https://aka.ms/az204labs az400'e bir bak

https://enterprise.**klaxoon**.com/participate/session/HA6ZKQ6

https://www.microsoftazuresponsorships.com/balance

https://virtualbreak.org/

https://openhack.microsoft.com/

https://docs.microsoft.com/en-us/biztalk/core/connect-to-azure-api-management

About this Course: Course Outline

- Module 01: Creating Azure App Service Web Apps
- Module 02: Implement Azure Functions
- Module 03: Develop solutions that use blob storage
- Module 04: Develop solutions that use Cosmos DB storage
- Module 05: Implement IaaS solutions
- Module 06: Implement user authentication and authorization
- Module 07: Implement secure cloud solutions
- Module 08: Implement API Management
- Module 09: Develop event-based solutions
- Module 10: Develop message-based solutions
- Module 11: Instrument solutions to support monitoring and logging
- Module 12: Integrate caching and content delivery within solutions

Certification Areas (AZ-204)

Study Areas	Weights
Develop Azure compute solutions	25-30%
Develop for Azure storage	10-15%
Implement Azure security	15-20%
Monitor, troubleshoot, and optimize Azure solutions	10-15%
Connect to and consume Azure and third-party services	25-30%

- Percentages indicate the relative weight of each area on the exam
- The higher the percentage, the more questions you are likely to see in that area

,

Microsoft Certifications (Optional)

Certifications give you a professional edge by providing globally recognized industry endorsed evidence of skills mastery, demonstrating your abilities and willingness to embrace new technologies.

Exam AZ-900: Microsoft Azure Fundamentals

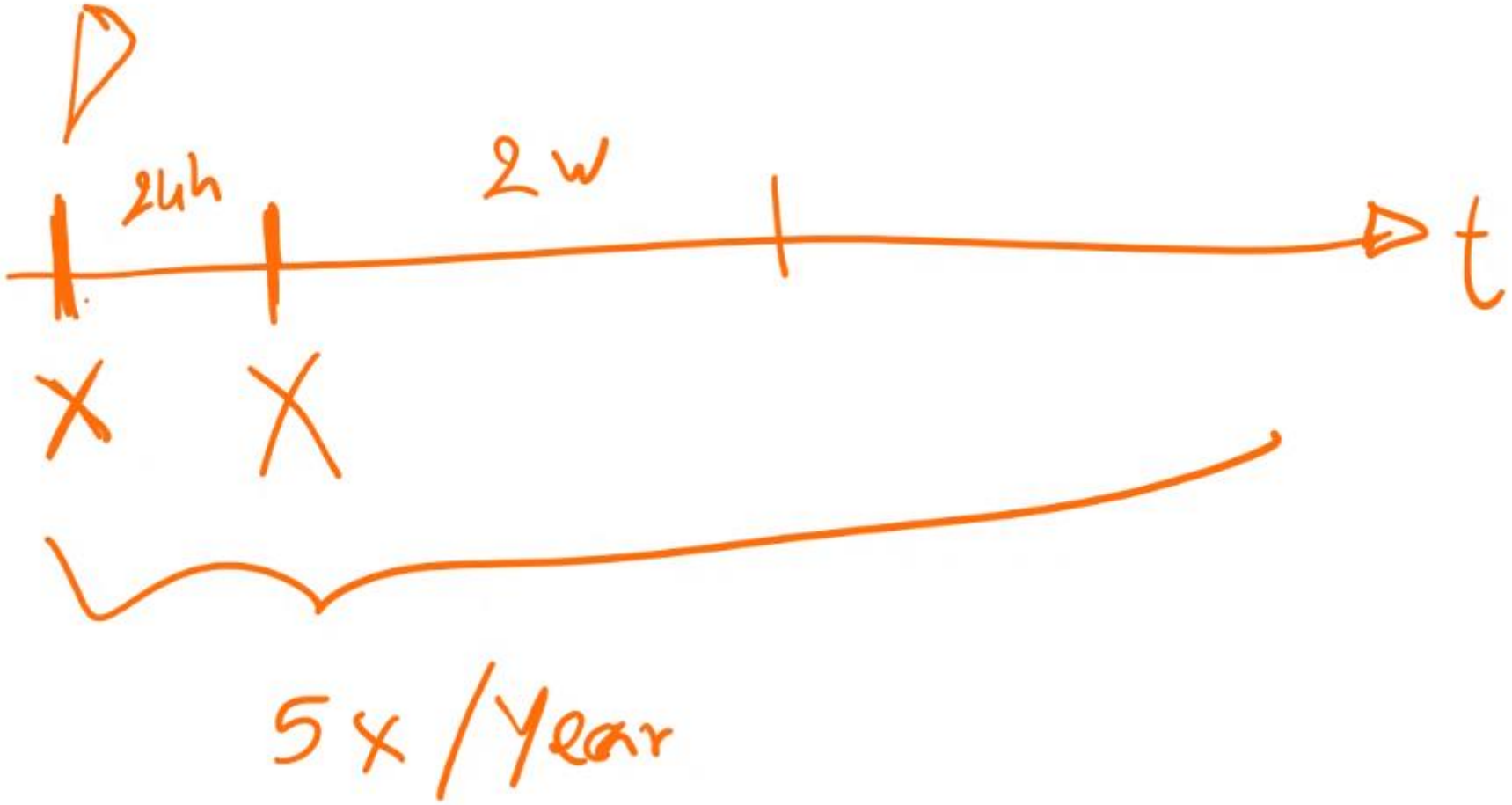
Designed for candidates looking to demonstrate foundational level knowledge of cloud services and how those services are provided.

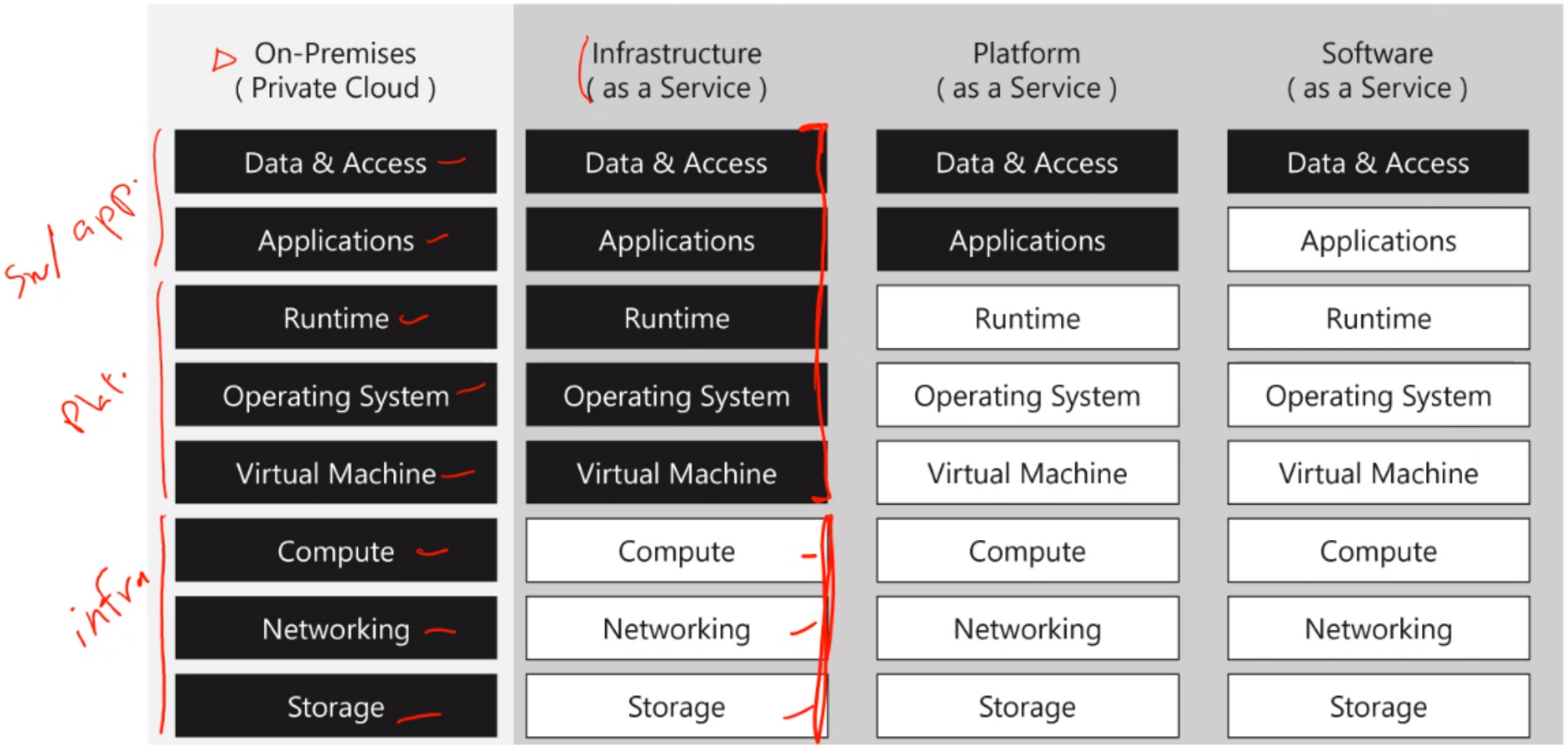
Microsoft Certified: Azure Developer Associate

Designed for Azure Developers who design, build, test, and maintain cloud solutions, such as applications and services.

Microsoft Certified: Azure Solutions Architect Expert

Designed for Azure Solutions Architects who create solutions for compute, network, storage, and security.


24h
2w
t
X
X
5x / Year


On-Premises
(Private Cloud)
Infrastructure
(as a Service)
Platform
(as a Service)
Software
(as a Service)
Data & Access
Applications
Runtime
Operating System
Virtual Machine
Compute
Networking
Storage
Data & Access
Applications
Runtime
Operating System
Virtual Machine
Compute
Networking
Storage
Data & Access
Applications
Runtime
Operating System
Virtual Machine
Compute
Networking
Storage
Data & Access
Applications
Runtime
Operating System
Virtual Machine
Compute
Networking
Storage

Key features of App Service Web Apps

- Multiple languages and frameworks:
  - First-class support for Microsoft ASP.NET, Java, Ruby, Node.js, PHP, or Python
- DevOps optimization:
  - Continuous integration and deployment with Visual Studio Team Services, GitHub, Bitbucket, Docker Hub, or Azure Container Registry
- Global scale with high availability:
  - Scale up or out manually or automatically. Host anywhere in the Microsoft global datacenter infrastructure
- Connections to SaaS platforms and on-premises data:
  - More than 50 connectors for enterprise systems (such as SAP), SaaS services (such as Salesforce), and internet services (such as Facebook)

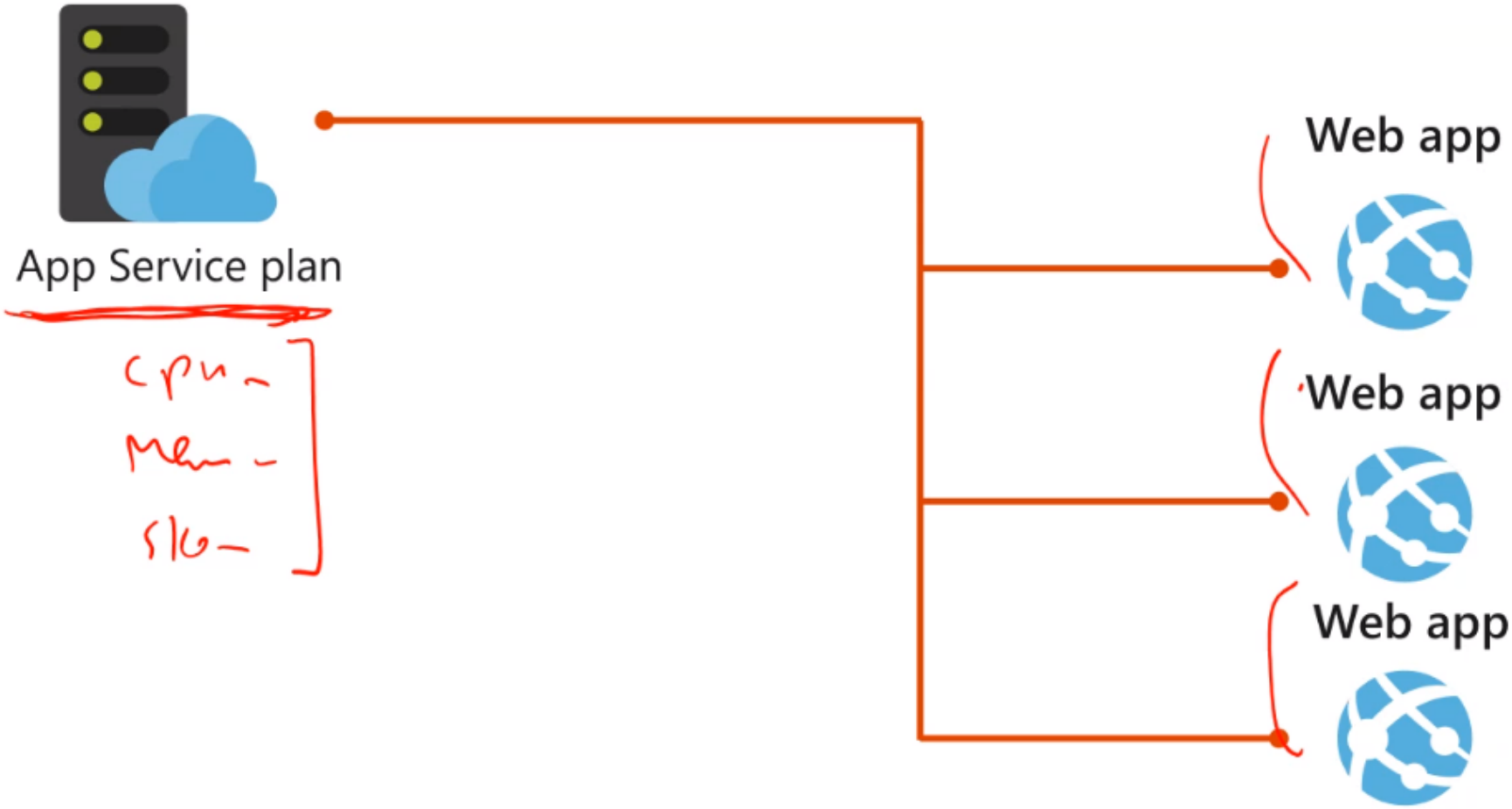
https://infrastructuremap.microsoft.com/explore

https://azurecharts.com/

Key features of App Service Web Apps (cont.)

- Security and compliance:
  - App Service is ISO, SOC, and PCI compliant
- Application templates:
  - Templates in the Azure Marketplace, such as WordPress, Joomla, and Drupal
- Visual Studio integration:
  - Streamline the work of creating, deploying, and debugging
- API and mobile features:
  - Turn-key Cross-Origin Resource Sharing (CORS) support for RESTful API scenarios, and enables authentication, offline data sync, push notifications, and more
- Serverless code:
  - Run code on-demand without having to explicitly provision or manage infrastructure

App Service plans (continued)



Azure App Service yaratırken github ile bağlantı yapacak mısın diye soruyor. İsteğe bağlı bir entegrasyon var.

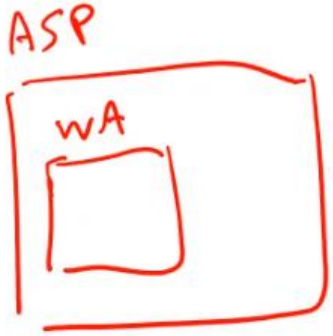
ASE

App Service environments (ASEs)

- App Service variant that provides a fully isolated and dedicated environment for securely running App Service apps at high scale
- Ideal for application workloads that require:
  - Very high scale, higher than typical App Service capacity
  - Network isolation and secure network access
  - High memory utilization
- Single or Multi-region
- Deployed to a virtual network
- An ASE is dedicated exclusively to a single subscription:
  - Max 100 instances

Creating a web app with Azure command-line interface (CLI) (continued)

```
# generate a unique name and store as a shell variable
webappname=mywebapp$RANDOM

# create a resource group
az group create --location westeurope --name myResourceGroup

# create an App Service plan
az appservice plan create --name $webappname --resource-group myResourceGroup --sku
FREE

# create a Web App
az webapp create --name $webappname
    --resource-group myResourceGroup
    --plan $webappname
```



SKU

https://docs.microsoft.com/en-us/azure/app-service/overview-hosting-plans

Deploying a web app with Azure CLI

```
# store a repository url as a shell variable
gitrepo=https://github.com/Azure-Samples/php-docs-hello-world

# deploy code from a Git repository
az webapp deployment source config --name $webappname --resource-group myResourceGroup
--repo-url $gitrepo --branch master --manual-integration

# print out the FQDN for the Web App
echo http://$webappname.azurewebsites.net
```

Creating a Web App with Azure PowerShell

Command	Notes
`New-AzResourceGroup`	Creates a resource group in which all resources are stored
`New-AzAppServicePlan`	Creates an App Service plan
`New-AzWebApp`	Creates an Azure Web App
`Set-AzResource`	Modifies a resource in a resource group

App Service on Linux

Why Linux?

- Many application stacks are optimized for Linux:
  - Ruby/Rails, PHP, Node, and others
  - Often, better tools are available on Linux for these stacks
- New and upcoming frameworks are built for Linux first and then Windows
- Portability of Docker containers
- Linux is at the forefront of innovations in nano and microservice architecture

https://hub.docker.com/

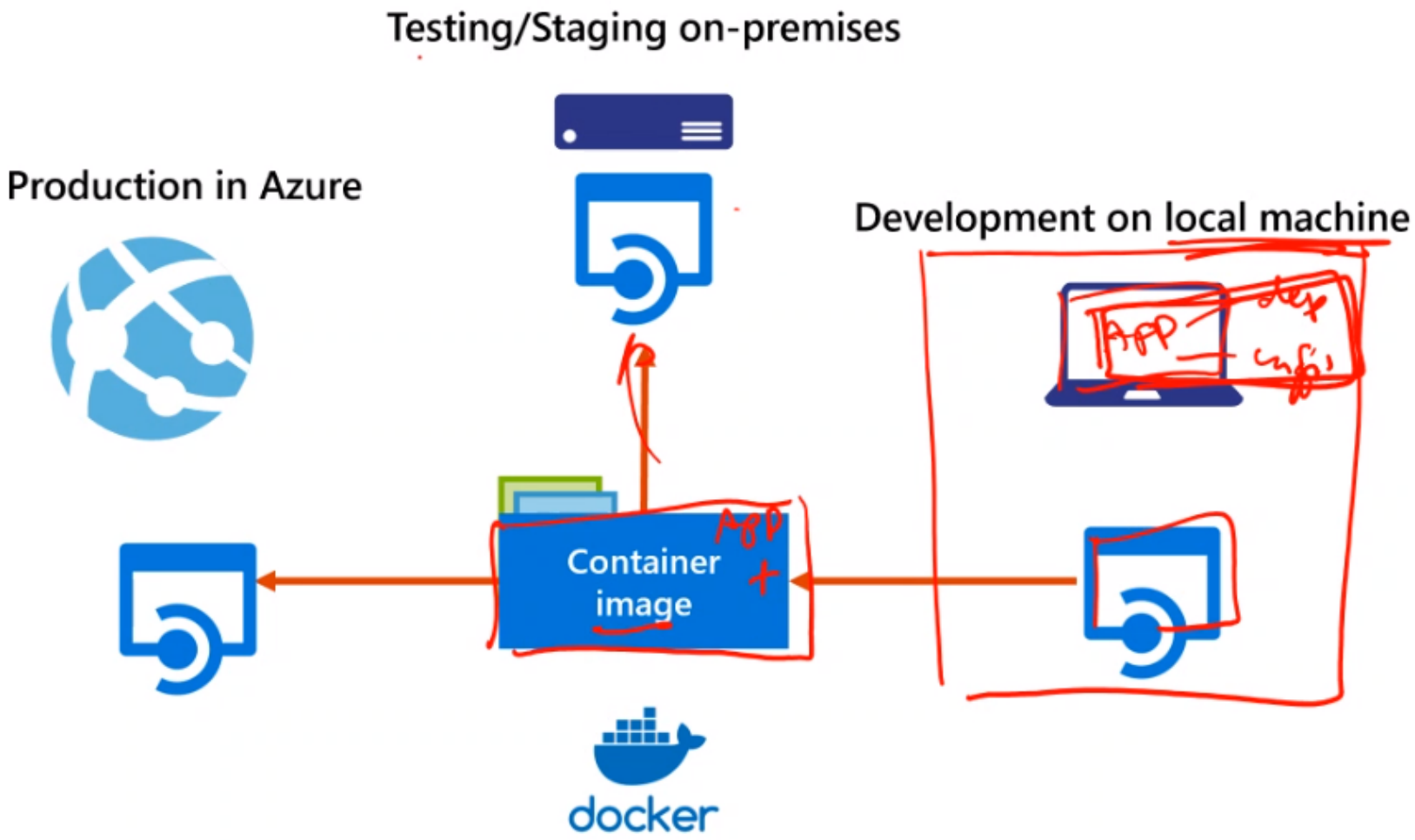
Linux ile uyumlu hazır docker imaj adetleri daha fazla

2001 Container dediğimiz şey zaten bir Linux teknolojisiydi.

2013 Docker container icat edildi. ~4MB

2016/17/18 Win. Containers ~100MB

Docker in App Service on Linux



Web apps for Linux containers

Deploy applications and solutions that are containerized directly to App Service Web Apps:

- Simplifies deployment
- Matches the already popular container workflow using:
  - CI/CD with Docker Hub, Azure Container Registry, or GitHub
- Compatible with existing App Service features:
  - Auto-scale, Deployment Slots, and others

Web apps for Linux containers (continued)

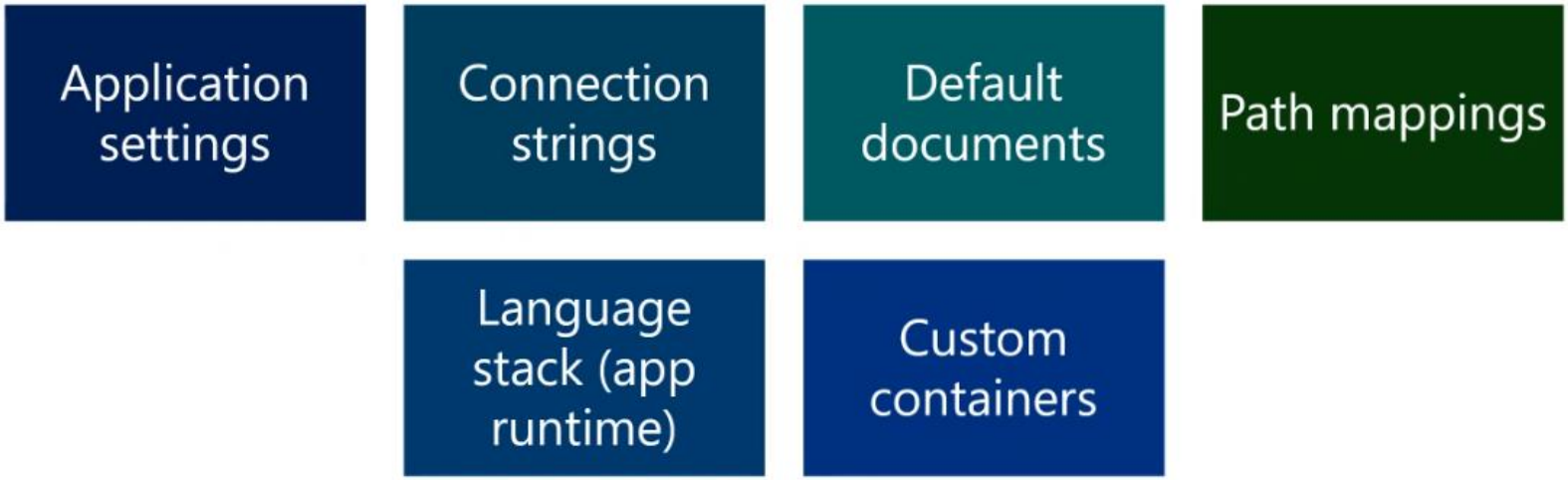
Containers can be sourced from your existing registries:

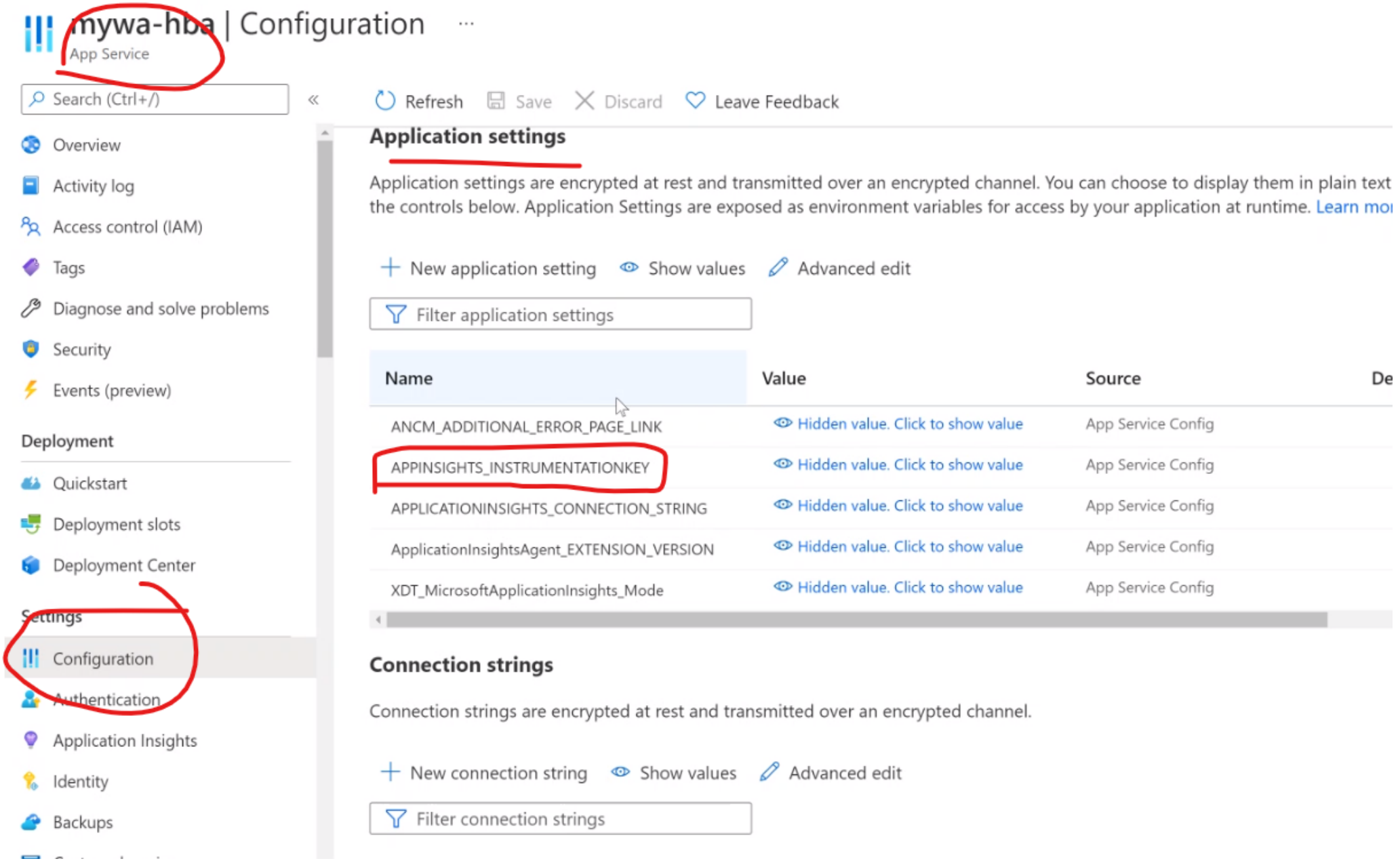
- Docker Hub:
  - Deploy images already shared on Docker Hub
  - Deploy the most popular official images
  - Private images are available on Docker Hub
- Azure Container Registry:
  - Managed service for hosting Docker images
  - Can deploy to Docker Swarm, Kubernetes, or Web App for Containers

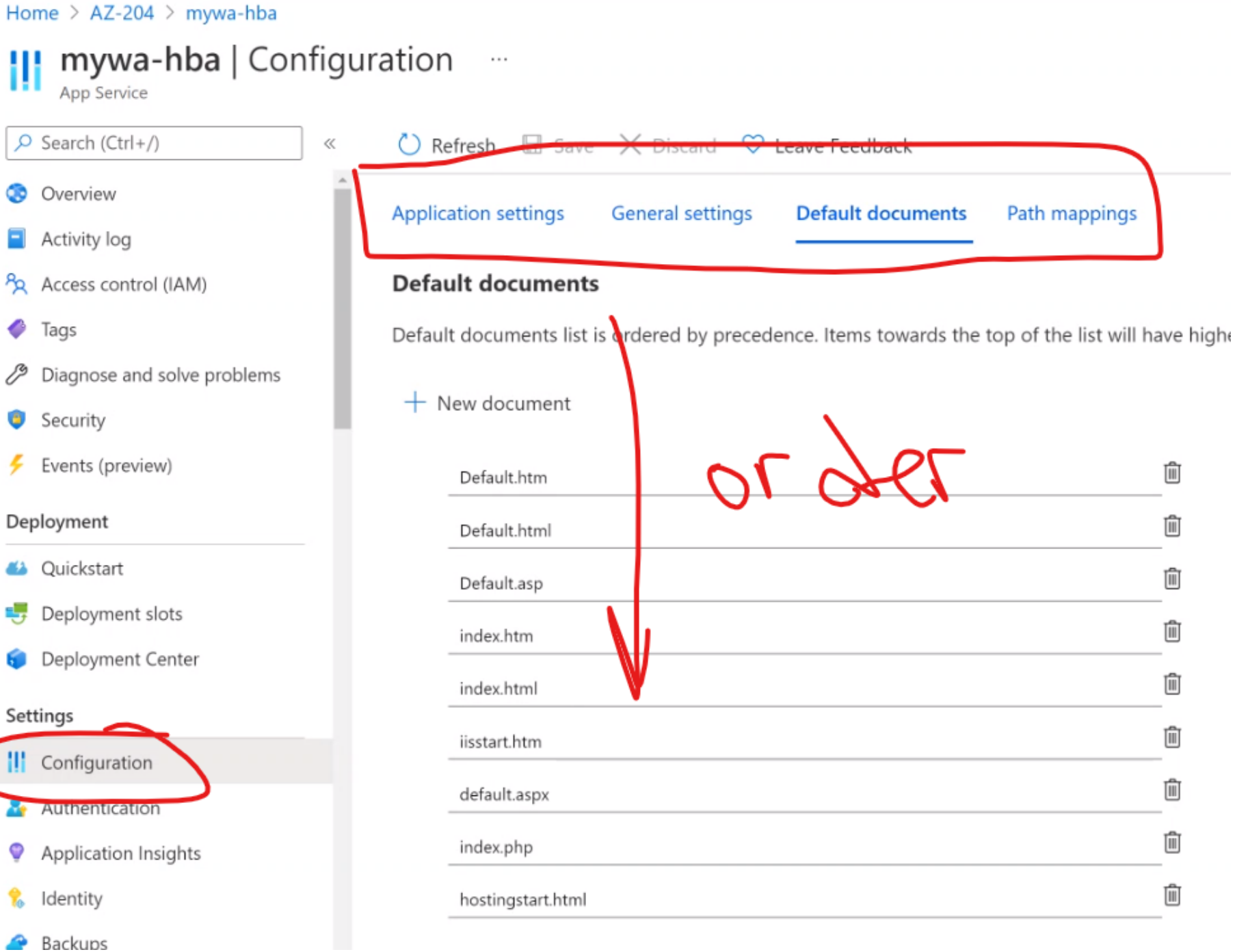
Lesson 03: Configuring and Monitoring App Service apps

App Service settings

- Overrides settings in Web.config or appsettings.json
- Hidden by default in Azure portal
- You can configure:




Home > AZ-204 > mywa-hba
mywa-hba | Configuration
App Service
Search (Ctrl+/)
Refresh
Save
Discard
Leave Feedback
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Security
Events (preview)
Deployment
Quickstart
Deployment slots
Deployment Center
Settings
Configuration
Authentication
Application Insights
Identity
Backups
Application settings
Application settings are encrypted at rest and transmitted over an encrypted channel. You can choose to display them in plain text
the controls below. Application Settings are exposed as environment variables for access by your application at runtime. Learn mo
New application setting
Show values
Advanced edit
Filter application settings
Name
Value
Source
ANCM_ADDITIONAL_ERROR_PAGE_LINK
Hidden value. Click to show value
App Service Config
APPINSIGHTS_INSTRUMENTATIONKEY
Hidden value. Click to show value
App Service Config
APPLICATIONINSIGHTS_CONNECTION_STRING
Hidden value. Click to show value
App Service Config
ApplicationInsightsAgent_EXTENSION_VERSION
Hidden value. Click to show value
App Service Config
XDT_MicrosoftApplicationInsights_Mode
Hidden value. Click to show value
App Service Config
Connection strings
Connection strings are encrypted at rest and transmitted over an encrypted channel.
New connection string
Show values
Advanced edit
Filter connection strings


Home > AZ-204 > mywa-hba
mywa-hba | Configuration
App Service
Search (Ctrl+/)
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Security
Events (preview)
Deployment
Quickstart
Deployment slots
Deployment Center
Settings
Configuration
Authentication
Application Insights
Identity
Backups
Refresh
Save
Discard
Leave Feedback
Application settings
General settings
Default documents
Path mappings
Default documents
Default documents list is ordered by precedence. Items towards the top of the list will have highe
New document
order
Default.htm
Default.html
Default.asp
index.htm
index.html
iisstart.htm
default.aspx
index.php
hostingstart.html

Updating app runtimes

```
az webapp config set --linux-fx-version "DOTNETCORE|3.1" --resource-group <groupname> -
-name <appname>

az webapp config set --php-version 7.0 --resource-group <groupname> --name <appname>

az webapp config set --python-version 3.4 --resource-group <groupname> --name <appname>

az webapp config set --resource-group <groupname> --name <appname> --java-version 1.8 -
-java-container Tomcat --java-container-version 9.0
```

Update .NET version

Update PHP version

Update Python version

Update Java version

Herhangi bir app settings'i de bu şekilde update edebilirsin.

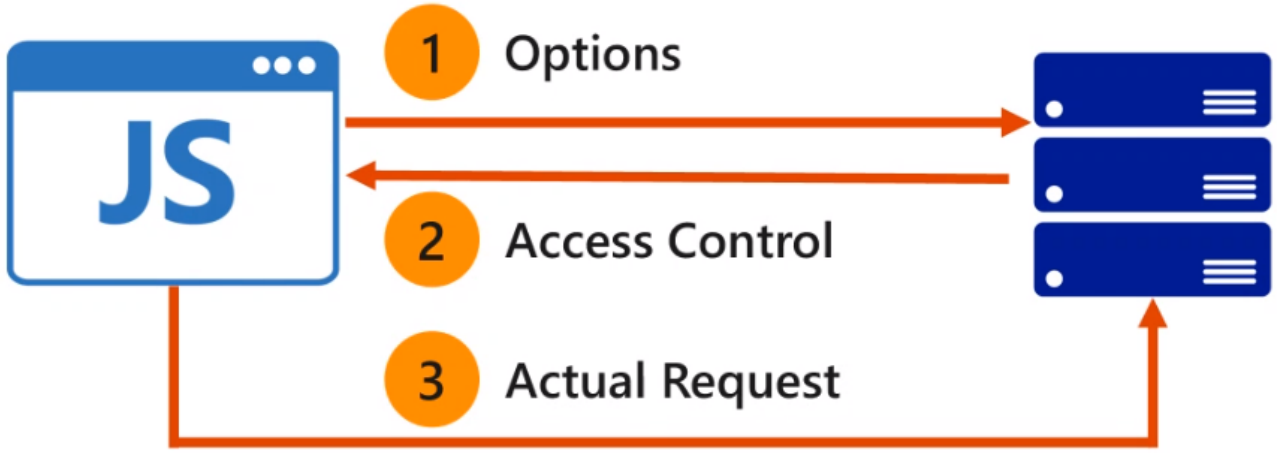
Updating app runtimes (Node.js)

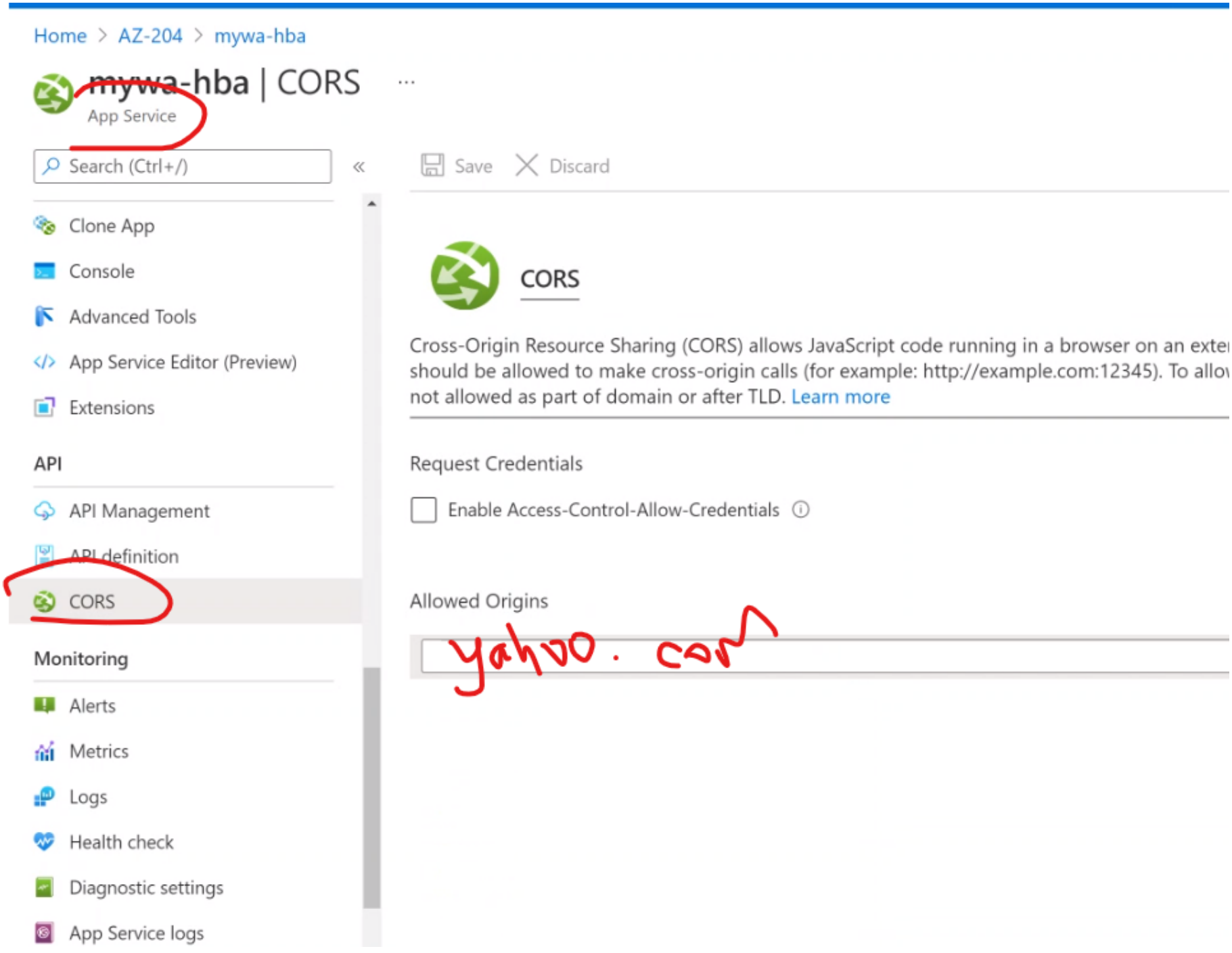
```
az webapp config appsettings set `
    --resource-group <groupname> `
    --name <appname> `
    --settings WEBSITE_NODE_DEFAULT_VERSION=8.9.3
```

Croos origin resource sharing : default disabled, not allowed

CORS

- Mechanism for servers to indicate that they support cross-site requests
  - Servers can specify:
    - Allowed HTTP verbs
    - Allowed origins
    - Allowed headers
- Directly supported by API Apps




Home > AZ-204 > mywa-hba
mywa-hba | CORS
App Service
Search (Ctrl+/)
Save
Discard
Clone App
Console
Advanced Tools
App Service Editor (Preview)
Extensions
API
API Management
API definition
CORS
Monitoring
Alerts
Metrics
Logs
Health check
Diagnostic settings
App Service logs
CORS
Cross-Origin Resource Sharing (CORS) allows JavaScript code running in a browser on an exte
should be allowed to make cross-origin calls (for example: http://example.com:12345). To allo
not allowed as part of domain or after TLD. Learn more
Request Credentials
Enable Access-Control-Allow-Credentials
Allowed Origins
yahoo.com

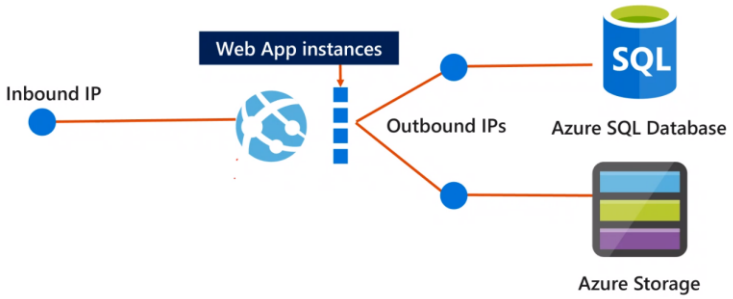
OS and runtime patching

- OS and application stack are managed by Azure on your behalf
- Monthly OS patching:
  - Physical servers
  - Guest virtual machines
- Stable versions of application runtimes are periodically added to App Services:
  - Some are installed side by side, while others replace existing versions
  - You can manually migrate from one application runtime to another

Inbound and outbound IP addresses

- Each app has a single inbound IP address:
  - Regardless of scale-out quantity
- Inbound IP address can change:
  - Delete an app and re-create it in a new resource group
  - Delete the last app in a resource group + region combination and re-create it
  - Delete an existing SSL binding
- You can opt to use a state inbound IP
- Each app has a set number of outbound IP addresses:
  - The set and quantity changes as you scale your app between tiers

Outbound IP addresses



App servisi silersen yeniden yaratırsan inbound adres değiştir.

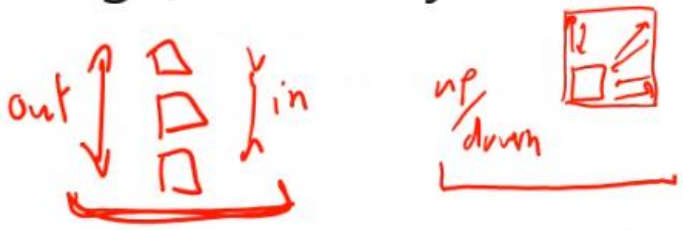
Find outbound IP addresses

```
# find Outbound IP address
az webapp show --resource-group <group_name> --name <app_name> --query
outboundIpAddresses --output tsv

# find all possible IP addresses (regardless of tier)
az webapp show --resource-group <group_name> --name <app_name> --query
possibleOutboundIpAddresses --output tsv
```

Lesson 04: Scaling App Service apps

Autoscale

- A primary advantage of the cloud is **elastic** scaling (the ability to use as much capacity as you need):
  - Scaling out as load increases
  - Scaling in when the extra capacity is not needed



- Many Microsoft Azure services provide the capability to scale both **manually** and **automatically**
- Autoscale refers to the capability of many of these services to **monitor** the application instances and **automatically scale** appropriately to handle the current usage of the application:
  - Using autoscale, your cloud service can scale out and in to exactly match the amount of instances needed for your specific computing pattern

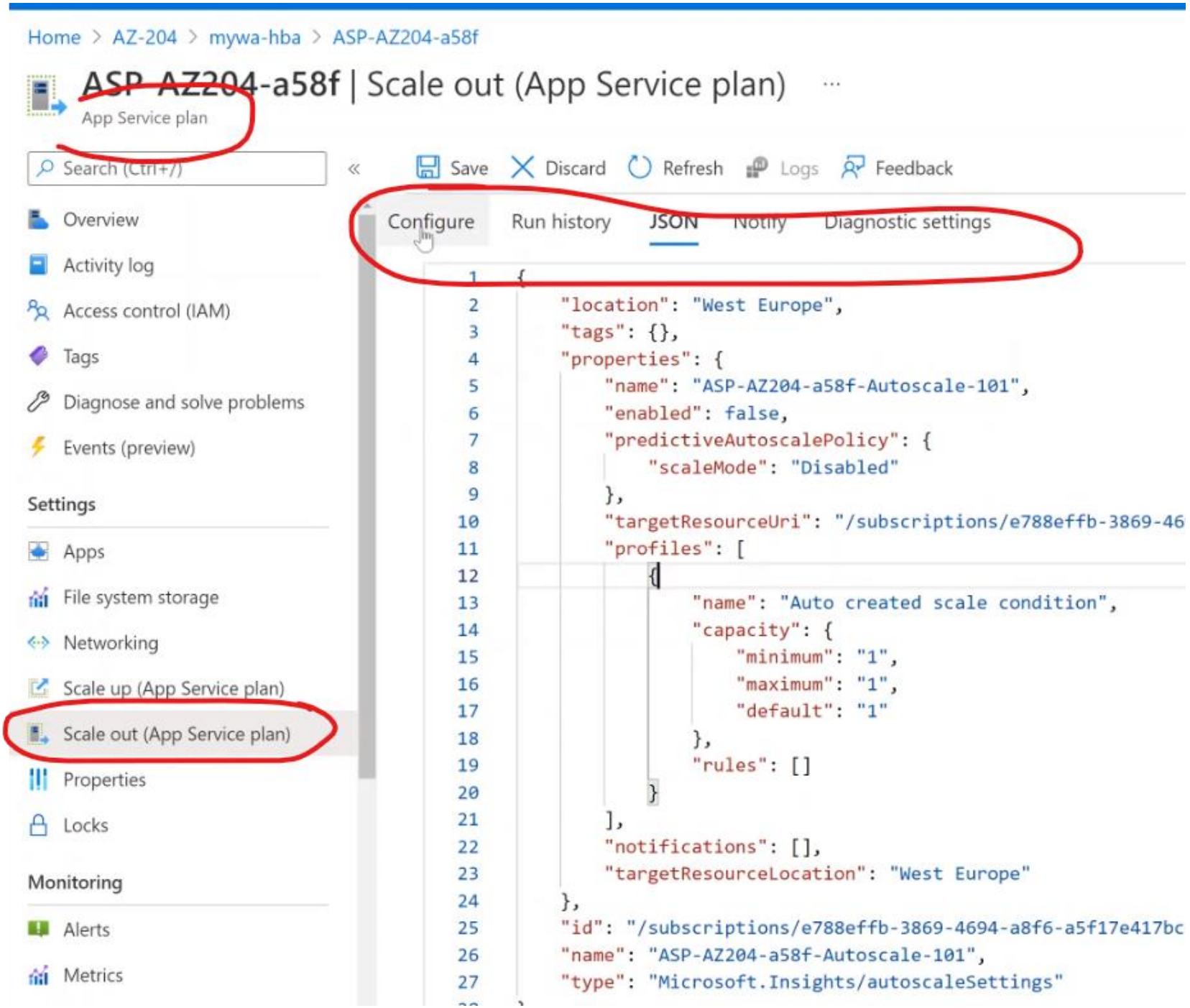
Autoscale metrics

Metric	Metric identifier	Description
CPU	CpuPercentage	The average amount of CPU time used across all instances of the plan
Memory	MemoryPercentage	The average amount of memory used across all instances of the plan
Data in	BytesReceived	The average incoming bandwidth used across all instances of the plan
Data out	BytesSent	The average outgoing bandwidth used across all instances of the plan
HTTP queue	HttpQueueLength	The average number of both read and write requests that were queued on storage. A high disk queue length is an indication of an application that might be slowing down due to excessive disk I/O.
Disk queue	DiskQueueLength	The average number of HTTP requests that had to sit in the queue before being fulfilled. A high or increasing HTTP queue length is a symptom of a plan under a heavy load.

Autoscale patterns

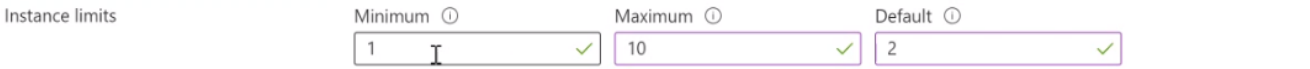
80%

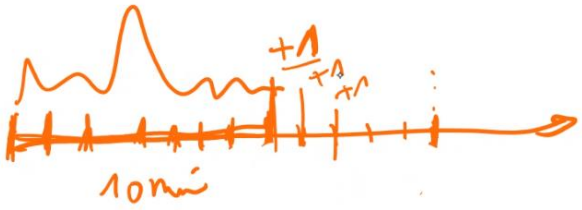
- Scale based on CPU
- Scale differently on weekdays vs. weekends
- Scale differently during holidays
- Scale based on custom metric

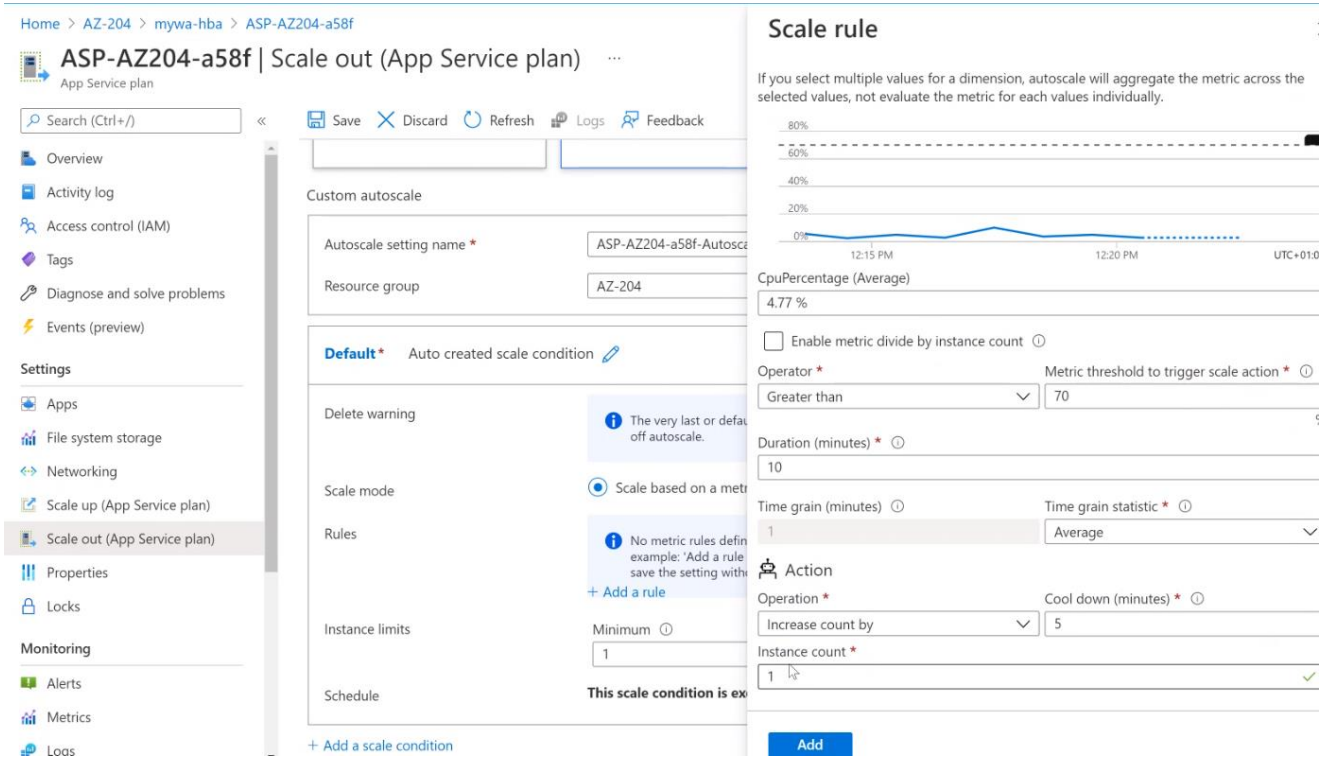

Home > AZ-204 > mywa-hba > ASP-AZ204-a58f
ASP-AZ204-a58f | Scale out (App Service plan)
App Service plan
Search (Ctrl+/)
Save
Discard
Refresh
Logs
Feedback
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Events (preview)
Settings
Apps
File system storage
Networking
Scale up (App Service plan)
Scale out (App Service plan)
Properties
Locks
Monitoring
Alerts
Metrics
Configure
Run history
JSON
Notify
Diagnostic settings
{
"location": "West Europe",
"tags": {},
"properties": {
"name": "ASP-AZ204-a58f-Autoscale-101",
"enabled": false,
"predictiveAutoscalePolicy": {
"scaleMode": "Disabled"
},
"targetResourceUri": "/subscriptions/e788effb-3869-46
"profiles": [
{
"name": "Auto created scale condition",
"capacity": {
"minimum": "1",
"maximum": "1",
"default": "1"
},
"rules": []
}
],
"notifications": [],
"targetResourceLocation": "West Europe"
},
"id": "/subscriptions/e788effb-3869-4694-a8f6-a5f17e417bc
"name": "ASP-AZ204-a58f-Autoscale-101",
"type": "Microsoft.Insights/autoscaleSettings"

Cooling down süresini girmezsen ateşi düşmeden istatistik almaya başlar ve yeni instance eklemeye devam eder.

