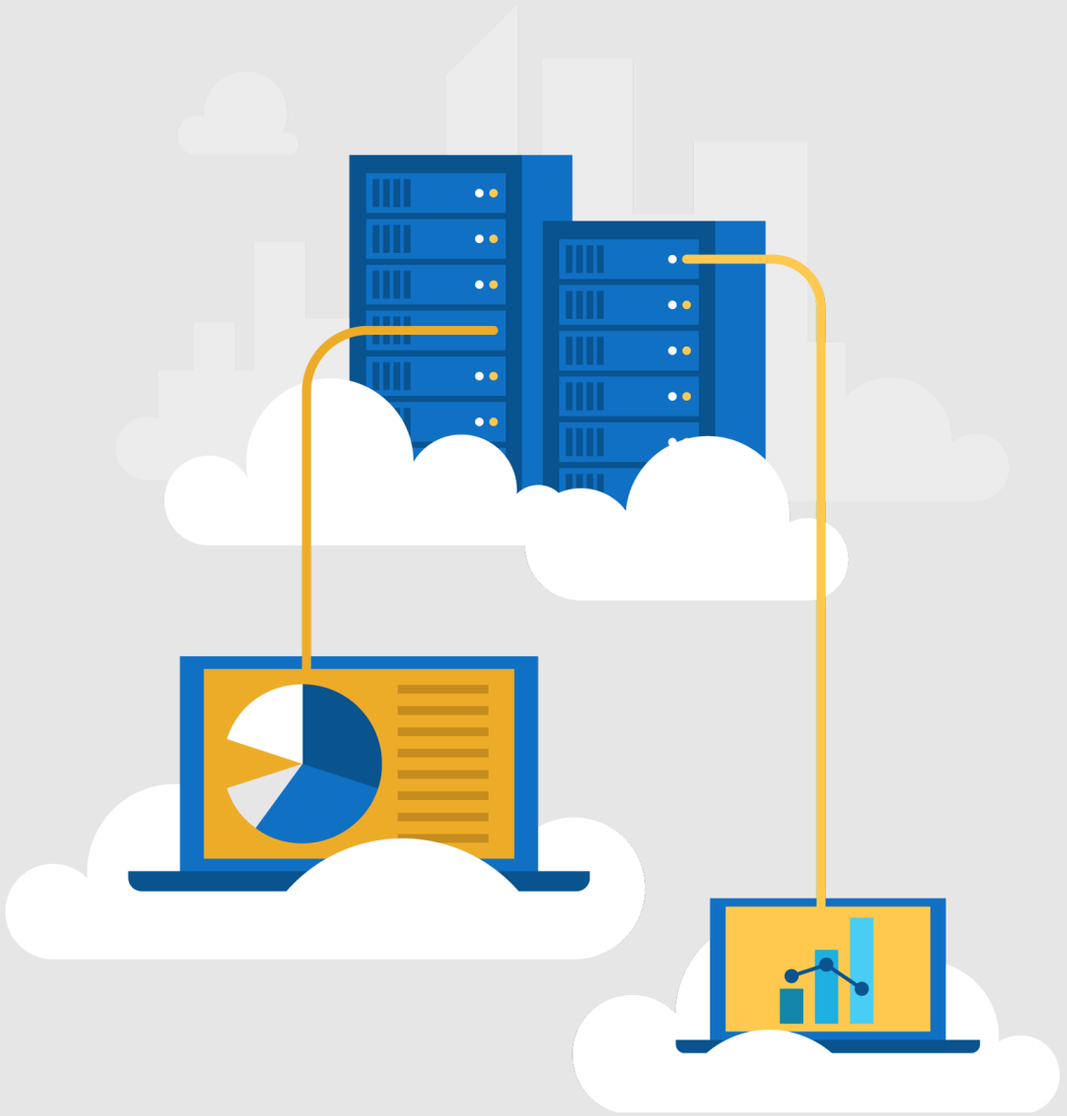
Microsoft

# Azure Study Group

## AZ-301 - Microsoft Azure Architect Design

Jeff Wagner
Partner Technology Strategist

# Agenda

1. Agenda
2. Speaker Introduction
3. Feedback Loop
4. Objective Review
5. Open Mic

# Series Agenda

| | |
|---|---|
| 1 | Determine Workload Requirements (10-15%) |
| 2 | Design for Identity and Security (20-25%) |
| 3 | Design a Data Platform Solution (15-20%) |
| 4 | Design a Business Continuity Strategy (15-20%) |
| 5 | Design for Deployment, Migration, and Integration (10-15%) |
| 6 | Design an Infrastructure Strategy (15-20%) |

https://aka.ms/azurecsg

# Series Agenda

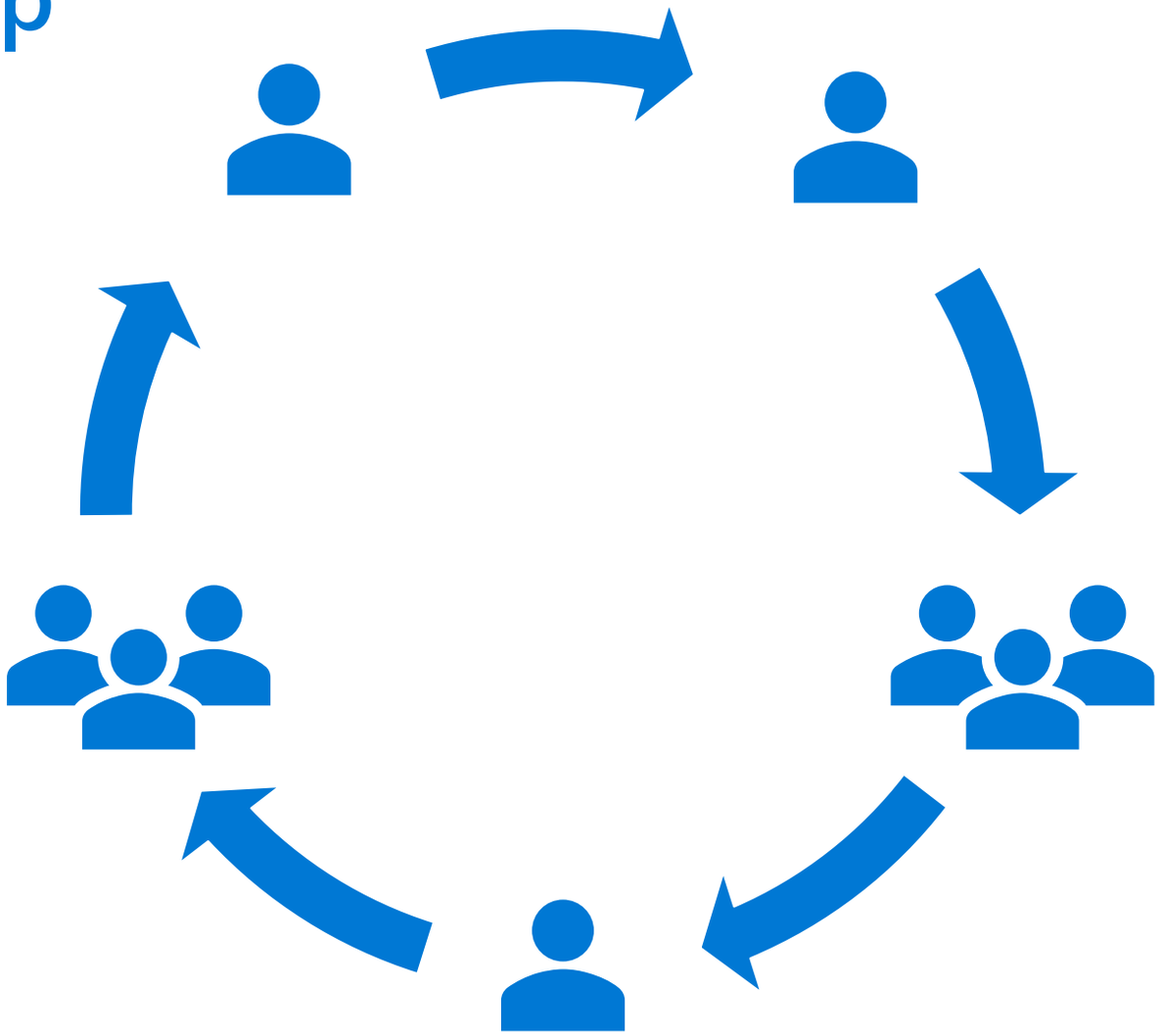
| | |
|---|---|
| 1 | Determine Workload Requirements (10-15%) |
| 2 | Design for Identity and Security (20-25%) |
| 3 | Design a Data Platform Solution (15-20%) |
| 4 | Design a Business Continuity Strategy (15-20%) |
| 5 | Design for Deployment, Migration, and Integration (10-15%) |
| 6 | Design an Infrastructure Strategy (15-20%) |

https://aka.ms/azurecsg

# Speaker Introduction - Jeff Wagner

- Partner Technology Strategist based in Atlanta
- 21+ years with Microsoft, more in the industry
- Constant learner - *Ancora Imparo*
- Working on the same certifications that you are

# Feedback Loop



# Objectives

**Design a Storage Strategy**
design a storage provisioning strategy
design storage access strategy
identify storage requirements
recommend a storage solution and storage management tools

**Design a Compute Strategy**
design compute provisioning and secure compute strategies
determine appropriate compute technologies (e.g., virtual machines, functions, service fabric, container instances, etc.)
design an Azure HPC environment
identify compute requirements
recommend management tools for compute

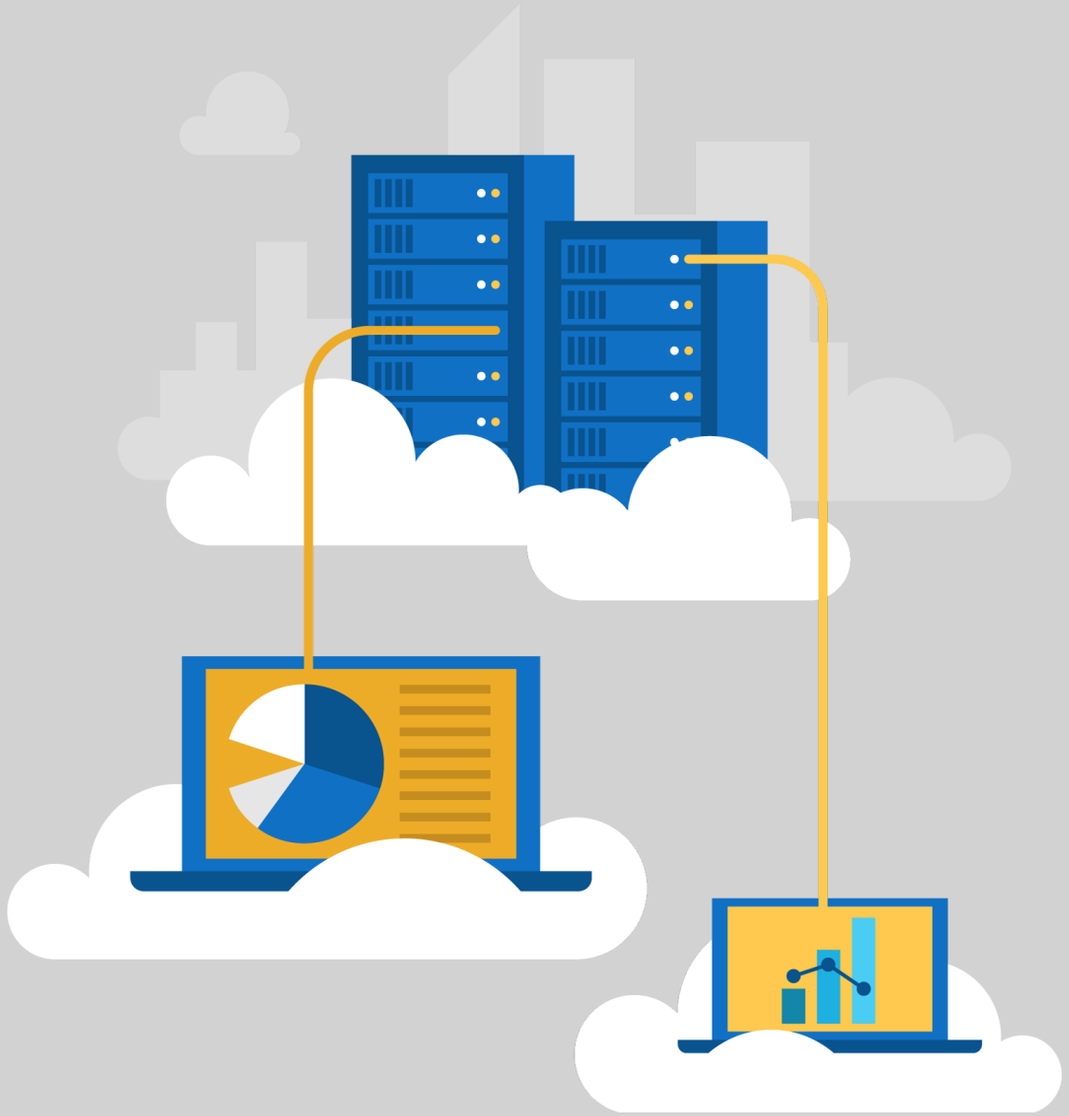
# Objectives (Cont.)

**Design a Networking Strategy**
design network provisioning and network security strategies
determine appropriate network connectivity technologies
identify networking requirements
recommend network management tools

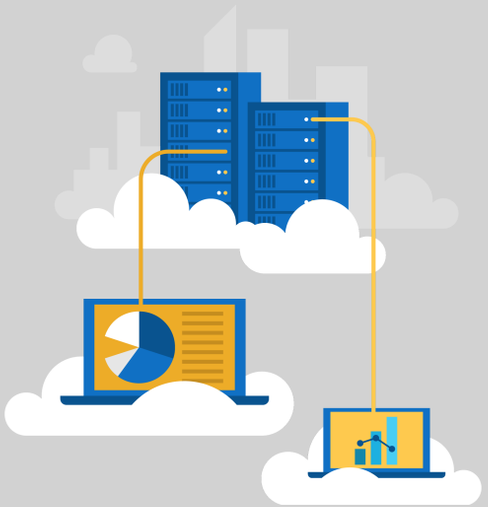
**Design a Monitoring Strategy for Infrastructure**
design for alert notifications
design an alert and metrics strategy

Microsoft

# Design a Storage Strategy

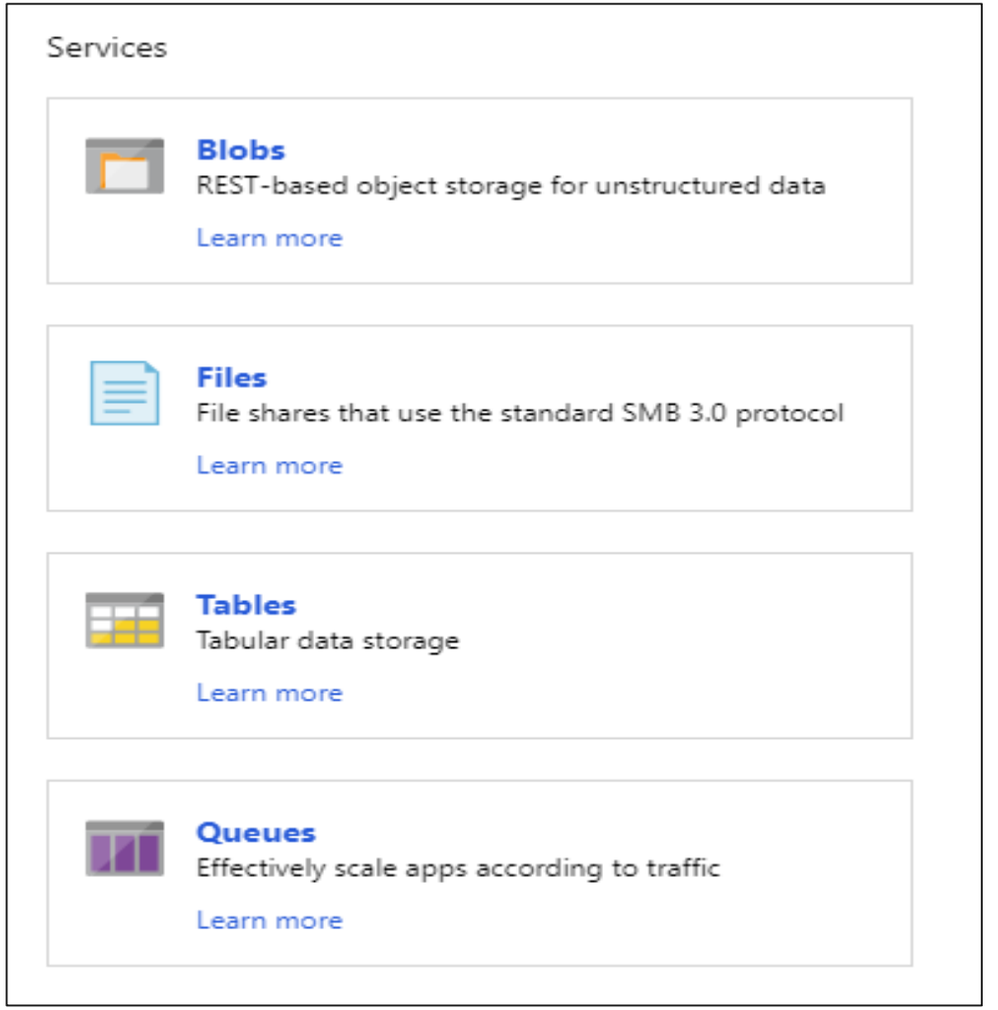


# Azure Storage Accounts

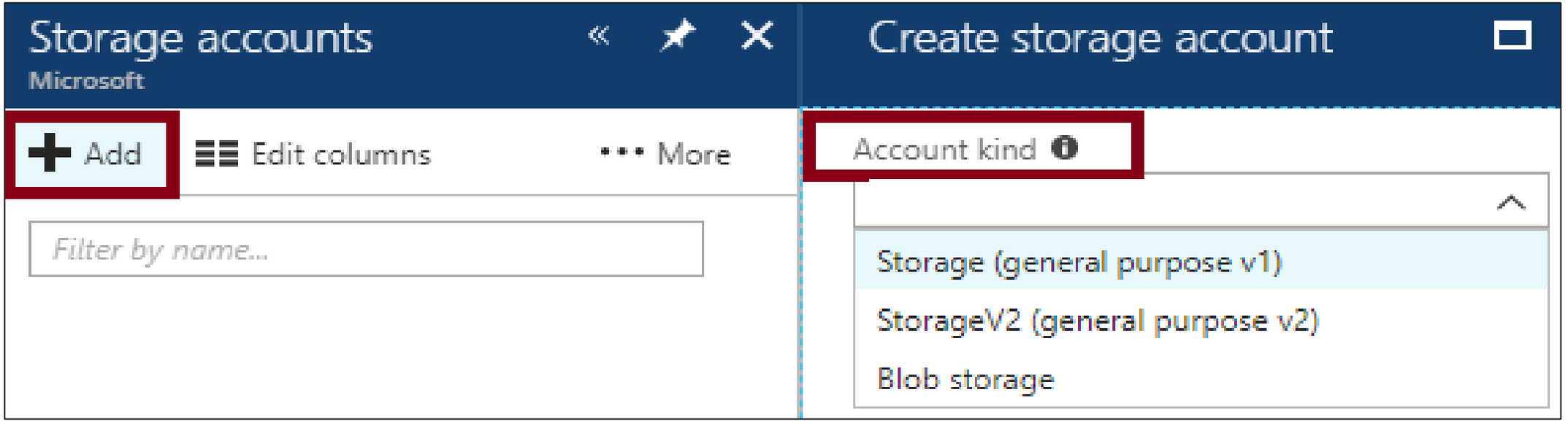


# Azure Storage

- A service that you can use to store files, messages, tables, and other types of information
- Three categories of Azure storage:
  - Storage for virtual machines – Disks and File Shares
  - Unstructured data – Blobs and Data Lake Store
  - Structured data - Tables, Cosmos DB, and Azure SQL DB

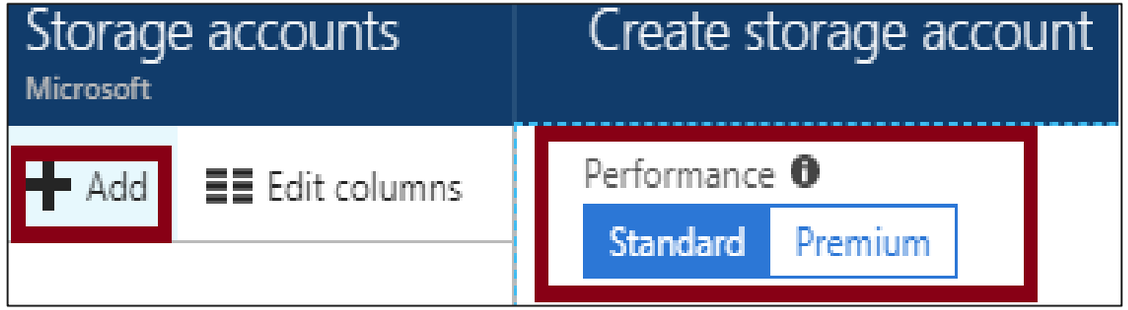


# Azure Storage Accounts




- Two types of Storage: General purpose and Blob storage
- General purpose storage has two tiers: Standard and Premium.
- Blob storage has three tiers: Hot, Cool, and Archive

