Advanced Azure

Develop a passion for learning.

TEKsystems Global Services

WORKFORCE DEVELOPMENT

PARTICIPANT GUIDE

Content Usage Parameters

Content refers to material including instructor guides, student guides, lab guides, lab or hands-on activities, computer programs, etc. designed for use in a training program

1

Content is subject to copyright protection

2

Content may only be leveraged by students enrolled in the training program

3

Students agree not to reproduce, make derivative works of, distribute, publicly perform and publicly display content in any form or medium outside of the training program

4

Content is intended as reference material only to supplement the instructor-led training

LOGISTICS

Class Hours:

- Instructor will set class start and end times.
- There will be regular breaks in class.

Telecommunication:

- Turn off or set electronic devices to silent (not vibrate)
- Reading or attending to devices can be distracting to other students
- Try to delay until breaks or after class

Miscellaneous:

- Courseware
- Bathroom
- Fire drills

COURSE OBJECTIVES

At the end of this course, you will:

- Have an understanding of the different services Azure offers and be able to choose the correct solution for different workloads
- Deploy applications in a resilient manner
- Use Entra ID for authentication
- Monitor applications running in Azure
- More!

HI!

Jason Smith

Cloud Consultant with a Linux sysadmin background. Focused on cloud-native technologies: automation, containers & orchestration

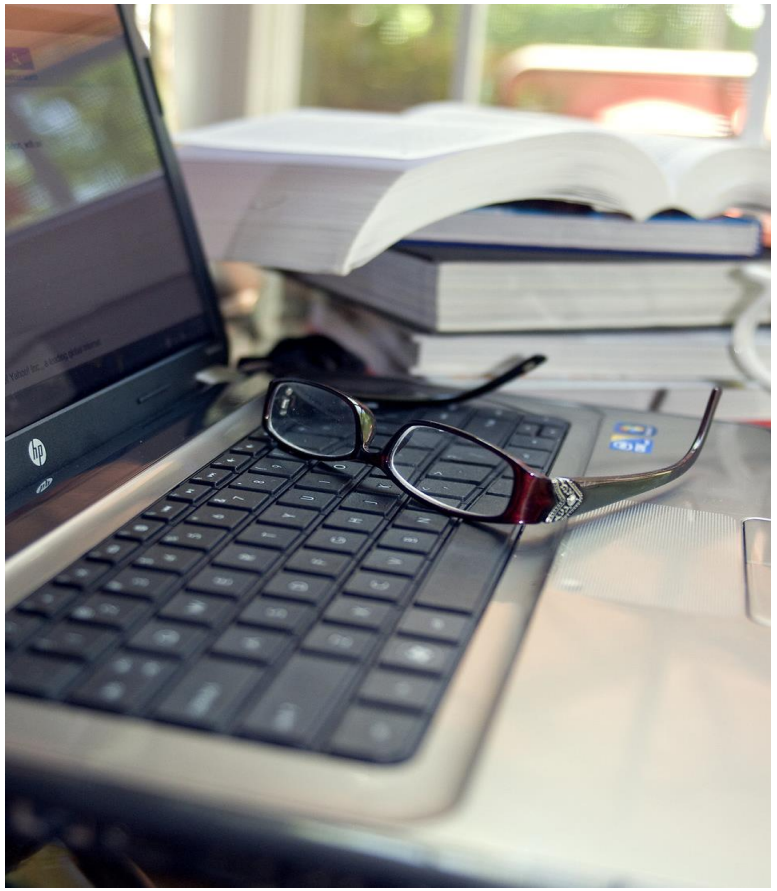
github
https://github.com/jruels

mail
jason@innovationinsoftware.com

Expertise

- Cloud
- Automation
- CICD
- Docker
- Kubernetes

LAB-FOCUSED COURSE



This is a hands-on, functional course.

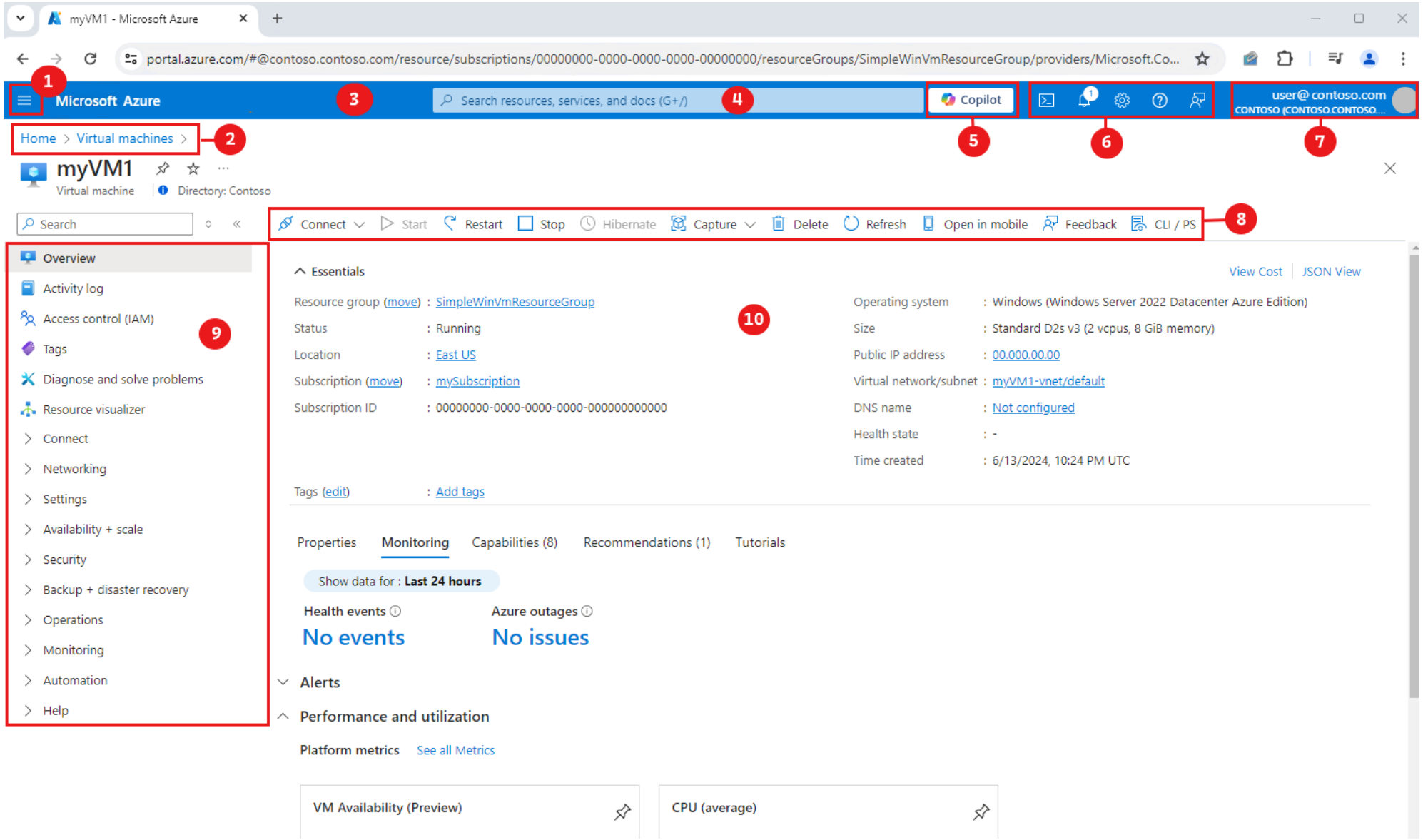
We will focus on doing labs and learning through interaction with Azure

INTRODUCTIONS

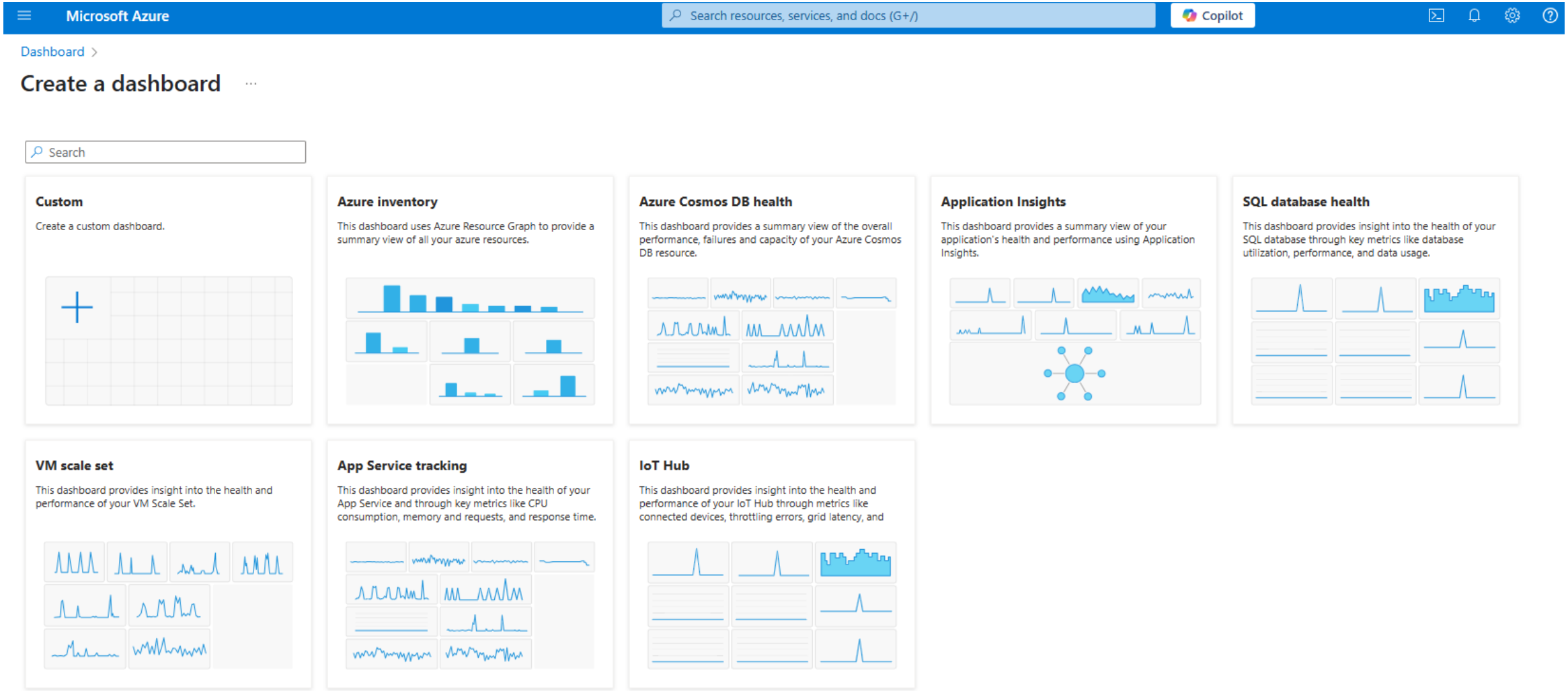
- Name
- Job Role
- Cloud/Azure experience
- Expectations for course

AZURE CLOUD SERVICES

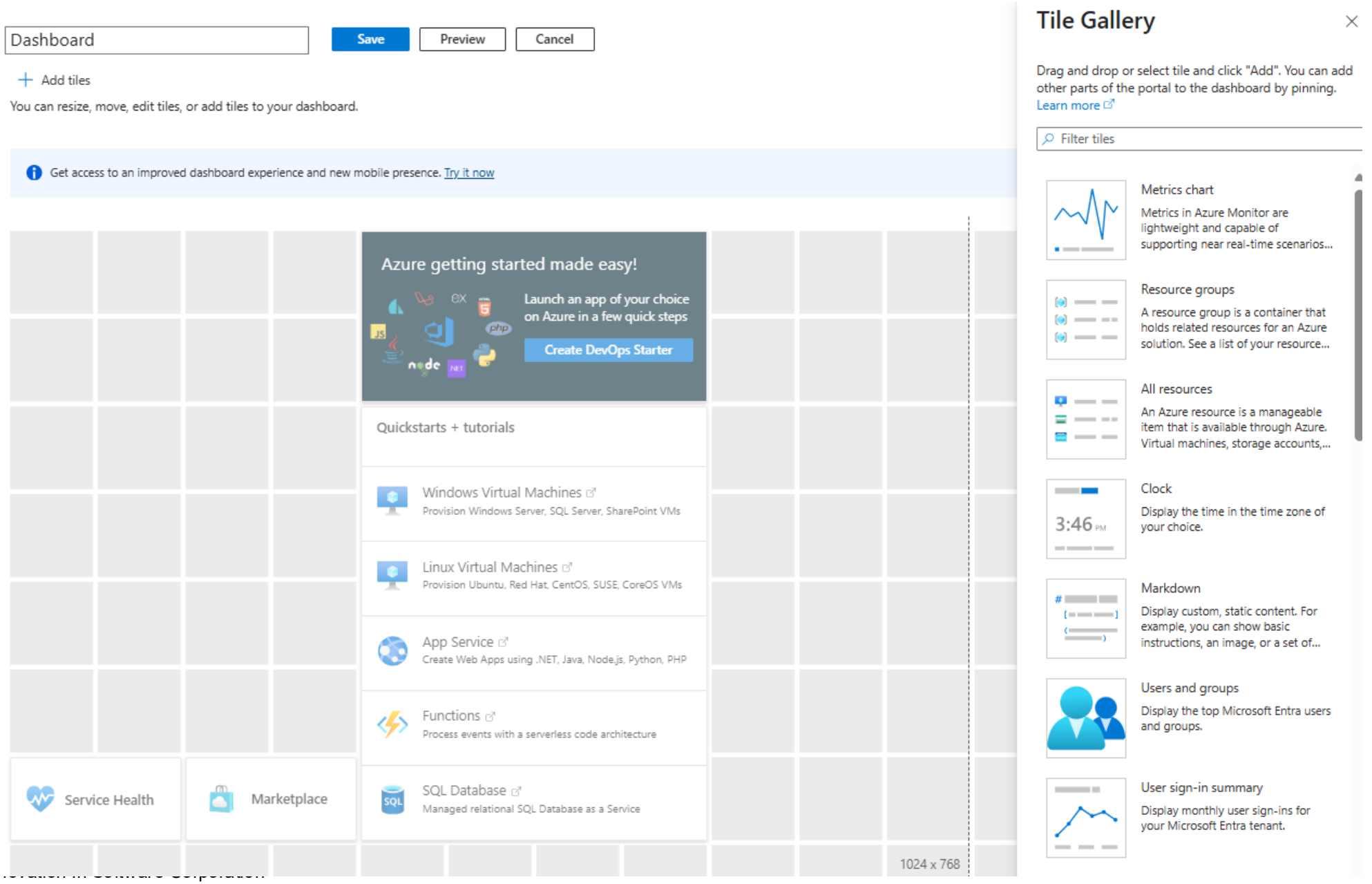
PORTAL OVERVIEW



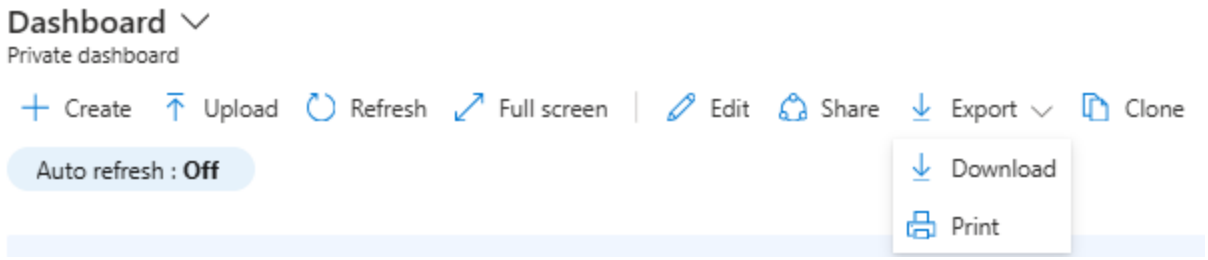
AZURE DASHBOARDS HUB



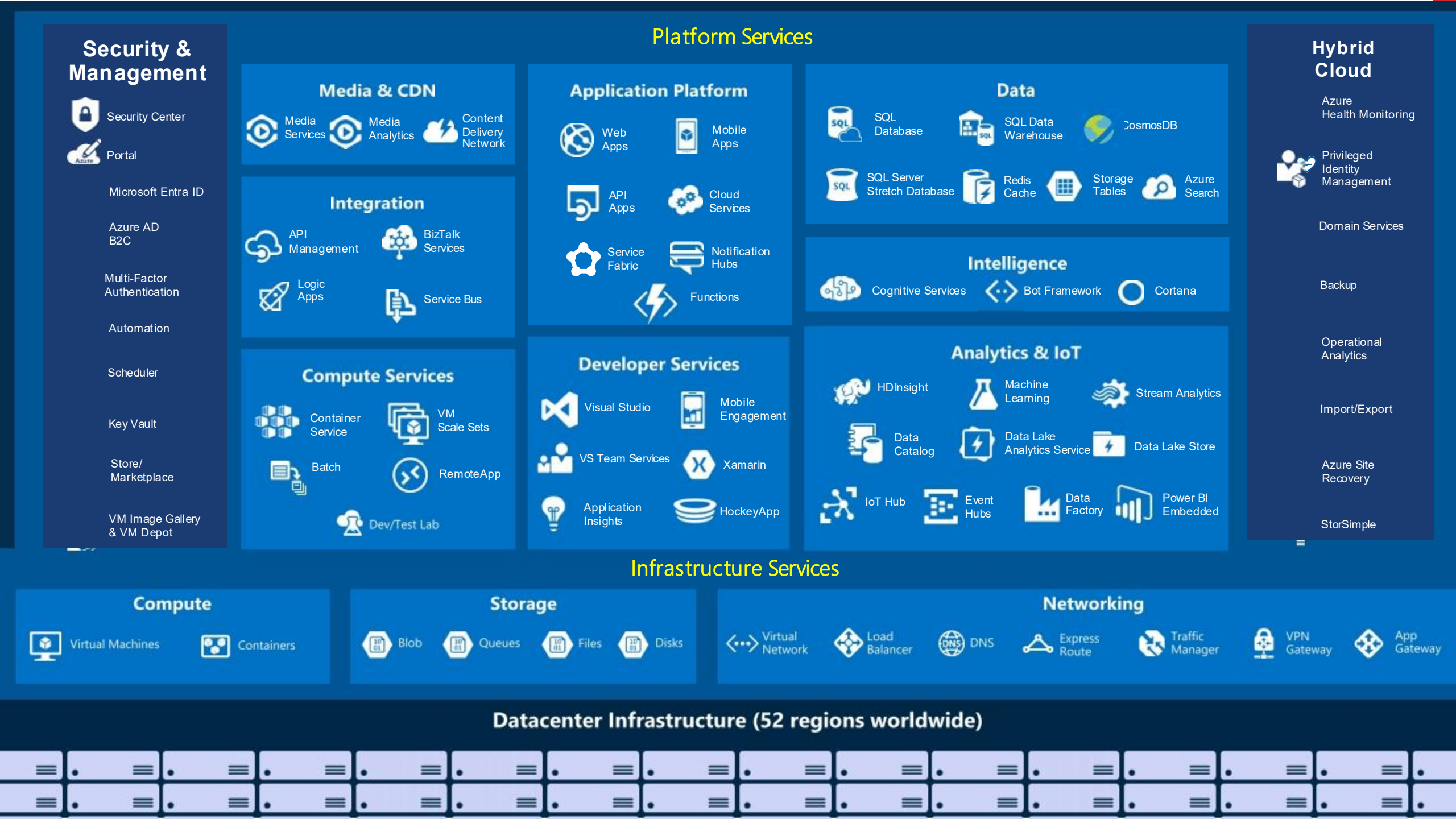
CUSTOMIZING DASHBOARDS



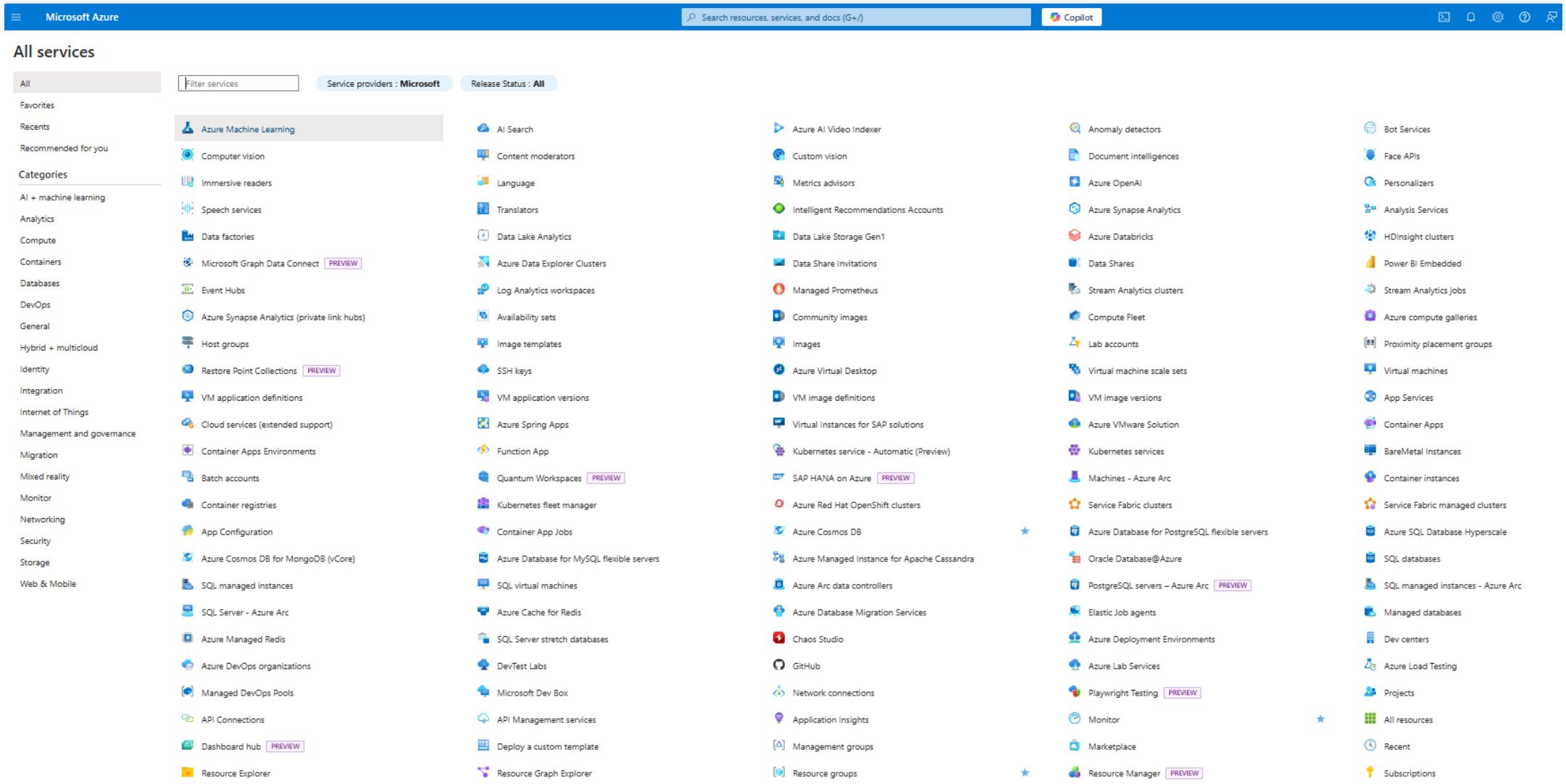
EXPORTING DASHBOARDS



You can export dashboards from the Azure Portal's Dashboard hub to inspect or reuse their structure programmatically. After export, you can also use the output as a template for future dashboards.