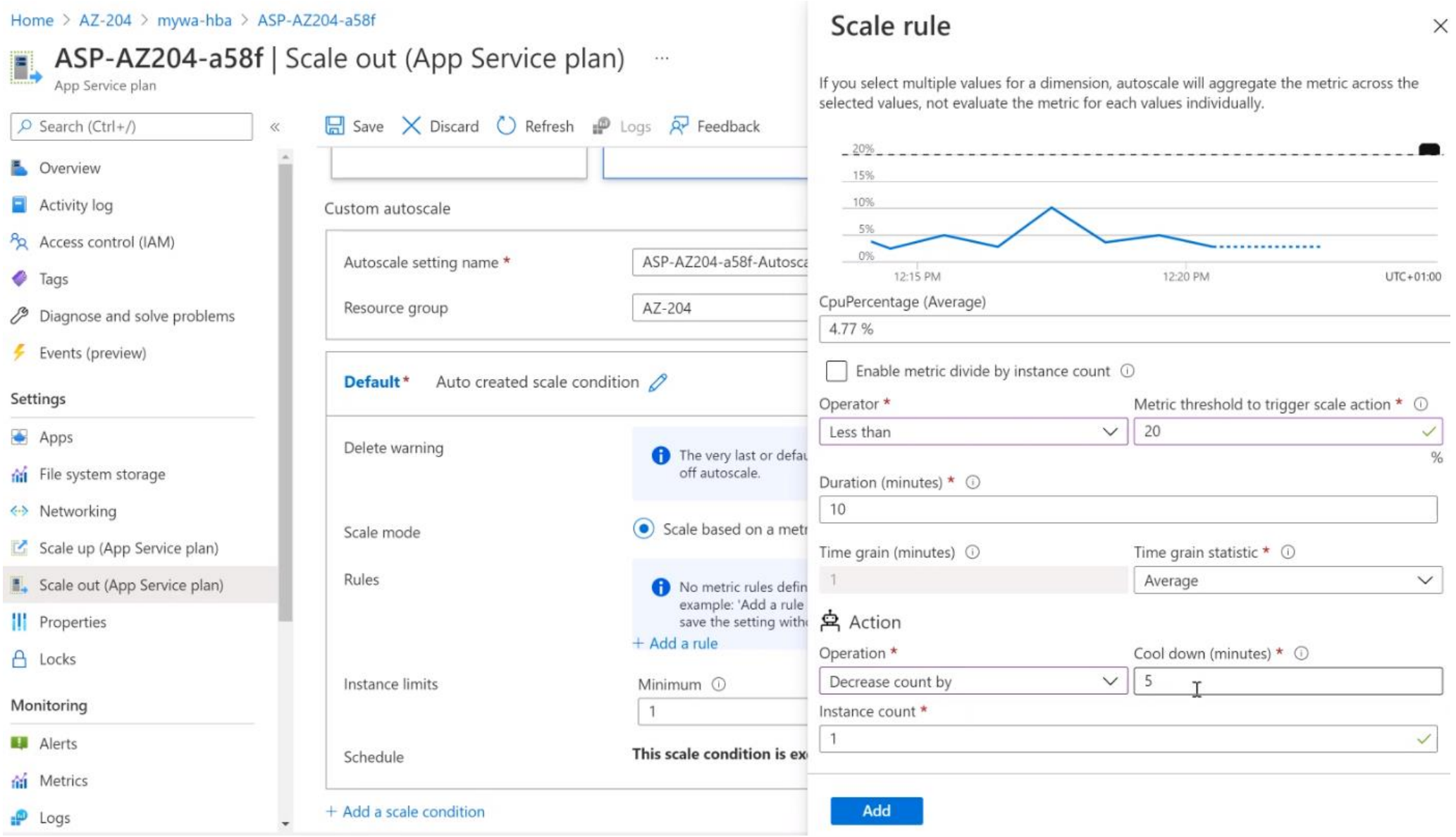
Scale-put için max. İnstance sayısını koyabilirsin.







Scale out tanımladıysan, scale in ‘i de tanımlaman lazım. Nasıl eksilteceğini bilmesi gerekir.



Autoscale setting schema

```
{
    "id": "...", "name": "demoSetting", "type": "Microsoft.Insights/autoscaleSettings",
    "location": "East US",
    "properties": {
        "enabled": true, "targetResourceUri": "...",
        "profiles": [
            {
                "name": "mainProfile",
                "capacity": {
                    "minimum": "1", "maximum": "4", "default": "1"
                },
                "rules": [ ... ]
            }
        ]
    }
}
```

Autoscale setting schema – scale-out rule

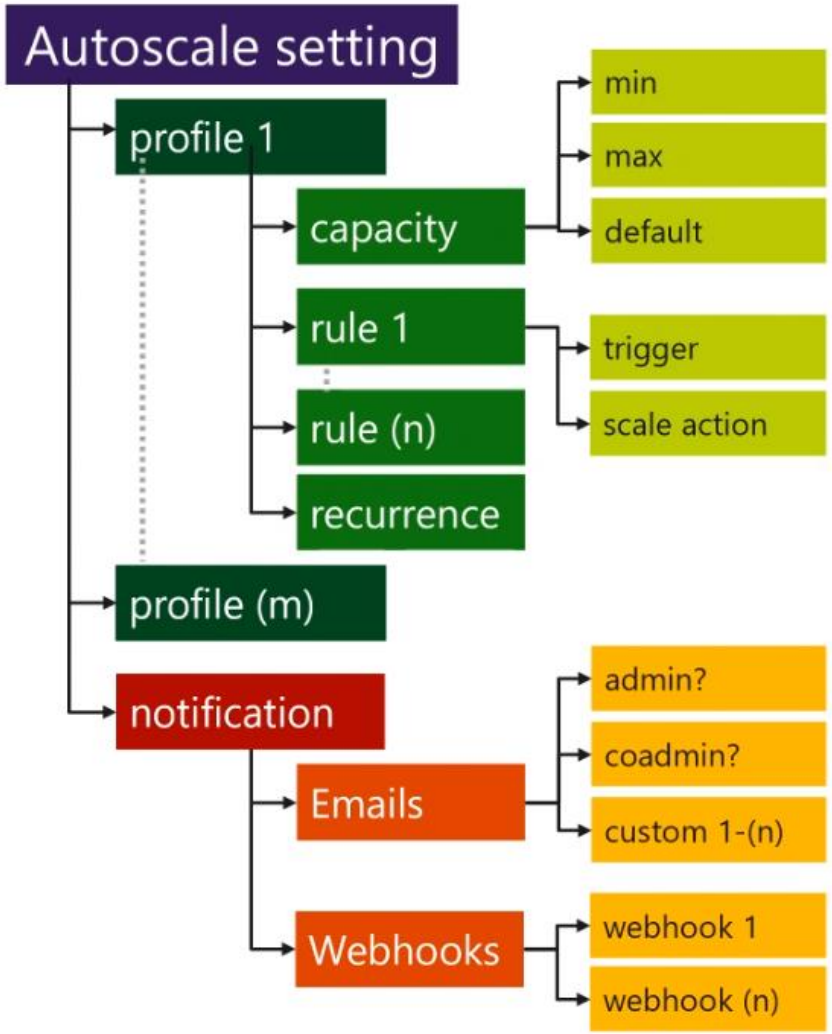
```
{
    "metricTrigger": {
        "metricName": "Percentage CPU", "metricResourceUri": "...",
        "timeGrain": "PT1M",
        "statistic": "Average",
        "timeWindow": "PT10M",
        "timeAggregation": "Average",
        "operator": "GreaterThan",
        "threshold": 85
    },
    "scaleAction": {
        "direction": "Increase", "type": "ChangeCount",
        "value": "1",
        "cooldown": "PT5M"
    }
}
```

Autoscale setting schema – scale-in rule

```
{
    "metricTrigger": {
        "metricName": "Percentage CPU", "metricResourceUri": "...",
        "timeGrain": "PT1M",
        "statistic": "Average",
        "timeWindow": "PT10M",
        "timeAggregation": "Average",
        "operator": "LessThan",
        "threshold": 60
    },
    "scaleAction": {
        "direction": "Decrease", "type": "ChangeCount",
        "value": "1",
        "cooldown": "PT5M"
    }
}
```

Autoscale hierarchy

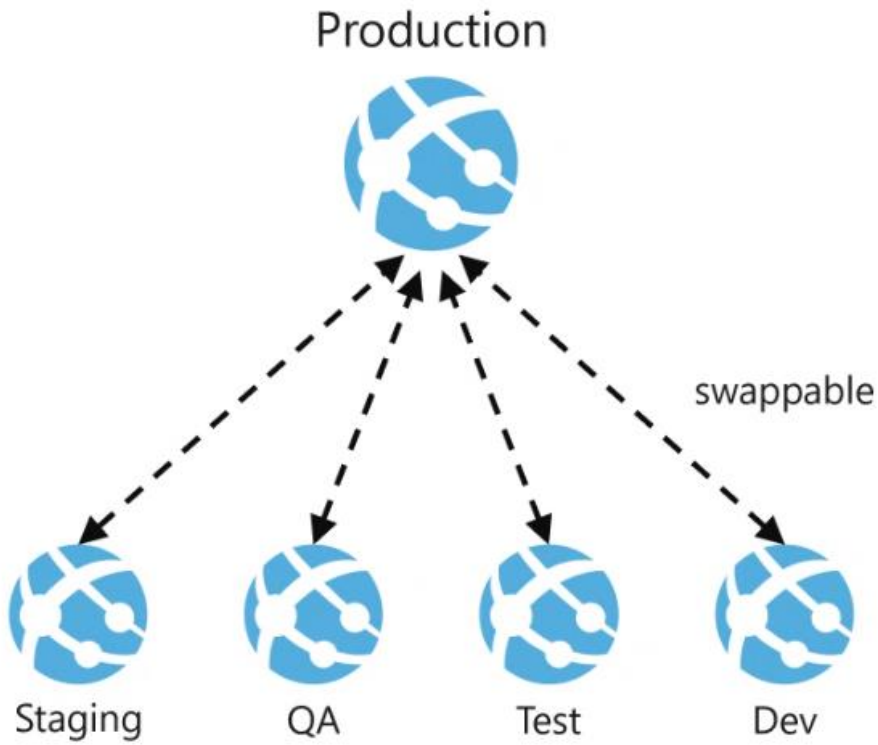
- One autoscale setting
- Settings have one or more profiles
- Profiles have one or more rules:
  - Profiles can also have recurrences and capacity settings
- Notifications can be directly associated with an autoscale setting



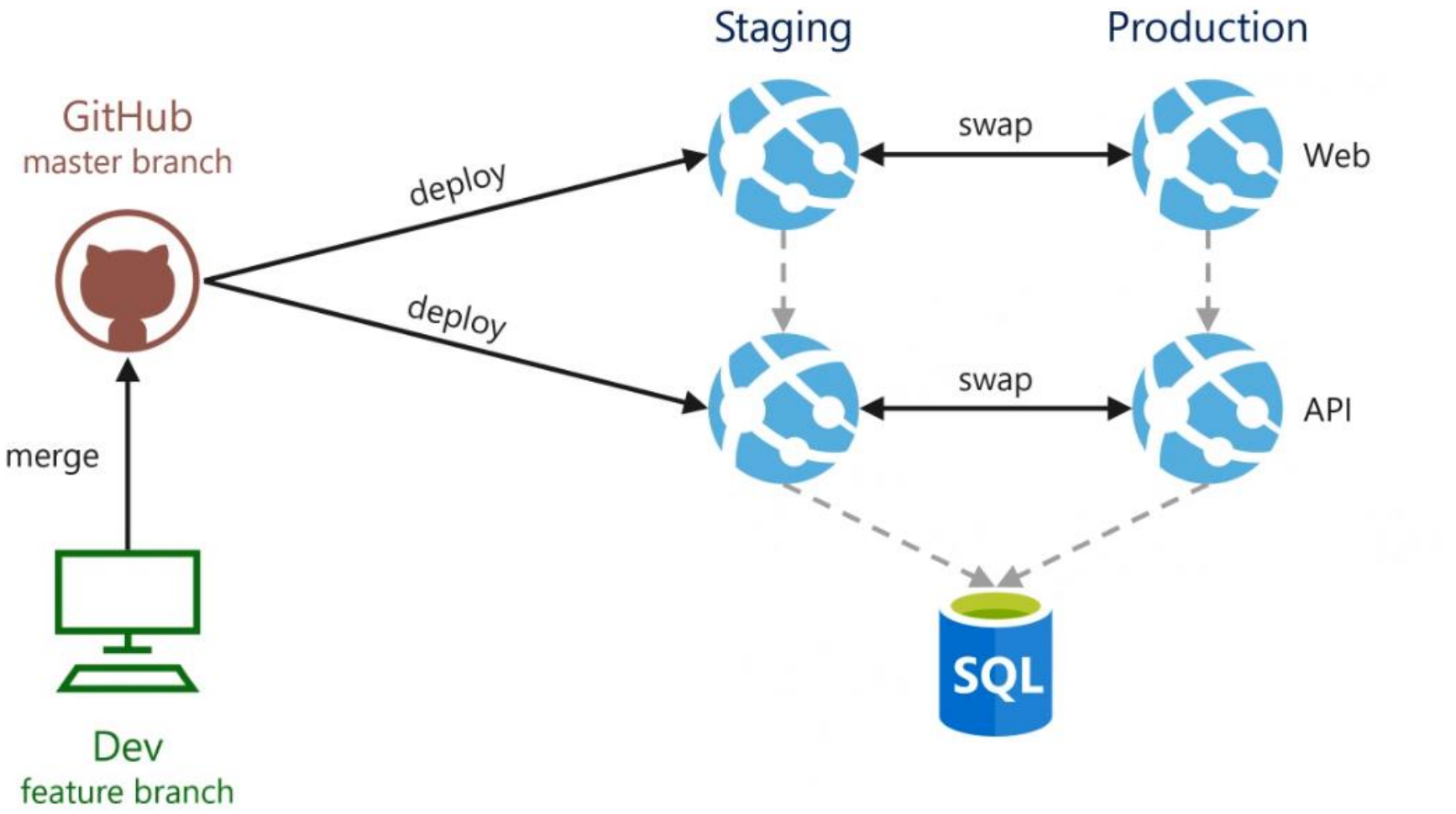
Lesson 05: Azure App Service staging environments

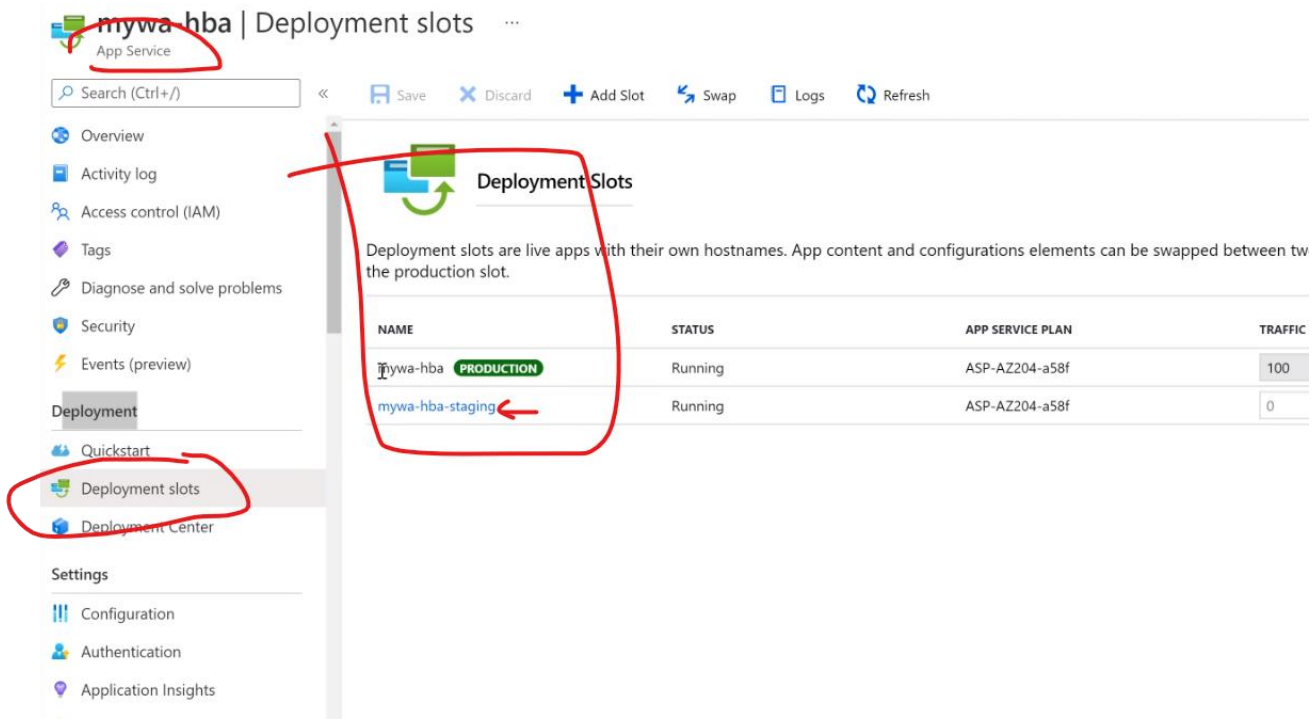
Deployment slots

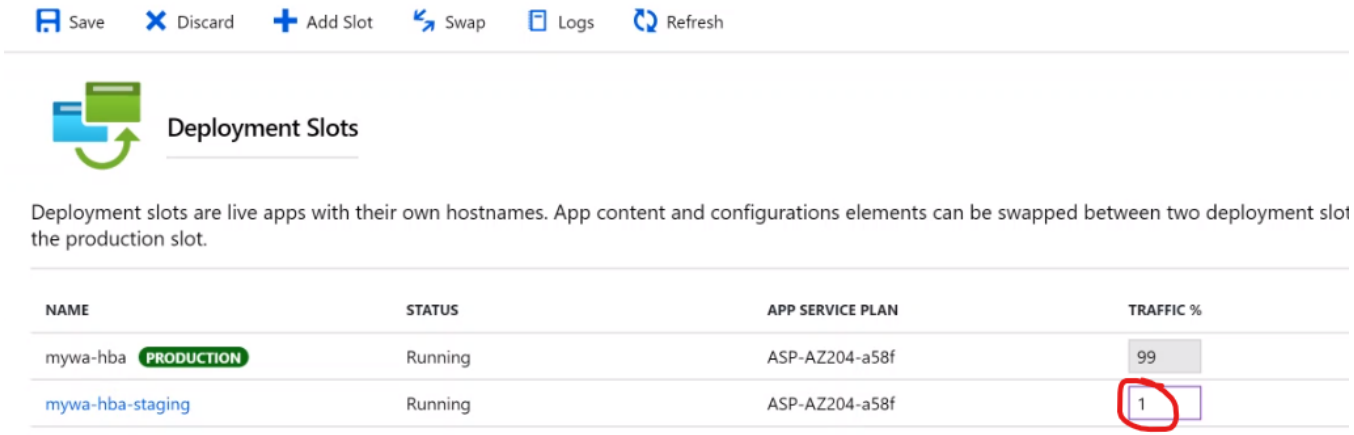
- Live apps with their own:
  - Host names
  - Content
  - Configuration
- Can be swapped between each other. For example:
  - Staging ↔ Production
  - Production ↔ Staging
  - Dev ↔ Test
  - Test ↔ QA
  - QA ↔ Staging

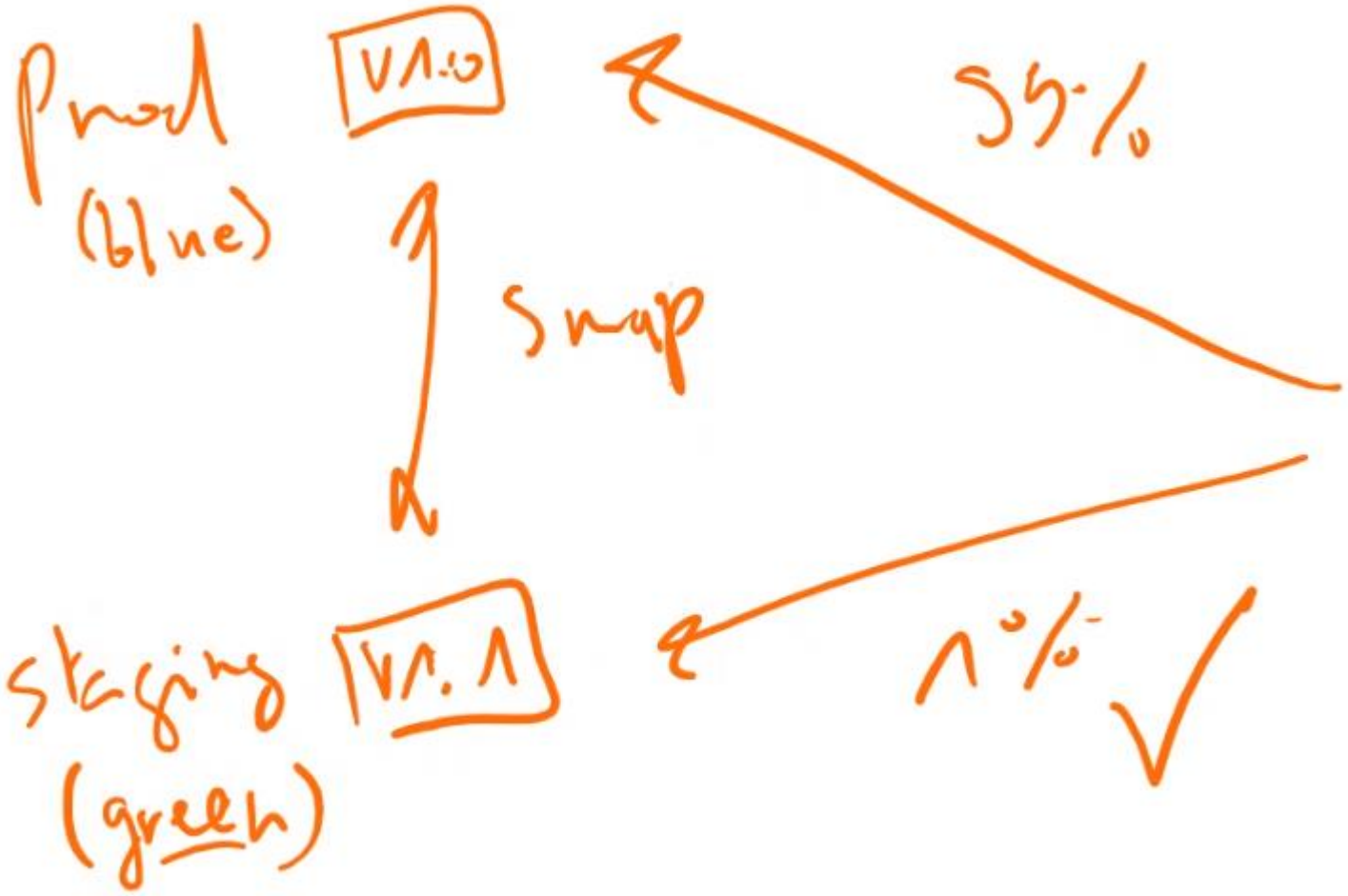


Modern deployment workflow




mywa-hba | Deployment slots
App Service
Search (Ctrl+/)
Save
Discard
Add Slot
Swap
Logs
Refresh
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Security
Events (preview)
Deployment
Quickstart
Deployment slots
Deployment Center
Settings
Configuration
Authentication
Application Insights
Deployment Slots
Deployment slots are live apps with their own hostnames. App content and configurations elements can be swapped between tw
the production slot.
NAME
STATUS
APP SERVICE PLAN
TRAFFIC
mywa-hba
PRODUCTION
Running
ASP-AZ204-a58f
100
mywa-hba-staging
Running
ASP-AZ204-a58f
0


Save
Discard
Add Slot
Swap
Logs
Refresh
Deployment Slots
Deployment slots are live apps with their own hostnames. App content and configurations elements can be swapped between two deployment slot
the production slot.
NAME
STATUS
APP SERVICE PLAN
TRAFFIC %
mywa-hba
PRODUCTION
Running
ASP-AZ204-a58f
99
mywa-hba-staging
Running
ASP-AZ204-a58f
1


prod
(blue)
V1.0
95%
swap
staging
(green)
V1.1
5%

Automate slot management - Azure CLI

```
az webapp deployment slot create --name <appname> --resource-group <groupname> --
slot staging
```

Create new slot

Swap slots

```
az webapp deployment slot swap --resource-group <groupname> --name <appname>
--slot staging --target-slot production
```

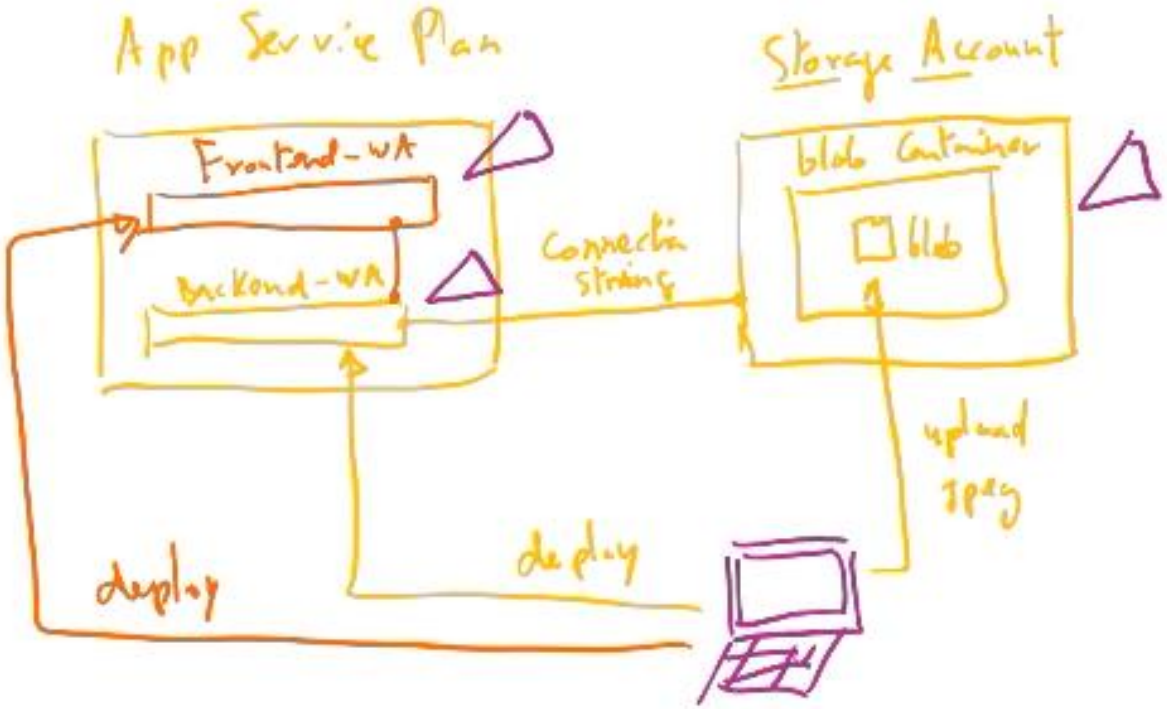
Source

Target

LABS

[Microsoft Azure Sponsorships | Balance]

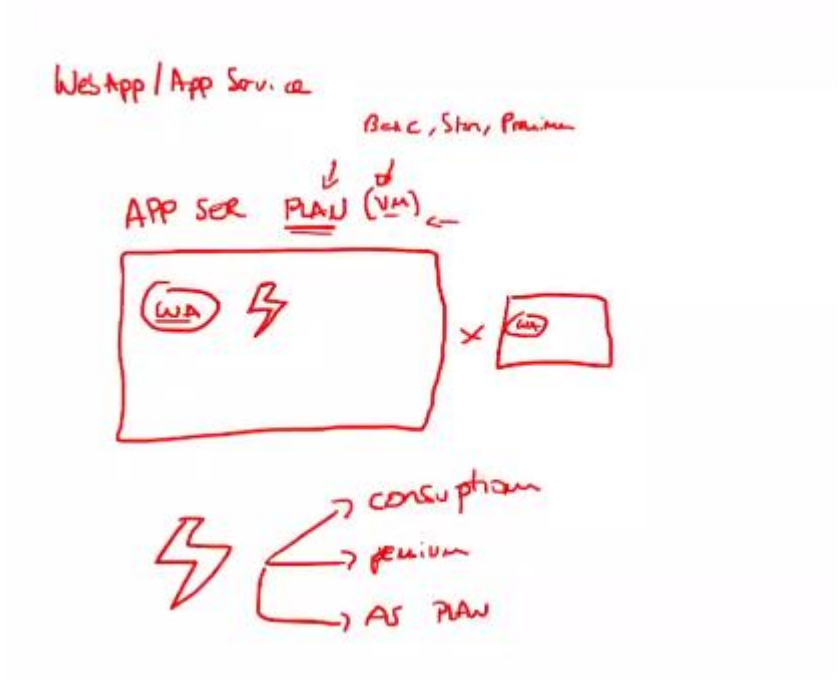
Lab01 EX1 EX2



Azure Functions

- Solution for running **small pieces of code, or "functions," in the cloud**:
  - Write only code that is relevant to business logic
  - Removes the necessity to write "plumbing" code to connect or host application components
- Build on open-source WebJobs code
- Supports a wide variety of programming languages, for instance:

- Even supports scripting languages, such as:



Azure function içimde birden fazla function tabımnlayabiliryor. Sanki mikrı service gibi

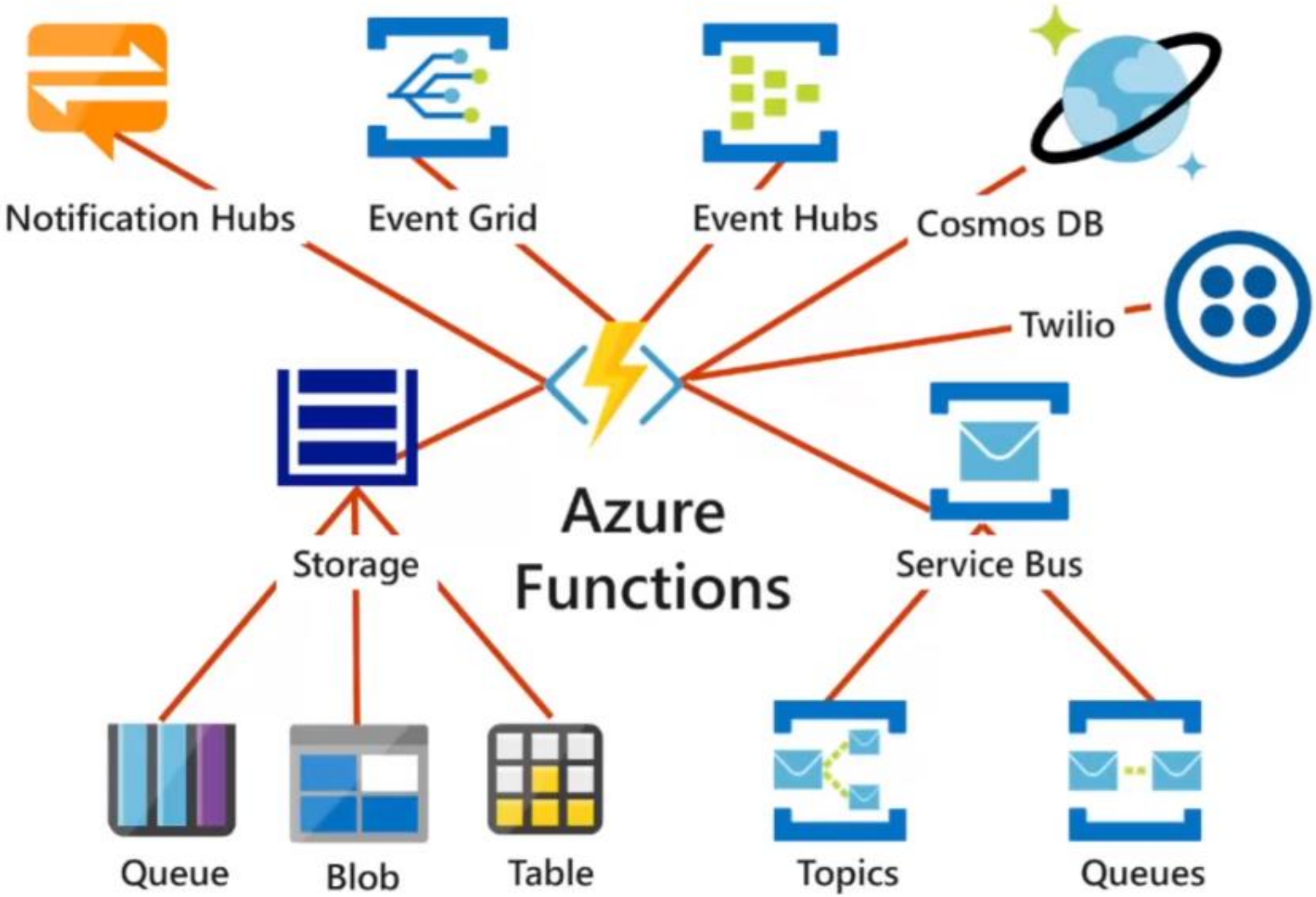
Scenarios

In many cases, a function integrates with an array of cloud services to provide feature-rich implementations.

The following are a common, *but by no means exhaustive*, set of scenarios for Azure Functions.

If you want to...	then...
Build a web API	Implement an endpoint for your web applications using the HTTP trigger
Process file uploads	Run code when a file is uploaded or changed in blob storage
Build a serverless workflow	Chain a series of functions together using durable functions
Respond to database changes	Run custom logic when a document is created or updated in Cosmos DB
Run scheduled tasks	Execute code at set times
Create reliable message queue systems	Process message queues using Queue Storage, Service Bus, or Event Hubs
Analyze IoT data streams	Collect and process data from IoT devices
Process data in real time	Use Functions and SignalR to respond to data in the moment

Function integrations



Azure Function (Java program – Function.java)

```
public class Function {
    public String echo(
        @HttpTrigger(
            name = "request",
                methods = {"post"},
            authLevel = AuthorizationLevel.ANONYMOUS
        )
        String request, ExecutionContext context) {
        return String.format(request);
    }
}
```

Azure Function (Python script – __init__.py)

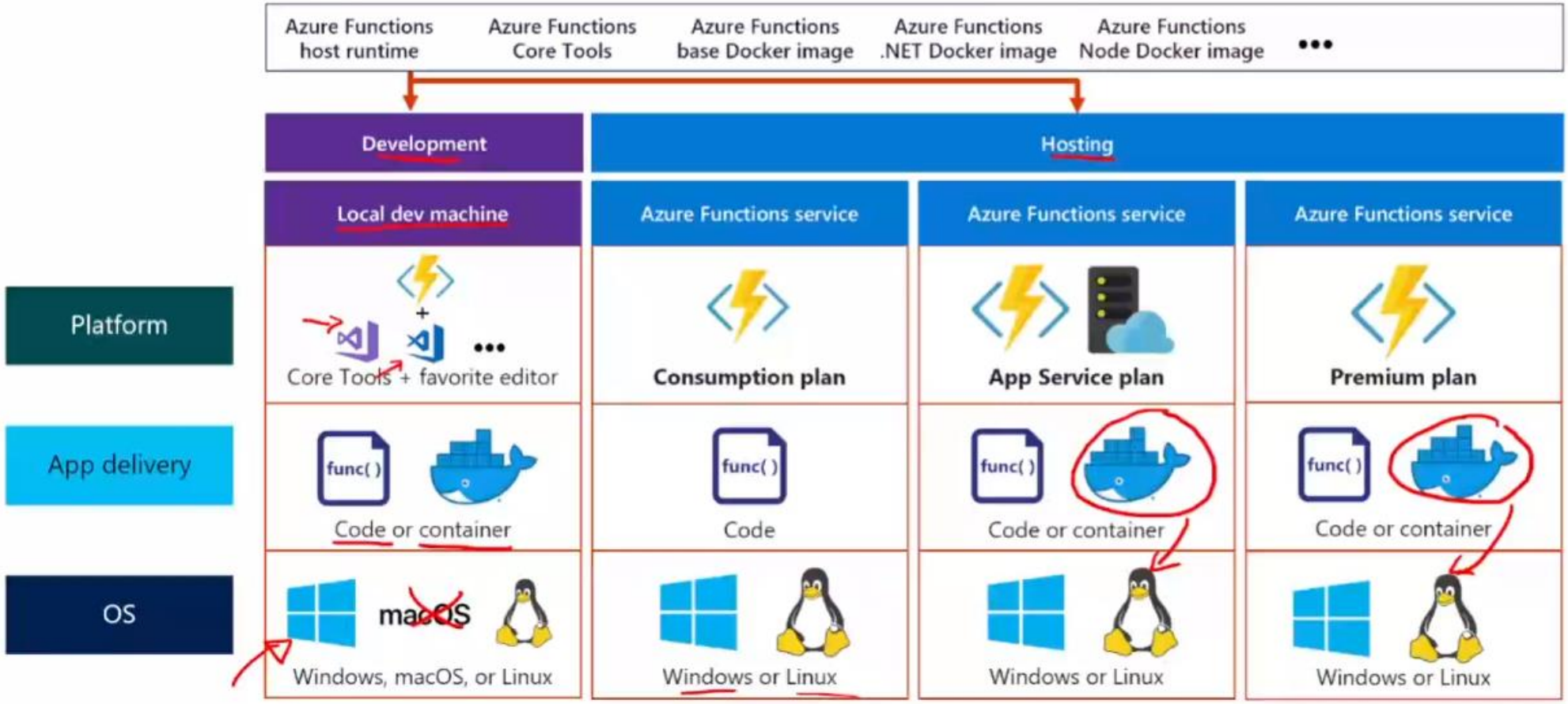
```python
import logging

import azure.functions as func

def main(myblob: func.InputStream):
    logging.info(f"Python blob trigger function processed\n"
        f"Name: {myblob.name}\n"
        f"Blob Size: {myblob.length} bytes")
```

Scale and hosting

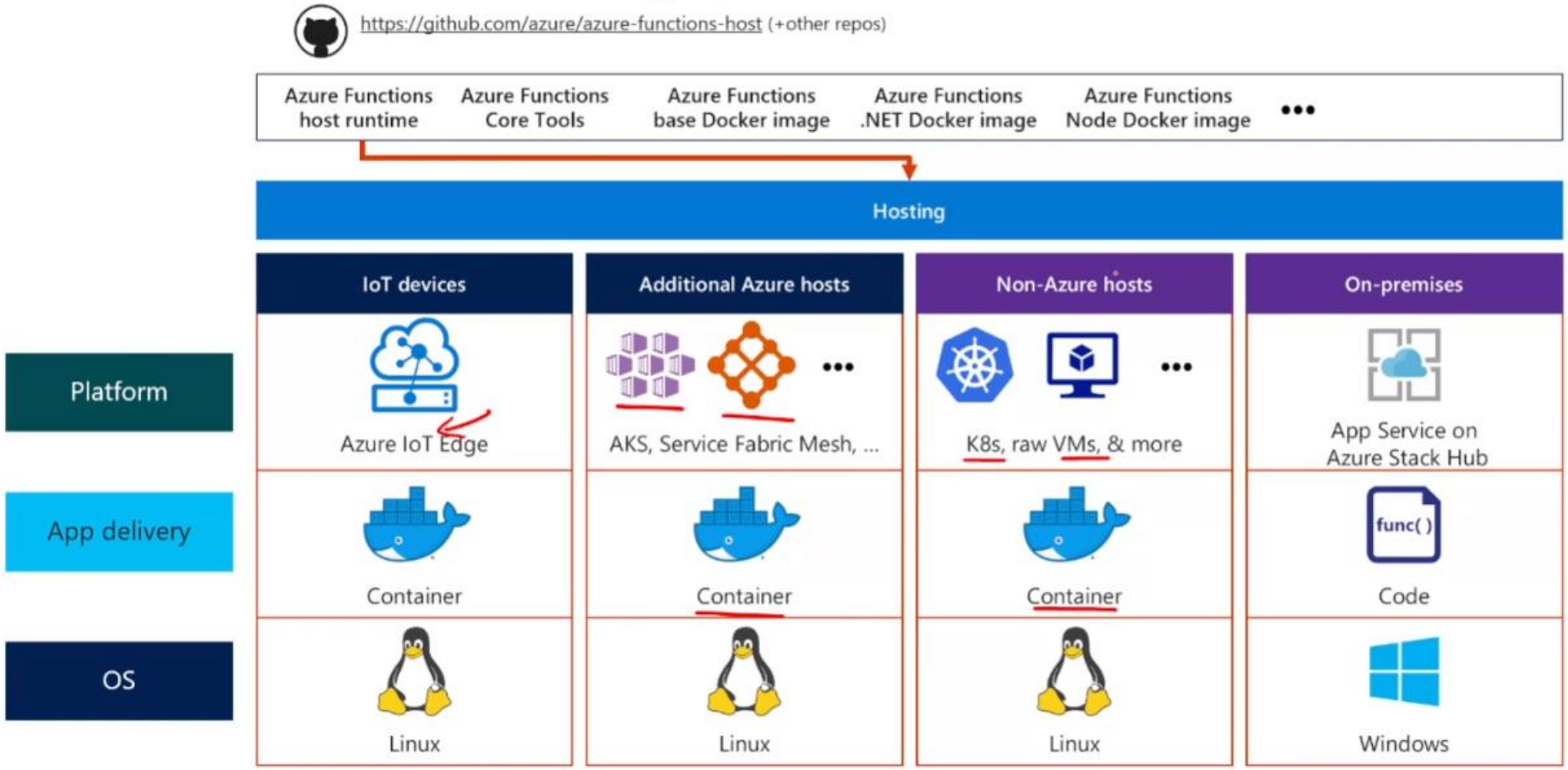
- You can choose between three types of plans:
  - **Consumption**:
    - Instances are dynamically instanced and you are charged based on compute time
  - **Premium**
    - Instances of the Azure Functions host are added and removed based on the number of incoming events just like the Consumption plan, but provides additional features like: VNet connectivity; unlimited execution duration; and more predictable pricing, docker container, ...
  - **App Service plan (Dedicated):**
    - Traditional App Services model used with Web Apps, API Apps, and Mobile Apps
- The type of plan controls:
  - How host instances are scaled out
  - The resources that are available to each host

Azure Functions hosting

https://github.com/azure/azure-functions-host (+other repos)



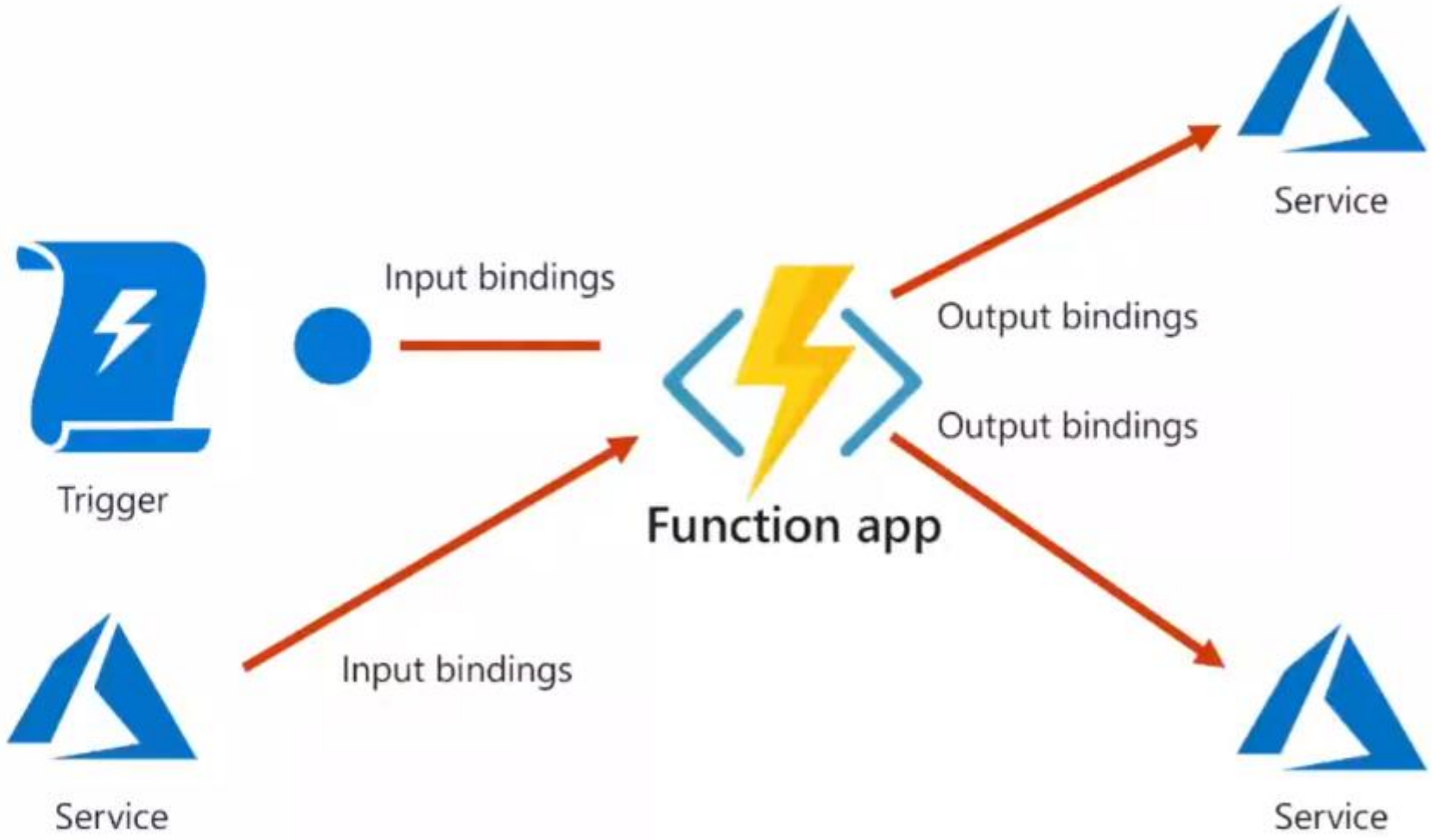
Azure Functions hosting (continued)



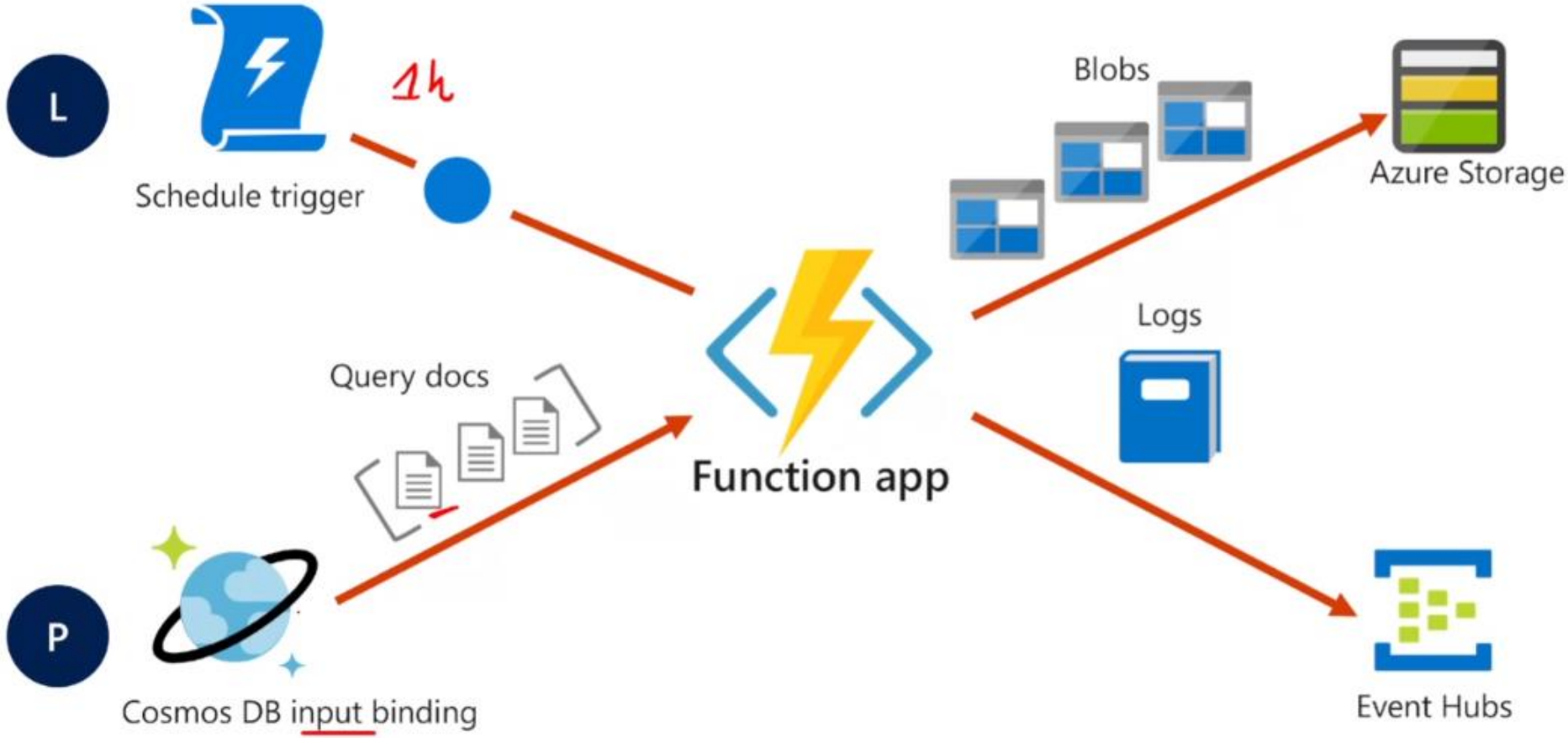
Trigger types

- Triggers based on Azure services:
  - Cosmos DB
  - Blob and queues
  - Service Bus
  - Event Hub
- Triggers based on common scenarios:
  - HTTP request
  - Scheduled timer
- Triggers based on third-party services:
  - GitHub
- And more...

