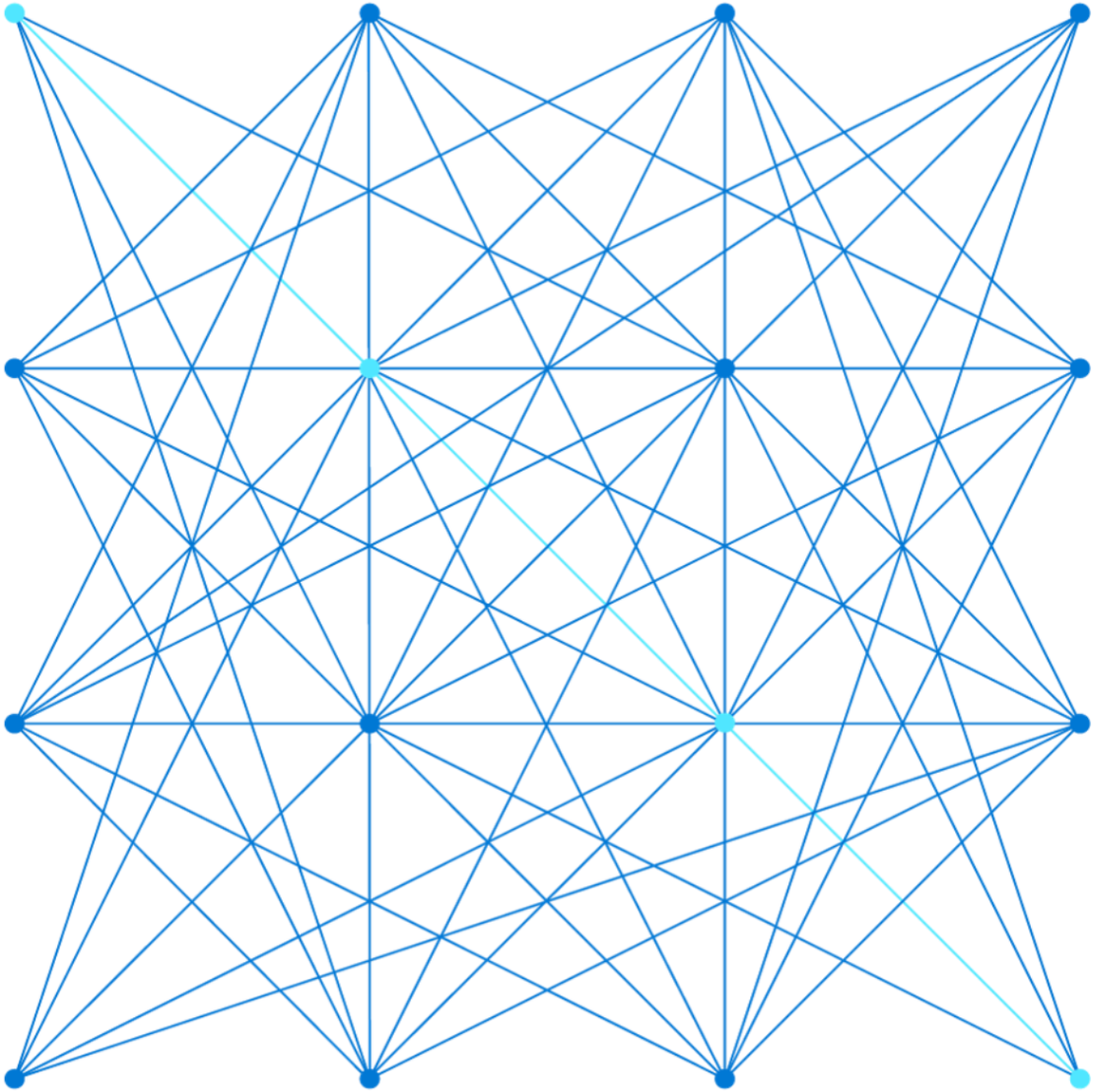
Microsoft Azure

AZ-104

Tag 4

Administer PaaS Compute Options

Guten Morgen!



About this course: Course Outline

01: Administer Identity

02: Administer Governance and Compliance

03: Administer Azure Resources

04: Administer Virtual Networking

05: Administer Intersite Connectivity

06: Administer Network Traffic Management

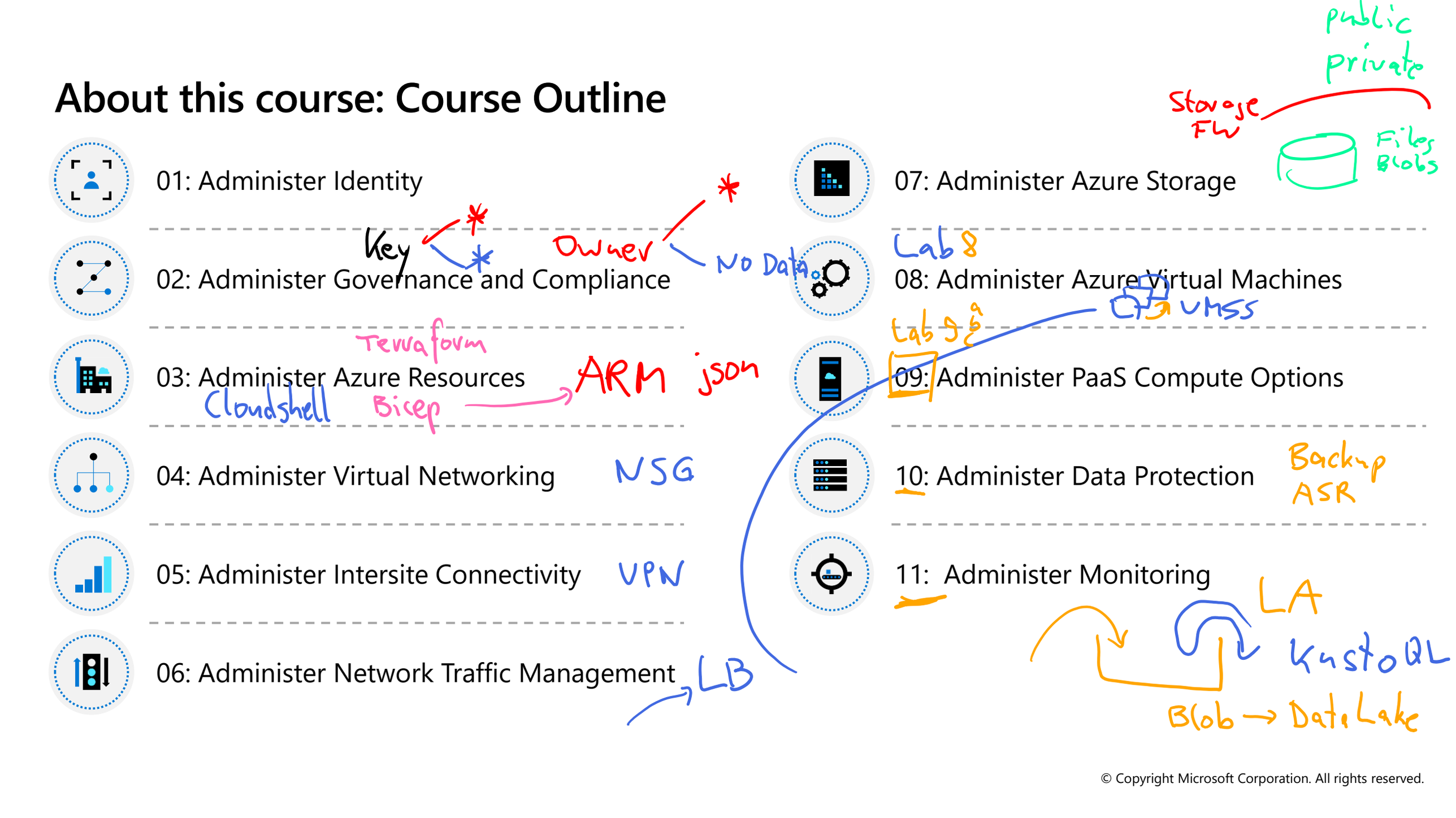
07: Administer Azure Storage

08: Administer Azure Virtual Machines

09: Administer PaaS Compute Options

10: Administer Data Protection

11: Administer Monitoring



Administer PaaS Compute Options Introduction

Configure Azure App Service Plans

Configure Azure App Services

Configure Azure Container Instances

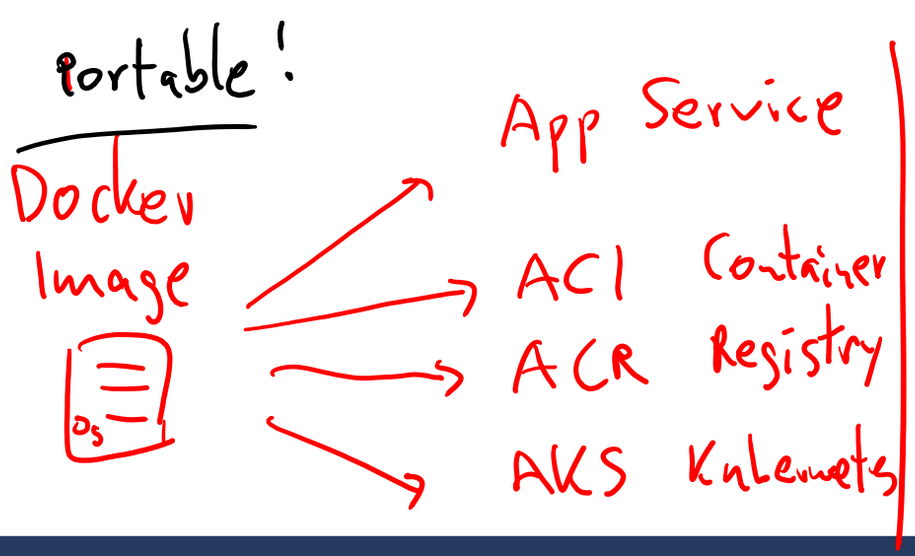
Configure Azure Kubernetes Service

Lab 09a - Implement Web Apps

Lab 09b - Implement Azure Container Instances

Lab 09c - Implement Azure Kubernetes Service (optional)