# Standard and Premium Storage Accounts




- Standard:
  - Backed by magnetic drives (HDD)
  - Lowest cost per GB
- Premium:
  - Backed by solid state drives (SSD)
  - Can only be used with Azure VM disks
  - 99.99% SLA

# Storage Account Endpoints

- Every object has a unique URL address
- The storage account name forms the subdomain of that address
- The subdomain and domain name forms an *endpoint*
  - **Blob service**: http://*mystorageaccount*.blob.core.windows.net
  - **Table service**: http://*mystorageaccount*.table.core.windows.net
  - **Queue service**: http://*mystorageaccount*.queue.core.windows.net
  - **File service**: http://*mystorageaccount*.file.core.windows.net

# Configuring Custom Domain Names

- Direct CNAME mapping

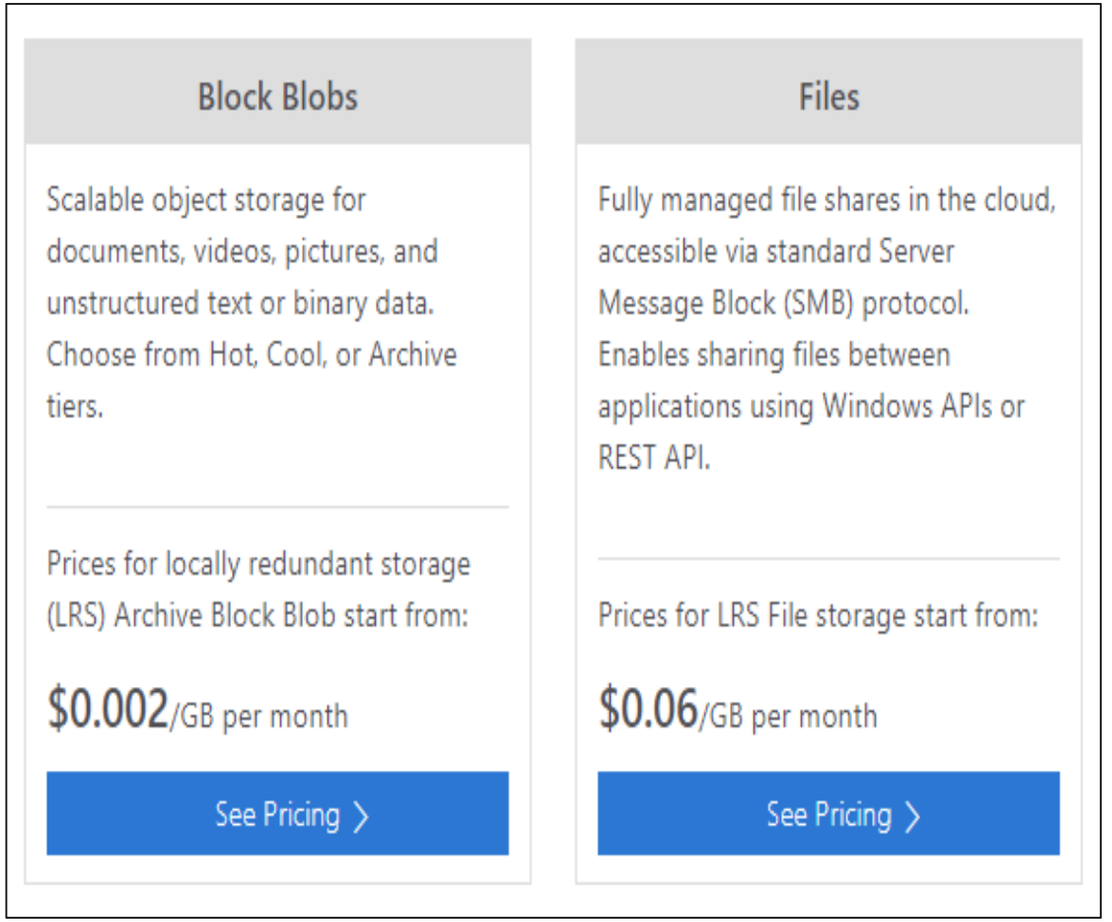
| CNAME record | Target |
| --- | --- |
| blobs.contoso.com | contosoblobs.blob.core.windows.net |

- Intermediary mapping with *asverify*

| CNAME record | Target |
| --- | --- |
| asverify.blobs.contoso.com | asverify.contosoblobs.blob.core.windows.net |
| blobs.contoso.com | contosoblobs.blob.core.windows.net |

# Storage Pricing and Billing

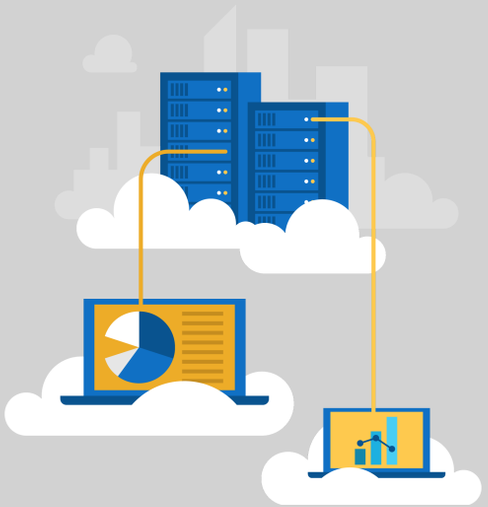
- Storage costs
- Blob storage
- Data access costs
- Transaction costs
- Geo-Replication data transfer costs
- Outbound data transfer costs
- Changing the storage tier



# Practice: Storage Account Management

- How to create a GPv2 storage account
- How to convert a GPv1 or Blob storage account to a GPv2 storage account
- How to set the account in a GPv2 storage account
- How to set a blob tier in a Blob storage or GPv2 storage account

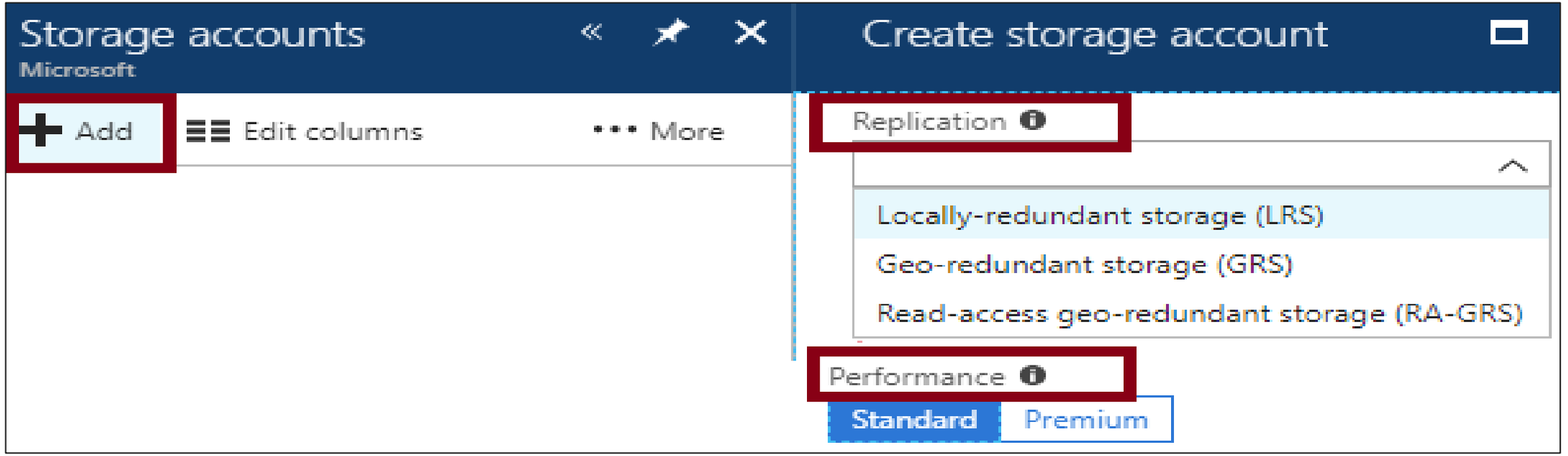
# Data Replication



# Planning Storage

- Types of storage services
- Storage design
- Storage billing
- Storage account management
- Storage account access/tools

# Replication Options




- Replication ensures durability and high availability
- Replicate your data within the same data center, across zonal data centers within the same region, and even across regions

# Locally Redundant Storage

| Replication | Copies | Strategy |
| --- | --- | --- |
| Locally redundant storage (LRS) | Maintains three copies of your data. | Data is replicated three time within a single facility in a single region. |

- Maintains three copies of your data at a single facility
- All copies of the data exist within the same region
- Use if data can be easily reconstructed
- Use if there are regional governance requirements
- Low-cost option

# Geo-redundant Storage

| Replication | Copies | Strategy |
| --- | --- | --- |
| Geo-redundant storage (GRS) | Maintains six copies of your data. | Data is replicated three times within the primary region and is also replicated three times in a secondary region hundreds of miles away from the primary region. |
| Read access geo-redundant storage (RA-GRS) | Maintains six copies of your data. | Data is replicated to a secondary geographic location and provides read access to your data in the secondary location. |

- GRS replicates your data to another data center in a secondary region, but that data is available to be read only during a failure.
- RA-GRS is based on GRS and replicates data to another data center in another region. Provides read access from the secondary region, even without a failure.

# Zone Redundant Storage

| Replication | Copies | Strategy |
| --- | --- | --- |
| Zone-redundant storage (ZRS) | Maintains three copies of your data. | Data is replicated three times across two to three facilities, either within a single region or across two regions. |

- Replicates your data across three storage clusters in a single region
- Each storage cluster is physically separated from the others and resides in its own availability zone
- Each availability zone, and the ZRS cluster within it, is autonomous, with separate utilities and networking capabilities
- Not available in all regions

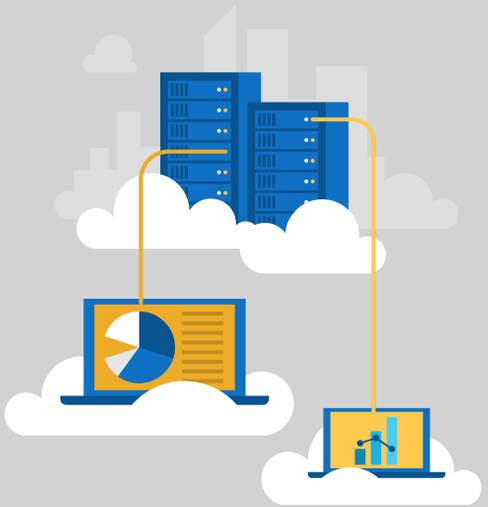
# Replication Option Comparison

| Replication Option | LRS | ZRS | GRS | RA-GRS |
|---|---|---|---|---|
| Node unavailability within a data center | Yes | Yes | Yes | Yes |
| An entire data center (zonal or non-zonal) becomes unavailable | No | Yes | Yes | Yes |
| A region-wide outage | No | No | Yes | Yes |
| Read access to your data (in a remote, geo-replicated region) for region-wide unavailability | No | No | No | Yes |
| Available in storage account types | GPv1, GPv2, Blob | Standard, GPv2 | GPv1, GPv2, Blob | GPv1, GPv2, Blob |

# Storage Accounts PowerShell Tasks

| Task | Example |
| --- | --- |
| Check to see if a storage account name is available. | **Get-AzureRmStorageAccountNameAvailability** -Name 'mystorageaccount' |
| Create a storage account. | **New-AzureRmStorageAccount** -ResourceGroupName MyResourceGroup -AccountName mystorageaccount -Location westus -SkuName Standard_GRS |
| Retrieve a specific storage account or all the storage accounts in a resource group or subscription. | **Get-AzureRmStorageAccount** -ResourceGroupName "RG01" -AccountName "mystorageaccount" |
| Modify storage account properties, such as type. | **Set-AzureRmStorageAccount** -ResourceGroupName "MyResourceGroup" -AccountName "mystorageaccount" -Type "Standard_RAGRS" |

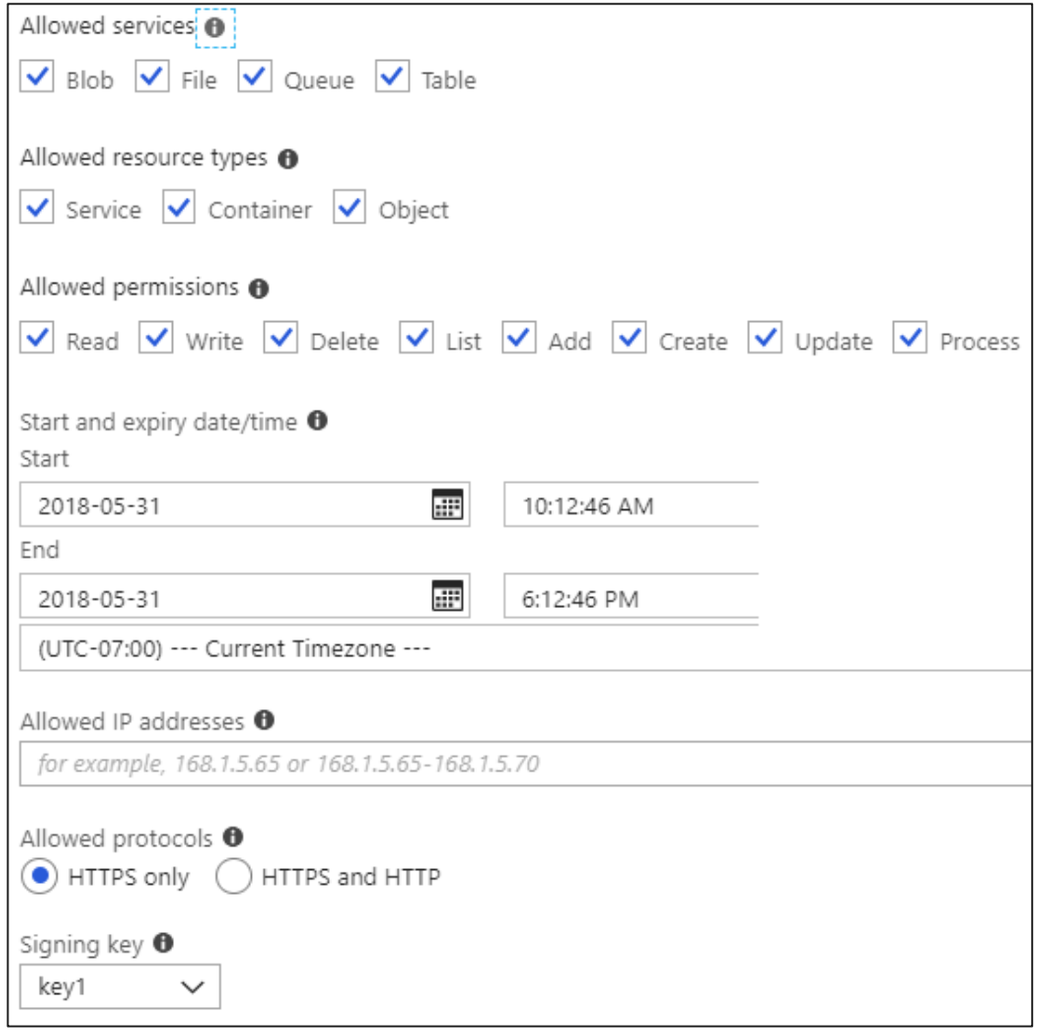
# Shared Access Keys



# Shared Access Signature (SAS)


- Provides delegated access to resources
- Grants access to clients without sharing your storage account keys
- The account SAS delegates access to resources in one or more of the storage services: Blob, Queue, Table, or File service
- The service SAS delegates access to a resource in just one of the storage services

# Configuring SAS Parameters




Account level SAS, full permissions

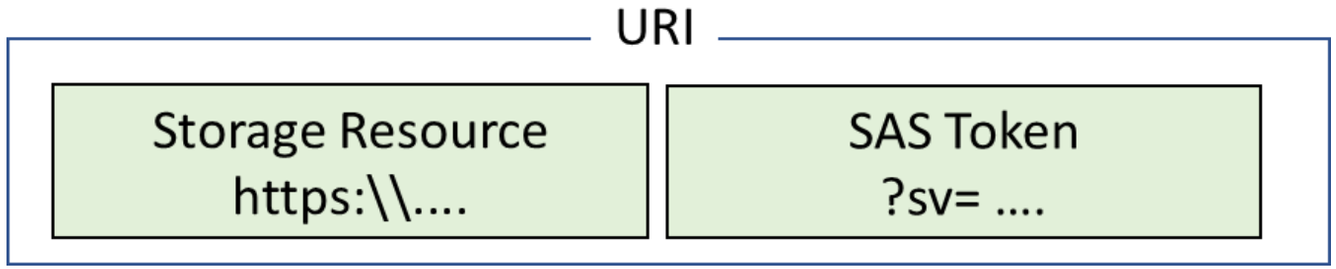
```
New-AzureStorageAccountSASToken
  -Service  Blob,File,Table,Queue
  -ResourceType Service,Container,Object
  -Permission "racwdlup"
```

Blob level SAS, full permissions

```
New-AzureStorageBlobSASToken
   -Container "ContainerName"
   -Blob "BlobName"
   -Permission rwd
```

# URI and SAS Parameters

- A SAS is a signed URI that points to one or more storage resources
- Consists of a storage resource URI and the SAS token

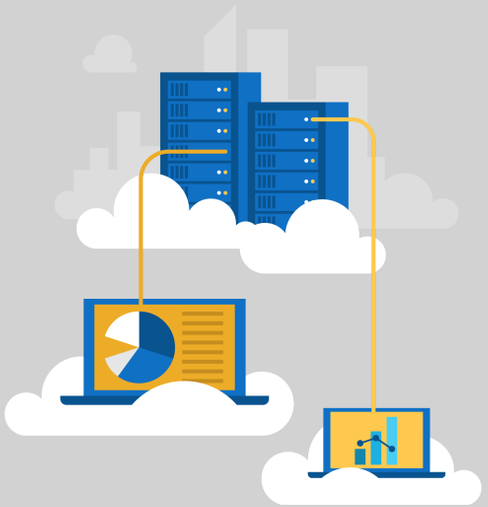



- Includes parameters for resource URI, storage services version, services, resource types, start time, expiry time, resource, permissions, IP range, protocol, signature

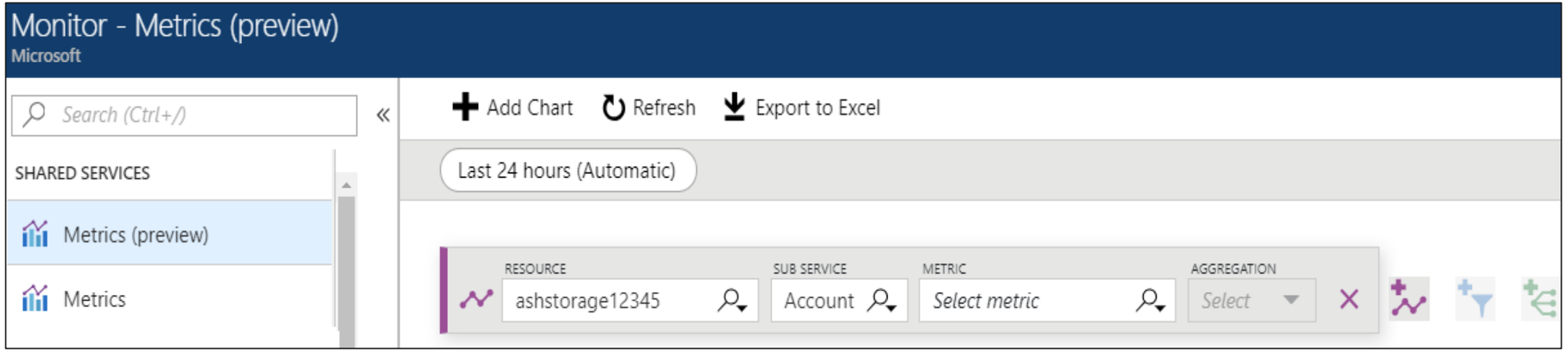
# Best Practices

- Always use HTTPS to create or distribute an SAS
- Reference stored access policies where possible
- Be careful with SAS start time
- Be specific with the resource to be accessed