Input and Output Bindings

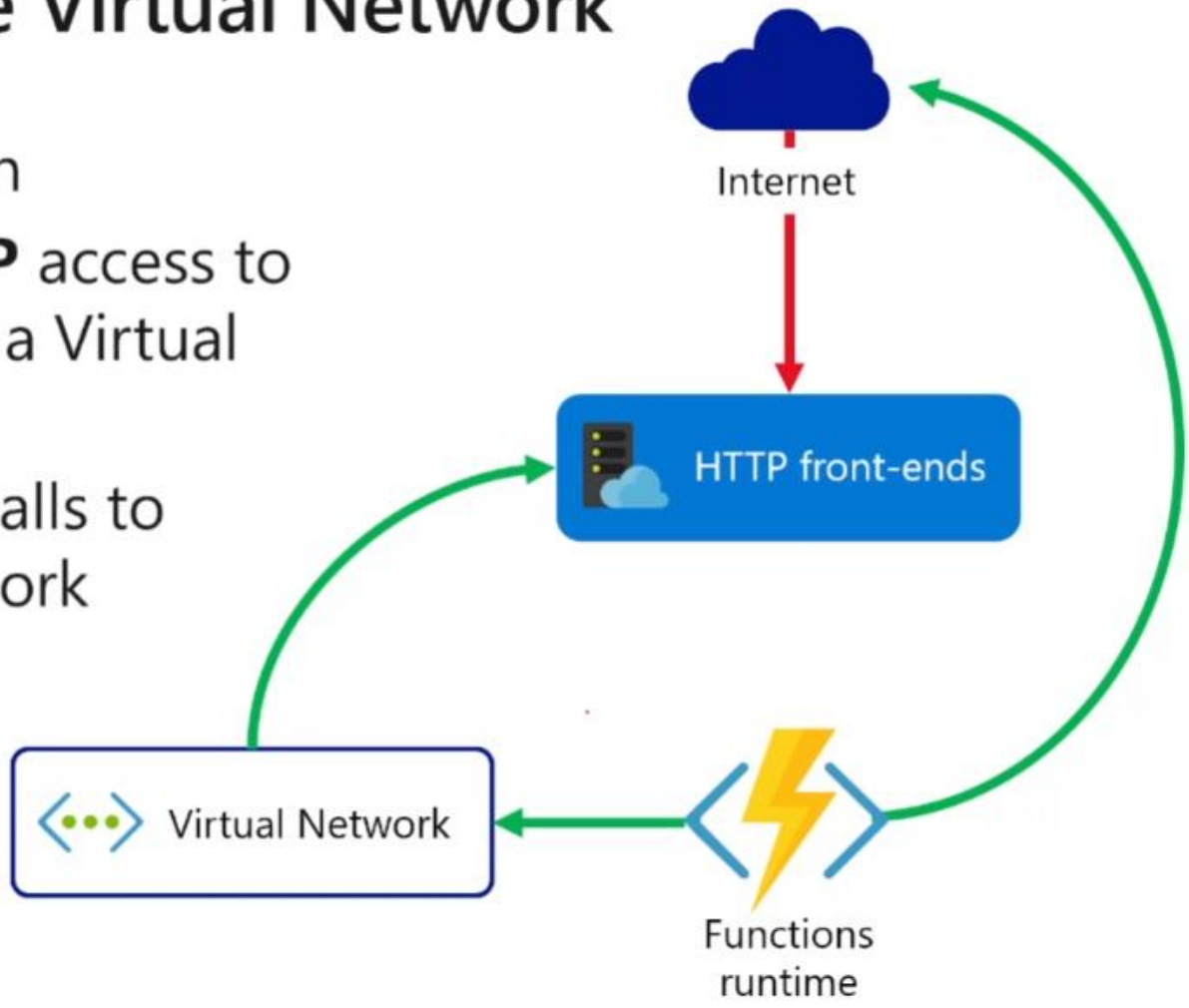


Trigger and Bindings example

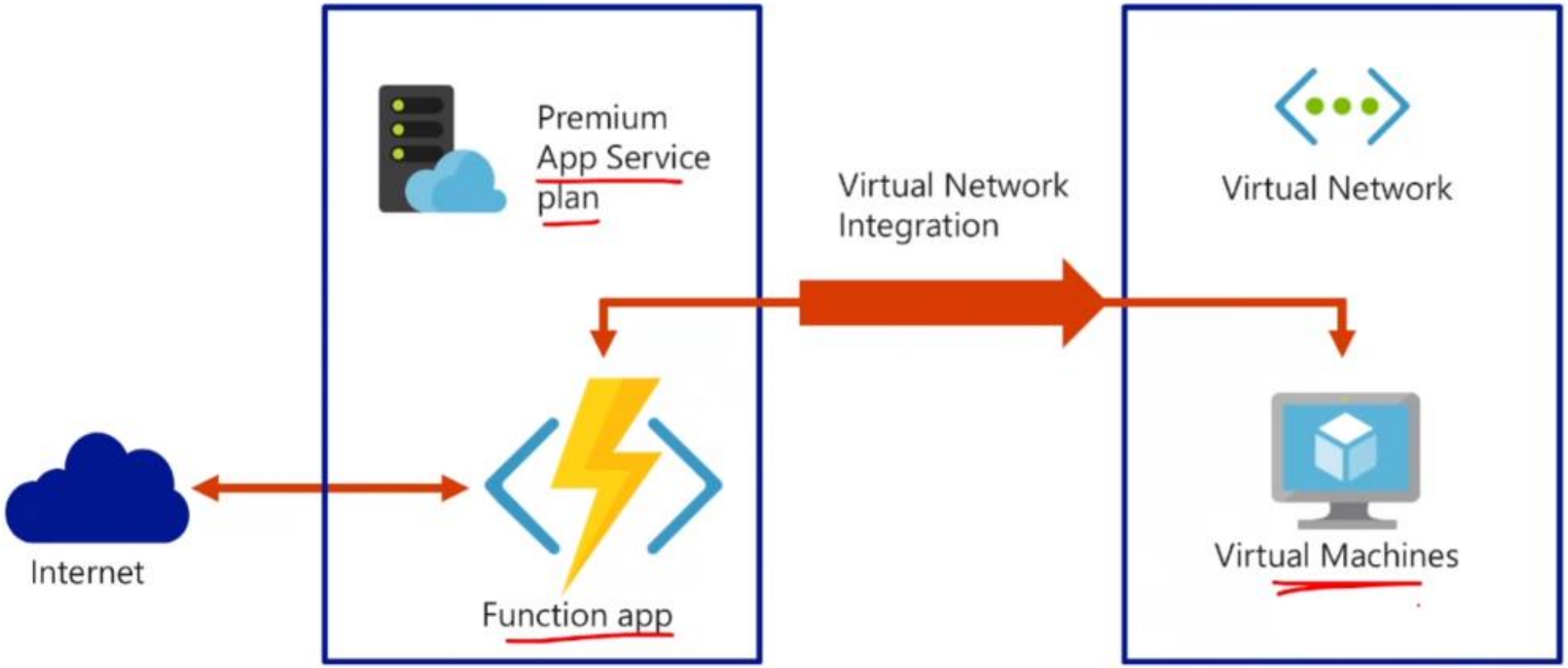


Integrating with Azure Virtual Network

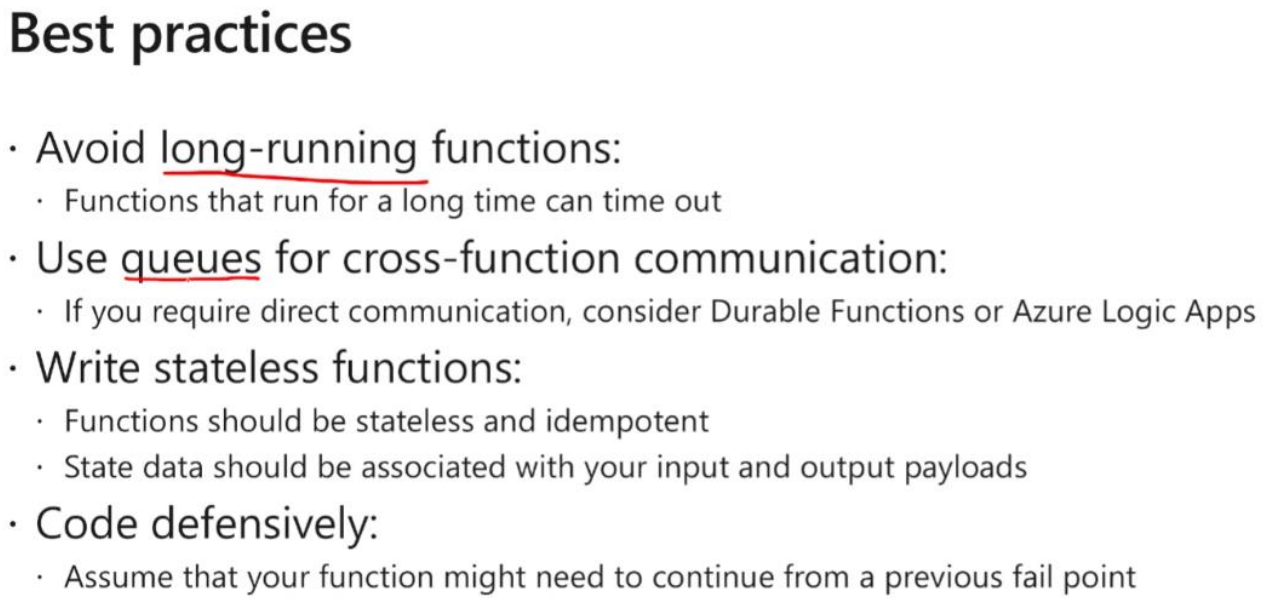
- Requires the **Premium** plan
- Secures the **inbound HTTP** access to your app to one **subnet** in a Virtual Network
- Allows **secure outbound** calls to resources in a Virtual Network



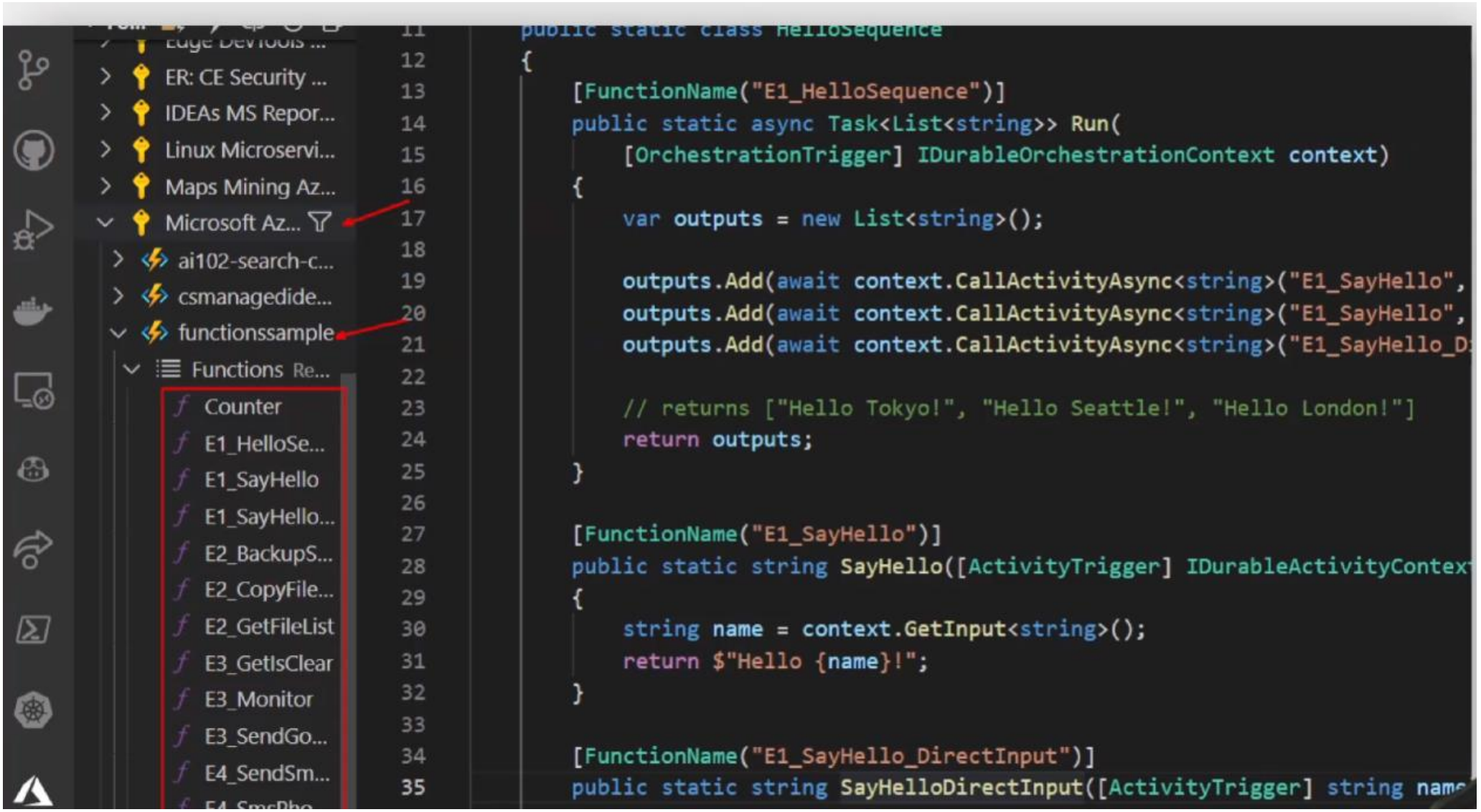
Azure Virtual Network integration example

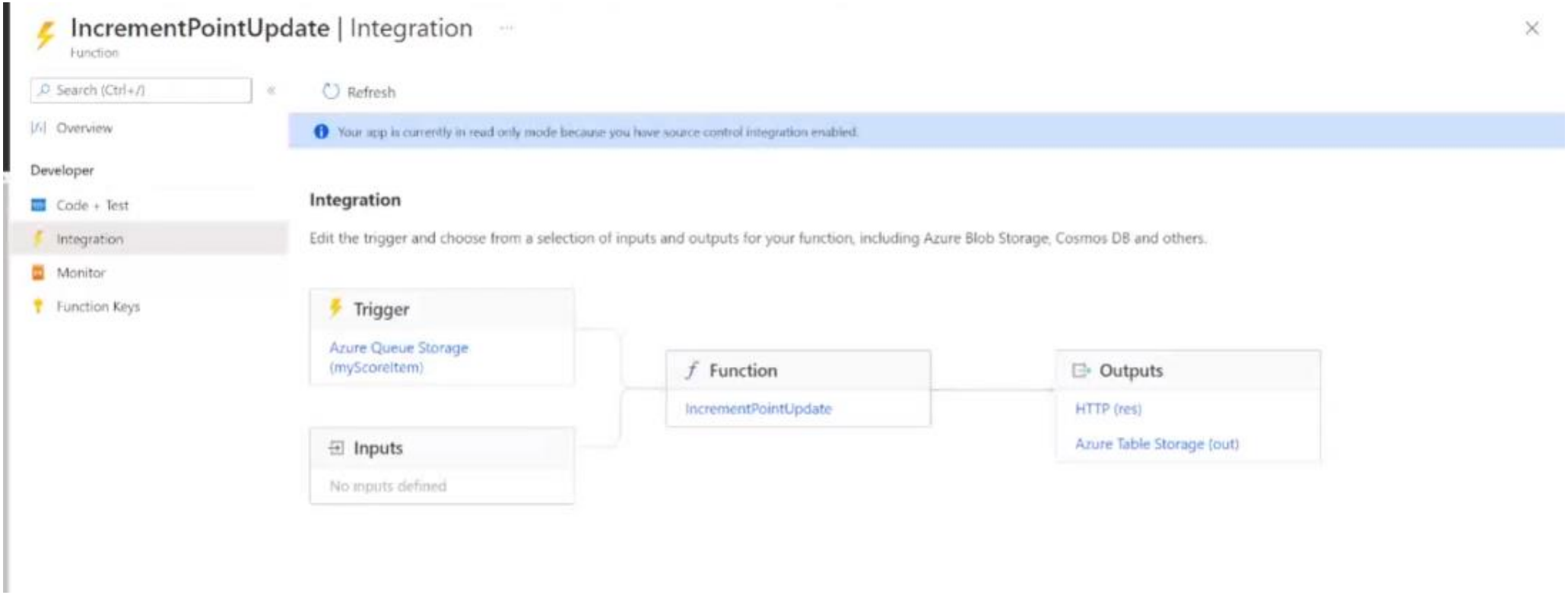


Best practices
- Avoid lon -running functions:
° Functions that run for a long time can time out
- Use queues for cross—function communication:
' If you require direct communication, consider Durable Functions or Azure Logic Apps
- Write stateless functions:
- Functions should be stateless and idempotent
~ State data should be associated with your input and output payloads
- Code defensively:
- Assume that your function might need to continue from a previous fail point



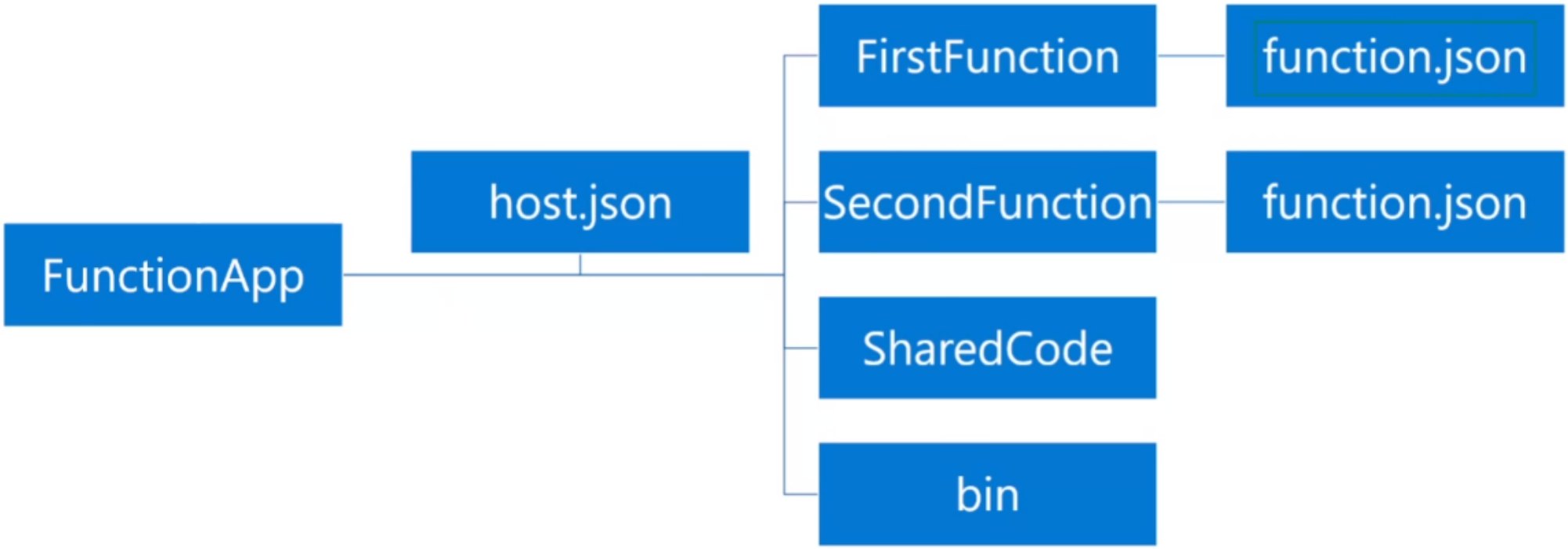
Lesson 02: Developing Azure Functions





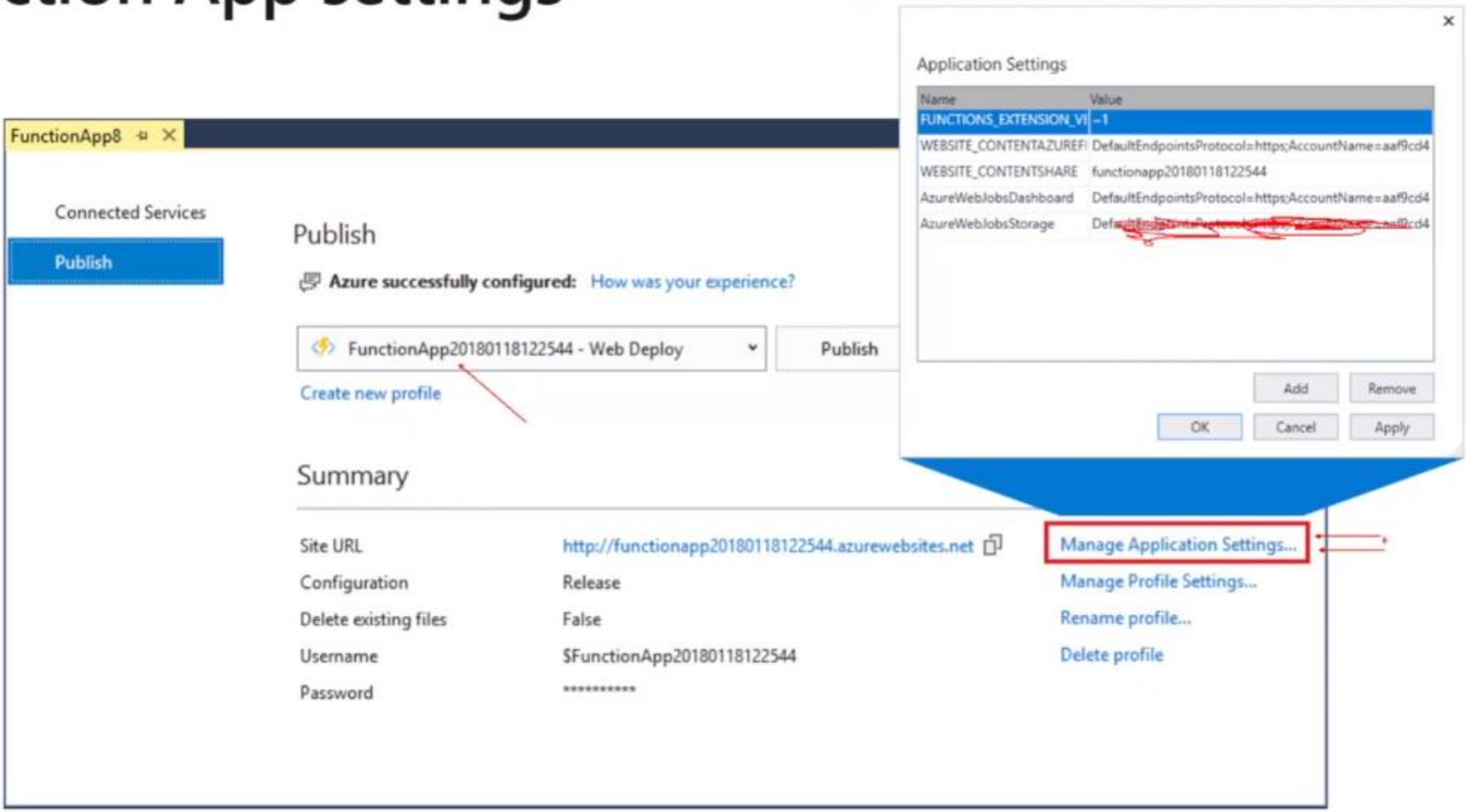
VSCode’dan yarattığın kodu/fonksiyonu portal içinden değiştiremiyorsun.

Function folder structure (C#)



Function App settings

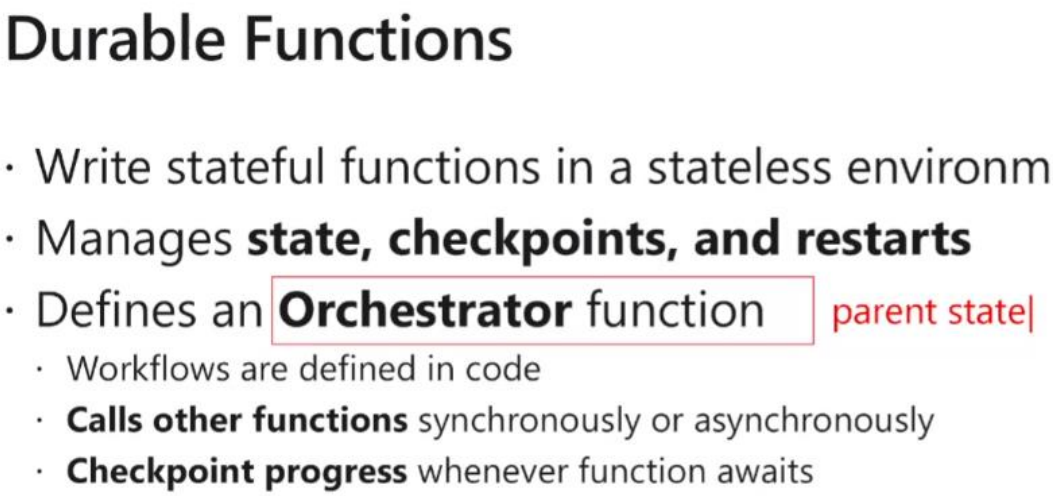
function (cosmosdb)



Local'de farklı config, publish ettiğin yerde başka bir config olmalıdır.

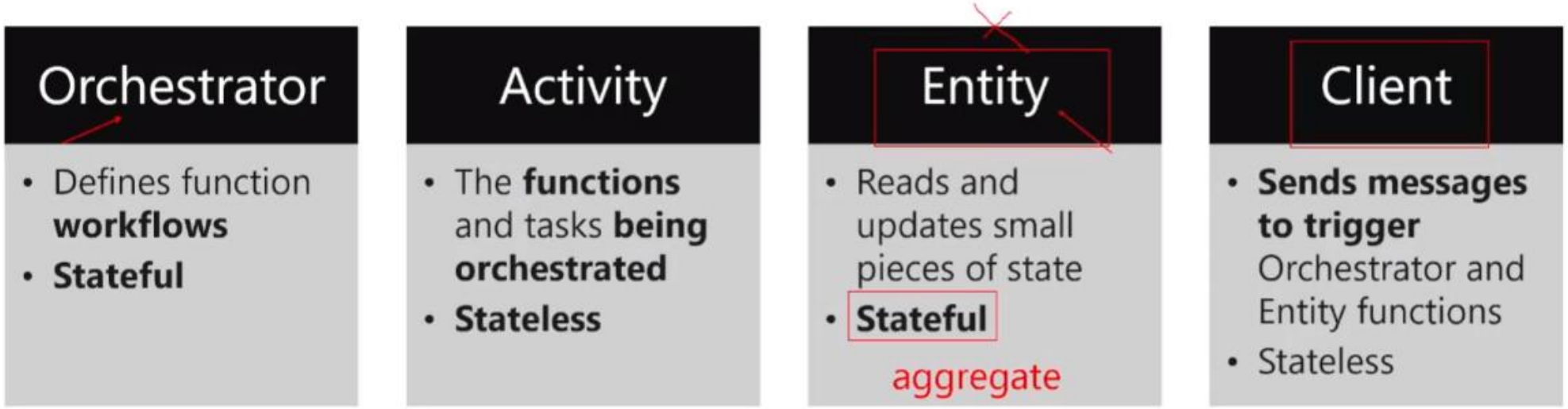
Lesson 03: Implement Durable Functions

Durable Functions
- Write stateful functions in a stateless environment
- Manages state, checkpoints, and restarts
- Defines an Orchestrator function
- Workflows are defined in code
- Calls other functions synchronously or asynchronously
- Checkpoint progress whenever function awaits



Durable Functions types

azure function --> storage account |

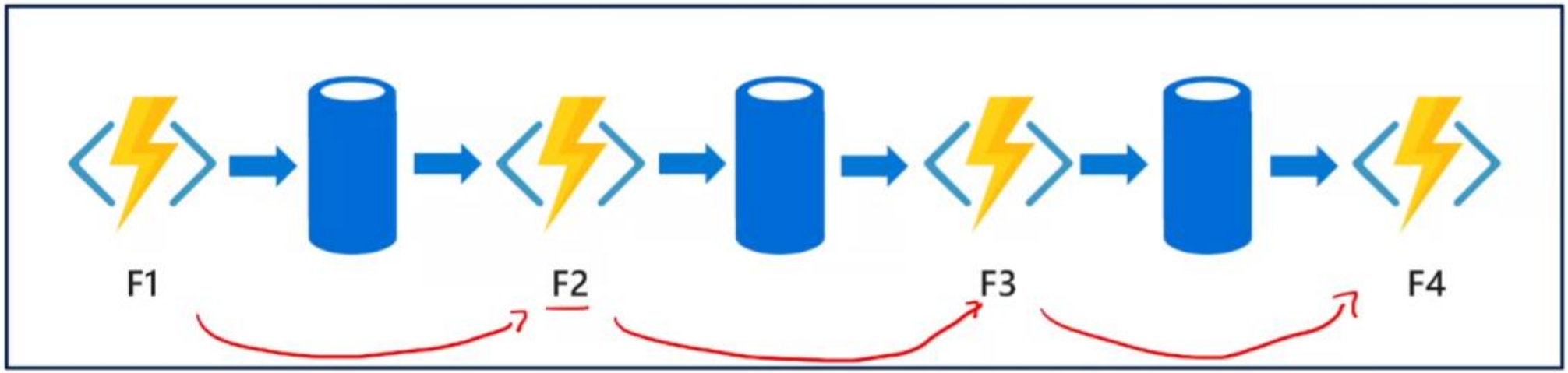


- State is checkpointed and maintained in Azure Storage

Orkestratör yönetiyor. State'i var.

Durable Function scenario - Chaining

Function chaining refers executing a sequence of functions in a particular order. Often, the output of one function needs to be applied to the input of another function.



Durable Function scenario - Chaining code

```
public static async Task<object> Run(DurableOrchestrationContext ctx)
{
    try
    {
        var x = await ctx.CallActivityAsync<object>("F1");
        var y = await ctx.CallActivityAsync<object>("F2", x);
        var z = await ctx.CallActivityAsync<object>("F3", y);
        return await ctx.CallActivityAsync<object>("F4", z);
    }
    catch (Exception)
    {
        // error handling/compensation goes here
    }
}
```

```
public static class HelloSequence
{
    [FunctionName("E1_HelloSequence")]
    public static async Task<List<string>> Run(
        [OrchestrationTrigger] IDurableOrchestrationContext context)
    {
        var outputs = new List<string>();

        outputs.Add(await context.CallActivityAsync<string>("E1_SayHello", "Tokyo"));
        outputs.Add(await context.CallActivityAsync<string>("E1_SayHello", "Seattle"));
        outputs.Add(await context.CallActivityAsync<string>("E1_SayHello_DirectInput", "London"));

        // returns ["Hello Tokyo!", "Hello Seattle!", "Hello London!"]
        return outputs;
    }

    [FunctionName("E1_SayHello")]
    public static string SayHello([ActivityTrigger] IDurableActivityContext context)
    {
        string name = context.GetInput<string>();
        return $"Hello {name}!";
    }

    [FunctionName("E1_SayHello_DirectInput")]
    public static string SayHelloDirectInput([ActivityTrigger] string name)
    {
```

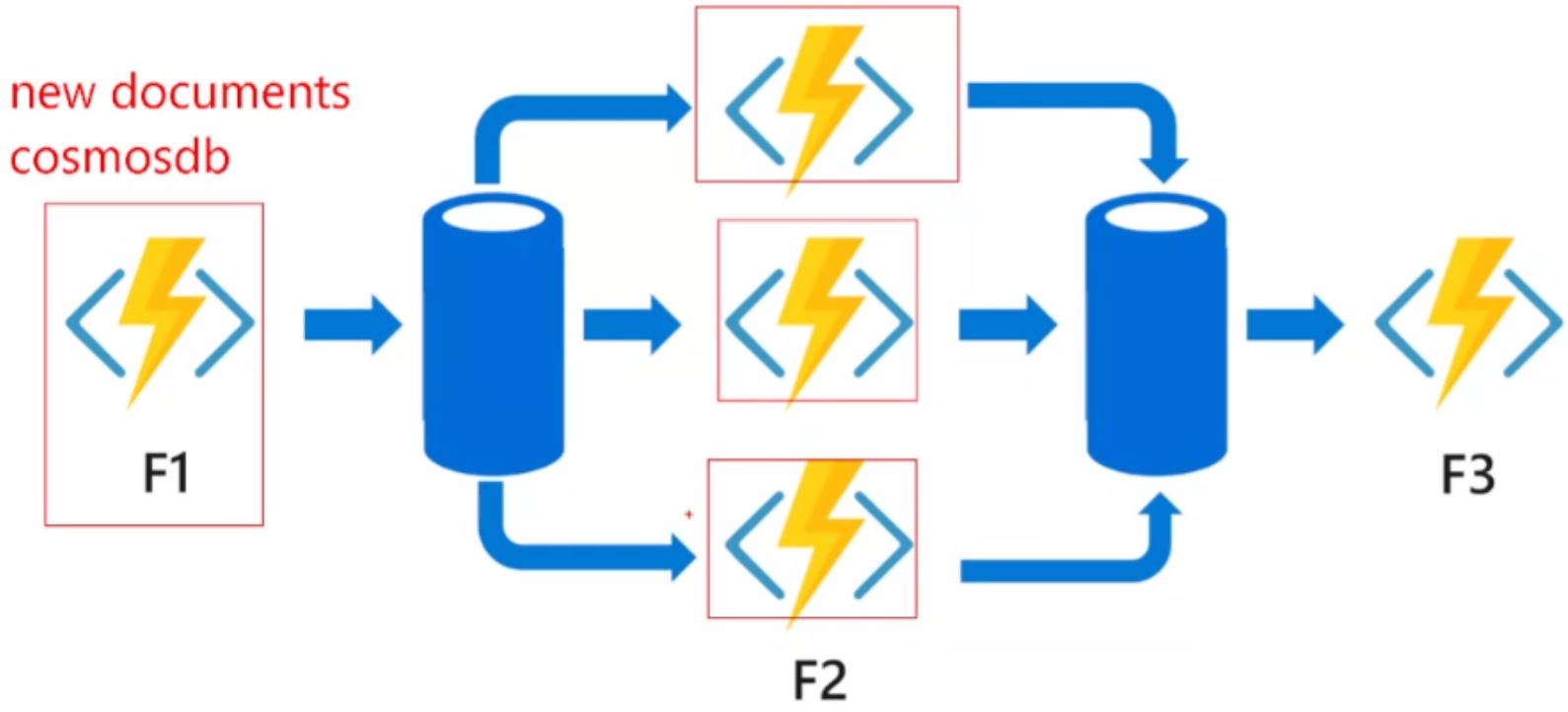
Filter by name...

Name	Trigger
Counter	Entity
E1_HelloSequence	Orchestration
E1_SayHello	Activity
E1_SayHello_DirectInput	Activity
E2_BackupSiteContent	Orchestration
E2_CopyFileToBlob	Activity
E2_GetFileList	Activity
E3_GetIsClear	Activity
E3_Monitor	Orchestration
E3_SendGoodWeatherAlert	Activity

PRemium’da hemen para yazar. Warmup olduğu için. +çalışma başına.

Durable Function scenario - Fan-out/fan-in

Fan-out/fan-in refers to the pattern of executing **multiple functions in parallel**, and then **waiting** for all to finish

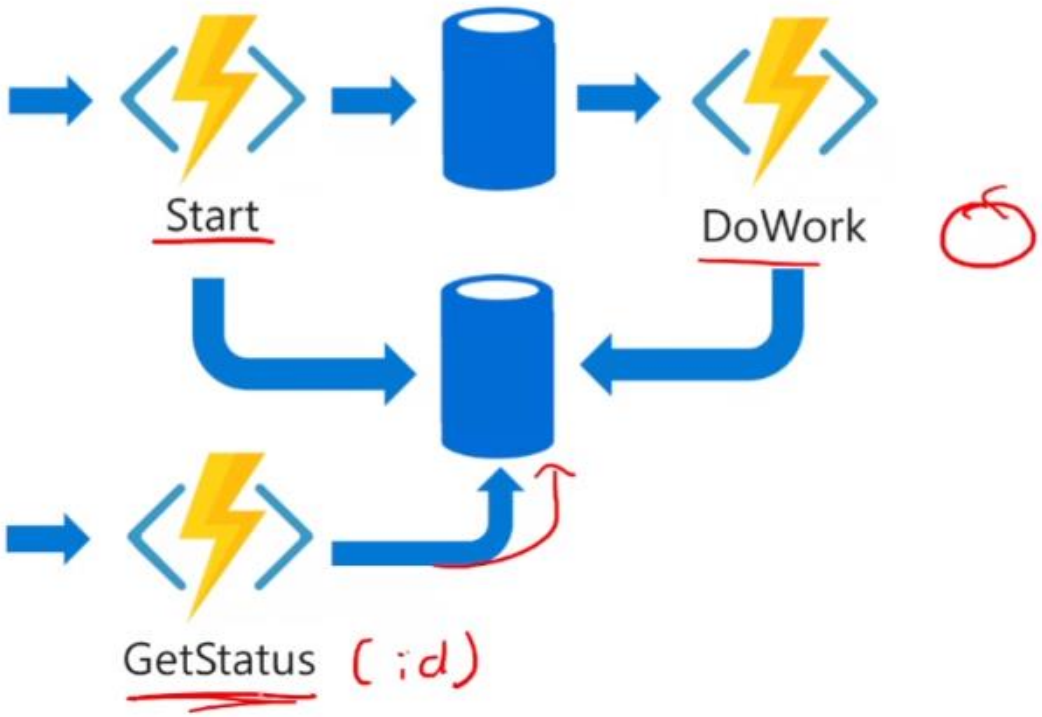


Durable Function scenario - Fan-out/fan-in code

```
public static async Task Run(DurableOrchestrationContext ctx)
{
    var parallelTasks = new List<Task<int>>();
    // get a list of N work items to process in parallel
    object[] workBatch = await ctx.CallActivityAsync<object[]>("F1");
    for (int i = 0; i < workBatch.Length; i++)
    {
        Task<int> task = ctx.CallActivityAsync<int>("F2", workBatch[i]);
        parallelTasks.Add(task);
    }
    await Task.WhenAll(parallelTasks);
    // aggregate all N outputs and send result to F3
    int sum = parallelTasks.Sum(t => t.Result);
    await ctx.CallActivityAsync("F3", sum);
}
```

Durable Function scenario - Async HTTP APIs

Durable Functions provides built-in APIs that simplify the code that you write for interacting with long-running function executions

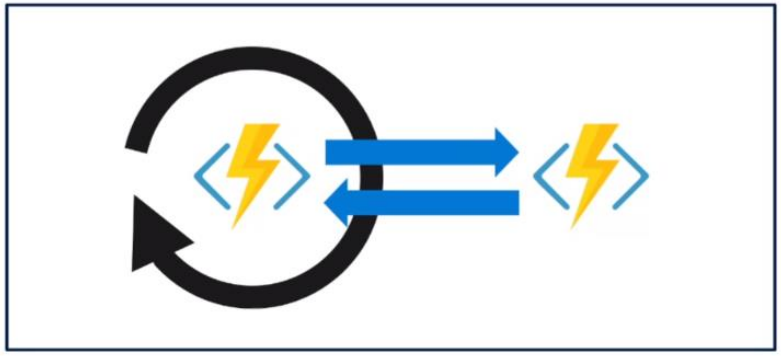


Durable Function scenario - Async HTTP APIs code

```
// HTTP-triggered function to start a new orchestrator function instance.
public static async Task<HttpResponseMessage> Run(
    HttpRequestMessage req,
    DurableOrchestrationClient starter,
    string functionName,
    ILogger log)
{
    // Function name comes from the request URL.
    // Function input comes from the request content.
    dynamic eventData = await req.Content.ReadAsAsync<object>();
    string instanceId = await starter.StartNewAsync(functionName, eventData);

    log.LogInformation($"Started orchestration with ID = '{instanceId}'.");

    return starter.CreateCheckStatusResponse(req, instanceId);
}
```

Durable Function scenario - Monitoring

The monitor pattern refers to a flexible recurring process in a workflow—for example, **polling until certain conditions are met**

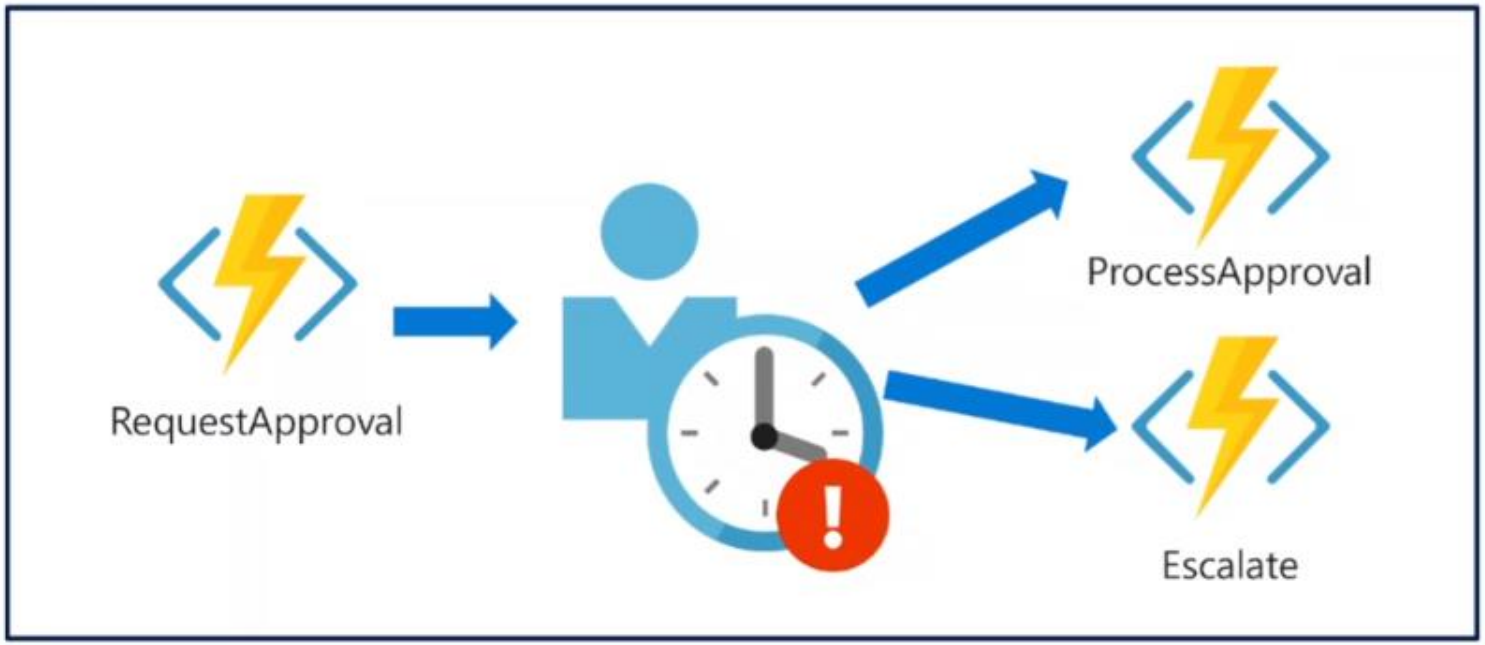


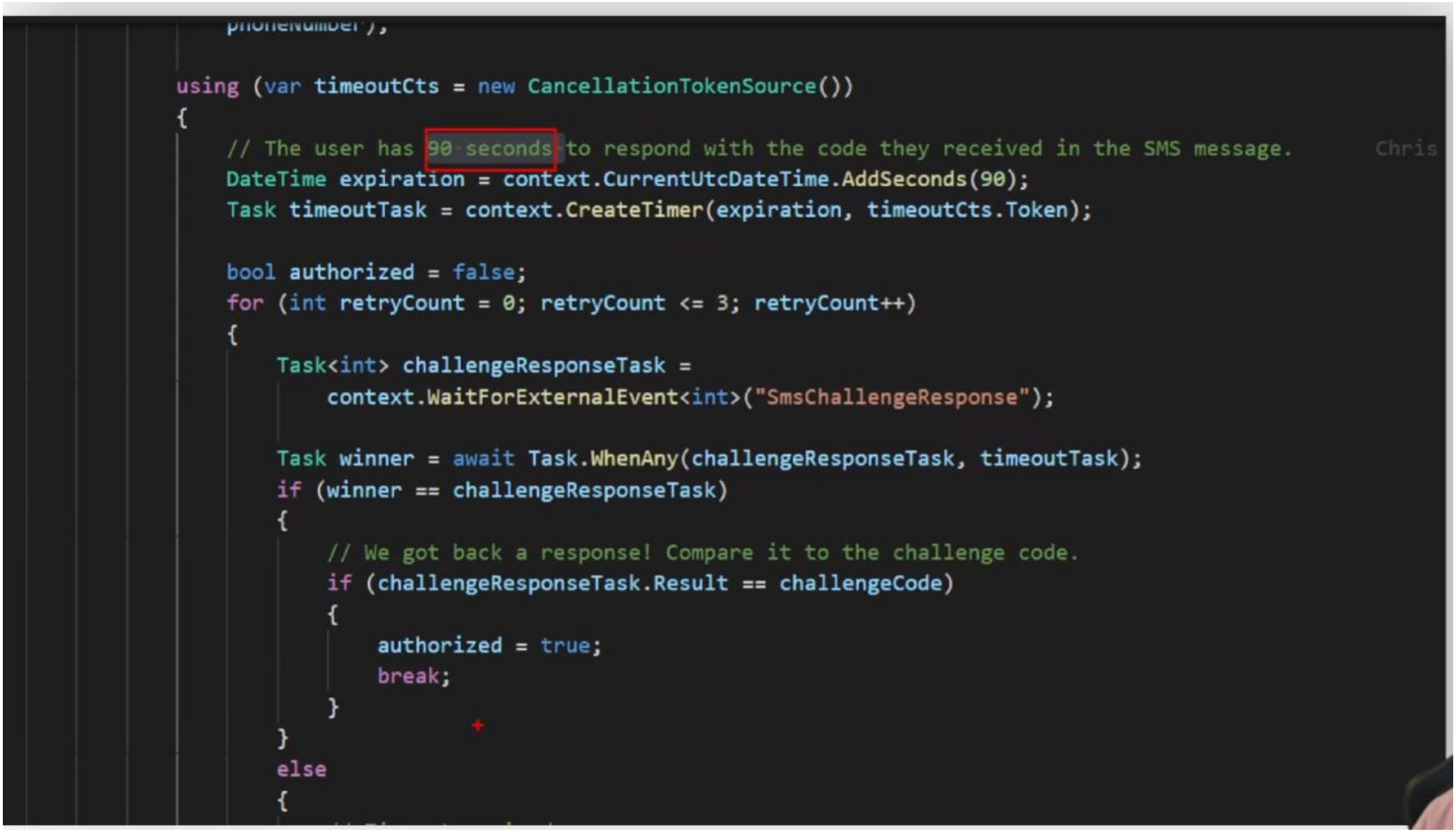
Durable Function scenario - Monitoring code

```
public static async Task Run(DurableOrchestrationContext ctx)
{
    int jobId = ctx.GetInput<int>(); int pollingInterval = GetPollingInterval();
    DateTime expiryTime = GetExpiryTime();
    while (ctx.CurrentUtcDateTime < expiryTime)
    {
        var jobStatus = await ctx.CallActivityAsync<string>("GetJobStatus", jobId);
        if (jobStatus == "Completed")
        { await ctx.CallActivityAsync("SendAlert", machineId); break; }
        // Orchestration will sleep until this time
        var nextCheck = ctx.CurrentUtcDateTime.AddSeconds(pollingInterval);
        await ctx.CreateTimer(nextCheck, CancellationToken.None);
    }
    // Perform further work here, or let the orchestration end
}
```

Durable Function scenario - Human interaction

Many processes involve human interaction. Automated processes must allow for human low availability, and they often do so by using time-outs and compensation logic.




phoneNumber");
using (var timeoutCts = new CancellationTokenSource())
{
// The user has 90 seconds to respond with the code they received in the SMS message.
DateTime expiration = context.CurrentUtcDateTime.AddSeconds(90);
Task timeoutTask = context.CreateTimer(expiration, timeoutCts.Token);
bool authorized = false;
for (int retryCount = 0; retryCount <= 3; retryCount++)
{
Task<int> challengeResponseTask =
context.WaitForExternalEvent<int>("SmsChallengeResponse");
Task winner = await Task.WhenAny(challengeResponseTask, timeoutTask);
if (winner == challengeResponseTask)
{
// We got back a response! Compare it to the challenge code.
if (challengeResponseTask.Result == challengeCode)
{
authorized = true;
break;
}
}
else
{

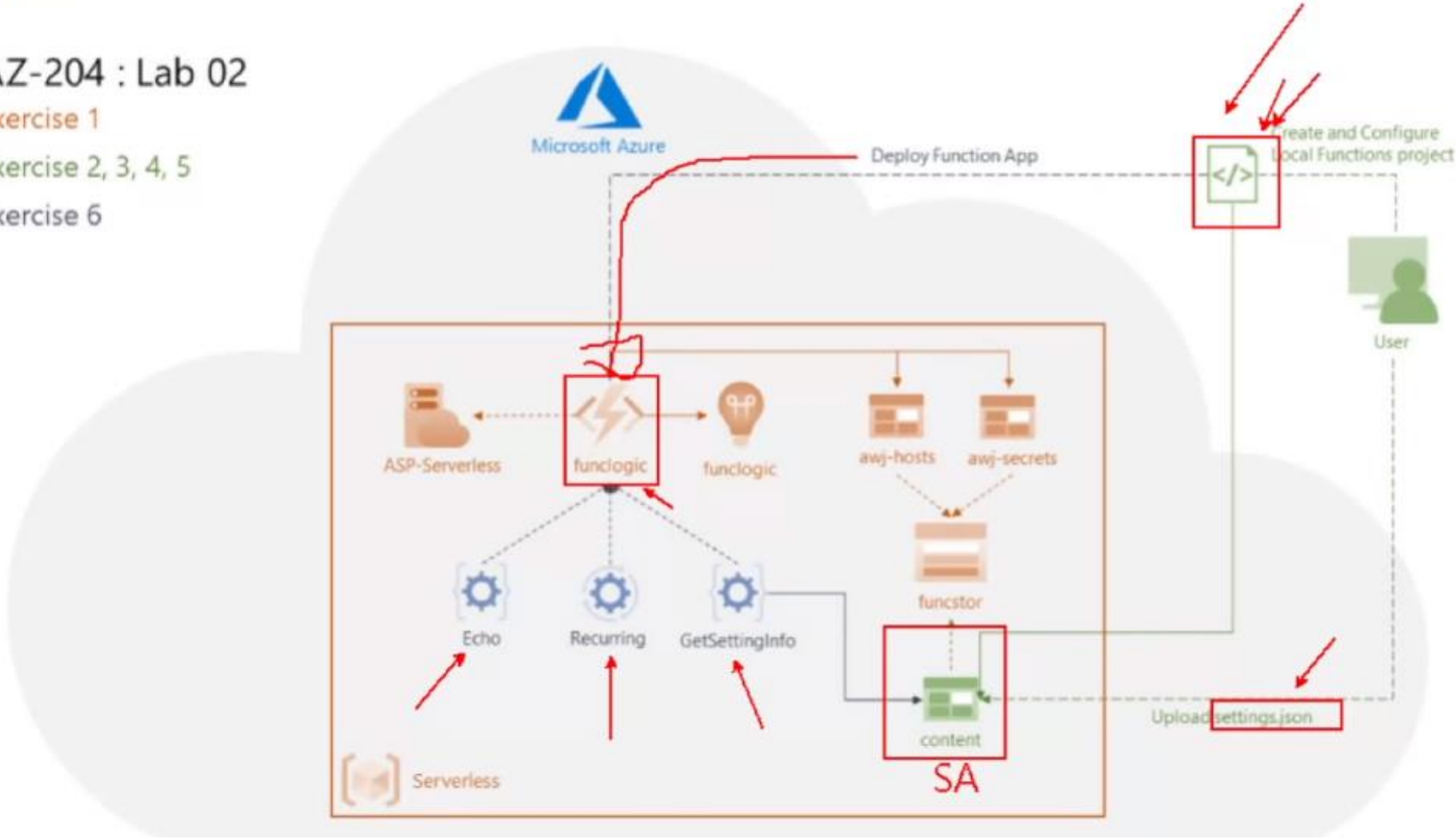
Excercises

AZ-204 : Lab 02

Exercise 1

Exercise 2, 3, 4, 5

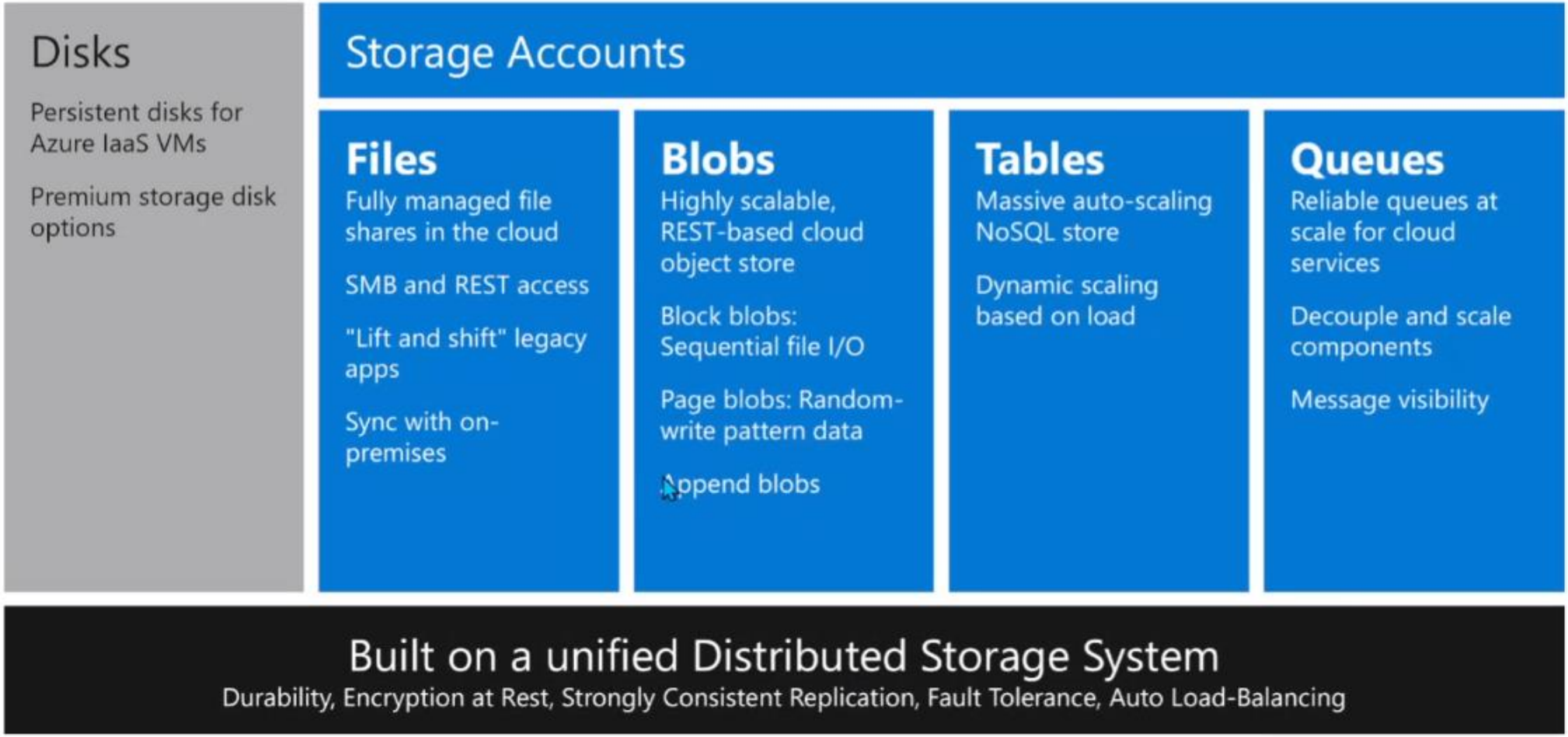
Exercise 6



Topics

- Azure Blob storage core concepts
- Managing the Azure Blob storage lifecycle
- Working with Azure Blob storage