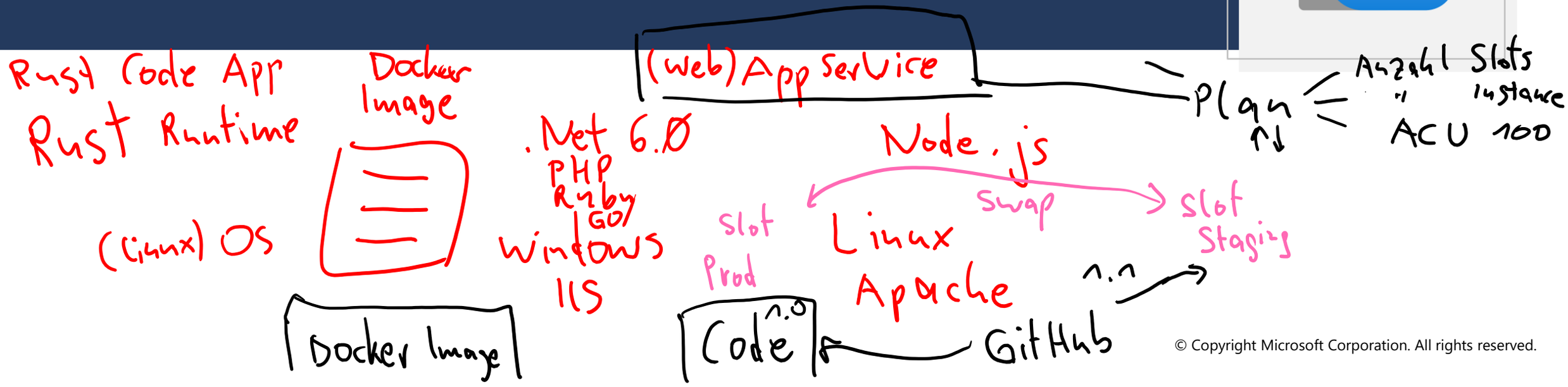
Configure Azure App Service Plans



Tim Berners-Lee

www

404 Cern



Configure Azure App Service Plans Introduction

Implement Azure App Service Plans

Determine App Service Plan Pricing

Scale Up and Scale Out the App Service Plan

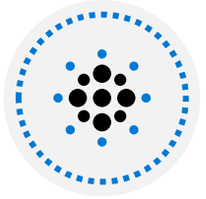
Configure App Service Plan Scaling

Demonstration – Create an App Service Plan

Summary and Resources

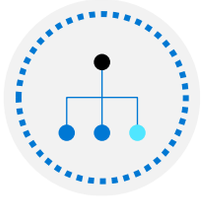
Implement Azure App Service Plans

Define a set of compute resources for a web app to run



Determines performance, price, and features

One or more apps can be configured to run in the same App Service plan



Region where compute resources will be created

Number of virtual machine instances

Size of virtual machine instances

Pricing tier (next slide)

Determine App Service Plan Pricing

Selected Features	Free	Shared (dev/test)	Basic (dedicated dev/test)	Standard (production workloads)	Premium (enhanced scale and performance)	Isolated (high-performance, security and isolation)
Web, mobile, or API apps	10	100	Unlimited	Unlimited	Unlimited	Unlimited
Disk space	1 GB	1 GB	10 GB	50 GB	250 GB	1 TB
Auto Scale	–	–	–	Supported	Supported	Supported
Deployment Slots	0	0	0	5	20	20
Max Instances	–	–	Up to 3	Up to 10	Up to 30	Up to 100

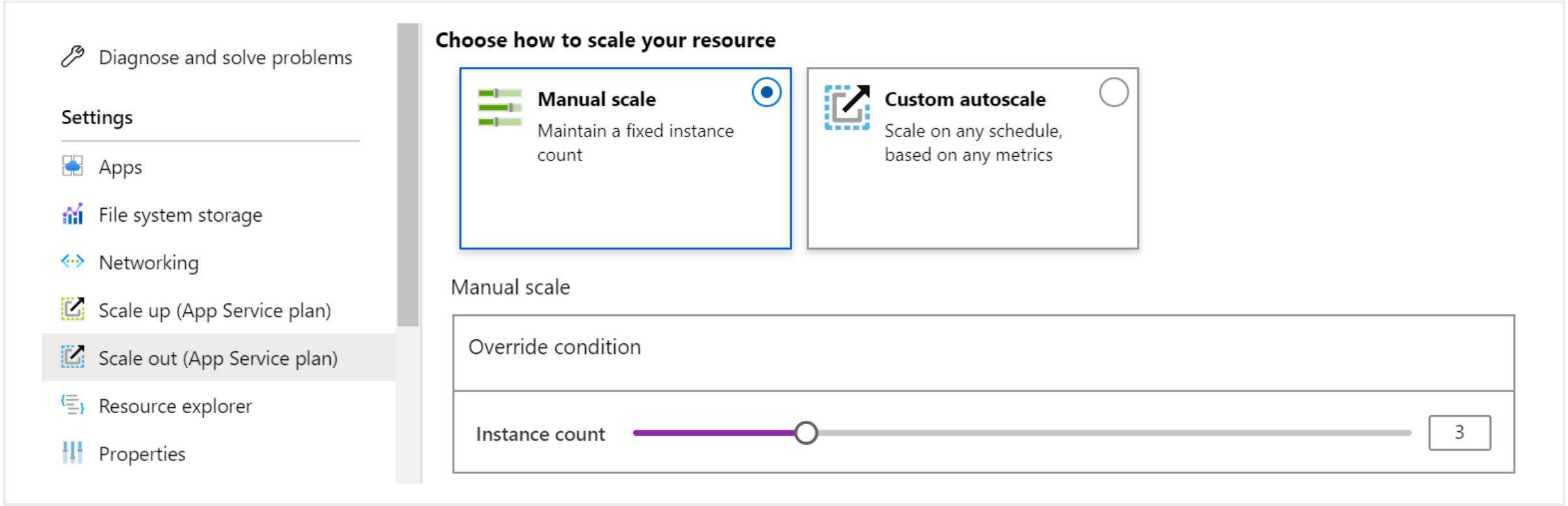
60min/24h

Shared compute (Free and Shared). Run apps on the same Azure VM as other App Service apps, and the resources cannot scale out

Dedicated compute (Basic, Standard, Premium). Run apps in the same plan in dedicated Azure VMs

Isolated. Runs apps on dedicated Azure VMs in dedicated Azure virtual networks

Scale Up and Scale Out the App Service Plan



Scale up (change the App Service plan):

More hardware (CPU, memory, disk)

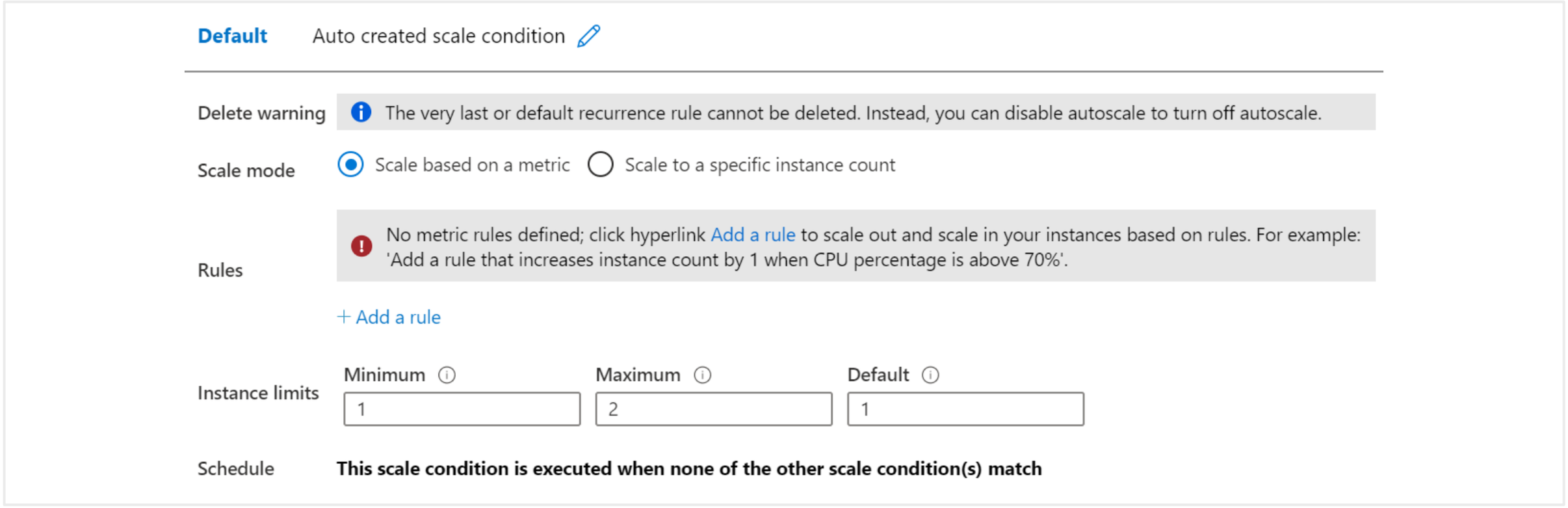
More features (dedicated virtual machines, staging slots, autoscaling)

Scale out (increase the number of VM instances):

Manual (fixed number of instances)

Auto scale (based on predefined rules and schedules)

Configure App Service Plan Scaling



Adjust available resources based on the current demand

Improves availability and fault tolerance

Scale based on a metric (CPU percentage, memory percentage, HTTP requests)

Scale according to a schedule (weekdays, weekends, times, holidays)

Can implement multiple rules – combine metrics and schedules

Don't forget to scale in

Demonstration – Create an App Service plan

Create an App Service Plan in the Azure Portal

Review Pricing Tiers

Configure Autoscaling

Summary and Resources – Configure Azure App Service Plans

Knowledge Check Questions

Microsoft Learn Modules (docs.microsoft.com/Learn)

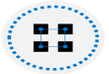
Scale an App Service web app to efficiently meet demand with App Service scale up and scale out

Configure Azure App Services

Configure Azure App Services Introduction


Implement Azure App Service


Create an App Service


Create Deployment Slots


Add Deployment Slots


Secure an App Service


Create Custom Domain Names


Backup an App Service


Demonstration – Create an App Service


Summary and Resources

Implement Azure App Service

Includes Web Apps, API Apps, Mobile Apps, and Function Apps

Fully managed environment enabling high productivity development

Platform-as-a-service (PaaS) offering for building and deploying highly available cloud apps for web and mobile