# Metrics and Alerts



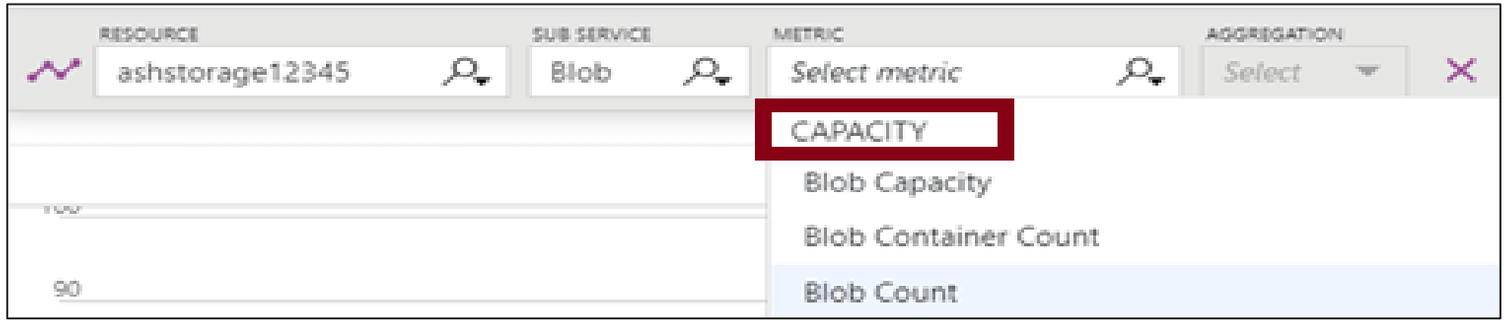
# Monitor Metrics



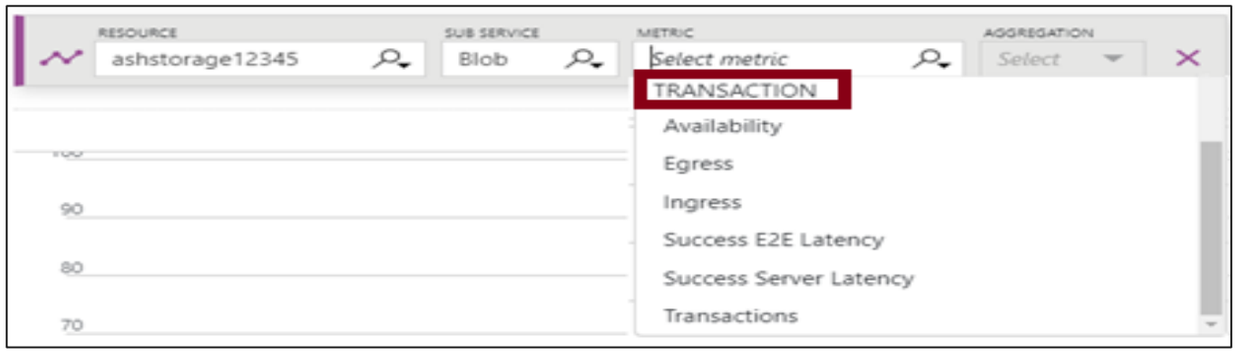

- Azure Monitor provides unified user interfaces for monitoring across different Azure services
- Azure Storage integrates Azure Monitor by sending metric data to the Azure Monitor platform
- Access metrics with: portal, Monitor APIs, OMS, and Event Hubs

# Capacity and Transaction Metrics

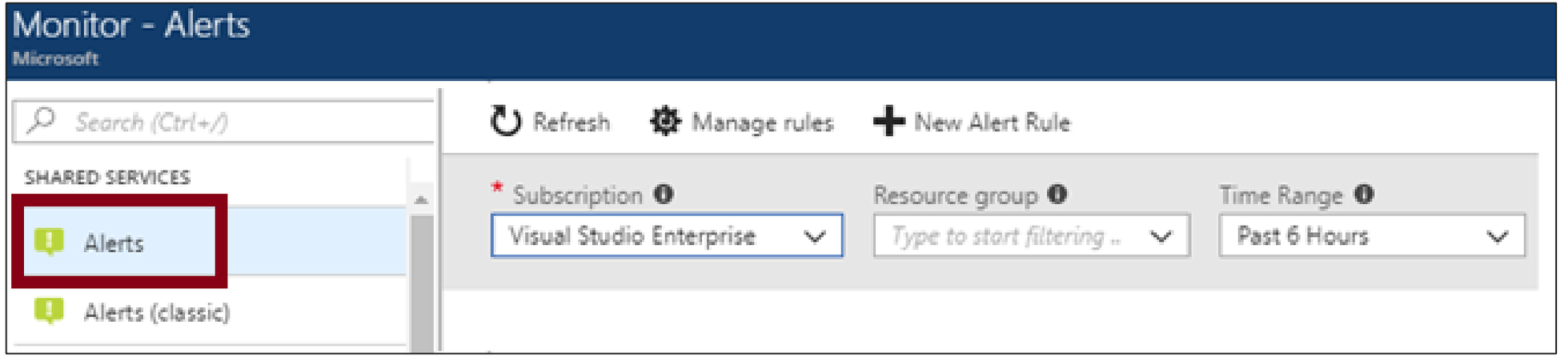
- Capacity metrics values are sent every hour and refreshed daily




- Transaction metrics are sent every minute for both account and service level



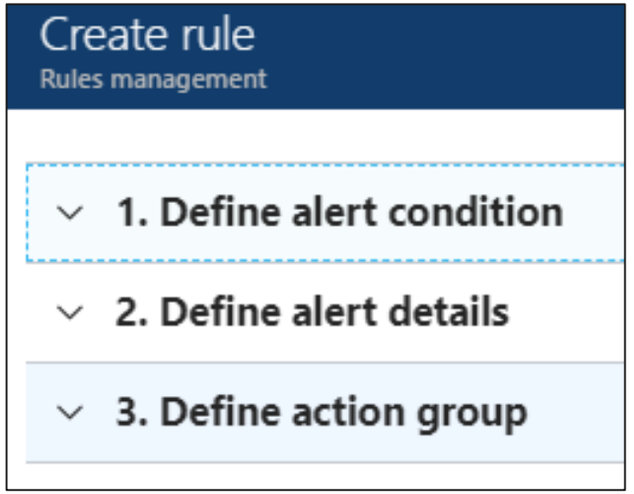
# Azure Monitor Alerts




- Better notification system
- Unified authoring experience
- Log Analytics alerts display in Azure portal
- Separation of Fired Alerts and Alert Rules
- Improved workflow

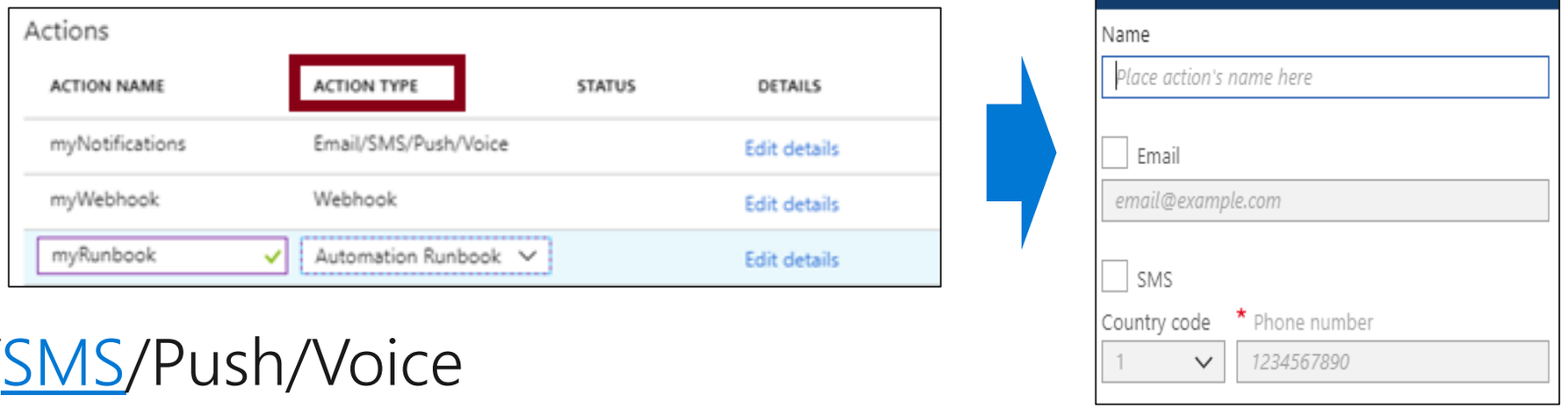
# Alert Rules

1. **Define alert conditions**: Target selection, Alert criteria, and Alert logic
2. **Define alert details**: Alert rule name, description, and severity (0 to 4)
3. **Define action group**: notify your team via email and text messages or automate actions using webhooks and runbooks.



# Action Groups

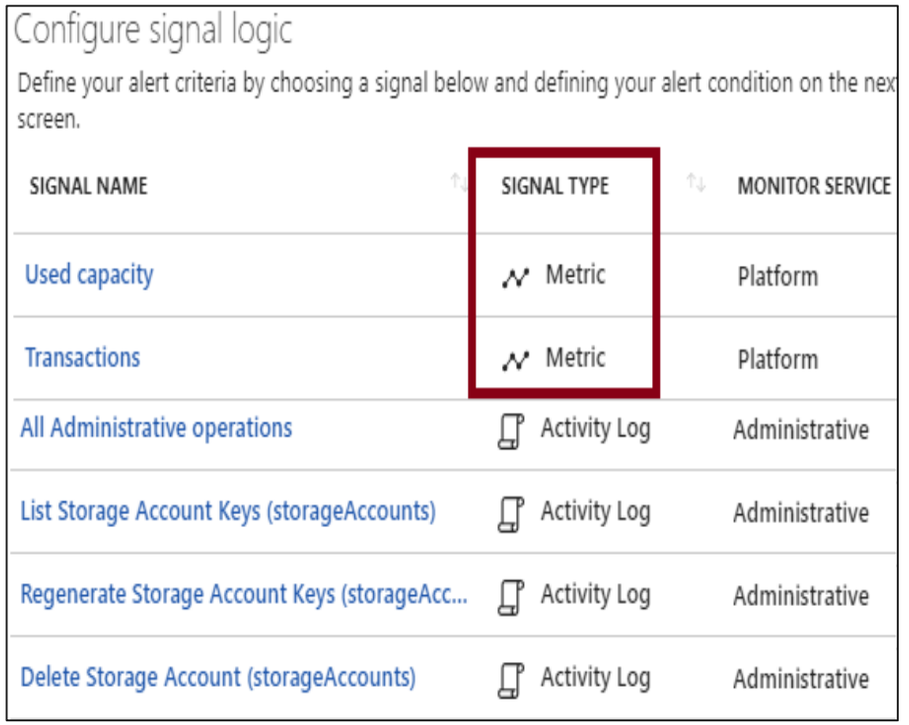
- Configure a list of actions to take when an alert is triggered




- Email/SMS/Push/Voice
- Logic App
- Webhook
- IT Service Management
- Automation Runbook

# Signal Types and Metrics

- Signal types:
  - Metric
  - Activity log
  - Application Insights
  - Logs
- Enable newer metric alerts:
  - Improved latency
  - Multi-dimensional metrics support
  - Combined monitoring for multiple metrics
  - Metrics from Logs



# Monitoring Storage

- Enable monitoring for a new or existing storage account
  - Aggregate metrics
  - Per-API metrics
  - Logs
- Not supported for Premium storage accounts
- Metrics and logs are stored in the same storage account
- Metrics can be displayed in the Monitoring lens
- Metric-based alerts
  - Delivered through email
  - Routed to a Webhook

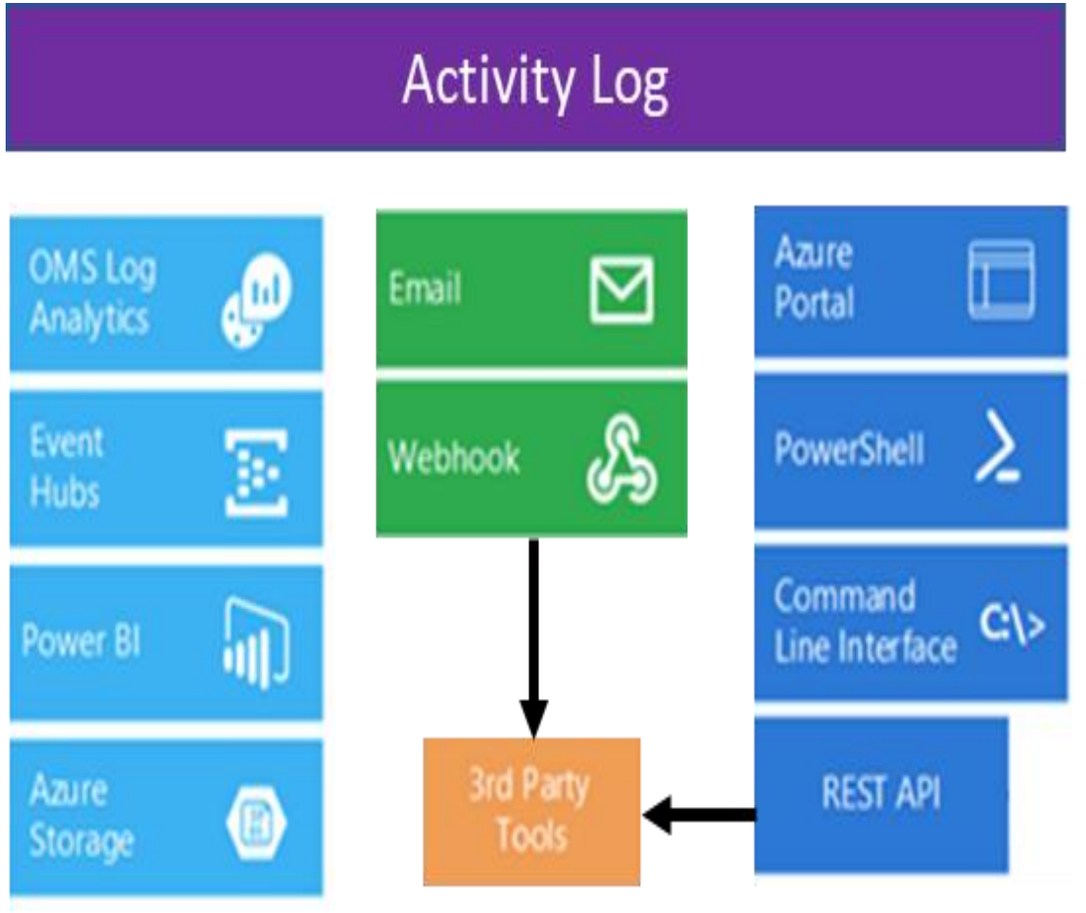
# Practice: Audit and receive notifications about important actions in your Azure subscription

- Create a network security group
- Browse the Activity Log in the portal
- Browse an event in the Activity log
- Create an Activity log alert
- Test the activity log alert

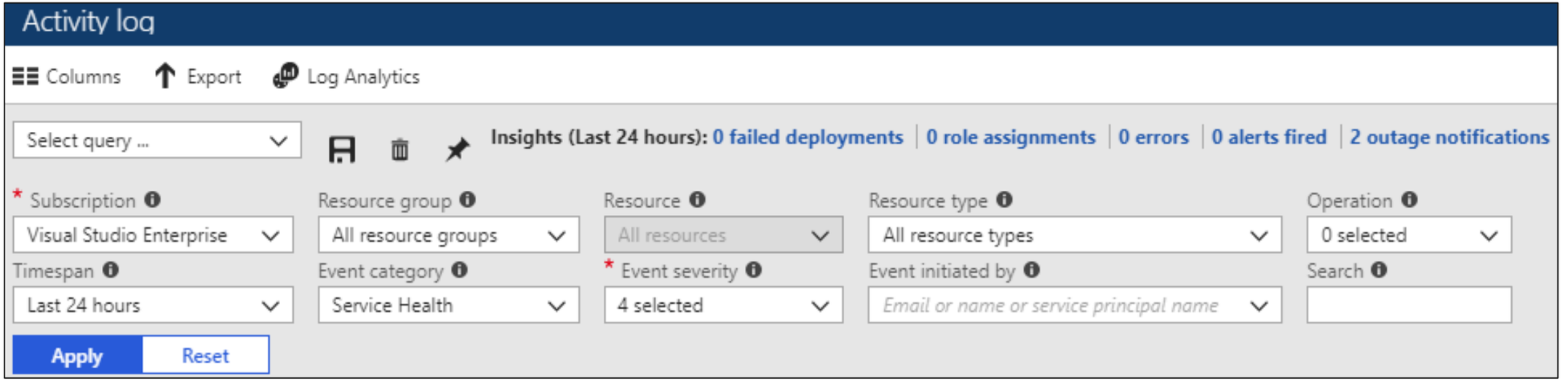
# Activity Log

# Activity Log

- Log with insight into subscription-level events
- Different from Diagnostic Logs
- Send data to Log Analytics
- Query or manage events
- Stream information to Event Hub
- Archive data to a storage account
- Analyze data with Power BI



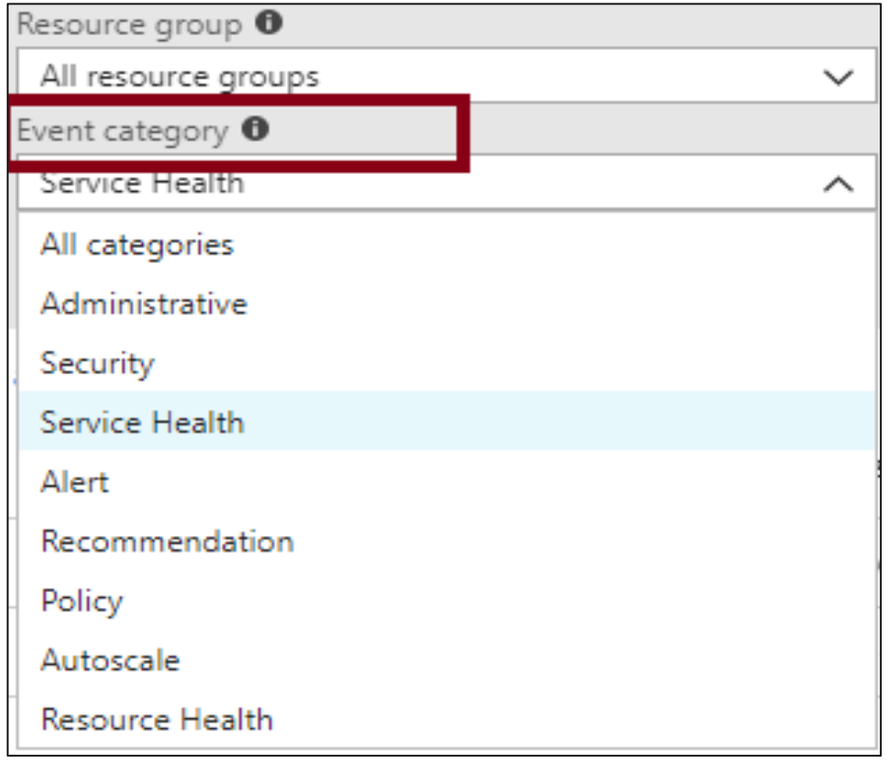
# Query the Activity Log




- Define a set of filters
- Save it as a query that is persisted across sessions
- Pin to your Azure dashboard

# Event Categories

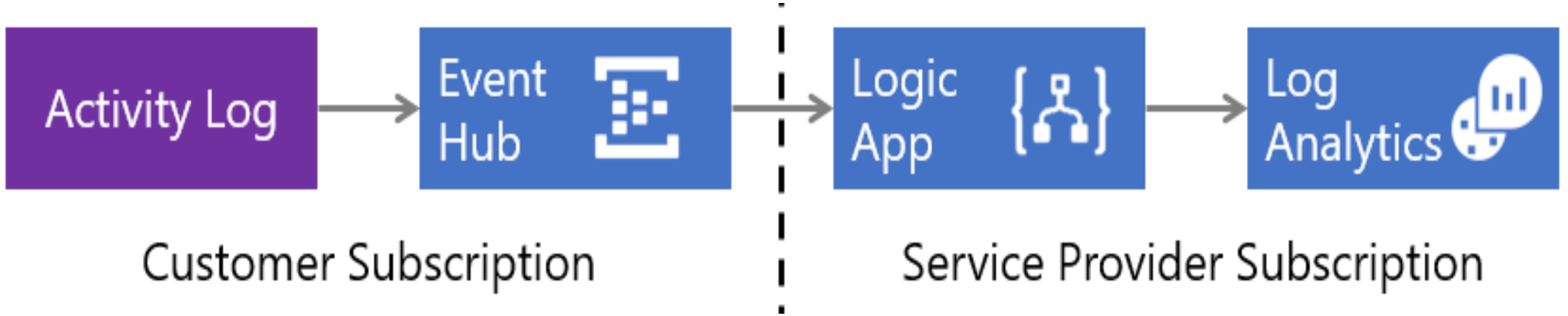
- Administrative - create, update, delete, and action operations
- Service Health – five varieties
  - Alert activations
  - Autoscale events
  - Recommendation on resource usage
  - Azure Security Center events
  - Policy and Resource Health (reserved)



# Activity Log and Log Analytics

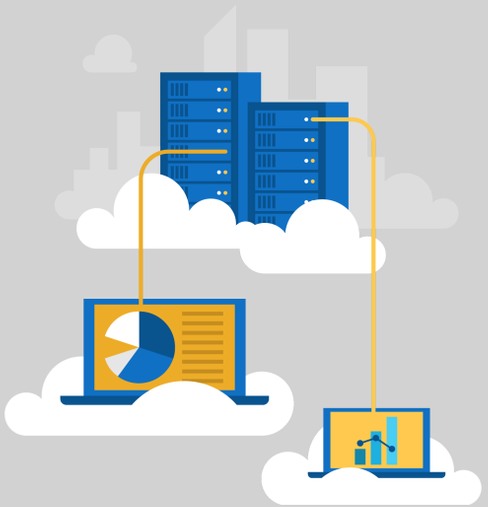
- Analyze activity logs using pre-defined views
- Analyze and search activity logs
- Keep activity logs for more than 90 days
- Correlate activity logs with other Azure platform and application data
- View operational activities by status
- View trends of activities
- Report authorization changes
- Identify outage or service health issues
- Use Log Search to correlate information to other logs or metrics