Azure Storage overview



Azure Blob storage

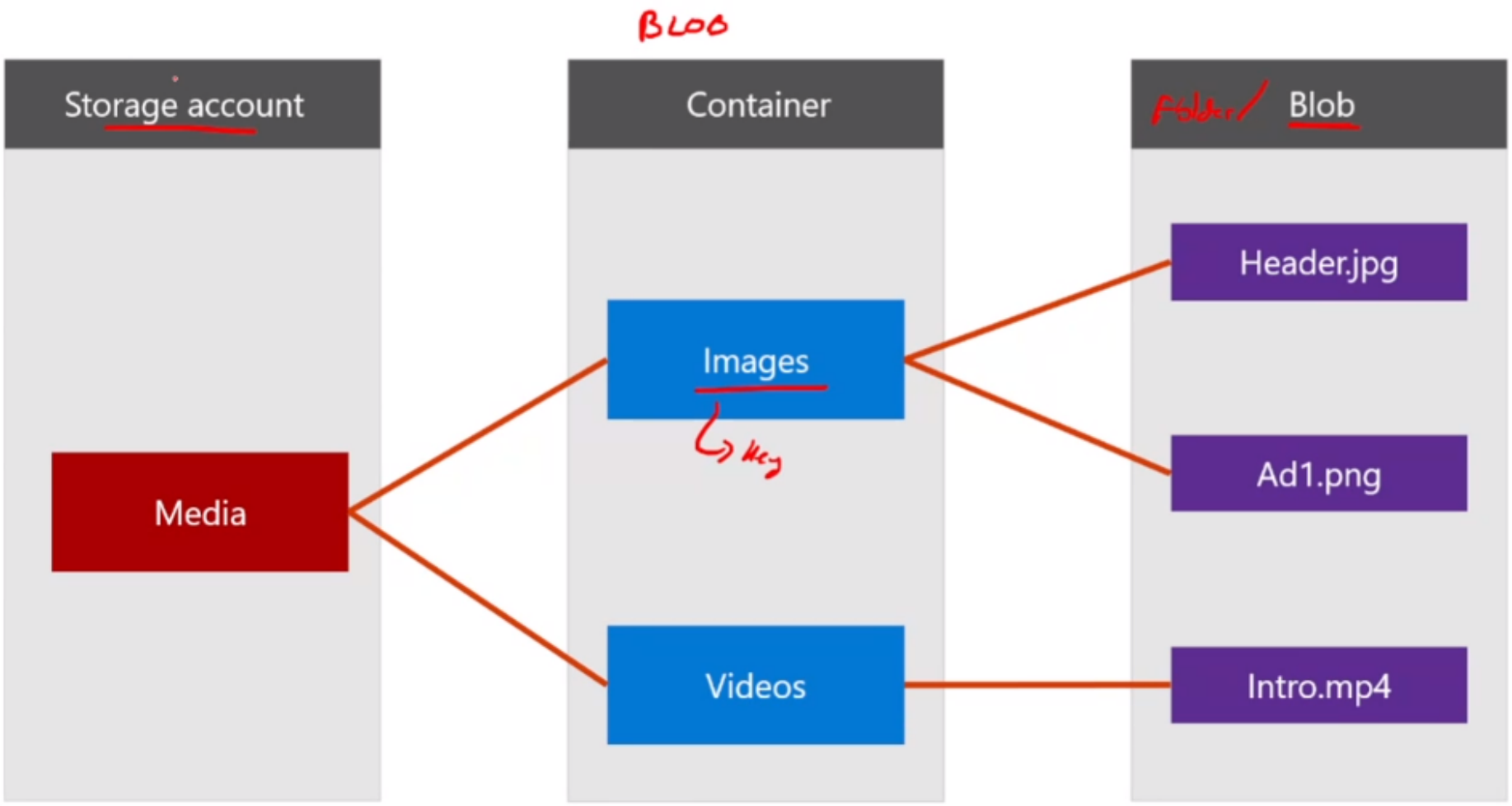
- Object storage solution in the cloud
- Blob storage is designed for:
  - Serving **images or documents** directly to a browser
  - **Storing files** for distributed access
  - Streaming video and audio
  - Writing to **log files**
  - Storing data for **backup and restore, disaster recovery**, and archiving
  - **Storing data for analysis** by an on-premises or Azure-hosted service
- Accessible via a HTTP/HTTPS API

Storage Account types

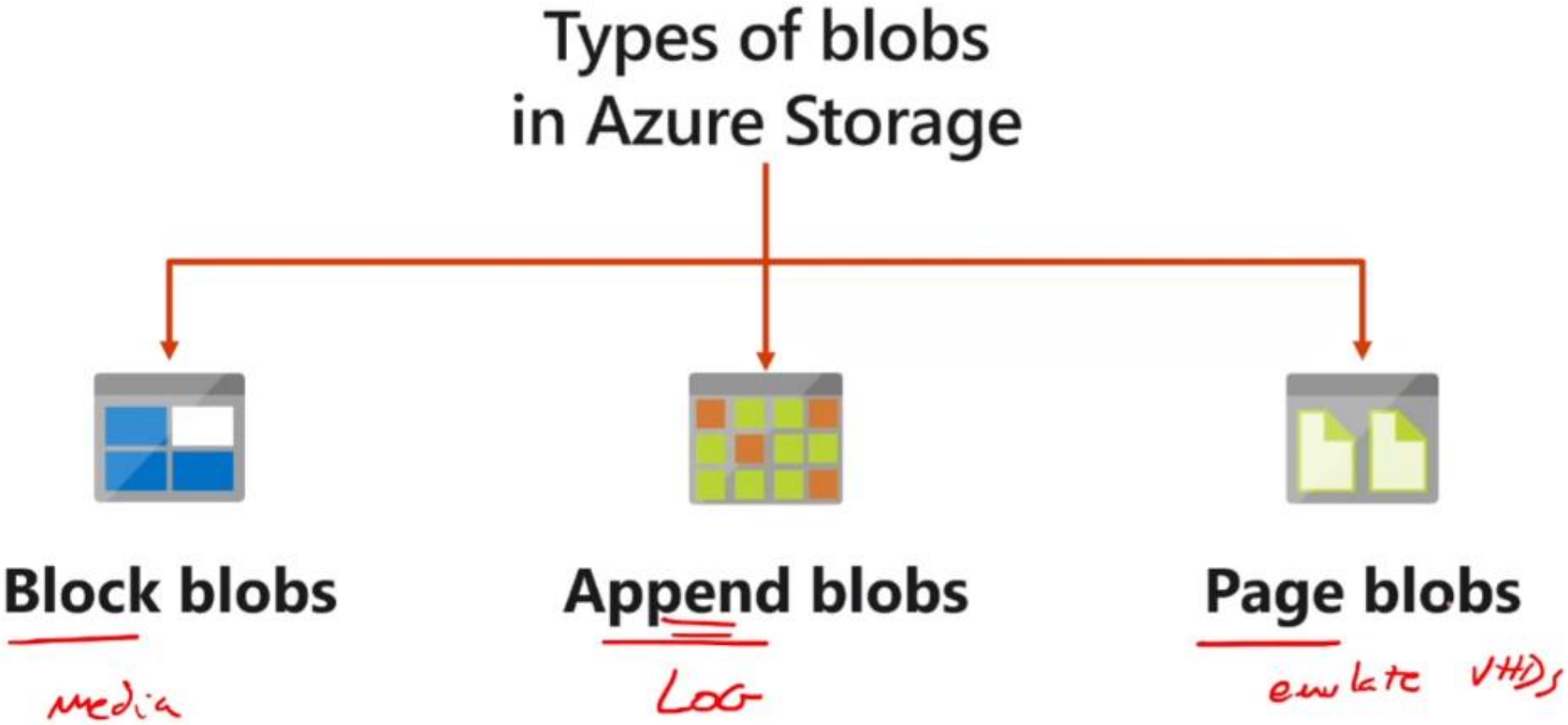
Type of storage account	Supported storage services	Redundancy options	Usage
Standard general-purpose v2	Blob (including Data Lake Storage[1]), Queue, and Table storage, Azure Files	LRS/GRS/RA-GRS ZRS/GZRS/RA-GZRS[2]	Standard storage account type for blobs, file shares, queues, and tables. Recommended for most scenarios using Azure Storage. Note that if you want support for NFS file shares in Azure Files, use the premium file shares account type.
Premium block blobs[3]	Blob storage (including Data Lake Storage[1])	LRS ZRS[2]	Premium storage account type for block blobs and append blobs. Recommended for scenarios with high transactions rates, or scenarios that use smaller objects or require consistently low storage latency. Learn more about example workloads.
Premium file shares[3]	Azure Files	LRS ZRS[2]	Premium storage account type for file shares only. Recommended for enterprise or high-performance scale applications. Use this account type if you want a storage account that supports both SMB and NFS file shares.
Premium page blobs[3]	Page blobs only	LRS	Premium storage account type for page blobs only. Learn more about page blobs and sample use cases.

PREMIUM DAHA GÜVENİLİR. YEDEKLEME OPSIYONLARINA BAK. +SSD

Azure Blob storage resource hierarchy

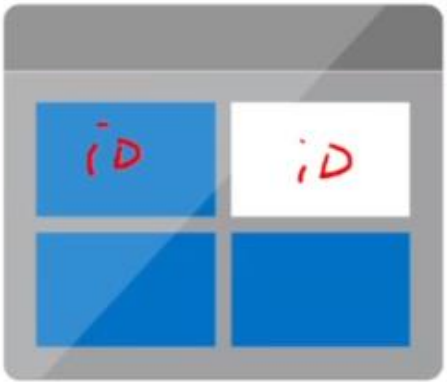


Blob types


Types of blobs
in Azure Storage
Block blobs
Append blobs
Page blobs
media
Log
emulate VHDs

Block blobs

- Comprise blocks of data
- Ideal for data that is stored in blocks—up to 100-MB chunks
- Simultaneous upload of large blobs with a single write operation
- A single block blob can include up to 50,000 blocks



4.75 TB

Append blobs

- Append blobs include the following characteristics:
  - They are composed of blocks
  - They are optimized for append operations
  - They are ideal for performant logging

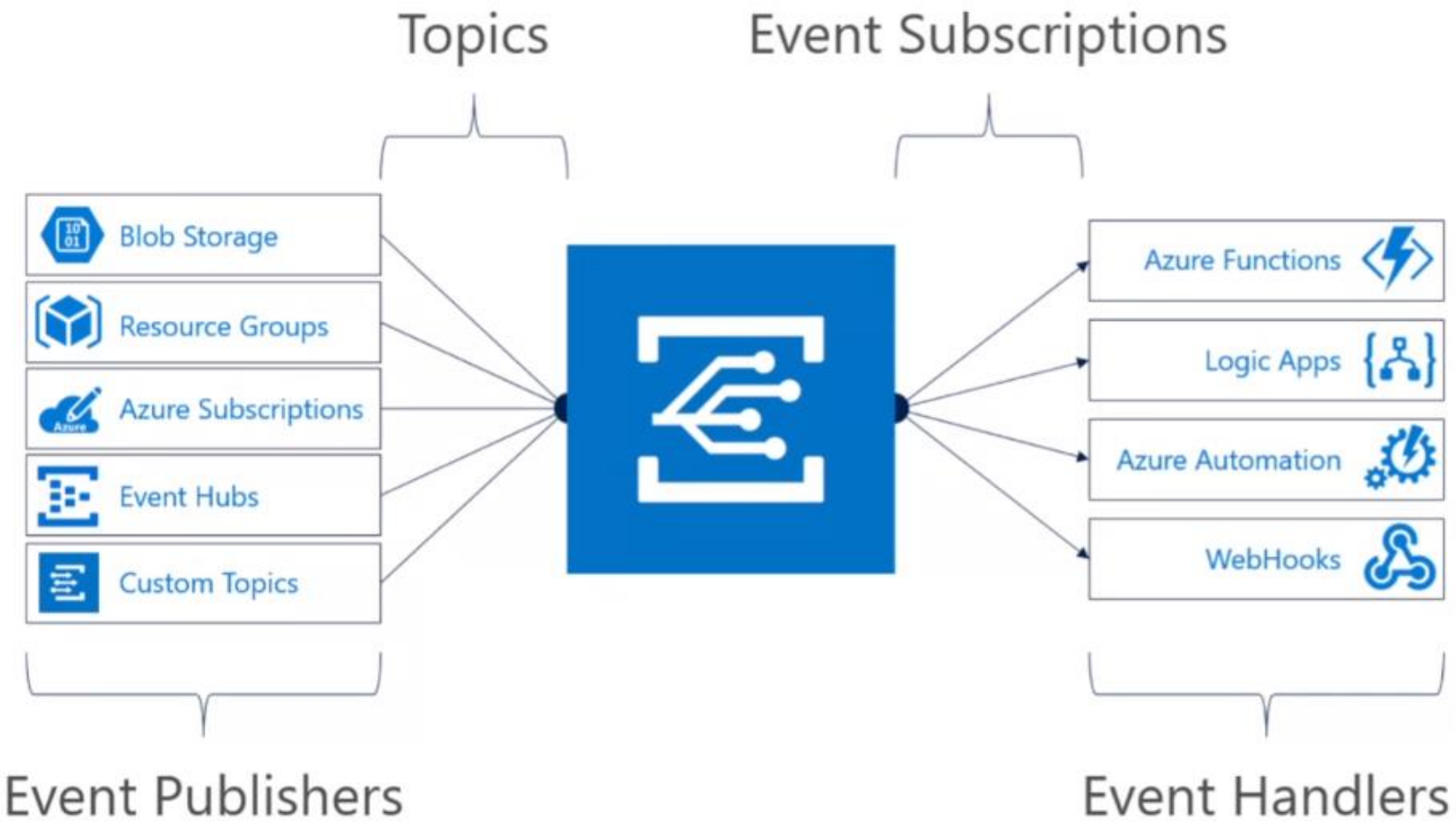
4MB

195 GB

Page blobs

- Composed of 512-byte pages
- Similar to hard disk storage
- Ideal for virtual hard disks VHD
- Pages created by initializing the page blob and specifying the size
- Content to be added within 512-byte page boundaries
- Writes to page blobs commit immediately

Blob events



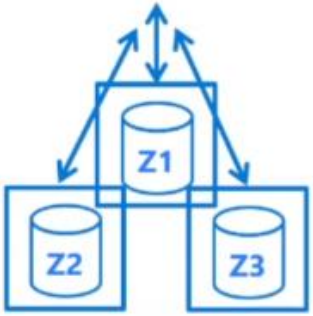
Event grid : A File added / deleted / updated event atabilirsin. Real time integragration.

Storage durability options

Single region

LRS

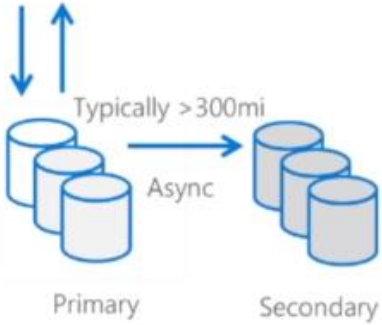
- Three replicas, one region
- Protects against disk, node, rack failures
- Write is acknowledged when all replicas are committed
- Superior to dual-parity RAID



ZRS

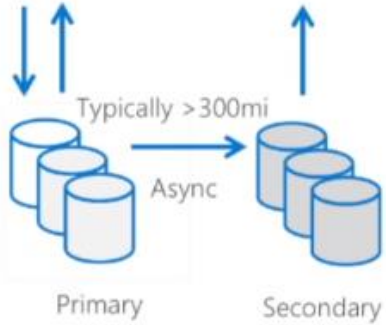
- Three replicas, three zones, one region
- Protects against disk, node, rack, and zone failures
- Synchronous writes to all three zones

Multiple regions



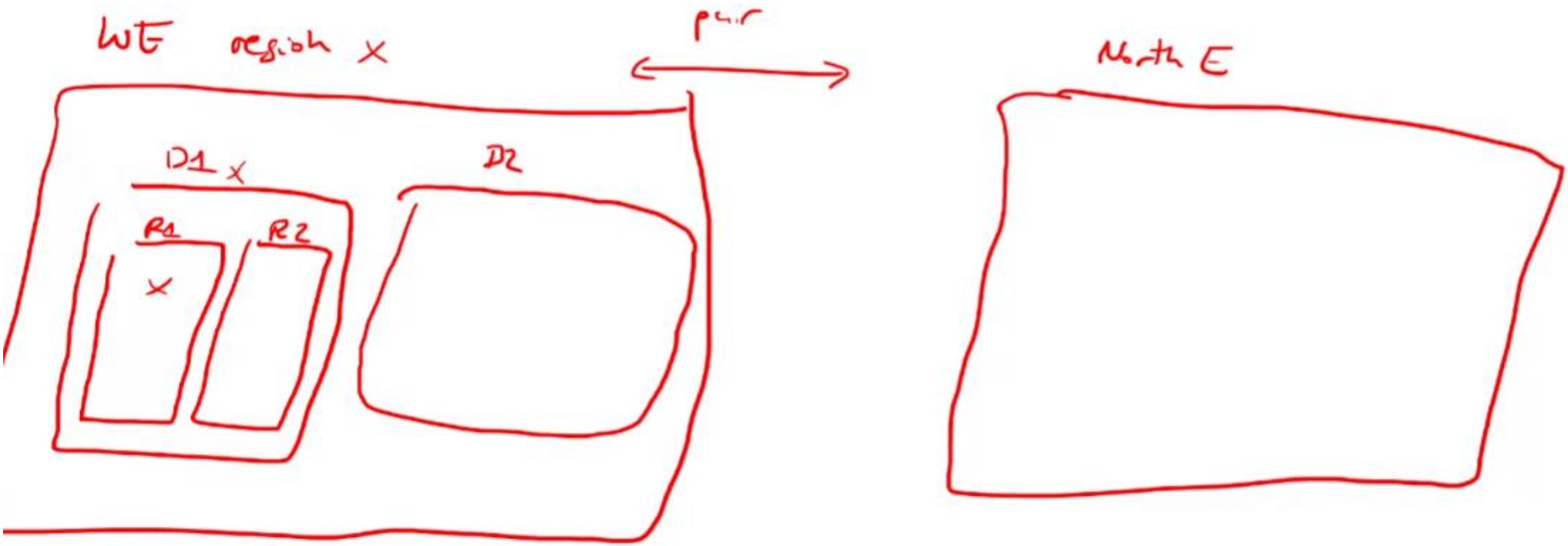
GRS

- Six replicas, two regions (three per region)
- Protects against major regional disasters
- Asynchronous copy to secondary



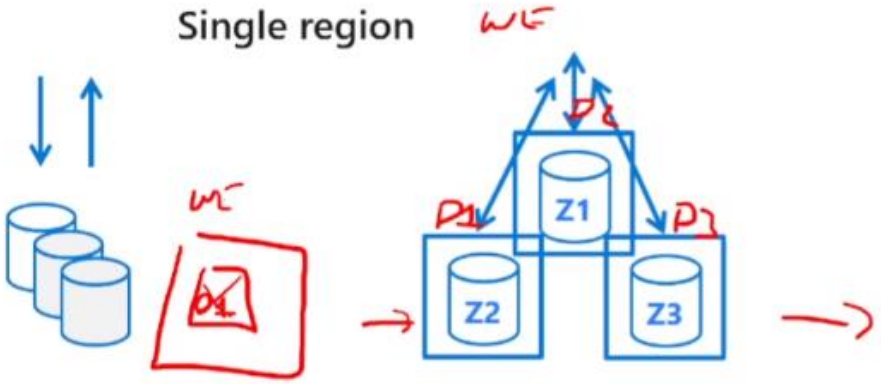
RA-GRS

- GRS + read access to secondary
- Separate secondary endpoint
- Recovery point objective (RPO) delay to secondary can be queried


WE region X
peer
North E
D1 X
D2
R1
R2
X

Storage durability options

Single region



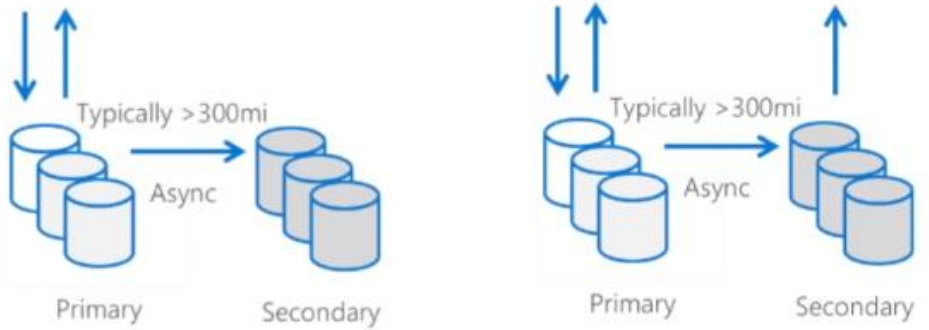
LRS

- Three replicas, one region
- Protects against disk, node, rack failures
- Write is acknowledged when all replicas are committed
- Superior to dual-parity RAID

ZRS

- Three replicas, three zones, one region
- Protects against disk, node, rack, and zone failures
- Synchronous writes to all three zones

Multiple regions



GRS

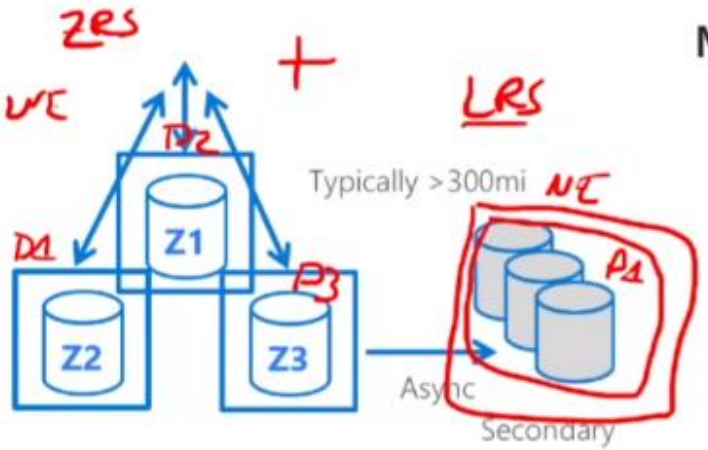
- Six replicas, two regions (three per region)
- Protects against major regional disasters
- Asynchronous copy to secondary

RA-GRS

- GRS + read access to secondary
- Separate secondary endpoint
- Recovery point objective (RPO) delay to secondary can be queried

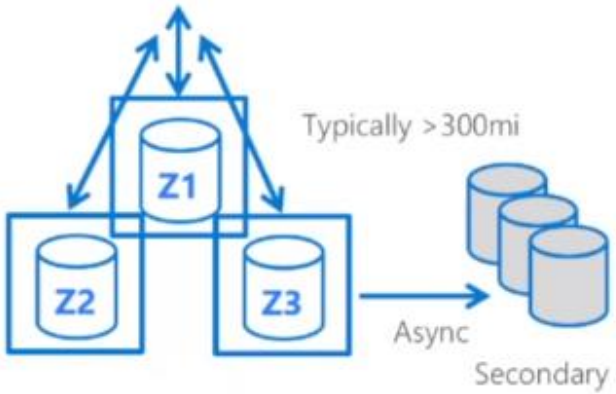
Storage durability options (continued)

Multiple regions



GZRS

- Six replicas, 3+1 zones, two regions
- Protects against disk, node, rack, zone, and region failures
- Synchronous writes to all three zones and asynchronous copy to secondary

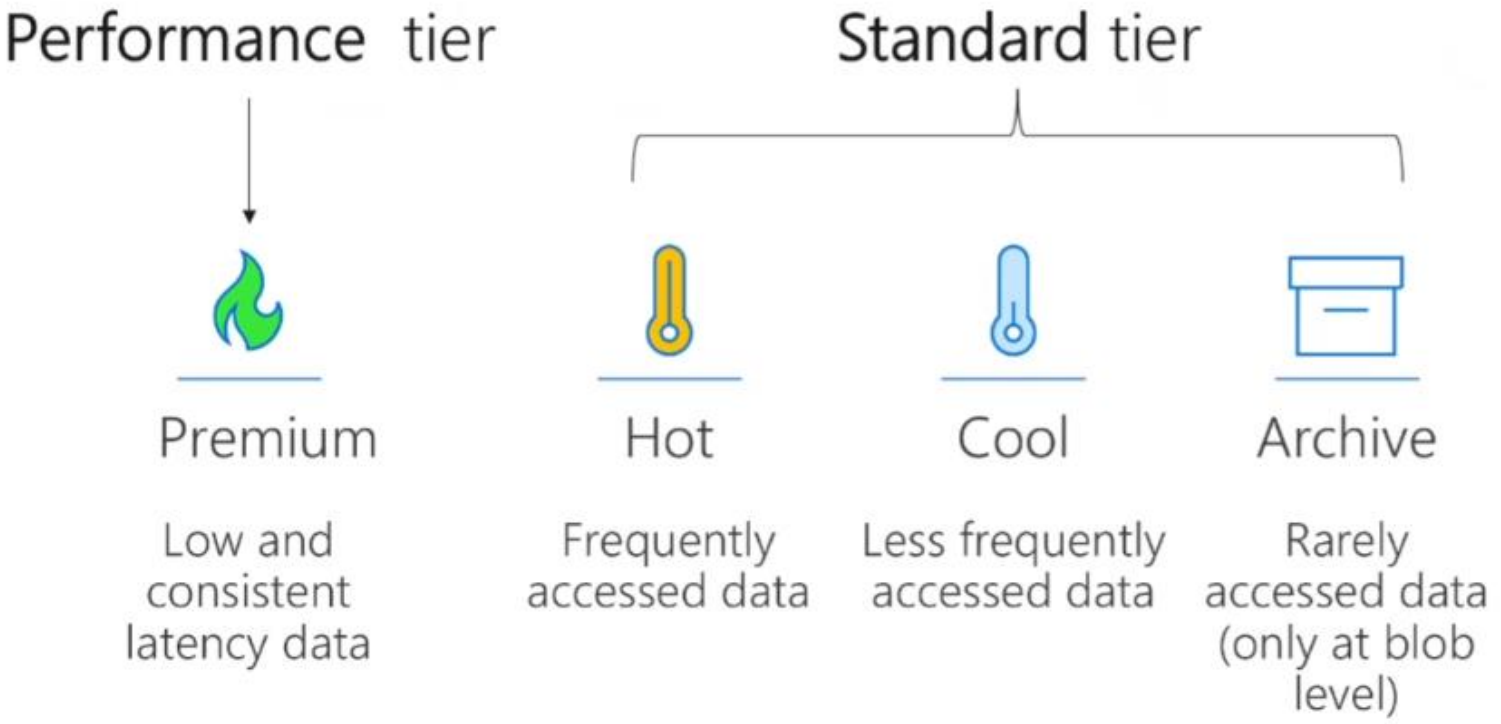


RA-GZRS

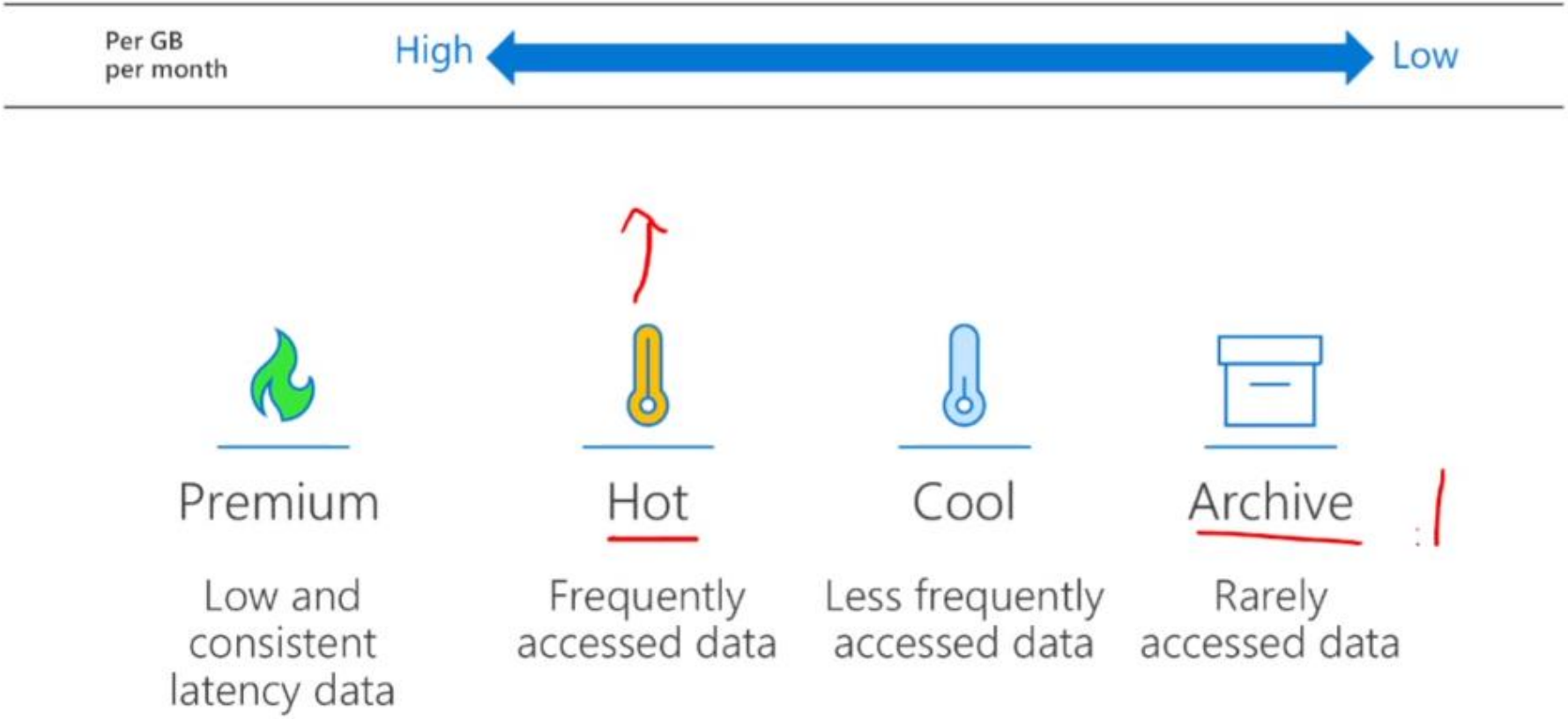
- GZRS + read access to secondary
- Separate secondary endpoint
- RPO delay to secondary can be queried

Storage tiers

You can use storage tiers to tune performance and cost to a ratio that's ideal for your solution

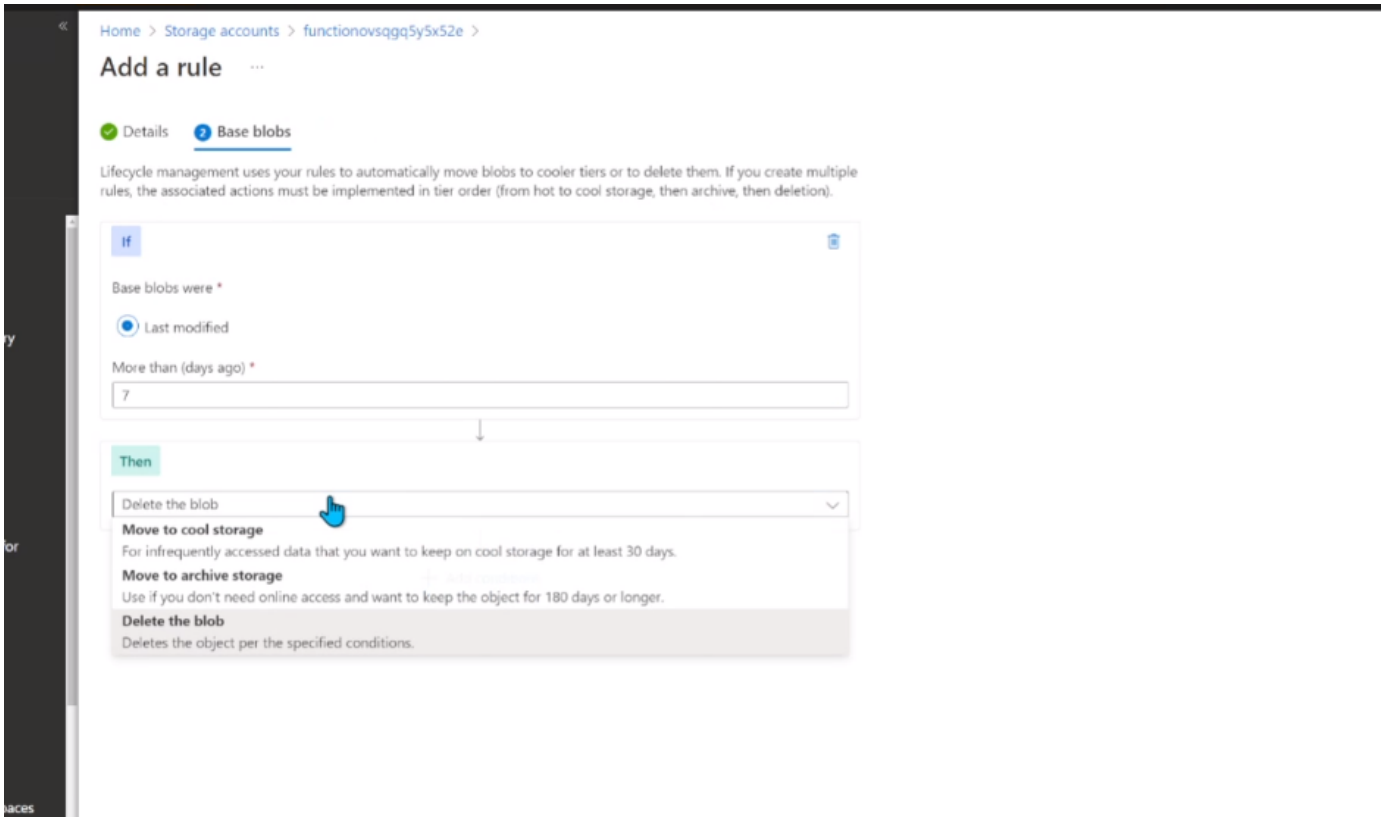


Storage tier pricing



Süper bir özellik.

https://docs.microsoft.com/en-us/azure/storage/blobs/archive-rehydrate-overview



Azure Storage Blobs client library for .NET v12

- Blob storage offers three types of resources:
  - **Storage account**: The `BlobServiceClient` class represents your Azure storage account. Use this class to authorize access to Blob storage using your account access keys.
  - **Container**: The `BlobContainerClient` class allows you to manipulate Azure Storage containers and their blobs.
  - **Blob**: The `BlobClient` class allows you to manipulate Azure Storage blobs.

Blob container properties

Property	Description
ETag	This is a standard HTTP header that gives a value that is **unchanged unless a property of the container is changed**. This value can be **used to implement optimistic concurrency** with the blob containers.
LastModified	This property indicates when the container **was last modified.**
PublicAccess	This property indicates the **level of public access** that is allowed on the container. Valid values include Blob, Container, Off, and Unknown.
HasImmutabilityPolicy	This property indicates whether the container has an immutability policy. An immutability policy will help ensure that blobs are stored for a minimum amount of retention time.
HasLegalHold	This property indicates whether the container has an active legal hold. A legal hold will help ensure that blobs remain unchanged until the hold is removed.

Manage blob properties and metadata in .NET

```
// Create a BlobServiceClient object which will be used to create a container client
BlobServiceClient blobServiceClient = new BlobServiceClient(connectionString);
BlobContainerClient containerClient =
          await blobServiceClient.CreateBlobContainerAsync(containerName);

await containerClient.SetMetadataAsync(metadata);

await containerClient.GetPropertiesAsync();

// Example code to show how to access returned properties and metadata
Console.WriteLine($"Container last modified {properties.Value.LastModified}");
foreach (var metadataItem in properties.Value.Metadata)
{
    Console.WriteLine($"\tKey: {metadataItem.Key}");
    Console.WriteLine($"\tValue: {metadataItem.Value}");
}
```

Lease Blob operation

- Establishes a lock on a blob for write and delete
  - Duration is typically 15 to 60 seconds
  - Optionally, you can establish an infinite lock
- Operation has five modes
  - Acquire
  - Renew
  - Change
  - Release
  - Break (end the lease but prevent other clients from acquiring a new lease)

Control Access

- Azure Active Directory:
  - **Use Azure Active Directory (Azure AD) credentials to authenticate** a user, group, or other identity for access to blob and queue data.
- Shared Key authorization:
  - Use your storage **account access key to construct a connection string** that your application uses at runtime to access Azure Storage.
- Shared access signature
  - A shared access signature is **a token that encapsulates all of the information** needed to authorize a request to Azure Storage on the URL.
    - **Storage resource, the permissions granted, and the interval**

Control Access
- Azure Active Directory:
- Use Azure Active Directory (Azure AD) credentials to authenticate a user, group, or other identity for access to blob and queue data.
- Shared Key authorization:
- Use your storage account access key to construct a connection string that your application uses at ru ntime to access Azure Storage.
- Shared access signature
- A shared access signature is a token that encapsulates all of the information needed to authorize a request to Azure Storage on the URL.
- Storage resource, the permissions granted, and the interval

Control Access

- Azure Active Directory:
  - **Use Azure Active Directory (Azure AD) credentials to authenticate** a user, group, or other identity for access to blob and queue data.
- Shared Key authorization:
  - Use your storage **account access key to construct a connection string** that your application uses at runtime to access Azure Storage.
- Shared access signature
  - A shared access signature is **a token that encapsulates all of the information** needed to authorize a request to Azure Storage on the URL.
    - **Storage resource, the permissions granted, and the interval**

Shared Access Signatures

- Mechanism **to grant access** to storage account objects (blobs, tables) to *clients*, **without exposing account key**
- Enable you to grant specified set of permissions to a client for a specified period of time
- The SAS is a URI, which contains metadata, including a token **and is signed with storage key**
- To access storage resources, client only needs to pass in the SAS
- Applied to **entire storage account or a specific resource** service (blob, queue, table or file)

1000

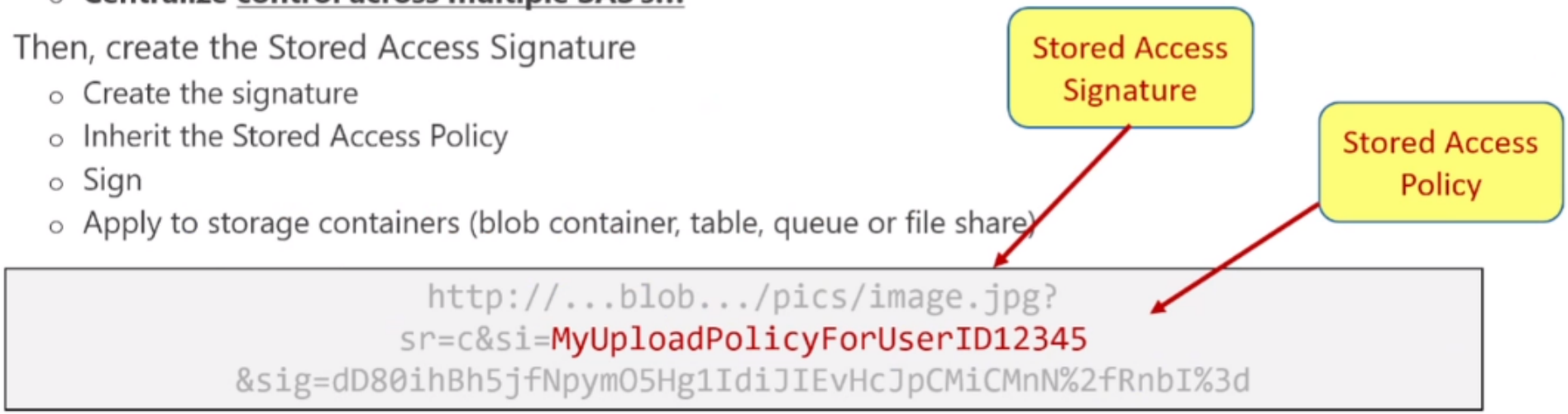
```
https://myaccount.blob.core.windows.net/sascontainer/sasblob.txt?
sv=2015-04-05&st=2015-04-29T22%3A18%3A26Z&se=2015-04-
30T02%3A23%3A26Z&sr=b&sp=rw&sip=168.1.5.60-
168.1.5.70&spr=https&sig=Z%2FRHIX5Xcg0Mq2rqI3OlWTjEg2tYkboXr1P9ZUXDtkk%3D
```

SV	Storage Version
ST	Start Time
SE	Expiration TIme
SR	Storage Resource
SP	Permissions
SIG	Signature

- Two broad approaches:
  - Ad hoc
  - Policy-based

Policy-Based Signatures

- Provide revocable (can remove) permissions to specific users and groups
- Two Parts: (1) The *Stored Access Signature* and (2) the *Stored Access Policy*
- First, create Stored Access Policy…
  - Specify constraints: StartTime, ExpirationTime, and Permissions
  - **Centralize control across multiple SAS's!!!**
- Then, create the Stored Access Signature
  - Create the signature
  - Inherit the Stored Access Policy
  - Sign
  - Apply to storage containers (blob container, table, queue or file share)



Control Access

- Azure Active Directory:
  - **Use Azure Active Directory (Azure AD) credentials to authenticate** a user, group, or other identity for access to blob and queue data.
- Shared Key authorization:
  - Use your storage **account access key to construct a connection string** that your application uses at runtime to access Azure Storage.
- Shared access signature
  - A shared access signature is **a token that encapsulates all of the information** needed to authorize a request to Azure Storage on the URL.
    - **Storage resource, the permissions granted, and the interval**

Module 04: Develop solutions that use Cosmos DB storage

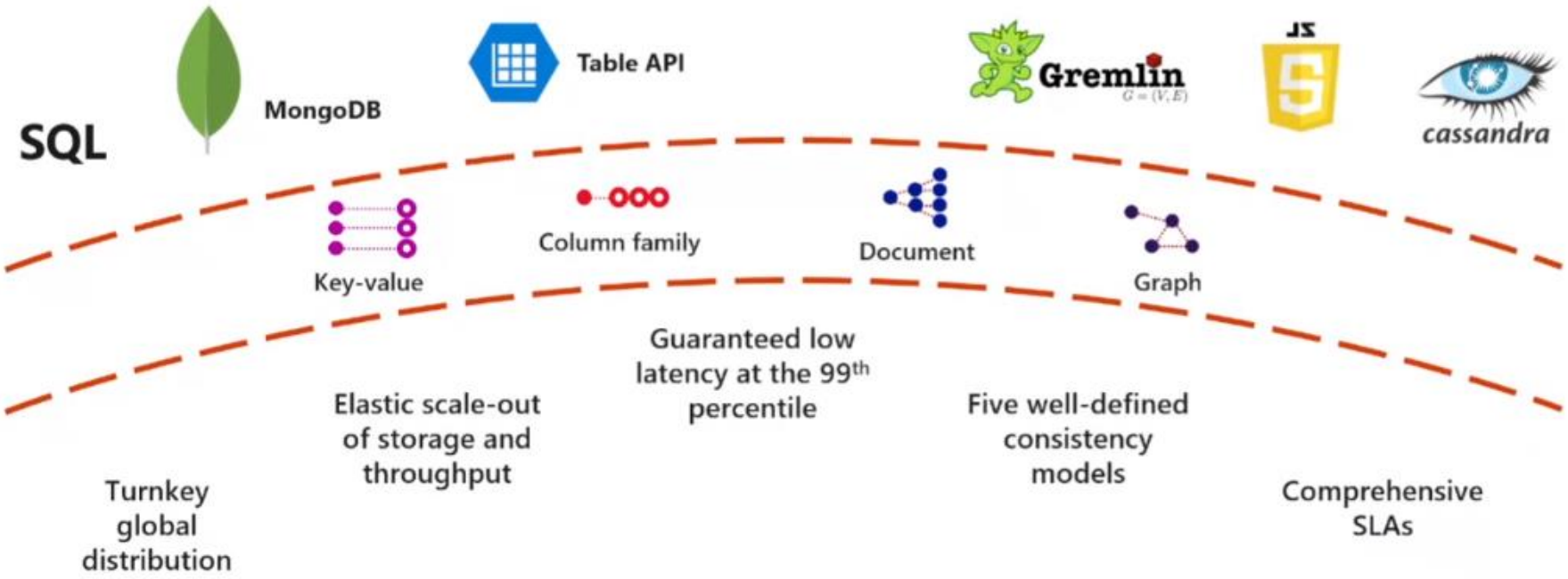
Topics

- Azure Cosmos DB overview

- Azure Cosmos DB data structure

- Create and update documents by using code

Azure Cosmos DB

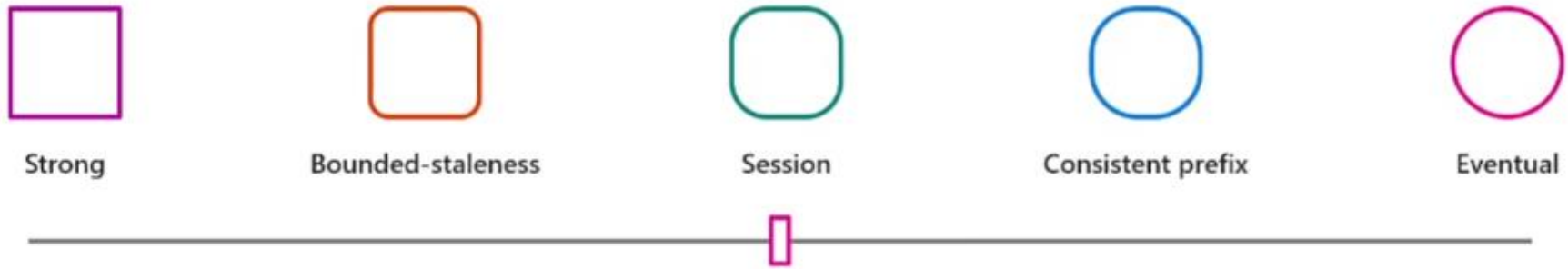


Global Replication

Turnkey global distribution automatically replicates data to other Azure datacenters across the globe without the need to manually write code or build a replication infrastructure

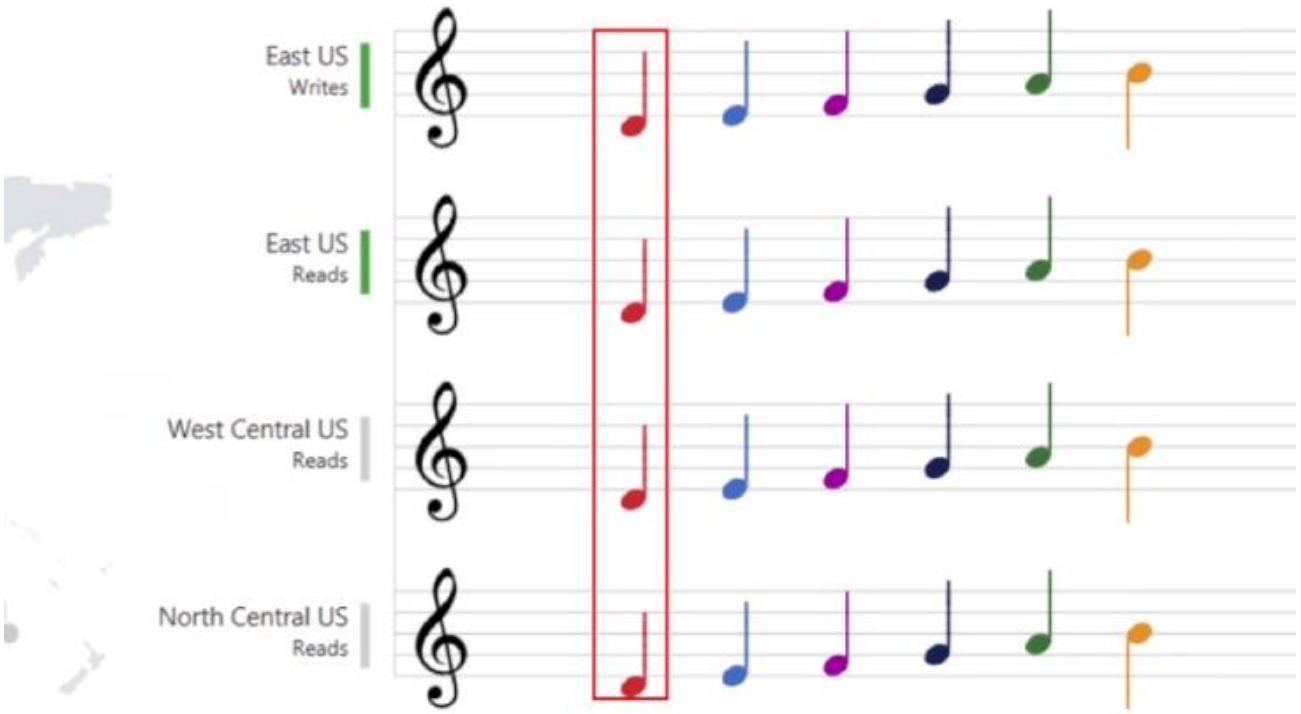
Consistency levels

Azure Cosmos DB provides five consistency levels:

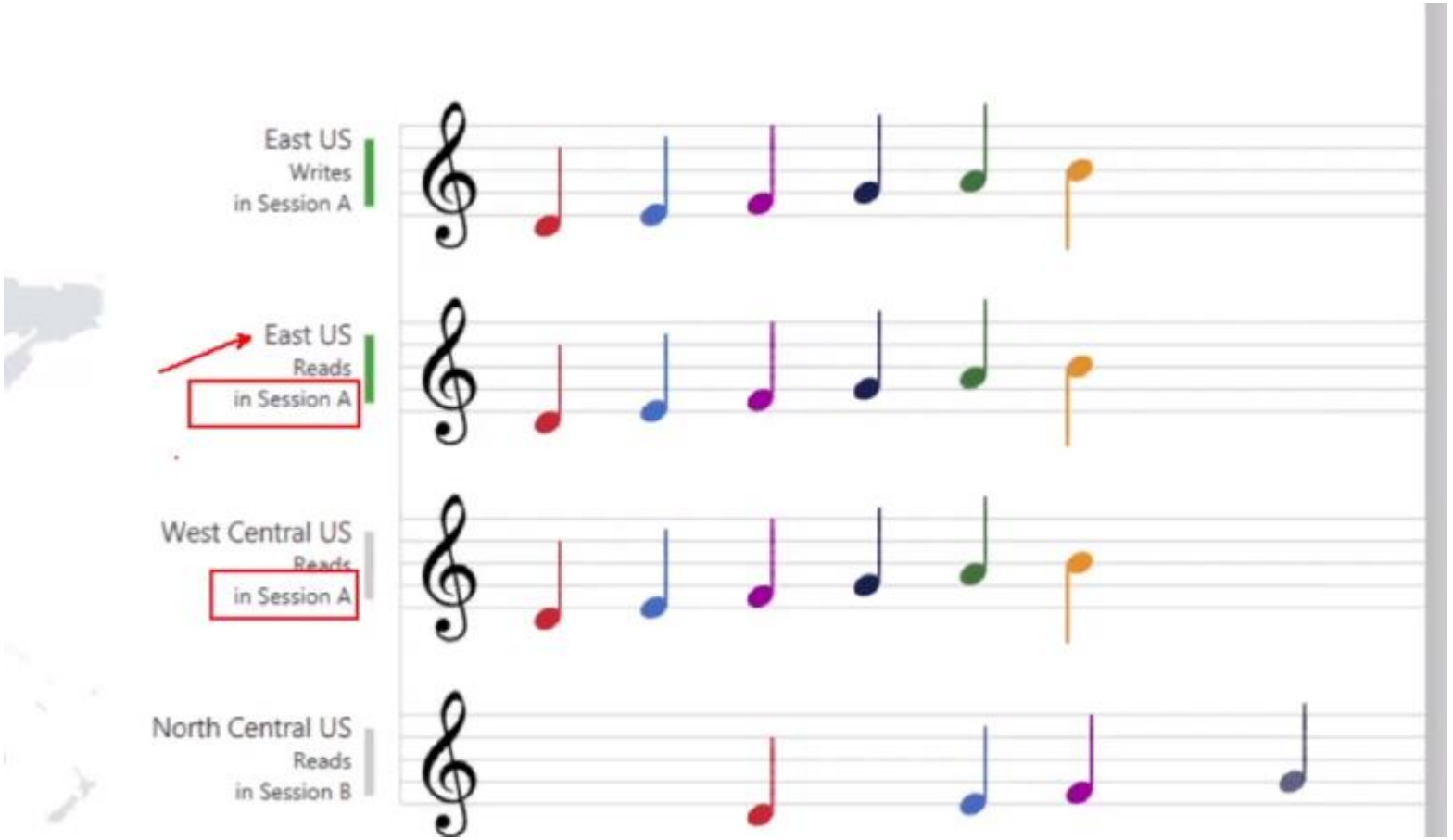


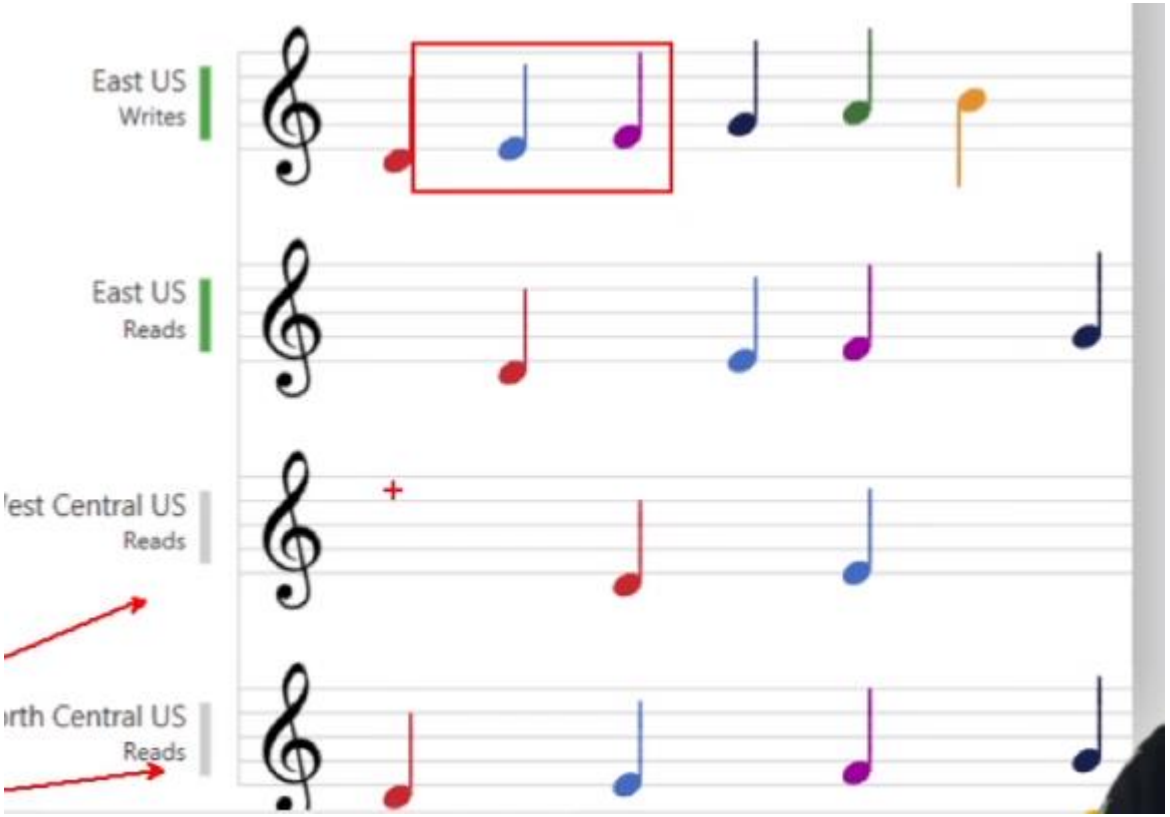
Consistency Level	Description
Strong	When a write operation is performed on your primary database, the write operation is replicated to the replica instances. The **write operation is committed (and visible)** on the primary only after it has been **committed and confirmed by all replicas.**
Bounded Staleness	This level is similar to the Strong level with the major difference that you can configure **how stale documents can be within replicas**. Staleness refers to **the quantity of time (or the version count) a replica document can be behind** the primary document.
Session	This level guarantees that **all read and write operations are consistent within a user session**. Within the user session, all reads and writes are monotonic and guaranteed to be **consistent across primary and replica instances.**
Consistent Prefix	This level has loose consistency but **guarantees that when updates show up in replicas, they will show up in the correct order** (that is, as prefixes of other updates) without any gaps.
Eventual	This level has the loosest consistency and essentially **commits any write operation against the primary immediately**. **Replica** transactions are **asynchronously handled and will eventually (over time) be consistent** with the primary. This tier has the **best performance**, because the primary database does not need to wait for replicas to commit to finalize it's transactions.

https://docs.microsoft.com/en-us/azure/cosmos-db/consistency-levels

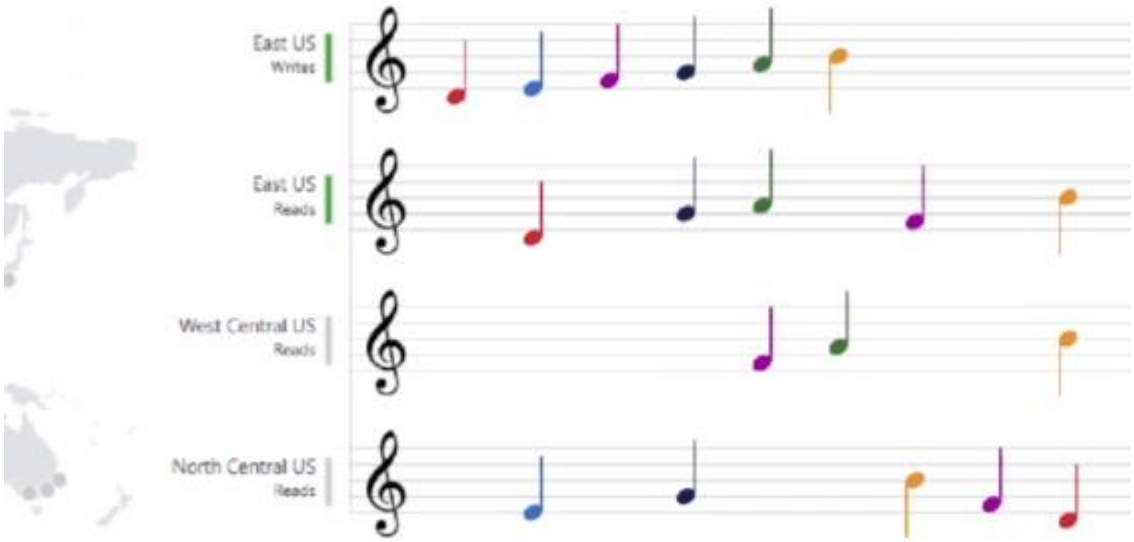


Bounded session’da: kayıt saatleri aynı


East US
Writes
in Session A
East US
Reads
in Session A
West Central US
Reads
in Session A
North Central US
Reads
in Session B


East US
Writes
East US
Reads
est Central US
Reads
rth Central US
Reads

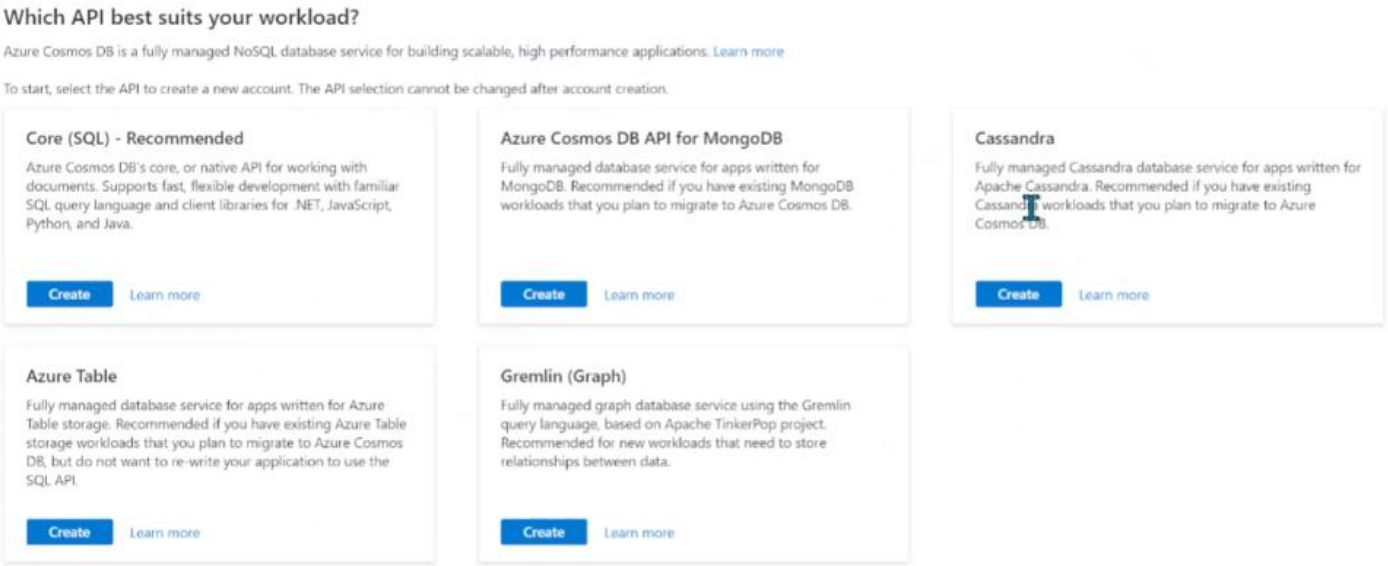
Sonuncu: tweeter, facebook, kim önce yorum attı bunun yedek kayıtlarda aynı olmasına gerek yok.



Yaratırken seçiyorsun

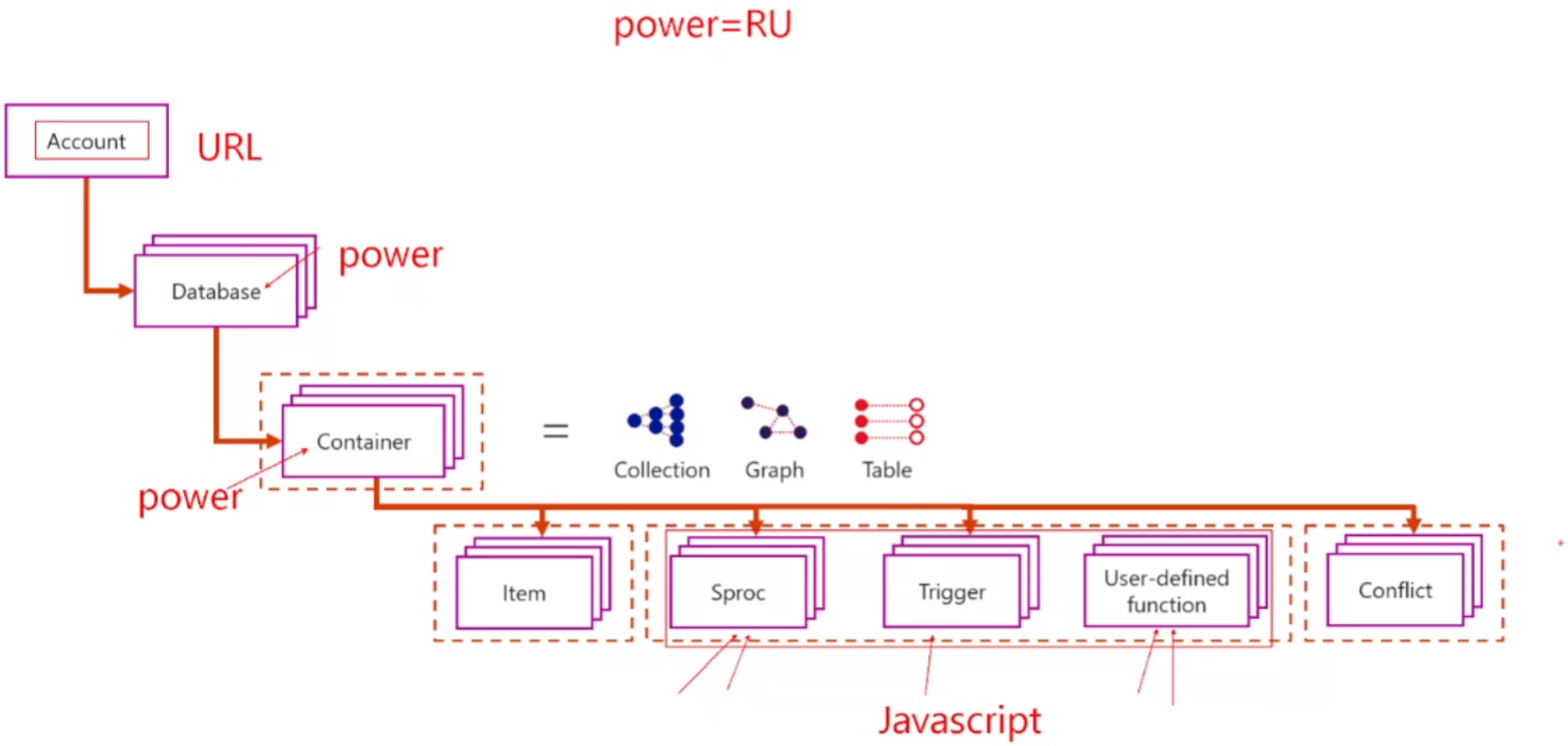
APIs

- MongoDB API
  - Acts as a massively scalable **MongoDB service powered by the Azure Cosmos** DB platform
  - **Compatible with existing MongoDB** libraries, drivers, tools, and applications
- Table API
  - A **key-value database service** built to provide **premium capabilities to existing Azure Table storage applications** without making any app changes
- Gremlin API
  - A fully managed, horizontally **scalable graph database service**
  - **Easy-to-build** and run applications that work with highly connected datasets supporting **Open Graph APIs** (based on the Apache TinkerPop specification, Apache Gremlin)

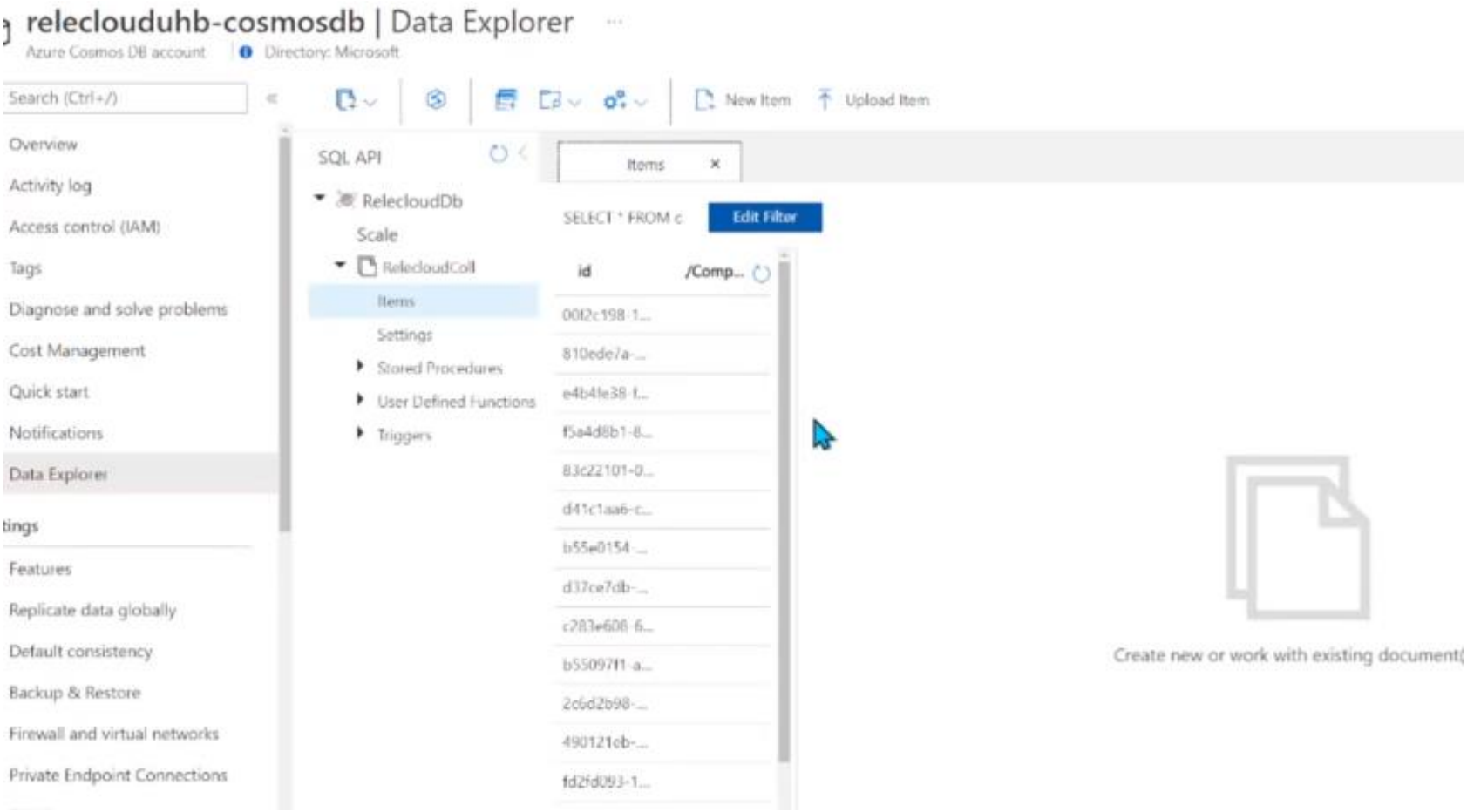


Lesson 02: Azure Cosmos DB data structure

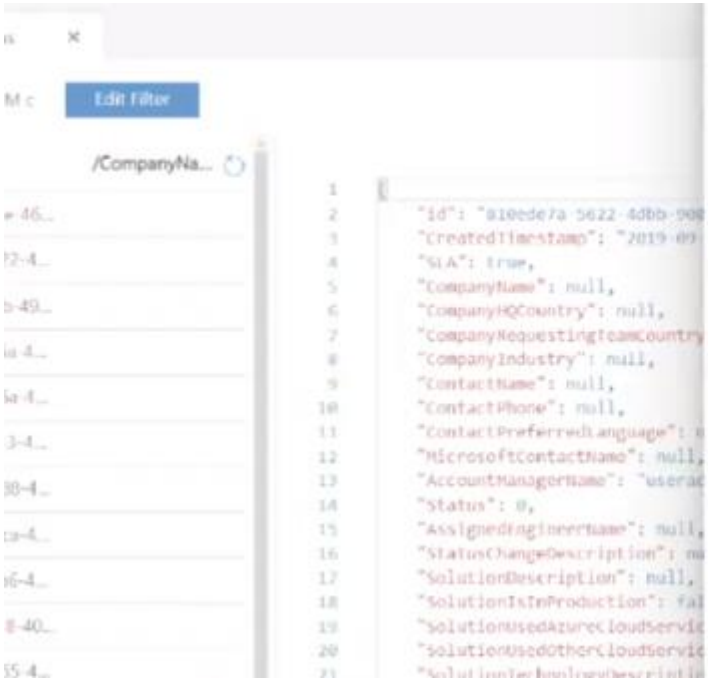
Resource hierarchy

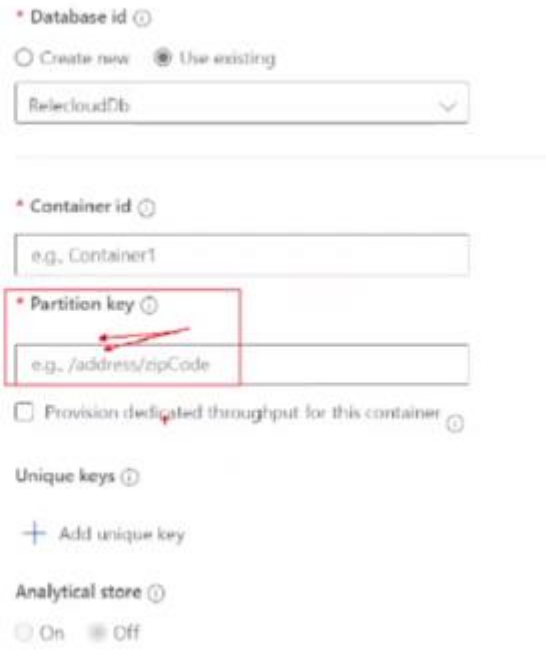


Resource	Description
Account	A set of databases
Database	Logical container for containers that can (optionally) share throughput across the containers
Collection (container)	A group of Items and programmatic resources usually related in some way
Document (item)	An arbitrary unit of content In many cases, this would be a JSON document
Stored procedure (sproc)	Application logic written in JavaScript **executed** within the database engine **as a transaction**
Trigger	Application logic written in JavaScript **executed before or after** either an insert, replace, or delete operation
User-defined function	Application logic written in JavaScript to **extend** the SQL API query language

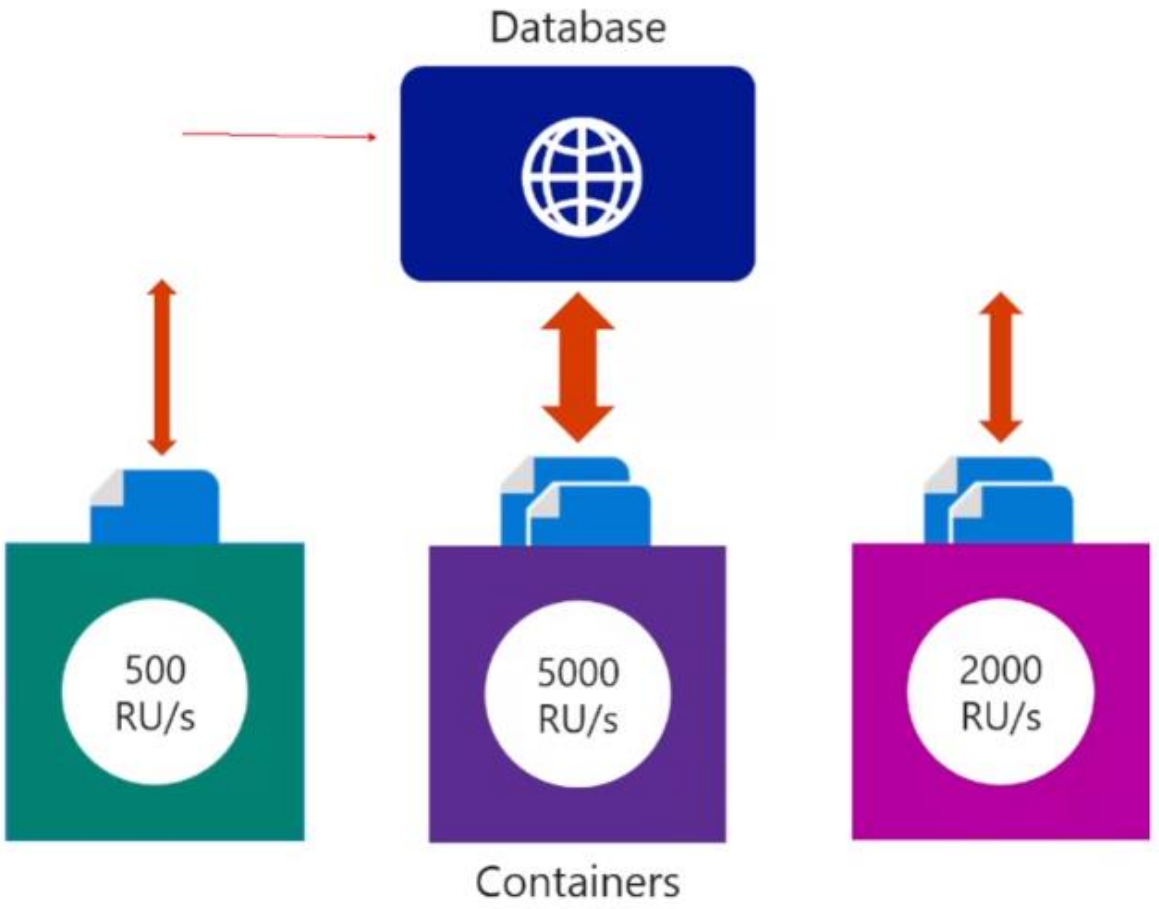


Bu tanım çok önemli

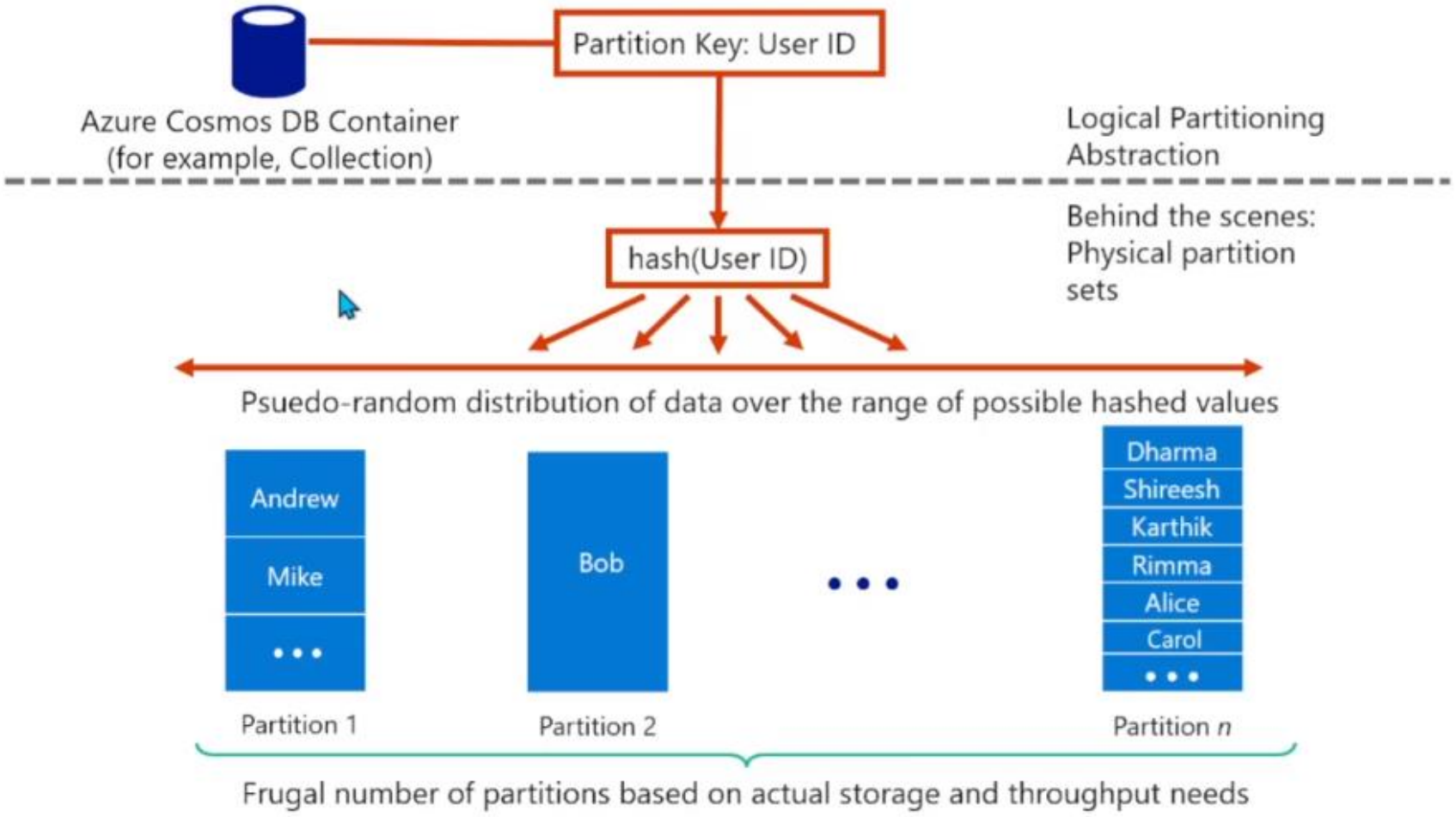

Edit Filter
/CompanyNa...


* Database id
Create new
Use existing
RelecloudDb
* Container id
e.g., Container1
* Partition key
e.g., /address/zipCode
Provision dedicated throughput for this container
Unique keys
Add unique key
Analytical store
On
Off

Containers

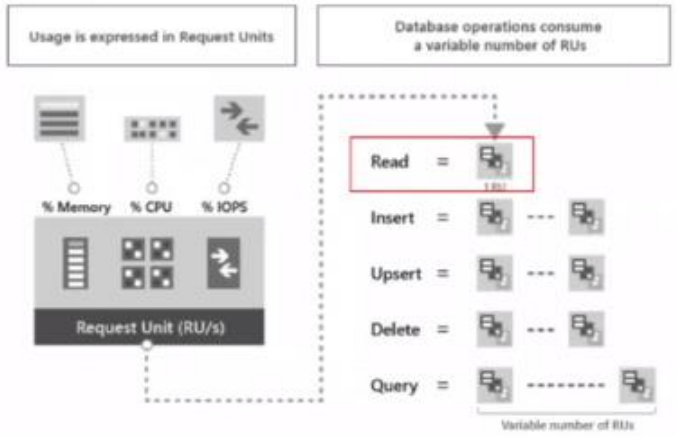


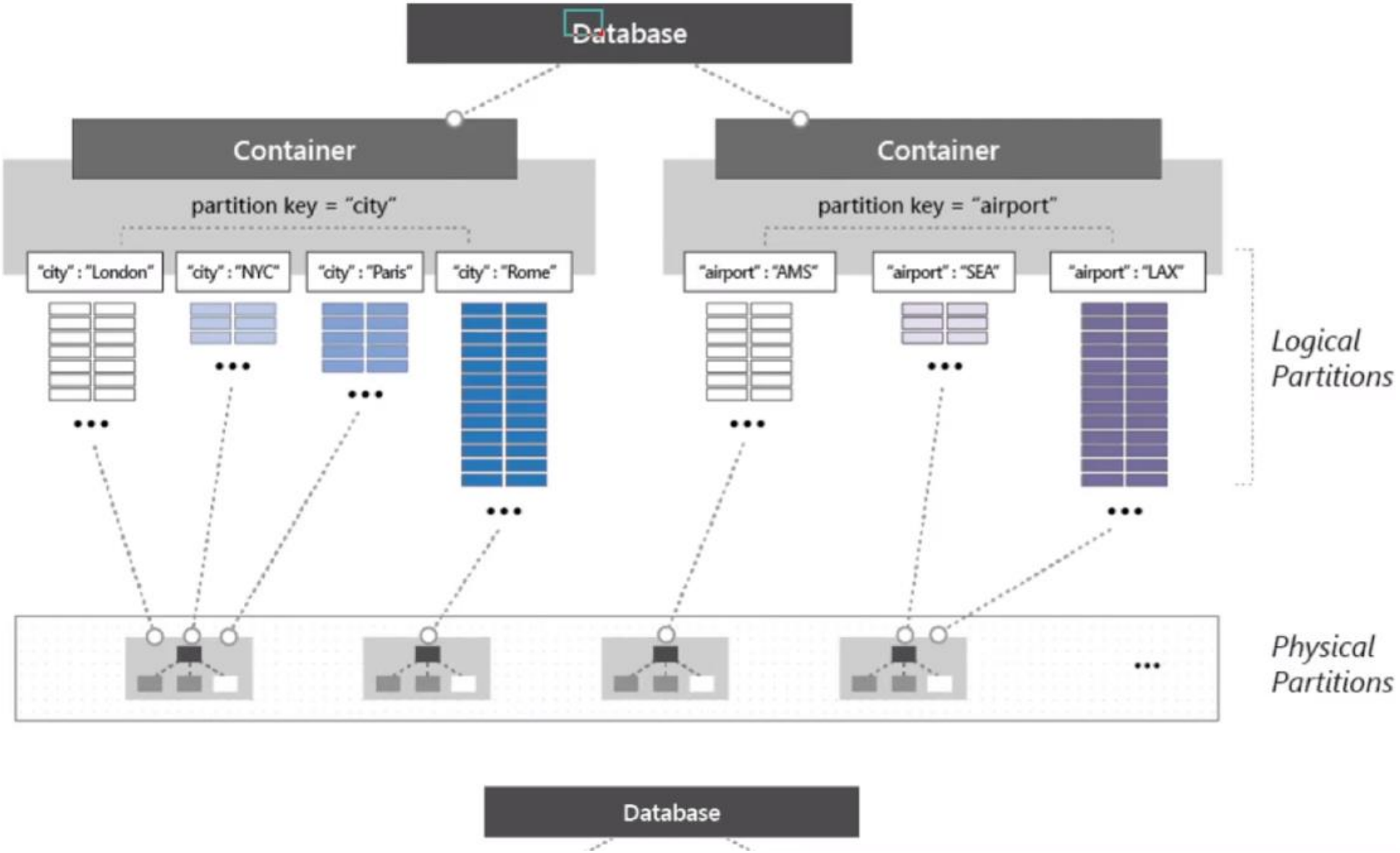
Partitioning implementation



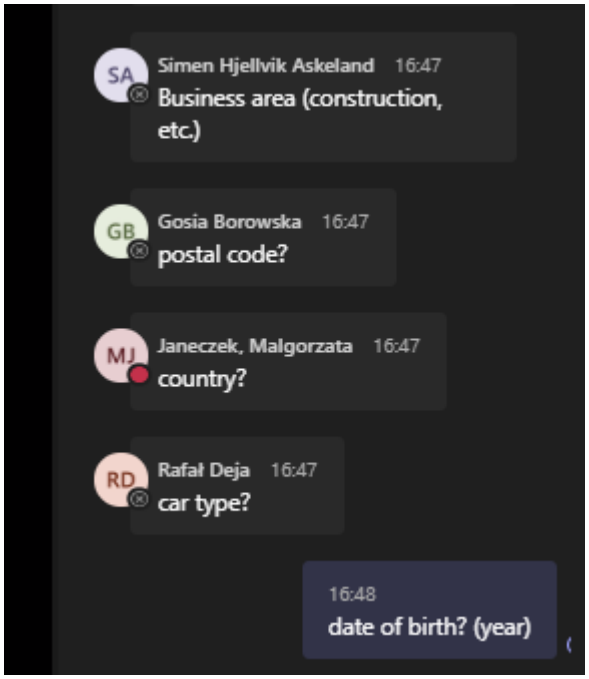
Cosmos DB Throughput

- **Pay for the throughput you provision and the storage** you consume on an hourly basis.
- Each database **operation consumes system resources** based on the complexity of the operation.
- Database operations is **normalized by *Request Units*** (or RUs, for short).
    - The cost to read a 1 KB item is 1 Request Unit (or 1 RU). A minimum of 10 RU/s is required to store each 1 GB of data.




Database
Container
partition key = "city"
"city" : "London"
"city" : "NYC"
"city" : "Paris"
"city" : "Rome"
Container
partition key = "airport"
"airport" : "AMS"
"airport" : "SEA"
"airport" : "LAX"
Logical Partitions
Physical Partitions
Database

Partition’ı iyi planlamak gerekiyor. Daha işin başında.. cinsiyet olmaz mesela sadece 2 ye böler. Dogum yılı olabilir belki..



“Azure cosmos db capasity calculator” diye bir şey var.

https://devblogs.microsoft.com/cosmosdb/serverless-preview/

Accessing a collection/container by using .NET

```
CosmosClient client = new CosmosClient(endpoint, key);
Database database = client.GetDatabase(databaseName);
string collectionName = "ExampleCollection";
```

```
Container container = database
    .GetContainer(collectionName);
```

Reference existing container

OR

```
Container container = await database
    .CreateContainerIfNotExistsAsync(
        containerName, partitionKey,
        throughput: 400
    );
```

Create new container

Creating documents by using .NET

```
// Get container reference
CosmosClient client = new CosmosClient(endpoint, key);
Container container = client.GetContainer(databaseName, collectionName);

// create anonymous type in .NET
Product orangeSoda = new Product {
    id = "7cc3212d-0e2c-4a13-b348-f2d879c43342",
    name = "Orange Soda", group = "Beverages",
    diet = false, price = 1.50m, quantity = 2000
};

// Upload document
Product item = await container.CreateItemAsync(orangeSoda);

Product item = await container.UpsertItemAsync(orangeSoda);
```

Create new document

Create or replace document

Querying documents by using .NET

```
// Get container reference
CosmosClient client = new CosmosClient(endpoint, key);
Container container = client.GetContainer(databaseName, collectionName);

// Use SQL query language
FeedIterator<Product> iterator = container.GetItemQueryIterator<Product>(
    "SELECT * FROM products p WHERE p.diet = false"
);

// Iterate over results
while (iterator.HasMoreResults)
{
    FeedResponse<Product> batch = await iterator.ReadNextAsync();
    foreach(Product item in batch)
    { }
}
```

Check for new batch of results

Get next batch of results

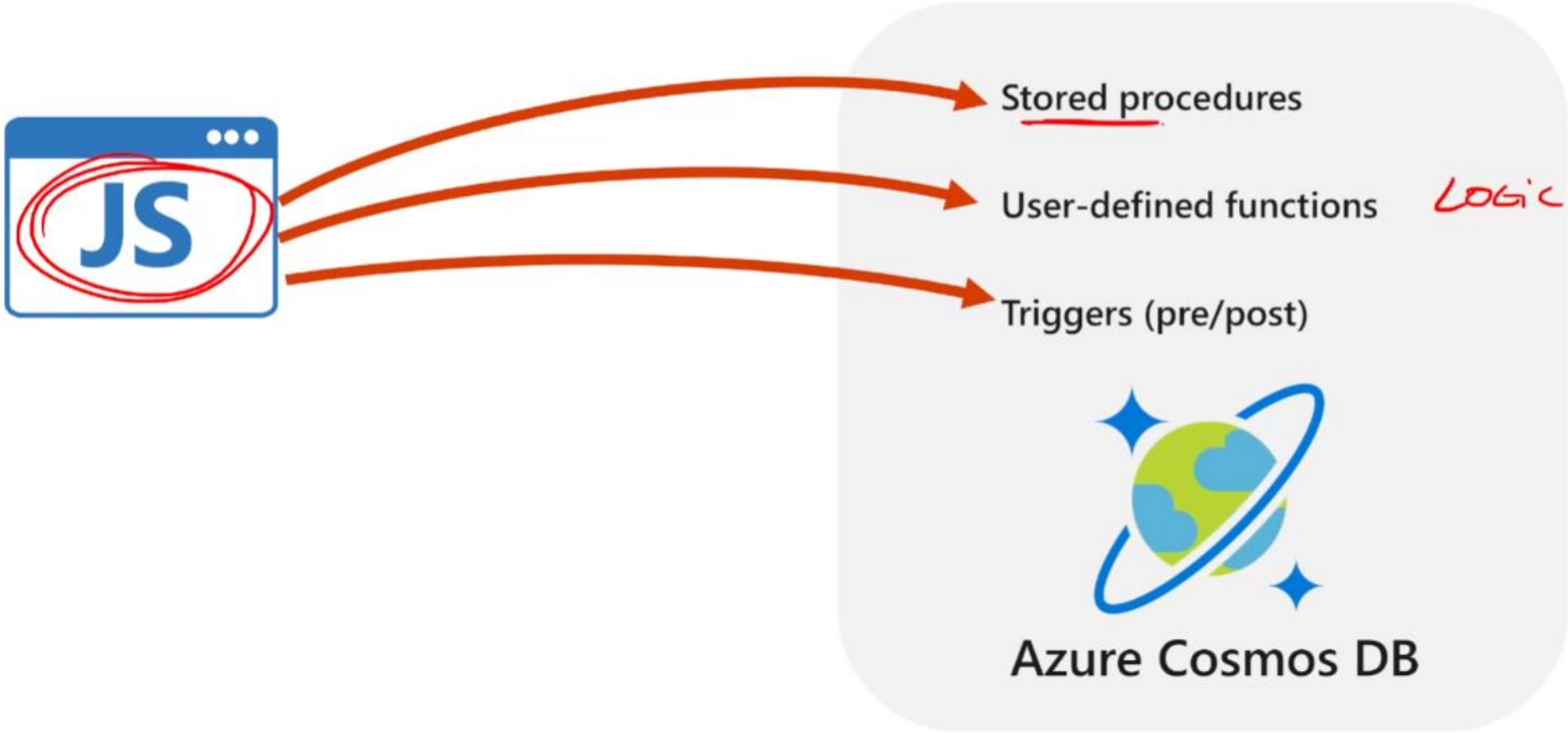
Querying documents by using .NET (continued)

```
// Get container reference
CosmosClient client = new CosmosClient(endpoint, key);
Container container = client.GetContainer(databaseName, collectionName);

// Use LINQ query language
FeedIterator<Product> iterator = container.GetItemLinqQueryable<Product>()
    .Where(p => !p.diet)
    .ToFeedIterator();

// Iterate over results
while (iterator.HasMoreResults)
{
    FeedResponse<Product> batch = await iterator.ReadNextAsync();
    foreach(Product item in batch)
    { }
}
```

Translate LINQ expression to SQL

https://github.com/SaadMarwan/Time-Series-Data-in-CosmosDb-API-for-MongoDB

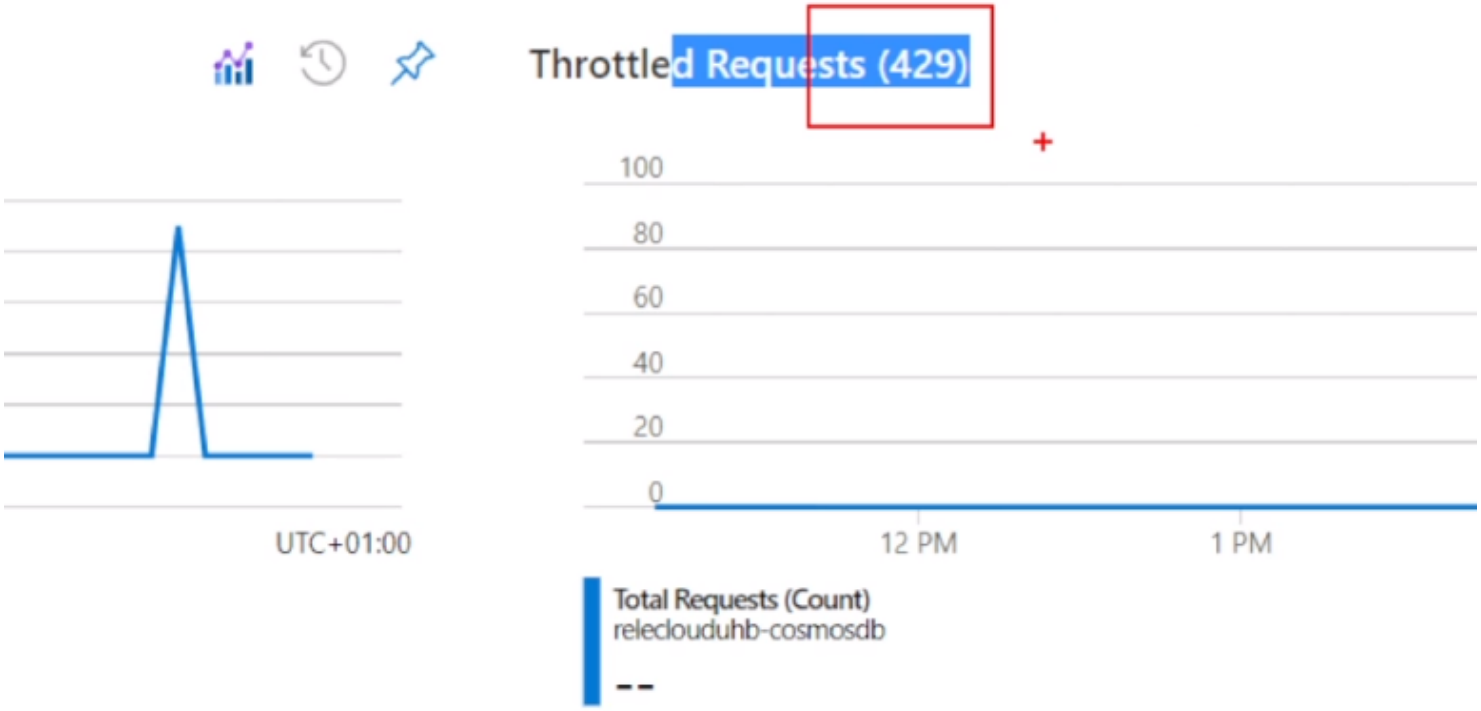
JavaScript and Azure Cosmos DB



Controlling concurrency in .NET

```
try
{
    var ac = new AccessCondition { Condition = readDoc.ETag, Type =
        AccessConditionType.IfMatch };
    await client.ReplaceDocumentAsync(readDoc, new RequestOptions {
        AccessCondition = ac });
}
catch (DocumentClientException dce)
{
    if (dce.StatusCode == HttpStatusCode.PreconditionFailed)
    {
        Console.WriteLine("Another process has updated the record");
    }
}
```

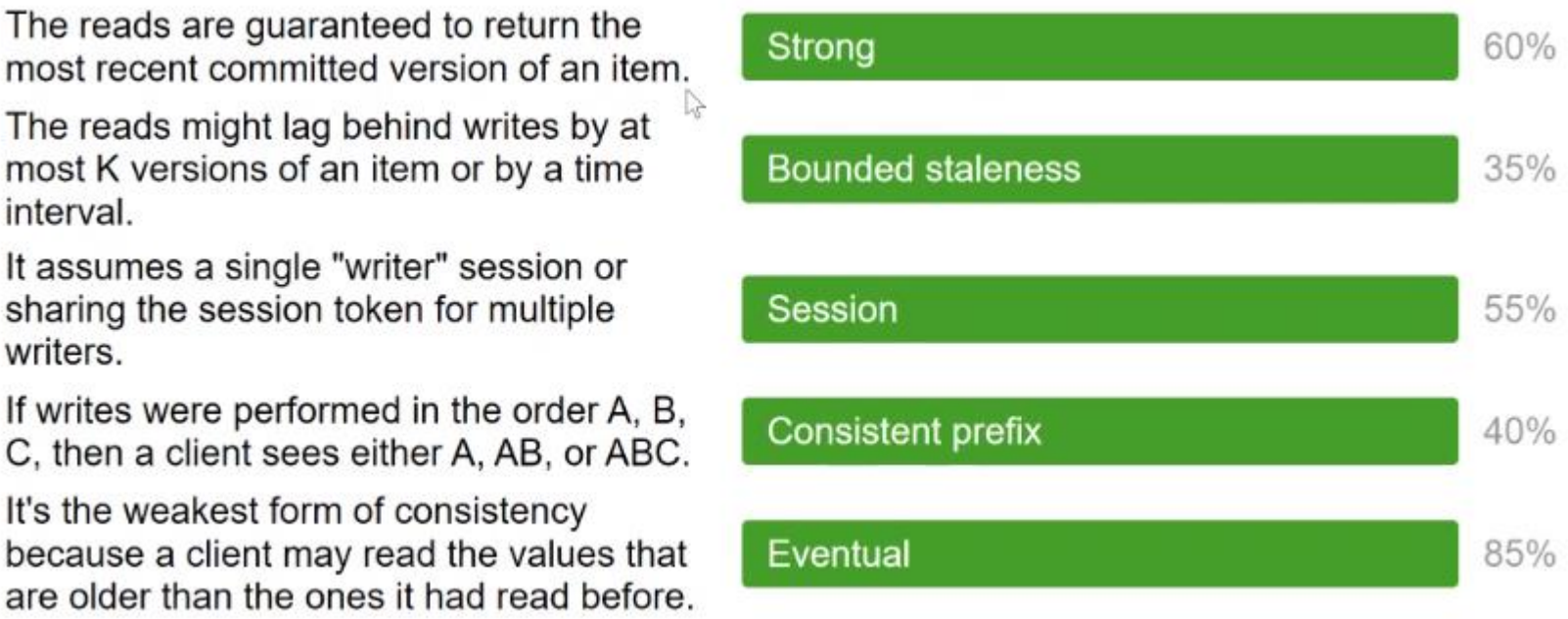
http 429 -> Cosmos DB -> RU kapasiteni aşmışsın demektir.