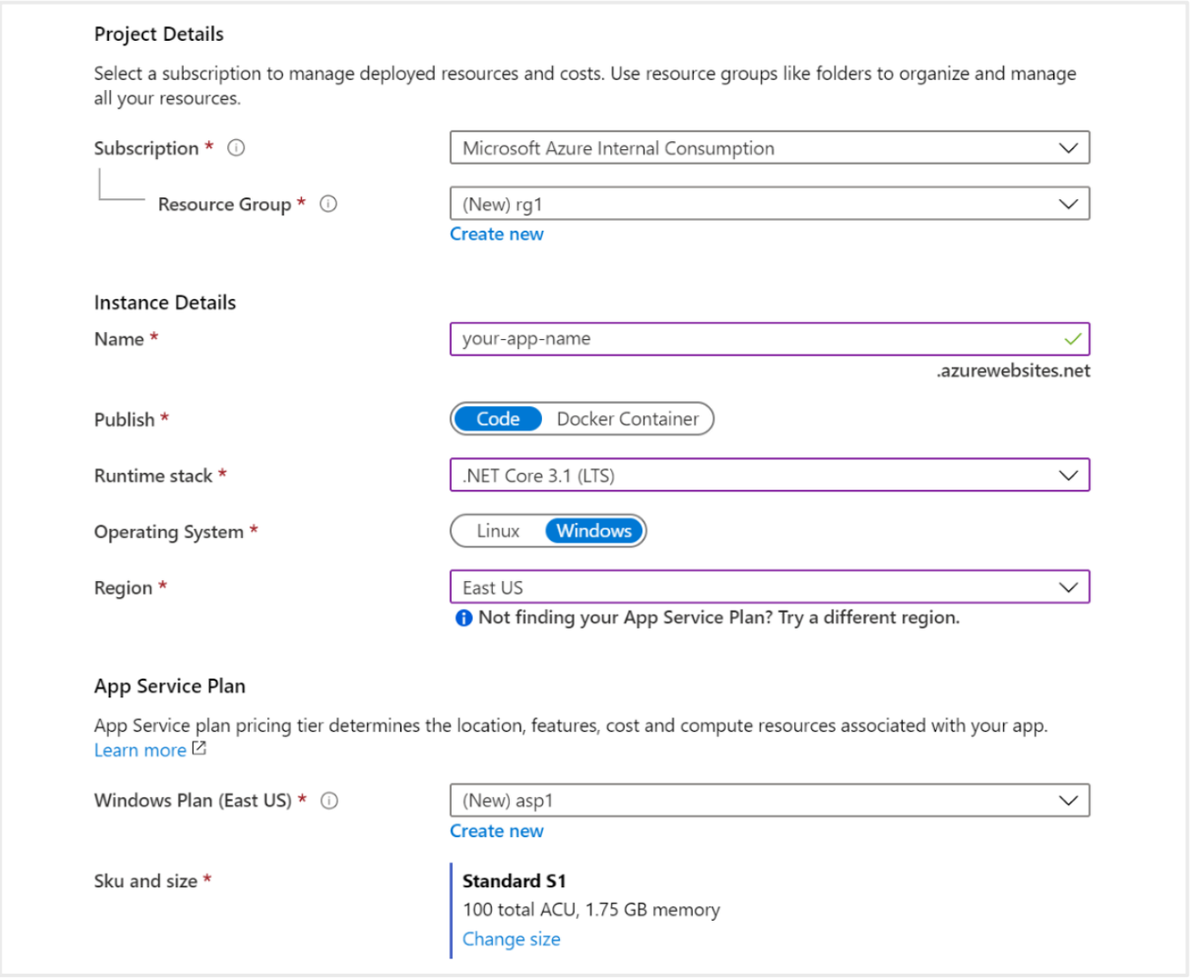
Platform handles infrastructure so developers focus on core web apps and services

Developer productivity using .NET, .NET Core, Java, Python and a host of others

Provides enterprise-grade security and compliance

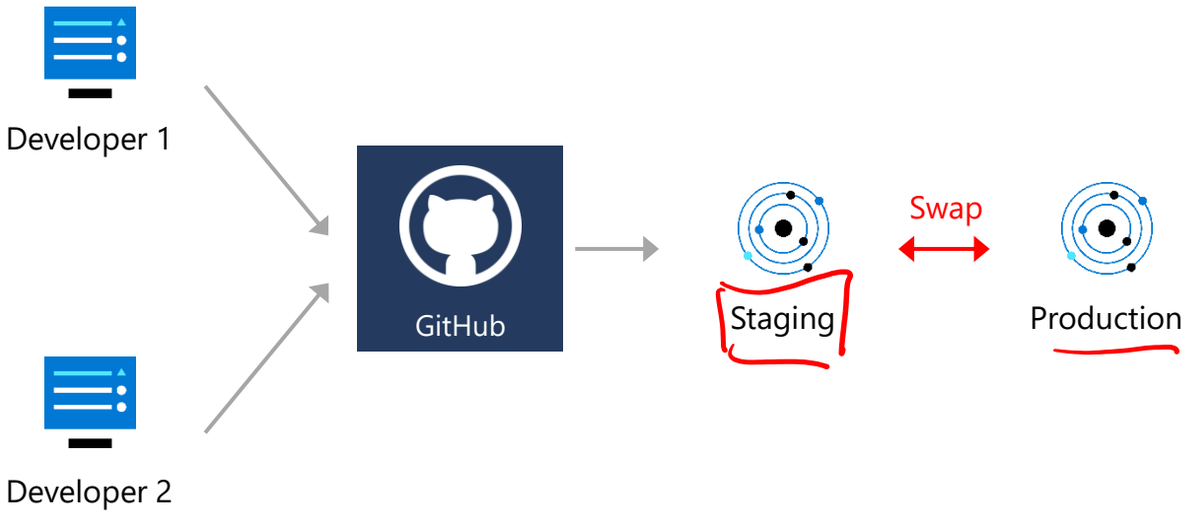
Create an App Service

- Name must be unique
- Access using *azurewebsites.net* – can map to a custom domain
- Publish Code (Runtime Stack)
- Publish Docker Container
- Linux or Windows
- Region closest to your users
- App Service Plan



Create Deployment Slots

Continuous Deployment with Stage Slot



Service Plan	Slots
Free, Shared, Basic	0
Standard	Up to 5
Premium	Up to 20
Isolated	Up to 20

Deploy to a different deployment slots (depends on service plan)

Validate changes before sending to production

Deployment slots are live apps with their own hostnames

Avoids a cold start – eliminates downtime

Fallback to a last known good site

Auto Swap when pre-swap validation is not needed

Add Deployment Slots

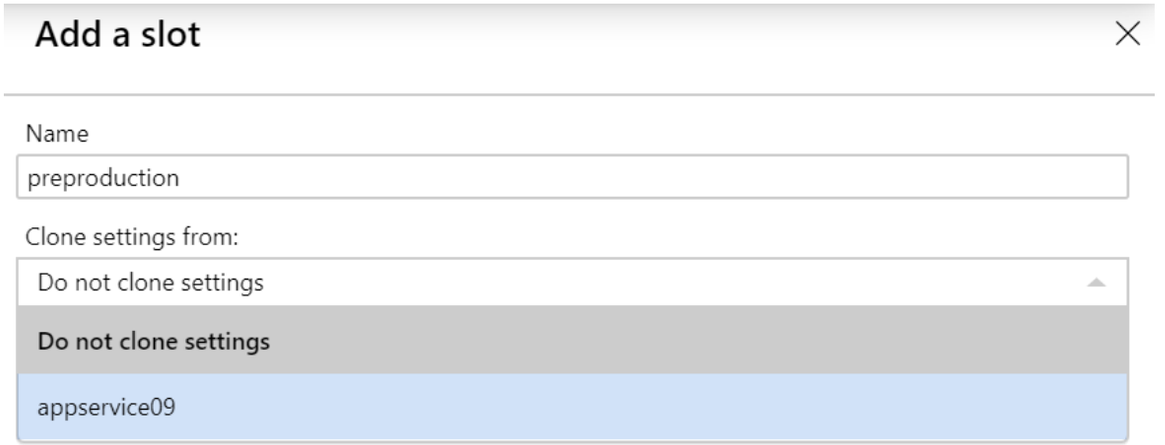
Select whether to clone an app configuration from another deployment slot

When you clone, pay attention to the settings:

- Slot-specific app settings and connection strings
- Continuous deployment settings
- App Service authentication settings

Not all settings are sticky (endpoints, custom domain names, SSL certificates, scaling)

Review and edit your settings before swapping



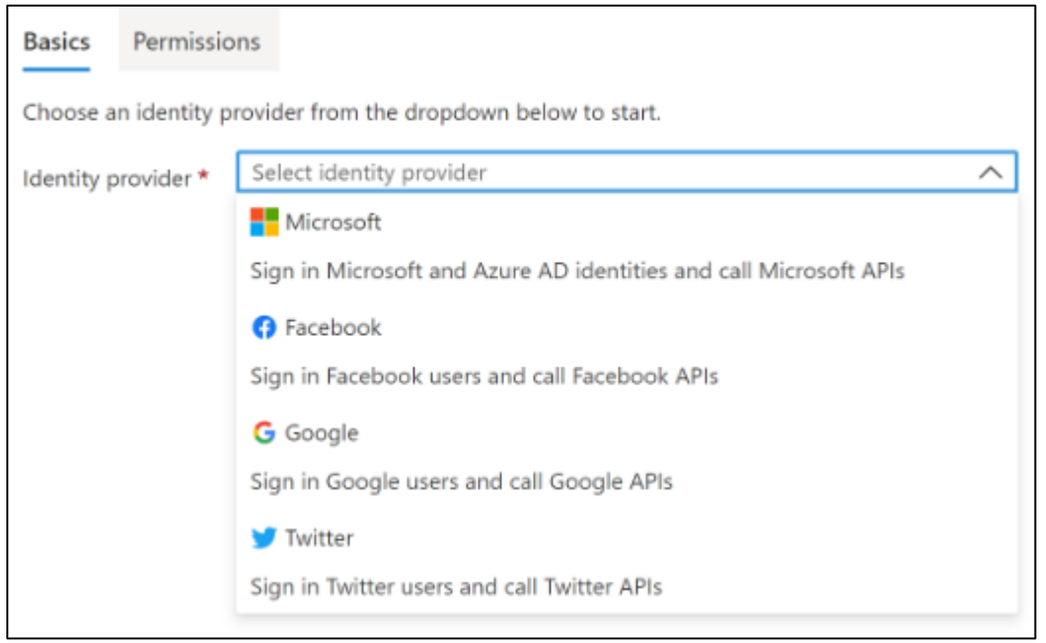
Secure an App Service

Authentication:

- Enable authentication – default anonymous
- Log in with a third-party identity provider

Security:

- Troubleshoot with Diagnostic Logs – failed requests, app logging
- Add an SSL certificate – HTTPS
- Define a priority ordered allow/deny list to control network access to the app
- Store secrets in the Azure Key Vault



https:// www. . azurewebsites.net

Create Custom Domain Names

Redirect the default web app URL

Validate the custom domain in Azure

Use the DNS registry for your domain provider – create a CNAME or A record with the mapping