Security & Management
Security Center
Portal
Microsoft Entra ID
Azure AD B2C
Multi-Factor Authentication
Automation
Scheduler
Key Vault
Store/ Marketplace
VM Image Gallery & VM Depot
Platform Services
Media & CDN
Media Services
Media Analytics
Content Delivery Network
Integration
API Management
BizTalk Services
Logic Apps
Service Bus
Compute Services
Container Service
VM Scale Sets
Batch
RemoteApp
Dev/Test Lab
Application Platform
Web Apps
Mobile Apps
API Apps
Cloud Services
Service Fabric
Notification Hubs
Functions
Developer Services
Visual Studio
Mobile Engagement
VS Team Services
Xamarin
Application Insights
HockeyApp
Data
SQL Database
SQL Data Warehouse
CosmosDB
SQL Server Stretch Database
Redis Cache
Storage Tables
Azure Search
Intelligence
Cognitive Services
Bot Framework
Cortana
Analytics & IoT
HDInsight
Machine Learning
Stream Analytics
Data Catalog
Data Lake Analytics Service
Data Lake Store
IoT Hub
Event Hubs
Data Factory
Power BI Embedded
Hybrid Cloud
Azure Health Monitoring
Privileged Identity Management
Domain Services
Backup
Operational Analytics
Import/Export
Azure Site Recovery
StorSimple
Infrastructure Services
Compute
Virtual Machines
Containers
Storage
Blob
Queues
Files
Disks
Networking
Virtual Network
Load Balancer
DNS
Express Route
Traffic Manager
VPN Gateway
App Gateway
Datacenter Infrastructure (52 regions worldwide)

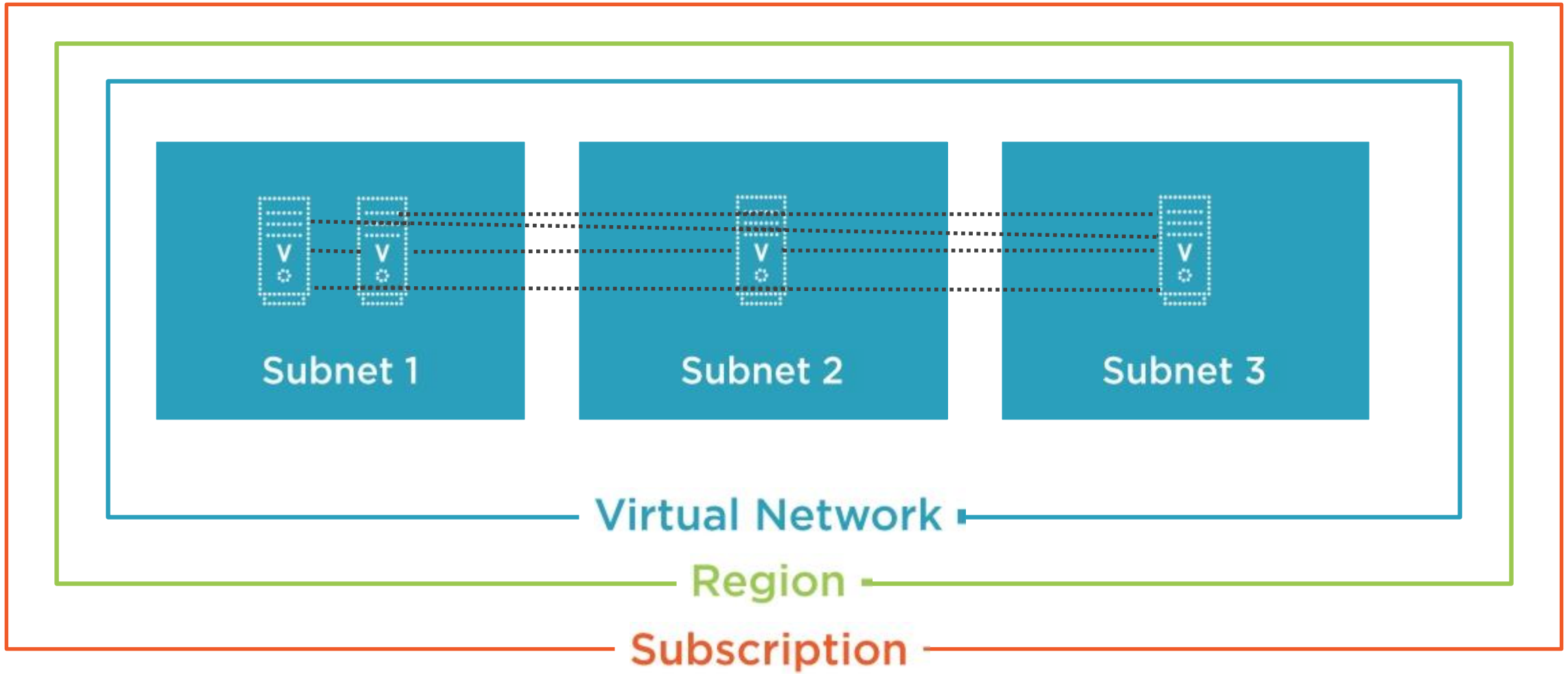
AZURE RESOURCES



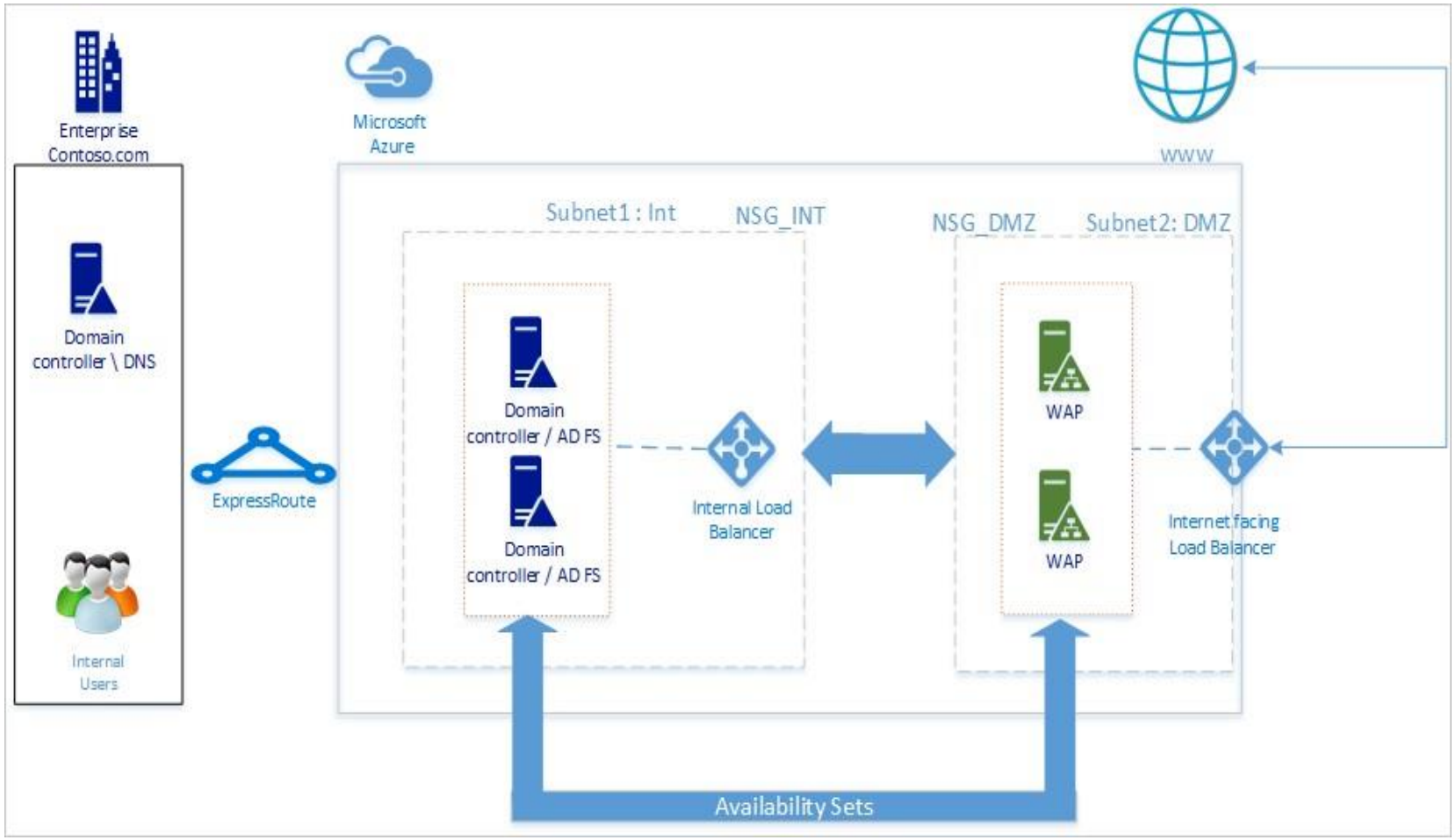
AZURE VIRTUAL NETWORKS (VNET)

- A VNet is a representation of your own network in the cloud. A VNet is a logical isolation of the Azure cloud dedicated to your subscription.
- Provides network access between Cloud resources
- Private network only accessible in your cloud

VNET DIAGRAMS



VNET DIAGRAMS

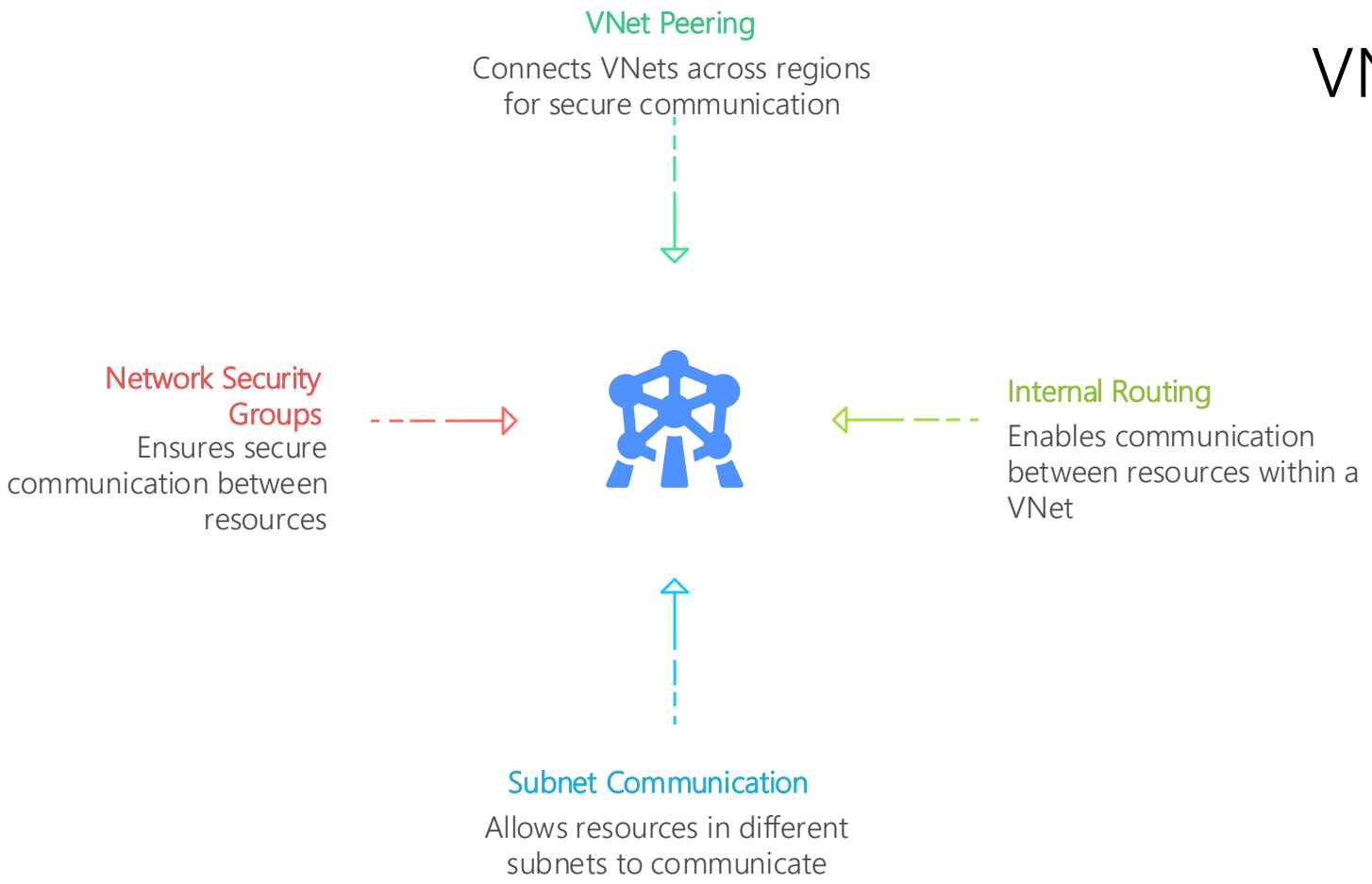


VNET CAPABILITIES

Isolation

- VNets are isolated from one another
- Highly customizable
  - VNets for Dev, Test, Prod with same CIDR
  - VNets with different CIDRs connected
- Azure provides internal DNS resolution for Vnets (Optional)

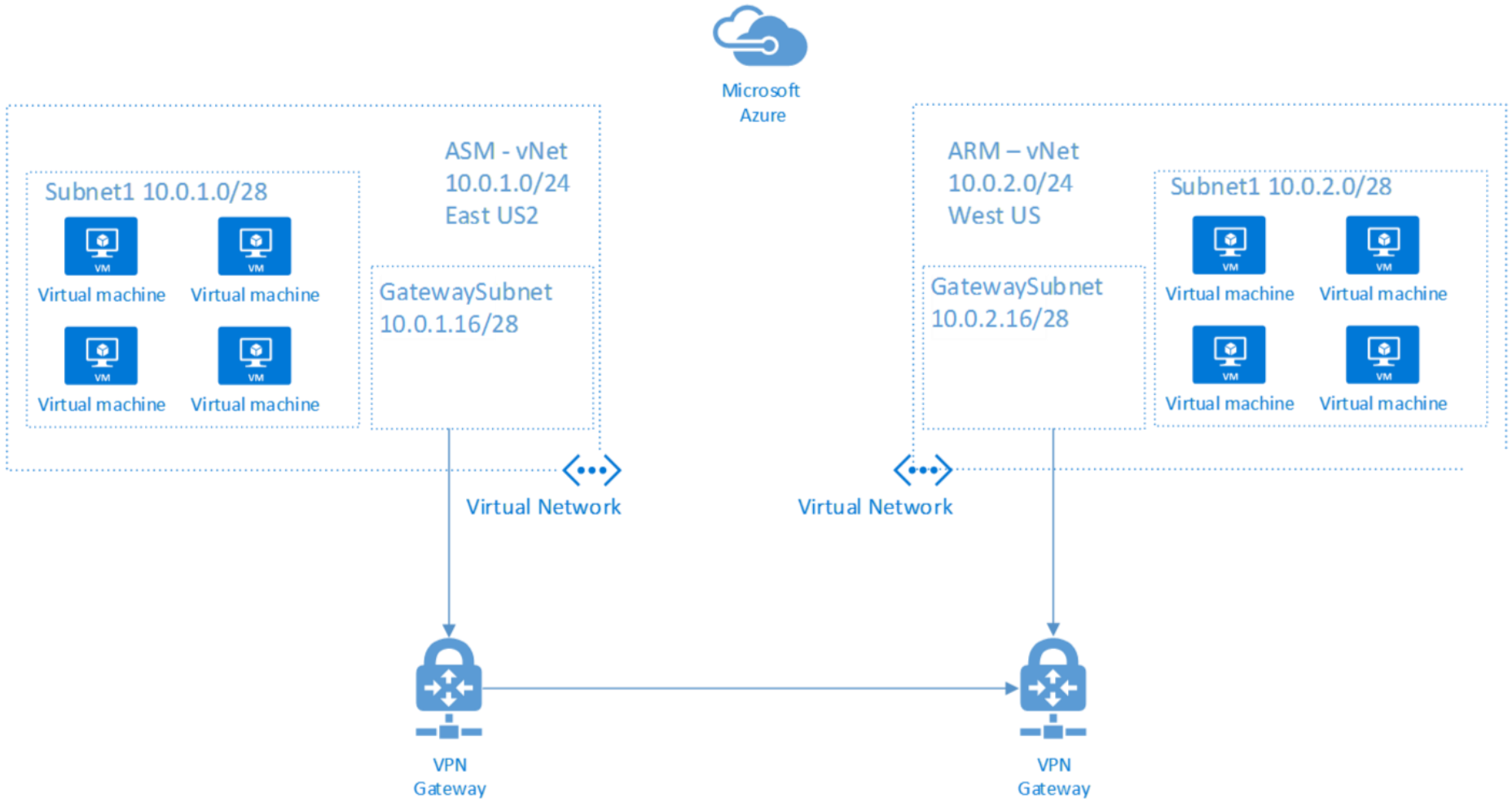
VNET CAPABILITIES



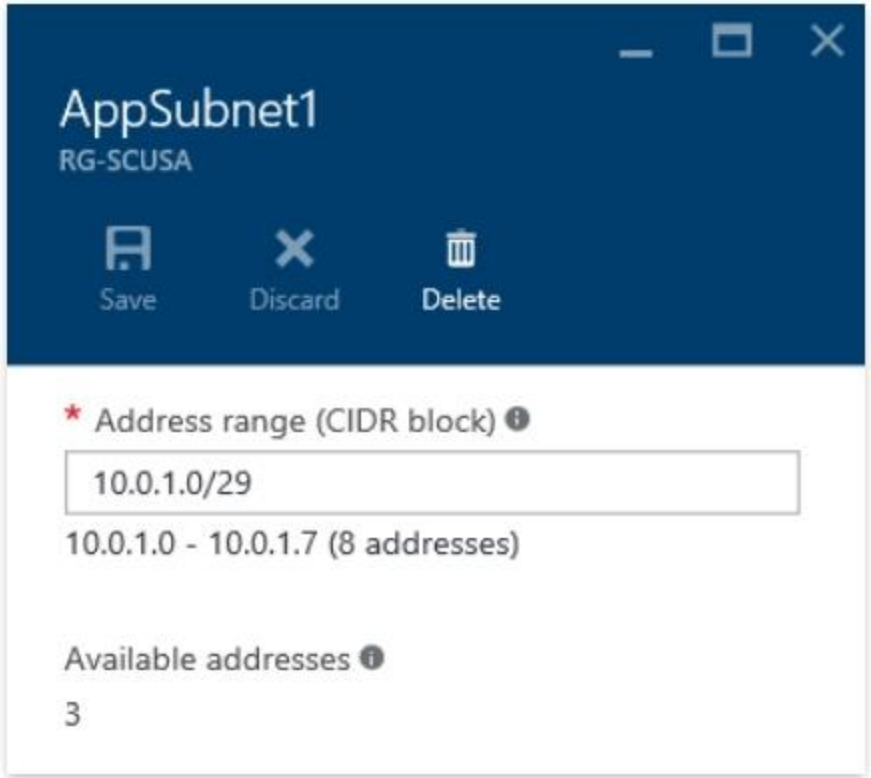
VNet connectivity

- VNets can be connected to each other.
- Any resource can communicate with any other resource even on different Subnets.
- App server can connect to database server on different Subnet

VNET DIAGRAMS

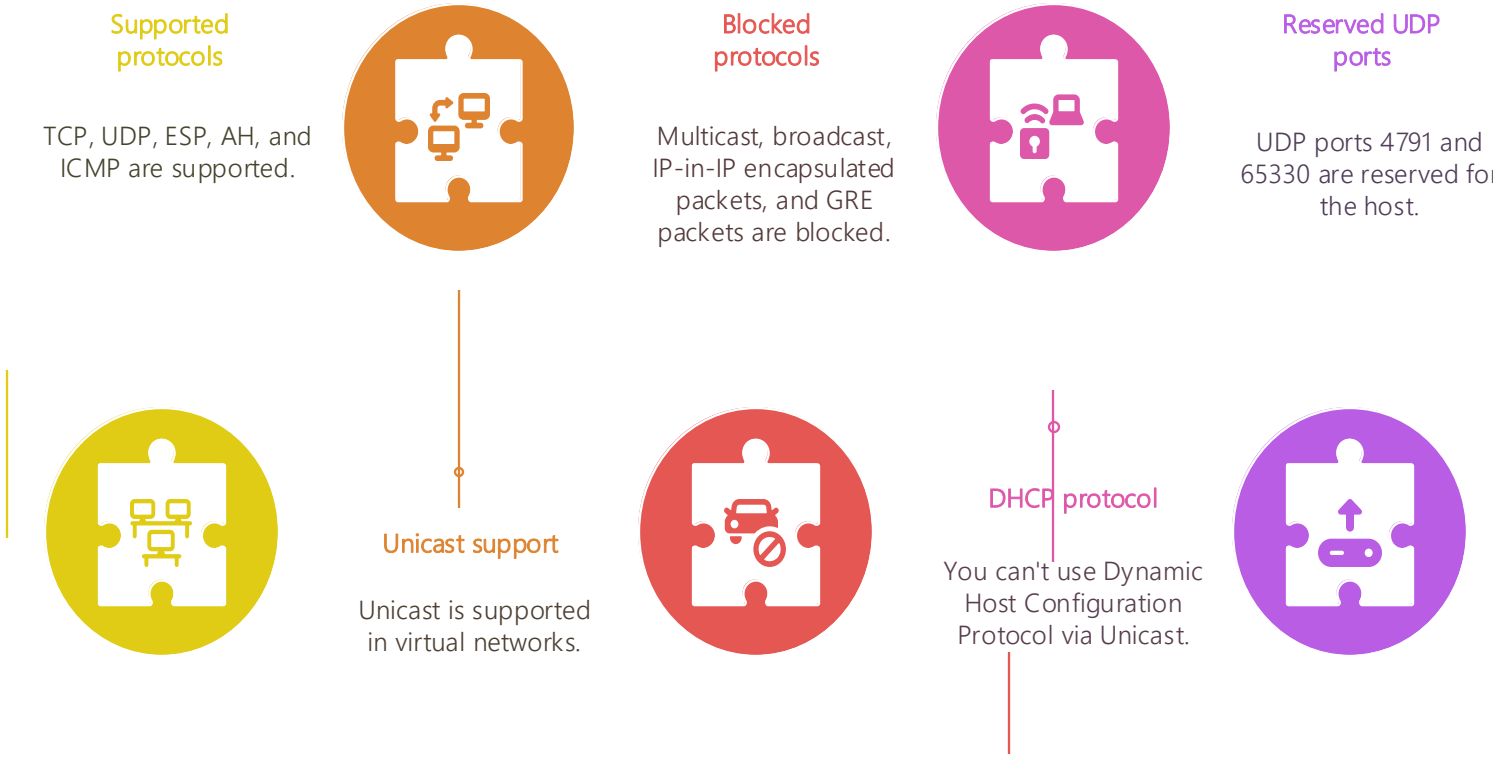


VNET RESERVED IPS



- Certain IPs reserved for Azure's internal use
- First and last reserved per protocol for host ID and broadcast
- The first 3 IP addresses are reserved

VNET PROTOCOL SUPPORT AND LIMITATIONS



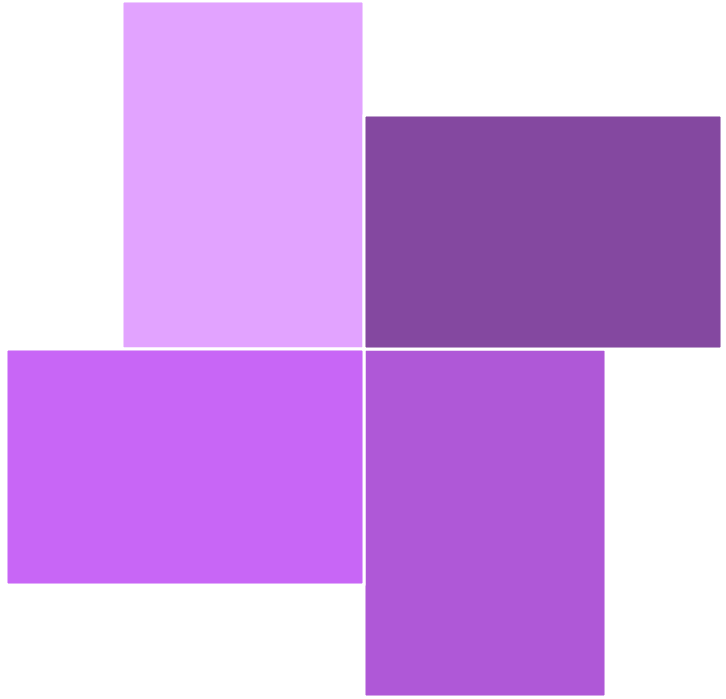
TRAFFIC FLOW IN A VNET

VPN/ExpressRoute

Communication is possible via VPN or ExpressRoute connections.

Internet Access

VMs have default outbound access to anything on the internet.



Same VNET

VMs within the same virtual network subnets can communicate.

Peered VNETs

VMs in peered virtual networks across subnets are able to communicate.

TRAFFIC FLOW IN A VNET

This may not always be desired:

- A multi-tiered application may only require neighbor tiers to communicate.
- May want to restrict public outbound access (DMZ)
- May want to restrict types of traffic.

NETWORK SECURITY GROUPS

Enable rules and assign to NSGs
In Azure, you create custom security rules and assign them to a **Network Security Group (NSG)**, which acts as a ruleset container.

Apply NSGs to subnets or NICs (or both)
NSGs can be associated with either an entire **subnet** or individual **VM NICs** — or even both.

Subnet-level NSGs aren't edge firewalls
Associating an NSG to a subnet doesn't create a perimeter-only filter — **rules are enforced per NIC**. Every VM NIC in that subnet enforces those rules as if applied locally.

Order of rule evaluation:
Inbound traffic: Subnet NSG is evaluated first; if allowed, the NIC NSG is evaluated next. Both must allow traffic for it to pass.
Outbound traffic: NIC NSG rules are evaluated first, then subnet NSG rules. Again, both must permit the traffic to succeed.