Localde çalışmak için…

https://docs.microsoft.com/en-us/azure/cosmos-db/local-emulator?tabs=ssl-netstd21

Özet sorular



https://docs.microsoft.com/en-us/azure/azure-functions/storage-considerations

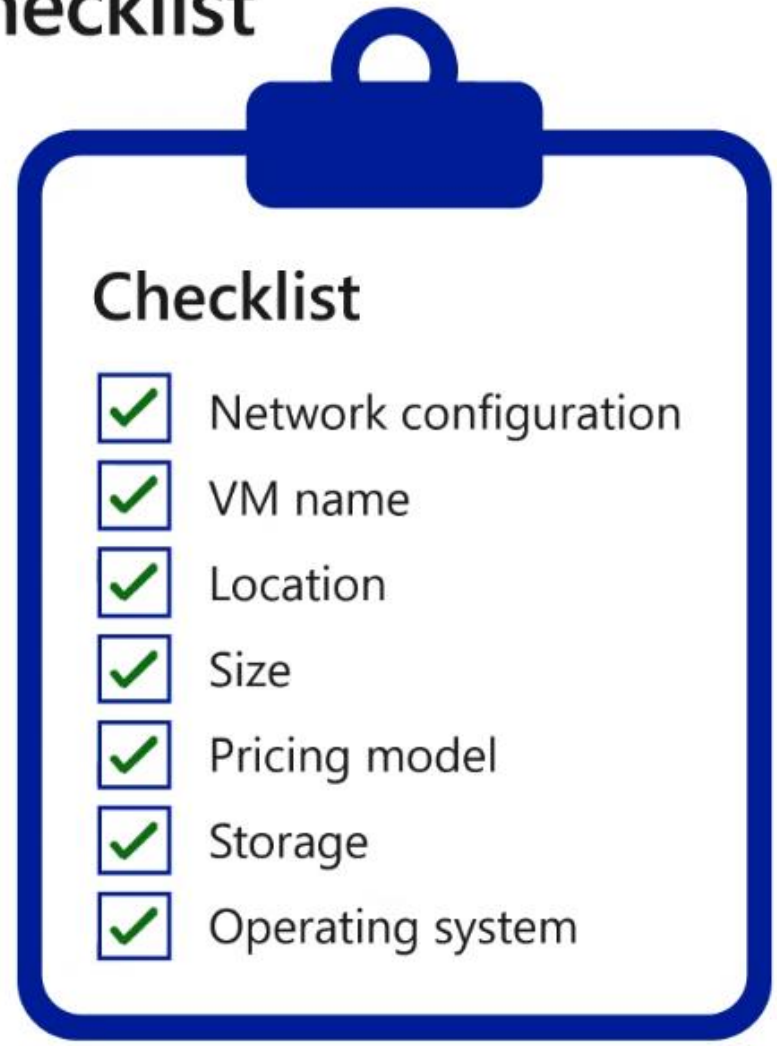
3. Gün

Topics

- Provisioning VMs in Azure
- Create and deploy Azure Resource Manager templates
- Create container images for solutions
- Publish a container image to Azure Container Registry
- Create and run container images in Azure Container Instances

Azure virtual machine creation checklist

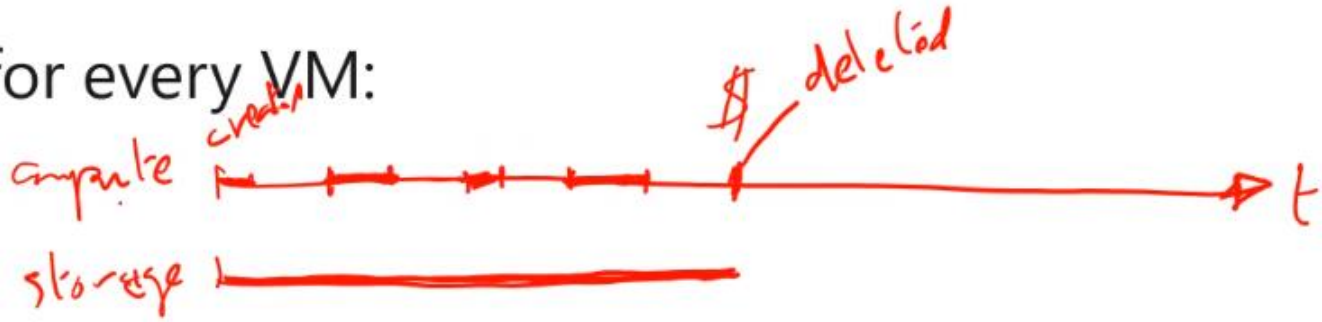
- Before you create a VM, you should consider the following:



Element	Example	Notes
Environment	dev, prod, QA	Identifies the environment for the resource
Location	uw (US West), ue (US East)	Identifies the region into which the resource is deployed
Instance	01, 02	For resources that have more than one named instance (such as web servers)
Product or Service	service	Identifies the product, application, or service that the resource supports
Role	sql, web, messaging	Identifies the role of the associated resource

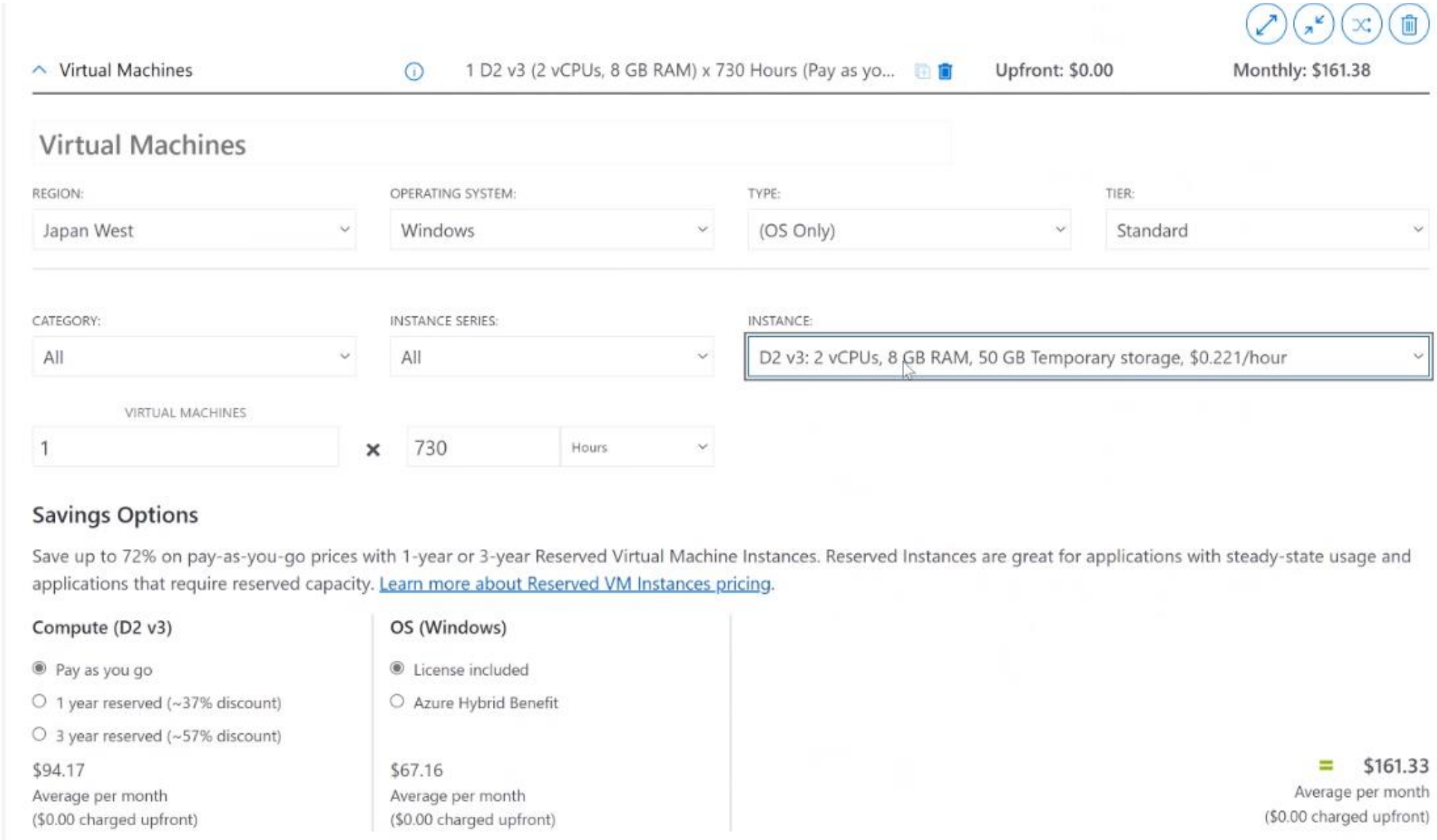
VM pricing models

- Two primary costs for every VM:
  - **Storage**
  - **Compute**



- There are two payment options for compute costs:
  - **Pay as you go**
  - **Reserved instances**

[Pricing calculator | Microsoft Azure]

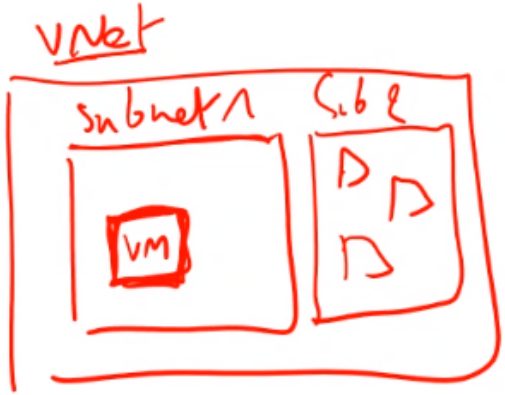


VM storage options

- Virtual disks can be backed by either **Standard** or **Premium Storage** accounts
  - Azure Premium Storage leverages solid-state drives (SSDs) to enable high performance and low latency for VMs running I/O-intensive workloads
- You can choose either **unmanaged** disks or **managed** disks

Create an Azure VM by using PowerShell

```
Connect-AzAccount

New-AzResourceGroup -Name "myResourceGroup" -Location EastUS

New-AzVM `
    -ResourceGroupName "myResourceGroup" `
    -Name "myVM" `
    -Location "East US" `
    -VirtualNetworkName "myVnet" `
    -SubnetName "mySubnet" `
    -SecurityGroupName "myNetworkSecurityGroup" `
    -PublicIpAddressName "myPublicIpAddress" `
    -OpenPorts 80,3389
```

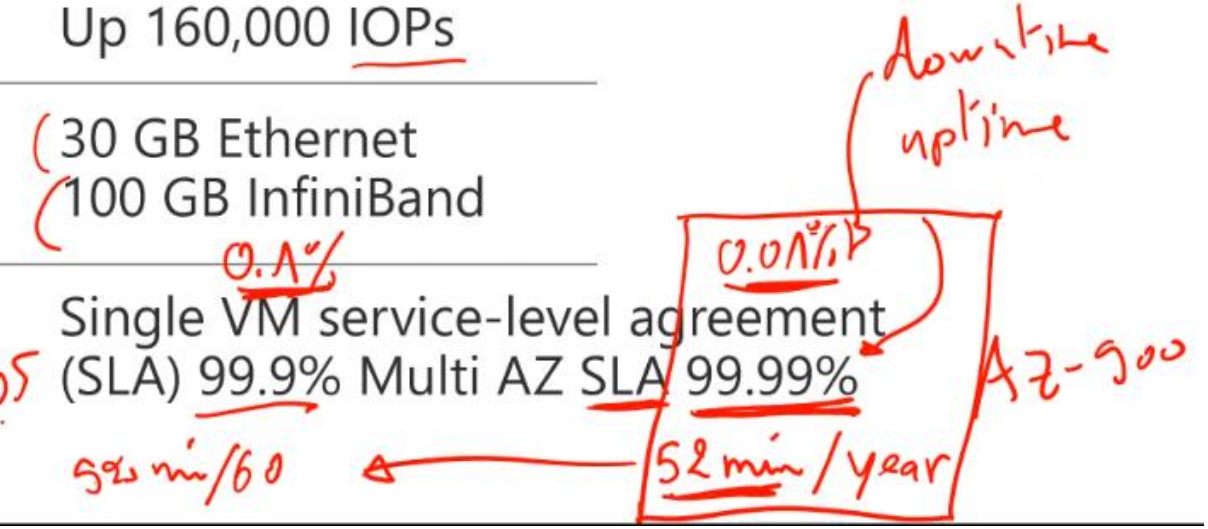


VM configuration options

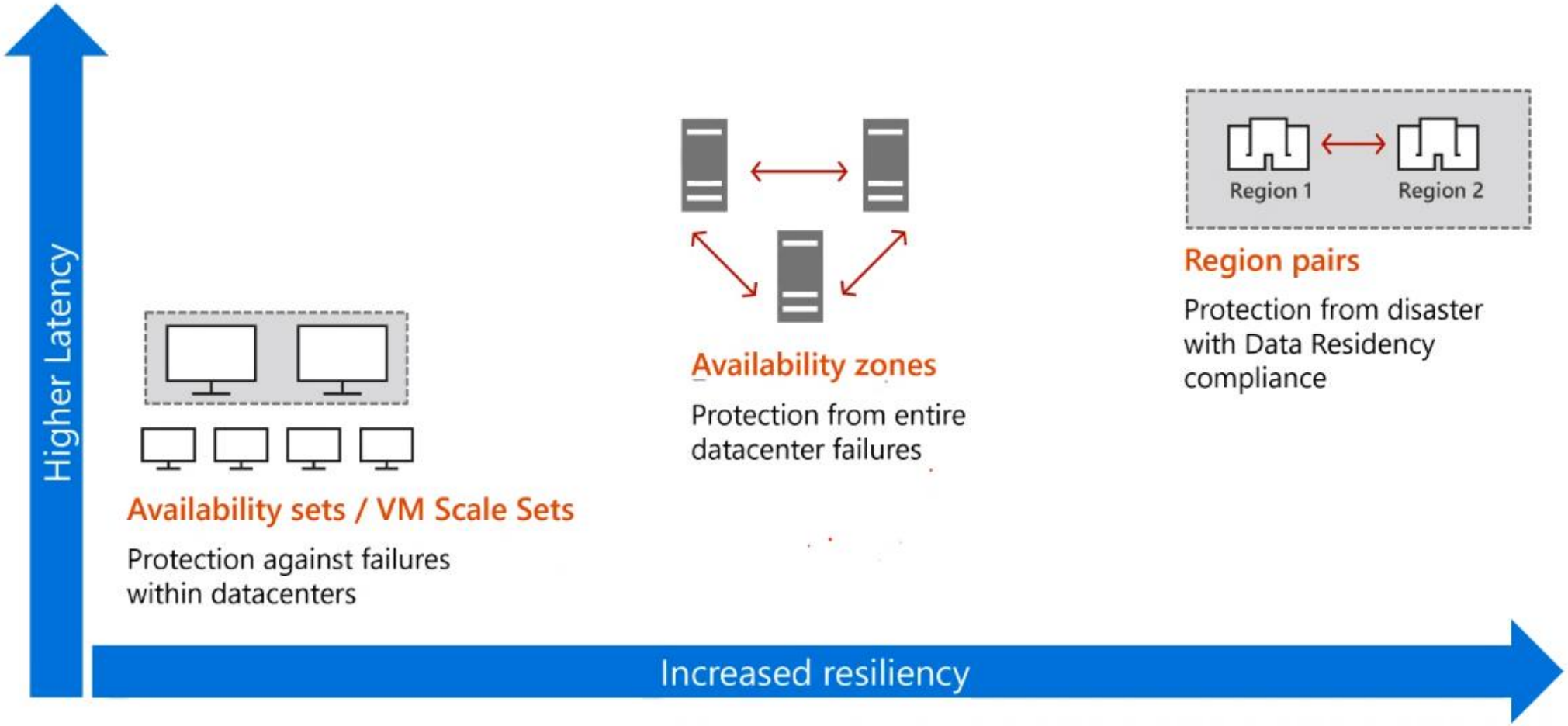
Computational performance	1 virtual CPU (vCPU) - 128 vCPUs
Memory	1 gibibyte (GiB) - 4 tebibyte (TiB)
Disk storage	4GiB - 64TiB Up 160,000 IOPs
Networking	30 GB Ethernet 100 GB InfiniBand
Availability	Single VM service-level agreement (SLA) 99.9% Multi AZ SLA 99.99%

VM configuration options

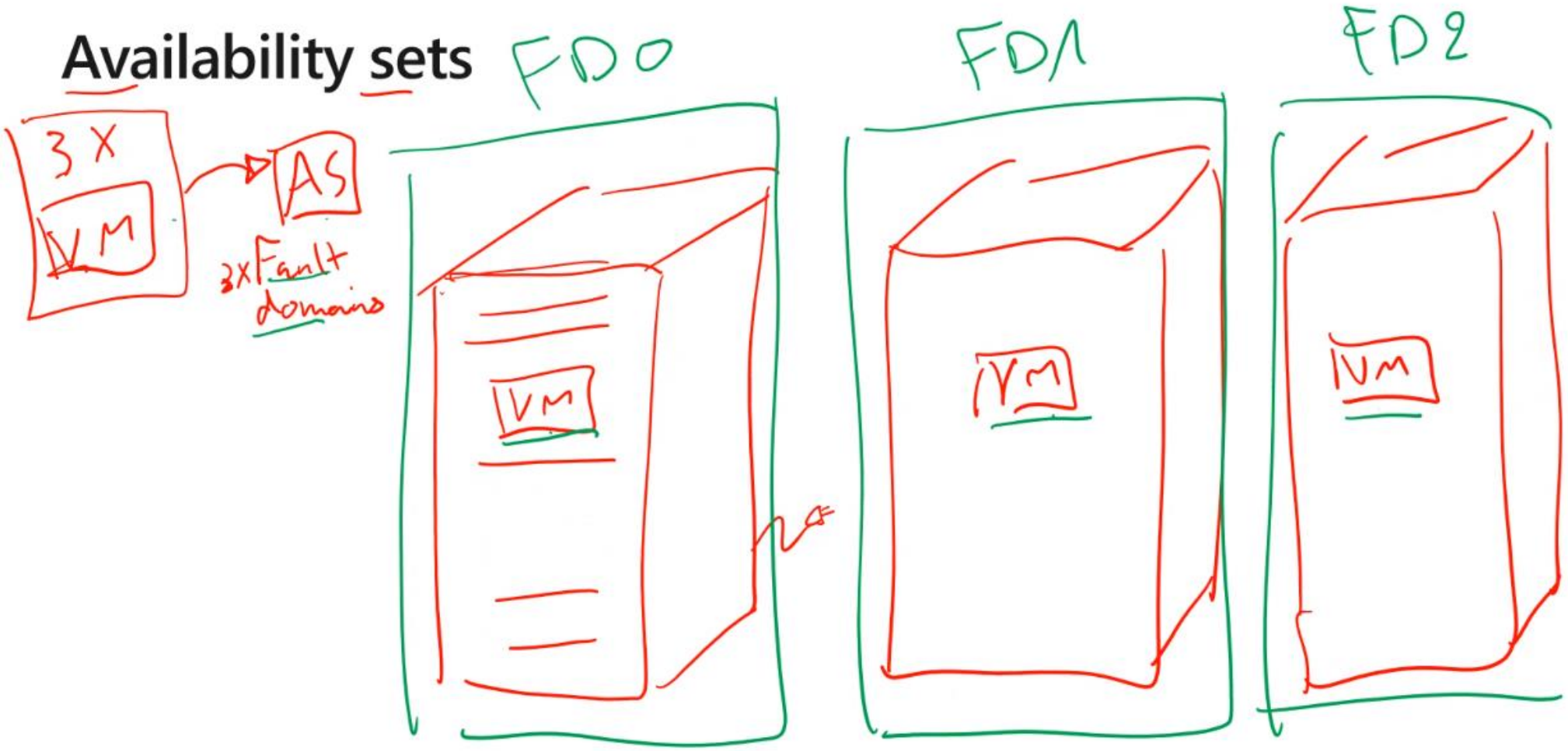
Computational performance	1 virtual CPU (vCPU) - 128 vCPUs
Memory	1 gibibyte (GiB) - 4 tebibyte (TiB)
Disk storage	4GiB - 64TiB Up 160,000 IOPs
Networking	30 GB Ethernet 100 GB InfiniBand
Availability	Single VM service-level agreement (SLA) 99.9% Multi AZ SLA 99.99%



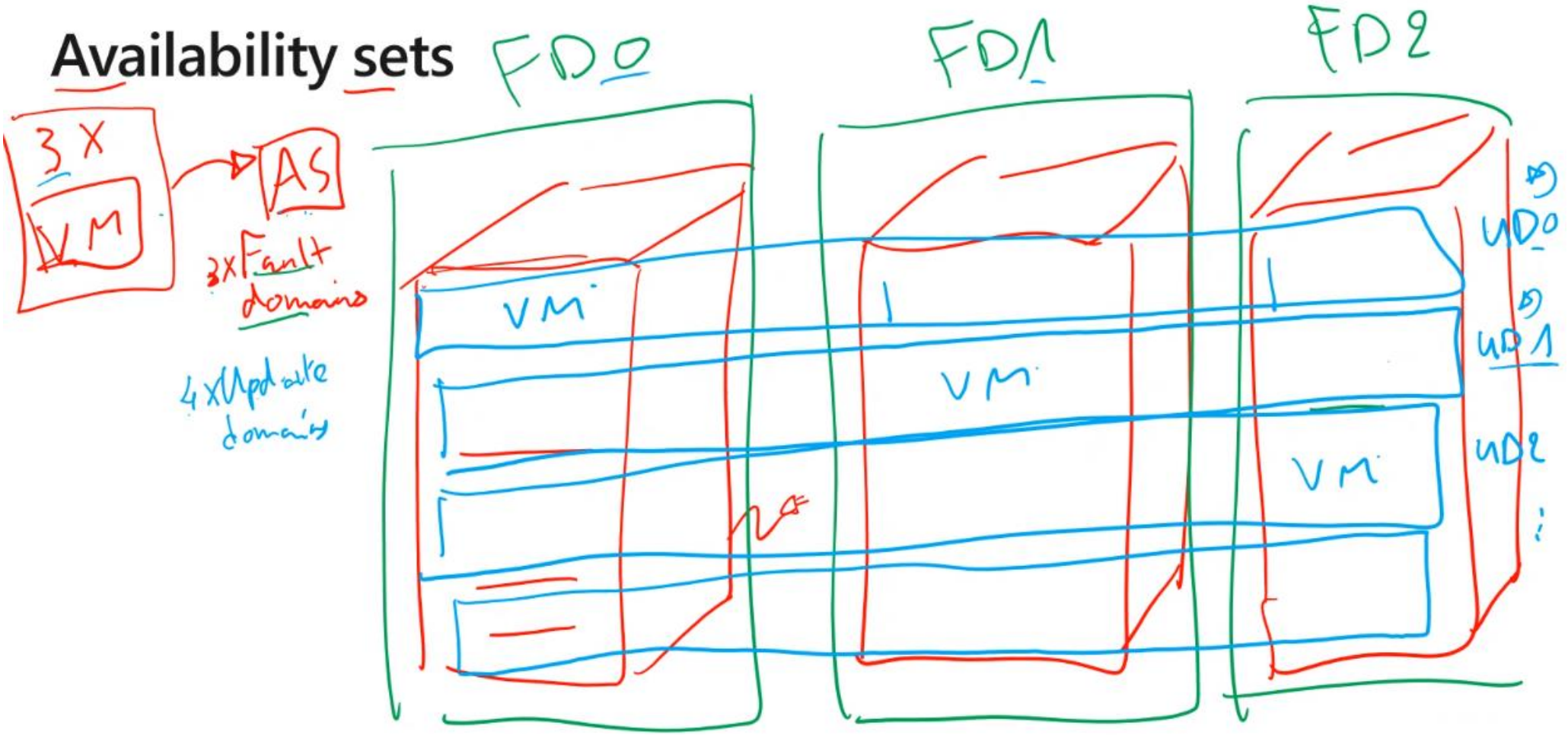
High availability and disaster recovery



Availability sets

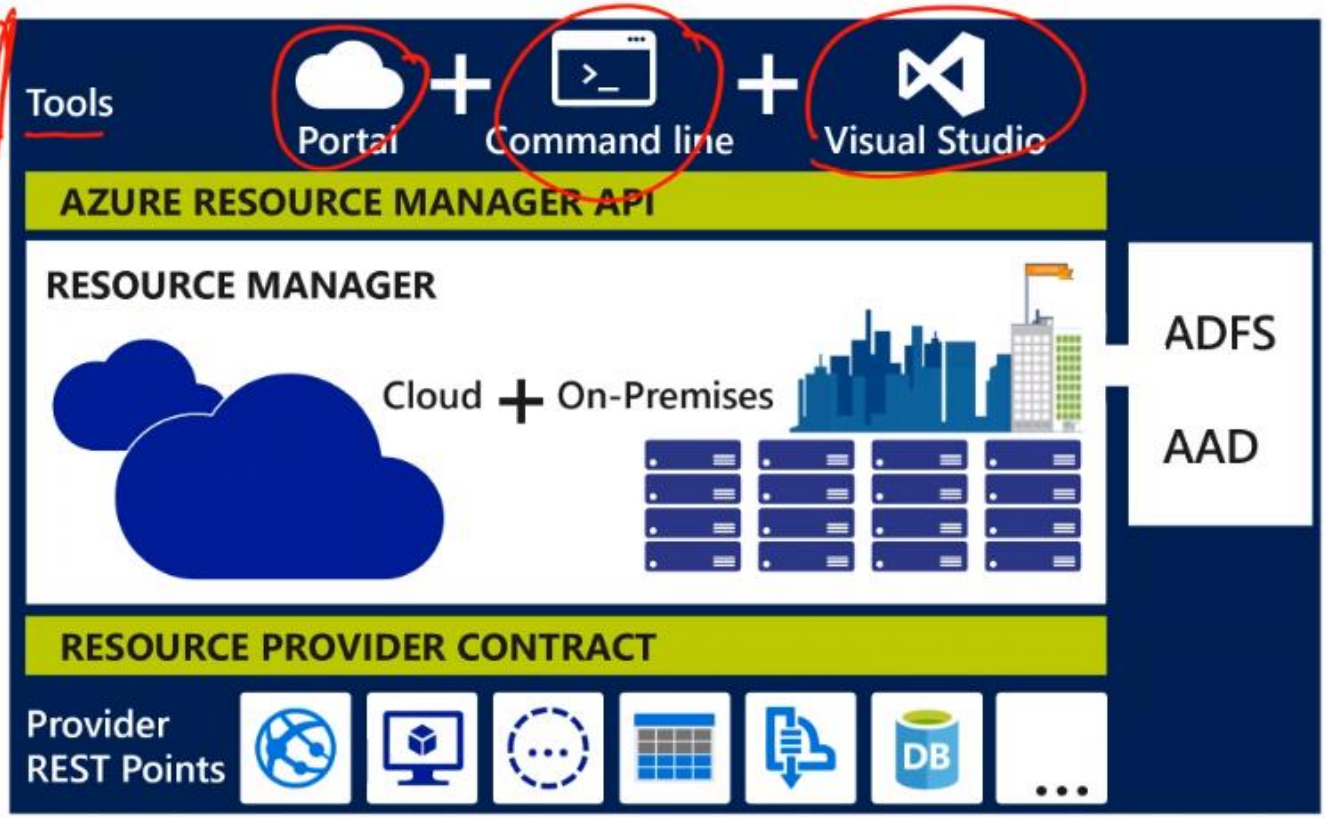


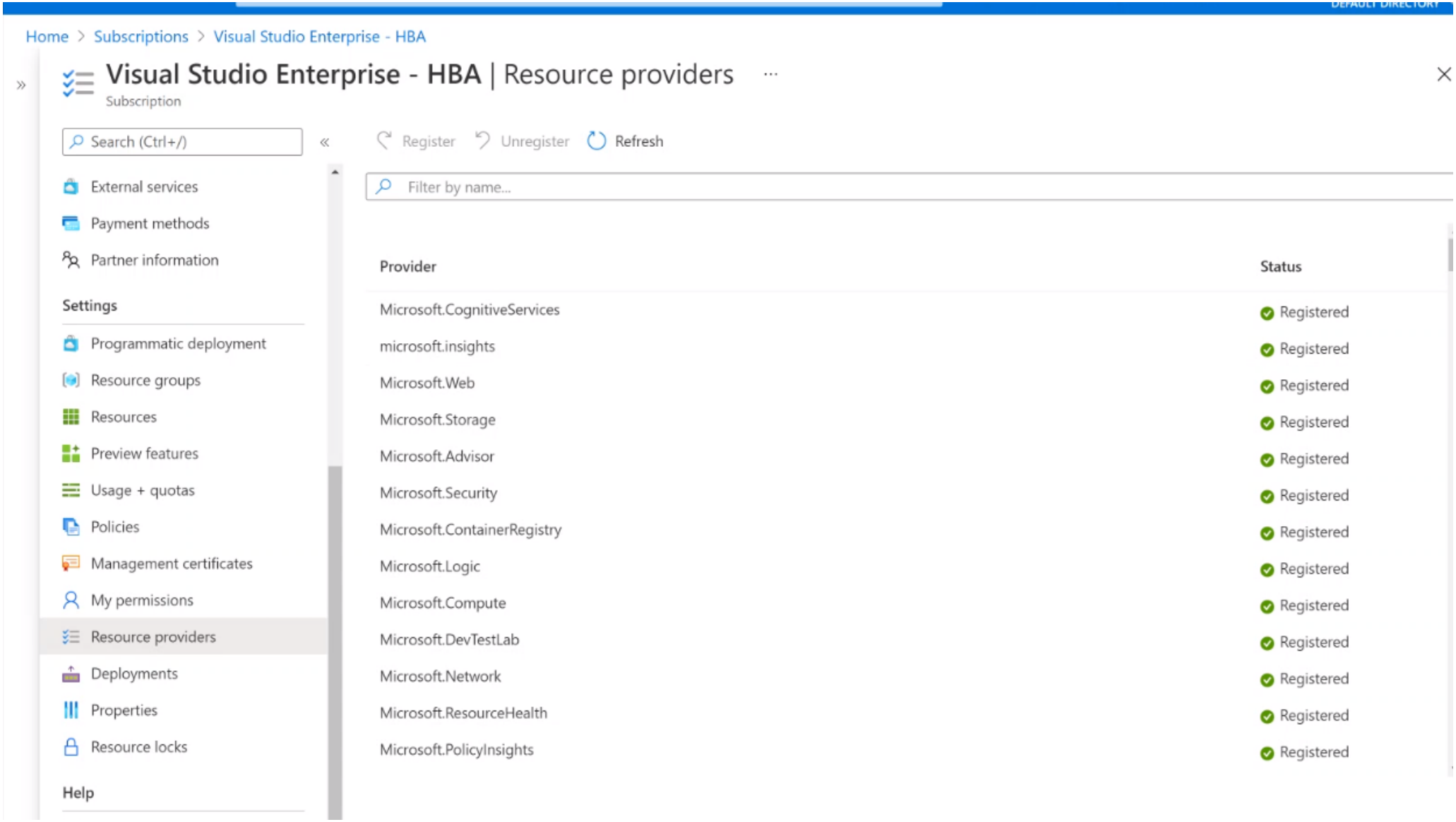
Availability sets

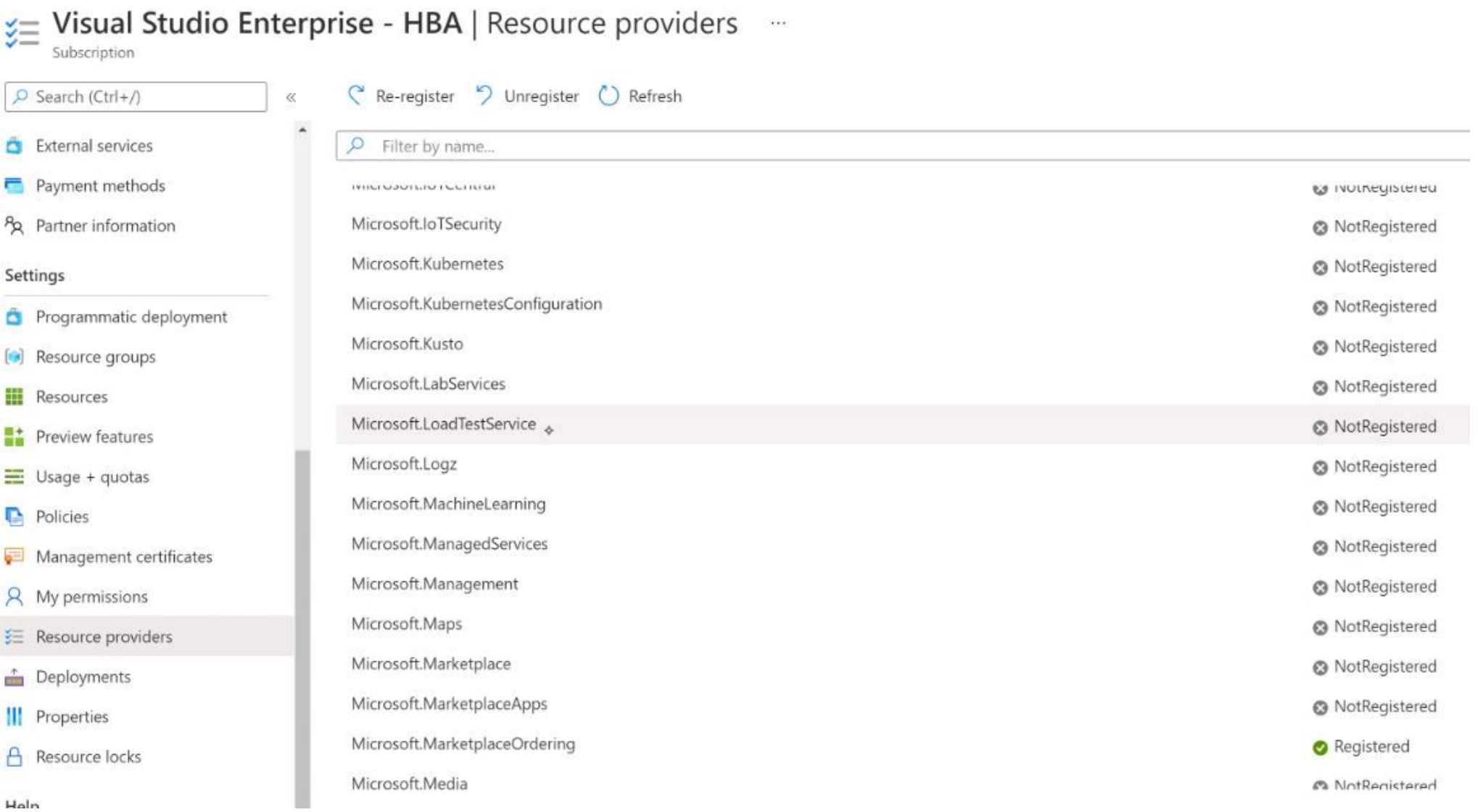


Azure Resource Manager overview

- Resource Manager provides a consistent management layer to perform tasks
  - Azure PowerShell
  - Azure CLI
  - Azure portal
  - REST API
  - Client SDKs




Home > Subscriptions > Visual Studio Enterprise - HBA
Visual Studio Enterprise - HBA | Resource providers
Subscription
Search (Ctrl+/)
Register
Unregister
Refresh
Filter by name...
External services
Payment methods
Partner information
Settings
Programmatic deployment
Resource groups
Resources
Preview features
Usage + quotas
Policies
Management certificates
My permissions
Resource providers
Deployments
Properties
Resource locks
Help
Provider
Status
Microsoft.CognitiveServices
Registered
microsoft.insights
Registered
Microsoft.Web
Registered
Microsoft.Storage
Registered
Microsoft.Advisor
Registered
Microsoft.Security
Registered
Microsoft.ContainerRegistry
Registered
Microsoft.Logic
Registered
Microsoft.Compute
Registered
Microsoft.DevTestLab
Registered
Microsoft.Network
Registered
Microsoft.ResourceHealth
Registered
Microsoft.PolicyInsights
Registered


Visual Studio Enterprise - HBA | Resource providers
Subscription
Search (Ctrl+/)
Re-register
Unregister
Refresh
External services
Payment methods
Partner information
Settings
Programmatic deployment
Resource groups
Resources
Preview features
Usage + quotas
Policies
Management certificates
My permissions
Resource providers
Deployments
Properties
Resource locks
Help
Filter by name...
Microsoft.IoTSecurity NotRegistered
Microsoft.Kubernetes NotRegistered
Microsoft.KubernetesConfiguration NotRegistered
Microsoft.Kusto NotRegistered
Microsoft.LabServices NotRegistered
Microsoft.LoadTestService NotRegistered
Microsoft.Logz NotRegistered
Microsoft.MachineLearning NotRegistered
Microsoft.ManagedServices NotRegistered
Microsoft.Management NotRegistered
Microsoft.Maps NotRegistered
Microsoft.Marketplace NotRegistered
Microsoft.MarketplaceApps NotRegistered
Microsoft.MarketplaceOrdering Registered
Microsoft.Media

Terminology

- Resource
  - Single manageable item available through Azure
- Resource group
  - Container holding related resources
- Resource provider
  - Service that supplies resource instances in accordance with a predefined contract
- Resource Manager template
  - JSON file that defines one or more resources, specifying their resource providers, to be deployed to a resource group
- Declarative syntax
  - The act of describing your resources by using a template instead of manually creating the resources

https://portal.azure.com/#create/Microsoft.Template

Microsoft Azure | Search resources, services, and docs (G+/)

Home >

Custom deployment ...

Deploy from a custom template

Select a template | Basics | Review + create

Automate deploying resources with Azure Resource Manager templates in a single, coordinated operation. Create or select a template below to get started. Learn more about template deployment

Build your own template in the editor

Common templates

- Create a Linux virtual machine
- Create a Windows virtual machine
- Create a web app
- Create a SQL database
- Azure landing zone

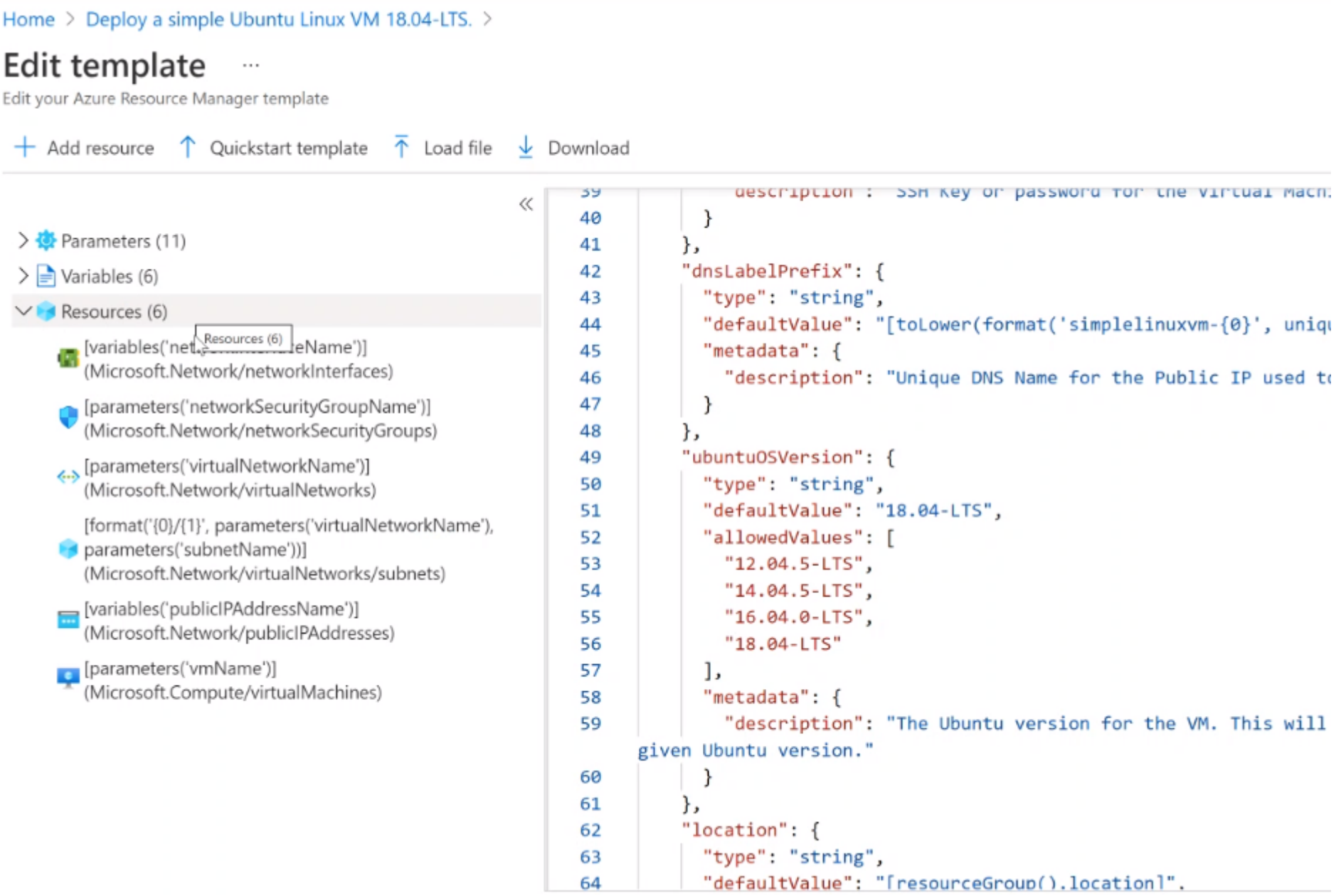
Start with a quickstart template or template spec

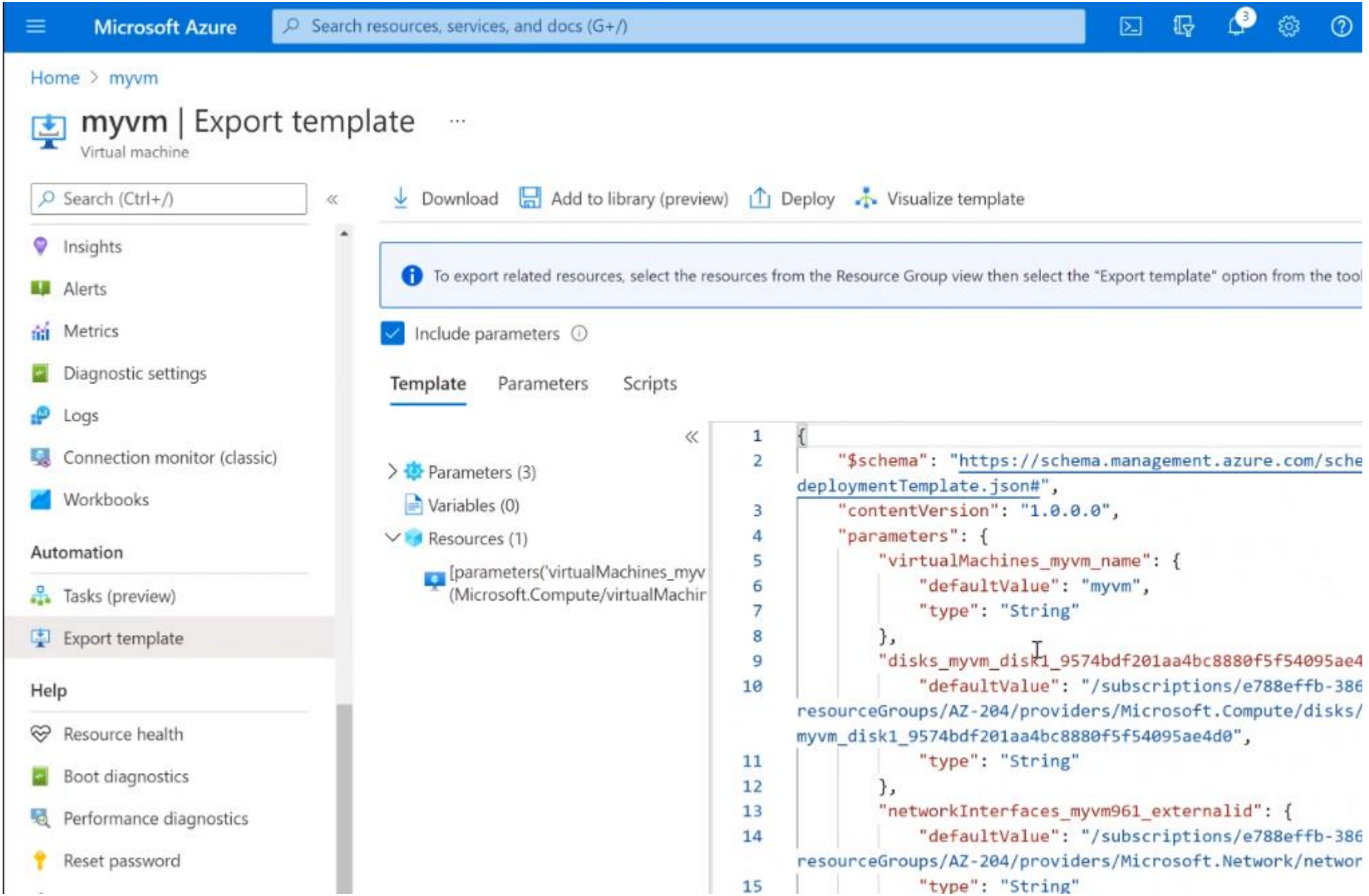
Template source

- (•) Quickstart template
- ( ) Template spec

Quickstart template (disclaimer)

Template içine parameters koyup formdan seçtirebiliryorsun



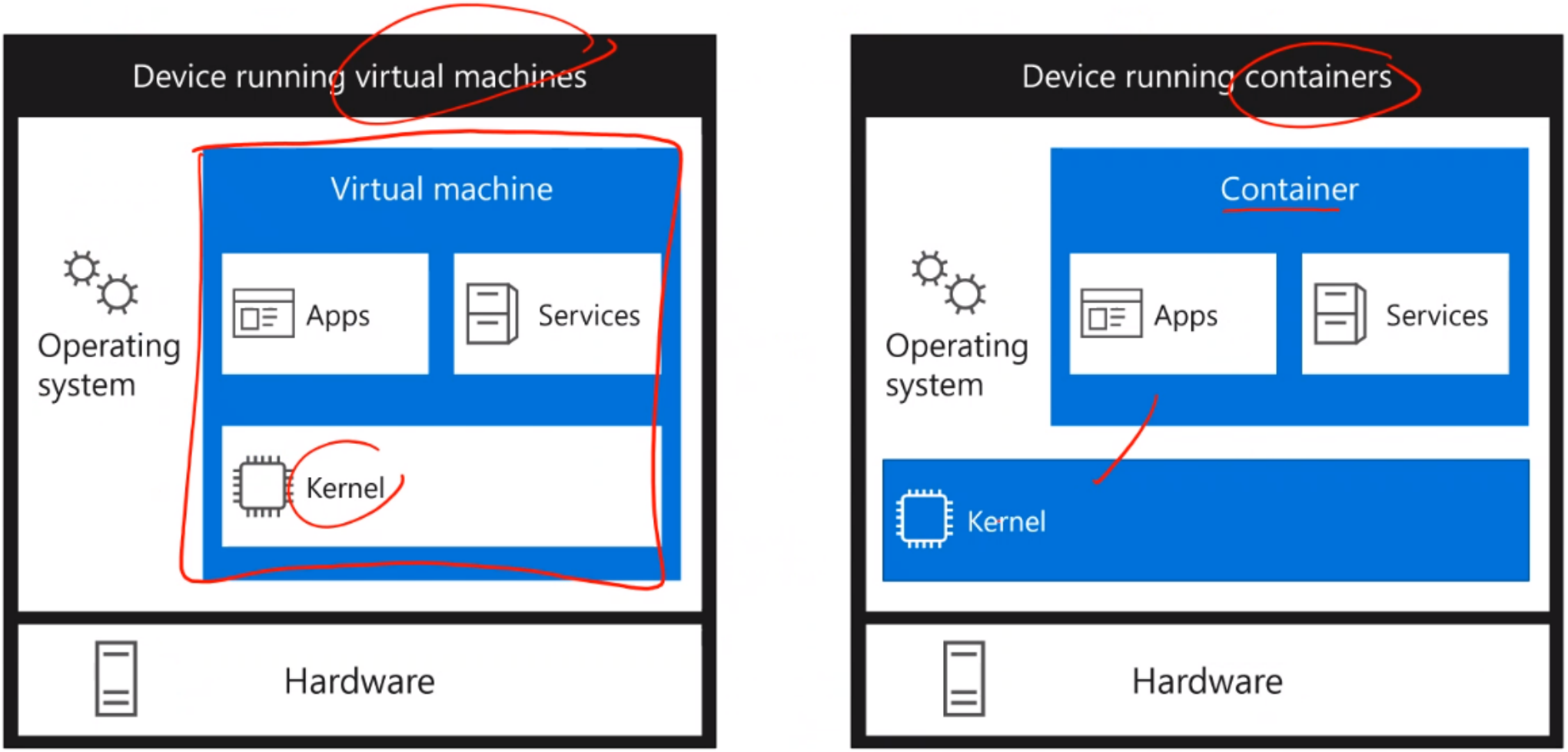

Microsoft Azure
Search resources, services, and docs (G+/)
Home > myvm
myvm | Export template
Virtual machine
Search (Ctrl+/)
Download
Add to library (preview)
Deploy
Visualize template
Insights
Alerts
Metrics
Diagnostic settings
Logs
Connection monitor (classic)
Workbooks
Automation
Tasks (preview)
Export template
Help
Resource health
Boot diagnostics
Performance diagnostics
Reset password
To export related resources, select the resources from the Resource Group view then select the "Export template" option from the tool
Include parameters
Template
Parameters
Scripts
Parameters (3)
Variables (0)
Resources (1)
[parameters('virtualMachines_myv
(Microsoft.Compute/virtualMachir
{
"$schema": "https://schema.management.azure.com/sche
deploymentTemplate.json#",
"contentVersion": "1.0.0.0",
"parameters": {
"virtualMachines_myvm_name": {
"defaultValue": "myvm",
"type": "String"
},
"disks_myvm_disk1_9574bdf201aa4bc8880f5f54095ae4
"defaultValue": "/subscriptions/e788effb-386
resourceGroups/AZ-204/providers/Microsoft.Compute/disks/
myvm_disk1_9574bdf201aa4bc8880f5f54095ae4d0",
"type": "String"
},
"networkInterfaces_myvm961_externalid": {
"defaultValue": "/subscriptions/e788effb-386
resourceGroups/AZ-204/providers/Microsoft.Network/networ
"type": "String"

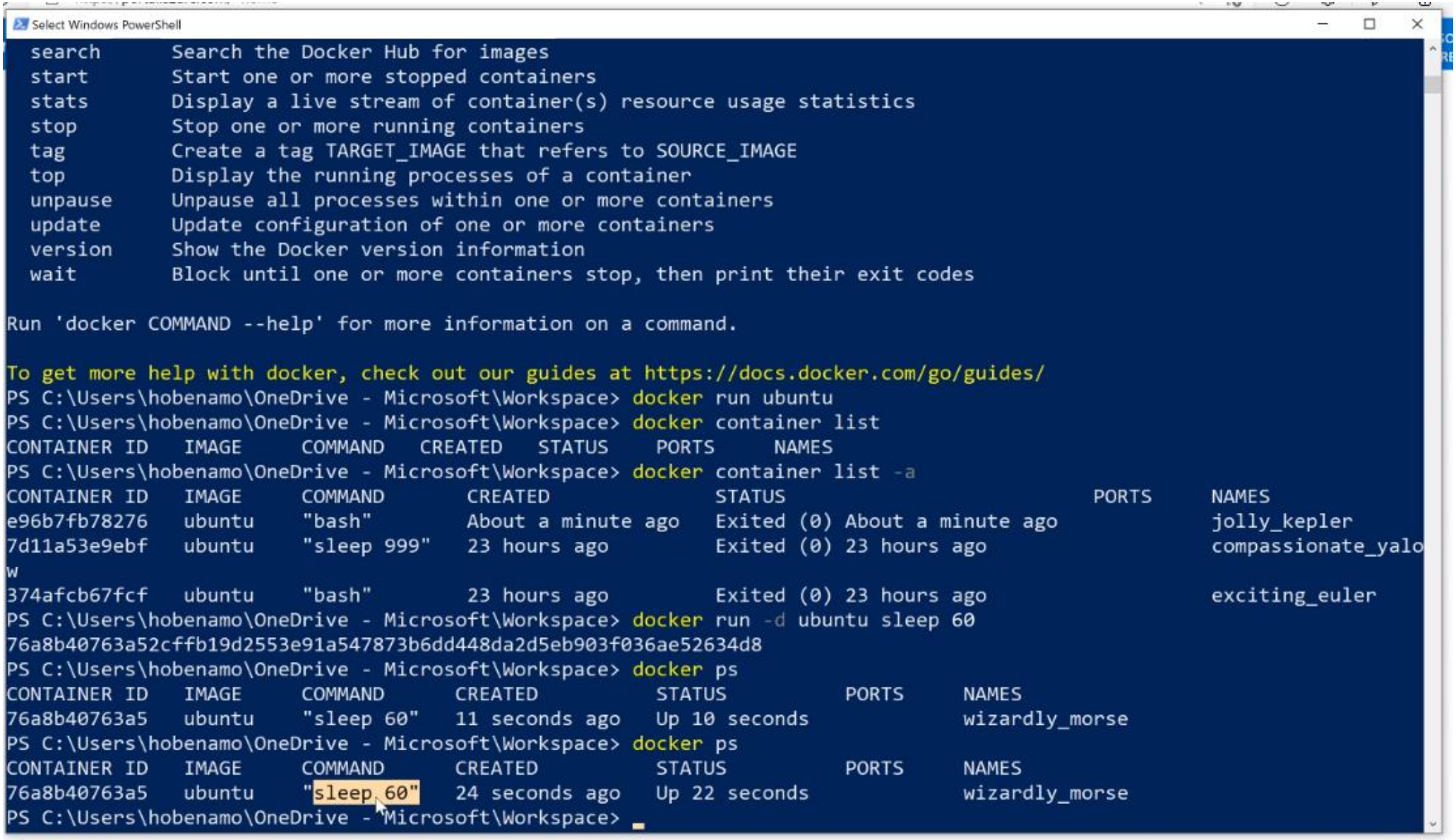
Export edip başka yerde kullanmadan önce disk tanımlaması gibi sabitleri silmek, yerine başka disk bilsgisini yazmak veya parametrik olarak eklemek gerekebilir.