Ensure App Service plan supports custom domains

Backup an App Service

Create app backups manually or on a schedule

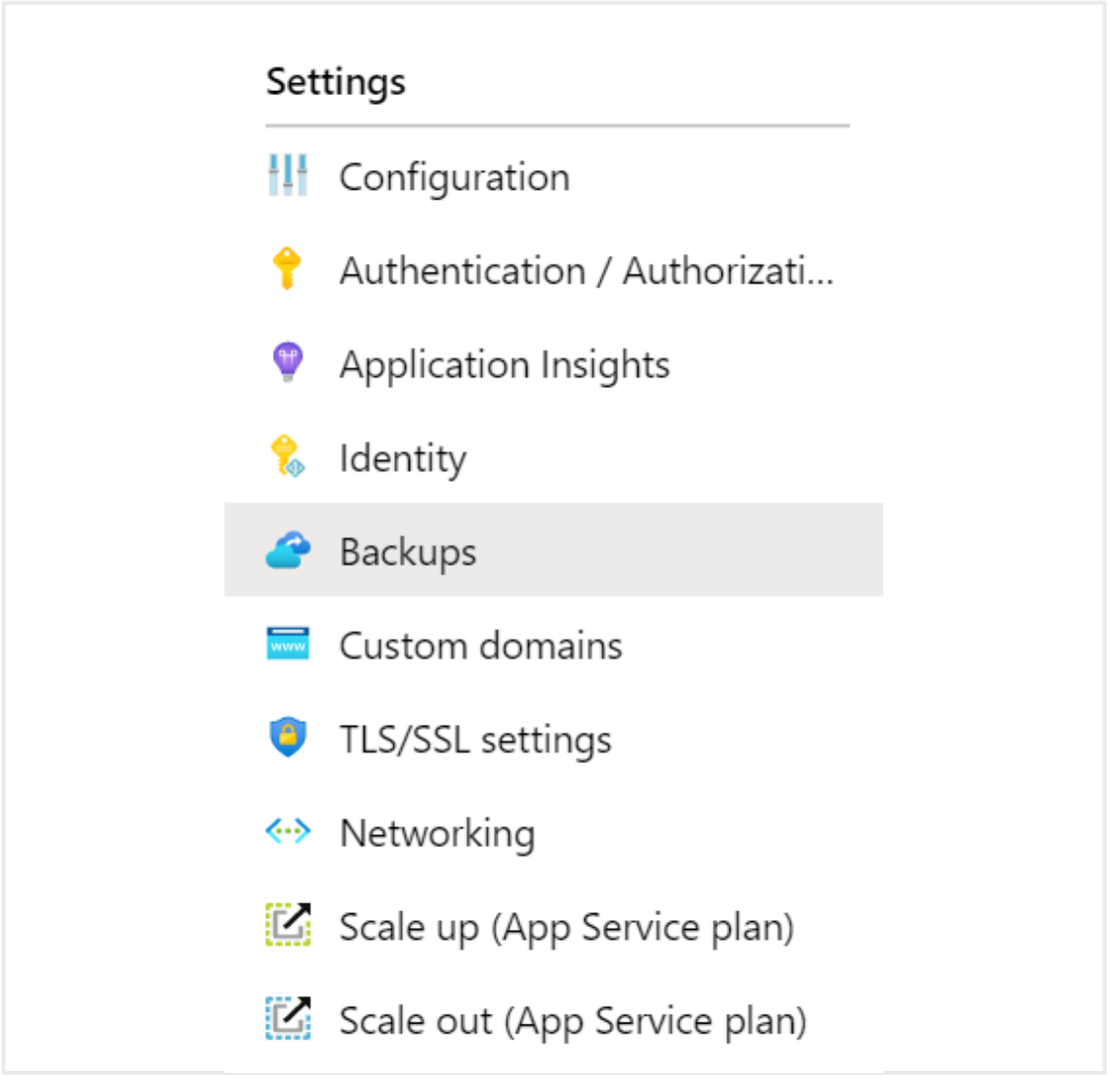
Backup the configuration, file content, and database connected to the app

Requires Standard or Premium plan

Backups can be up to 10 GB of app and database content

Configure partial backups and exclude items from the backup

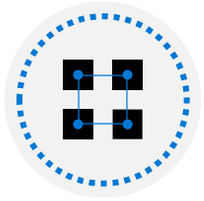
Restore your app on-demand to a previous state, or create a new app



Demonstration – Create an App Service

Create a Web App in the Azure Portal

Test the Web App



Configure Deployment Slots

Configure Backup

Summary and Resources – Configure Azure App Services

Knowledge Check Questions

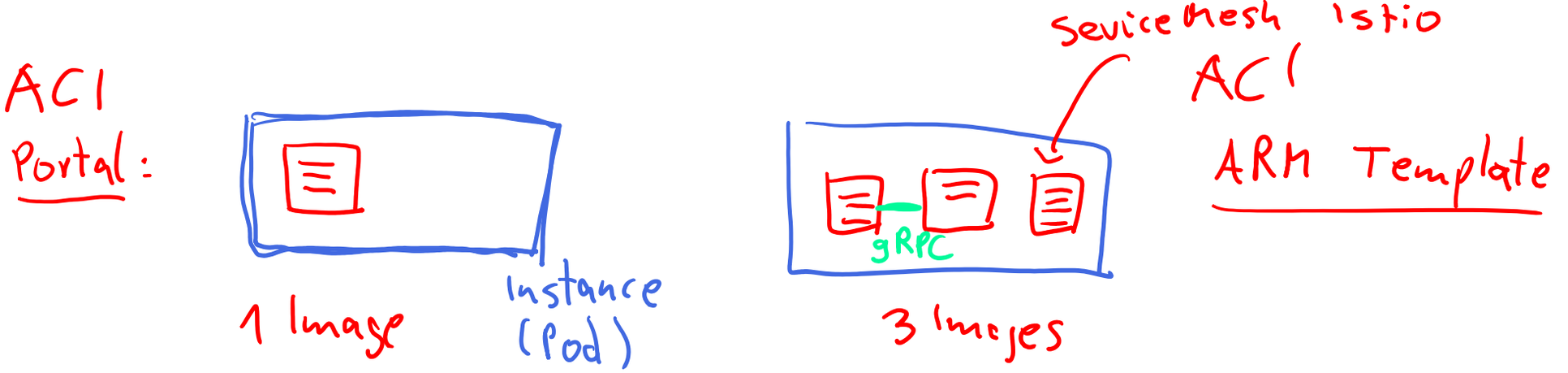
Microsoft Learn Modules (docs.microsoft.com/Learn)

Host a web application with Azure App Service (Sandbox)

Stage a web app deployment for testing and rollback by using App Service deployment slots

Dynamically meet changing web app performance requirements with autoscale rules

A *sandbox* indicates a hands-on exercise.



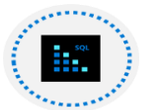
Configure Azure Container Instances

Configure Azure Container Instances Introduction

Compare Containers to Virtual Machines

Explore Azure Container Instances Benefits

Implement Container Groups



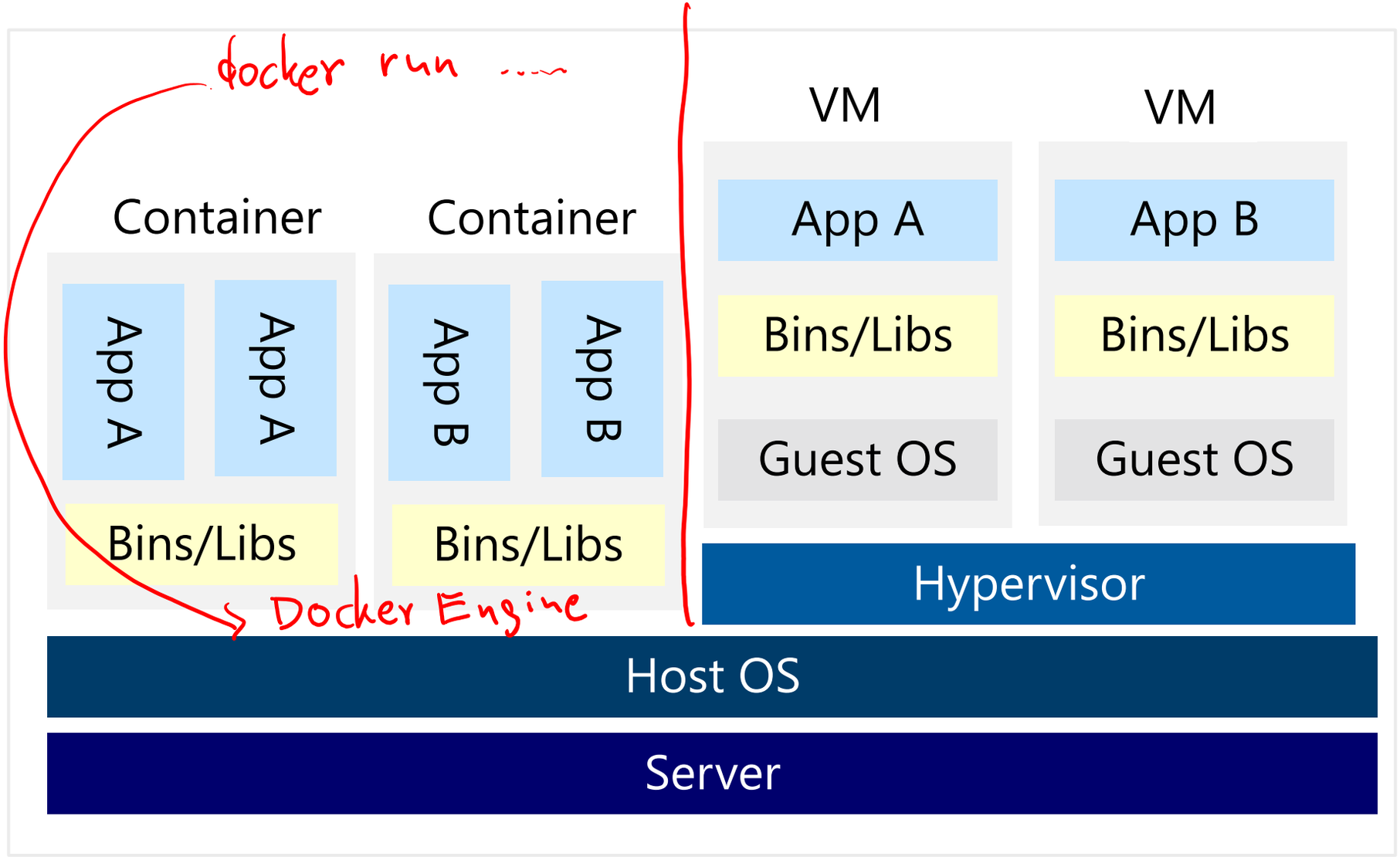
Understand the Docker Platform

Demonstration – Deploy Azure Container Instances

Summary and Resources

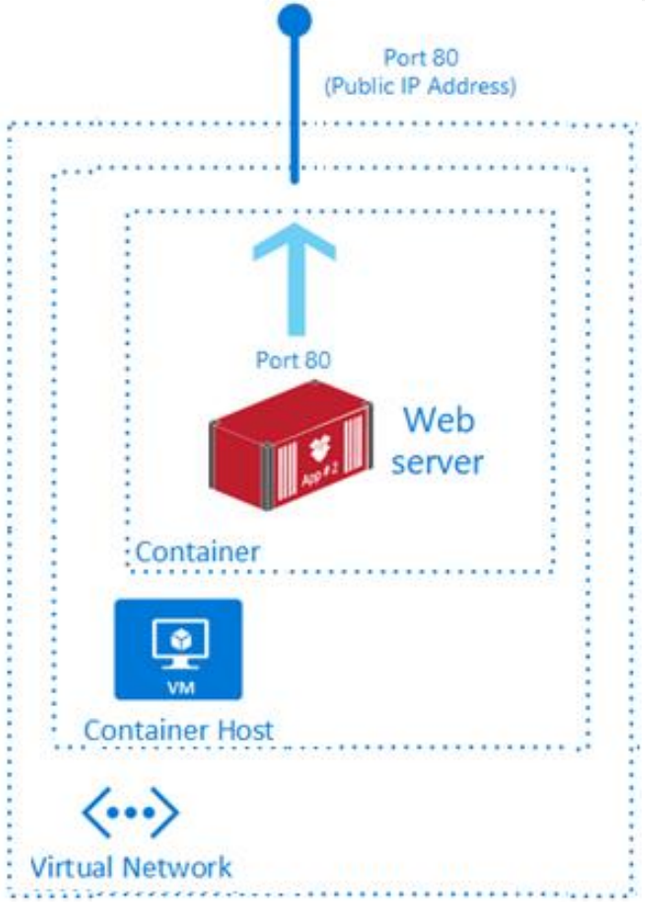
Compare Containers to Virtual Machines

- Isolation
- Operating System
- Deployment
- Persistent storage
- Fault tolerance