NSG RULES

Azure Network Security Group rules are defined using a **five-tuple** model. Each rule specifies:

Source IP – can be a specific IP, a CIDR range, a service tag (like Internet, VirtualNetwork), or an Application Security Group

Source port – can be a single port, a range (e.g., 1000-2000), or for any port

Destination IP – similar flexibility to source (IP, CIDR, tag, ASG)

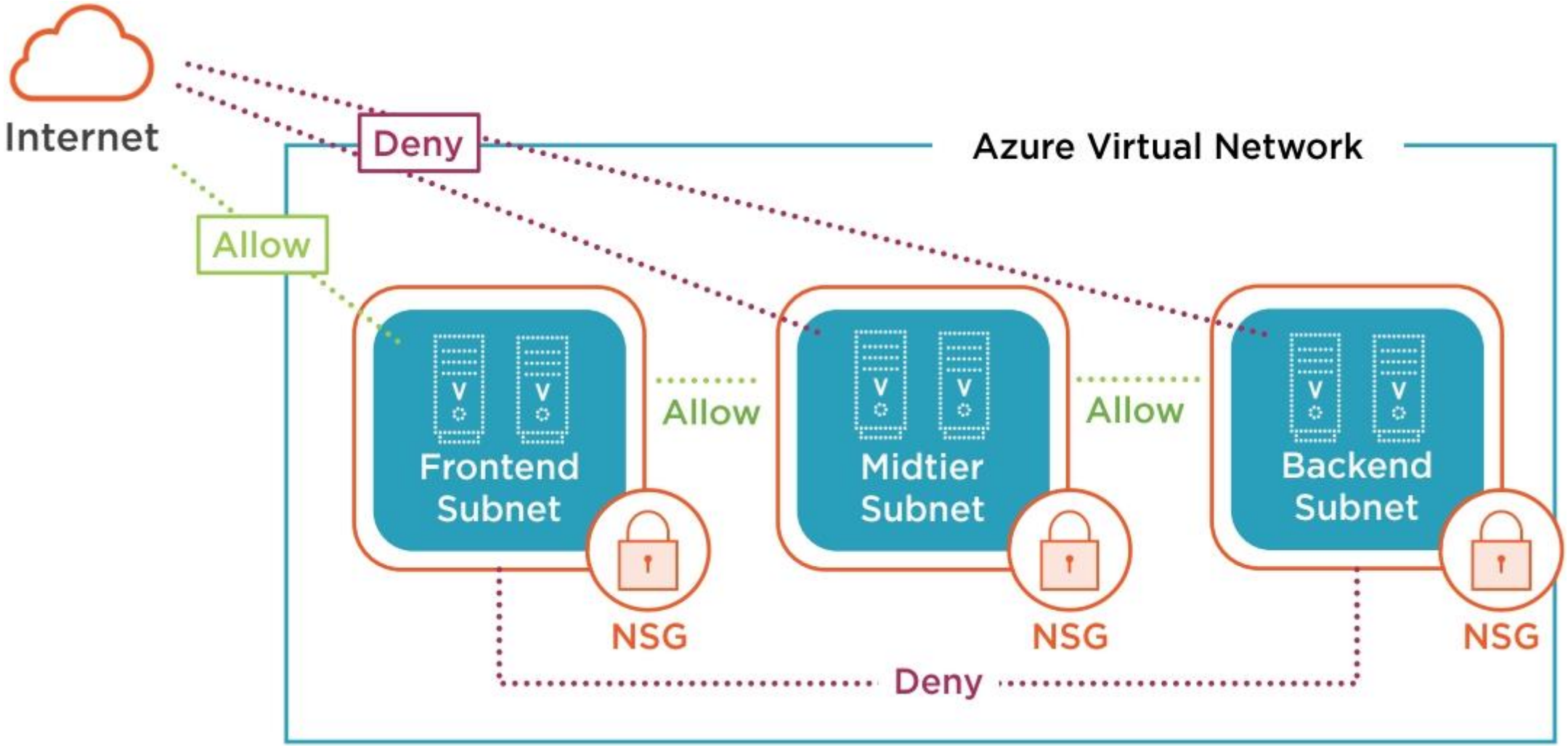
Destination port – can be a specific port, range, or wildcard

Protocol – TCP, UDP, ICMP, ESP, AH, or to match all

NSG RULES

- CIDR used to define source/destination
- Three special identifies
  - VIRTUAL_NETWORK: identifies traffic within the virtual network address space
  - AZURE_LOADBALANCER: The Azure infrastructure load balancer
  - INTERNET: IP address space external to the virtual network

NSG EXAMPLE



NSG EXAMPLE

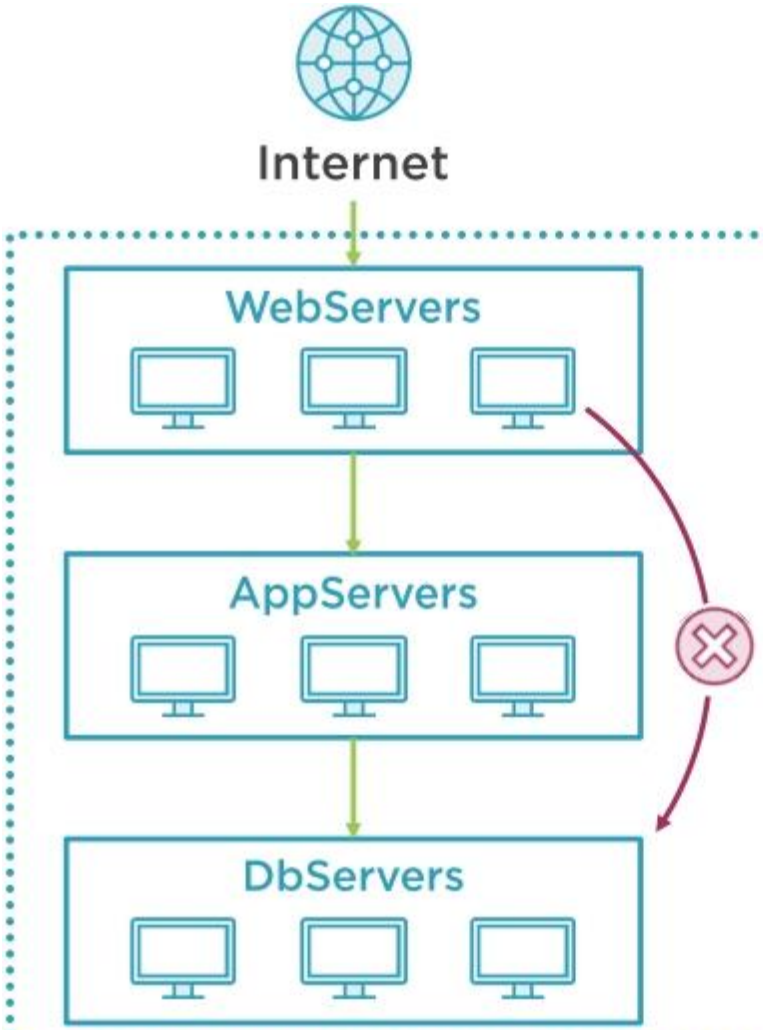
- Rules are combined in a NSG
- Based on the priority flexible configurations are possible
- Lower priority number means higher priority

Description	Priority	Source Address	Source Port	Destination Address	Destionation Port	Action
ILB	1010	AZURE_LOADBALANCER	*	*	10000	Allow
Inbound RDP	2005	VIRTUAL_NETWORK	*	*	3389	Allow
Deny all inbound	4000	*	*	*	*	Deny

APPLICATION SECURITY GROUPS

- NSGs are focused on the IP address ranges which may be difficult to maintain for growing environments
- ASGs enable monikers for different application roles to be defined, e.g. Web, DB, WebApp
- The NIC for a VM is made a member of one or more ASGs
- ASGs are used in rules that are part of NSGs to control flow of communication and can still use NSG features like service tags

ASG EXAMPLE



Network Security Group

Action	Name	Source	Destination	Port
Allow	AllowInternetToWeb	Internet	WebServers	80,443 (HTTP/HTTPS)
Allow	AllowWebToApp	WebServers	AppServers	443 (HTTPS)
Allow	AllowAppToDb	AppServers	DbServers	1443 (MSSQL)
Deny	DenyAllinbound	Any	Any	Any

POP QUIZ:

Scenario:
VNet-EastUS (10.1.0.0/24) is peered with VNet-WestEurope (10.2.0.0/24). You want only DB servers in WestEurope to talk to App servers in EastUS—nothing else should cross the peering.

What is the *simplest and most secure* way to enforce this?
A. Deploy Azure Firewall in EastUS to filter peered traffic
B. Use "Block traffic to remote VNet" on WestEurope peering + NSG allow only DB subnet
C. Apply NSG on EastUS subnet allowing only 10.2.0.0/24 → 10.1.0.0/24
D. Use User Defined Route to drop all remote VNet IP ranges except the DB subnet

POP QUIZ:

Scenario:
VNet-EastUS (10.1.0.0/24) is peered with VNet-WestEurope (10.2.0.0/24). You want only DB servers in WestEurope to talk to App servers in EastUS—nothing else should cross the peering.

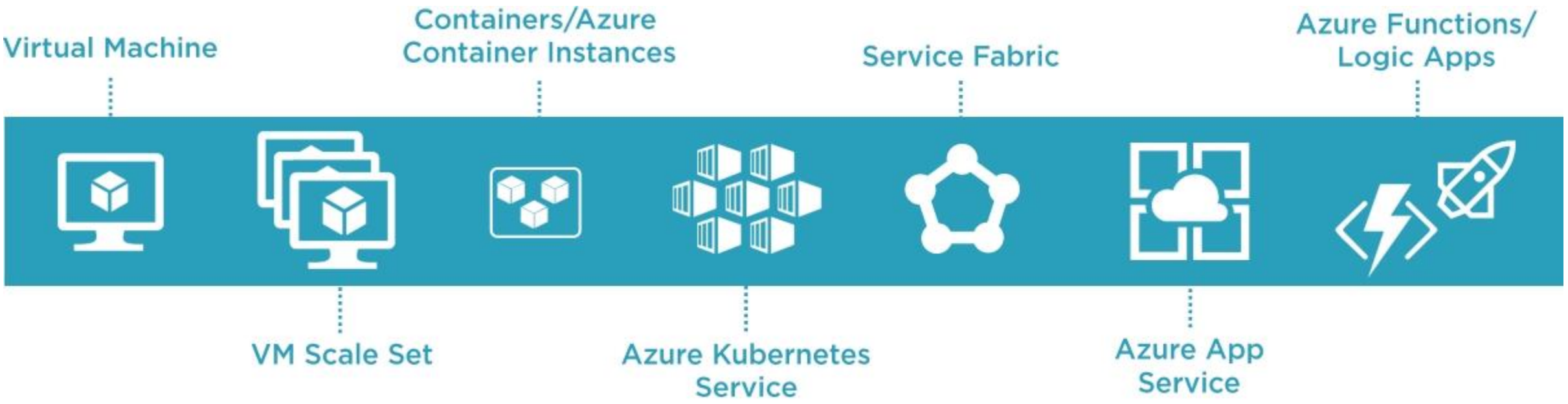
What is the *simplest and most secure* way to enforce this?
A. Deploy Azure Firewall in EastUS to filter peered traffic
B. Use "Block traffic to remote VNet" on WestEurope peering + NSG allow only DB subnet
C. Apply NSG on EastUS subnet allowing only 10.2.0.0/24 → 10.1.0.0/24
D. Use User Defined Route to drop all remote VNet IP ranges except the DB subnet

Choosing the Right Compute Service

KEY FACTORS WHEN SELECTING COMPUTE

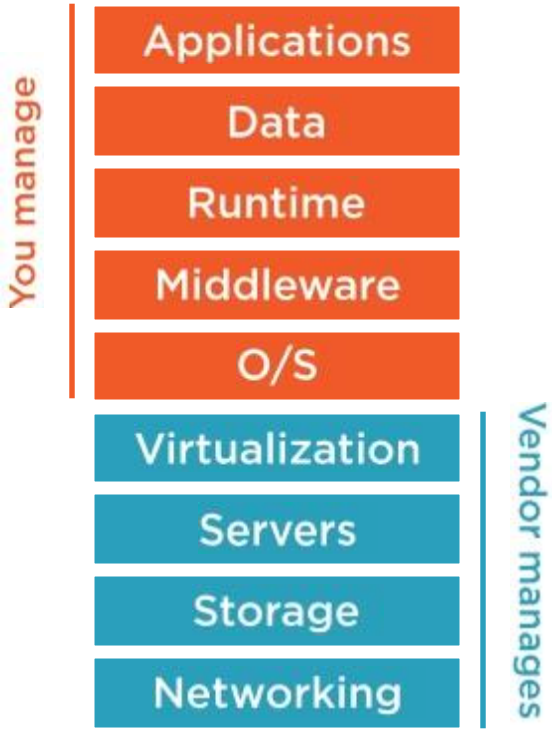
Characteristic	Azure VMs	App Service	Container Apps	Functions
OS Control	Full control	Azure managed	Azure managed	Azure managed
OS Support	Windows or Linux	Windows or Linux	Windows or Linux	Windows or Linux
Portability	Limited	Limited	Strong	Limited
Dev Productivity	Lower	Higher	Medium	Higher
Multi-Instance	Scale sets	Built-in	Built-in	Built-in
Cost	Higher	Predictable	Variable	Lower

COMPUTE OPTIONS



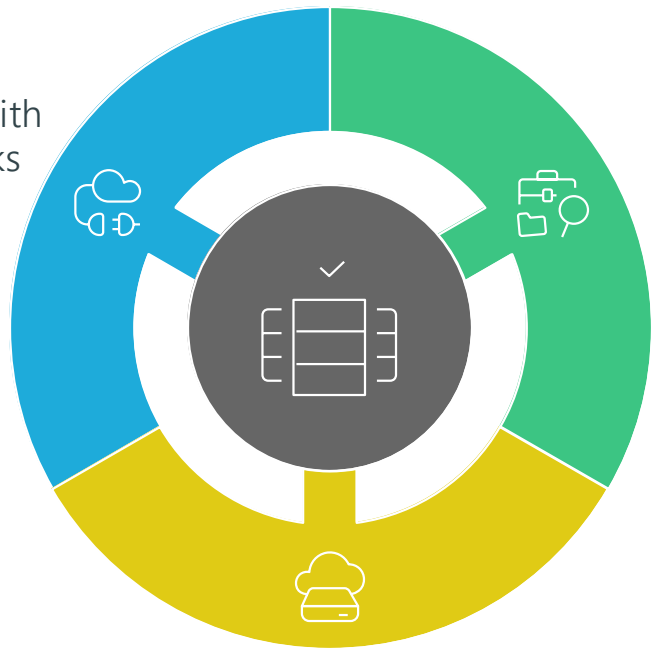
VIRTUAL MACHINES

- Provides a VM from a variety of series and sizes with different capabilities
- VMs are placed on a virtual network which can provide connectivity to on-premises networks
- Windows and Linux guest OS support with large library of extensions
- Full access and responsibility for everything inside the VM
- Template support for prescriptive, repeatable deployment



VIRTUAL MACHINES

When deploying Azure Virtual Machines, it's important to understand that they don't operate in isolation. They rely on several foundational Azure services to function properly



Networking
Secure and control traffic with virtual networks and security groups.

Resource Groups
Organize and manage related Azure resources efficiently.

Storage / Managed Disks
Ensure high availability and performance with managed storage solutions.

Storage

AZURE STORAGE ACCOUNTS

Blob Storage

- Object and Disk Storage
- Blog Storage Tiers
- Azure Search integration

File Storage

- SMB File Shares
- Attach to VMs as file shares
- Azure File Sync

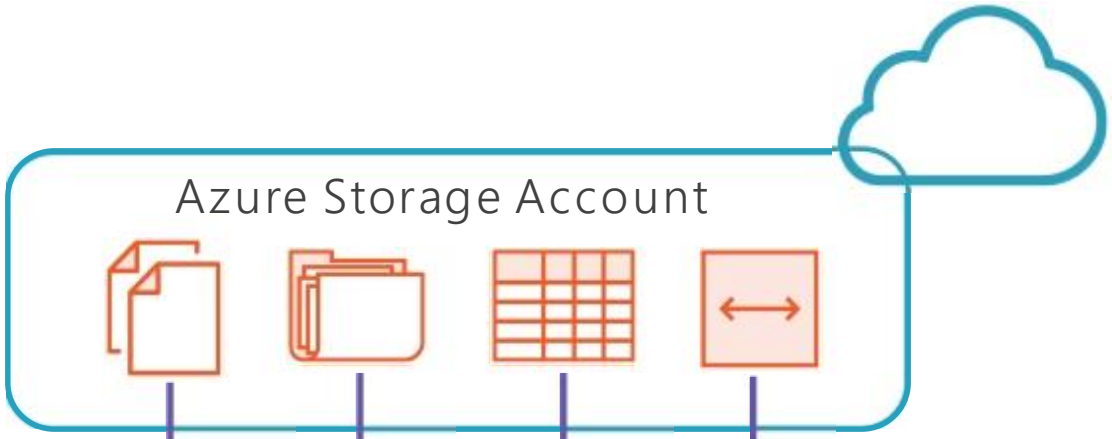
Table Storage

- NoSQL Data Store
- Schema-less design
- Azure Cosmos DB

Queue Storage

- Message based
- For building synchronous apps

AZURE STORAGE ACCOUNT ENDPOINTS



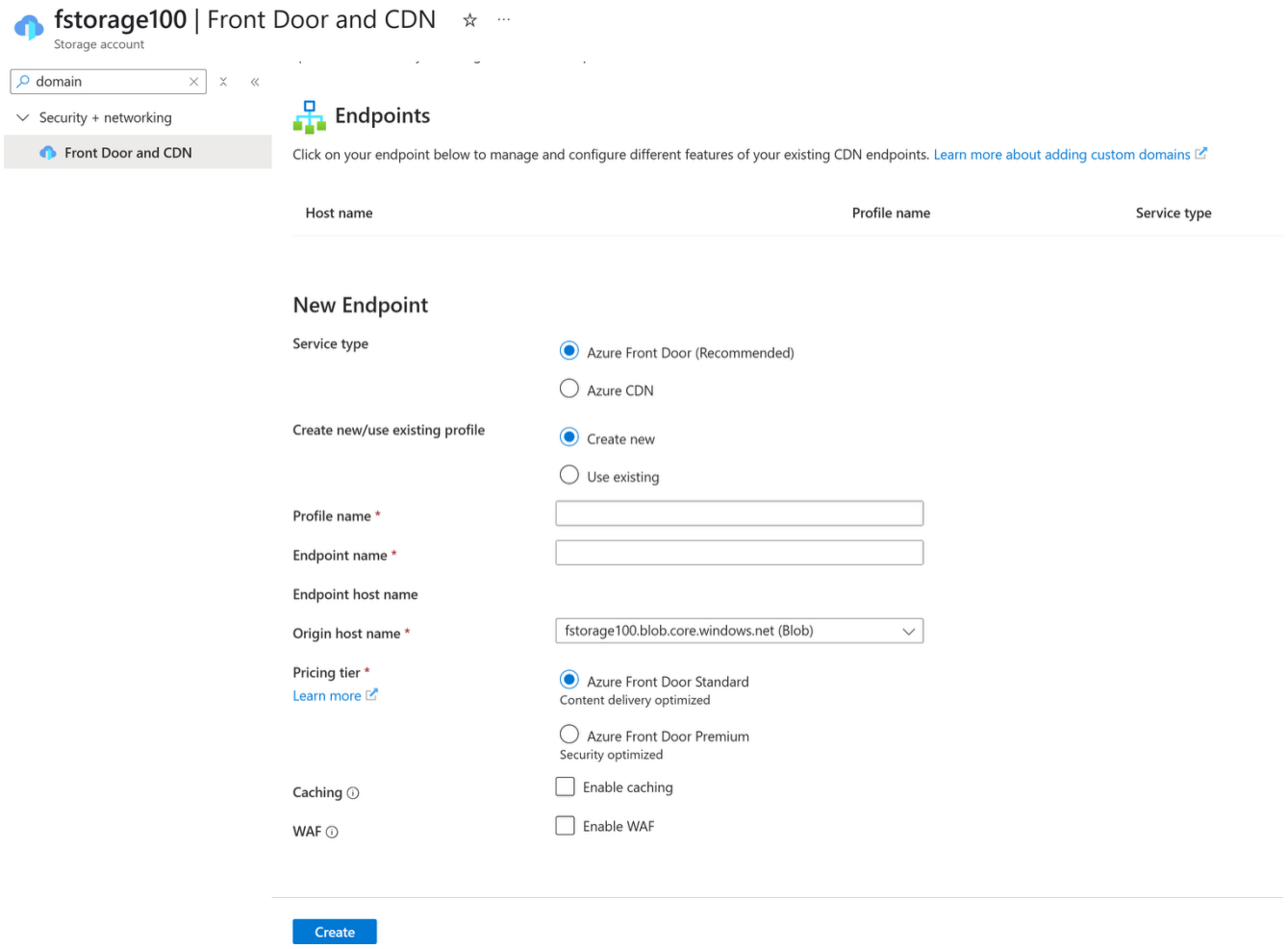
https://mystorageaccount.queue.core.windows.net

https://mystorageaccount.table.core.windows.net

https://mystorageaccount.file.core.windows.net

https://mystorageaccount.blob.core.windows.net

AZURE STORAGE CUSTOM DOMAIN NAME



Azure Storage accounts can be accessed using **custom domain names** setup with Azure Front Door.

AZURE BLOB STORAGE TYPES

Block Blob

- 190.7 TB Max file size
- Composed of 4000MB blocks
- Insert, replace, delete blocks

Append Blob

- 195GB Max file size
- Can only append blocks
- Ideal for log/audit files

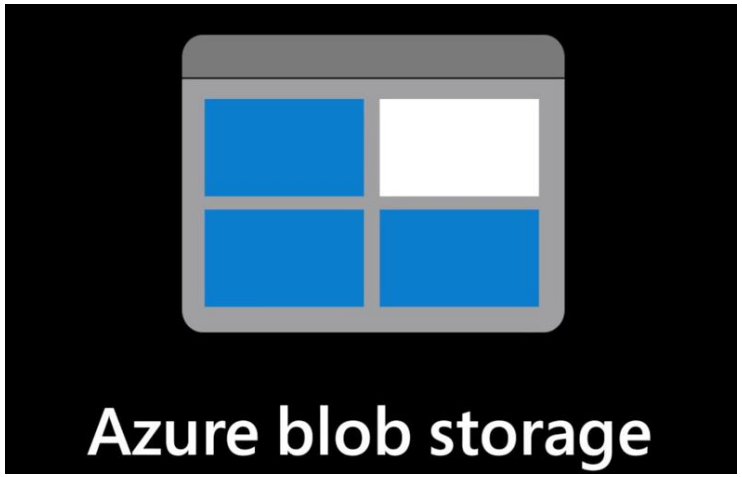
Page Blob

- 8TB max file size
- Frequent read/write operations
- Standard/Premium Storage

USE CASES

Use Cases

- Data Lakes (ADLS2)
- Cloud-native workloads
- Archives
- Machine learning
- High performance computing
- All Big Data use cases



BLOB STORAGE TIERS

	Hot	Cool	Archive
USE	Storing data that is frequently accessed or modified	Storing data that is infrequently accessed, but needs to remain accessible nonetheless	Storing data that is hardly ever accessed, but still needs to be kept
PROS	Low data access and transaction cost, and no minimum retention period	Storage costs are lower than the Hot tier	Has the lowest storage costs
CONS	The highest storage cost	Access and transaction costs are more than the Hot tier; also has a minimum data retention period of 30 days	The highest access and transaction cost and latency usually measured in hours; also has a minimum data retention period of 180 days

https://www.linkedin.com/pulse/azure-storage-lifecycle-management-optimizing-data-nidhi-kumari-5naxe/

DEMO – CREATE AZURE STORAGE ACCOUNT WITH PYTHON

Setup and Prerequisites

Azure Subscription
Ensure you have an active subscription and required permissions.

Python 3.8+ Environment
python --version

Install Required Packages
sudo apt-get update
pip install azure-identity azure-mgmt-resource azure-mgmt-storage azure-storage-blob

DEMO – CREATE AZURE STORAGE ACCOUNT WITH PYTHON

Create a project directory:

mkdir blobdemo & cd blobdemo
touch provision_blob.py

Add the provisioning code

Run:
az login
python provision_blob.py

```python
import os
from azure.identity import DefaultAzureCredential
from azure.mgmt.resource import ResourceManagementClient
from azure.mgmt.storage import StorageManagementClient

SUBSCRIPTION_ID = ""  # Replace with your Azure subscription ID
LOCATION = "eastus"
RG_NAME = "demo-rg"
STORAGE_ACCOUNT_NAME = "demostorageacct2030"  # Must be globally unique

cred = DefaultAzureCredential()
rg_client = ResourceManagementClient(cred, SUBSCRIPTION_ID)
stg_client = StorageManagementClient(cred, SUBSCRIPTION_ID)

rg_client.resource_groups.create_or_update(RG_NAME, {"location": LOCATION})
print(f"Resource group '{RG_NAME}' created in {LOCATION}")

stg_async = stg_client.storage_accounts.begin_create(
    RG_NAME, STORAGE_ACCOUNT_NAME,
    {
        "location": LOCATION,
        "sku": {"name": "Standard_LRS"},
        "kind": "StorageV2",
        "enable_https_traffic_only": True
    }
)
stg_account = stg_async.result()
print(f"Storage account '{STORAGE_ACCOUNT_NAME}' created")
```

DEMO – UPLOAD TO AZURE STORAGE ACCOUNT WITH PYTHON

Create an upload script

touch upload_blob.py

Add the file upload code

Run:
python upload_blob.py

```python
# upload_blob.py

import os, uuid
from azure.identity import DefaultAzureCredential
from azure.storage.blob import BlobServiceClient

ACCOUNT_NAME = "demostorageacct2030"  # Replace with your storage account name
ACCOUNT_URL = f"https://{ACCOUNT_NAME}.blob.core.windows.net"
CONTAINER_NAME = "demo-container"
LOCAL_FILE_PATH = "data/myfile.txt"
BLOB_NAME = f"{uuid.uuid4()}-{os.path.basename(LOCAL_FILE_PATH)}"

# Authenticate and connect
blob_service_client = BlobServiceClient(ACCOUNT_URL, credential=DefaultAzureCredential())

# 1. Create container if missing
try:
    container_client = blob_service_client.create_container(CONTAINER_NAME)
    print(f"Container '{CONTAINER_NAME}' created.")
except Exception:
    container_client = blob_service_client.get_container_client(CONTAINER_NAME)
    print(f"Using existing container '{CONTAINER_NAME}'.")

# 2. Upload blob
blob_client = container_client.get_blob_client(BLOB_NAME)
with open(LOCAL_FILE_PATH, "rb") as f:
    blob_client.upload_blob(f)
print(f"Uploaded blob: {BLOB_NAME}")
```

POP QUIZ:

Scenario: Your team must ensure blobs can never be deleted or overwritten for 7 days, except under legal hold. Which feature supports this?

A. Read-access geo-redundant storage (RA-GRS)
B. Immutable Blob Storage policies with retention
C. Soft-delete and versioning
D. Premium GPv2 with snapshots enabled

POP QUIZ:

Scenario: Your team must ensure blobs can never be deleted or overwritten for 7 days, except under legal hold. Which feature supports this?