# Collect Across Subscriptions




- Use the Log Analytics Data Collector connector for Logic Apps
- Collect Azure Activity Logs into a Log Analytics workspace
- Send events to an Event Hub where a Logic App sends them to your Log Analytics workspace
- Low latency solution; minimal code required

# Azure Storage Explorer

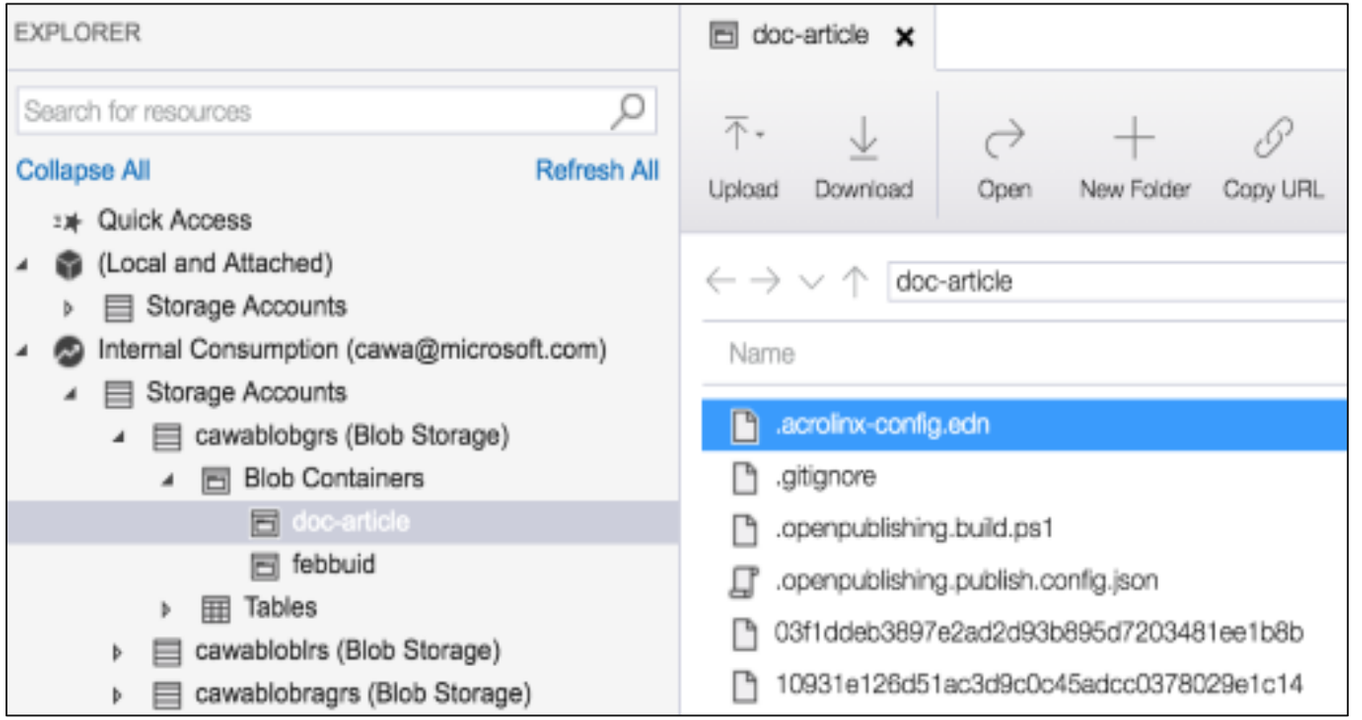


# Overview of Azure Storage Explorer

- Easily manage the contents of your storage account with Azure Storage Explorer
- Upload, download, and manage blobs, files, queues, tables, and Cosmos DB entities
- Gain easy access to manager your virtual machine disks

# Azure Storage Explorer

- Access multiple accounts and subscriptions
- Create, delete, view, edit storage resources
- View and edit Blob, Queue, Table, File, Cosmos DB storage and Data Lake Storage



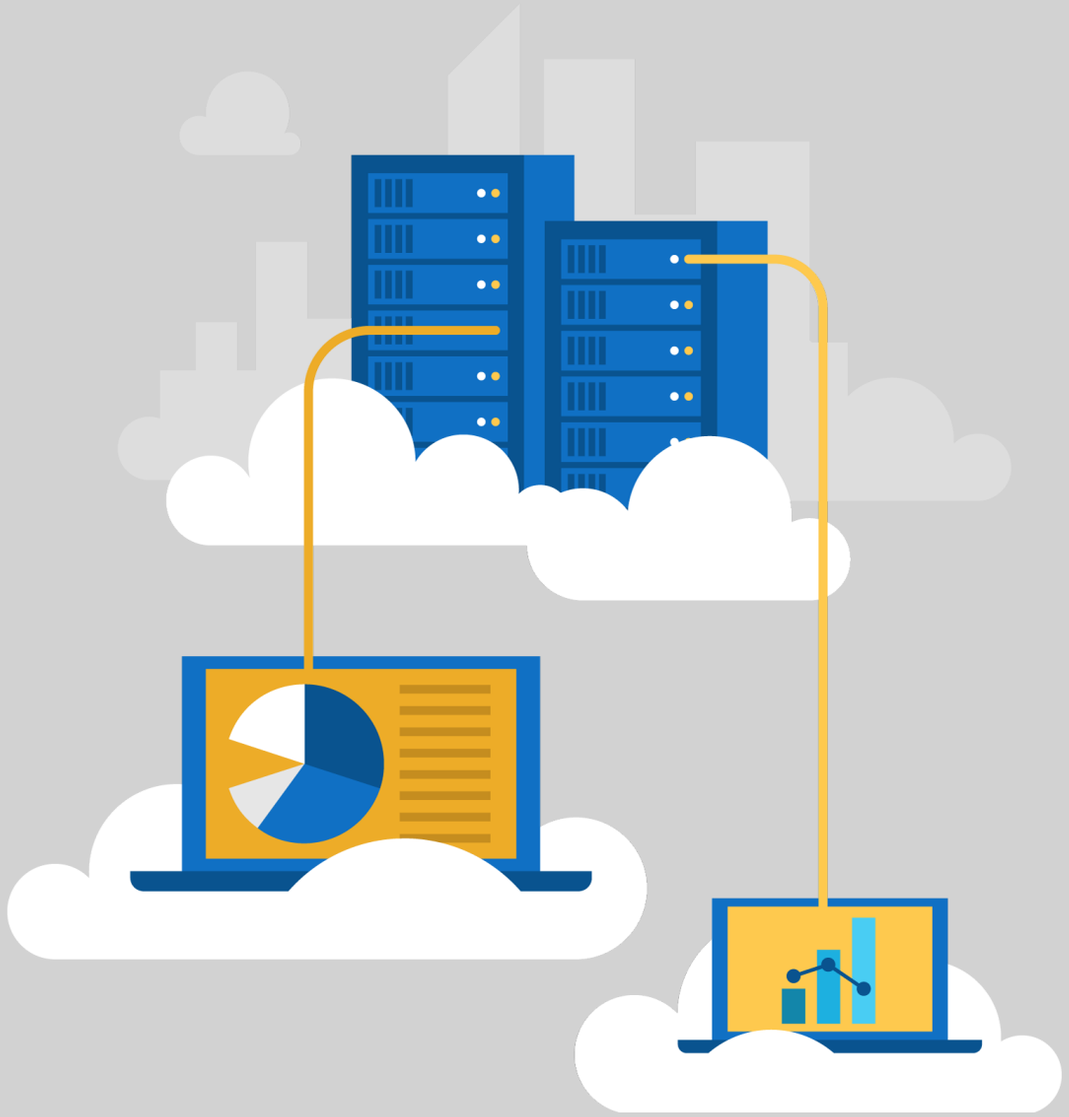

- Obtain shared access signature (SAS) keys
- Available for Windows, Mac, and Linux

# Storage Explorer Functionality

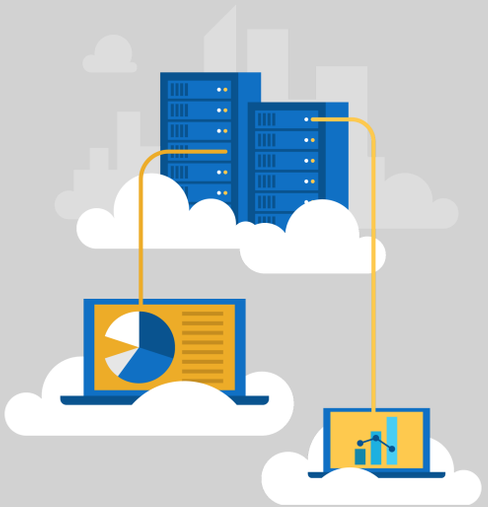
- Connect to an Azure subscription
- Work with local development storage
- Attach to external storage
- Attach a storage account by using an SAS
- Attach a service by using an SAS
- Connect to an Azure Cosmos DB account by using a connection string

Microsoft

# Design a Compute Strategy



# High Availability

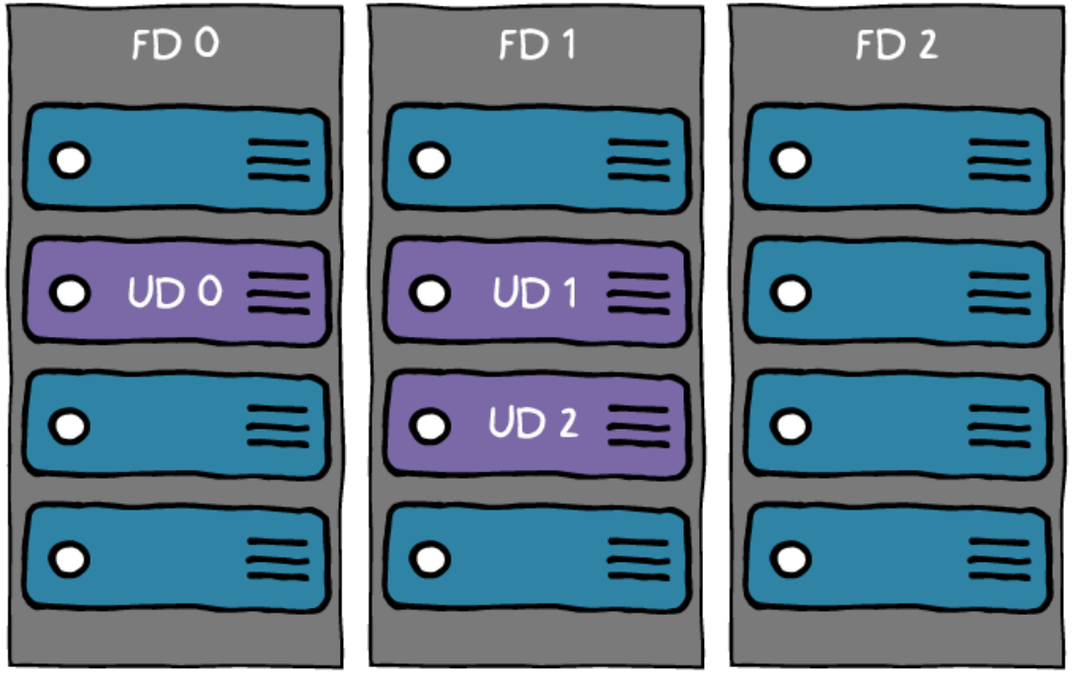


# Azure Availability

- Availability of Azure VMs can be impacted by:
  - **An unexpected downtime**
  - **Planned maintenance events**
  - **Emergency maintenance events**
- To optimize availability of Azure VMs:
  - **Deploy two or more identically configured Azure VMs into the same availability set or different availability zones.**
  - **Use managed disks for Azure VMs deployed into availability sets and availability zones.**
  - **Keep track of Azure Scheduled Events to respond proactively to planned downtime.**
  - **Place each tier of multi-tier applications in a separate availability set or zone.**
  - **Use a load balancer in combination with availability sets and availability zones.**
  - **For individual VMs running critical workloads, use exclusively Premium Storage disks.**

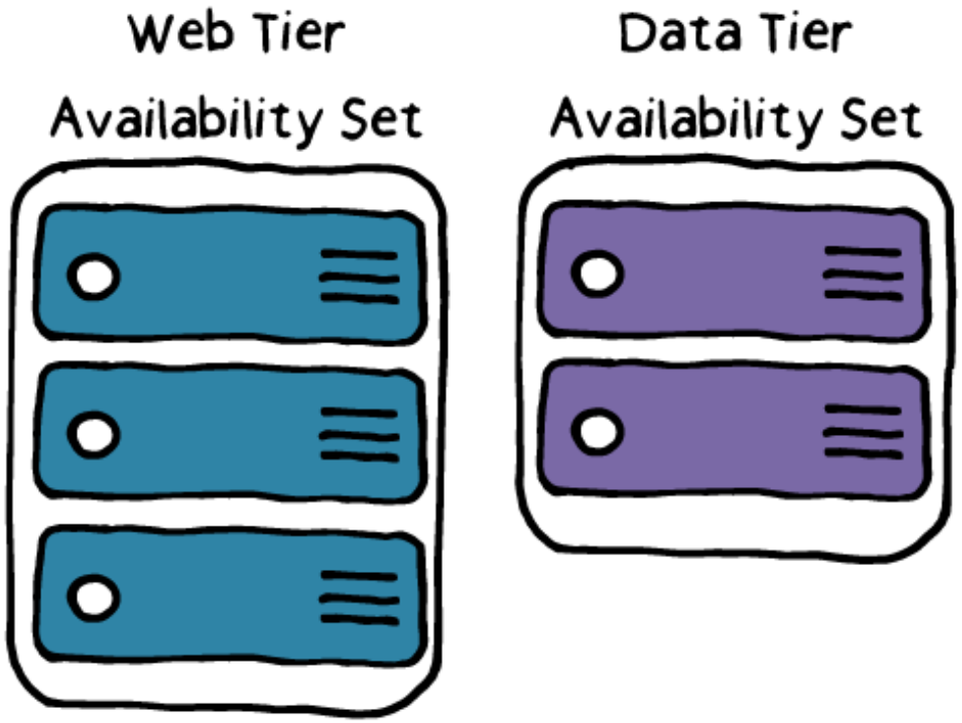
# Availability Sets

- Offer 99.95% availability for workloads running on two or more VMs.
- Minimize impact of planned and unplanned downtime.
- Enforce placement of Azure VMs across separate racks in the same datacenter.
- Comprise of:
  - **Fault domains**
  - **Update domains**



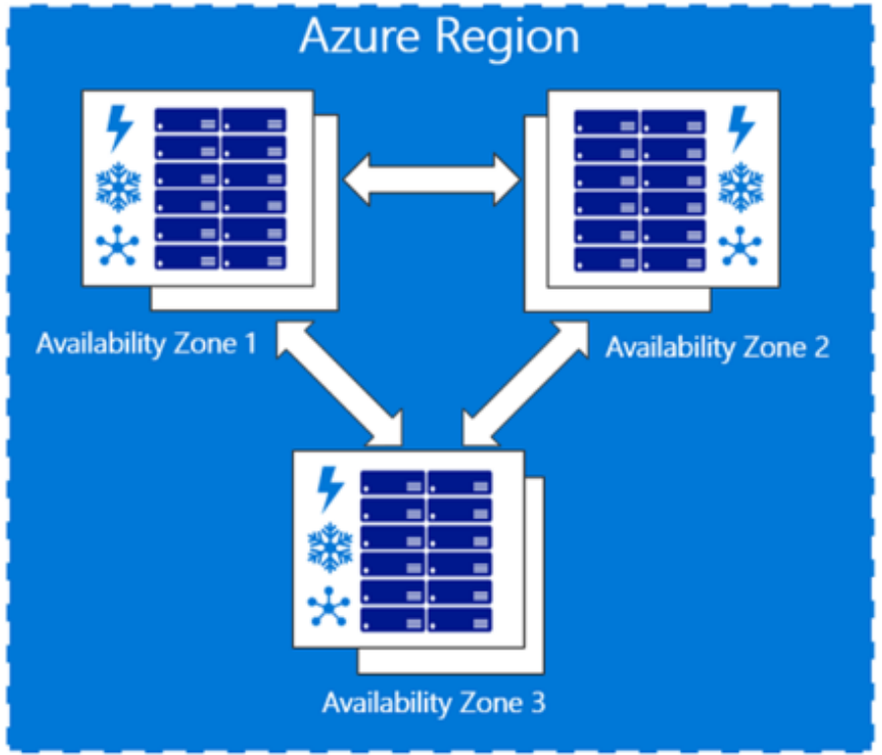
# Multiple Availability Sets

- In a multi-tier solution, place each tier in a separate availability set:
  - **The solution remains functional in case of downtime of individual VMs (one per tier)**
  - **For example:**
    - **Web tier availability set**
    - **Data tier availability set**

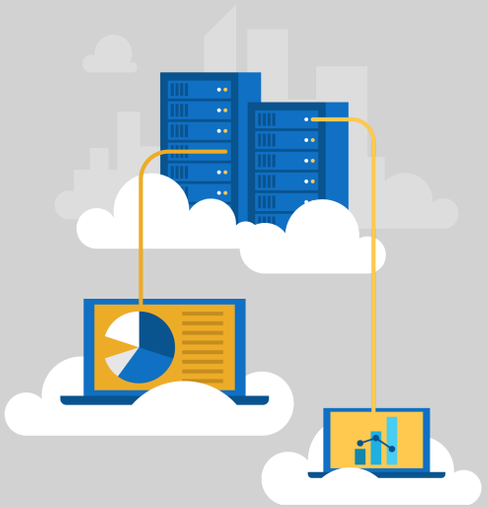


# Availability Zones

- Offer 99.99% availability for workloads running on two or more VMs.
- Minimize impact of planned and unplanned downtime
- Enforce placement of Azure VMs across separate datacenters (zones) in the same Azure region.
- Implicitly comprise of:
  - **Fault domains**
  - **Update domains**



# Templated Infrastructure

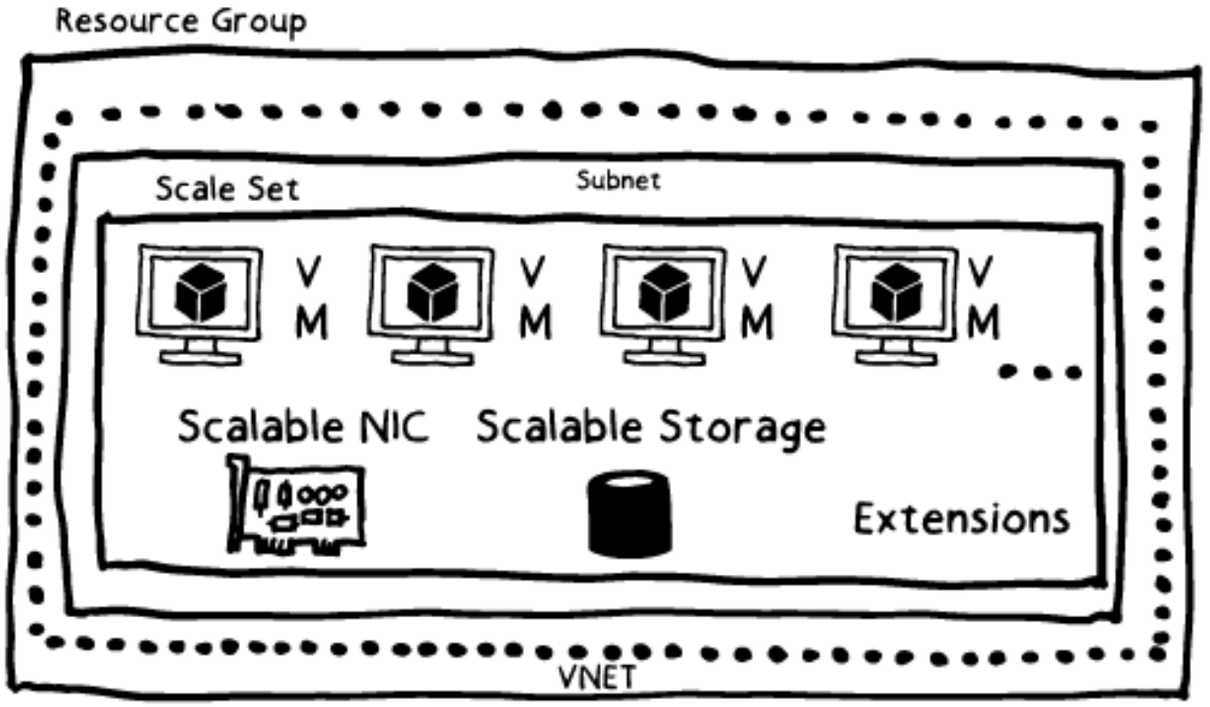


# Templated Infrastructure

- Azure VM Scale Sets:
  - **Offer high degree of control of resources**
  - **Auto-scale up to 1,000s of Azure VM instances**
  - **Are optimized for template-based deployments**
  - **Are suitable for hosting big compute and big data workloads**
  - **Automatically provision necessary storage, compute, and network resources**

# Virtual Machines Scale Sets (VMSS)

- Deploy into a subnet of a virtual network
- Provide access to individual VMs via NAT
- Scale for up to 1,000 VMs