Fundamentals of Bicep

https://docs.microsoft.com/en-us/learn/paths/fundamentals-bicep/

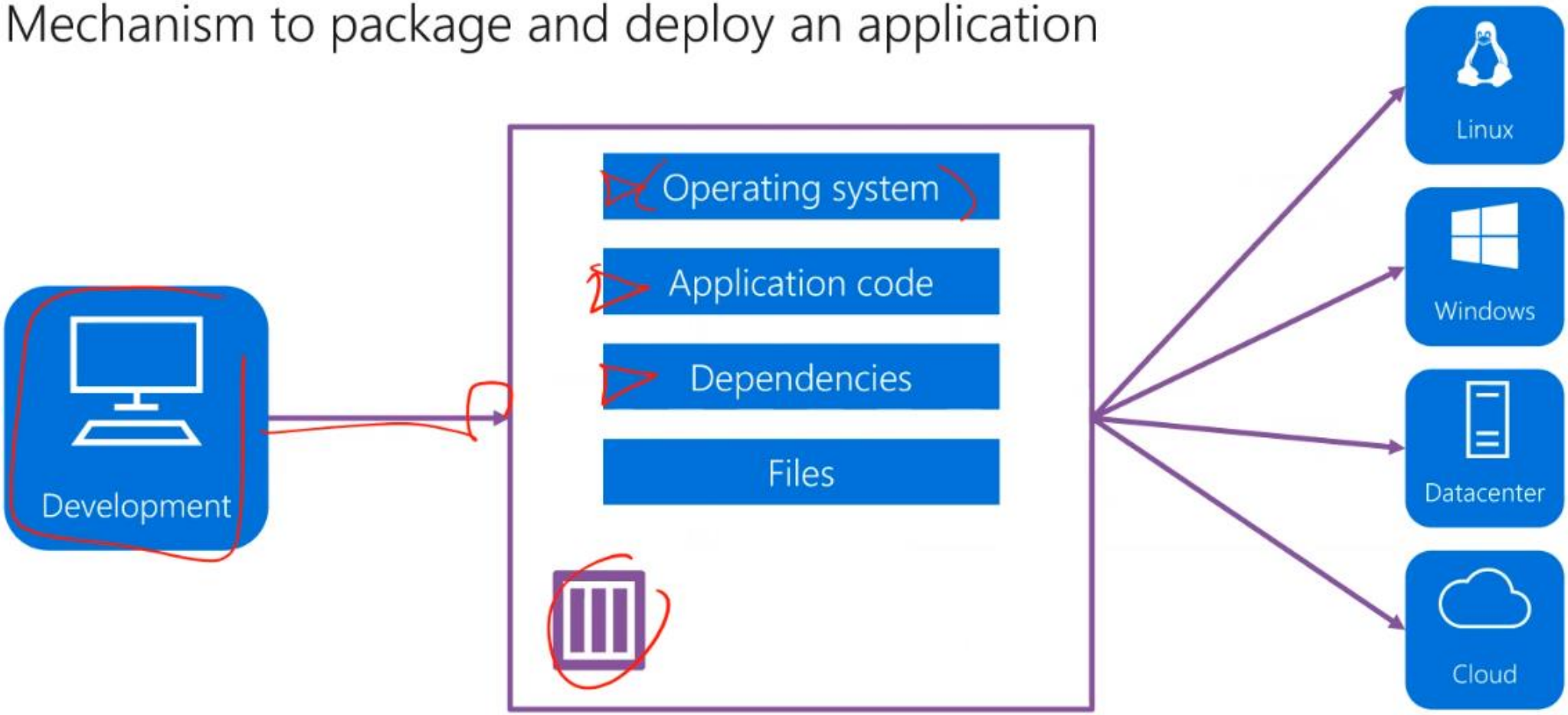
Virtualization and containers

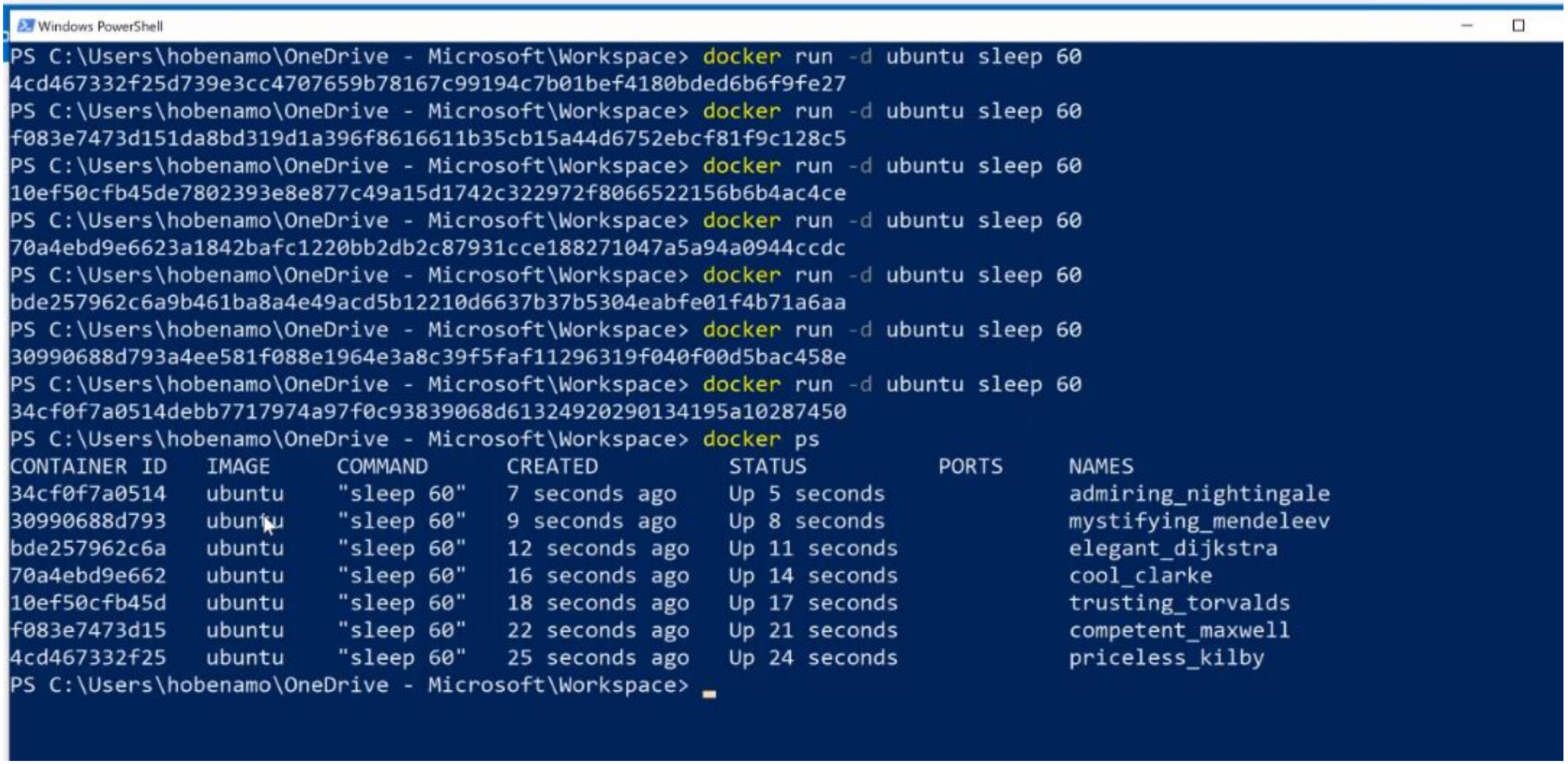



Select Windows PowerShell
search Search the Docker Hub for images
start Start one or more stopped containers
stats Display a live stream of container(s) resource usage statistics
stop Stop one or more running containers
tag Create a tag TARGET_IMAGE that refers to SOURCE_IMAGE
top Display the running processes of a container
unpause Unpause all processes within one or more containers
update Update configuration of one or more containers
version Show the Docker version information
wait Block until one or more containers stop, then print their exit codes
Run 'docker COMMAND --help' for more information on a command.
To get more help with docker, check out our guides at https://docs.docker.com/go/guides/
PS C:\Users\hobenamo\OneDrive - Microsoft\Workspace> docker run ubuntu
PS C:\Users\hobenamo\OneDrive - Microsoft\Workspace> docker container list
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
PS C:\Users\hobenamo\OneDrive - Microsoft\Workspace> docker container list -a
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
e96b7fb78276 ubuntu "bash" About a minute ago Exited (0) About a minute ago jolly_kepler
7d11a53e9ebf ubuntu "sleep 999" 23 hours ago Exited (0) 23 hours ago compassionate_yalo
w
374afcb67fcf ubuntu "bash" 23 hours ago Exited (0) 23 hours ago exciting_euler
PS C:\Users\hobenamo\OneDrive - Microsoft\Workspace> docker run -d ubuntu sleep 60
76a8b40763a52cffb19d2553e91a547873b6dd448da2d5eb903f036ae52634d8
PS C:\Users\hobenamo\OneDrive - Microsoft\Workspace> docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
76a8b40763a5 ubuntu "sleep 60" 11 seconds ago Up 10 seconds wizardly_morse
PS C:\Users\hobenamo\OneDrive - Microsoft\Workspace> docker ps
CONTAINER ID IMAGE COMMAND CREATED STATUS PORTS NAMES
76a8b40763a5 ubuntu "sleep 60" 24 seconds ago Up 22 seconds wizardly_morse
PS C:\Users\hobenamo\OneDrive - Microsoft\Workspace>

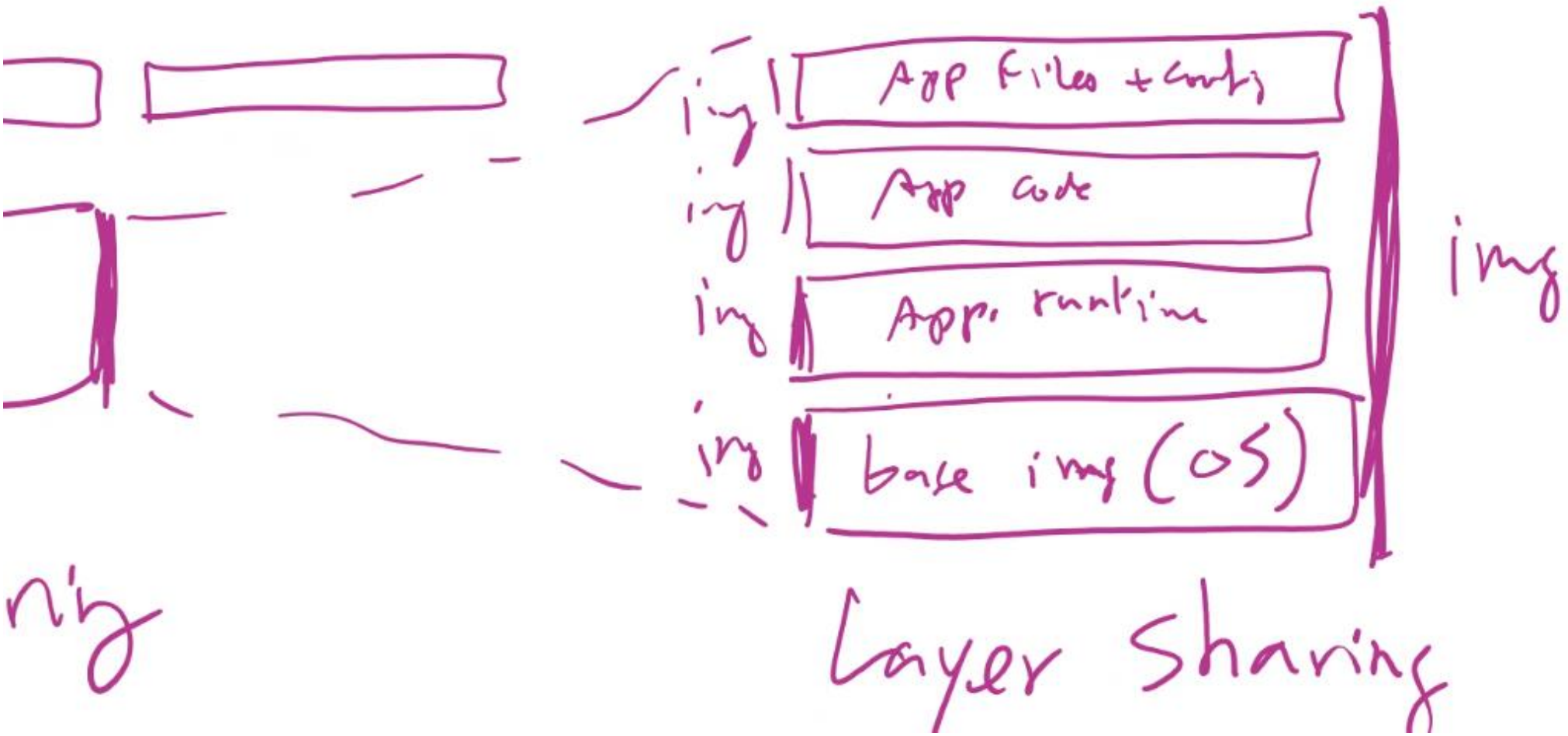
Containers

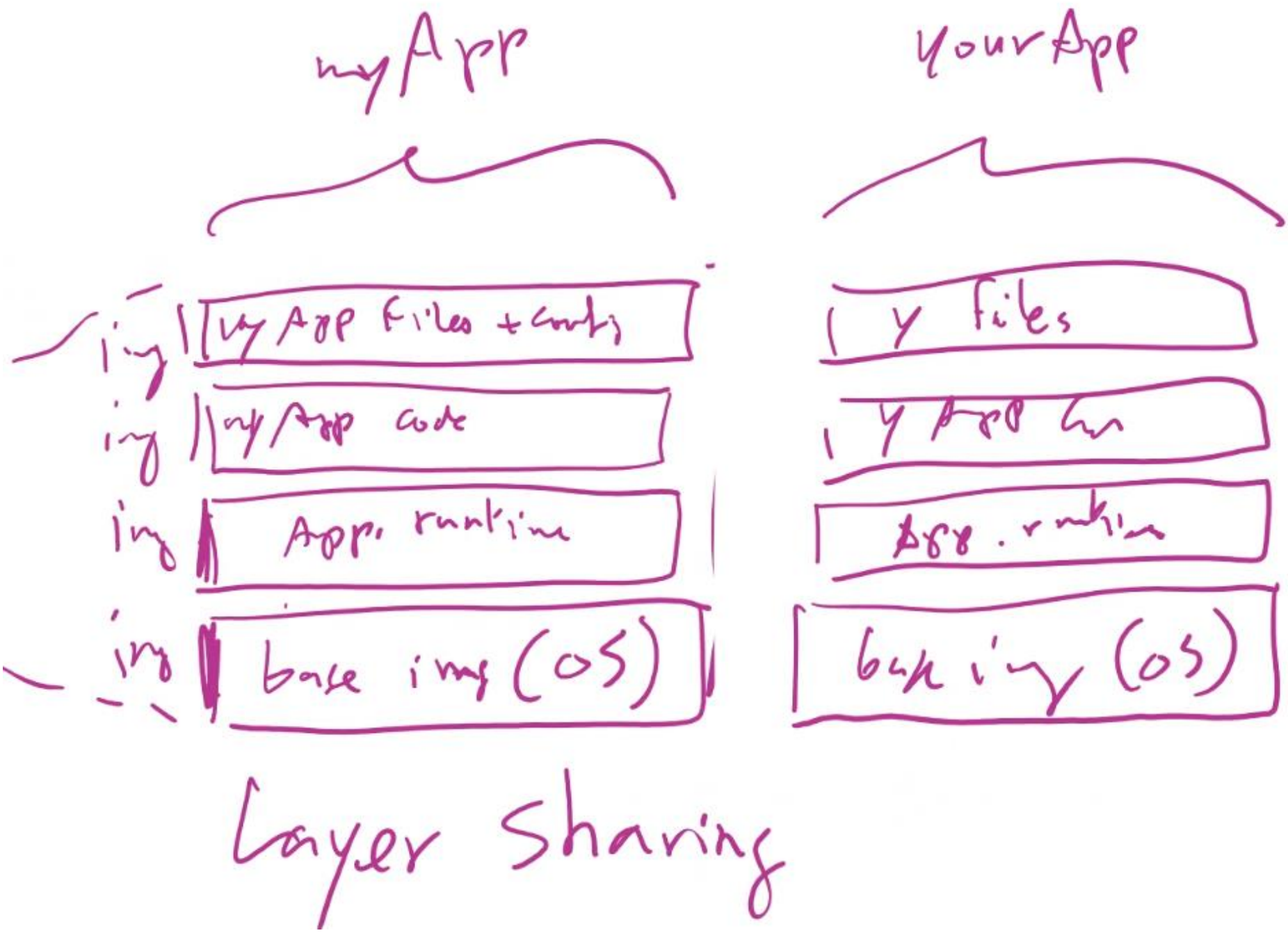
Mechanism to package and deploy an application

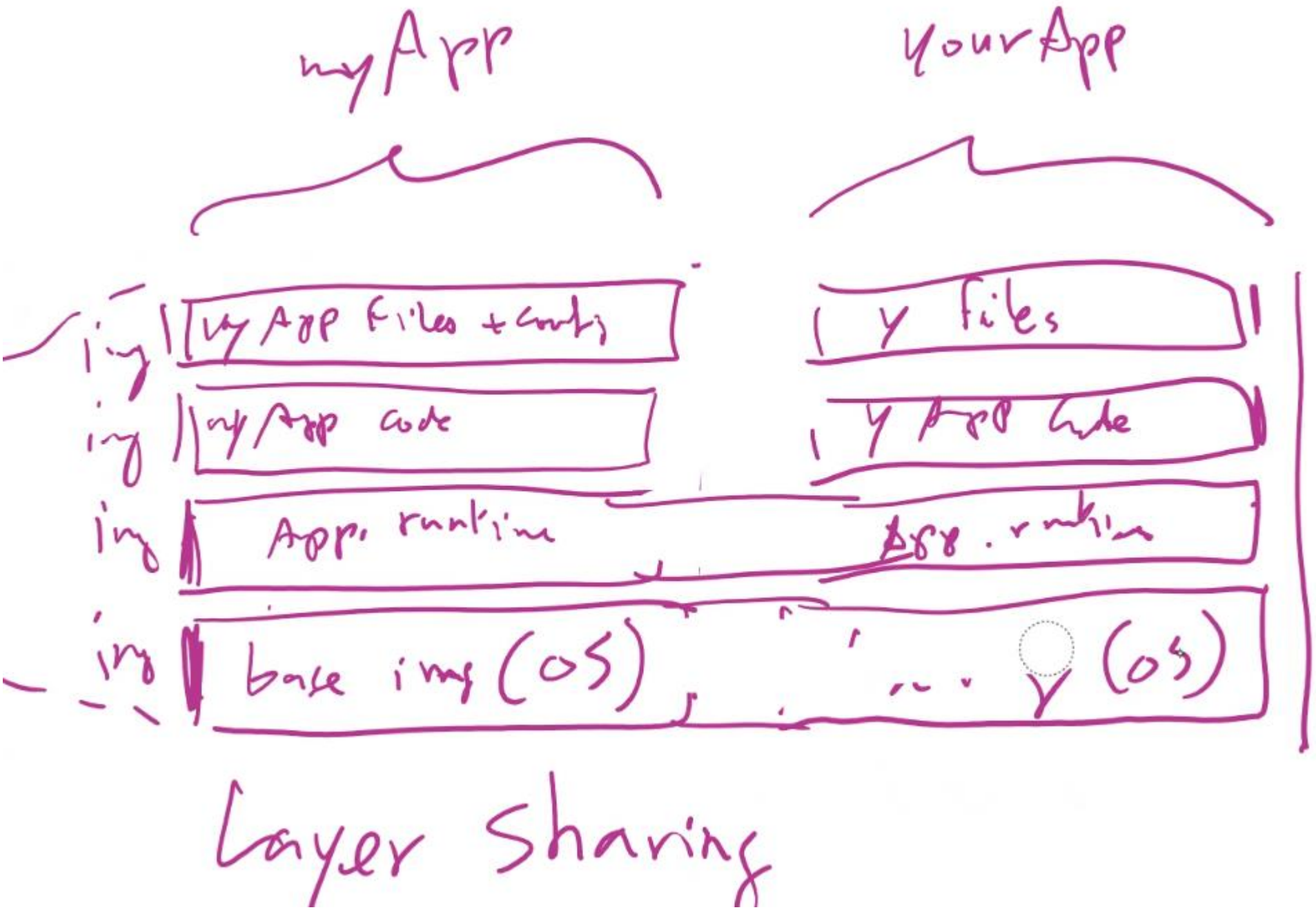




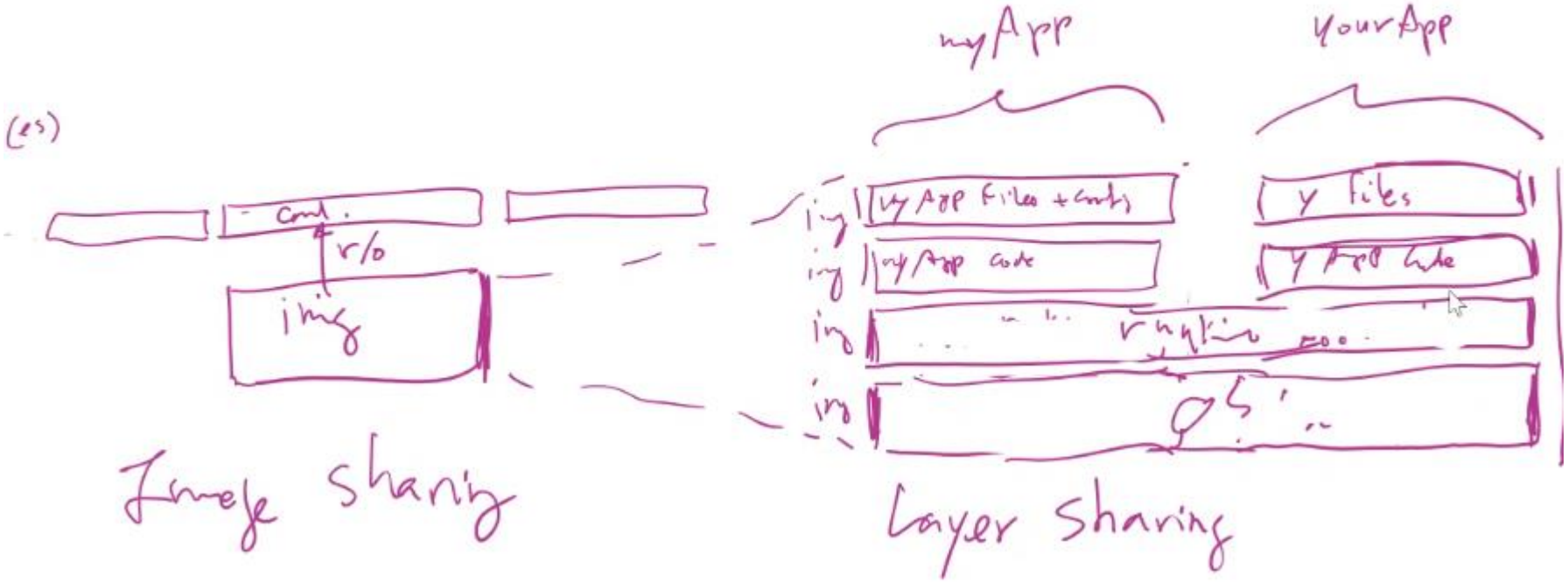
https://docs.microsoft.com/en-us/azure/app-service/deploy-run-package


App Files + Config
App code
App runtime
base img (OS)
img
img
img
img
img
Layer Sharing


myApp
YourApp
My App Files + Config
My App code
App. runtime
base img (OS)
Y files
Y App Cfg
App. runtim
base img (OS)
Layer Sharing


myApp
YourApp
ing
ing
ing
ing
base img (os)
Layer Sharing

(es)


myApp
YourApp
Cont.
r/o
img
img
MyApp Files + Config
MyApp code
Y files
Y App Code
img
Runtime ...
img
OS ...
Image sharing
Layer Sharing

Docker terminology

- **Container**
  - A standardized "unit of software" that contains everything required for an application to run
- **Container Image**
  - A template that can be used to create one or more containers
- **Build**
  - The process of creating a container image using a set of instructions
- **Pull**
  - The process of downloading a container image from a container registry
- **Push**
  - The process of uploading a container image to a container registry
- **Dockerfile**
  - A text file that contains instructions required to build a Docker image.

Creating a container image specification with a Dockerfile

```
FROM node:8.9.3-alpine
RUN mkdir -p /usr/src/app
COPY ./app/ /usr/src/app/
WORKDIR /usr/src/app
RUN npm install
CMD node /usr/src/app/index.js
```

Start with this container image

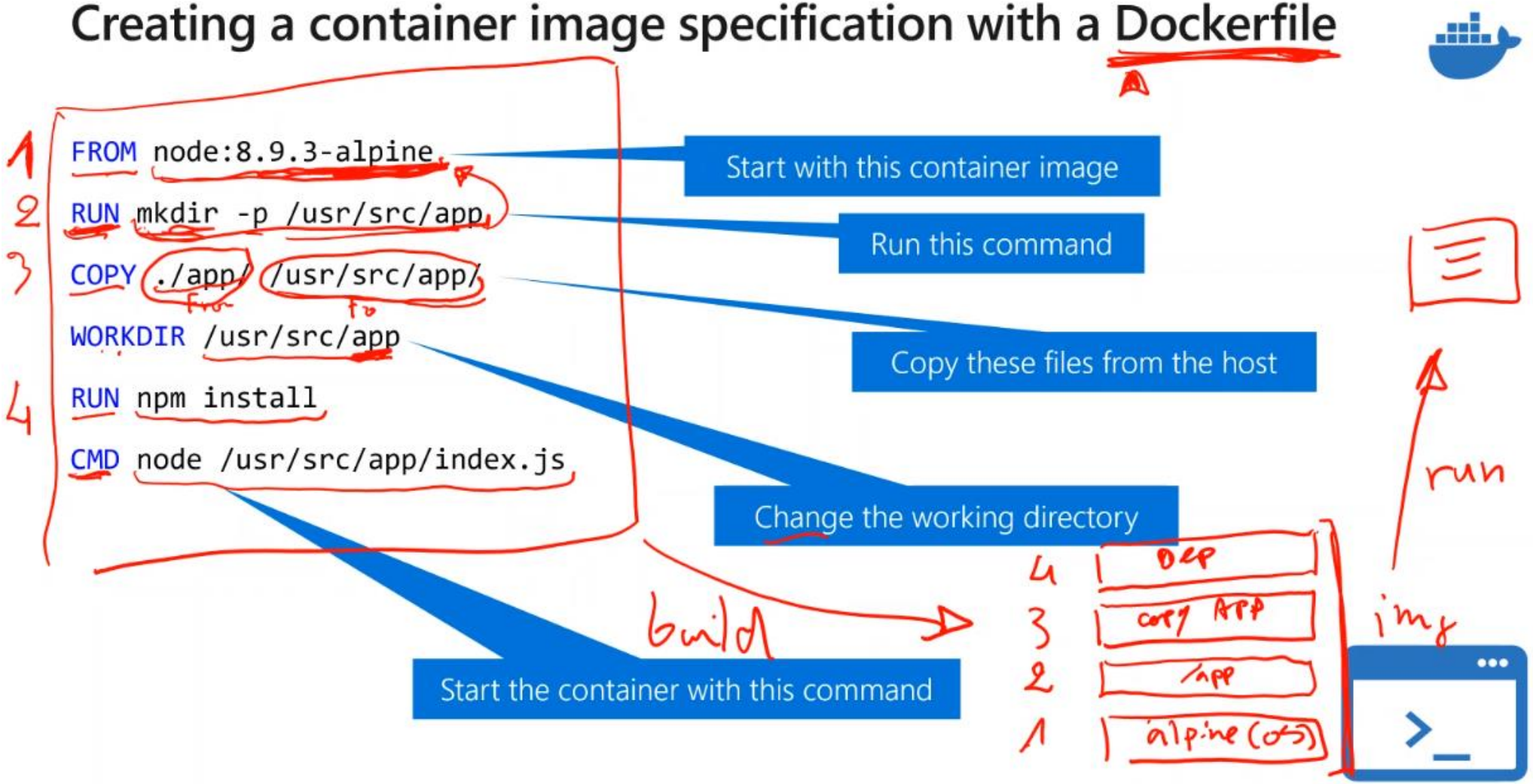
Run this command

Copy these files from the host

Change the working directory

Start the container with this command

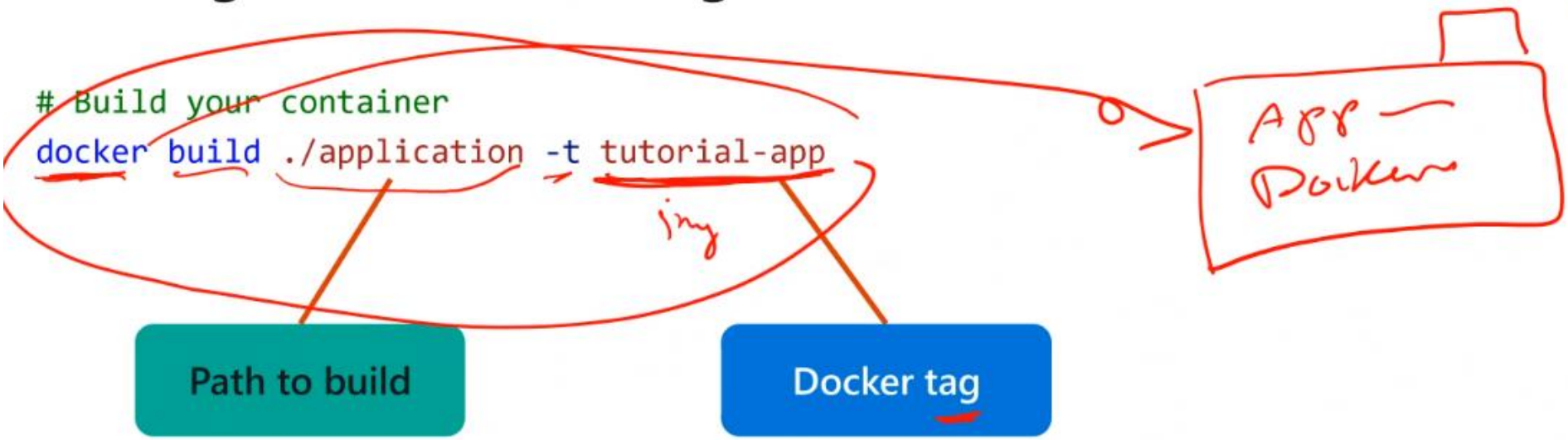
Creating a container image specification with a Dockerfile



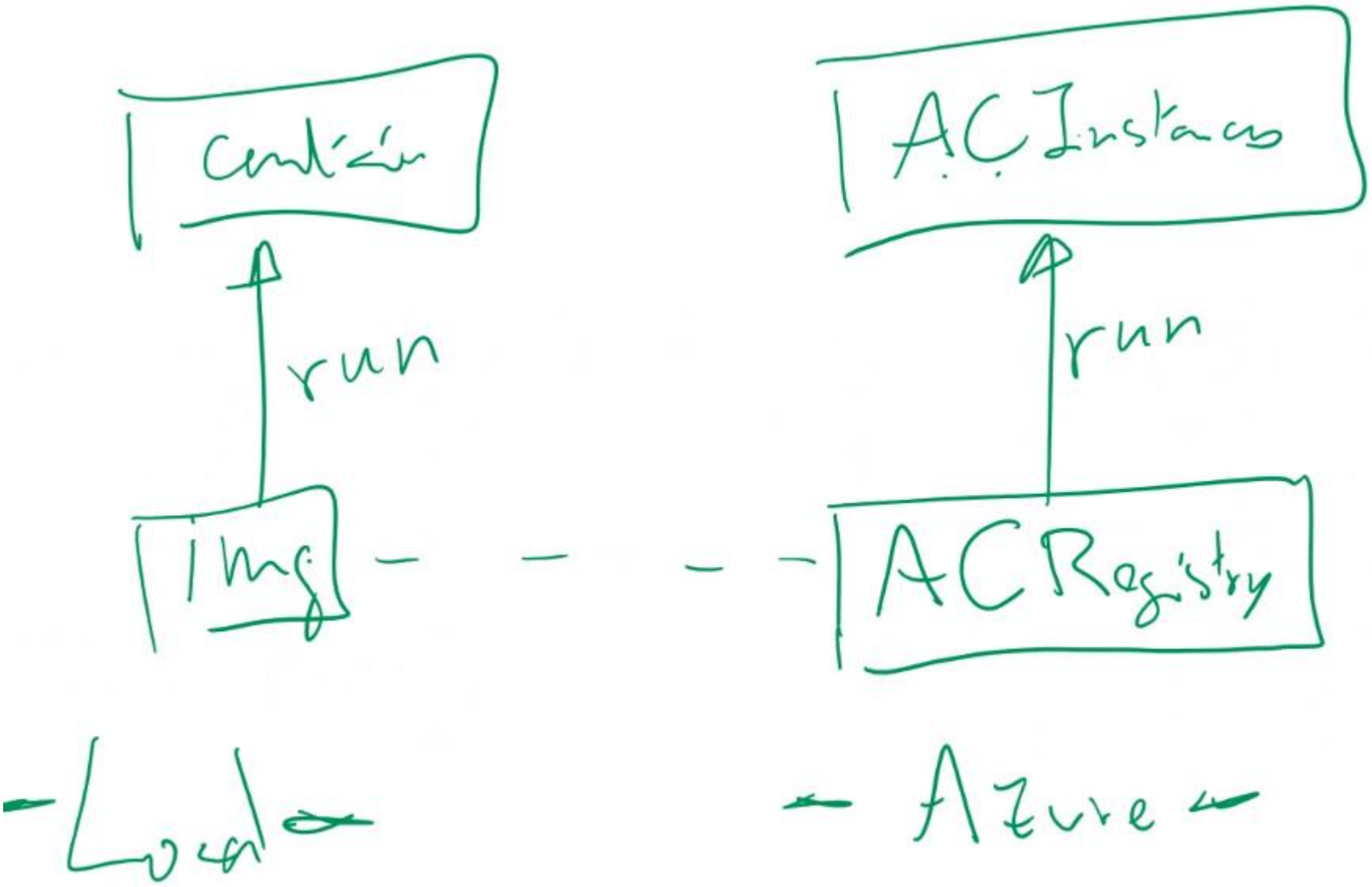
Sınavda gelir bu soru!!

https://docs.docker.com/config/containers/multi-service_container/

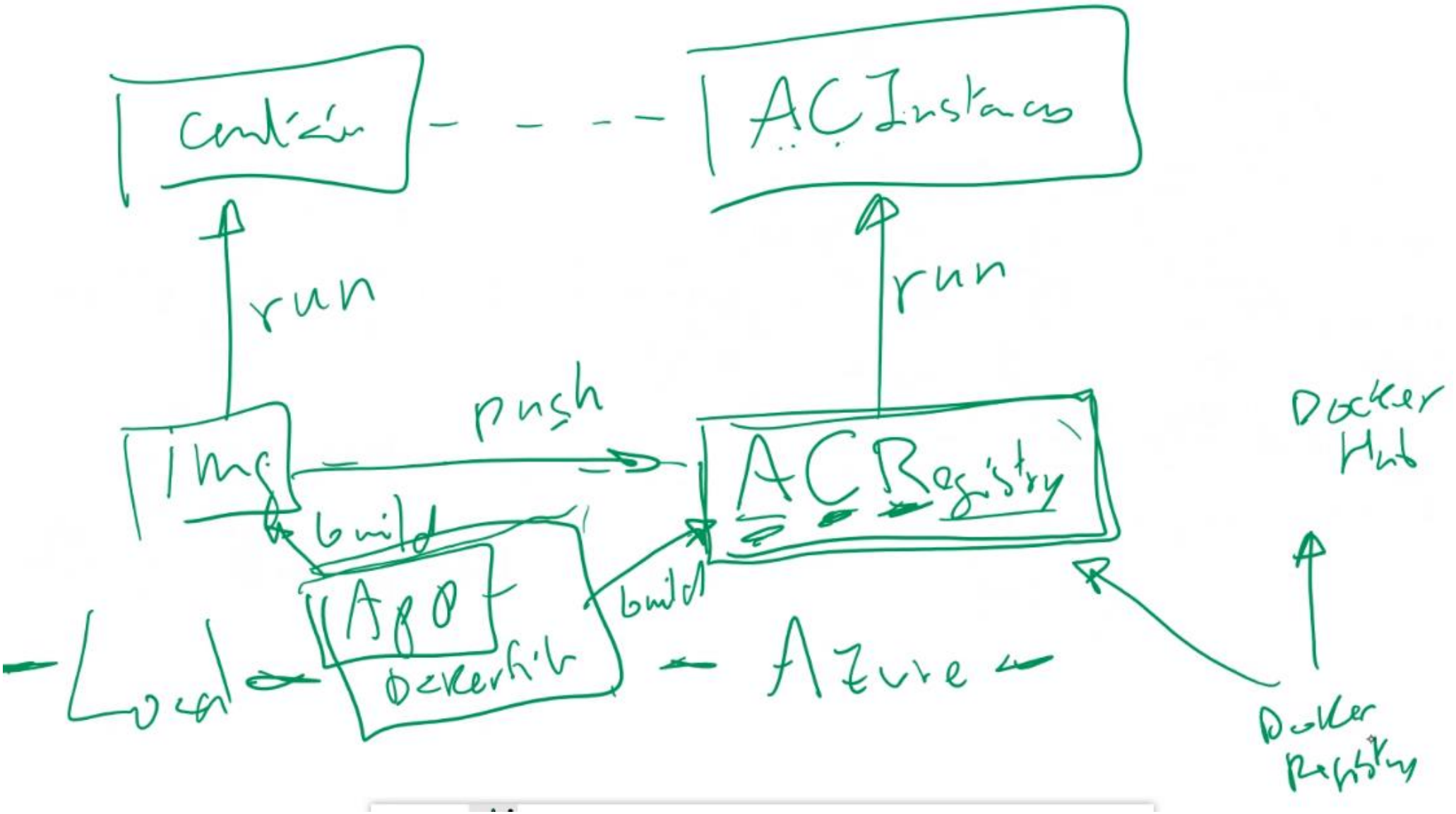
Building the container image




```
# Build your container
docker build ./application -t tutorial-app

# After building, use the following command to view your new container image
docker image list
```



Build işini Azure container Register yapıyor.!!


Container
ACI Instance
run
run
push
Img
ACR Registry
Docker Hub
build
build
API
Dockerfile
Local
Azure
Docker Registry

Create a container registry by using Azure CLI

```
# Create a Container Registry instance
az acr create --resource-group <group> --name <acr-name> --sku Basic

# Login to Container Registry
az acr login --name <acrName>
```

Trigger ACR Build by using Azure CLI

```
# Trigger build in Azure
az acr build --image <server>/<tag> --registry <registry> ./app
```

Registry server

Docker "tag"

Path to build

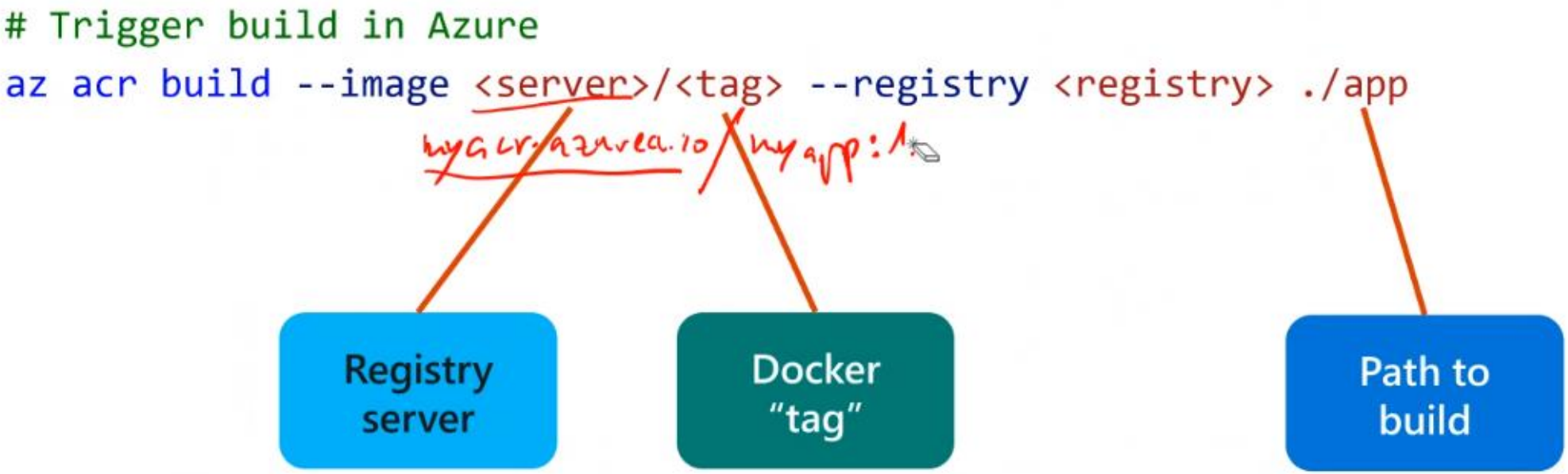
```
PS C:\Users\hobenamo\OneDrive - Microsoft\Workspace> docker run -d alpine
Unable to find image 'alpine:latest' locally
```

Pull ile imajı çekeriz

```
PS C:\Users\hobenamo\OneDrive - Microsoft\Workspace> docker run -d alpine
Unable to find image 'alpine:latest' locally
latest: Pulling from library/alpine
59bf1c3509f3: Pull complete
Digest: sha256:21a3deaa0d32a8057914f36584b5288d2e5ecc984380bc0118285c70fa8c9300
Status: Downloaded newer image for alpine:latest
7c2af165465832ae870843976fa4e1496842b25814a3fa5f66252134279bae14
PS C:\Users\hobenamo\OneDrive - Microsoft\Workspace>
PS C:\Users\hobenamo\OneDrive - Microsoft\Workspace> docker images
REPOSITORY   TAG       IMAGE ID       CREATED         SIZE
alpine       latest    c059bfaa849c   2 weeks ago     5.59MB
ubuntu       latest    ba6acccedd29   2 months ago    72.8MB
PS C:\Users\hobenamo\OneDrive - Microsoft\Workspace>
```

No Version means get the latests

The name of the image means something. Not random.



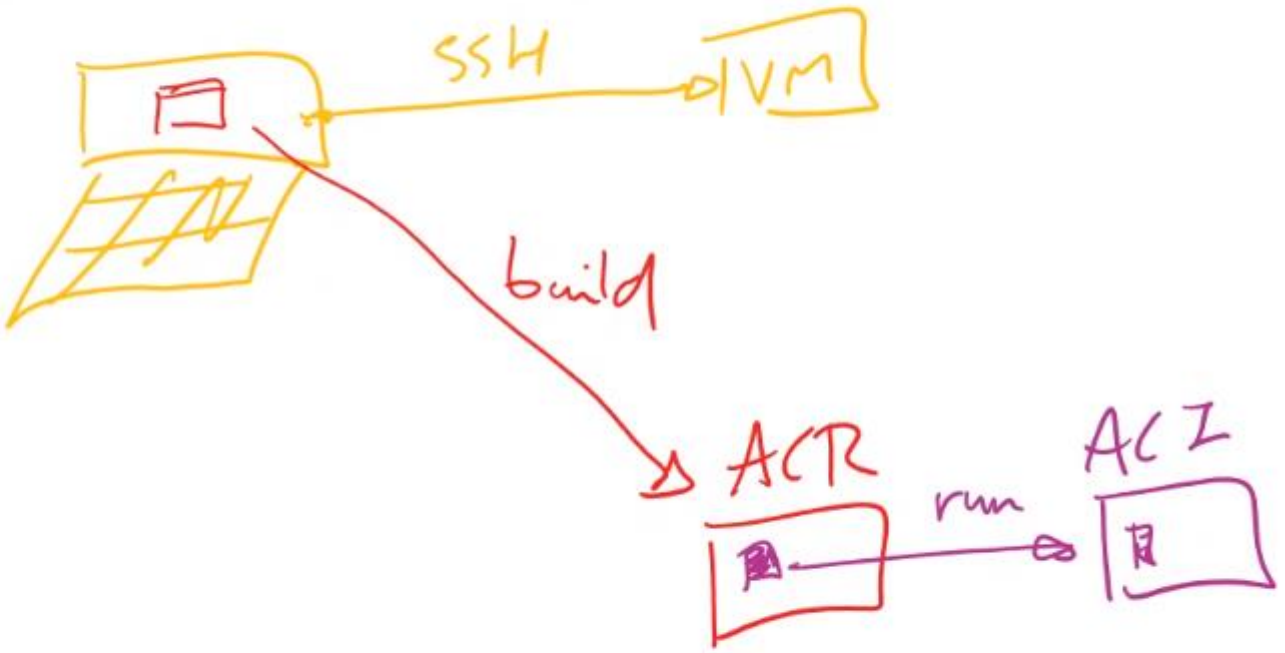
with the AZ container registry, if it is inside of the network domain, do you still need to login ->> YES

Lab 5:

Ex1

Ex2

Ex3



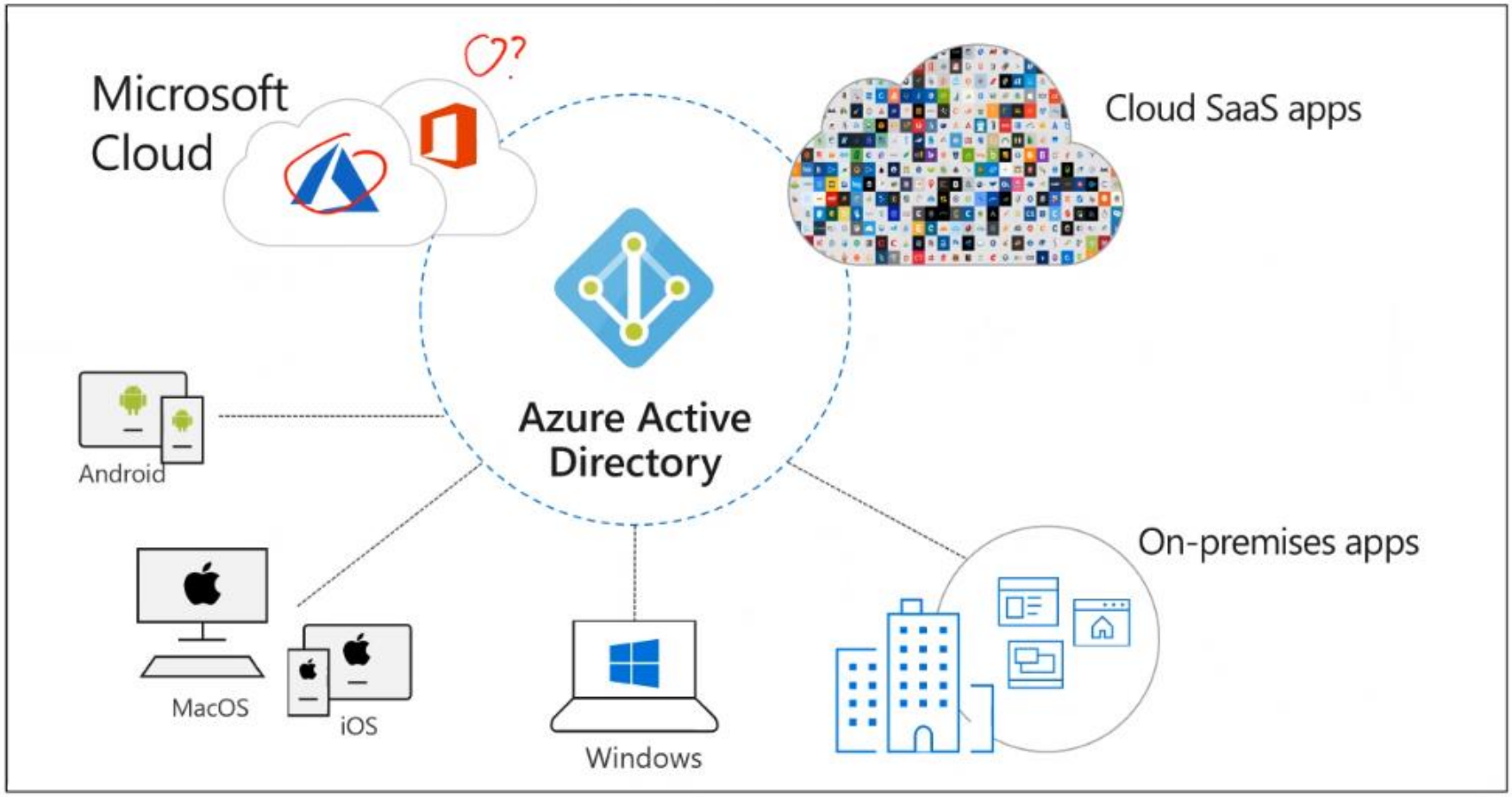
```dockerfile
# Start using the .NET Core 3.1 SDK container image
FROM mcr.microsoft.com/dotnet/sdk:3.1-alpine AS build

# Change current working directory
WORKDIR /app

# Copy existing files from host machine
COPY . ./

# Publish application to the "out" folder
RUN dotnet publish --configuration Release --output out

# Start container by running application DLL
ENTRYPOINT ["dotnet", "out/ipcheck.dll"]
```

```
Get-AzContainerRegistry -ResourceGroupName "MyResourceGroup" -Name "MyRegistry"
```

Topics

- Microsoft identity platform
- Microsoft Authentication Library (MSAL)
- Microsoft Graph
- Authorizing data operations in Azure Storage

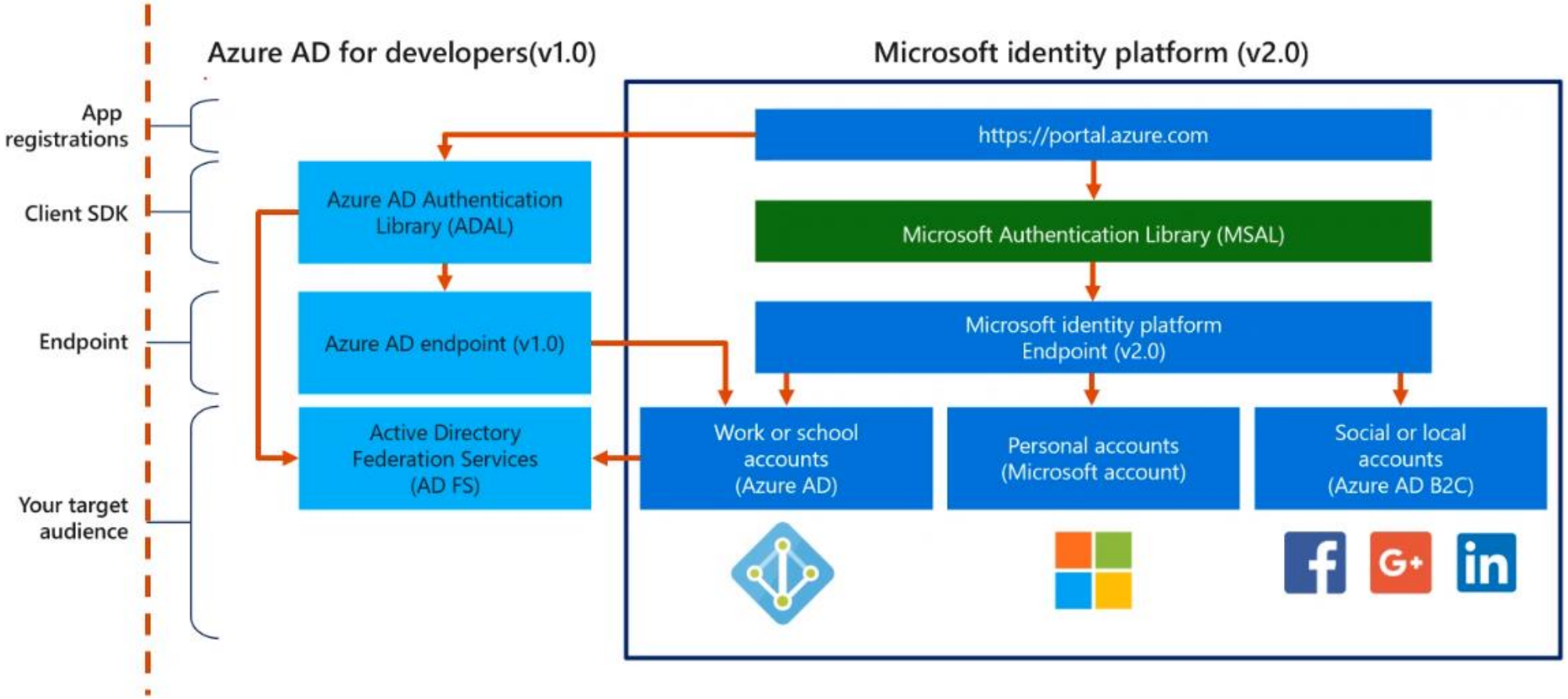
Azure Active Directory



Active Directory Authentication Library (ADAL)

- The library to streamline working with Azure Active Directory from code:
  - Obtains and manages tokens
  - Caches token using a configurable cache
  - Refreshes tokens automatically when they expire
  - Supports asynchronous invocation
- Available in multiple languages such as:
  - C#
  - JavaScript
  - Objective C
  - Java
  - Python

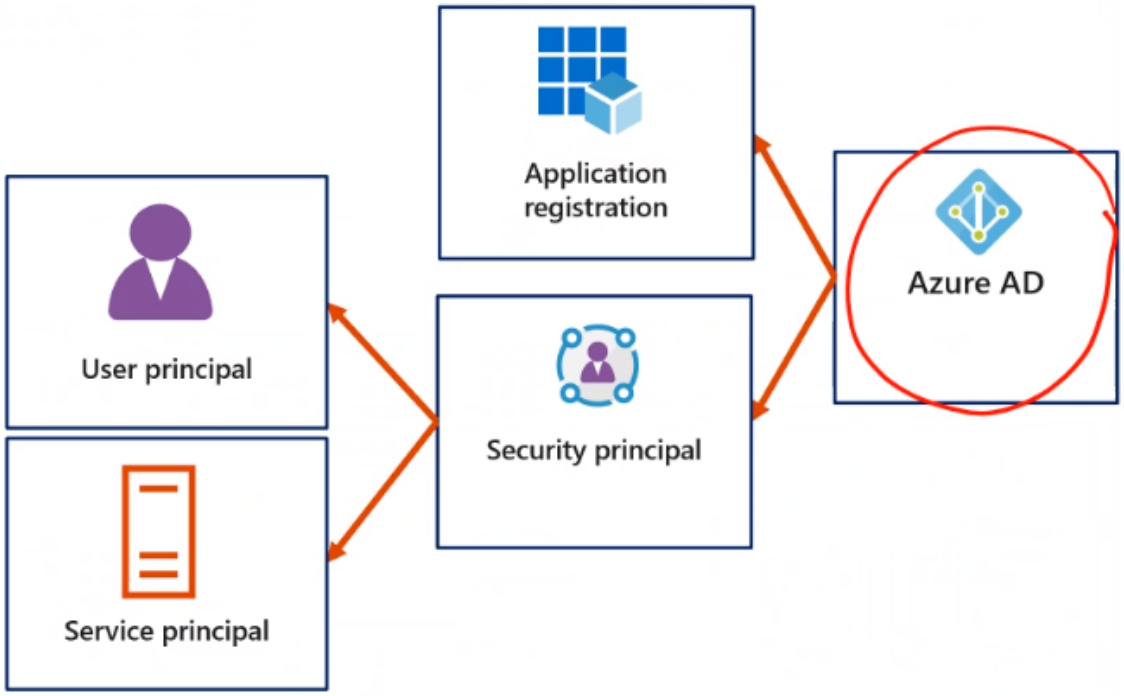
Microsoft identity platform



Objects in Azure AD

Azure AD includes two object types:

- Application registration
- Security principal:
  - User principal
  - Service principal