A. Read-access geo-redundant storage (RA-GRS)

B. Immutable Blob Storage policies with retention

C. Soft-delete and versioning

D. Premium GPv2 with snapshots enabled


RESOURCES

AZURE RESOURCE MANAGER

Brand new JSON-based API that is built on resource providers

Resources belong to a resource group and share a common life-cycle

Cloud services not used for IaaS (but still for PaaS)

JSON-based deployments are declarative, idempotent and understand dependencies between resources to govern creation order

AZURE RESOURCE MANAGER

Massive parallelism

Tags used to organize resources which are also visible in billing

Can also use PowerShell and the modern Azure Portal

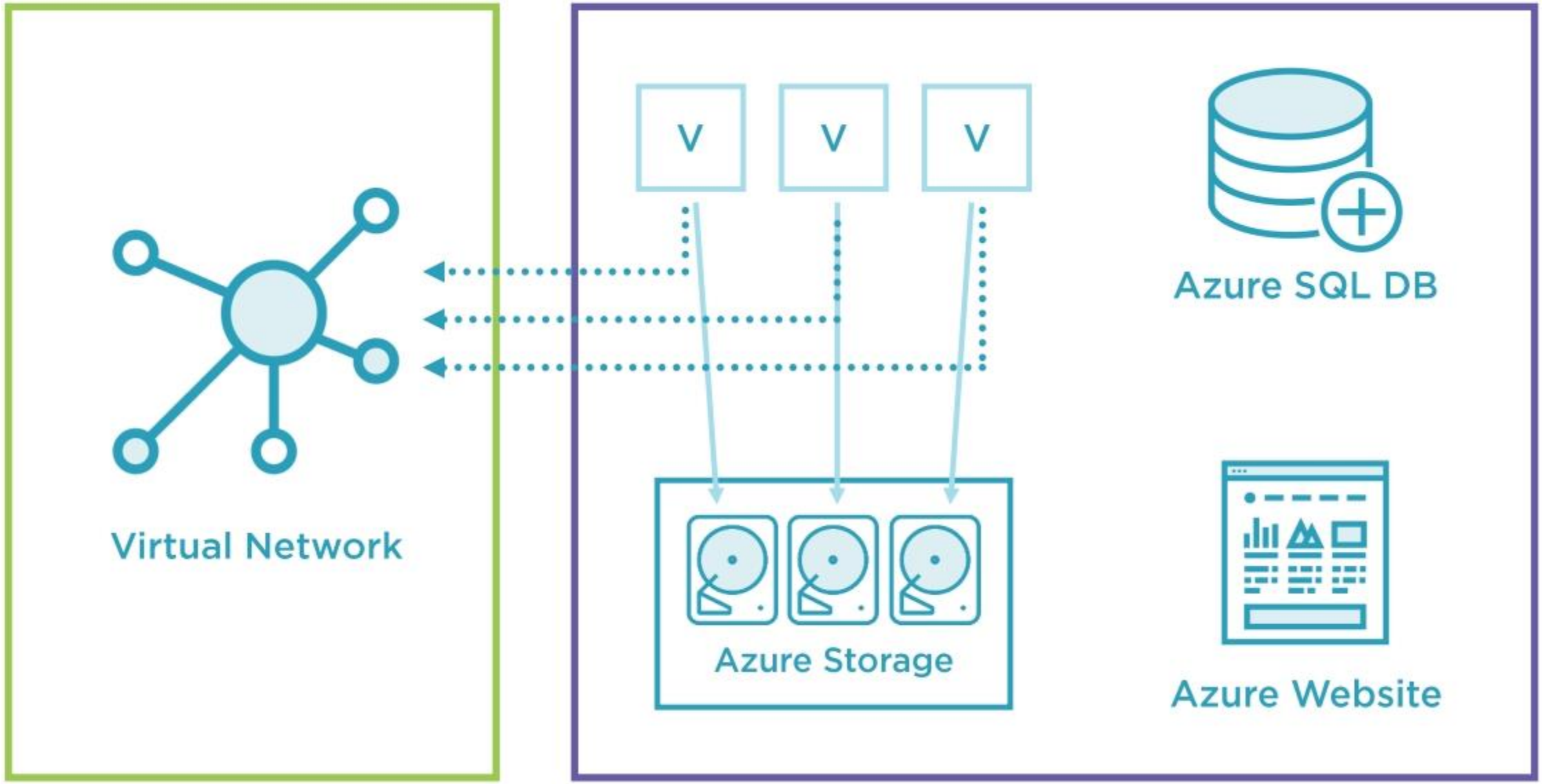
Resource Groups

- Every ARM resource exists in one, and only one, resource group
- While resource groups are created in a region they can contain resources from outside the region
- Resources can be moved between resource groups

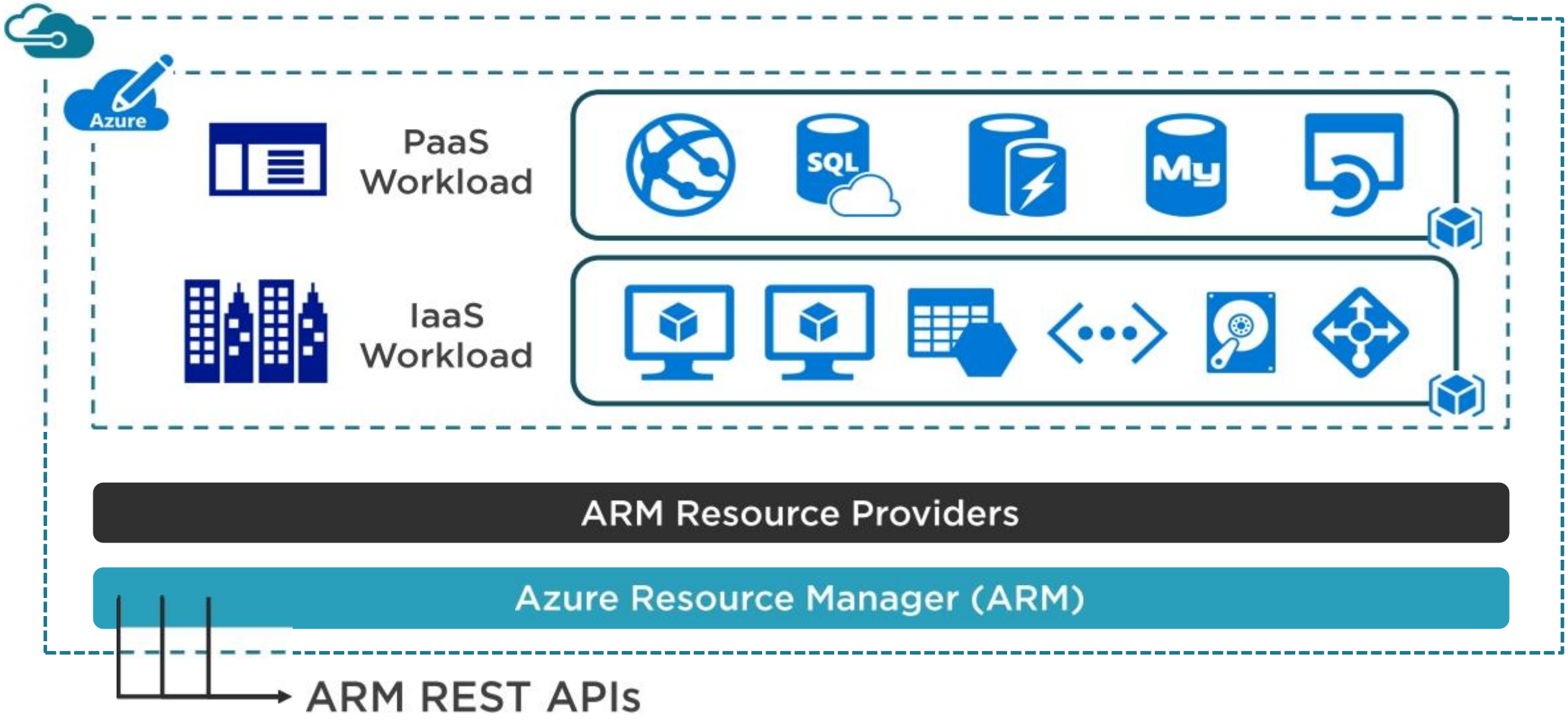
Resource Groups

- All resources in your group should share same lifecycle.
  - deploy, update and delete together
- Resource groups can be heterogeneous or homogeneous
- Resource groups are not a boundary of access

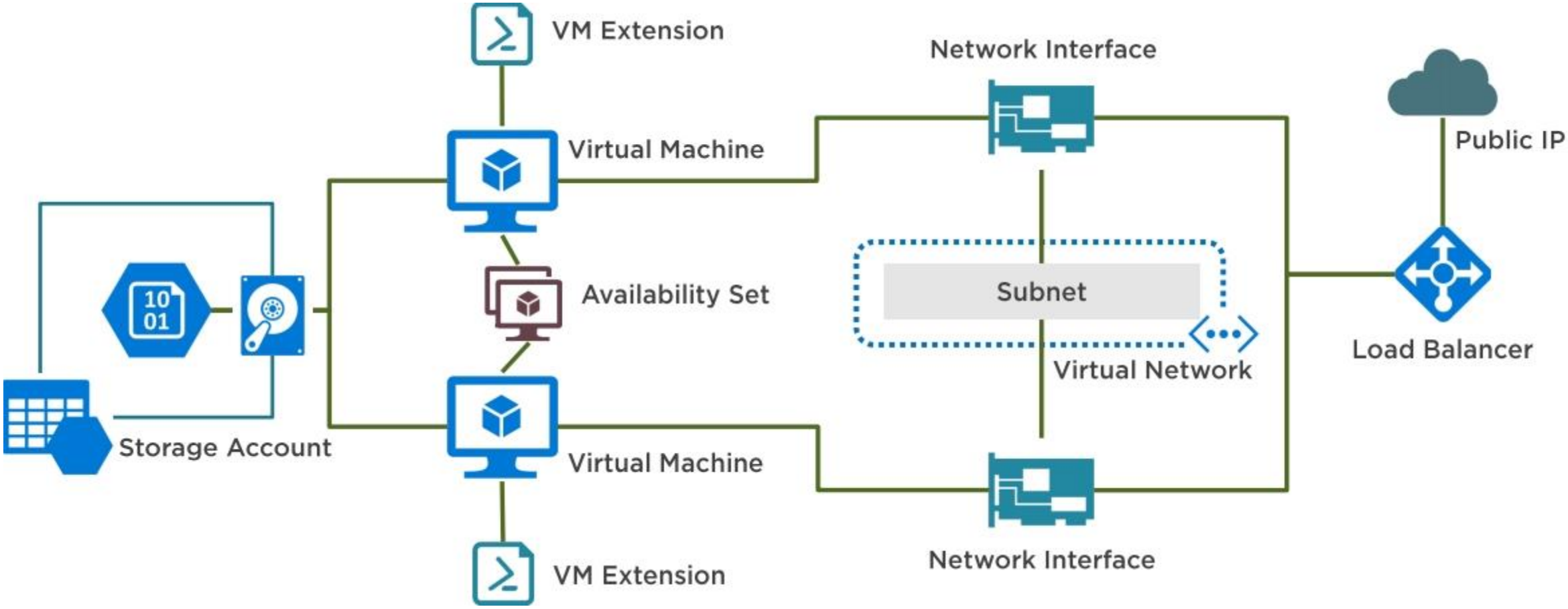
Resource Group Example


V
V
V
Azure SQL DB
Virtual Network
Azure Storage
Azure Website

AZURE RESOURCE MANAGER ARCHITECTURE



RESOURCES AND DEPENDENCIES



Resource Group Limits

- 800 resource groups per subscription
- 800 resources per resource type per resource group
- 15 tags per resource group
- Bound by subscription

RESOURCE PROVIDERS

```
PS C:\WINDOWS\system32> Get-AzureRmResourceProvider


ProviderNamespace : Microsoft.AlertsManagement
RegistrationState : Registered
ResourceTypes     : {alerts, alertsSummary, smartGroups, smartDetectorAlertRules...}
Locations         : {global, North Central US, West Central US, East US...}

ProviderNamespace : Microsoft.AAD
RegistrationState : Registered
ResourceTypes     : {DomainServices, DomainServices/oucontainer, locations, locations/operationresults...}
Locations         : {West US, Central US, East US, South Central US...}

ProviderNamespace : Microsoft.SqlVirtualMachine
RegistrationState : Registered
ResourceTypes     : {SqlVirtualMachineGroups, SqlVirtualMachines, SqlVirtualMachineGroups/AvailabilityGroupListeners, operations...}
Locations         : {Australia Central, Australia Central 2, Australia East, Australia Southeast...}

ProviderNamespace : microsoft.insights
RegistrationState : Registered
ResourceTypes     : {monitoredObjects, dataCollectionRules, dataCollectionRuleAssociations, dataCollectionEndpoints...}
Locations         : {Australia Southeast, Canada Central, Poland Central, Israel Central...}

ProviderNamespace : Microsoft.KeyVault
RegistrationState : Registered
ResourceTypes     : {vaults, vaults/secrets, vaults/accessPolicies, operations...}
Locations         : {North Central US, East US, North Europe, West Europe...}

ProviderNamespace : Microsoft.MachineLearningServices
RegistrationState : Registered
ResourceTypes     : {workspaces/batchEndpoints, workspaces/batchEndpoints/deployments, workspaces, workspaces/capabilityhosts...}
Locations         : {North Central US, Canada Central, Central India, UK South...}

ProviderNamespace : Microsoft.Storage
RegistrationState : Registered
ResourceTypes     : {locations/ActionsRPOperationStatuses, storageAccounts/reports, storageAccounts/storageTaskAssignments,
                    storageAccounts/storageTaskAssignments/reports...}
Locations         : {East US 2, Japan East, Japan West, South India...}
```

RESOURCE TYPES

```
PS C:\WINDOWS\system32> Get-AzureRmResourceProvider -ProviderNamespace Microsoft.Compute | Select-Object ResourceTypes,Locations | Sort-Object ResourceTypes

ResourceTypes                                                Locations
-------------                                                ---------
{availabilitySets}                                           {East US, East US 2, West US, Central US...}
{diskAccesses}                                               {Southeast Asia, East US 2, Central US, West Europe...}
{diskEncryptionSets}                                         {Southeast Asia, East US 2, Central US, West Europe...}
{locations/diskoperations}                                   {Southeast Asia, East US 2, Central US, West Europe...}
{snapshots}                                                  {Southeast Asia, East US 2, Central US, West Europe...}
{disks}                                                      {Southeast Asia, East US 2, Central US, West Europe...}
{galleries/applications/versions}                            {West Central US, South Central US, East US 2, Southeast Asia...}
{restorePointCollections/restorePoints/diskRestorePoints}    {Southeast Asia, East US 2, Central US, West Europe...}
{galleries/applications}                                     {West Central US, South Central US, East US 2, Southeast Asia...}
{locations/galleries}                                        {West Central US, South Central US, East US 2, Southeast Asia...}
{galleries/images/versions}                                  {West Central US, South Central US, East US 2, Southeast Asia...}
{galleries/images}                                           {West Central US, South Central US, East US 2, Southeast Asia...}
{galleries}                                                  {West Central US, South Central US, East US 2, Southeast Asia...}
{locations/capsoperations}                                   {West Central US, South Central US, East US 2, Southeast Asia...}
{locations/artifactPublishers}                               {West Central US, South Central US, East US 2, Southeast Asia...}
{payloadGroups}                                              {South Central US, West Europe}
{virtualMachineScaleSets/disks}                              {East US, East US 2, West US, Central US...}
{cloudServices}                                              {Southeast Asia, East US 2, Central US, West Europe...}
{cloudServices/roles}                                        {Southeast Asia, East US 2, Central US, West Europe...}
{locations/logAnalytics}                                     {East US, East US 2, West US, Central US...}
{locations/vmSizeRecommendations}                            {West Central US, Southeast Asia, East US 2, East US...}
{locations/placementRecommendations}                         {West Central US, Southeast Asia, East US 2, East US...}
{locations/placementScores}                                  {West Central US, Southeast Asia, East US 2, East US...}
{locations/diagnosticOperations}                             {West Central US, Southeast Asia, East US 2, East US...}
{locations/diagnostics}                                      {West Central US, Southeast Asia, East US 2, East US...}
{images}                                                     {Southeast Asia, East US 2, Central US, West Europe...}
{locations/usages}                                           {East US, West US, Central US, South Central US...}
{cloudServices/publicIPAddresses}                            {Southeast Asia, East US 2, Central US, West Europe...}
{cloudServices/roleInstances/networkInterfaces}              {Southeast Asia, East US 2, Central US, West Europe...}
{cloudServices/networkInterfaces}                            {Southeast Asia, East US 2, Central US, West Europe...}
{locations/cloudServiceOsFamilies}                           {Southeast Asia, East US 2, Central US, West Europe...}
{locations/cloudServiceOsVersions}                           {Southeast Asia, East US 2, Central US, West Europe...}
{locations/csoperations}                                     {Southeast Asia, East US 2, Central US, West Europe...}
{cloudServices/roleInstances}                                {Southeast Asia, East US 2, Central US, West Europe...}
{sharedVMImages/versions}                                    {West Central US, South Central US, East US 2, Southeast Asia...}
{hostGroups}                                                 {Central US, East US 2, West Europe, Southeast Asia...}
{sharedVMImages}                                             {West Central US, South Central US, East US 2, Southeast Asia...}
{locations/sharedGalleries}                                  {East US, East US 2, West US, Central US...}
{locations/virtualMachines}                                  {East US, East US 2, West US, Central US...}
{locations/runCommands}                                      {East US, East US 2, West US, Central US...}
{locations/vmSizes}                                          {East US, East US 2, West US, Central US...}
{locations/operations}                                       {East US, East US 2, West US, Central US...}
```

. . .

. . .

RESOURCE REST APIS

https://management.azure.com/subscriptions/{subscription-id}/providers/{provider-name}?&api-version={api-version}

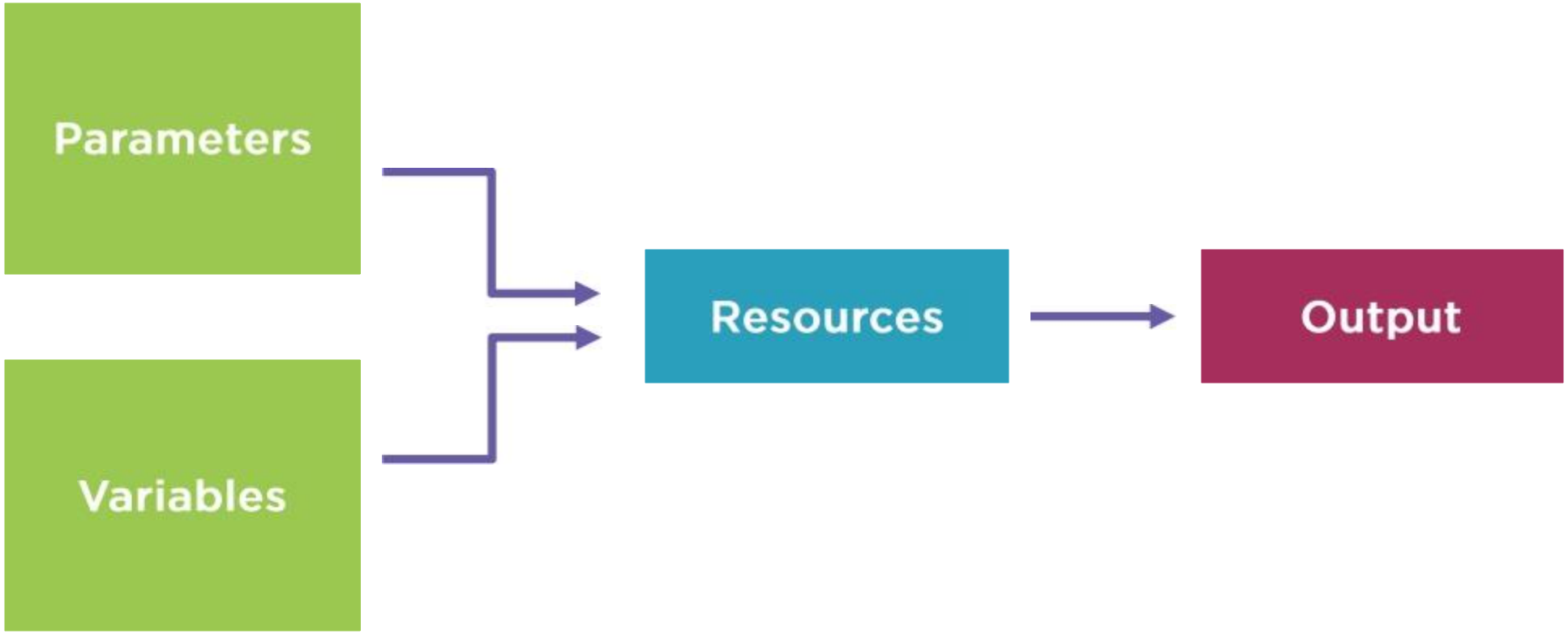
```
PS C:\WINDOWS\system32> ((Get-AzureRmResourceProvider -ProviderNamespace Microsoft.Compute).ResourceTypes | Where-Object {$_.ResourceTypeName -eq 'virtualMachines'}).ApiVersions


2024-11-01
2024-07-01
2024-03-01
2023-09-01
2023-07-01
2023-03-01
2022-11-01
2022-08-01
2022-03-01
2021-11-01
2021-07-01
2021-04-01
2021-03-01
2020-12-01
2020-06-01
2019-12-01
2019-07-01
2019-03-01
2018-10-01
2018-06-01
2018-04-01
2017-12-01
2017-03-30
2016-08-30
2016-04-30-preview
2016-03-30
2015-06-15
2015-05-01-preview
```

JSON TEMPLATES

- Provide a scalable and repeatable method for deploying Azure resources

- Behind the scenes all ARM resources are built on JSON

JSON FLOW



SAMPLE JSON TEMPLATE

```
{
    $schema: https://schema.management.azure.com/schemas/2019-04-01/deploymentTemplate.json#,
    contentVersion: 1.0.0.0,
    parameters: {
        storageAccountName: {
            type: string,
            defaultValue: mystorageacctdemo,
            metadata: {
                description: Name of the storage account
            }
        },
        location: {
            type: string,
            defaultValue: [resourceGroup().location],
            metadata: {
                description: Location for all resources
            }
        }
    },
    variables: {
        storageAccountType: Standard_LRS
    },
    resources: [
        {
            type: Microsoft.Storage/storageAccounts,
            apiVersion: 2022-09-01,
            name: [parameters('storageAccountName')],
            location: [parameters('location')],
            sku: {
                name: [variables('storageAccountType')]
            },
            kind: StorageV2,
            properties: {}
        }
    ],
    outputs: {
        storageAccountEndpoint: {
            type: string,
            value: [reference(parameters('storageAccountName')).primaryEndpoints.blob]
        }
    }
}
```

Template Highlights:

Parameters: Accepts storageAccountName and location.

Variables: Defines the storage account type (Standard_LRS).

Resources: Deploys a StorageV2 account using the provided parameters and variable.

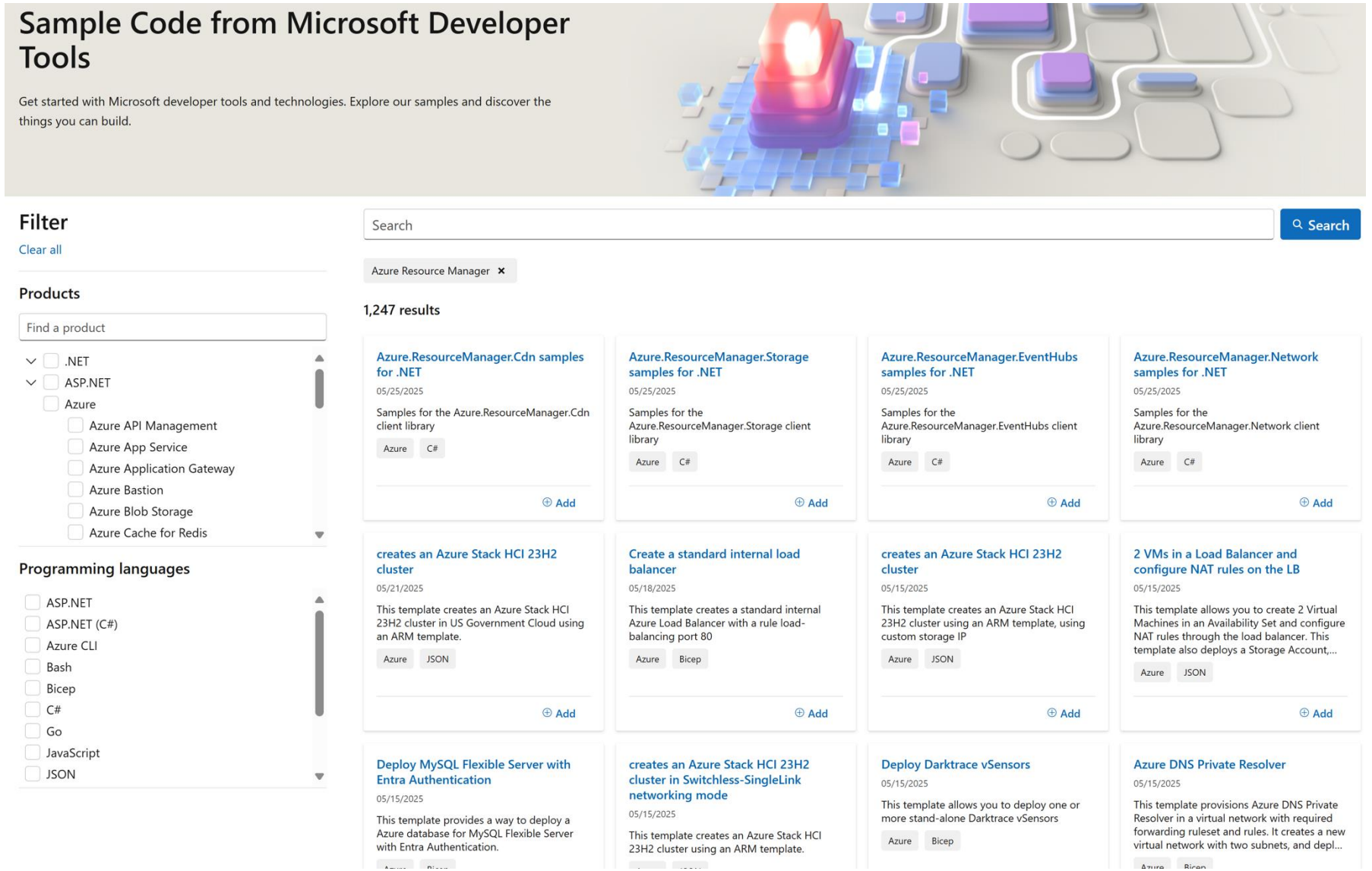
Outputs: Returns the blob endpoint of the created storage account.

SAMPLE BICEP TEMPLATE

```
param storageAccountName string
param location string = resourceGroup().location

var storageAccountType = 'Standard_LRS'

resource storageAccount 'Microsoft.Storage/storageAccounts@2022-
09-01' = {
  name: storageAccountName
  location: location
  sku: {
    name: storageAccountType
  }
  kind: 'StorageV2'
  properties: {}
}

output blobEndpoint string =
storageAccount.properties.primaryEndpoints.blob
```

Template Highlights:
Parameters:
Accepts storageAccountName and location.
Variables: Defines the storage account type (Standard_LRS).
Resources: Deploys a StorageV2 account using the provided parameters and variable.
Outputs: Returns the blob endpoint of the created storage account.