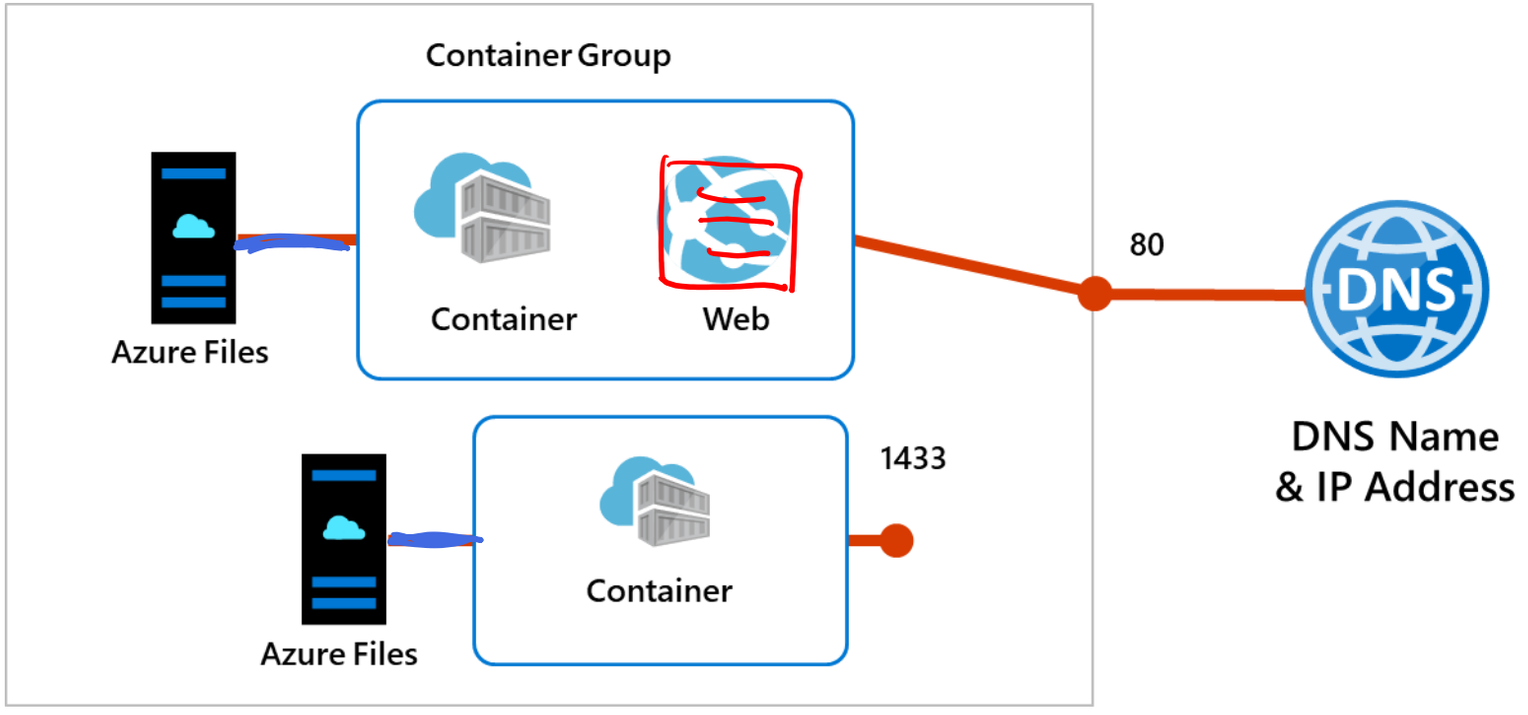
Explore Azure Container Instances Benefits

- PaaS Service
- Fast startup times
- Public IP connectivity and DNS name
- Isolation features
- Custom sizes
- Persistent storage
- Linux and Windows Containers
- Co-scheduled Groups
- Virtual network Deployment



Fastest way to run a container in Azure without provisioning a VM

Implement Container Groups

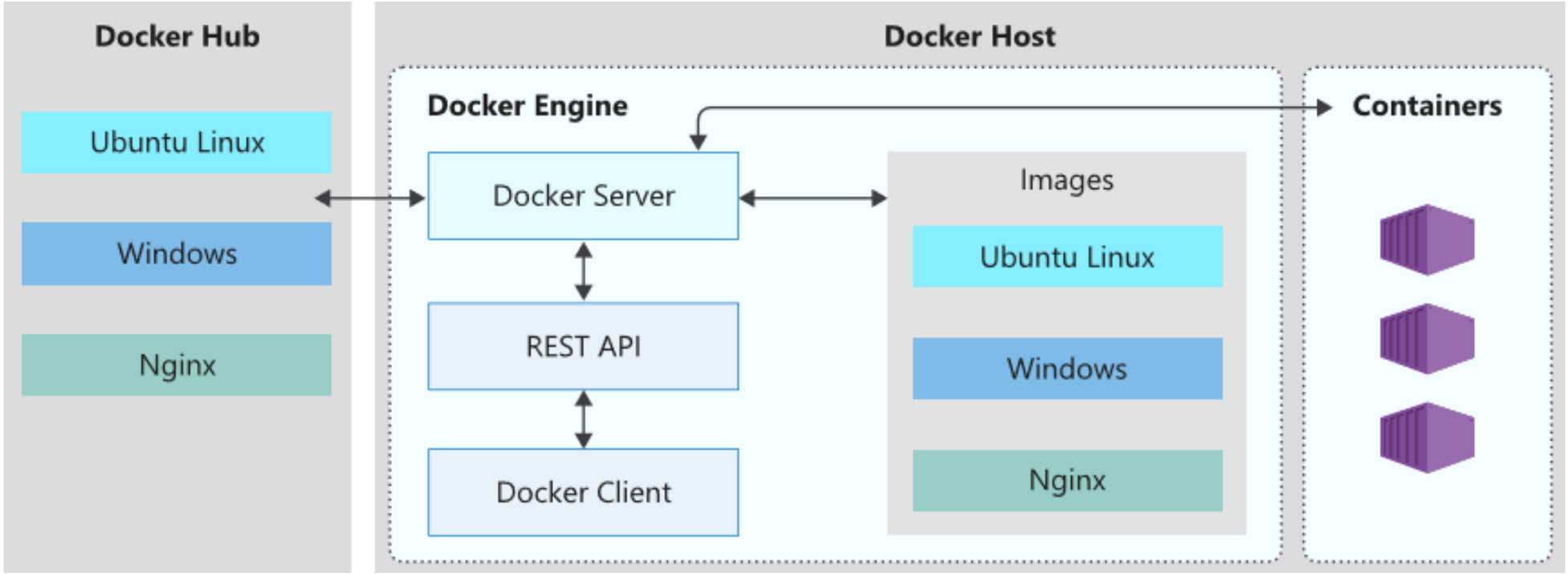


Top-level resource in Azure Container Instances

A collection of containers that get scheduled on the same host

The containers in the group share a lifecycle, resources, local network, and storage volumes

Understand the Docker Platform



Enables developers to host applications within a container

A container is a standardized “unit of software” that contains everything required for an application to run

Available on both Linux and Windows and can be hosted on Azure

Demonstration - Deploy Azure Container Instances

Create a container instance

Verify deployment of the container instance

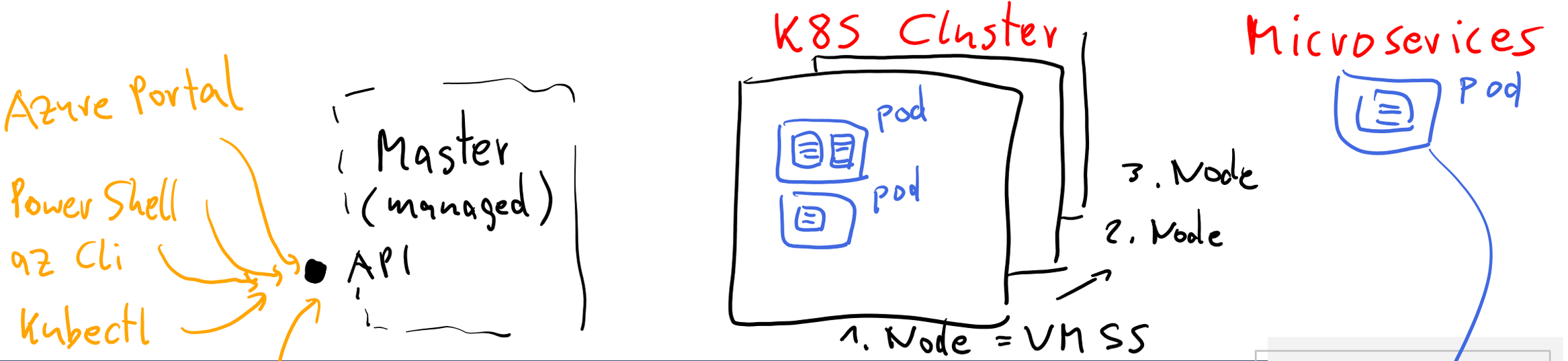
Summary and Resources – Configure Azure Container Instances

Knowledge Check Questions

Microsoft Learn Modules (docs.microsoft.com/Learn)

Run Docker containers with Azure Container Instances

Build a containerized web application with Docker



Configure Azure Kubernetes Service

Configure Azure Kubernetes Service Introduction


Understand AKS Terminology


Understand AKS Clusters and Nodes


Configure AKS Networking


Configure AKS Storage


Configure AKS Scaling


Configure AKS Scaling to ACI (optional)


Demonstration – Deploy Azure Kubernetes Service (optional)