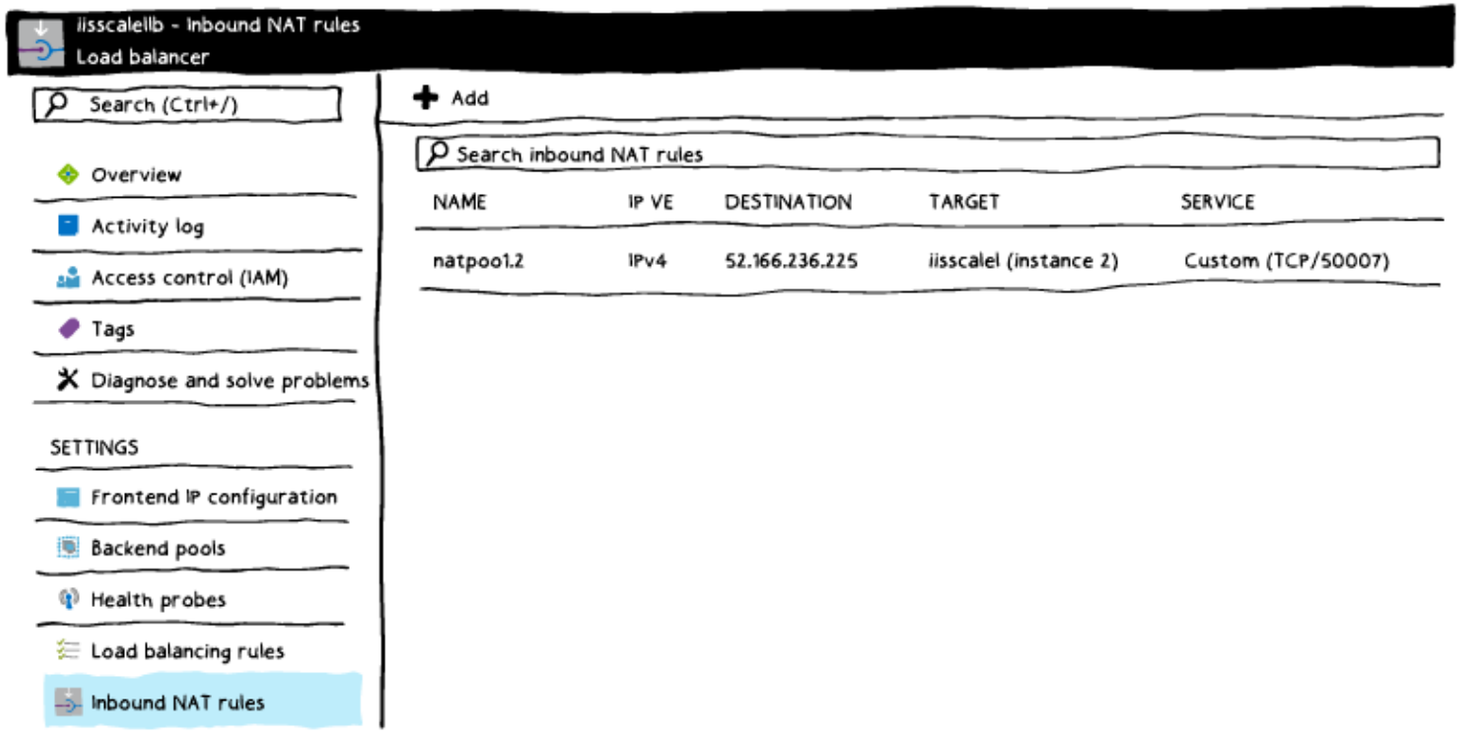


# Virtual Machines vs. Virtual Machine Scale Sets

- Azure VM Scale Sets:
  - **Support auto-scaling.**
  - **Offer overprovisioning to accelerate the speed of deployment and scaling out.**
  - **Offer customizable policy controlling upgrade of all VMs in a scale set.**
  - **Support attaching new data disks to all VMs in a scale set (but not individual ones).**
  - **Do not support snapshots or image capture.**
  - **Do not support conversion from unmanaged disks to managed disks.**
  - **Do not support IPv6.**
- Azure VMs:
  - **Support custom configuration, including new or existing data disks.**
  - **Support snapshots and image capture.**
  - **Support conversion of unmanaged disks to managed disks.**
  - **Support IPv6.**

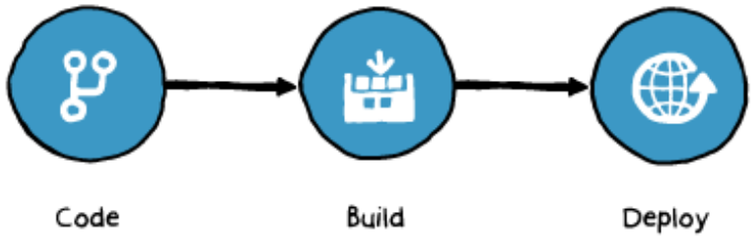
# Virtual Machine Scale Set Considerations

- To connect to a VM Scale Set instance VM form Internet:
  - **Use inbound NAT rules of an Internet-facing load balancer in front of the VM Scale Set**
  - **e.g. RDP access via 52.166.236.225:50007**



# Continuous Delivery in VMSS

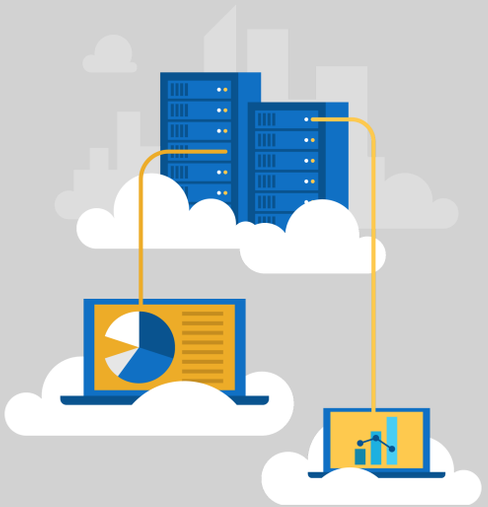
- Use continuous integration/continuous delivery pipelines in Visual Studio Team Services to update an application running in a VMSS:
  - **Code update triggers a new build**
  - **The new built triggers VMSS update:**
    - **By creating and redeploying an updated OS image**
    - **By updating the application  (via a VM extension)**
  - **Image based updates offer some benefits:**
    - **Predictability**
    - **Easy roll-back**
    - **Better scaling (no code to install on each VM as it is deployed)**



# Large VM Scale Sets

- VM Scale Sets with 100+ VMs:
  - **Require the singlePlacementGroup property set to false**
  - **Require the use of managed disks.**
  - **Scale up to 1,000 VMs when based on Azure Marketplace images.**
  - **Scale up to 300 VMs when based on custom images.**
  - **Require the use of Azure Load Balancer Standard SKU for layer-4 load balancing.**
  - **Support Azure Application Gateway for layer-7 load balancing.**
  - **Require sufficiently sized IP subnet (all VMs in the same VMSS are deployed into the same IP subnet)**
  - **Might require an increase in vCPU subscription level quotas.**

# Domain-Joined Virtual Machines

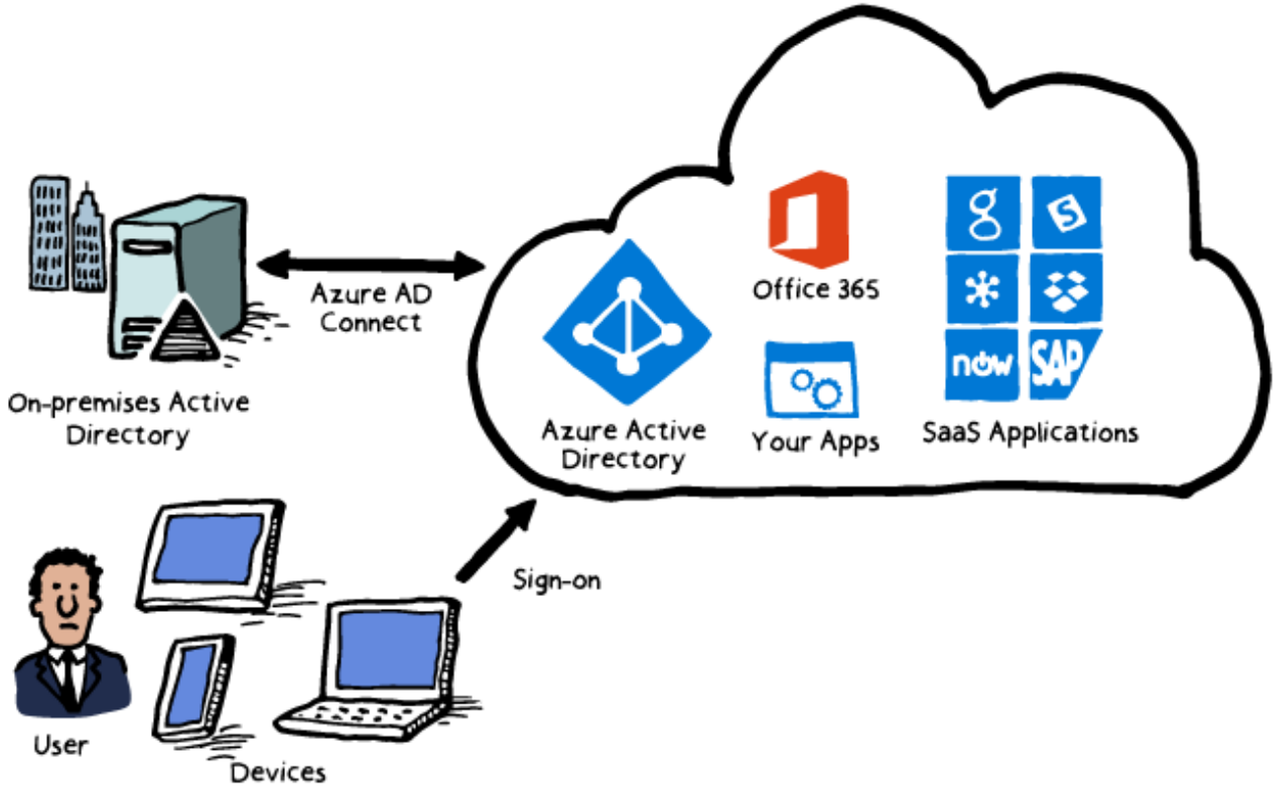


# Domain and IaaS Applications

- Identity services in Azure cloud and hybrid environments include:
  - Active Directory Domain Services (AD DS)
  - Azure AD
  - Azure AD DS

# Hybrid Connectivity

- To integrate AD DS with Azure AD, use Azure AD Connect
- Azure AD Connect:
  - **Synchronizes AD DS with Azure AD**
  - **Supports several SSO scenarios:**
    - **Password Hash Synchronization**
    - **Pass-through authentication**
    - **AD FS**



# Azure AD Domain Services

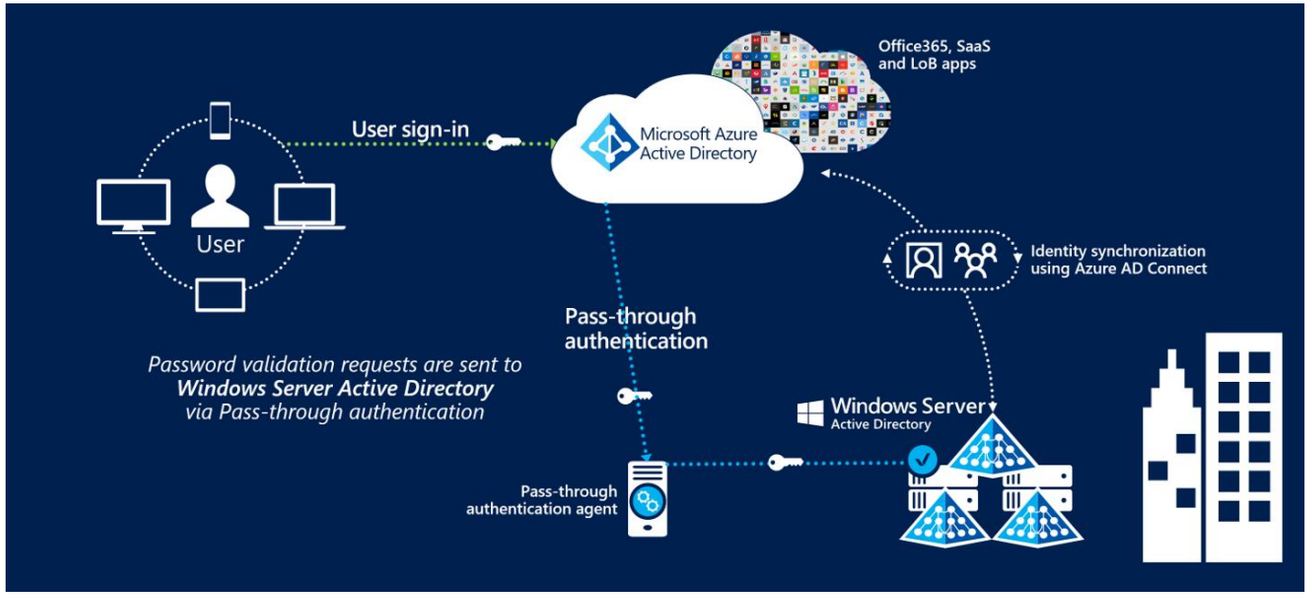
- Managed AD DS in Azure:
  - **Simple to deploy (a few clicks in the Azure portal)**
  - **No IaaS infrastructure required to provide AD DS authentication to Azure VMs.**
  - **Compatible with AD DS, including support for LDAP, Kerberos, NTLM, Group Policy, and domain join.**
  - **Cost-effective—No need to pay for Azure IaaS Virtual Machines.**
- Supports cloud-only and hybrid scenarios.
- In Azure AD DS cloud-only scenarios (greenfield deployments):
  - **All Azure AD DS objects are created and managed in Azure AD**
  - **There is no need to manage AD DS domain or its infrastructure.**

# Hybrid Cloud Tenant

- In Azure AD DS hybrid scenarios (an existing AD environment):
  - **All Azure AD DS objects are created and managed in AD DS.**
  - **It is necessary to use Azure AD Connect and enable password hash synchronization.**
  - **Users sign in to Azure AD DS using their AD DS credentials.**
  - **Managed Azure AD DS domain is separate from AD DS domain.**
  - **There is no need to manage Azure AD DS domain or its infrastructure.**

# Azure AD Pass-Through Authentication

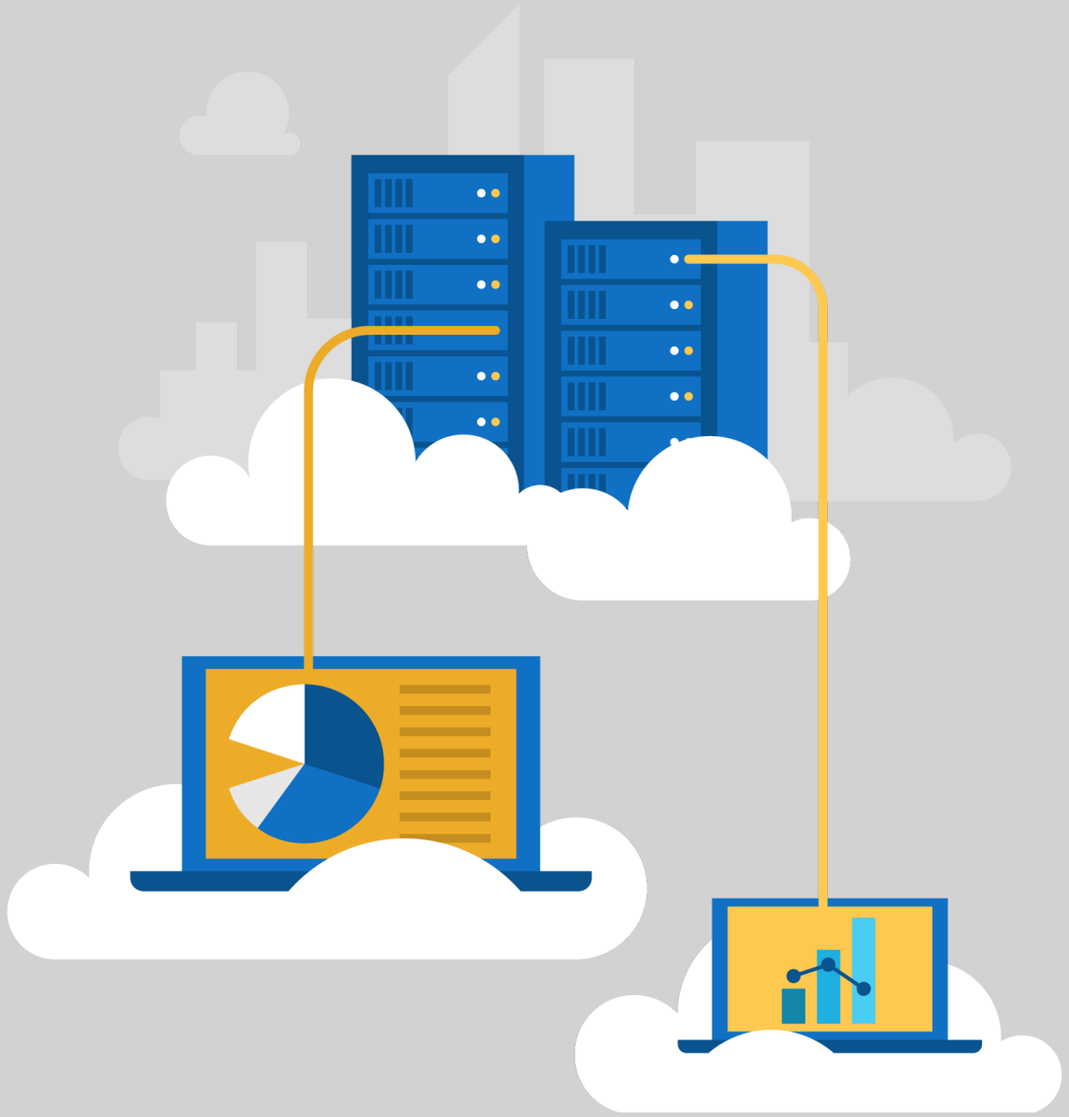
- Azure AD Connect-based and agent-based functionality:
  - **Supports single sign-on without:**
    - **AD FS**
    - **Password hash synchronization**
  - **Uses AD DS to validate passwords**



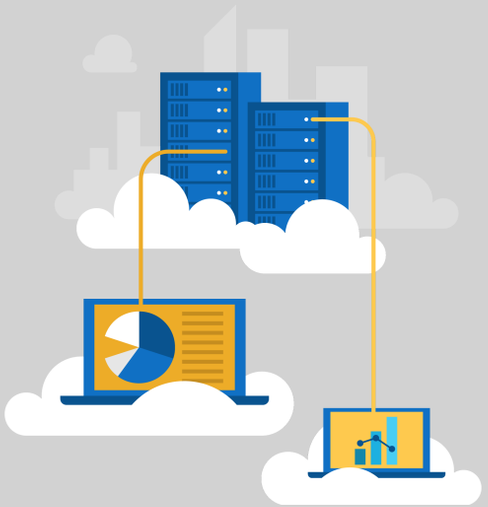

- Benefits:
  - **Simplified user experience (SSO)**
  - **Support for self-service password reset**
  - **Simple, lightweight agent-based implementation.**
  - **No need for changes to network perimeter (the agent communicates outbound)**
  - **Support for Azure AD Conditional Access policies, including MFA.**
  - **Support for high availability (by installing additional agents).**

Microsoft

# Design a Networking Strategy



# Virtual Networks

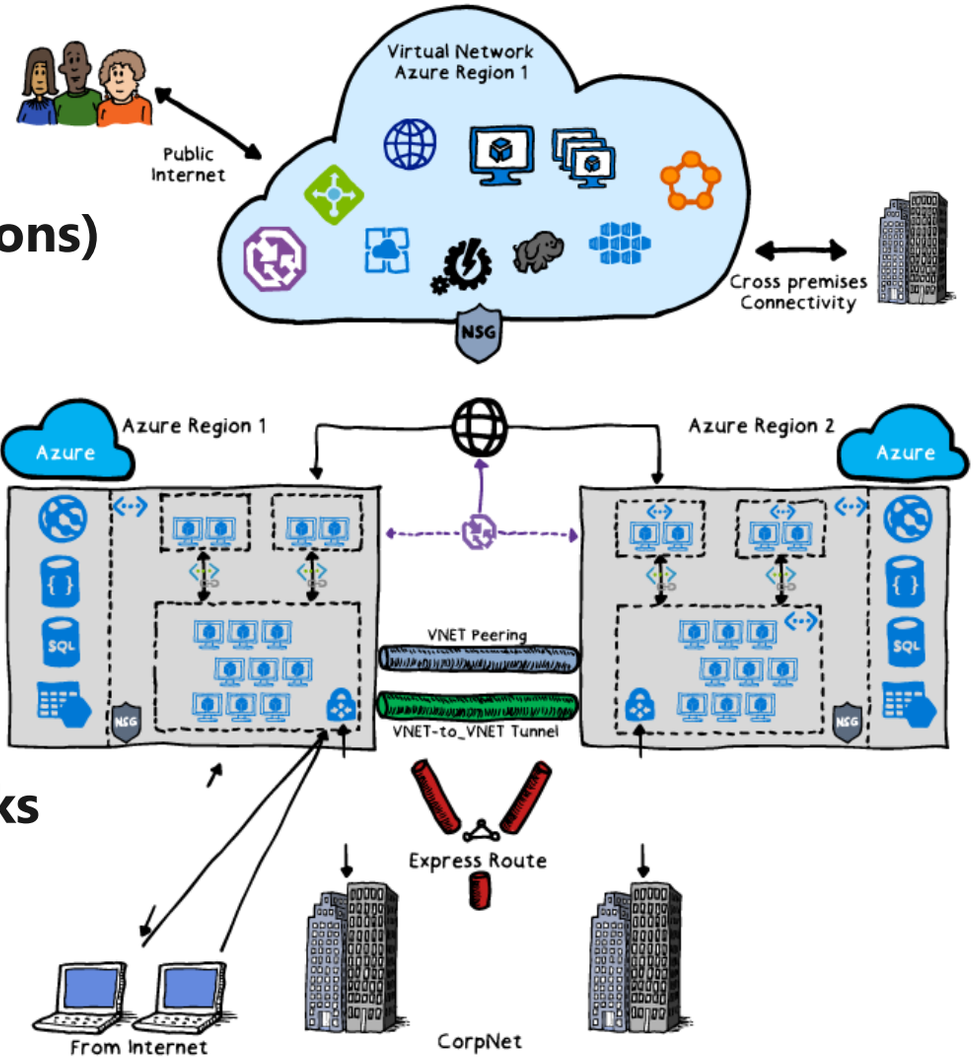


# Azure Virtual Network (VNET) Architecture

- Virtual networks constitute a core component of Azure networking:
  - **Facilitate connectivity:**
    - **Between Azure VMs in the same Azure region**
    - **Between Azure VMs across Azure regions**
    - **Between on-premises networks and Azure VMs**
  - **Support fundamental networking features:**
    - **IP addressing**
    - **DNS name resolution**
    - **Network traffic filtering**
    - **Routing**