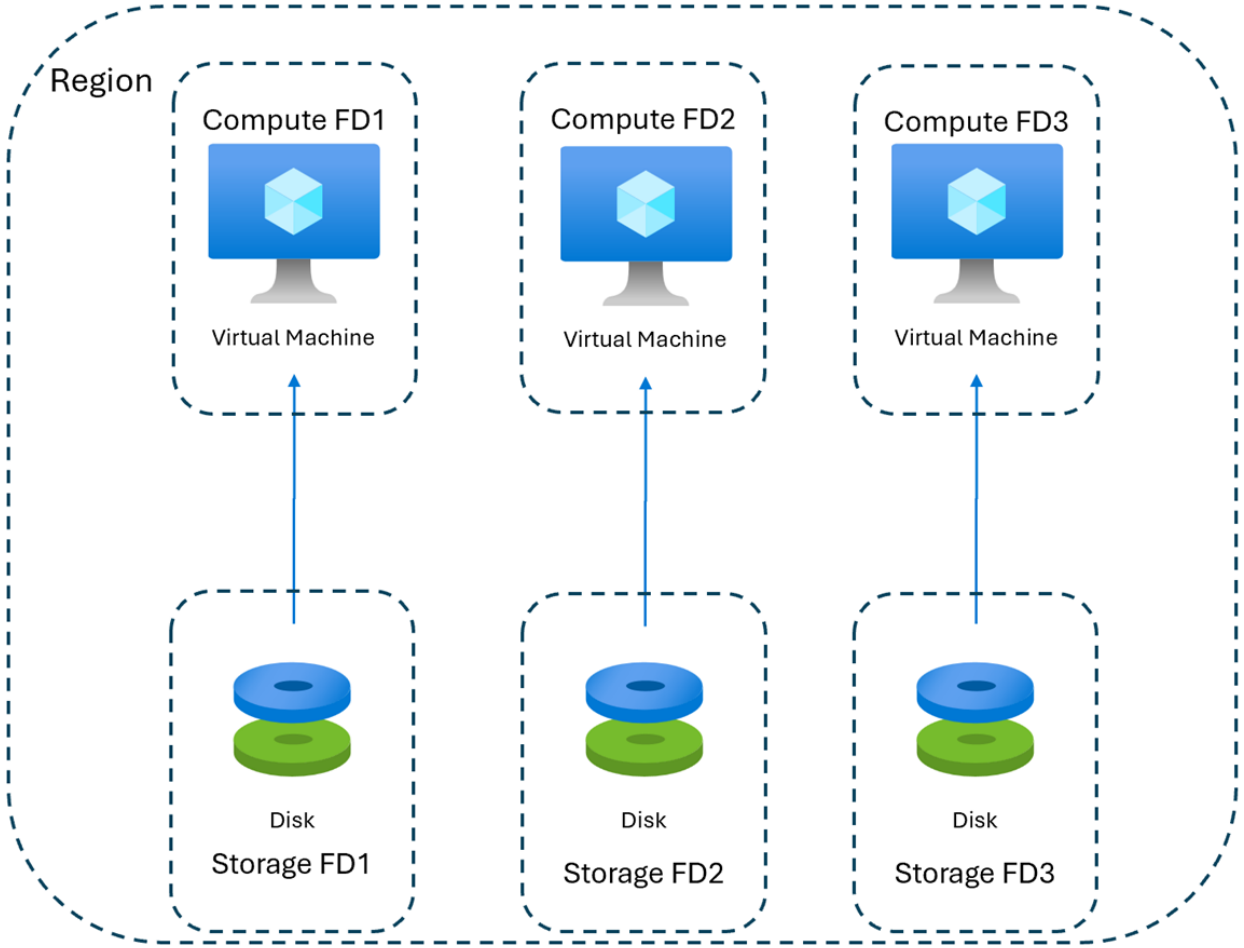
AZURE QUICKSTART TEMPLATES

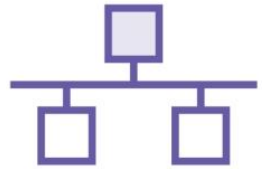


LAB: ARM Template

- Create ARM template
- Create linked template for storage resources
- Upload linked template to Azure Blog Storage
  - Generate SAS token
- Modify the main template to call linked template
- Modify main template to update dependencies
- Upload main template to Azure Blog Storage
- Deployed resources to Azure using AZ CLI

AVAILABILITY SETS





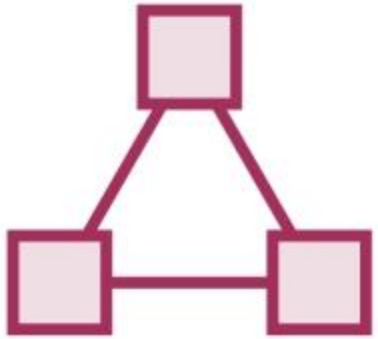
VMs are placed on nodes in a rack

Failures can occur at node and rack levels

Maintenance is also required on hosts and VMs are **not** live migrated

AVAILABILITY SETS

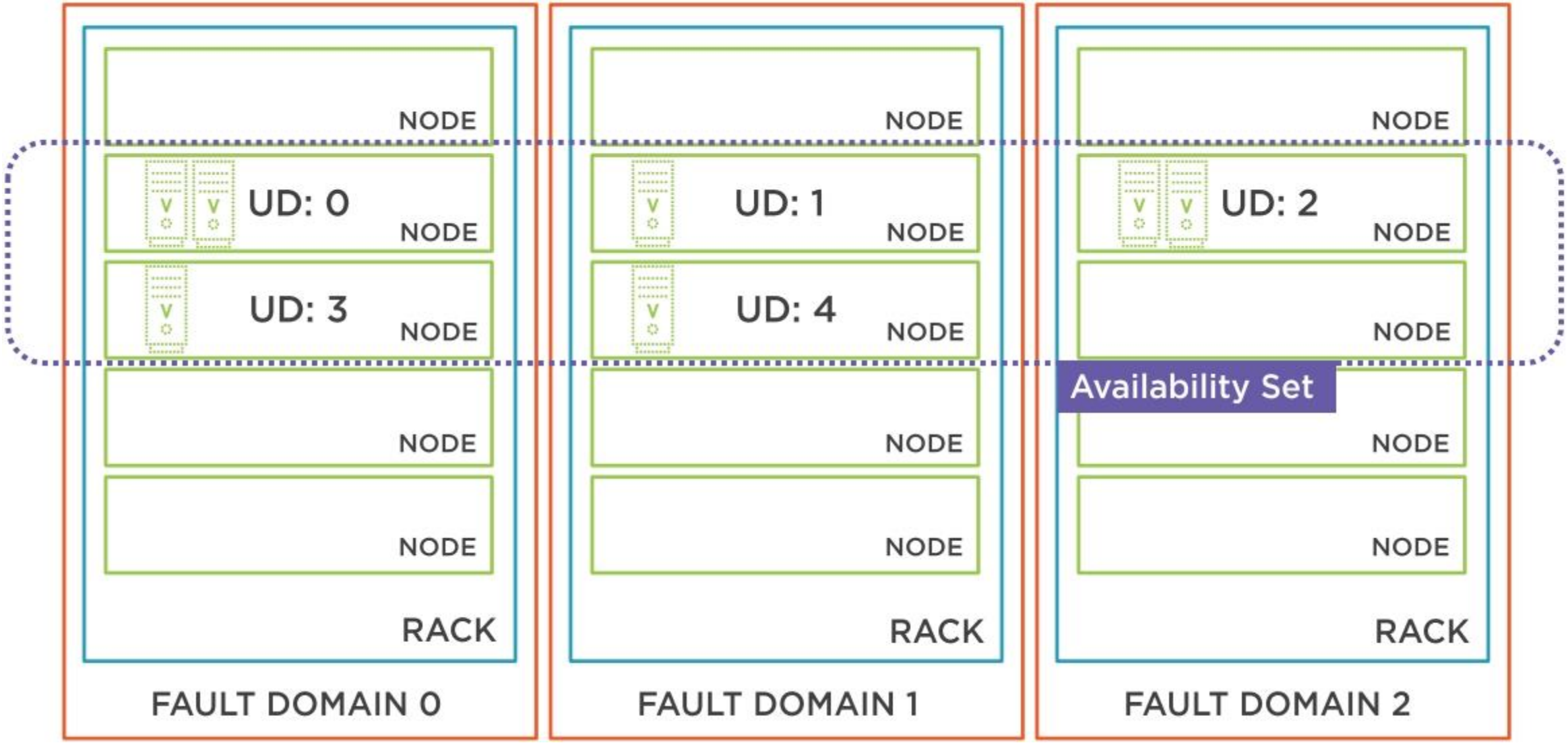
To ensure availability of services, always deploy minimum 2 instances of any service and place in a unique availability set



This ensures VMs are spread over three fault domains (racks) and five (by default) update domains

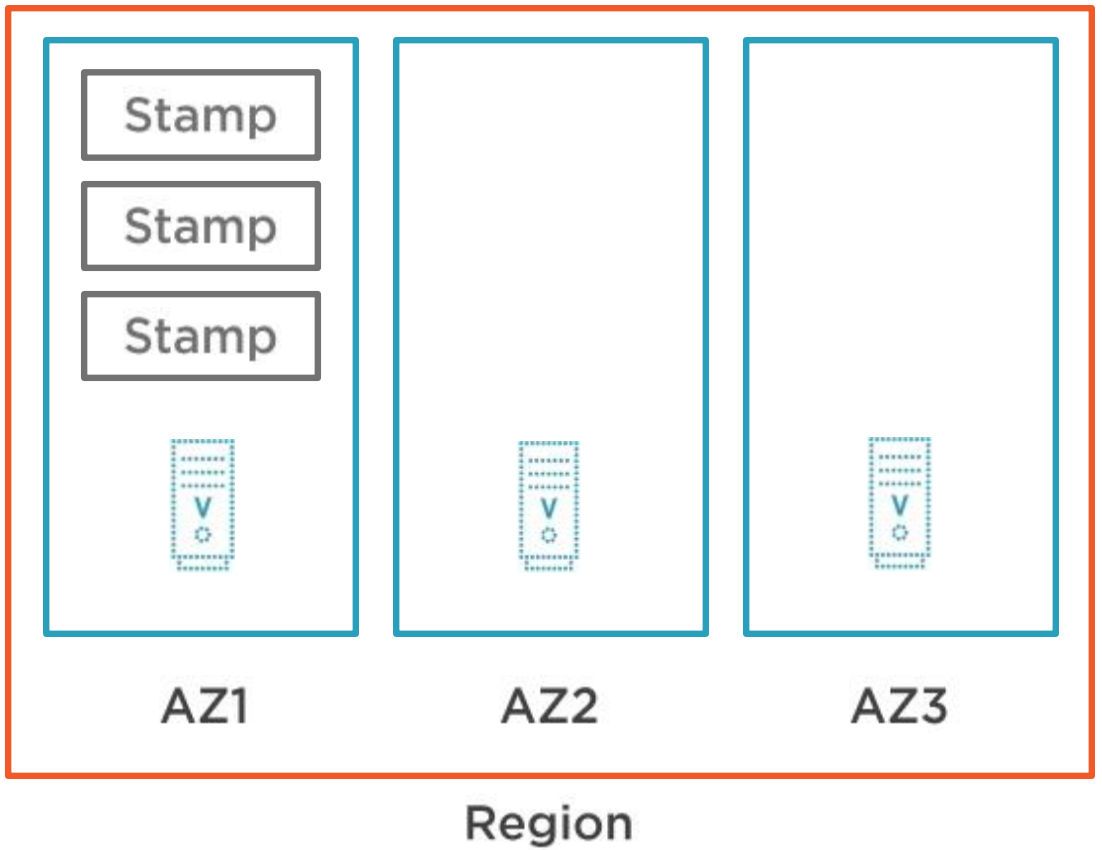
Must have minimum 2 VMs to receive SLA of 99.95%

AVAILABILITY SETS



AVAILABILITY ZONES

- Regions are broken up into physically separate AZs
- AZs have indepenent power, cooling and networking
- 3 AZs are exposed per subscription
- VMs spread over AZs receive 99.99% SLA
- Each AZ can be thought of as separate fault domain and update domain
- Virtual networks span AZs

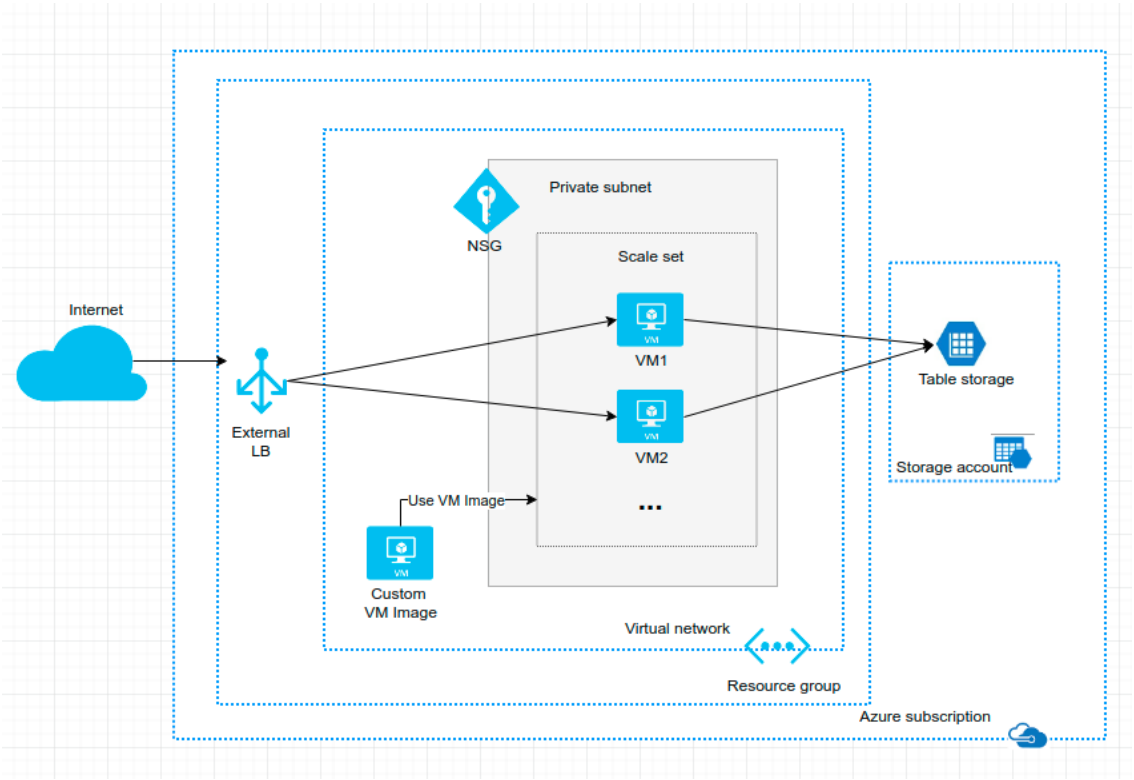


REGIONS

60+ regions, 140 countries

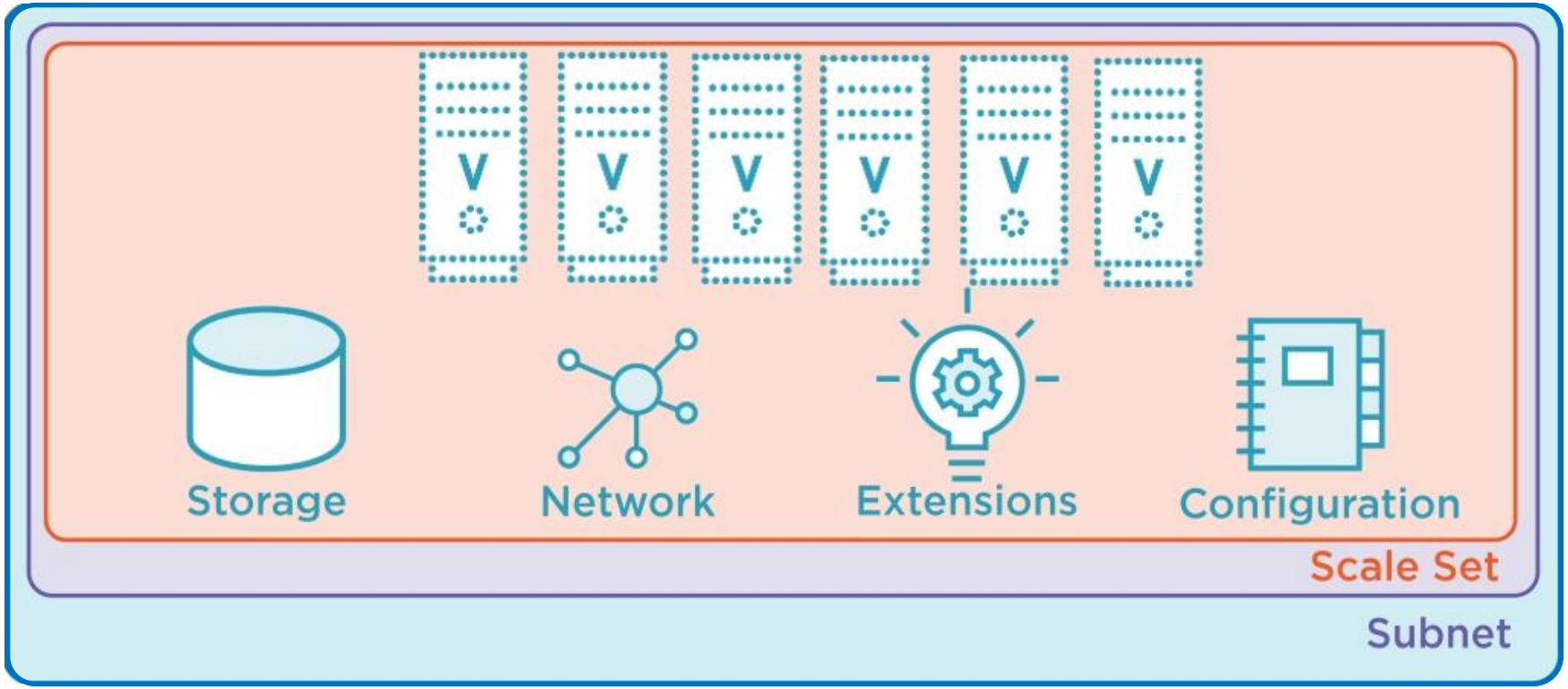
SCALE SETS

- Enables large scale deployment of VMs from single gold image with automatic configuration
- Supports auto-scale based on metrics and schedules
- Scale sets can be updated without taking down the entire set



SCALE SETS

- Azure VM Scale Sets enable the entire deployment to be managed in a simple fashion
- Enable rollout of updates without taking down the entire service.

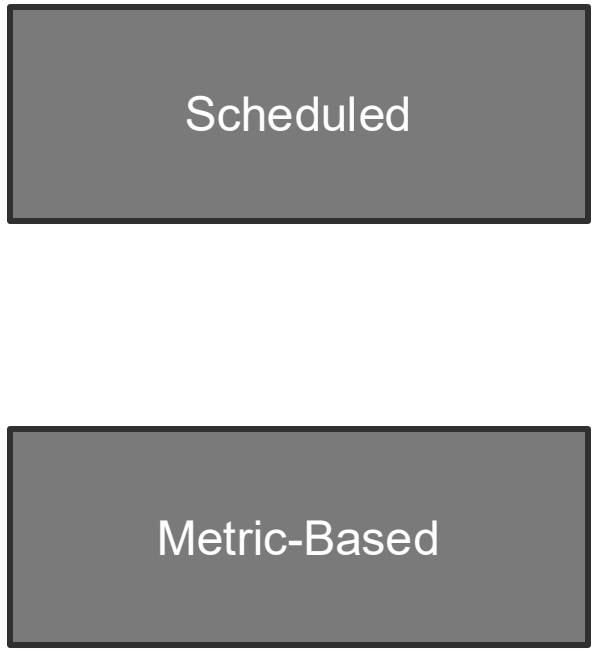


SCALE SETS LIMITATIONS

- 1000 VMs per Scale Set maximum and use with Azure Standard Load Balancer for matching scale

- 2000 Scale Sets per region per subscription

Auto-scaling Scale Sets



Scale Sets support many types of scale

- On a schedule both specific day and recurrence (enables scaling ahead of the load increase)
- Based on resource metrics
- Can combine (schedule rules take precedence over metric rules)

POP QUIZ:

You need high availability in a single region and want to support 200 instances in a scale set. Which option should you choose?
A. Regional scale set (non-zonal) without placement groups
B. Zonal scale set with true Availability Zones
C. Regional scale set with placement groups
D. Single VM in an Availability Set

POP QUIZ:

You need high availability in a single region and want to support 200 instances in a scale set. Which option should you choose?

A. Regional scale set (non-zonal) without placement groups

B. Zonal scale set with true Availability Zones

C. Regional scale set with placement groups

D. Single VM in an Availability Set

Notifications & Logs

Can notify on scaling changes

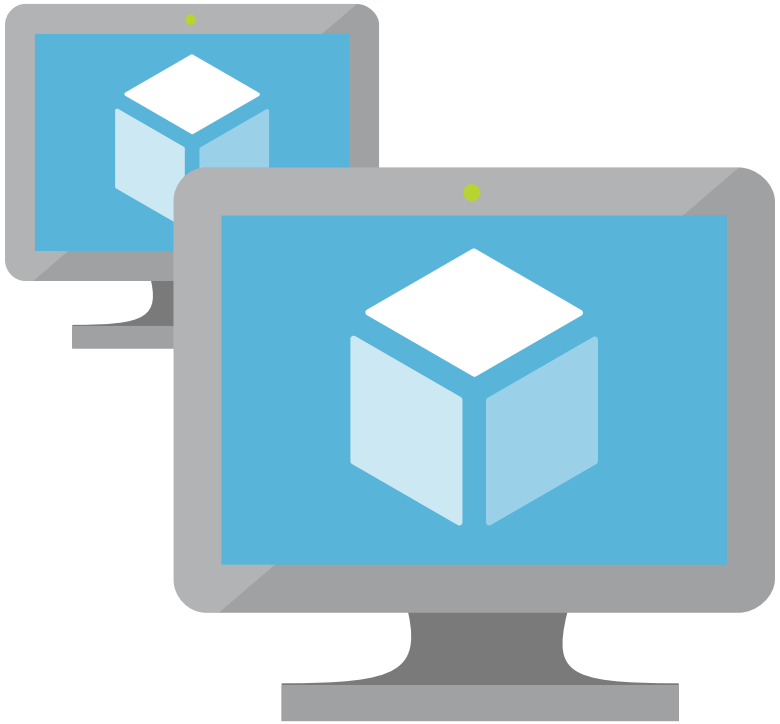
- Email a team
- Integrate with external APIs via event hooks

Can create JSON templates just to change the scale

Audit logs will show scale history actions

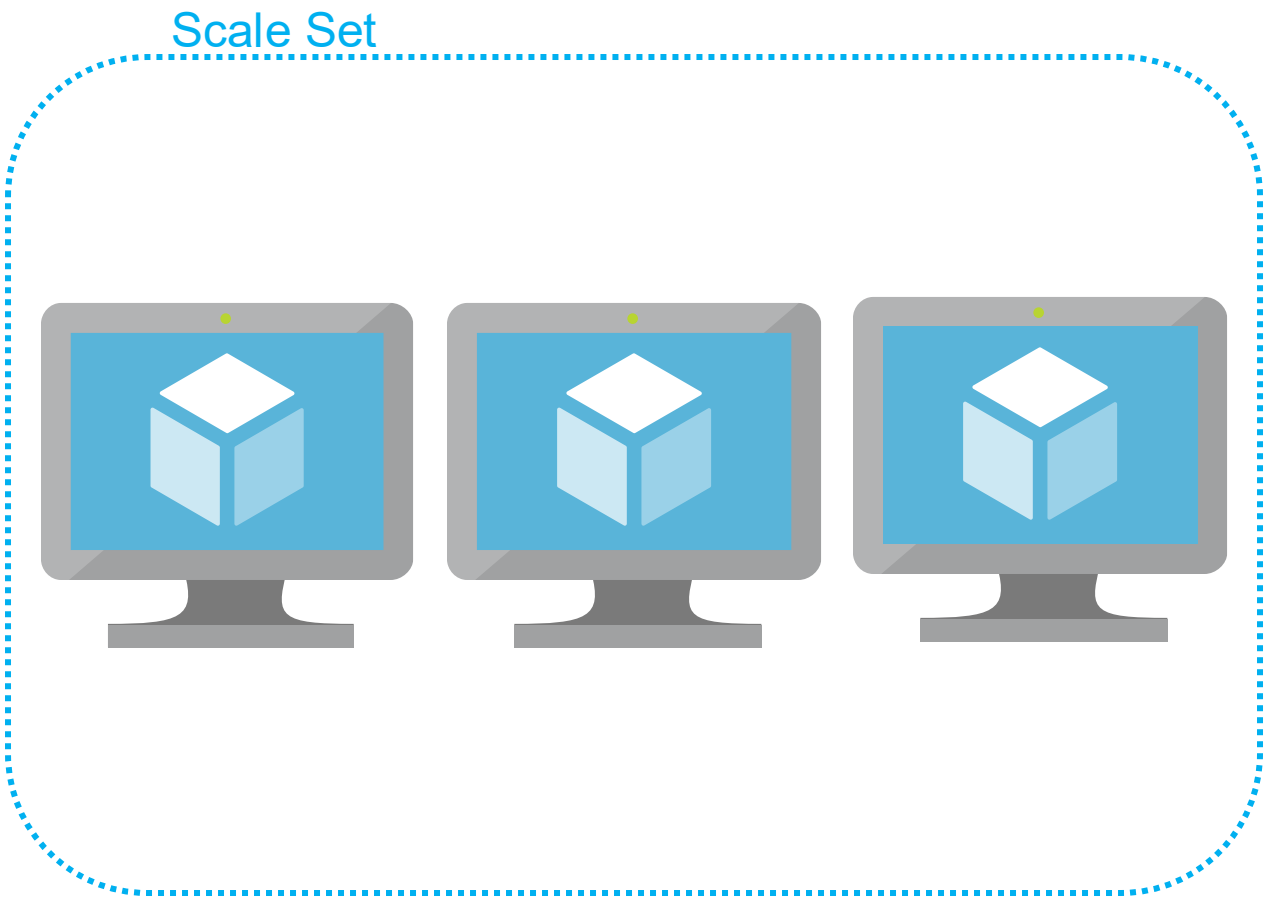
SCALE VERTICALLY

- Can be cheaper to scale vertically
  - Add more memory, CPU
  - Save money by scaling down
- Requires VM to reboot