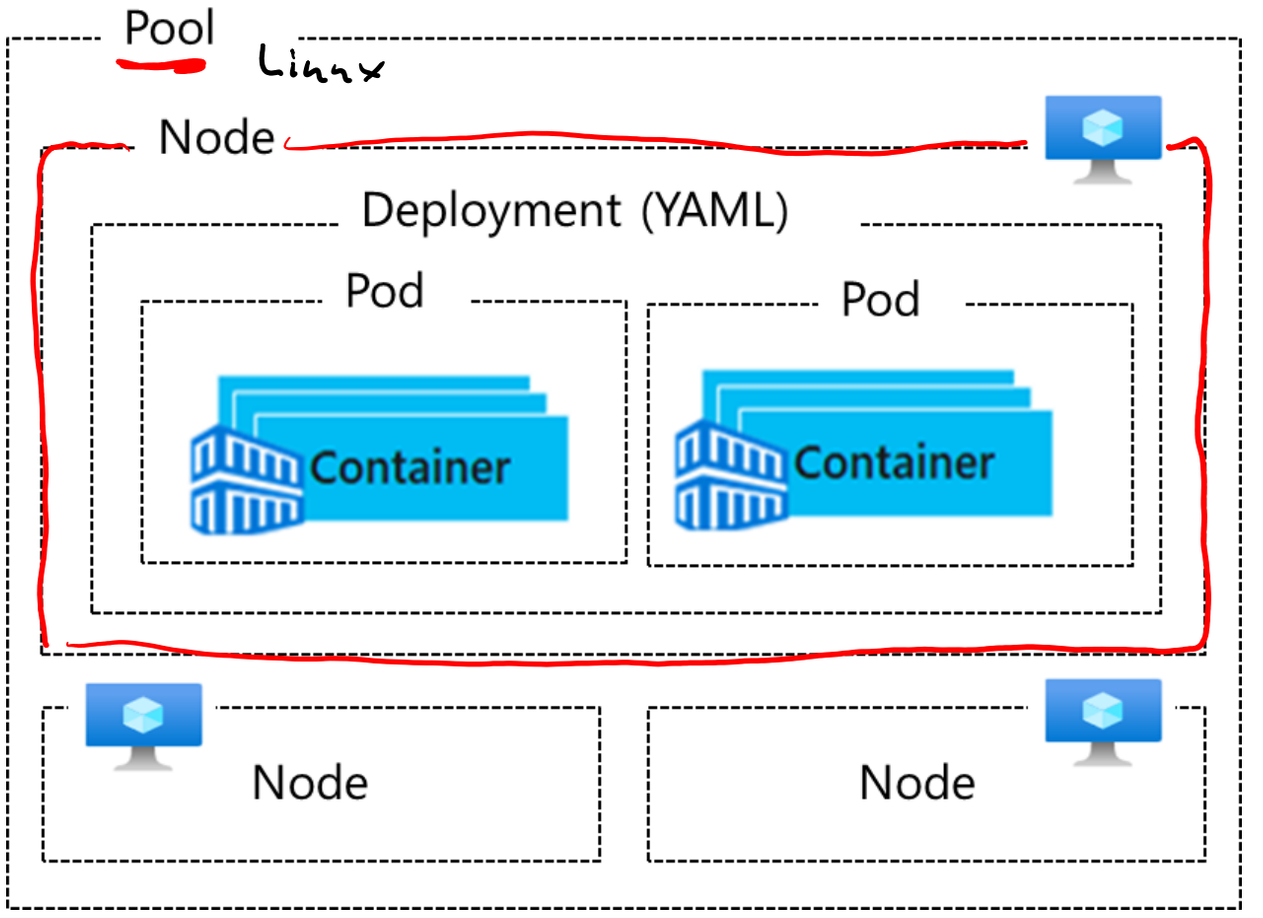

Summary and Resources

Brendan Burns

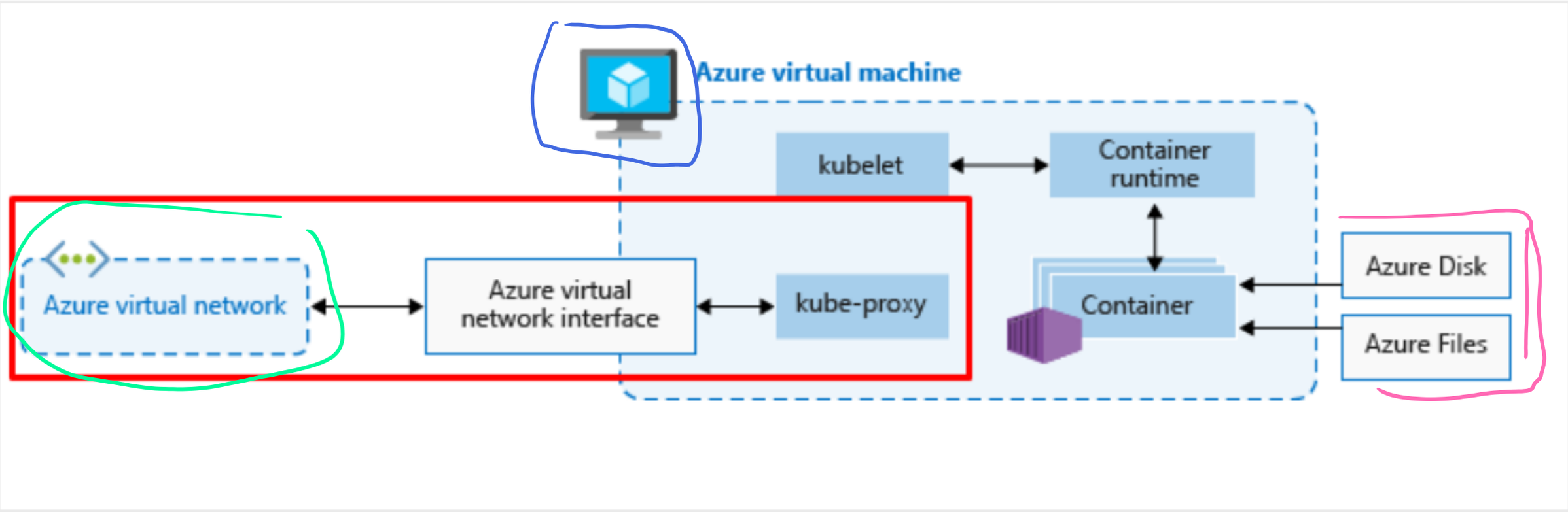
Pool 2 windows

Understand AKS Terminology

Term	Description
Pools	Groups of nodes with identical configurations
Nodes	Individual VMs running containerized applications
Pods	Single instance of an application. A pod can contain multiple containers
Deployment	One or more identical pods managed by Kubernetes
Manifest	YAML file describing a deployment