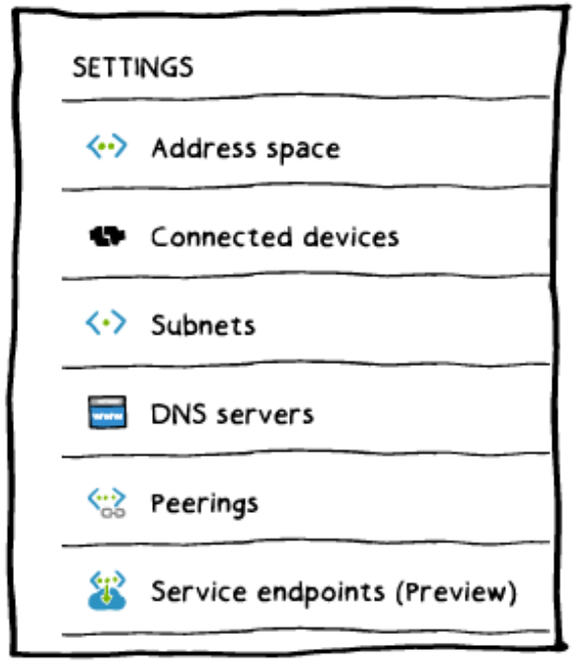
# Multi-Region Virtual Network Architecture

- Virtual network communication flows:
  - **Within a virtual network**
  - **Between virtual networks (the same or different regions)**
  - **Hybrid connectivity with on-premises networks**
  - **From a virtual network to Internet (outbound)**
  - **From Internet to a virtual network (inbound)**
- Virtual network connectivity:
  - **VNET Peering: between virtual networks**
  - **ExpressRoute: Hybrid only**
  - **Site-to-Site VPN: hybrid and between virtual networks**
  - **Load Balancers:**
    - **Traffic Manager: global load balancing from Internet**
  - **Network Security Groups: traffic filtering**

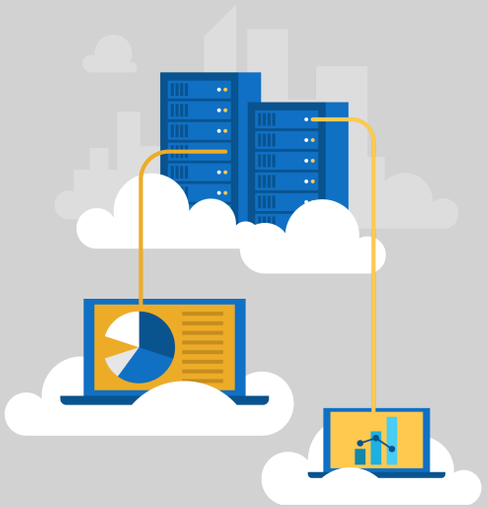


# VNETs & Subnets

- **Networking Topology**
  - Define one or more VNETs in an Azure region and provide a non-overlapping IP address space for each
  - Define one or more subnets within a VNET and provide an IP address range for each
  - Configure NSGs for each subnet
  - Deploy Azure VMs into subnets
- **Subnet IP addressing:**
  - Static and dynamic (default) IP address assignments are supported
  - 5 IP addresses in each subnet are reserved for internal use
- **Public IP addressing:**
  - Required for inbound traffic from Internet
- **Private IP addressing:**
  - Used for cross-premises and within/cross-VNET communication.
- **DNS name resolution:**
  - Configurable on the VNET and individual vNIC level
  - Azure provided DNS or custom DNS configuration

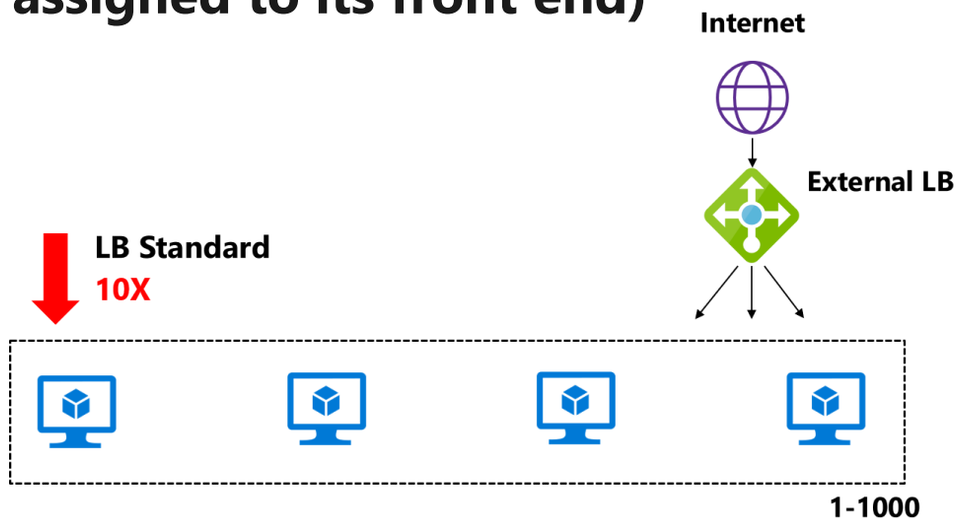


# Load Balancing



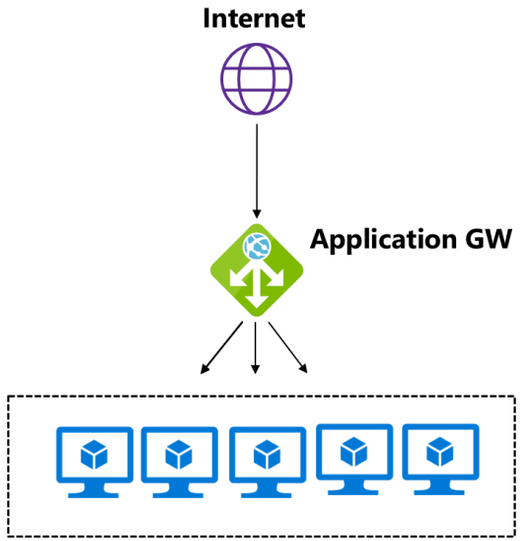
# Load Balancing Solutions

- Azure offers a number of Azure load balancing services:
  - **Azure Load Balancer**
  - **Azure Application Gateway**
  - **Azure Marketplace Load Balancing Appliance**
  - **Azure Traffic Manager**
- Azure Load Balancer – managed layer 4 (TCP/UDP) load balancer:
  - **Internal or External (private IP or public IP address assigned to its front end)**
  - **Basic SKU (free service):**
    - **Up to 100 Azure VMs in the same availability set**
    - **NSGs are optional**
  - **Standard SKU (paid service):**
    - **Up to 1,000 Azure VMs with availability zones support**
    - **NSGs are mandatory**
    - **Support for HA ports**



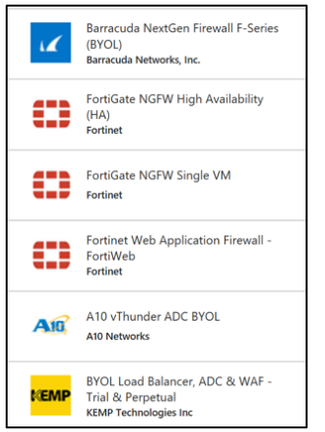
# Azure Application Gateway

- Managed layer 7 (HTTP/HTTPS) load balancer:
  - **SSL Offloading**
  - **HTTP to HTTPS redirect**
  - **Cookie Affinity**
  - **URL Based Routing**
  - **Web Application Firewall (WAF) protects against:**
    - **SQL Injection**
    - **Cross-site scripting**
    - **Protocol violations**
    - **Generic attacks**
    - **HTTP rate limiting**
    - **Scanner detection**
    - **Session fixation**
    - **LFI/RFI**



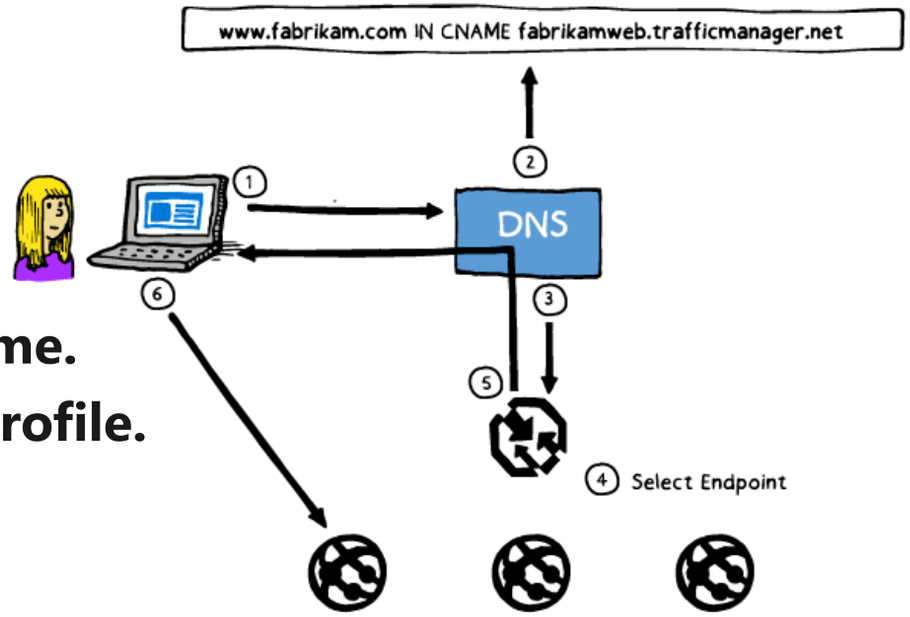
# Azure Load Balancing Marketplace Appliances

- Supplement built-in Azure load balancing services:
  - **BYOL: allows customers to reuse their existing licenses when migrating to Azure**
  - **Pay-Per-Use: the usage cost include licensing fees**

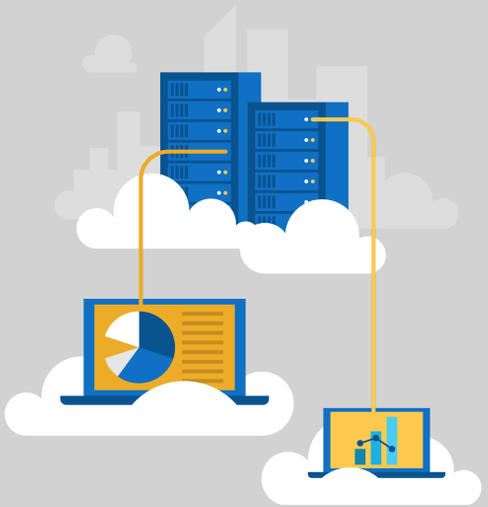



# Azure Traffic Manager

- A managed DNS-based global load balancing service
- Supports different traffic distribution algorithms:
  - **Weighted: distributes requests across endpoints according to custom weight values**
  - **Failover: always directs to the primary endpoint but supports failover to a secondary**
  - **Performance: directs requests to the closest endpoint**
  - **Geography: directs requests to the designated region**
- Traffic Manager workflow:
  - **1. A user request to the company DNS name.**
  - **2. The company DNS name to a Traffic Manager DNS name.**
  - **3. The Traffic Manager DNS name to a Traffic Manager profile.**
  - **4. Processing of the Traffic Manager profile rules.**
  - **5. An endpoint DNS domain name to the user.**
  - **6. The user request to the endpoint.**

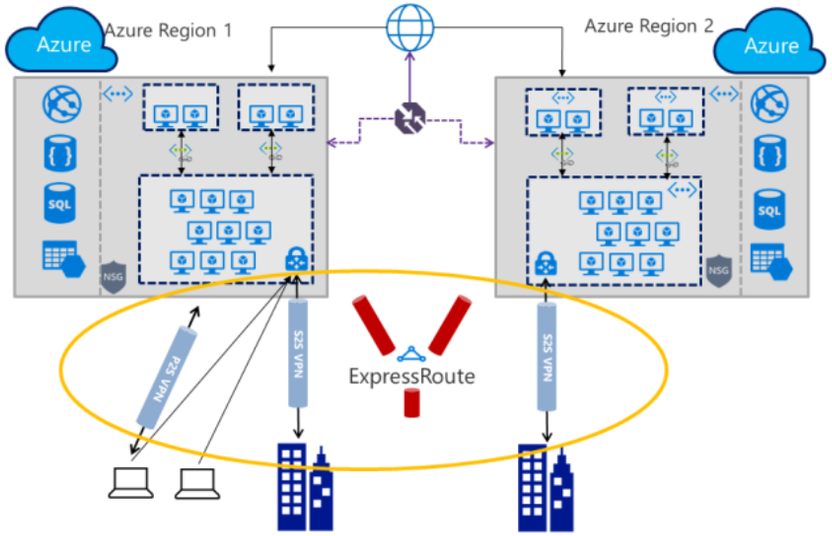


# External Connectivity



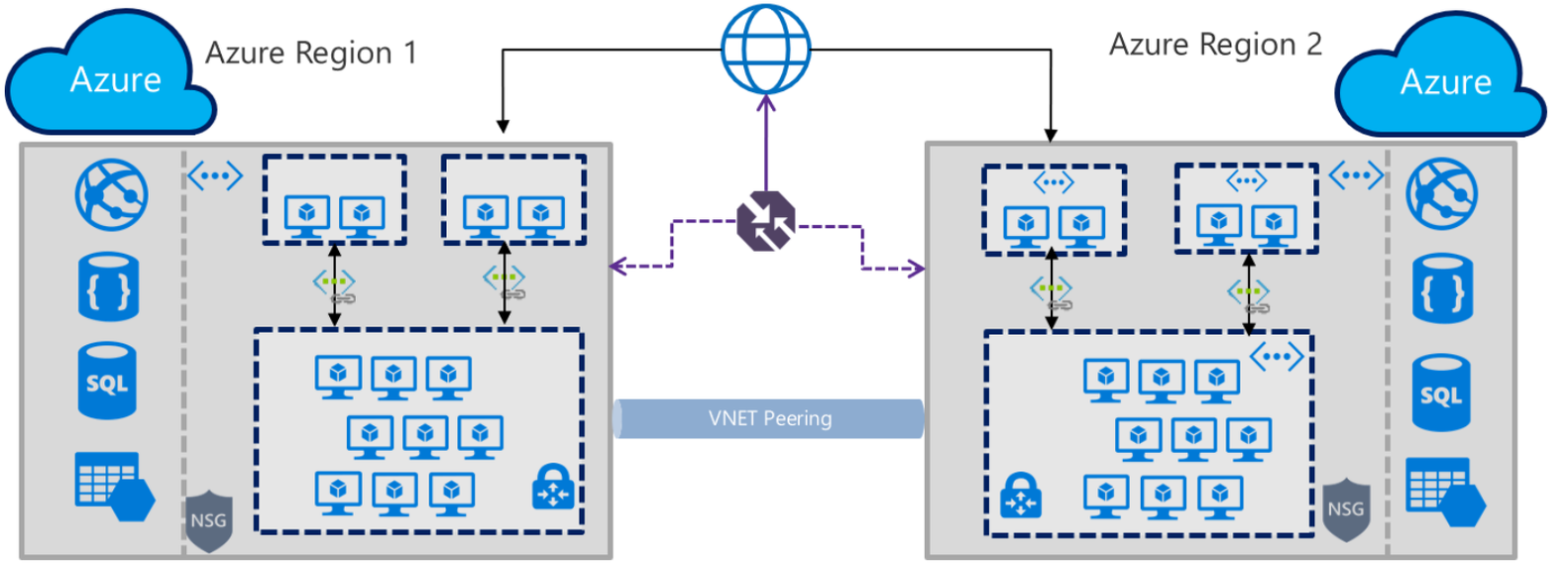
# On-Premises to Azure Connectivity

- There are three methods that facilitate hybrid connectivity to Azure:
  - **ExpressRoute:**
    - **Private connectivity between on-premises networks and Azure VNETs**
    - **Private connectivity between on-premises networks and Azure public services**
    - **The lowest latency and the highest throughput**
    - **Superior redundancy via multiple circuits**
    - **Support for ExpressRoute-VPN co-existence**
  - **Site-to-Site VPN:**
    - **VPN connectivity between on-premises networks and Azure VNETs**
    - **Built-in security provided by IPSec tunneling**
    - **Support for active-active configuration via BGP**
  - **Point-to-Site VPN:**
    - **VPN connectivity between individual computers and Azure VNETs**
    - **Support for Window and macOS**
    - **Authentication via Active Directory, RADIUS, or certificates**

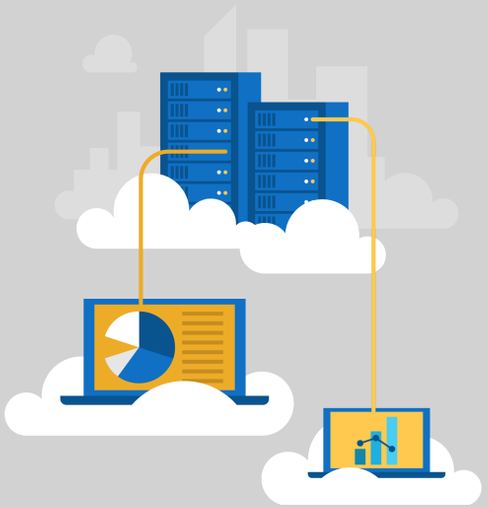


# VNET Peering

- Direct connection between two Azure VNETs:
  - **The same or different Azure regions**
  - **Traffic routed over Microsoft backbone**
  - **Support for connections to classic VNETs**
  - **No built-in encryption**
  - **Automatic routing**



# Secure Connectivity



# Network Security Groups

- Virtual network-based firewall mechanism:
  - **Can be associated to a subnet of a VNET and to a network adapter of an Azure VM**
  - **There is a limit of one NSG assignment per subnet and one per a network adapter**
  - **NSG blocks all traffic that is not explicitly allowed**
  - **NSG rules determine which traffic should be allowed or blocked:**
    - **Inbound rules**
    - **Outbound rules**
  - **Rules are processed in the order corresponding to their priority**
  - **Processing stops once the match is found**
  - **A single NSG can contains up to 1,000 rules**

# Default Network Security Group Rules

- **Have the lowest priority so can be overridden by custom rules but cannot be deleted**
- **Allow outbound connectivity to Internet and block inbound connectivity from Internet**

**Inbound**

| NAME | PRIORITY | SOURCE IP | SOURCE PORT | DESTINATION IP | DESTINATION PORT | PROTOCOL | ACCESS |
|---|---|---|---|---|---|---|---|
| ALLOW VNET INBOUND | 65000 | VIRTUAL_NETWORK | * | VIRTUAL_NETWORK | * | * | ALLOW |
| ALLOW AZURE LOAD BALANCER INBOUND | 65001 | AZURE_LOADBALANCER | * | * | * | * | ALLOW |
| DENY ALL INBOUND | 65500 | * | * | * | * | * | DENY |