SCALE HORIZONTALLY

- Recommended for cloud applications
- Added redundancy
- No downtime

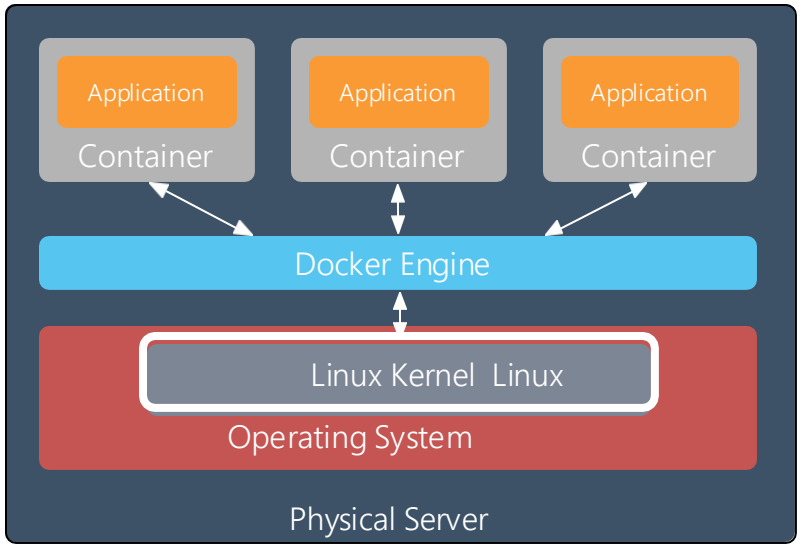


LAB: Deploy VMs

- Deploy Azure VMs by using the Azure portal, Azure PowerShell, and Azure Resource Manager templates
- Configure networking settings of Azure VMs running Windows and Linux operating systems
- Deploy and configure Azure VM scale sets

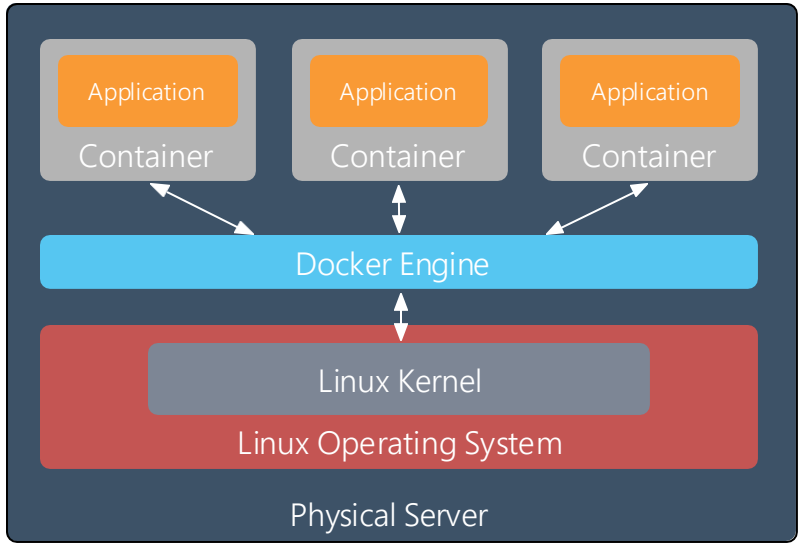
Azure Container Service

CONTAINERS



Shared kernel on the host to run multiple guest applications

CONTAINERS OVER VMS

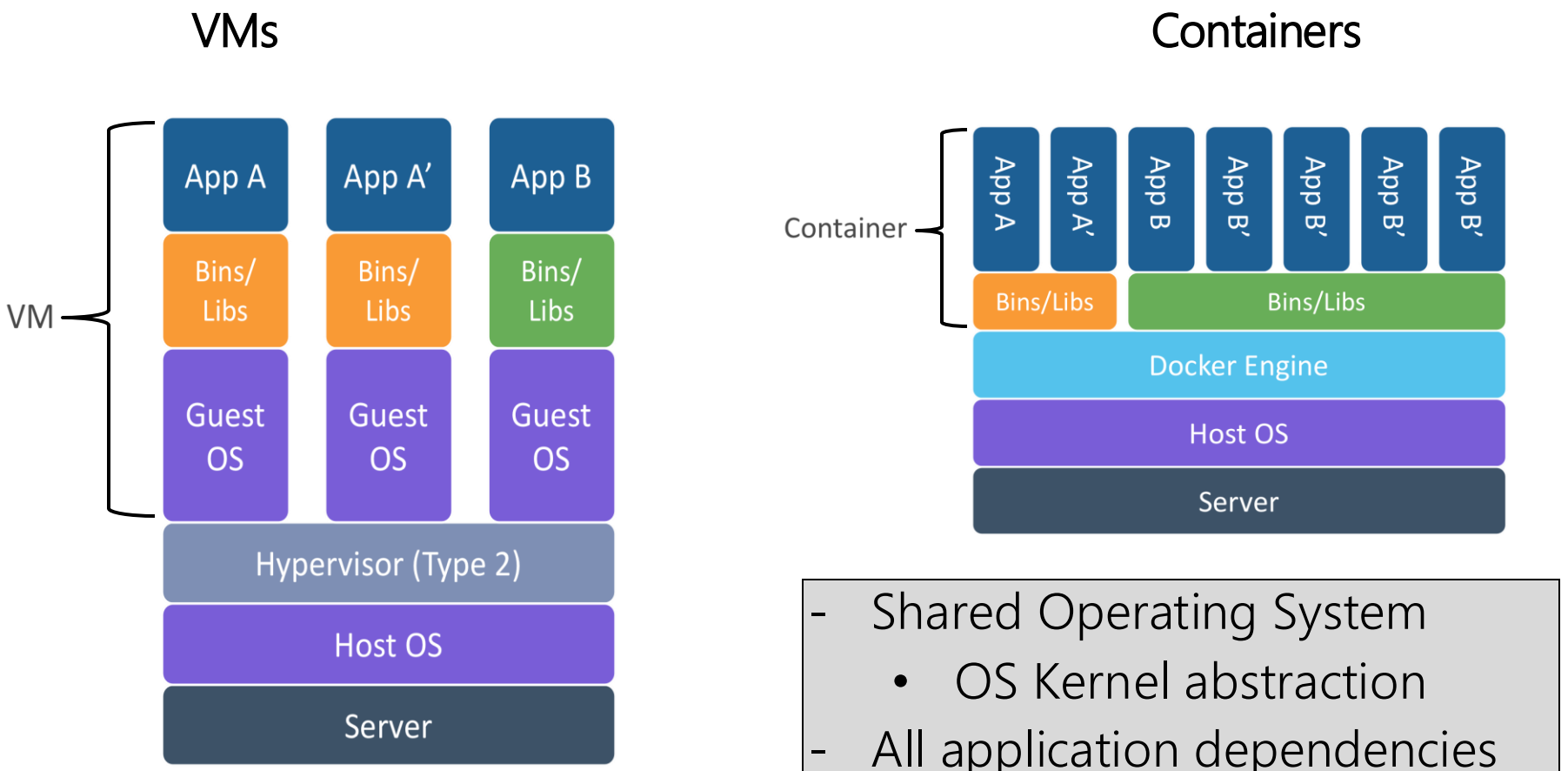


Shared kernel on the host to run multiple guest applications

Advantages over VMs

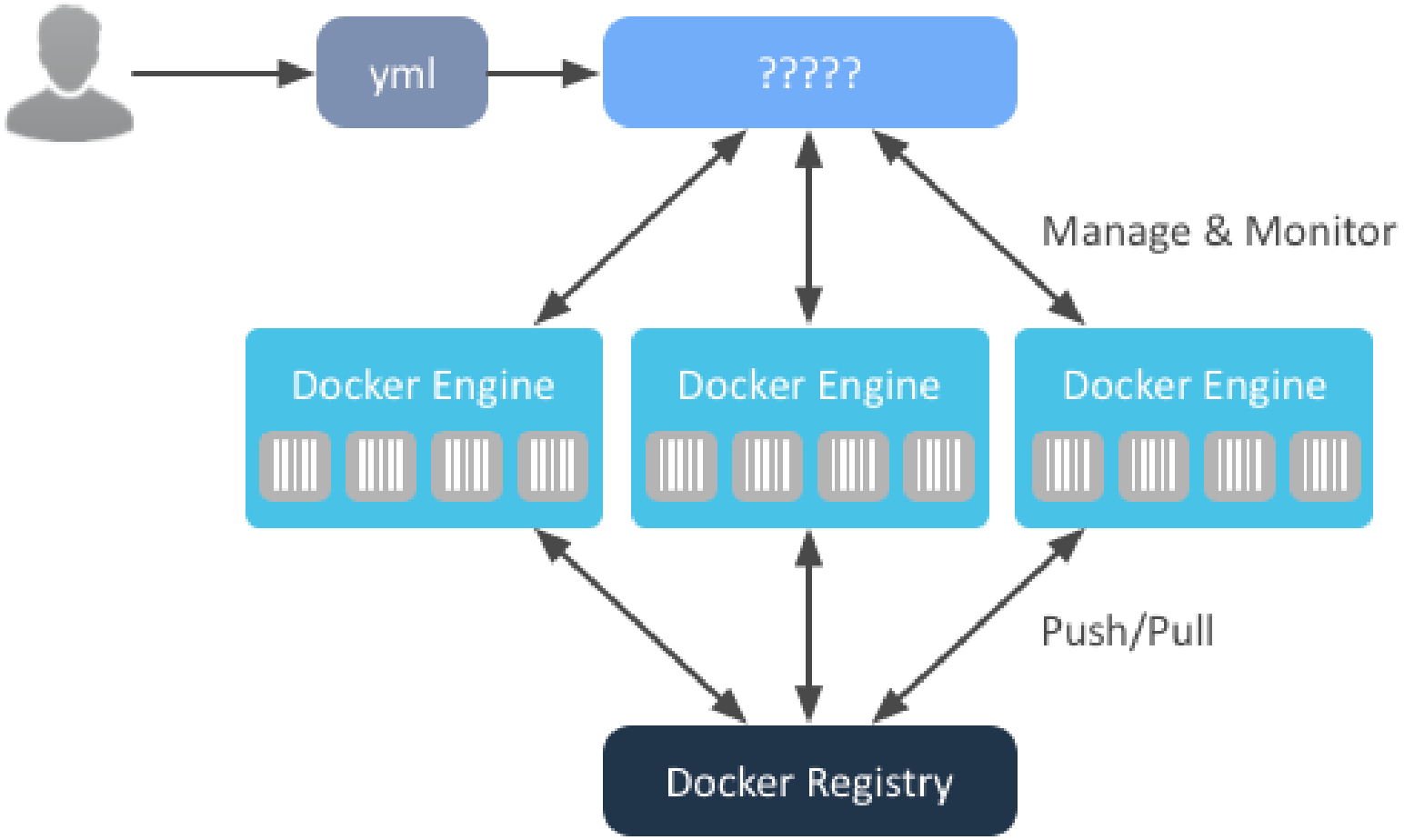
- Containers are more lightweight
- No need to install a guest Operating System
- Less CPU, RAM, storage overhead
- More containers per machine
- Greater portability

VM-BASED VS CONTAINER-BASED ARCHITECTURE



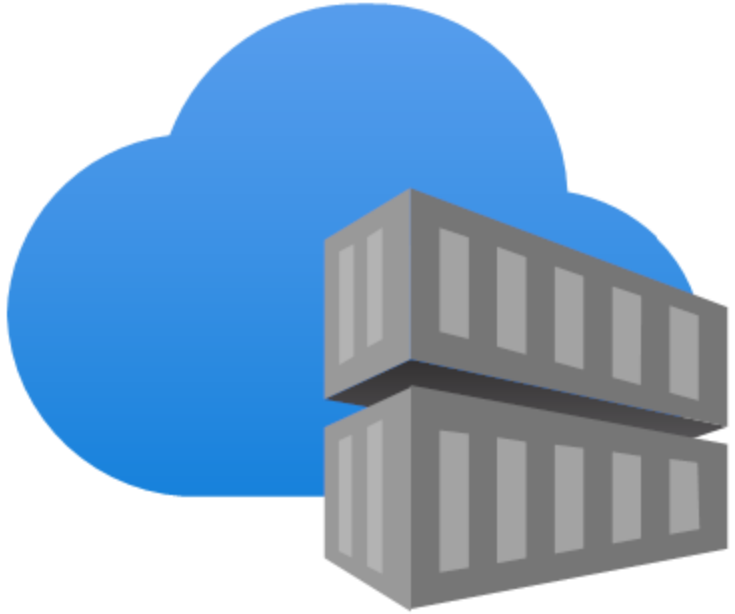
- Shared Operating System
  - OS Kernel abstraction
- All application dependencies contained w/in the container

END GOAL: MANAGE AS SINGLE POOL OF RESOURCES



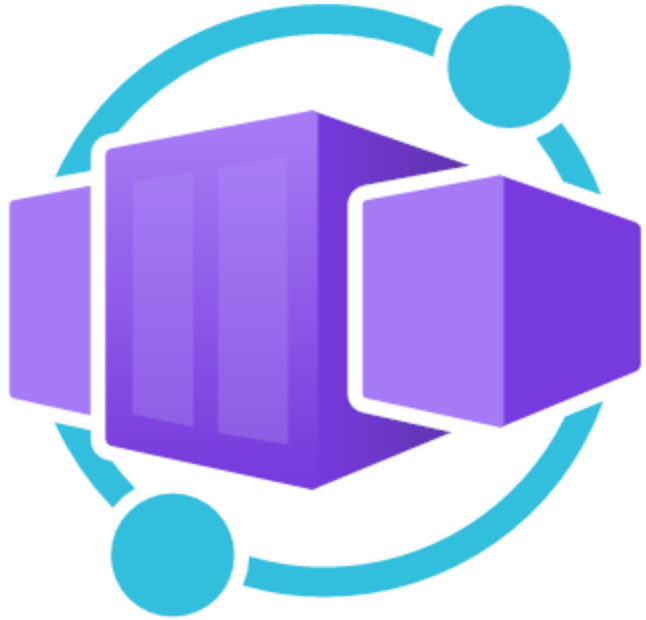
AZURE CONTAINER REGISTRY

Azure Container Registry (ACR) is a **fully managed, geo-replicated OCI-compliant registry** that simplifies the entire container lifecycle—from build to deployment.



AZURE CONTAINER APPS

Azure Container Apps is a **serverless container platform** designed for running **microservices, APIs, background jobs, and AI-powered applications** without managing infrastructure.

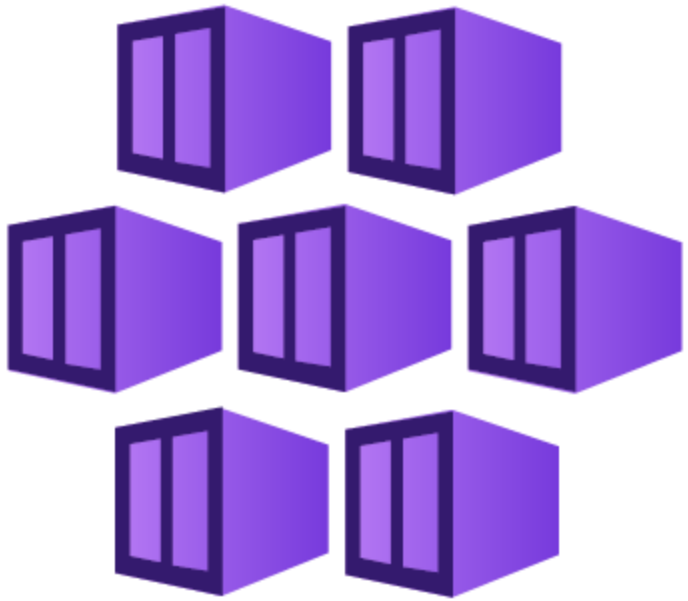


AZURE FUNCTIONS

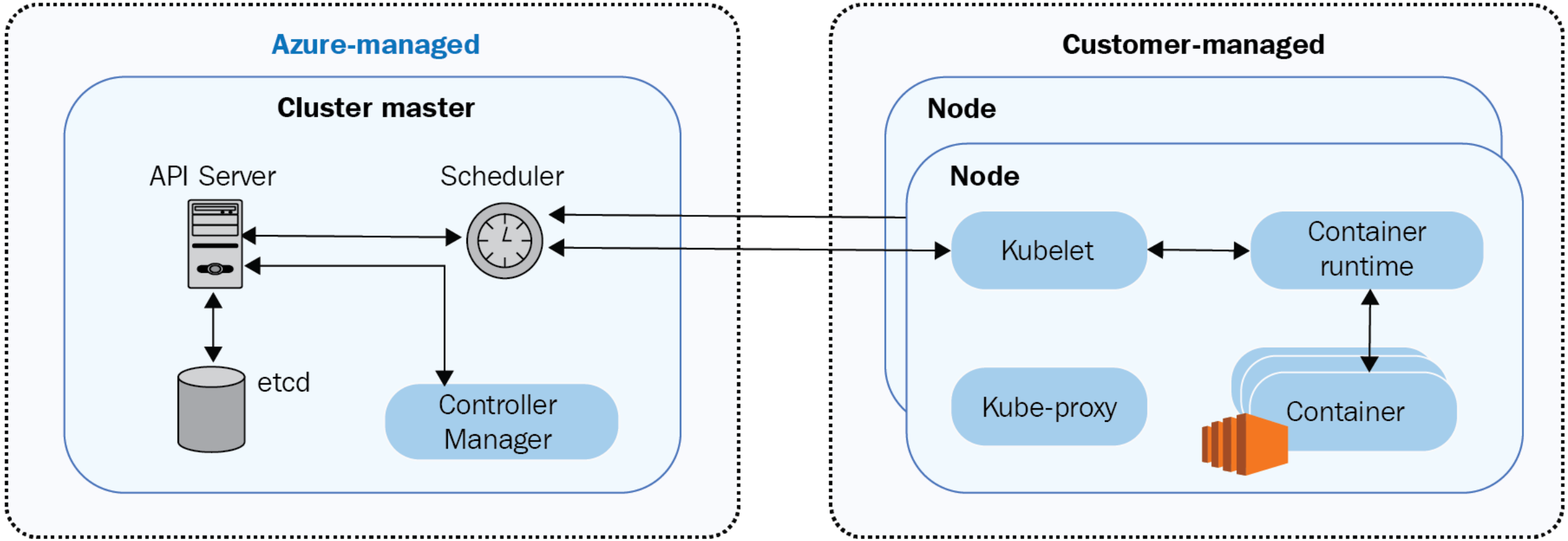
Execute code on-demand with a pay-per-use model, ideal for variable workloads.

AZURE KUBERNETES SERVICE (AKS)

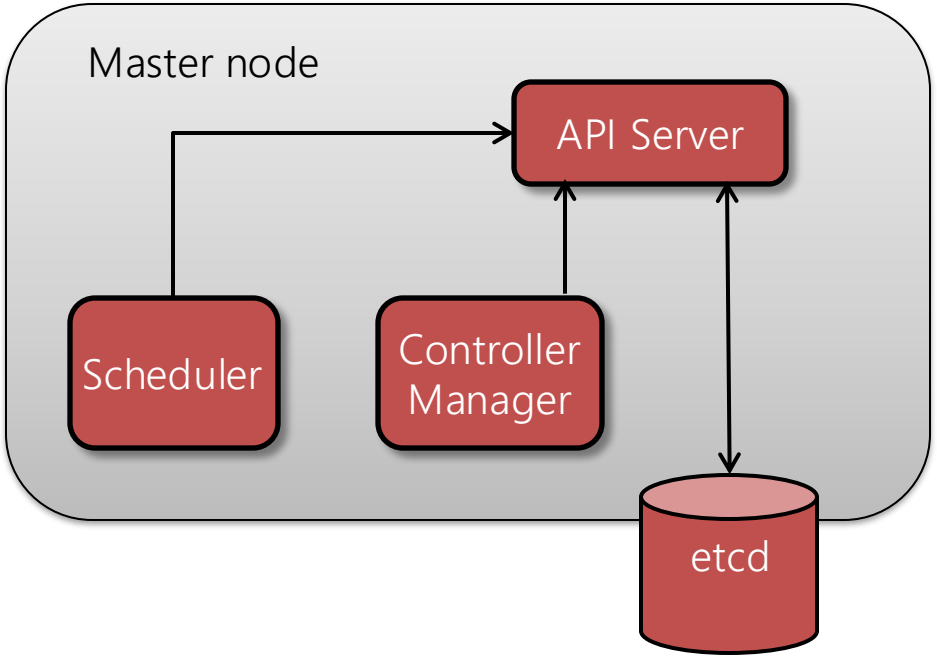
Azure Kubernetes Service (AKS) is a **fully managed Kubernetes platform** that simplifies the deployment, management, and scaling of containerized applications.

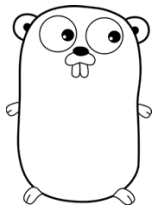


AKS ARCHITECTURE



KUBERNETES MASTER COMPONENTS



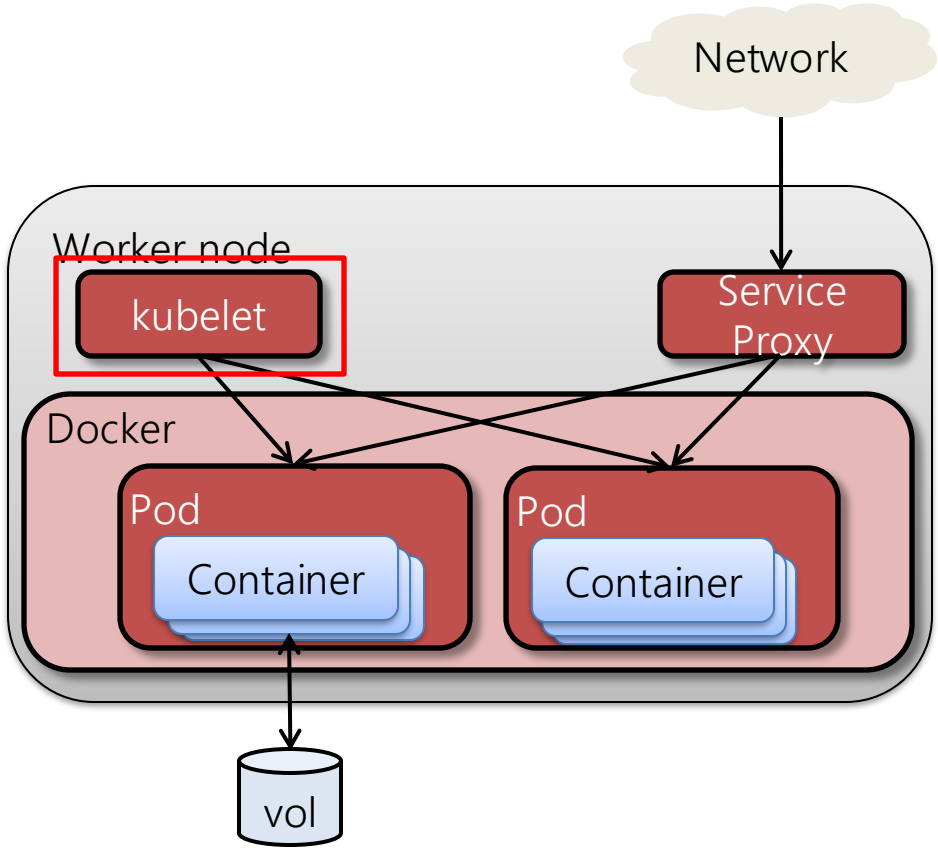


K8s components written in Go (golang.org)

- **API Server (kube-apiserver)**: exposes the Kubernetes REST API, and can be scaled horizontally
- **Scheduler (kube-scheduler)**: selects nodes for newly created pods to run on
- **Controller manager (kube-controller-manager)**: runs background controller processes for the system to enforce declared object states, e.g. Node Controller, Replication Controller, …
- **Persistent data store (etcd)**: all K8s system data is stored in a distributed, reliable key-value store. etcd may run on separate nodes from the master

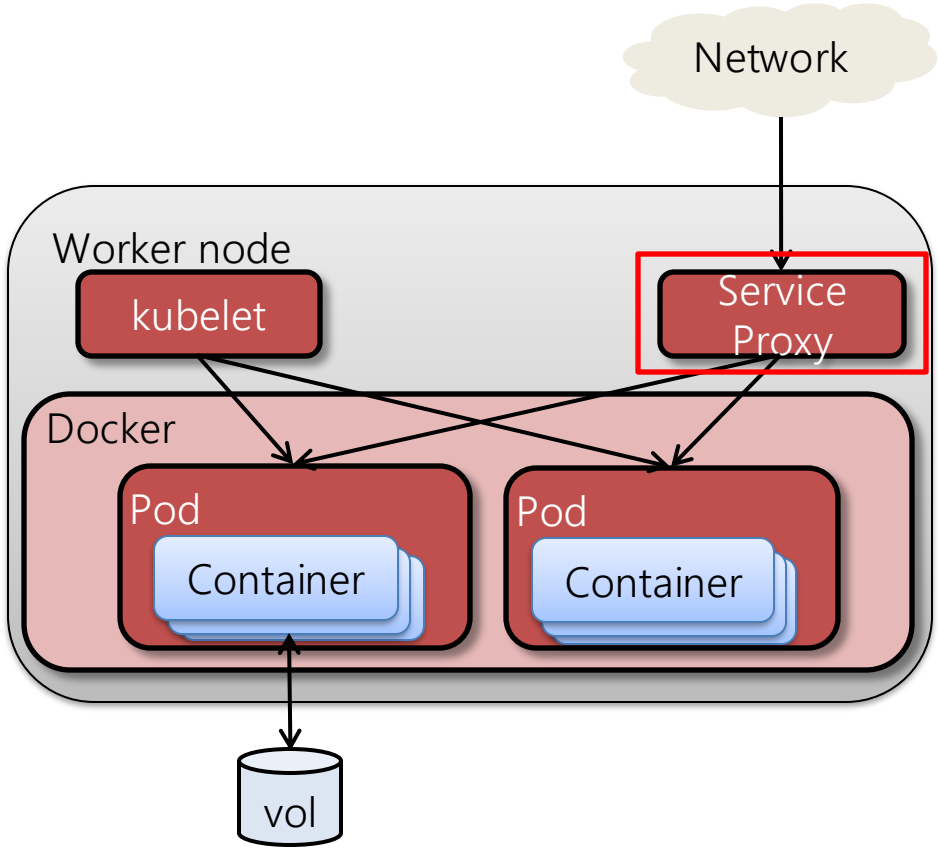
KUBERNETES WORKER COMPONENTS

- **kubelet**: local K8s agent that is responsible for operations on the node, including
  - Watching for pod assignments
  - Mounting pod required volumes
  - Running a pod's containers
  - Executing container liveness probes
  - Reporting pod status to system
  - Reporting node status to system
- **Service proxy (kube-proxy)**: enables K8s service abstractions by maintaining host network rules and forwarding connections
- **Docker**: runs the containers