Understand AKS Clusters and Nodes

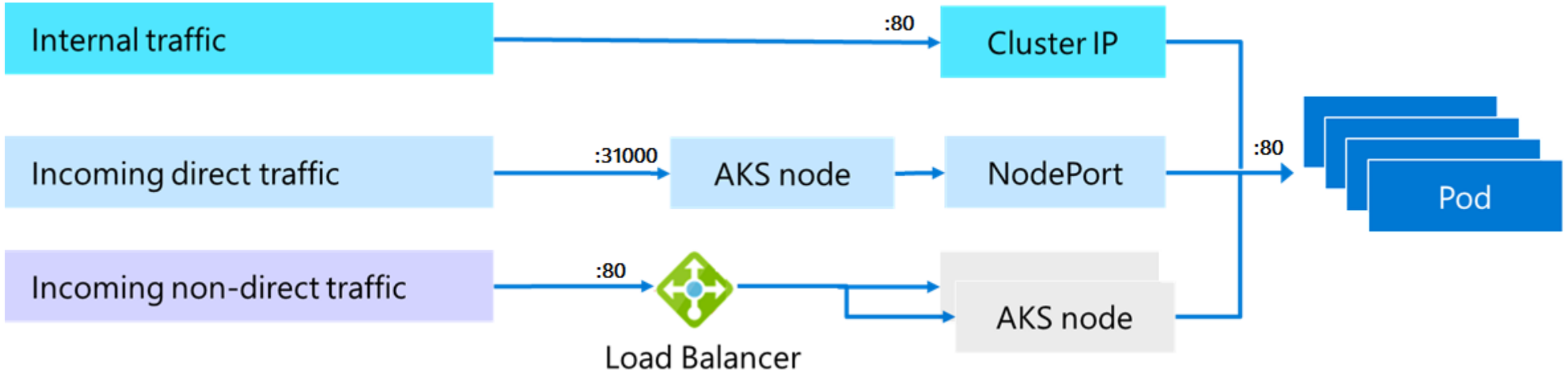


Azure-managed node provides core Kubernetes services and orchestration

Customer-managed nodes run applications and supporting services

Each individual node is an Azure virtual machine

Configure AKS Networking



Pods run an instance of your application

Services group pods together to provide network connectivity

ClusterIP provides internal traffic access

NodePort provides mapping for incoming direct traffic

LoadBalancer has external IP address for incoming non-direct traffic

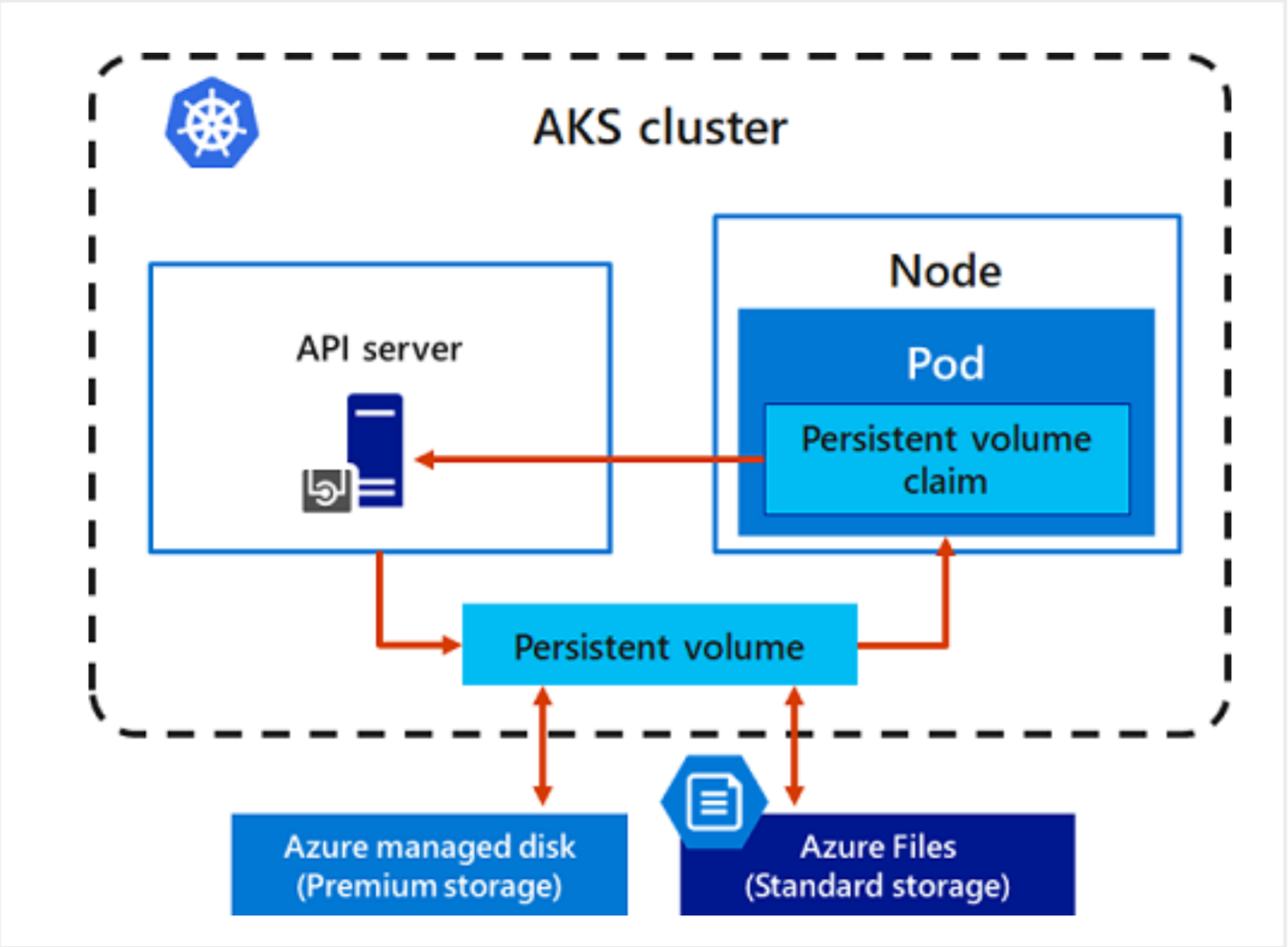
Configure AKS Storage

Local storage on the node is fast and simple to use

Local storage might not be available after the pod is deleted

Multiple pods may share data volumes

Storage could potentially be reattached to another pod



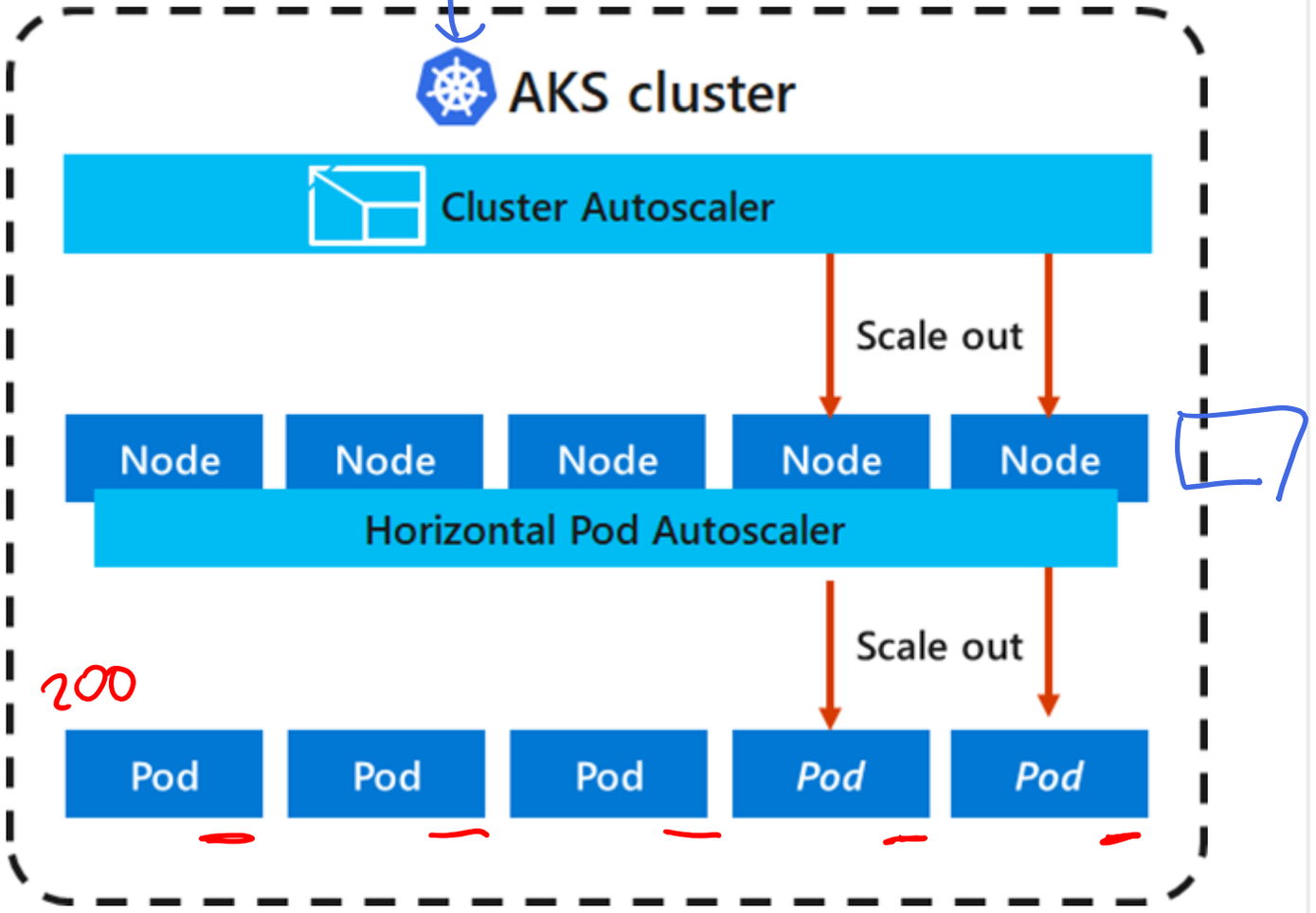
Configure AKS Scaling

Applications might grow beyond the capacity of a single pod

Kubernetes has built-in autoscalers

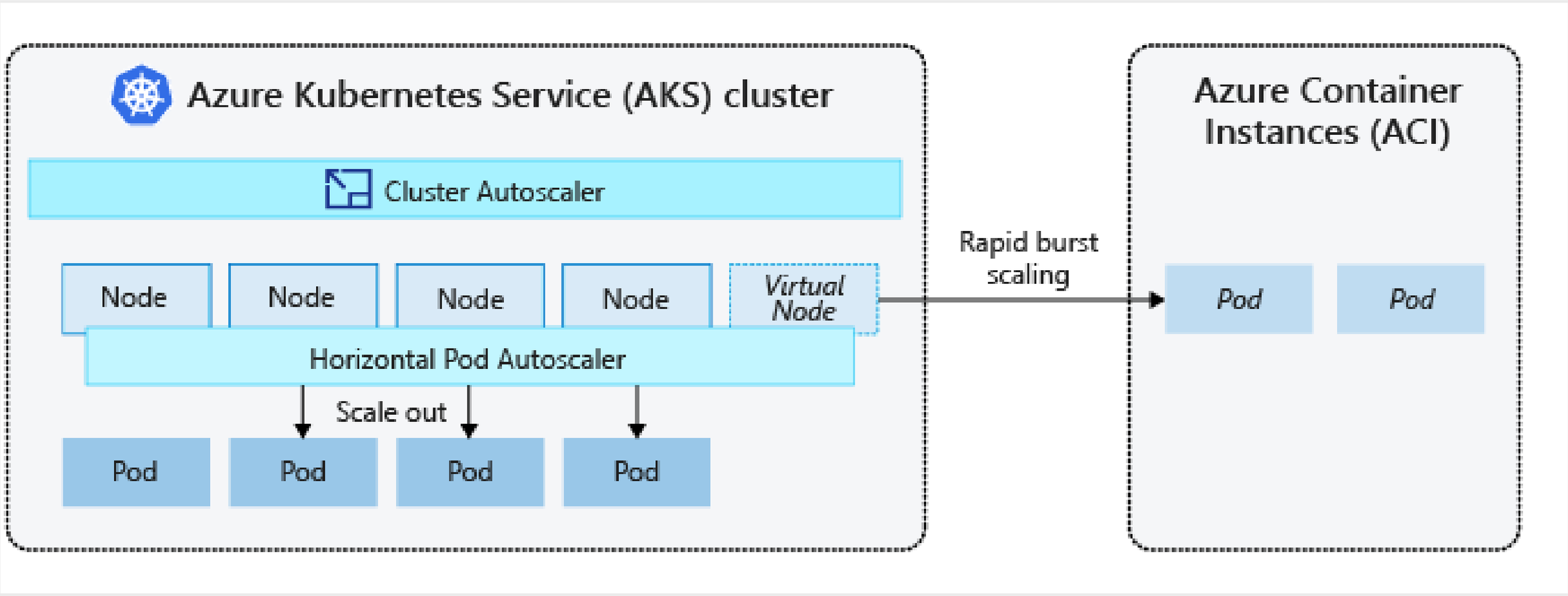
Cluster autoscaler scales based on compute resources

Horizontal pod autoscaler scales based on metrics



Configure AKS Scaling to ACI (optional)

If you need to rapidly grow your AKS cluster, you can create new pods in Azure Container Instances

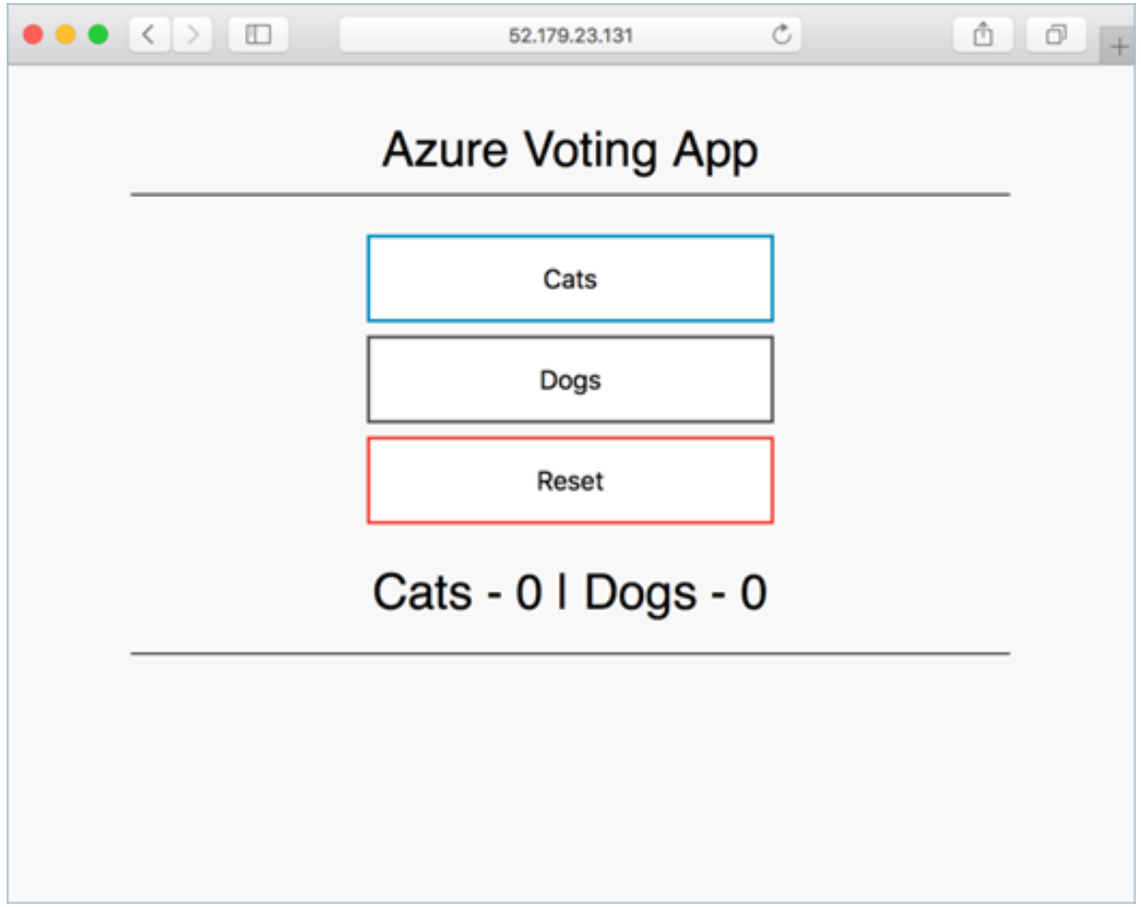


Demonstration – Deploy Azure Kubernetes Service (optional)

Create a Kubernetes service

Connect to the cluster

Test the applications



Summary and Resources – Configure Azure Kubernetes Service

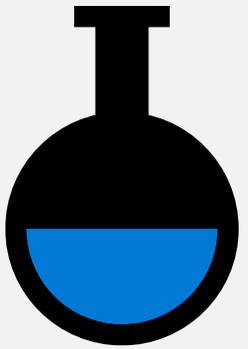
Knowledge Check Questions

Microsoft Learn Modules (docs.microsoft.com/Learn)

Introduction to Azure Kubernetes Service

Implement Azure Kubernetes Service (AKS)