**Outbound**

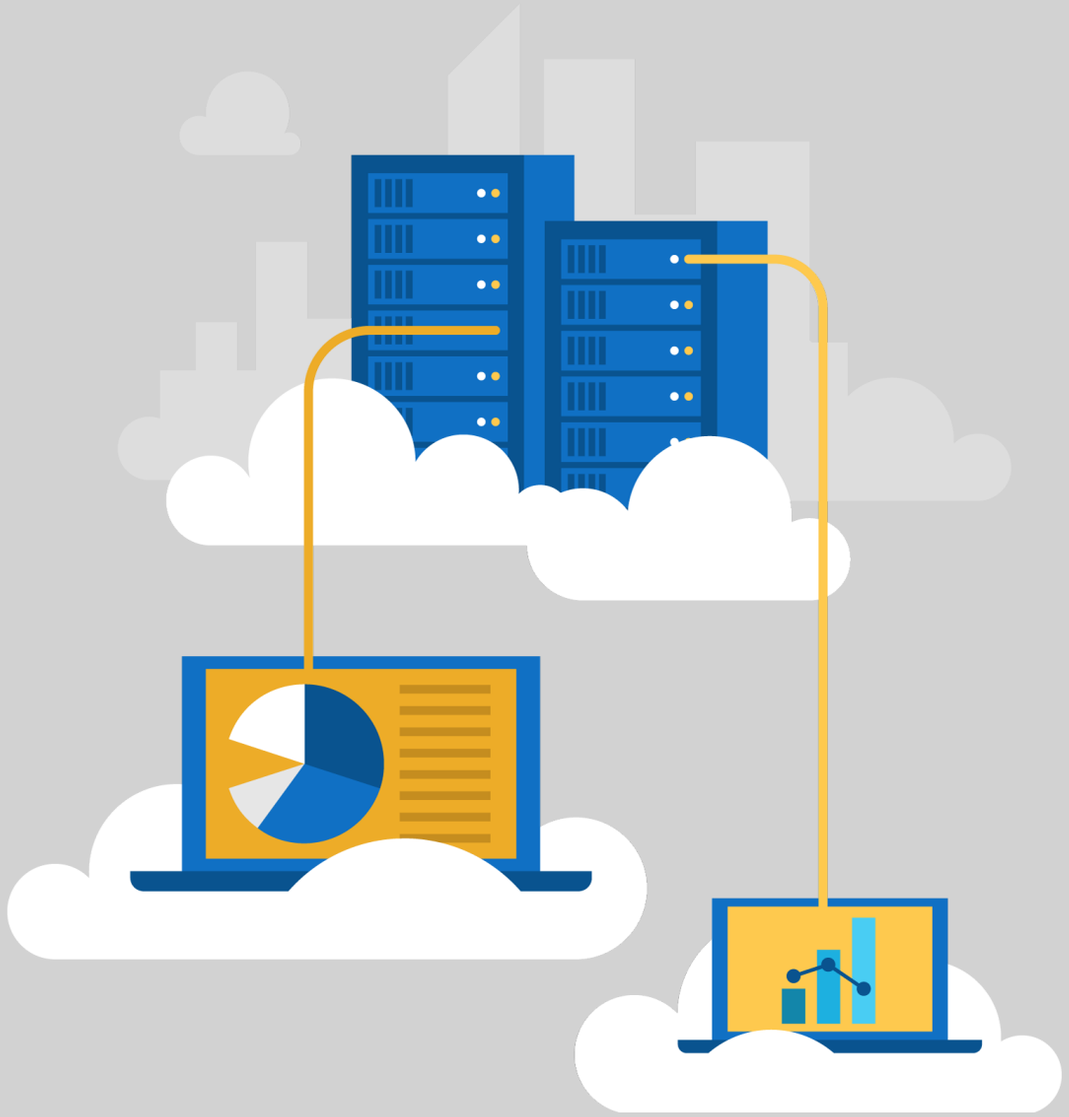
| NAME | PRIORITY | SOURCE IP | SOURCE PORT | DESTINATION IP | DESTINATION PORT | PROTOCOL | ACCESS |
|---|---|---|---|---|---|---|---|
| ALLOW VNET OUTBOUND | 65000 | VIRTUAL_NETWORK | * | VIRTUAL_NETWORK | * | * | ALLOW |
| ALLOW INTERNET OUTBOUND | 65001 | * | * | INTERNET | * | * | ALLOW |
| DENY ALL OUTBOUND | 65500 | * | * | * | * | * | DENY |

# Forced Tunneling and securing access to PaaS services

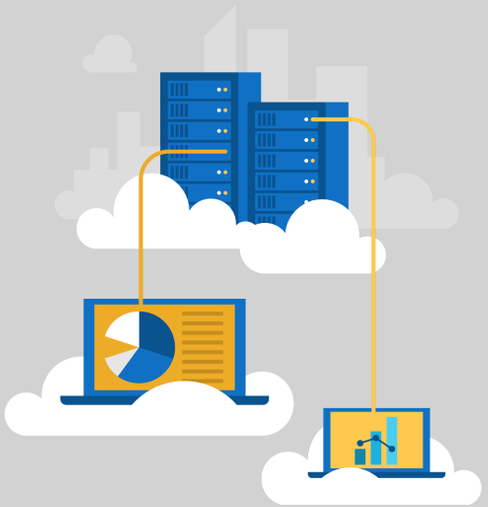
- Forced Tunneling:
  - **Routes Internet-bound traffic from an Azure VNET via an on-premises network**
  - **Can use Site-to-Site VPN or ExpressRoute**
- Securing Access to PaaS services:
  - **Relies on service endpoints**
  - **Available for:**
    - **Azure Storage**
    - **Azure SQL Database**
    - **Azure Database for PostgreSQL server**
    - **Azure Database for MySQL server**
    - **Azure Cosmos DB**
    - **Azure Key Vault**

Microsoft

# Design a Monitoring Strategy for Infrastructure



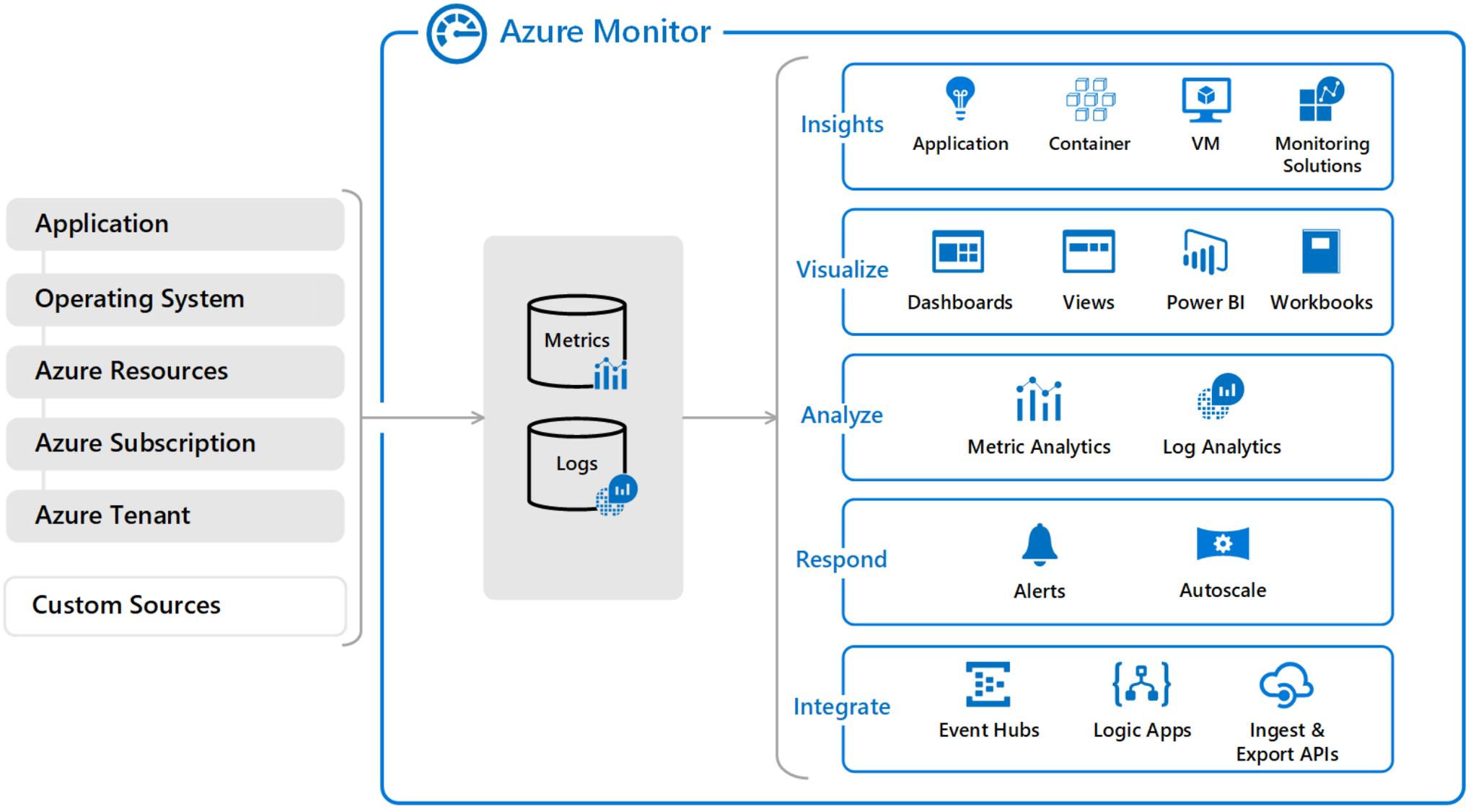
# Azure Monitor



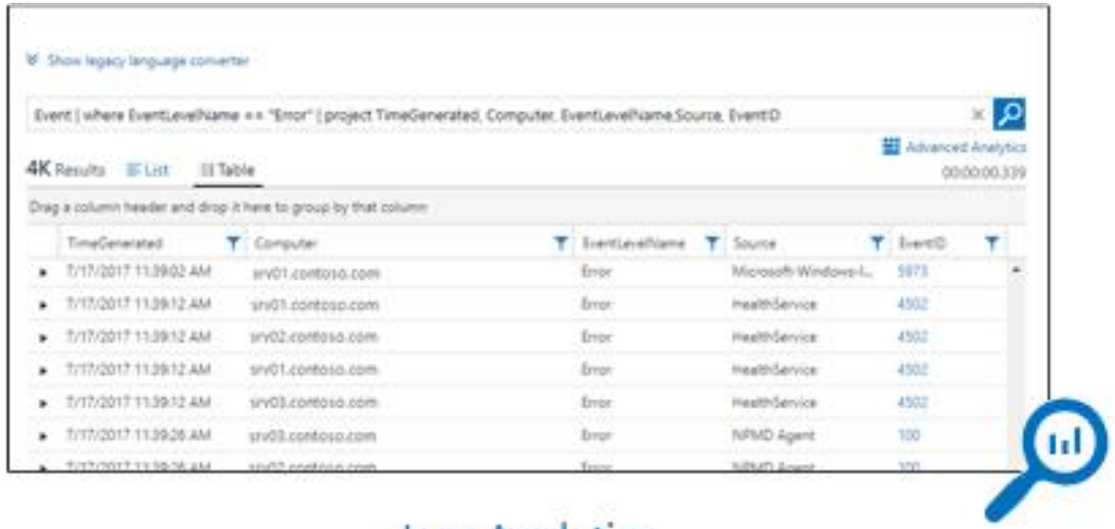
# Azure Monitor

- Centralized platform for telemetry
  - Collecting data
  - Analyzing data
  - Acting on data
- Supports cloud and on-premises environments
- Single consolidated “pane of glass”
  - Log Analytics
  - Application Insights

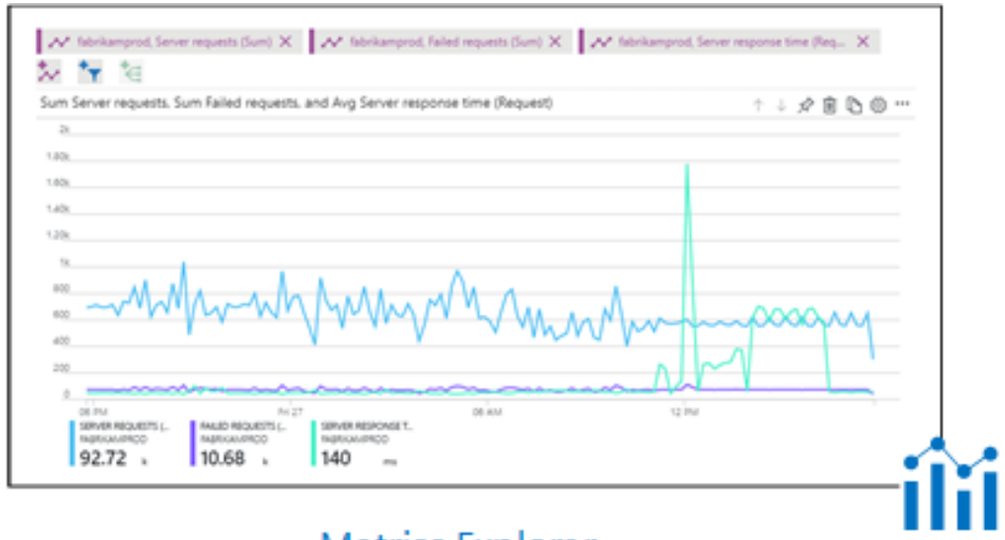
# Azure Monitor overview



# Monitoring data platform


Log Analytics

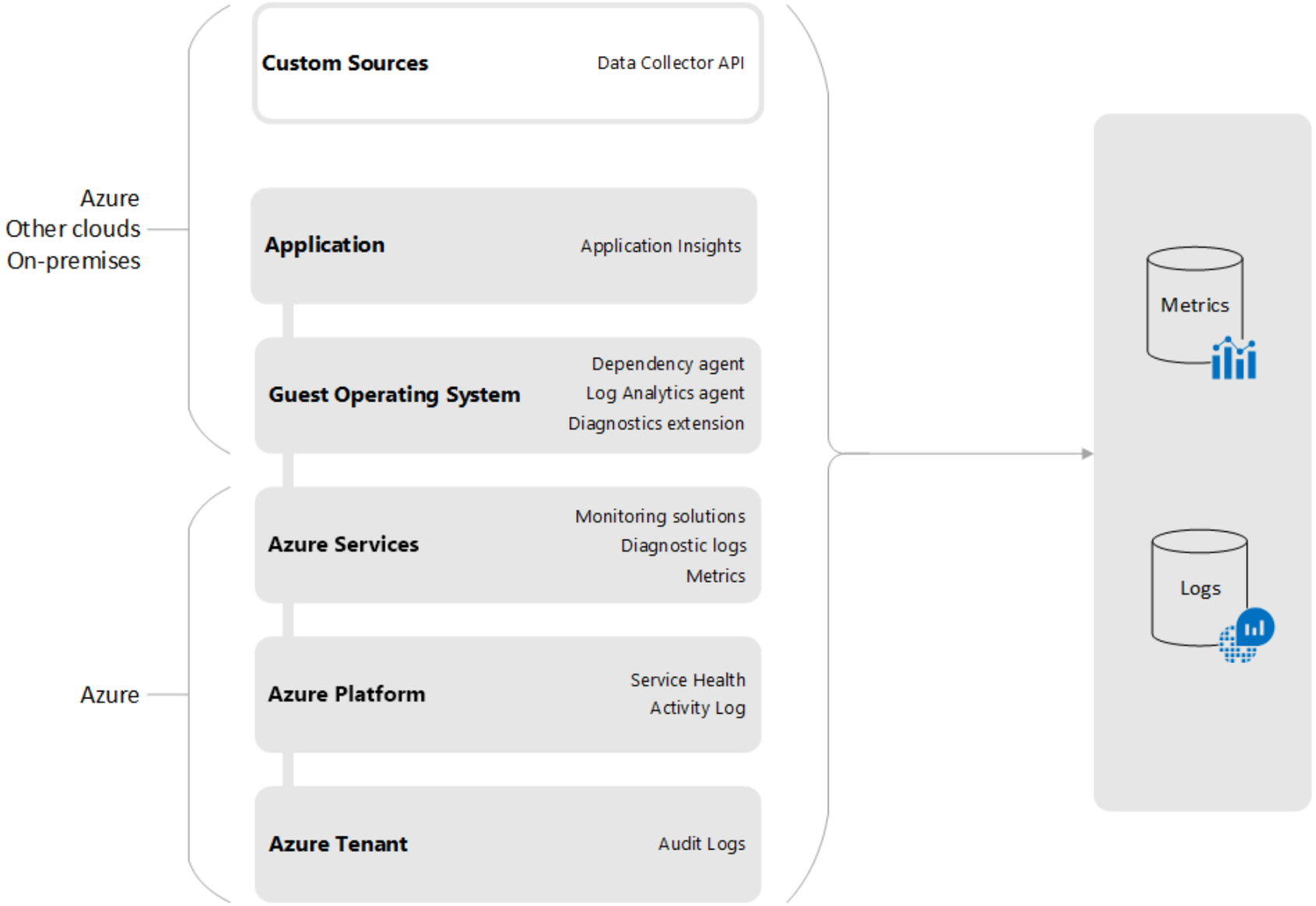




Metrics Explorer

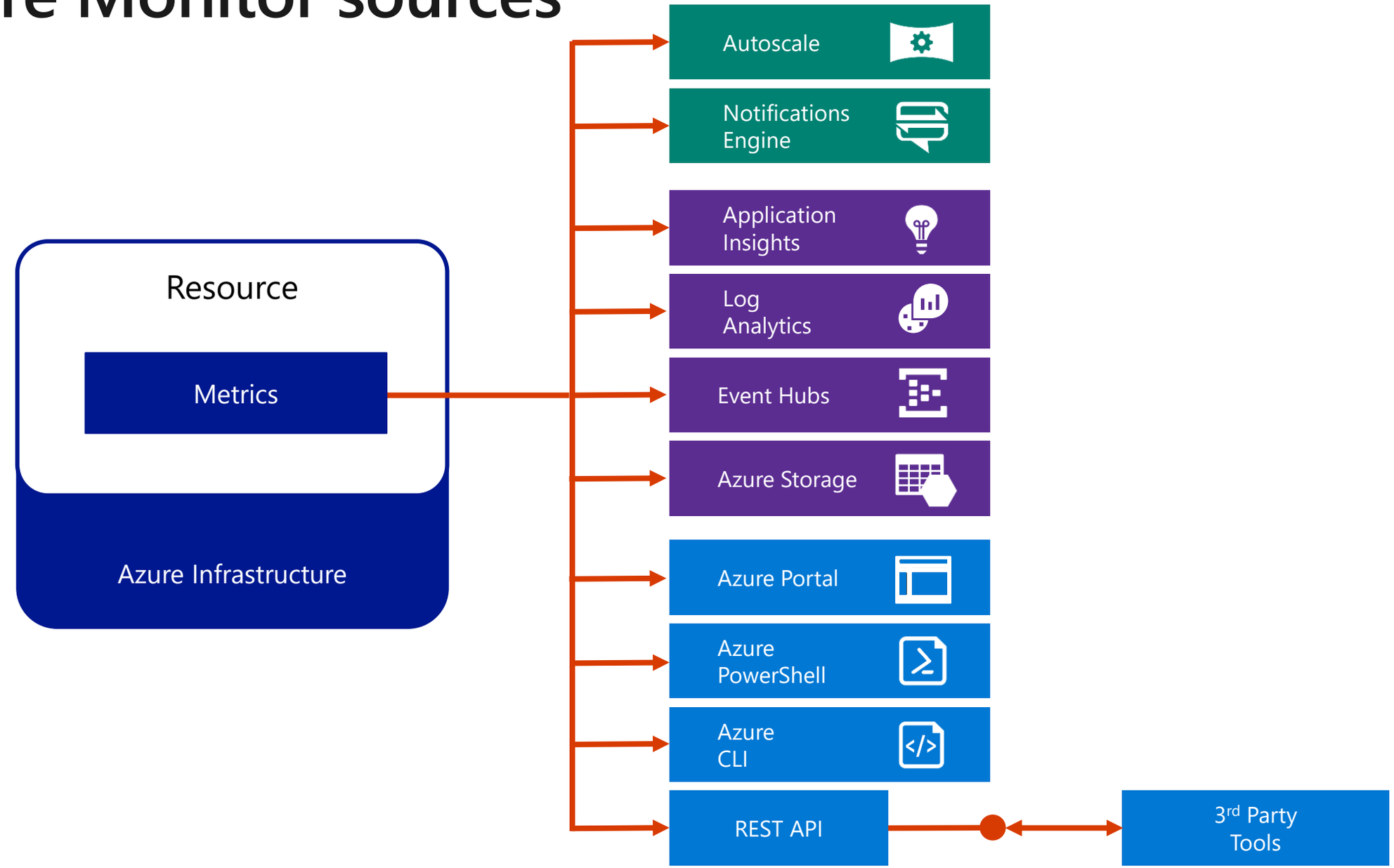
# What data does Azure Monitor collect?