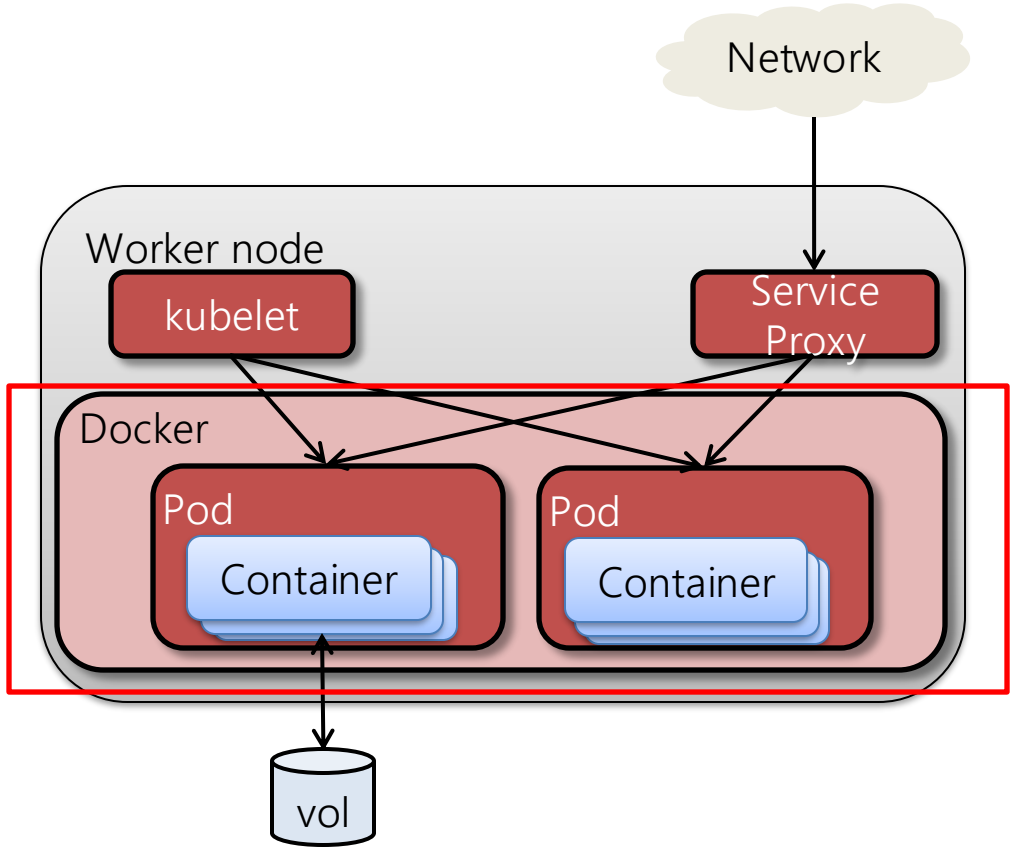
KUBERNETES WORKER COMPONENTS

- **kubelet**: local K8s agent that is responsible for operations on the node, including
  - Watching for pod assignments
  - Mounting pod required volumes
  - Running a pod's containers
  - Executing container liveness probes
  - Reporting pod status to system
  - Reporting node status to system
- **Service proxy (kube-proxy):** enables K8s service abstractions by maintaining host network rules and forwarding connections
- **Docker**: runs the containers

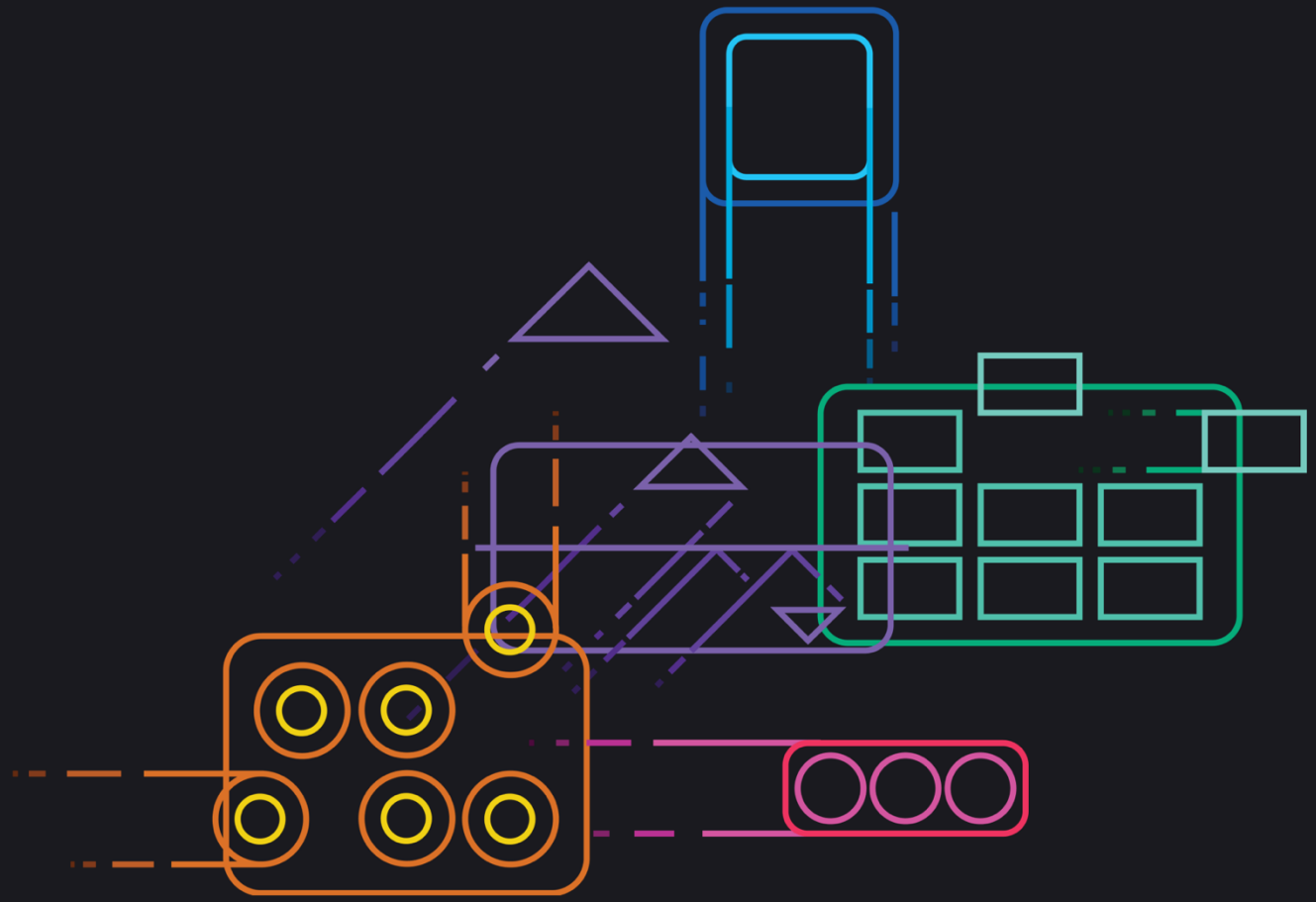


KUBERNETES WORKER COMPONENTS

- **kubelet**: local K8s agent that is responsible for operations on the node, including
  - Watching for pod assignments
  - Mounting pod required volumes
  - Running a pod's containers
  - Executing container liveness probes
  - Reporting pod status to system
  - Reporting node status to system
- **Service proxy (kube-proxy)**: enables K8s service abstractions by maintaining host network rules and forwarding connections
- **Docker**: container runtime

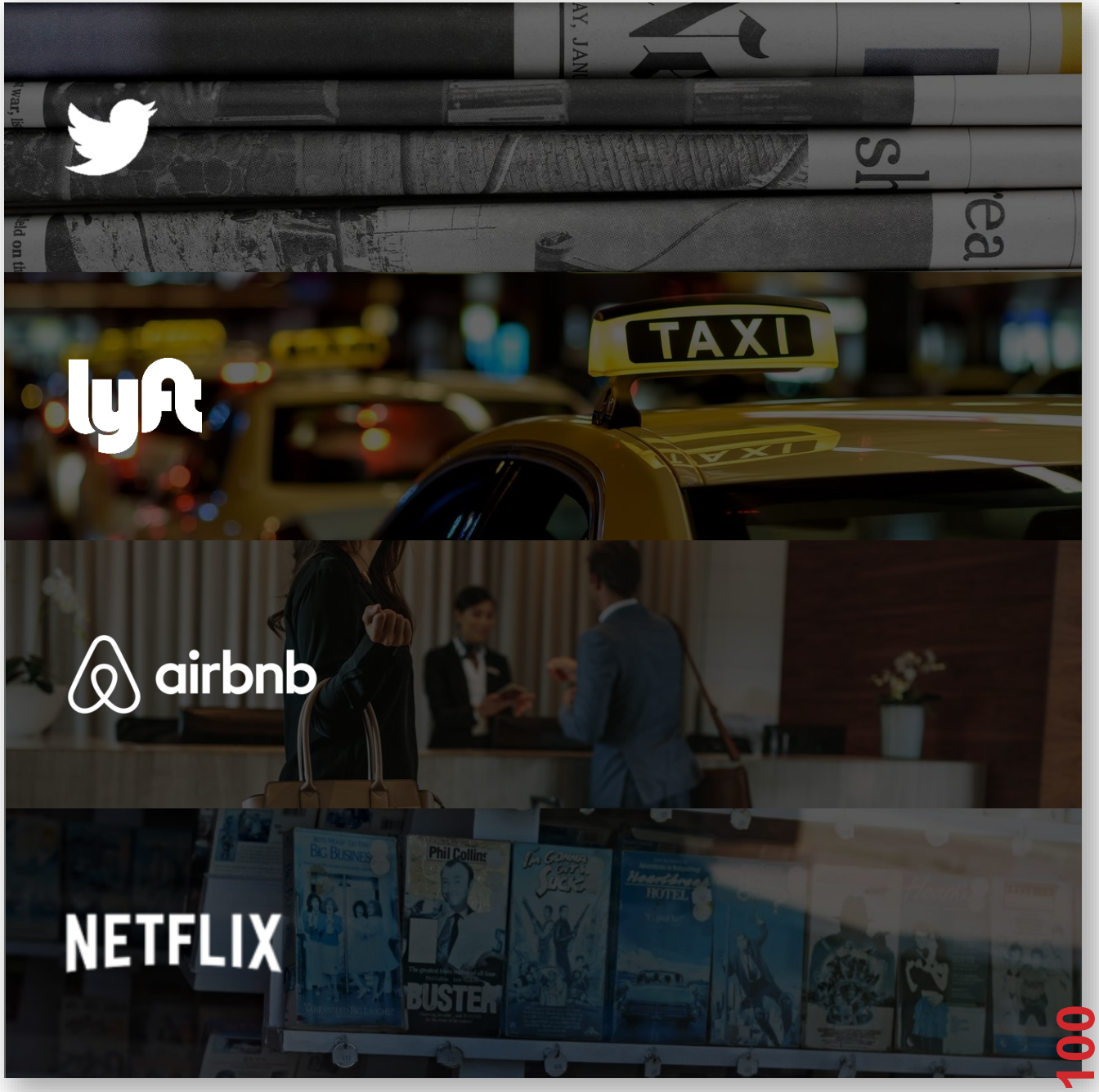


Azure Devops



50% of Fortune 500 in 2000 now replaced by organizations that use technology to deliver customer value faster, better, and cheaper

The Fortune 500 of today face the same challenge tomorrow

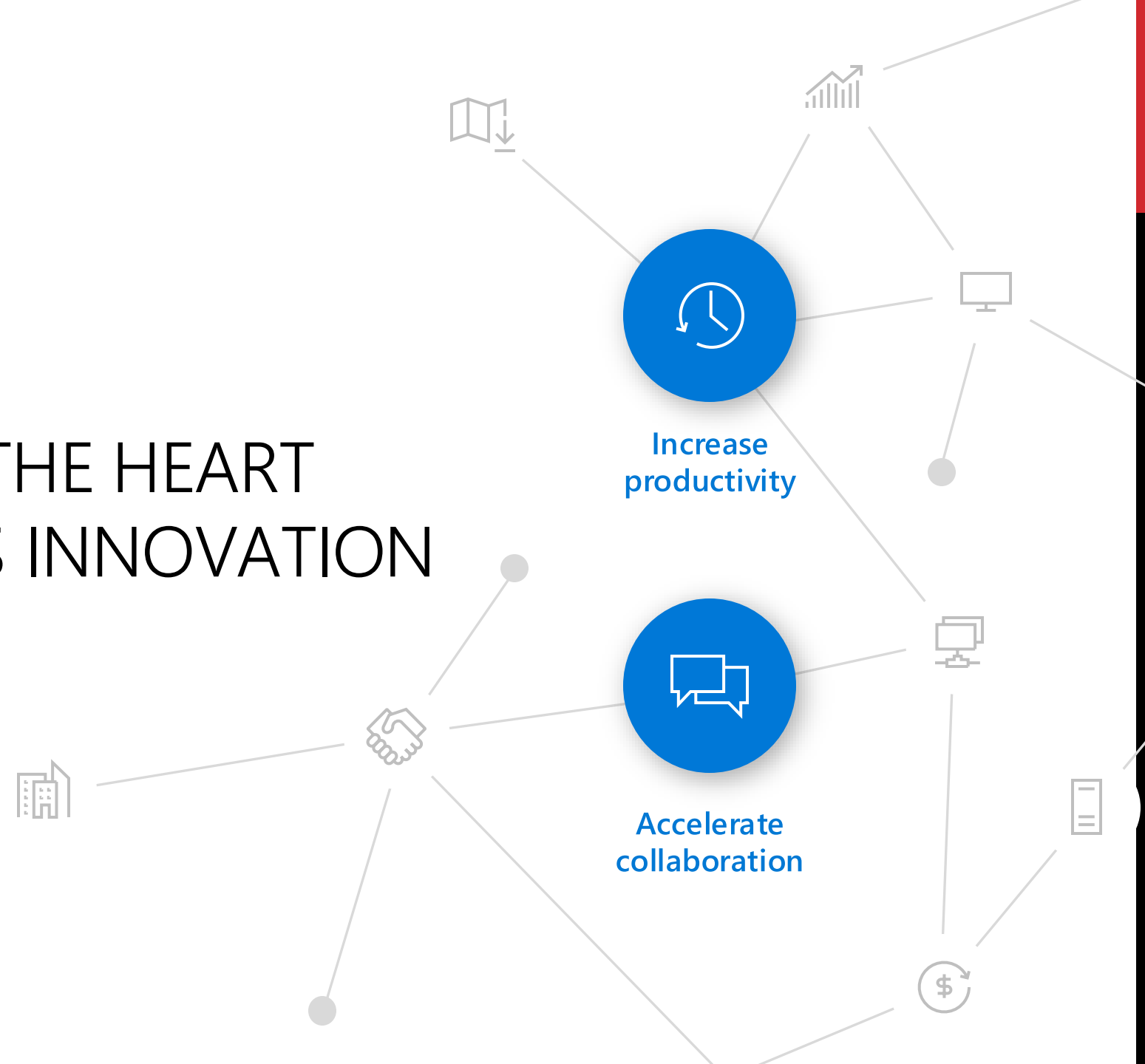


PRODUCT INNOVATIONS NEED TO REACH CUSTOMERS QUICKLY, AND STAY AVAILABLE

SOFTWARE DELIVERY PARADOX

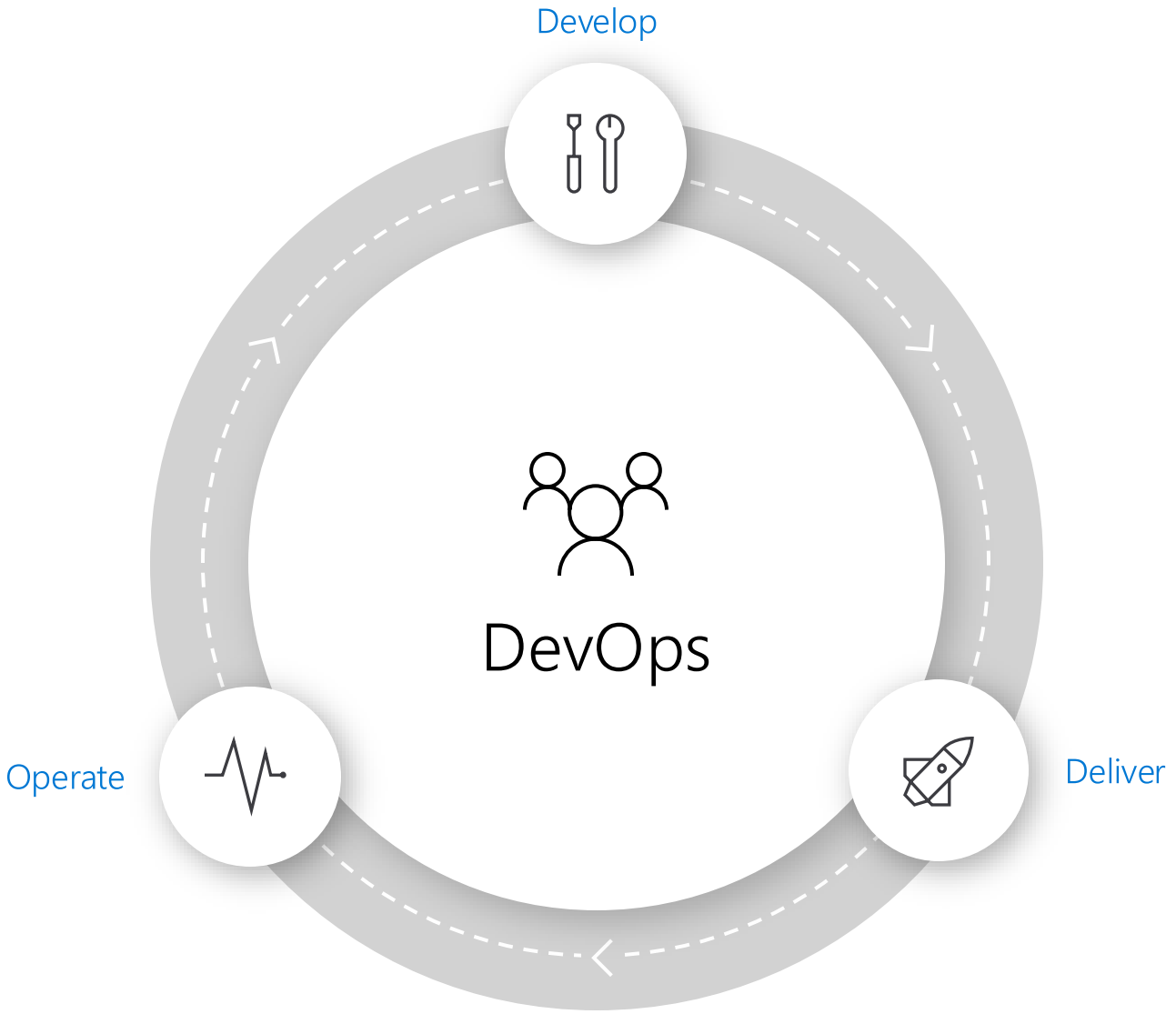
Speed vs. control impedes innovation

DEVELOPERS ARE AT THE HEART OF YOUR COMPANY'S INNOVATION



ACCELERATING DELIVERY WITH DEVOPS

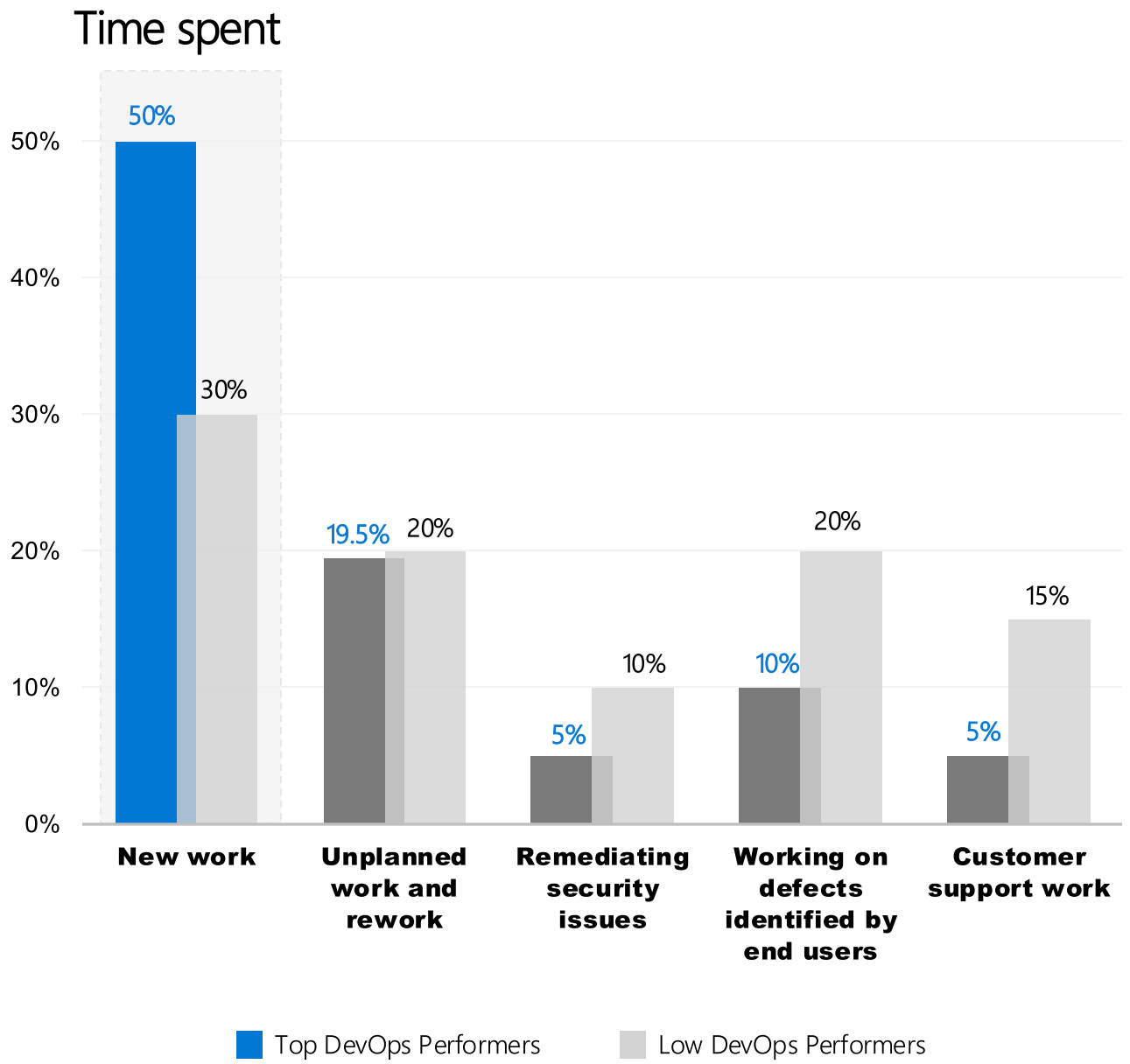
DevOps is the union of people, process, and products to enable continuous delivery of value to your end users.



INNOVATION WITH OVERSIGHT

Top performing DevOps companies spend more time innovating and less time keeping the lights on.