Lab 09a - Implement Web Apps
Lab 09b - Implement Azure Container Instances
Lab 09c - Implement Azure Kubernetes Service (optional)



Lab 09a – Implement web apps

Lab scenario

You need to evaluate the use of Azure Web apps for hosting Contoso's web sites, hosted currently in the company's on-premises data centers. The web sites are running on Windows servers using PHP runtime stack. You also need to determine how you can implement DevOps practices by leveraging Azure web apps deployment slots

Objectives

Task 1:
Create an Azure web app

Task 2:
Create a staging deployment slot

Task 3:
Configure web app deployment settings

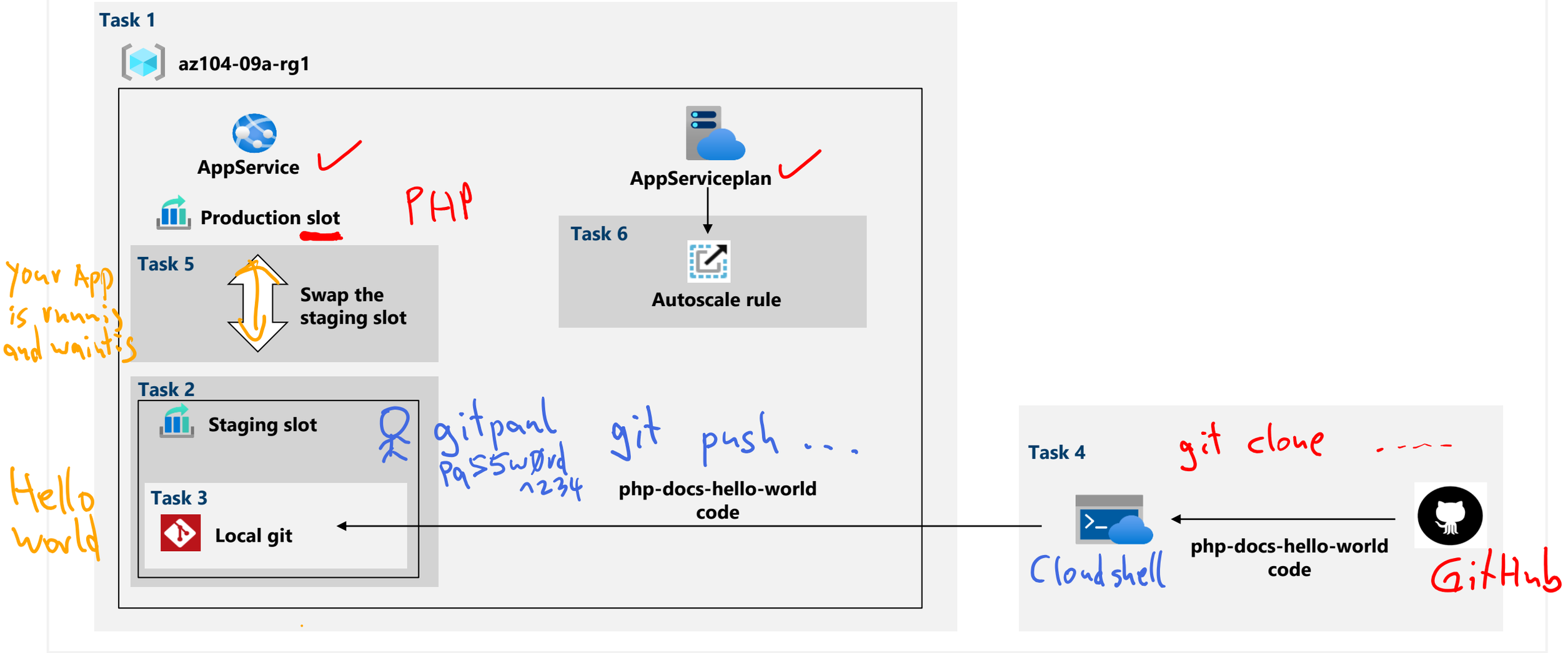
Task 4:
Deploy code to the staging deployment slot

Task 5:
Swap the staging slots

Task 6:
Configure and test autoscaling of the Azure web app

Next slide for an architecture diagram

Lab 09a – Architecture diagram



Lab 09b – Implement Azure Container Instances

Lab scenario

Contoso wants to find a new platform for its virtualized workloads. You identified several container images that can be leveraged to accomplish this objective. Since you want to minimize container management, you plan to evaluate the use of Azure Container Instances for deployment of Docker images

Objectives

Task 1:

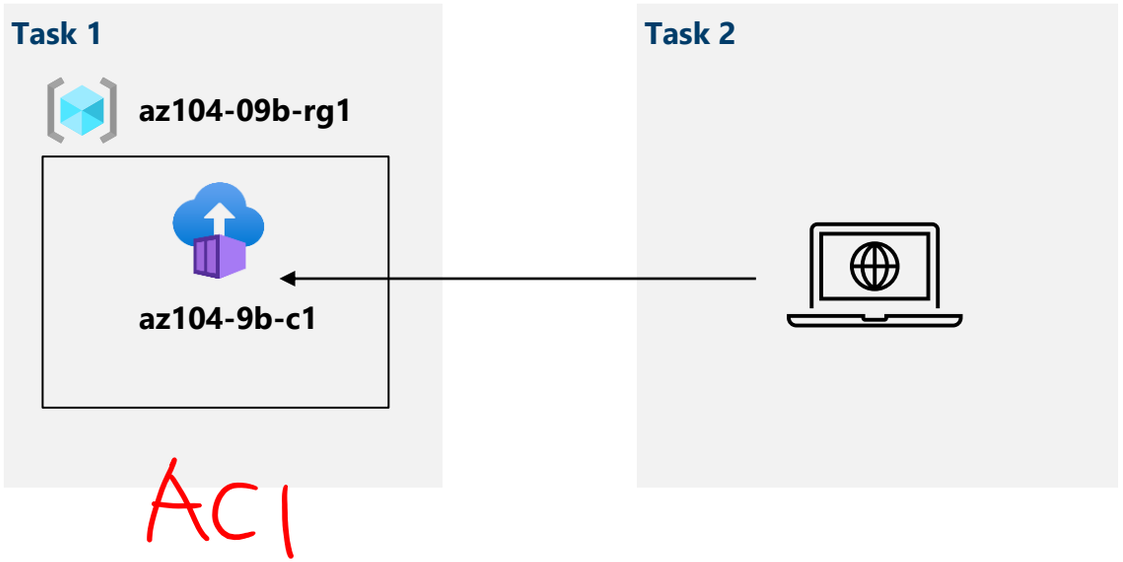
Deploy a Docker image by using the Azure Container Instance

Task 2:

Review the functionality of the Azure Container Instance

Next slide for an architecture diagram

Lab 09b – Architecture diagram



Lab 09c – Implement Azure Kubernetes service (optional)

Lab scenario

Contoso has several multi-tier applications that are not suitable to run by using Azure Container Instances. To determine whether they can be run as containerized workloads, you want to evaluate using Kubernetes as the container orchestrator. To minimize management overhead, you want to test Azure Kubernetes Service, including its simplified deployment experience and scaling

Objectives

Task 1:

Deploy an Azure Kubernetes Service cluster

Task 2:

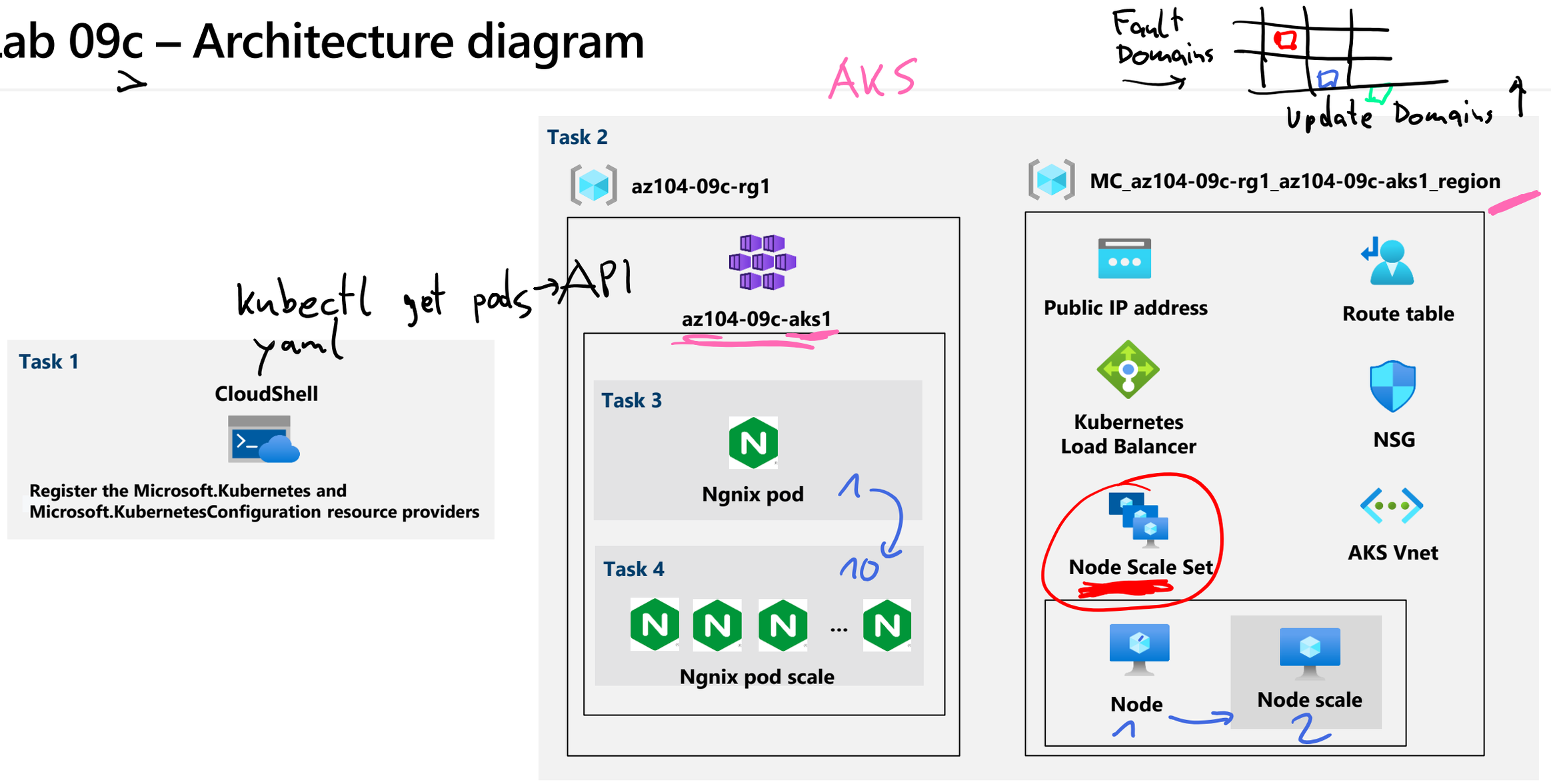
Deploy pods into the Azure Kubernetes Service cluster

Task 3:

Scale containerized workloads in the Azure Kubernetes service cluster

Next slide for an architecture diagram

Lab 09c – Architecture diagram



End of presentation