- Azure Monitor collects data from many sources:
  - Application monitoring data
  - Guest OS monitoring data
  - Azure resource monitoring data
  - Azure subscription monitoring data
  - Azure tenant monitoring data

# Data sources

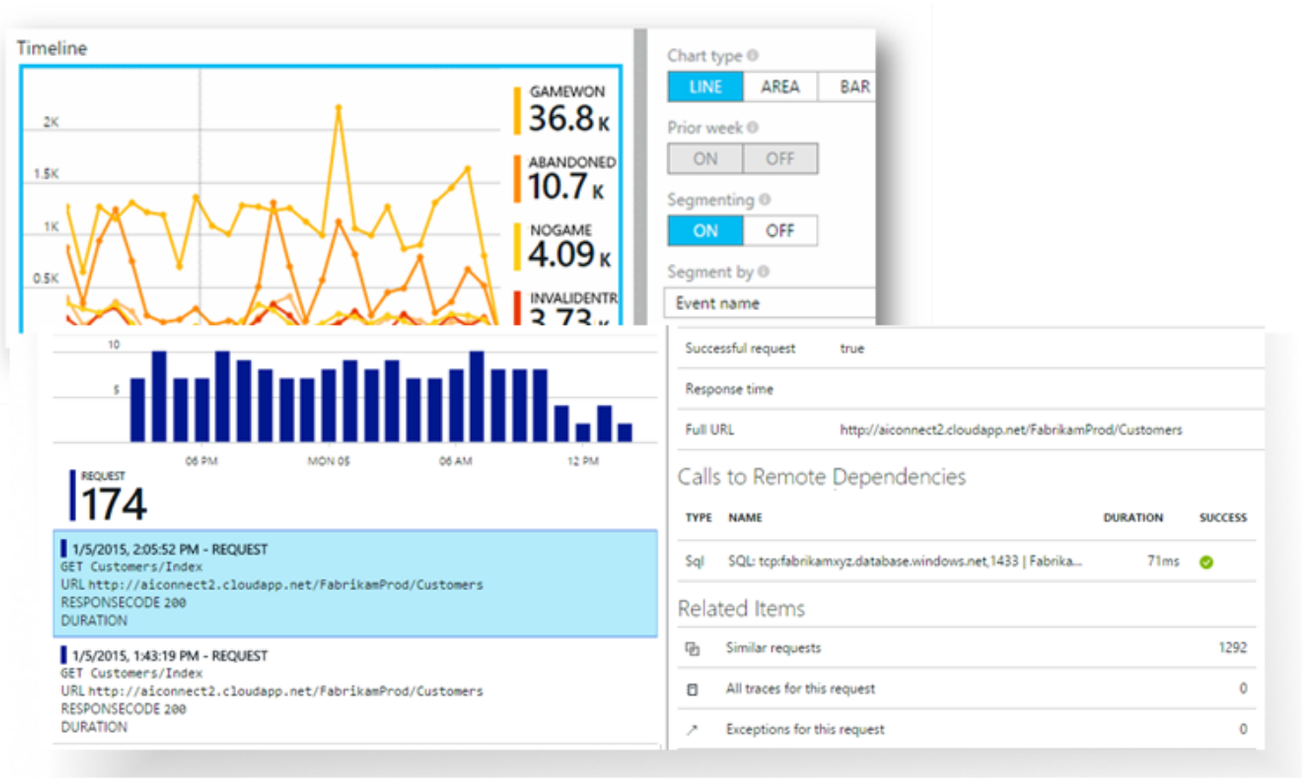


# Azure Monitor sources



# Application Insights

- Extensible application performance monitoring (APM) service for developers building and managing apps on multiple platforms
- Can be used to:
  - Monitor a live web application
  - Automatically detect performance anomalies
  - Diagnose issues by using analytical tools
  - Understand real-world user behavior by using custom queries and metric visualizations



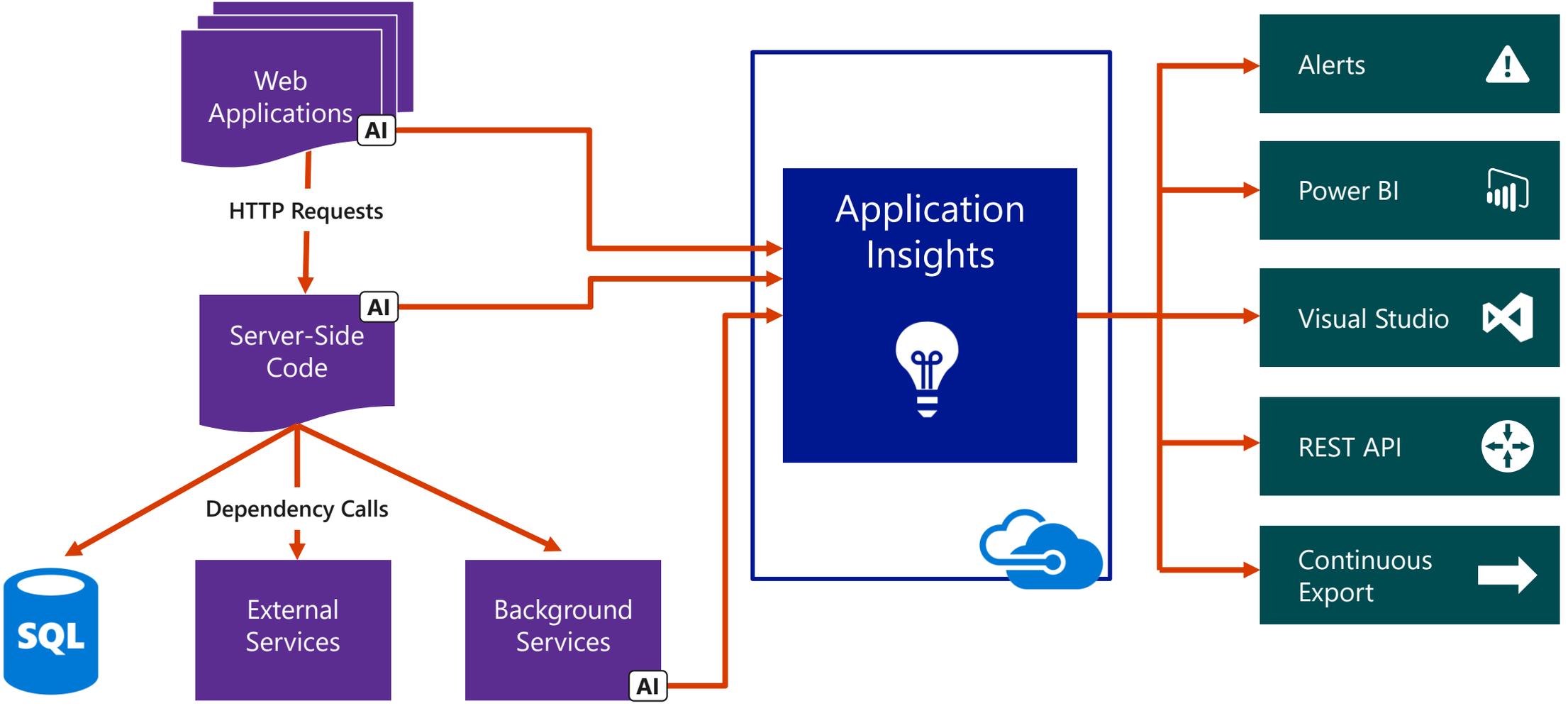
# Monitored metrics

- Request rates, response times, and failure rates
  - Find out which pages are most popular, what times of day are most popular, and where your users are. Observe which pages perform the best. If your response times and failure rates go high when there are more requests, you might have a resourcing problem.
- Dependency rates, response times, and failure rates
  - Find out whether external services are slowing you down
- Exceptions
  - Analyze the aggregated statistics, or pick specific instances and drill into the stack trace and related requests. Both server and browser exceptions are reported.
- Page views and load performance
  - Directly reported by your users' browsers
- User and session counts

# Monitored metrics (continued)

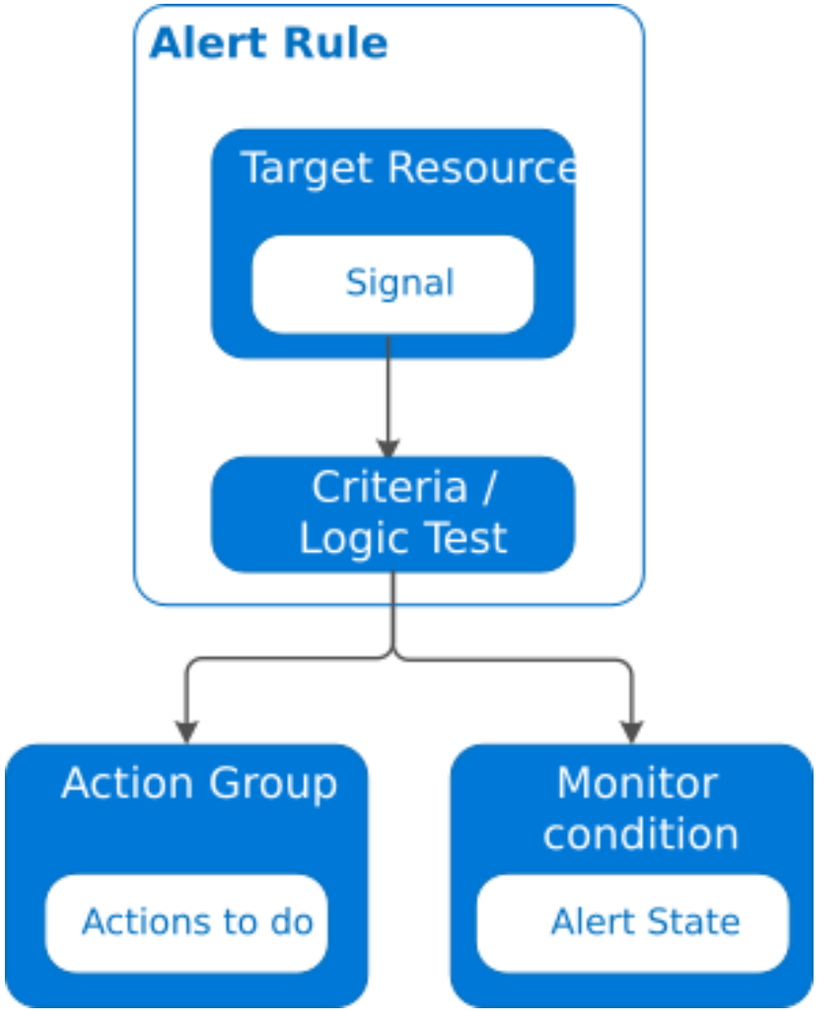
- Asynchronous JavaScript and XML (AJAX) calls
  - Rates, response times, and failure rates for these webpage-based calls
- Performance counters
  - Measured from your Windows Server or Linux server machines, such as counters for CPU, memory, and network usage
- Host diagnostics
  - Ingested from Docker or Azure
- Diagnostic trace logs
  - Logs from your app so that you can correlate trace events with requests
- Custom events and metrics
  - Custom metrics that you write yourself in the client or server code to track business events, such as the number of items sold or games won

# Application Insights architecture

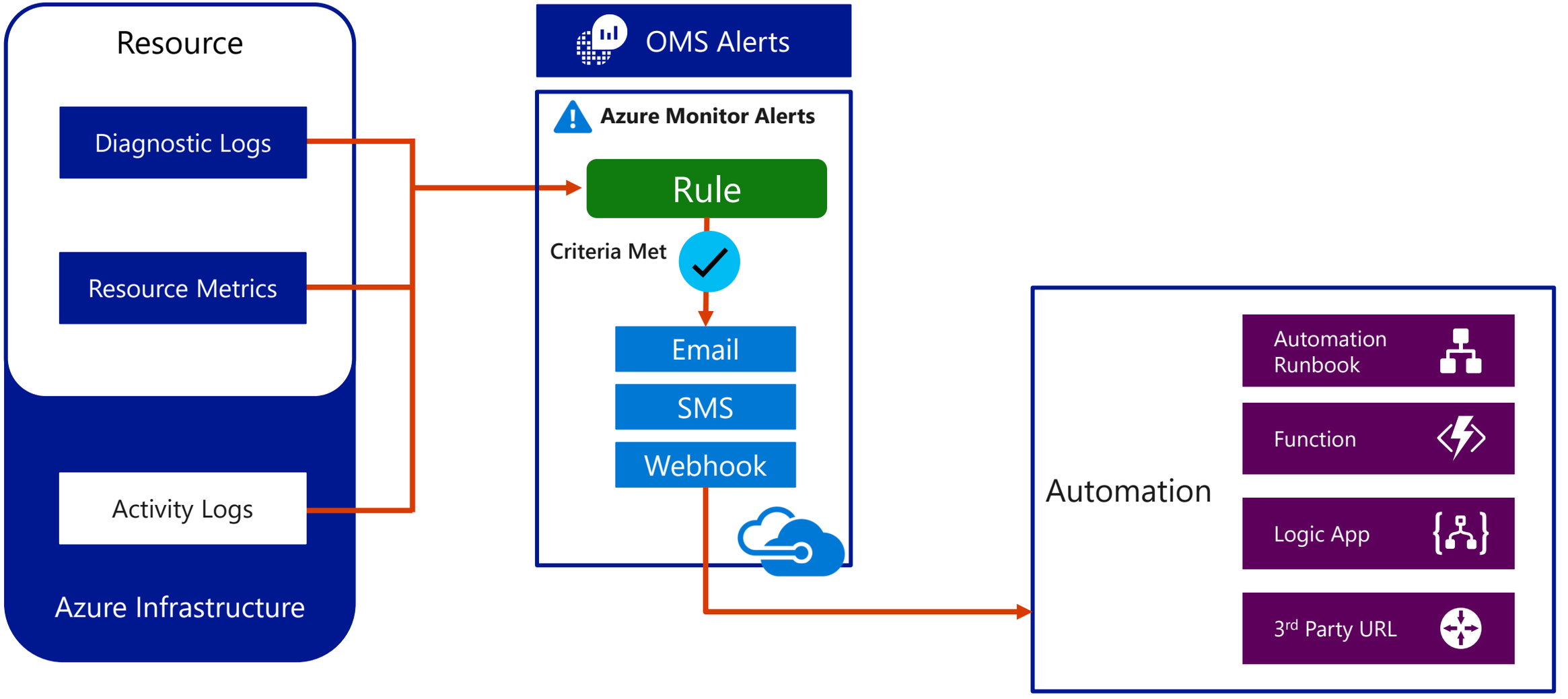


# Alerts

- Proactively notify you when conditions are met
  - Defined in alert rules
- Now unified across multiple services
  - Application Insights
  - Log Analytics
  - Azure Monitor



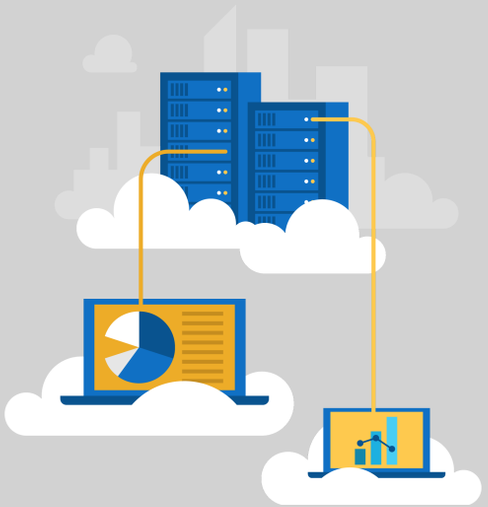
# Alerts workflow



# Alert state

| State | Description |
| --- | --- |
| **New** | The issue has just been detected and has not yet been reviewed. |
| **Acknowledged** | An administrator has reviewed the alert and started working on it. |
| **Closed** | The issue has been resolved. After an alert has been closed, you can reopen it by changing it to another state. |

# Configure instrumentation in an app or service



# Application Insights for webpages

- Monitor webpage or applications
  - Observe usage in near real-time
  - Gather performance metrics
- Can use a script to capture front-end telemetry
  - Page load time
  - Asynchronous JavaScript and XML (AJAX) calls
  - Browser exceptions
  - AJAX failures
  - User information
  - Session counts
- Segmented breakdowns
  - By users, page, client OS, browser version, geo-location, or other dimensions

# Application Insights for webpages - code

```
<script type="text/javascript">
var appInsights=window.appInsights||function(a){
  function b(a){c[a]=function(){var b=arguments;c.queue.push(function(){c[a].apply(c,b)})}}var
c={config:a},d=document,e=window;setTimeout(function(){var
b=d.createElement("script");b.src=a.url||"https://az416426.vo.msecnd.net/scripts/a/ai.0.js",d.getEle
mentsByTagName("script")[0].parentNode.appendChild(b)});try{c.cookie=d.cookie}catch(a){}c.queue=[];f
or(var
f=["Event","Exception","Metric","PageView","Trace","Dependency"];f.length;)b("track"+f.pop());if(b("
setAuthenticatedUserContext"),b("clearAuthenticatedUserContext"),b("startTrackEvent"),b("stopTrackEv
ent"),b("startTrackPage"),b("stopTrackPage"),b("flush"),!a.disableExceptionTracking){f="onerror",b("
_"+f);var g=e[f];e[f]=function(a,b,d,e,h){var
i=g&&g(a,b,d,e,h);return!0!==i&&c["_"+f](a,b,d,e,h),i}}return c
  }({
      instrumentationKey:"<your instrumentation key>"
  });

window.appInsights=appInsights,appInsights.queue&&0===appInsights.queue.length&&appInsights.trackPag
eView();
</script>
```

# Application Insights for web pages - config

```
// Send telemetry immediately without batching.
// Remember to remove this when no longer required, as it can affect browser performance.
enableDebug: boolean,
// Don't log browser exceptions.
disableExceptionTracking: boolean,
// Don't log ajax calls.
disableAjaxTracking: boolean,
// Limit number of Ajax calls logged, to reduce traffic.
maxAjaxCallsPerView: 10, // default is 500
// Time page load up to execution of first trackPageView().
overridePageViewDuration: boolean,
// Set dynamically for an authenticated user.
accountId: string,
```

# Application Insights for console applications

- Install latest **Microsoft.ApplicationInsights** NuGet package
- Install latest **Microsoft.ApplicationInsights.DependencyCollector** NuGet package
- Set the instrumentation key
  - By using the static **TelemetryConfiguration.Active.InstrumentationKey** property
  - By using the **APPINSIGHTS_INSTRUMENTATIONKEY** environment variables
  - By using the **ApplicationInsights.config** file

# Application Insights for console applications - config

```
TelemetryConfiguration.Active.InstrumentationKey = " *your key* ";
var telemetryClient = new TelemetryClient();
telemetryClient.TrackTrace("Hello World!");
```

# Application Insights for console applications - files

```
// Reads ApplicationInsights.config file if present
TelemetryConfiguration config = TelemetryConfiguration.Active;


using System.IO;
TelemetryConfiguration configuration =
TelemetryConfiguration.CreateFromConfiguration(File.ReadAllText("C:\\ApplicationInsight
s.config"));
var telemetryClient = new TelemetryClient(configuration);
```

# Application Insights for console applications - code

```
var module = new DependencyTrackingTelemetryModule();

// Prevent Correlation Id to be sent to certain endpoints.
module.ExcludeComponentCorrelationHttpHeadersOnDomains.Add("core.windows.net");
// enable known dependency
module.IncludeDiagnosticSourceActivities.Add("Microsoft.Azure.EventHubs");

// initialize the module
module.Initialize(configuration);
```

# Application Insights for desktop apps

- Can be configured in a manner very similar to Application Insights for console apps
- Install latest **Microsoft.ApplicationInsights** NuGet package
- Install latest **Microsoft.ApplicationInsights.DependencyCollector** NuGet package
- Set the instrumentation key
  - By using the static **TelemetryConfiguration.Active.InstrumentationKey** property
  - By using the **ApplicationInsights.config** file

# Application Insights for desktop apps - code

```
public partial class Form1 : Form
{
    private TelemetryClient tc = new TelemetryClient();
    private void Form1_Load(object sender, EventArgs e)
    {
        // Alternative to setting ikey in config file:
        tc.InstrumentationKey = "key copied from portal";
        // Set session data:
        tc.Context.User.Id = Environment.UserName;
        tc.Context.Session.Id = Guid.NewGuid().ToString();
        tc.Context.Device.OperatingSystem = Environment.OSVersion.ToString();
        // Log a page view:
        tc.TrackPageView("Form1");
        ...
    }
}
```

# Application Insights platforms

- Official support – languages
  - .NET (C# & Microsoft Visual Basic)
  - Java
  - JavaScript
  - Node.js
- Unofficial support – languages
  - F#
  - PHP
  - Python
  - Ruby
- Official support – platforms/frameworks
  - ASP.NET (including ASP.NET Core)
  - Android
  - Angular
  - Azure (Azure App Service, Azure Cloud Services, Azure Functions)
  - Docker
  - Glimpse
  - iOS
  - Java 2 Platform Enterprise Edition (J2EE)
  - OS X
  - Spring
  - Universal Windows Platform (UWP)
  - Windows Communication Foundation (WCF)

# Other monitoring tools

- Cloudyn
  - Manages and optimizes multi-platform, hybrid cloud deployments to help enterprises fully realize their cloud potential. The software as a service (SaaS) solution delivers visibility into usage, performance, and cost. It provides insights and actionable recommendations for smart optimization and cloud governance.
- AppDynamics
  - Application Performance Management (APM), which enables application owners to rapidly troubleshoot performance bottlenecks and optimize the performance of their applications running in an Azure environment.
- Datadog
  - Monitoring service that gathers monitoring data from your containers within your Azure Container Service cluster. Datadog has a Docker Integration Dashboard, where you can view specific metrics within your containers. Metrics gathered from your containers are organized by CPU, memory, network, and I/O.

# Other monitoring tools (continued)

- The Elasticsearch, Logstash, and Kibana (ELK) stack
  - Combination of Elasticsearch, Logstash, and Kibana that provides an end-to-end stack that can be used to monitor and analyze logs in your cluster. The ELK stack is popular for monitoring container clusters, because the monitoring stack itself is open source and already containerized.
- New Relic Application Performance Management
  - Popular APM add-in for .NET applications. New Relic Application Performance Management can be used similarly in the Azure platform and has first-class support in role-based access control scenarios.

# Questions?

Microsoft

# Homework Assignment

# GET EXAM READY!

Microsoft Azure

# Open Mic