The result: better products, delivered faster, to happier customers by more engaged teams



DELIVER WITH AZURE DEVOPS

Azure Boards

Azure Repos

Azure Pipelines

Azure Artifacts

Azure Test Plans

Deliver

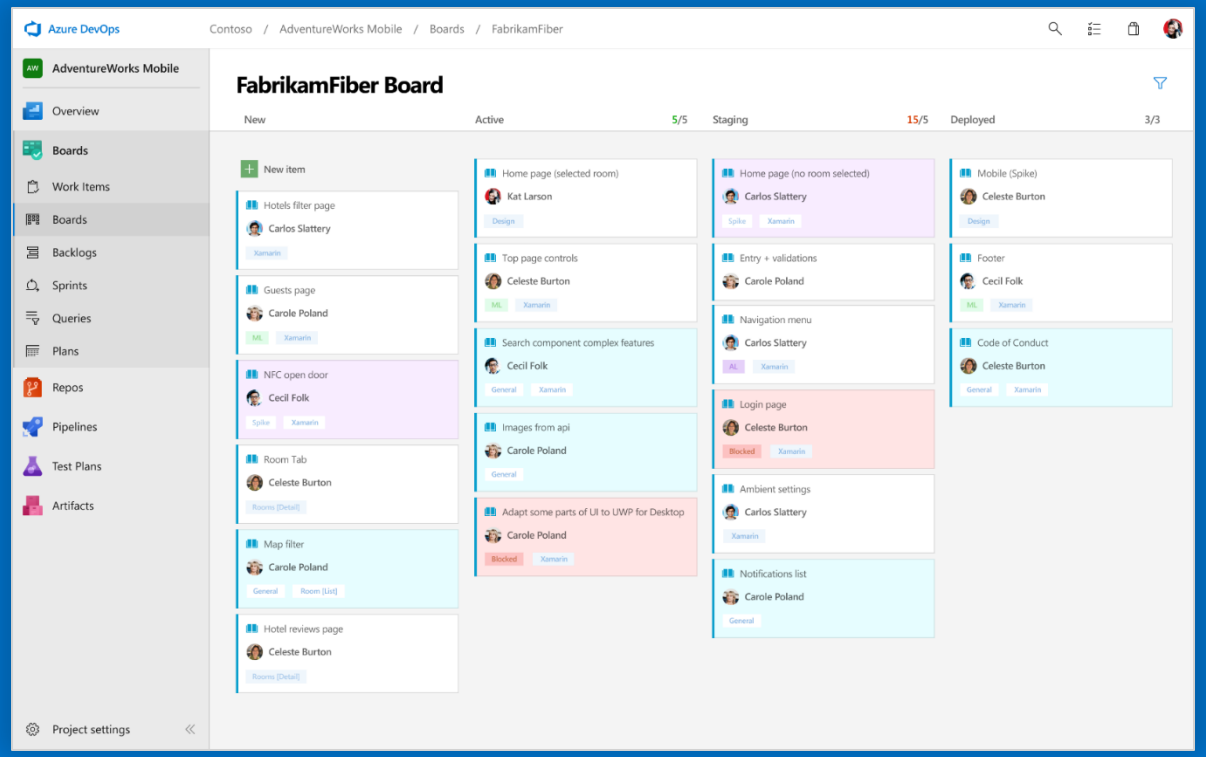
Azure Boards

Azure Repos

Azure Pipelines

Azure Artifacts

Azure Test Plans



Connecting ideas to releases

Scrum ready to help your teams run sprints, stand-ups, and plan work

Integrated with GitHub commits and pull requests

Insights into project status and health

Deliver

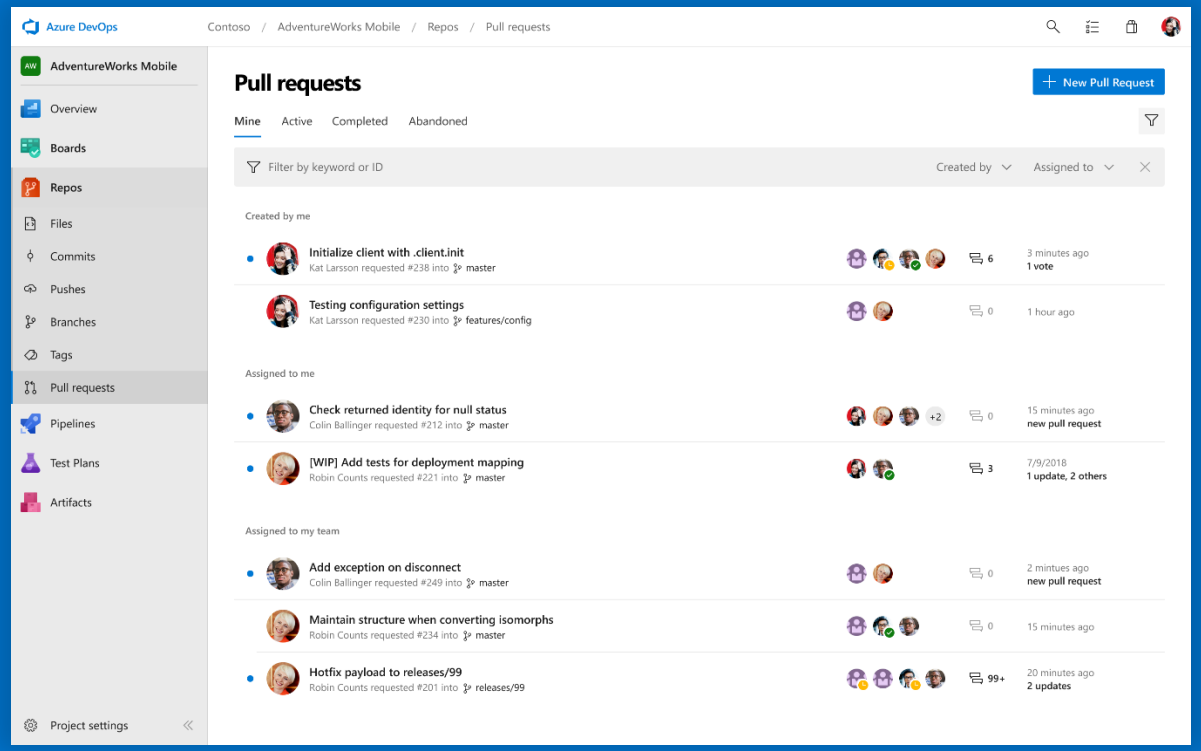
Azure Boards

Azure Repos

Azure Pipelines

Azure Artifacts

Azure Test Plans



Private Git and TFVC repos for your teams

Code review via branch pull requests

Branch policies and build validation

Easy migration path to / from GitHub

Deliver

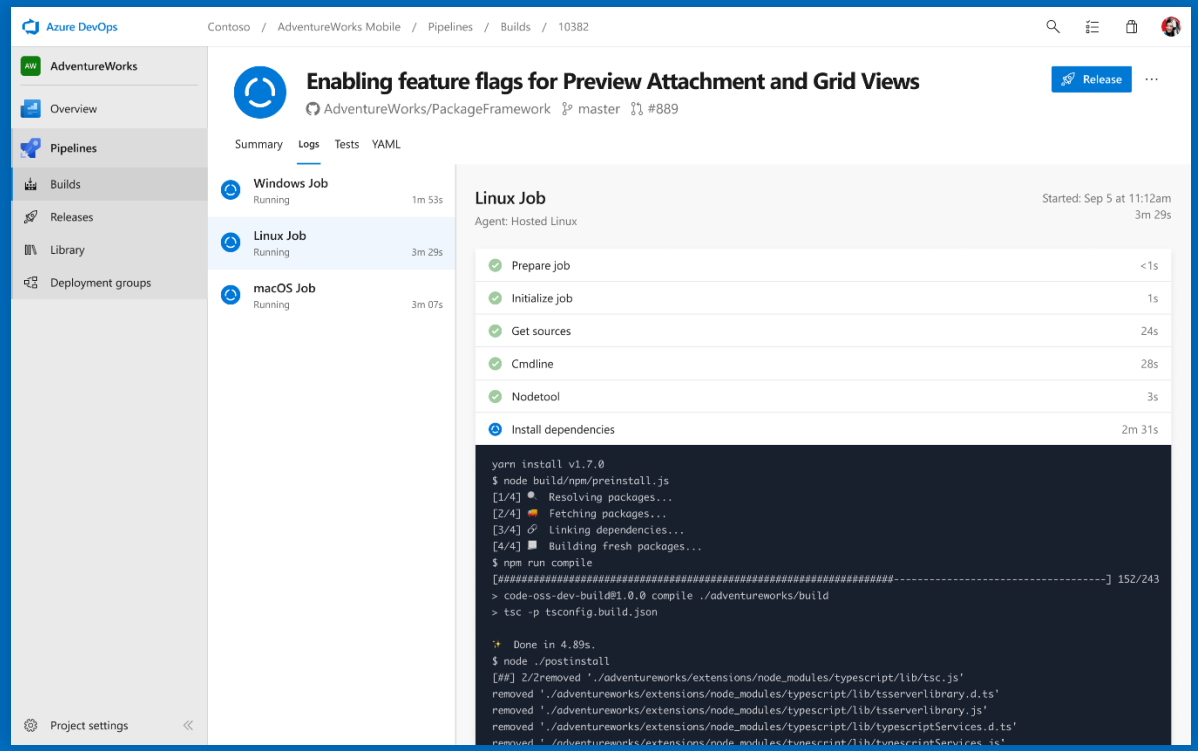
Azure Boards

Azure Repos

Azure Pipelines

Azure Artifacts

Azure Test Plans



Cloud-hosted pipelines for Linux, macOS and Windows

Any language, any platform, any cloud

Native support for containers and Kubernetes

Best-in-class for open source

Deliver

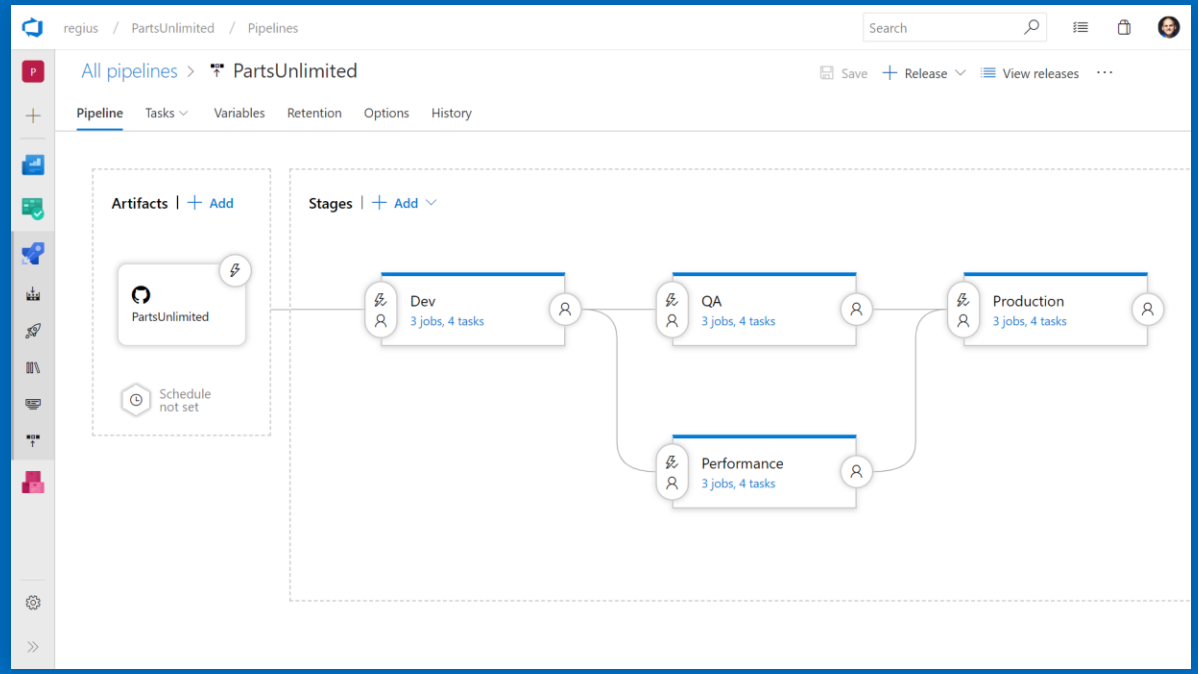
Azure Boards

Azure Repos

Azure Pipelines

Azure Artifacts

Azure Test Plans



Deploy to on-premises, ANY cloud or a hybrid of cloud and on-prem

Staged environment releases

Pre and post deployment approvals with gates to automate approval based on conditions

Deliver

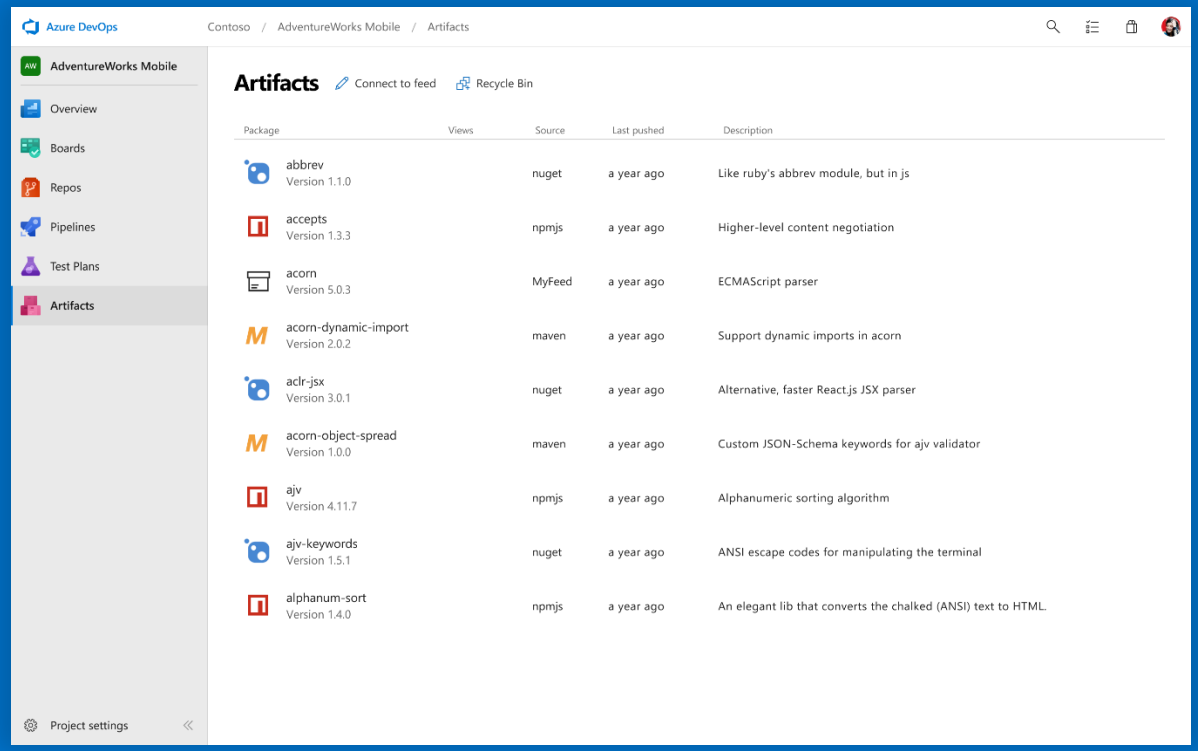
Azure Boards

Azure Repos

Azure Pipelines

Azure Artifacts

Azure Test Plans



Share code efficiently

Keep your Maven, npm, NuGet and Python packages and more in the same place

Aggregate from public registries and internal teams

Publish and track from any pipeline

Deliver

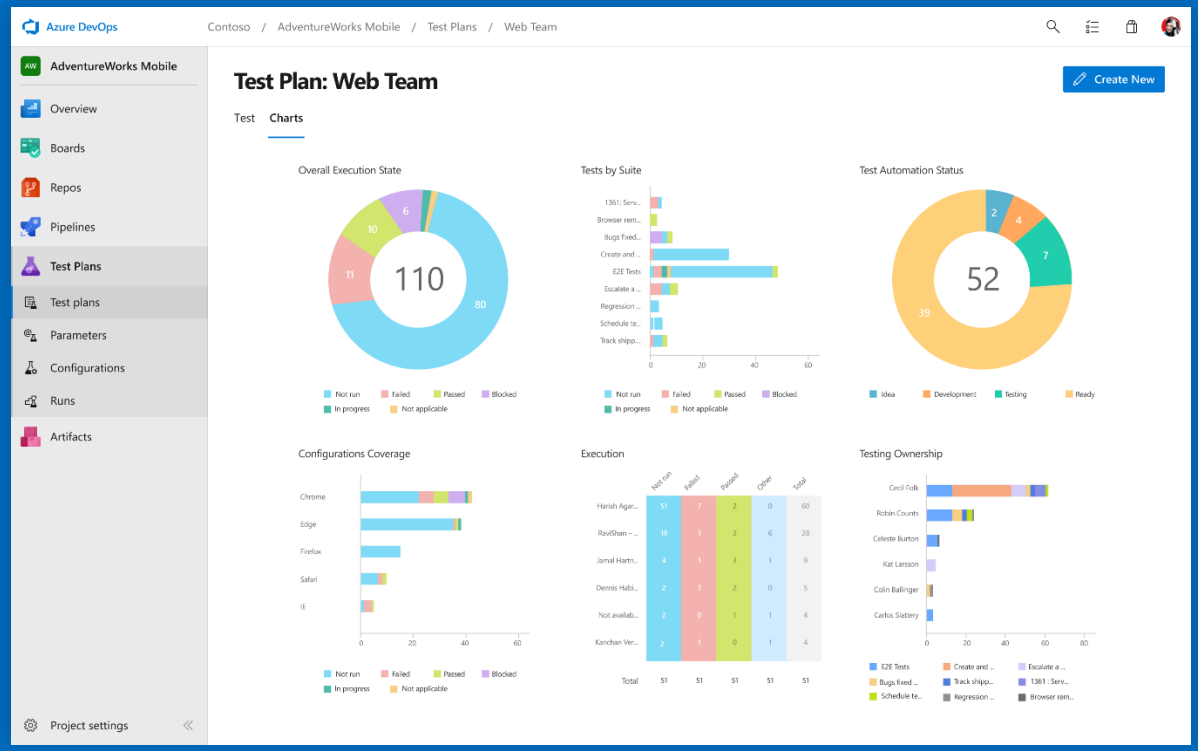
Azure Boards

Azure Repos

Azure Pipelines

Azure Artifacts

Azure Test Plans



Run tests and log defects from your browser

Track and assess quality throughout your lifecycle

Capture rich data for reproducibility

Create tests directly from exploratory sessions

Deliver with

Azure Boards

Azure Repos

Azure Pipelines

Azure Artifacts

Azure Test Plans

GitHub

GitHub brings the premier social coding workflows to your team, breaking down silos and enabling Inner Source through:

- Expertise sharing
- Cross-team collaboration
- Improved code reuse
- Increased velocity

THE #1 DEVELOPER COMMUNITY ON THE PLANET

Most contributions 1.1B in 2018

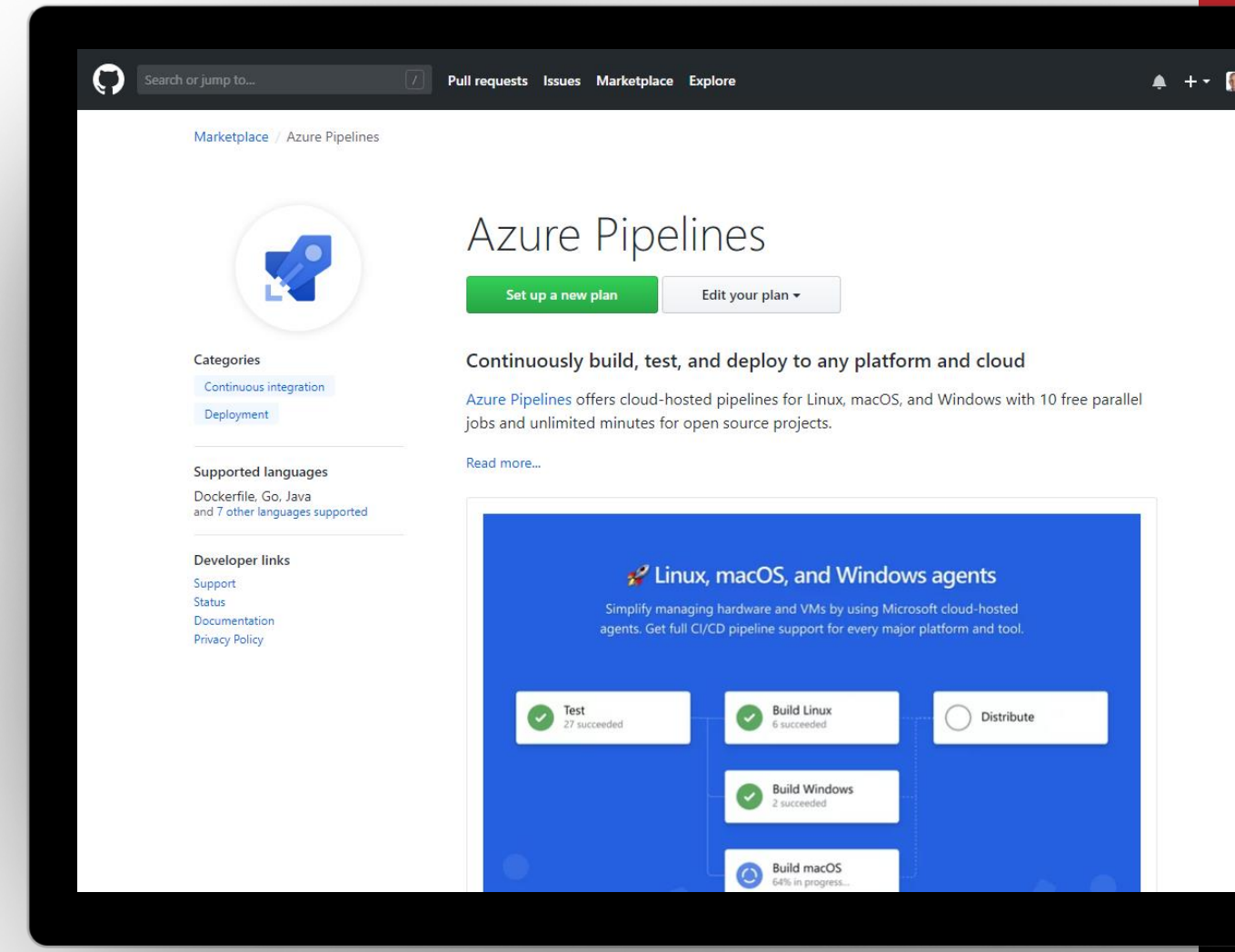
Most developers 33M

Most Repos 100M

Most secure 5M vulnerability alerts in 2018

AZURE DEVOPS INTEGRATED WITH GITHUB

Rich integration with Azure Pipelines and Azure Boards



AZURE PIPELINES

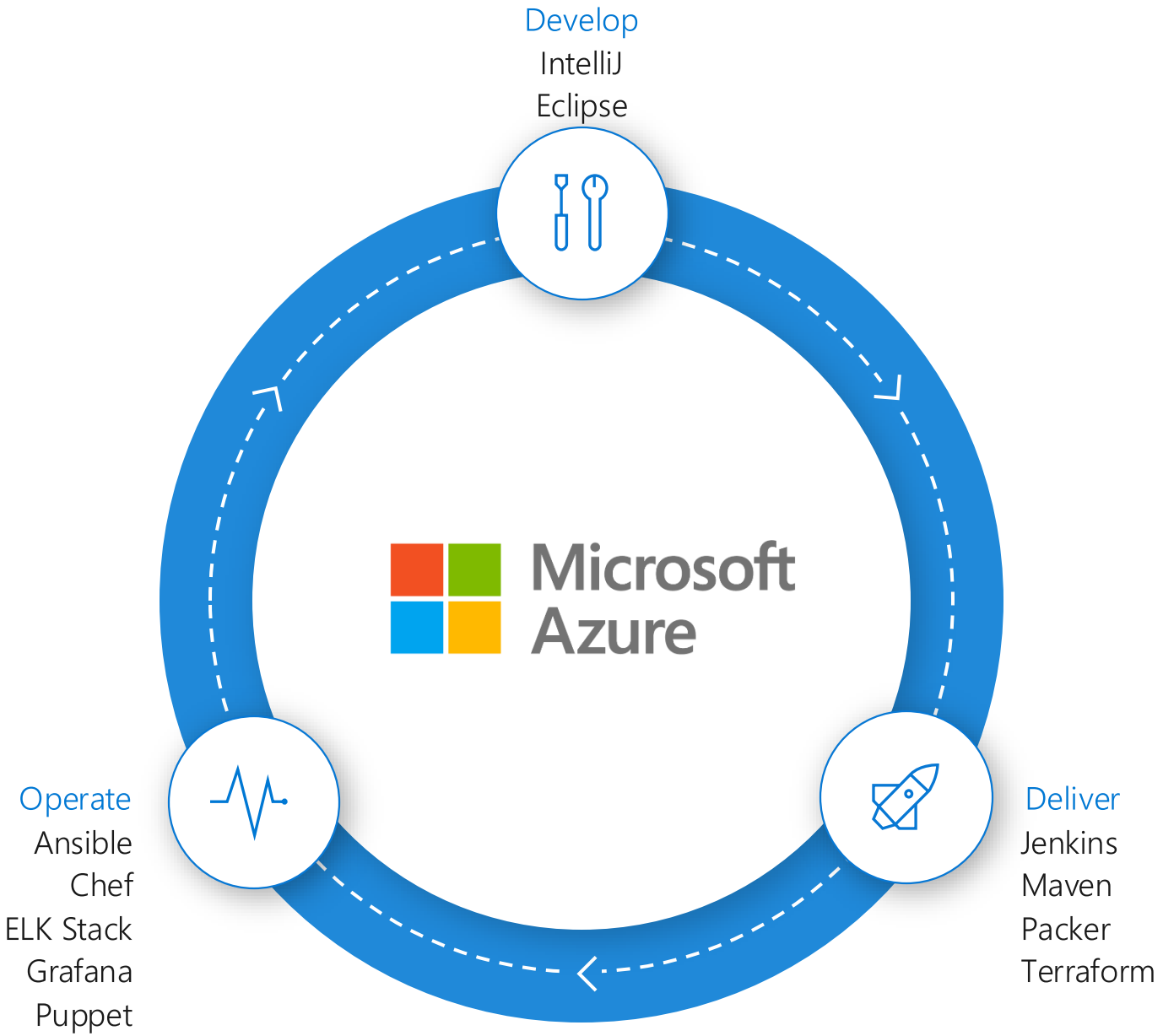
Free unlimited build minutes for public projects

Up to 10 free parallel jobs across Windows, Linux and macOS

https://azure.com/pipelines

Microsoft ♥ Open Source

Azure and GitHub are fully integrated with your existing tools and workflow



Azure DevOps Services Pricing

Open Source Projects

Free

Unlimited users and build time

- Azure Pipelines: 10 parallel jobs with unlimited minutes for CI/CD
- Azure Boards: Work item tracking and Kanban boards
- Azure Repos: Unlimited public Git repos

Small Teams

Free

Start free with up to 5 users

- Azure Pipelines: Run 1 Microsoft-hosted job for 1,800 minutes per month and 1 self-hosted job for any amount of time
- Azure Boards: Work item tracking and Kanban boards
- Azure Repos: Unlimited public Git repos
- Azure Artifacts: package management
- Unlimited stakeholders

Teams of any size

Starts at $6

per user, per month for Boards & Repos

Easy pricing that grows with your team

- Azure Pipelines: Run 1 Microsoft-hosted job for 1,800 minutes per month and 1 self-hosted job for any amount of time
- Azure Boards: Work item tracking and Kanban boards
- Azure Repos: Unlimited public Git repos
- Azure Artifacts: package management
- Unlimited stakeholders
- Boards & Repos included for Visual Studio subscribers

LAB: AKS

- Deploy Azure Kubernetes cluster
- Configure Kubectl
- Deploy multi-tier application
- Scale frontend of application
- Perform rolling update

INDIVIDUAL KEY TAKEAWAYS

Write down three key insights from today's session.

Highlight how these take aways influence your work.

Q&A AND OPEN DISCUSSION


THANK
YOU