Application registration

- To outsource authentication to Azure AD, applications must be registered in one or more Azure AD tenants:
  - Single-tenant: common with line-of-business (LOB) applications
  - Multitenant: common with SaaS applications developed by ISVs
- The application registration might include, depending on the type:
  - Application ID URI
  - Reply URL and redirect URI
  - Application ID
  - Key

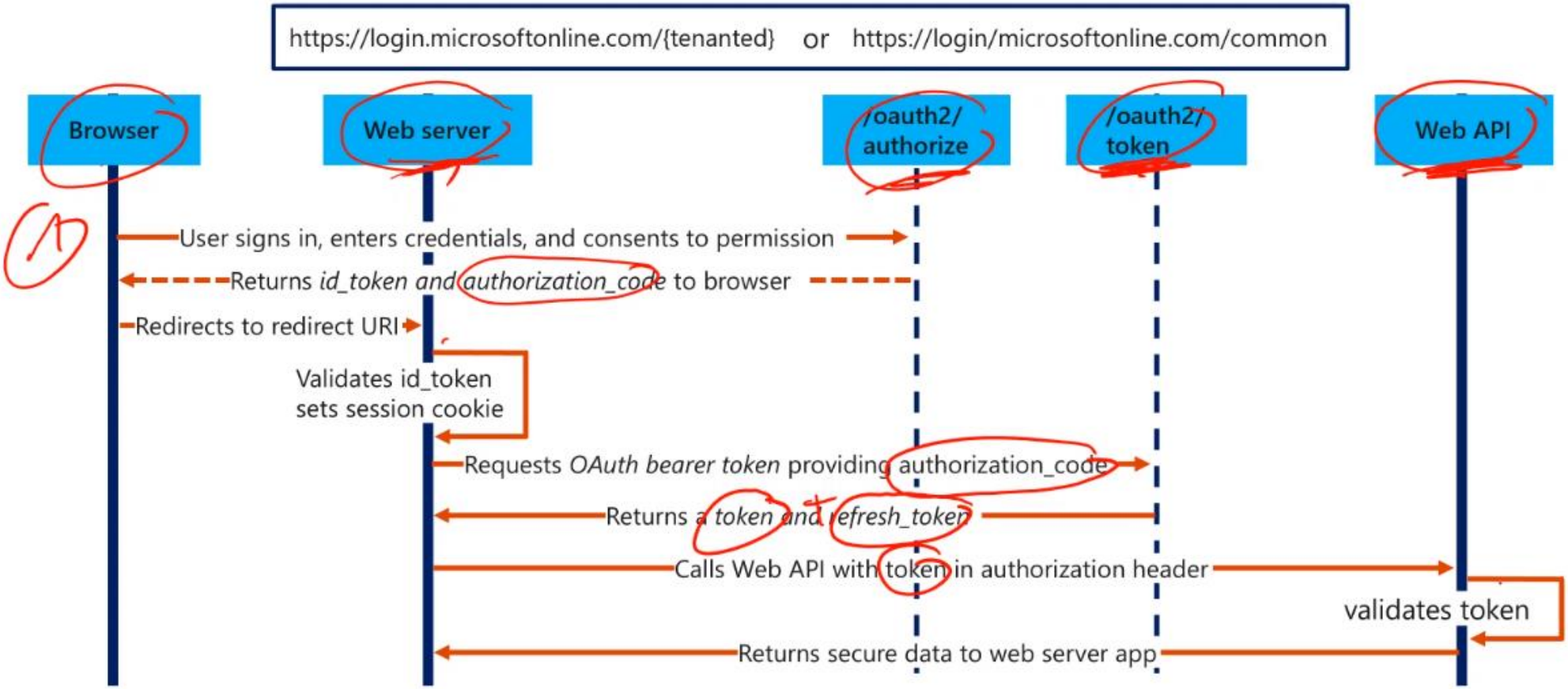
Azure AD

- Management Groups (optional)
  - Subscriptions
    - RG and Resources

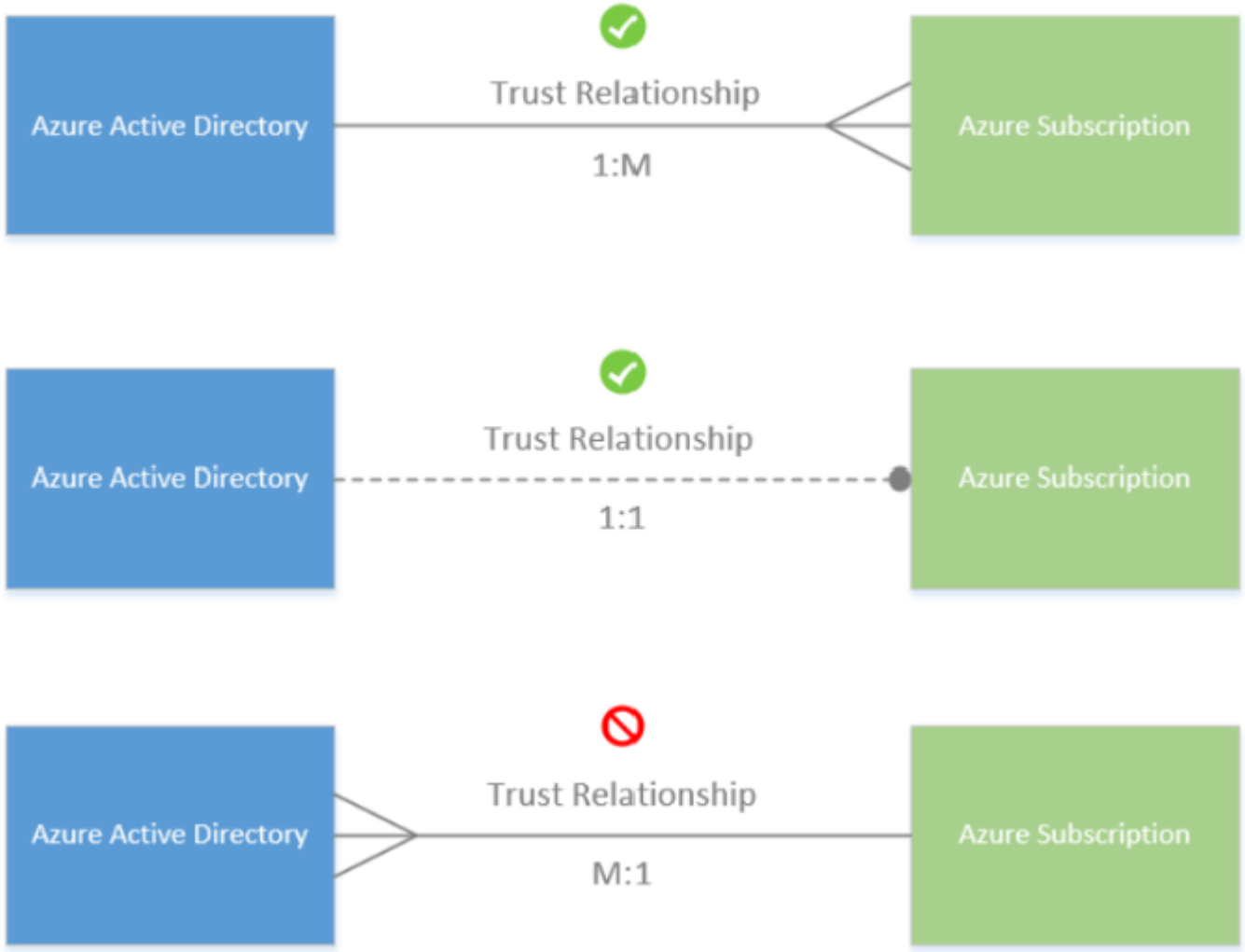
Understanding the OAuth 2.0 implicit grant flow in Azure AD

- The OAuth 2.0 authorization code grant relies on two separate endpoints:
  - The authorization endpoint: used during the user interaction phase
  - The token endpoint: used by a client to exchange the authorization code for an access token and, optionally, refresh tokens
- The OAuth 2.0 implicit grant is a variant of an authorization grant:
  - It allows the client to obtain an access token (and id_token, when using OpenID Connect) directly from the authorization endpoint, without relying on the token endpoint
  - It never returns refresh tokens to the client
  - It is intended for JavaScript applications running in a browser (such as SPAs)
  - It should not be used for:
    - Native clients
    - Web applications that include a back end and consume an API from the back-end code

Authorize access to web applications by using OAuth



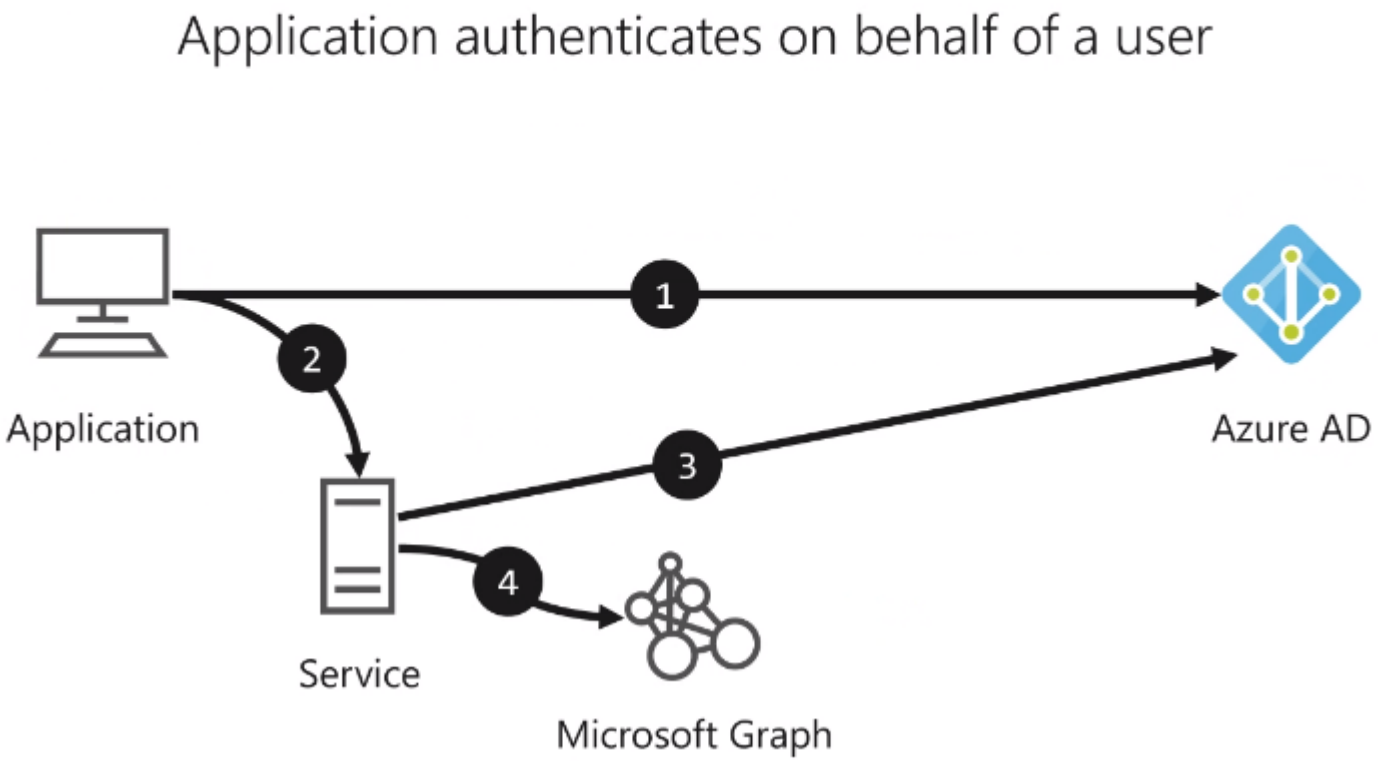
https://docs.microsoft.com/en-us/azure/active-directory/fundamentals/active-directory-how-subscriptions-associated-directory#associate-a-subscription-to-a-directory

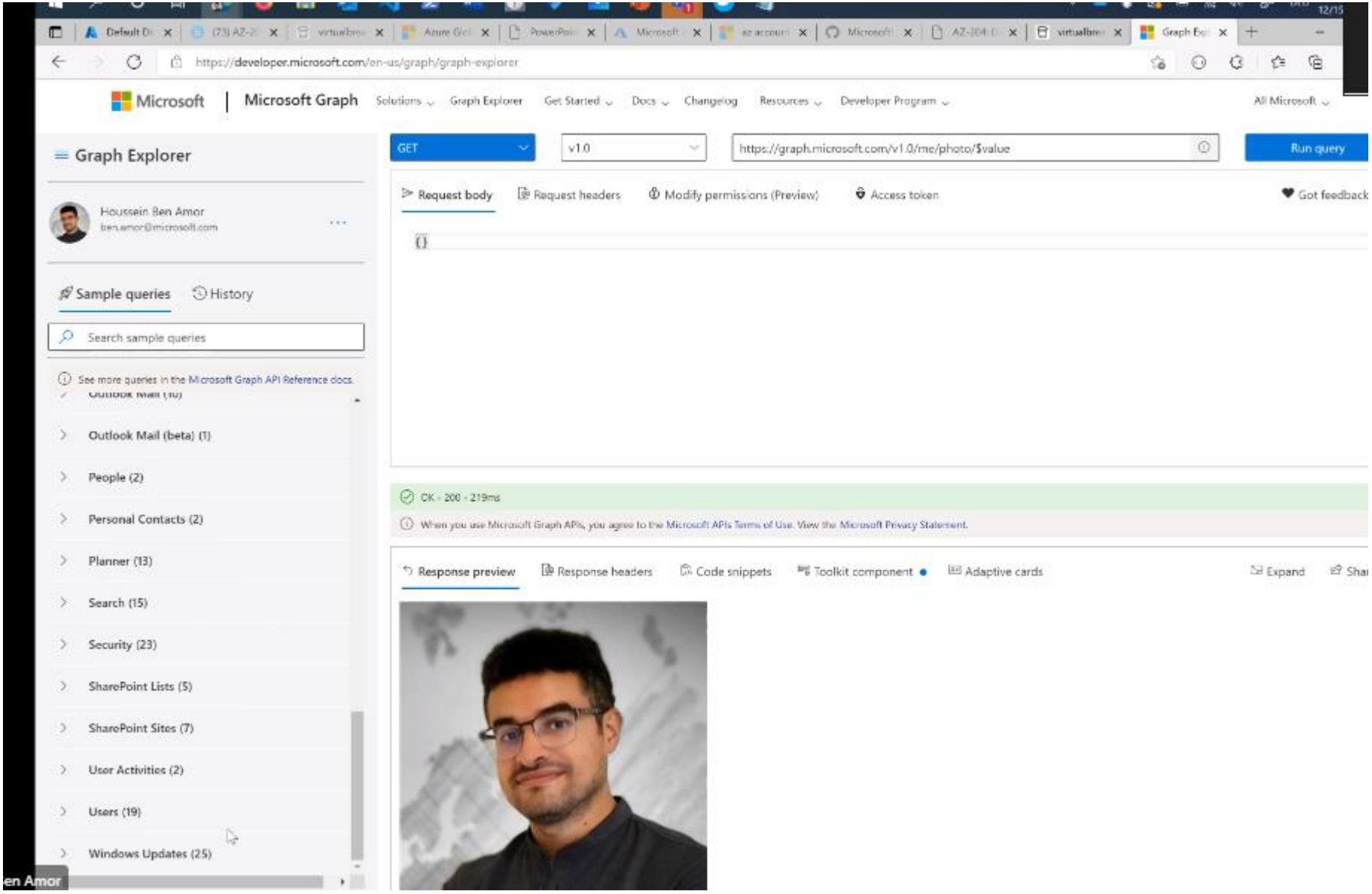


Browser very bad place to store tokens..

Cookie^de saklama, session’da sakla diyor.

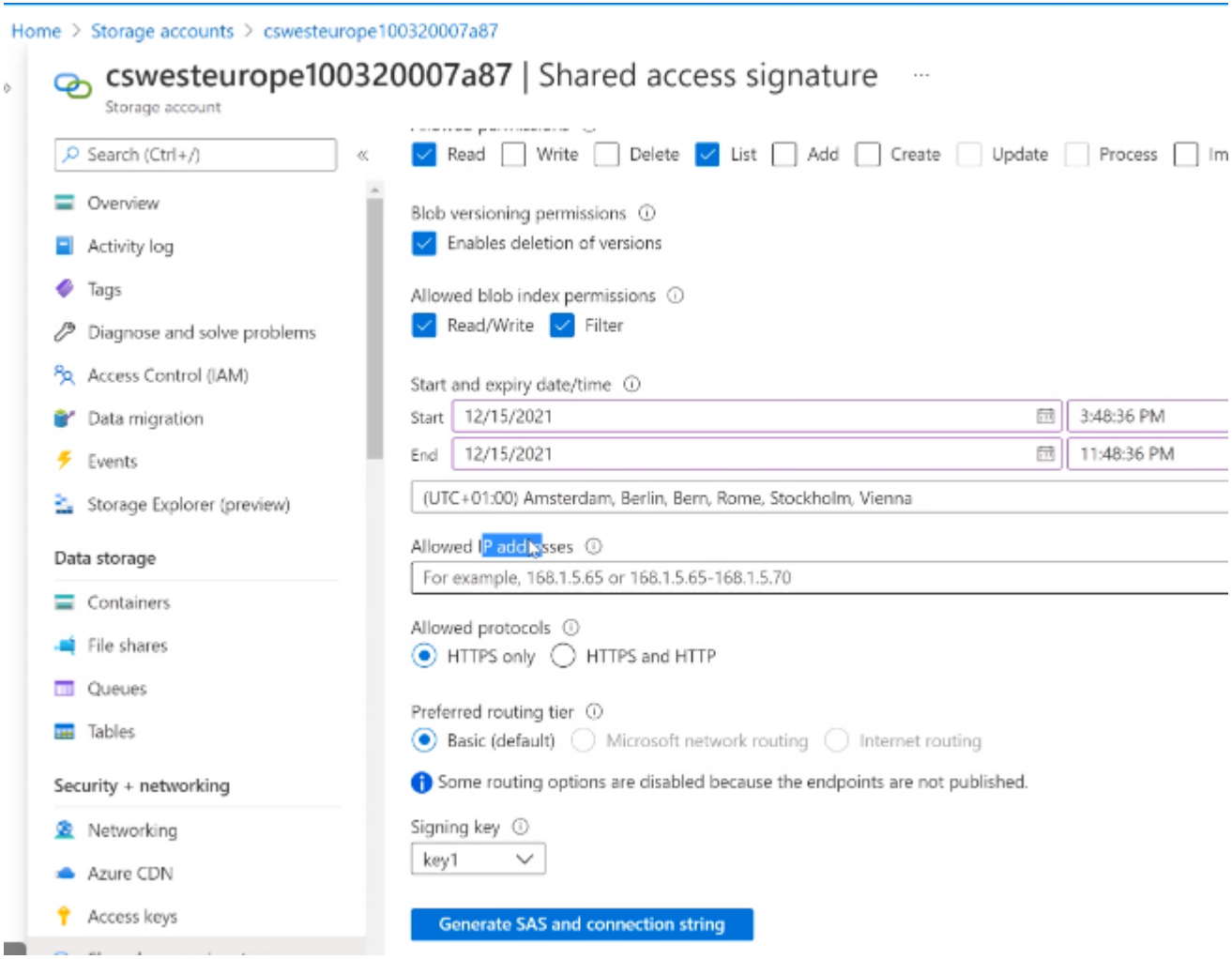
On-Behalf-Of authentication flow




https://developer.microsoft.com/en-us/graph/graph-explorer
Microsoft | Microsoft Graph
Solutions
Graph Explorer
Get Started
Docs
Changelog
Resources
Developer Program
All Microsoft
Graph Explorer
Houssein Ben Amor
ben.amor@microsoft.com
Sample queries
History
Search sample queries
See more queries in the Microsoft Graph API Reference docs.
Outlook Mail (beta) (1)
People (2)
Personal Contacts (2)
Planner (13)
Search (15)
Security (23)
SharePoint Lists (5)
SharePoint Sites (7)
User Activities (2)
Users (19)
Windows Updates (25)
GET
v1.0
https://graph.microsoft.com/v1.0/me/photo/$value
Run query
Request body
Request headers
Modify permissions (Preview)
Access token
Got feedback
{}
OK - 200 - 219ms
When you use Microsoft Graph APIs, you agree to the Microsoft APIs Terms of Use. View the Microsoft Privacy Statement.
Response preview
Response headers
Code snippets
Toolkit component
Adaptive cards
Expand

Container permissions

- There are three levels of container access that are available:
  - **Full public read access:**
    - Enumerate container blobs
    - Read individual blobs
    - Cannot enumerate containers
  - **Public read access for blobs only:**
    - Read individual blobs
  - **No public read access:**
    - No access to blobs, containers, or enumerating contents



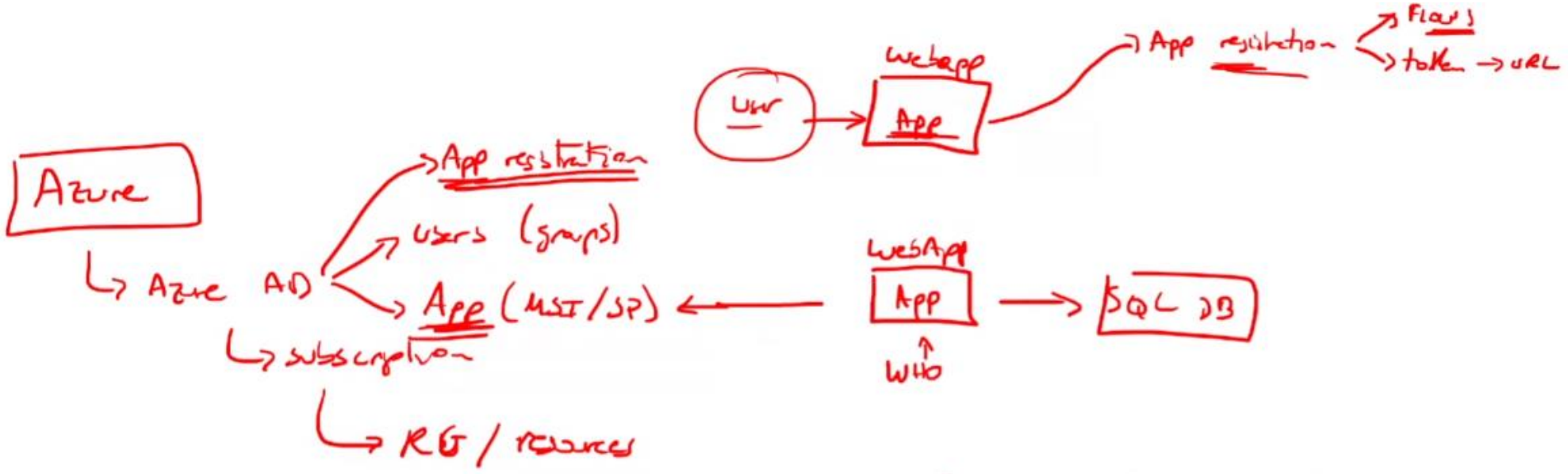
in the end 3 ways to authenticate against Storage account:

- RBAC : for Azure AD user, groups, and managed identities
- SAS: ad hoc or policy based
- access key: full access

Mod 7 → KV, MSI, App Configuration

Mod 8 → API Management (lab)

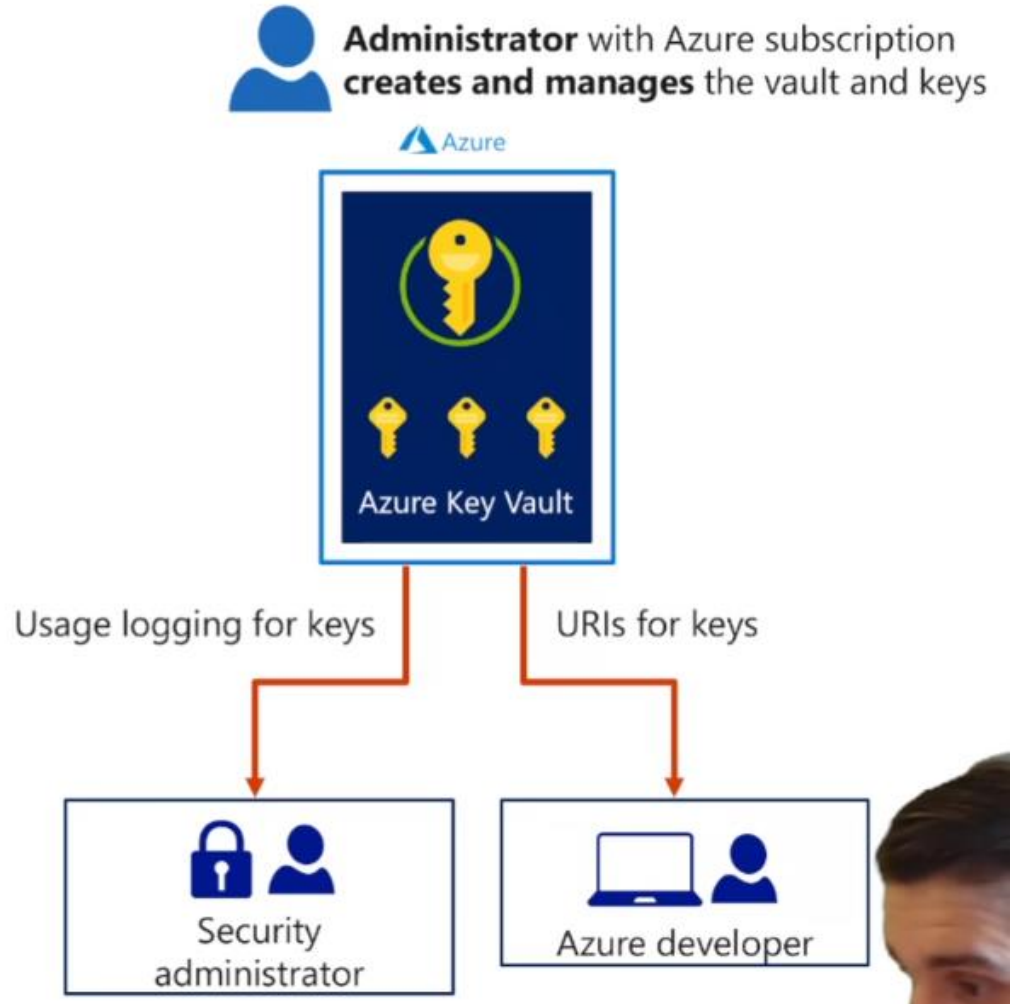
Mod 9 → Event Hub / Event Grid



Lesson 01: Manage keys, secrets, and certificates by using the KeyVault API

Azure Key Vault

- Safeguard cryptographic keys and other secrets that cloud apps and services use:
  - Increase security and control over keys and passwords
  - Applications have no direct access to keys
  - Use FIPS 140-2 Level 2 validated hardware security modules (HSMs)
  - Create and import:
    - Encryption keys
    - API keys
    - Secrets
    - Passwords
    - SSL/TLS certificates



Key Vault concepts

- Vault:
  - Logical group of secrets
- Vault owner:
  - Identity that has full control over the vault
  - Can **grant** other identities consumer (**scoped**) **access** to the vault
- Vault consumer:
  - **Identity** that can **perform actions** on the assets inside the vault (with permission)

Key Vault authentication

- Managed identity:
  - **Assigned identity** for an Azure resource
  - Fastest way to access the vault from a service without sharing or exposing credentials
- Service principal:
  - Can provide **certificate or secret**

Key Vault authentication

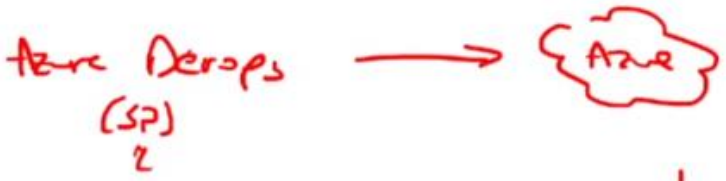
- Managed identity:
  - **Assigned identity** for an Azure resource
  - Fastest way to access the vault from a service without sharing or exposing credentials
- Service principal:
  - Can provide **certificate or secret**

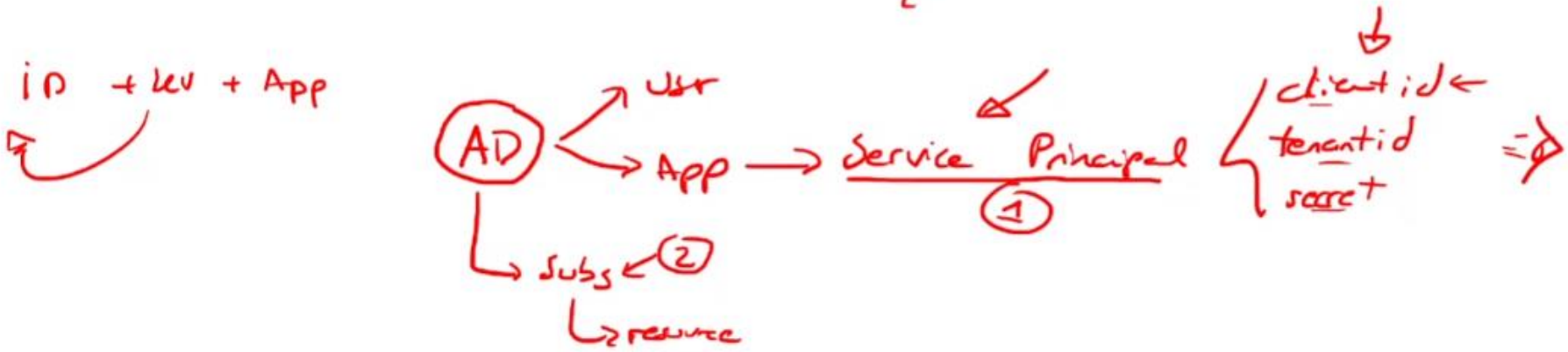
Key Vault authentication

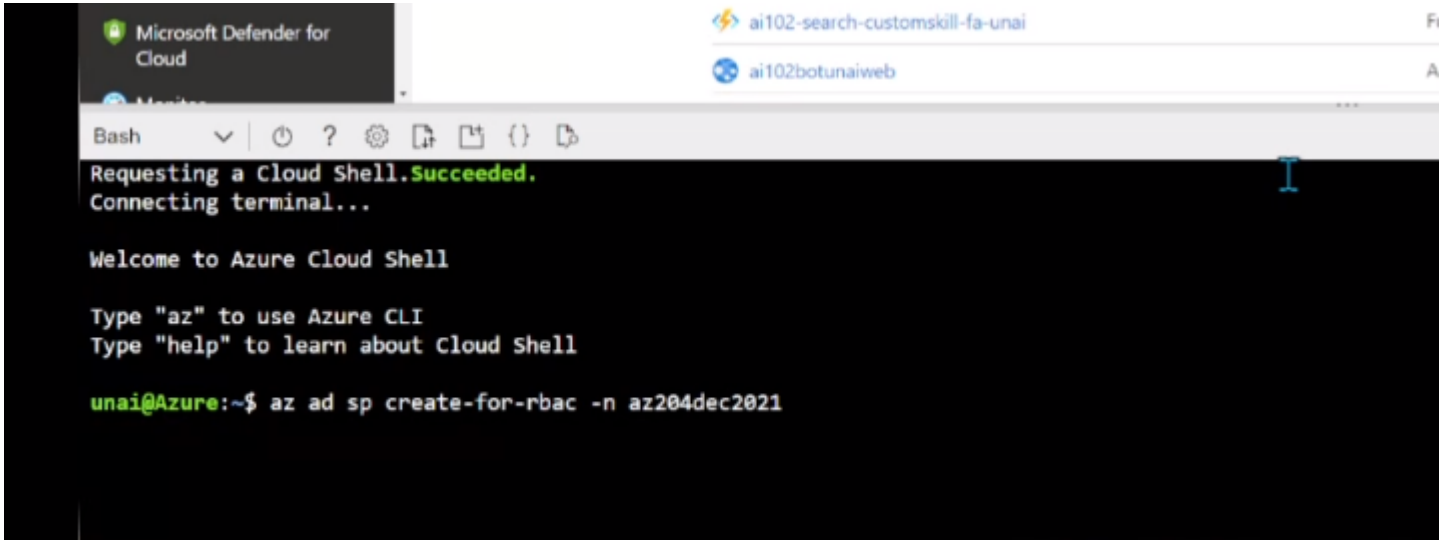
- Managed identity:
  - **Assigned identity** for an Azure resource
  - Fastest way to access the vault from a service without sharing or exposing credentials
- Service principal:
  - Can provide **certificate or secret**

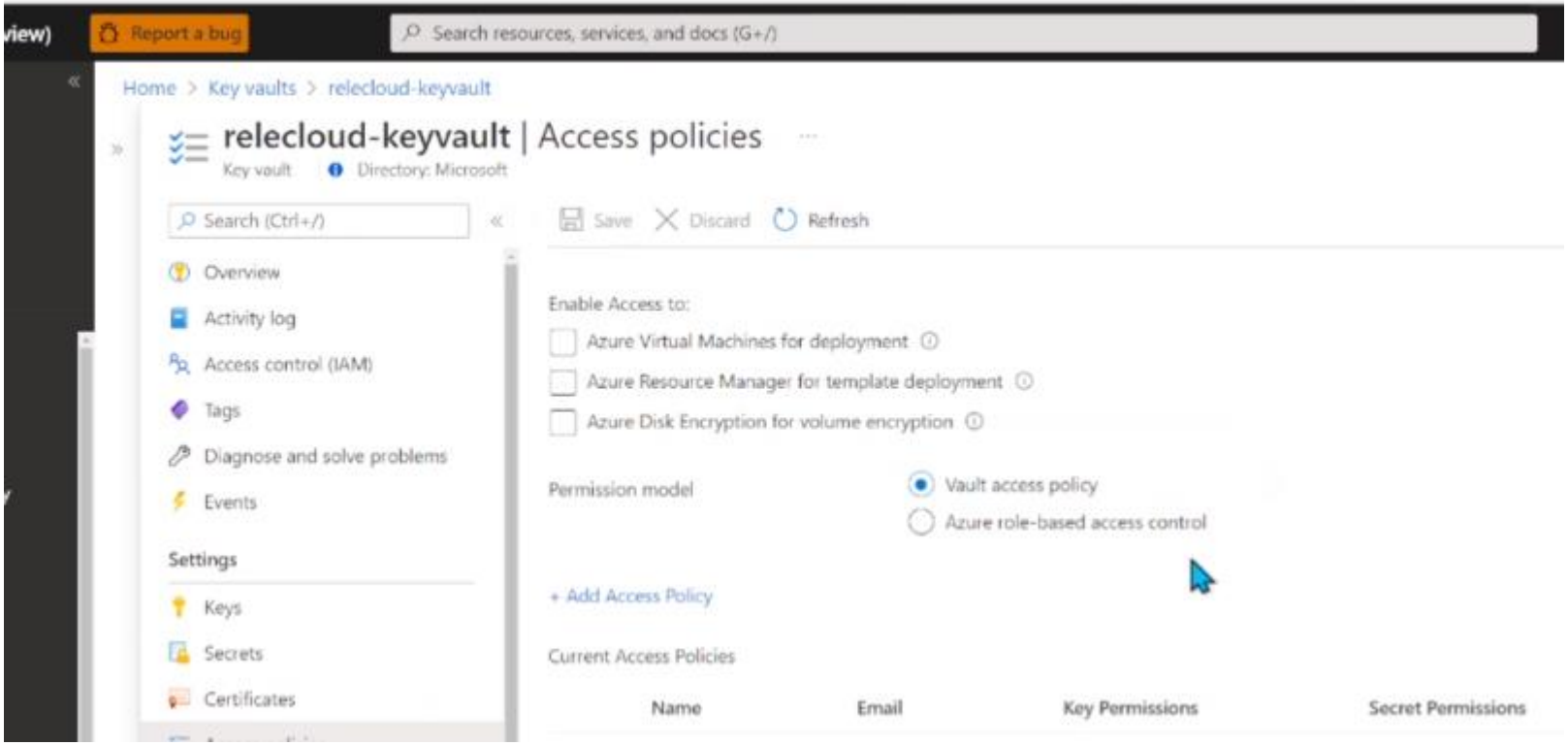
Key Vault authentication

- Managed identity:
  - **Assigned identity** for an Azure resource
  - Fastest way to access the vault from a service without sharing or exposing credentials
- Service principal:
  - Can provide **certificate or secret**






Microsoft Defender for Cloud
ai102-search-customskill-fa-unai
ai102botunaiweb
Bash
Requesting a Cloud Shell.Succeeded.
Connecting terminal...
Welcome to Azure Cloud Shell
Type "az" to use Azure CLI
Type "help" to learn about Cloud Shell
unai@Azure:~$ az ad sp create-for-rbac -n az204dec2021


Report a bug
Search resources, services, and docs (G+/)
Home > Key vaults > relecloud-keyvault
relecloud-keyvault | Access policies
Key vault
Directory: Microsoft
Search (Ctrl+/)
Save
Discard
Refresh
Overview
Activity log
Access control (IAM)
Tags
Diagnose and solve problems
Events
Settings
Keys
Secrets
Certificates
Enable Access to:
Azure Virtual Machines for deployment
Azure Resource Manager for template deployment
Azure Disk Encryption for volume encryption
Permission model
Vault access policy
Azure role-based access control
+ Add Access Policy
Current Access Policies
Name
Email
Key Permissions
Secret Permissions

Create Key Vault secret by using Azure CLI

```
# Create resource group
az group create --name SecurityGroup --location westus
```

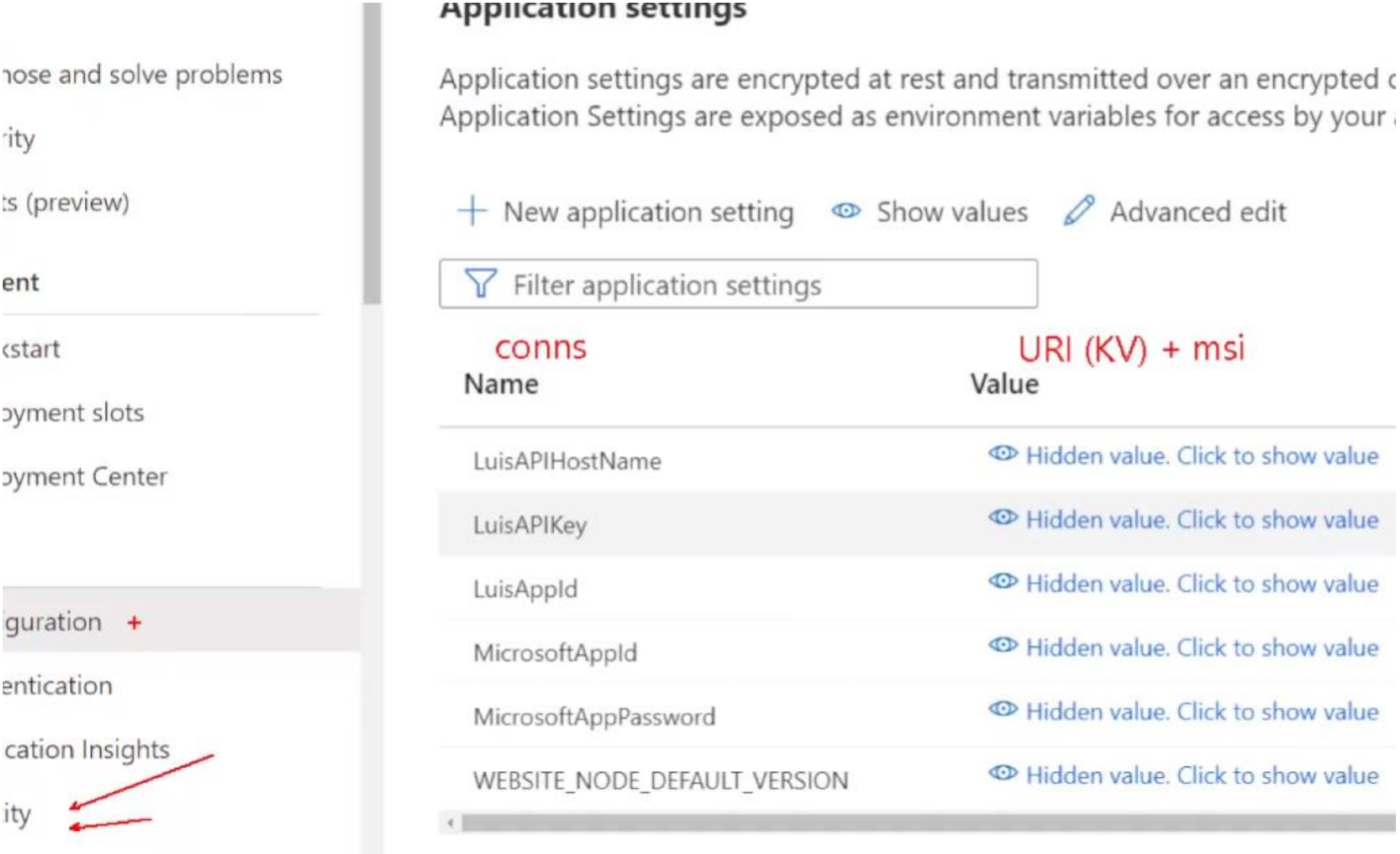
NOUN VERB →

```
# Create Key Vault resource
az keyvault create --name contosovault --resource-group SecurityGroup --location westus

# Set secret in Key Vault
az keyvault secret set --vault-name contosovault --name DatabasePassword --value
'Pa55w.rd'

# Show value of secret in Key Vault
az keyvault secret show --vault-name contosovault --name DatabasePassword
```

Bu şekilde olmaz.

Get Key Vault secret by using C# (SDK v3)

```
string secretUri = "https://contoso-vault2.vault.azure.net/secrets/example/0932840309";
var securityToken = "...";

// Create Key Vault client
var client = new KeyVaultClient(
    new KeyVaultClient.AuthenticationCallback(securityToken)
);

// Get secret
var secretBundle = await client.GetSecretAsync(secretUri);

// Get value of secret
var secret = secretBundle.Value;
```


hose and solve problems
rity
ts (preview)
ent
kstart
oyment slots
oyment Center
guration +
entication
cation Insights
ity
Application settings
Application settings are encrypted at rest and transmitted over an encrypted c
Application Settings are exposed as environment variables for access by your
New application setting
Show values
Advanced edit
Filter application settings
conns
URI (KV) + msi
Name
Value
LuisAPIHostName
Hidden value. Click to show value
LuisAPIKey
Hidden value. Click to show value
LuisAppId
Hidden value. Click to show value
MicrosoftAppId
Hidden value. Click to show value
MicrosoftAppPassword
Hidden value. Click to show value
WEBSITE_NODE_DEFAULT_VERSION
Hidden value. Click to show value

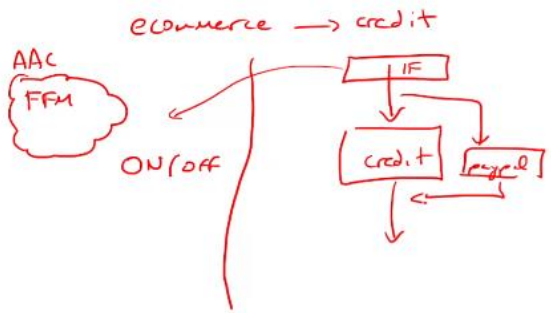
Burada koptum

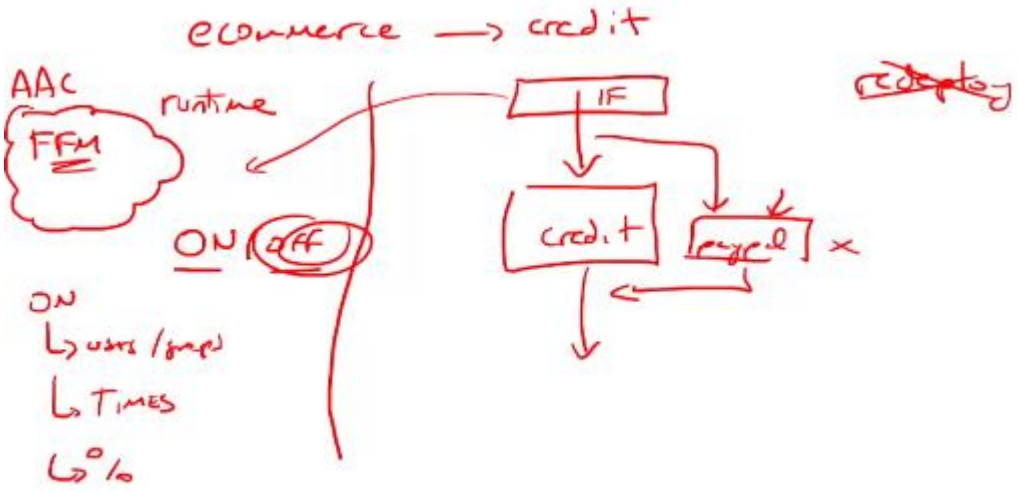
Azure App Configuration

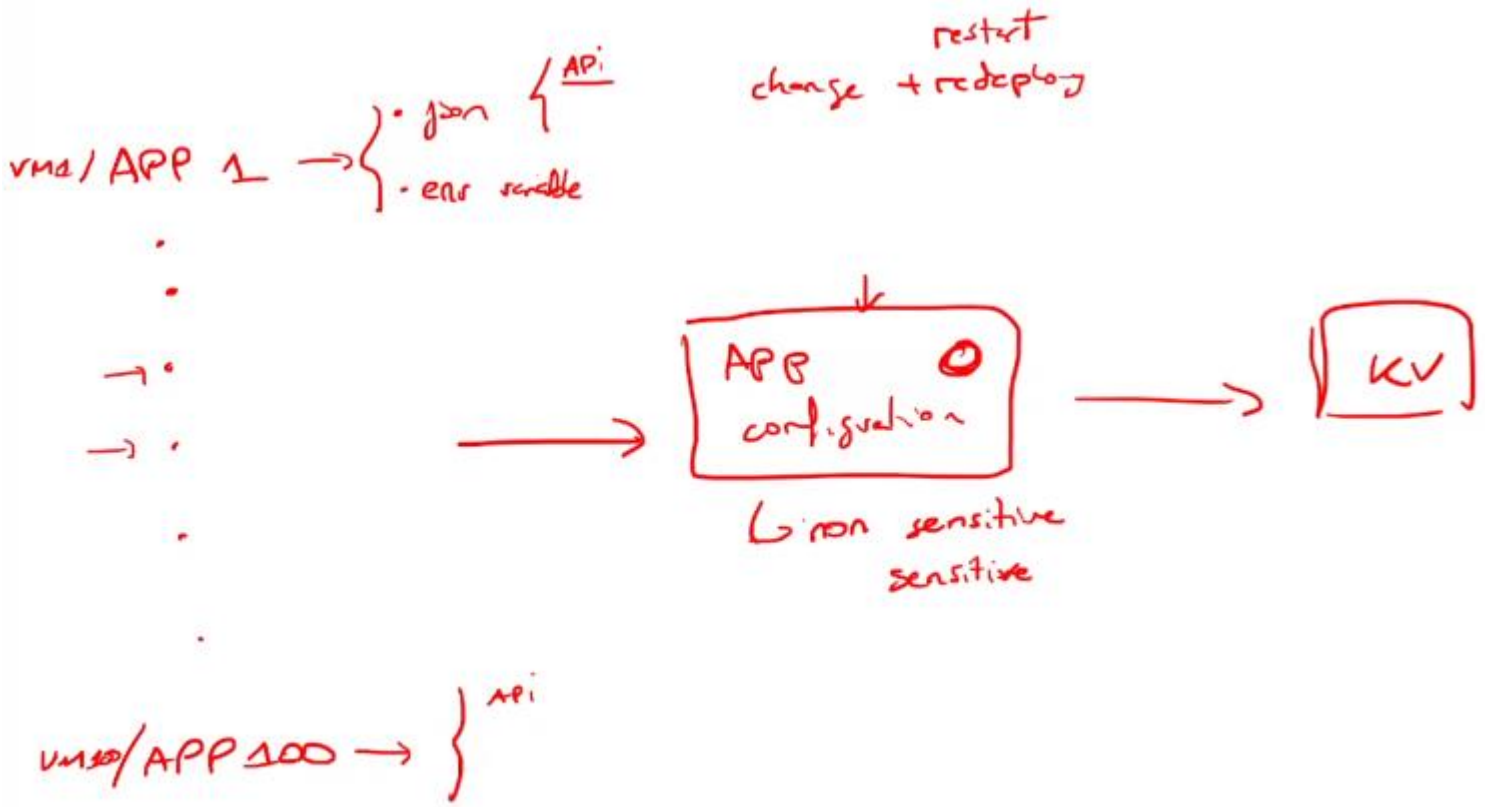
- Service to centrally **manage application settings and feature flags**
- Dedicated UI for **feature flag management**
- Enhanced security through **Azure-managed identities**
- Complete data encryptions at rest or in transit

Programming languages and framework	How to connect
.NET Core and ASP.NET Core	App Configuration provider for .NET Core
.NET Framework and ASP.NET	App Configuration builder for .NET
Java Spring	App Configuration client for Spring Cloud
Others	App Configuration REST API

Feature flag





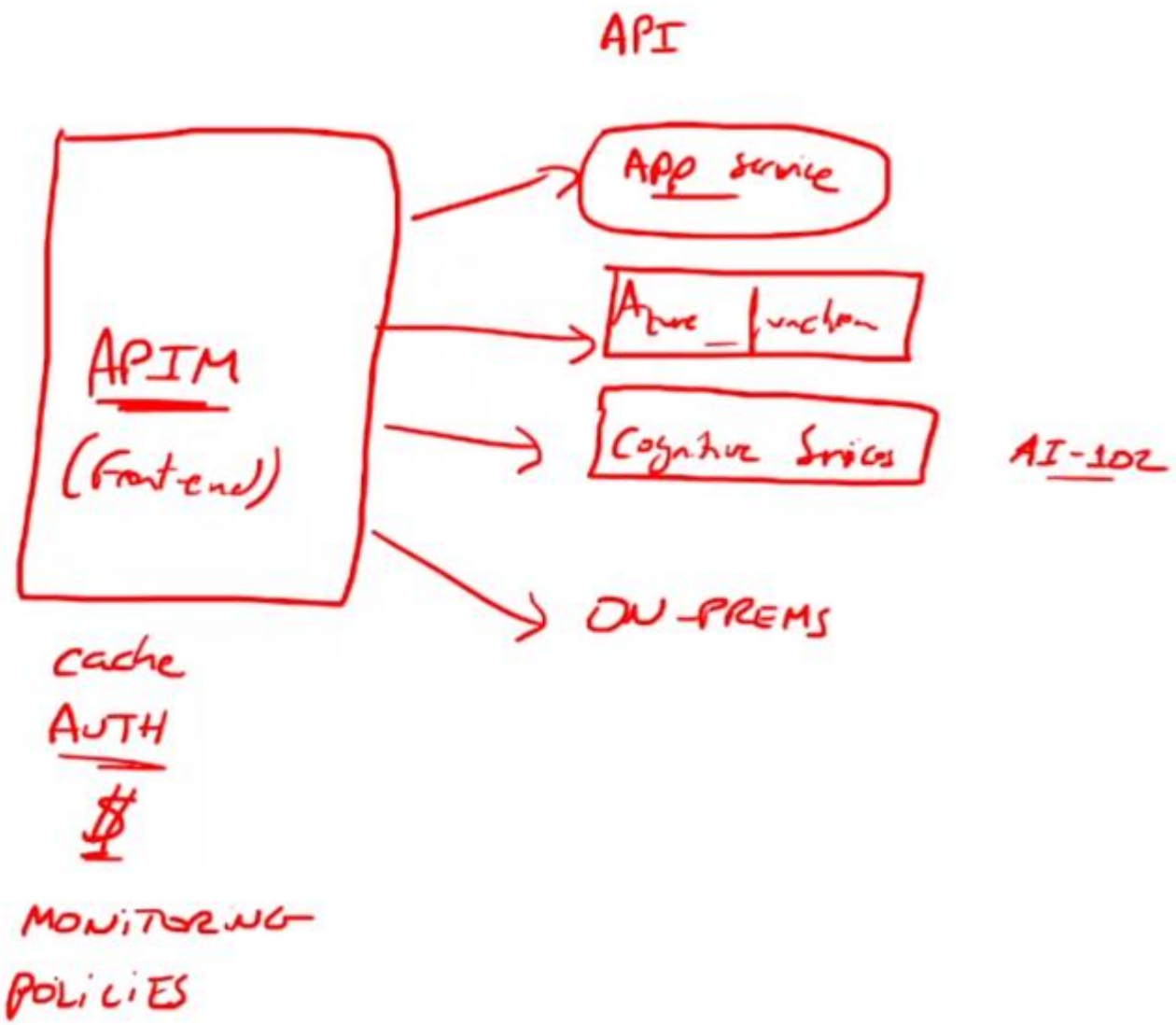


Restart yapmadan config uyarlaması

https://docs.microsoft.com/en-us/azure/azure-app-configuration/enable-dynamic-configuration-aspnet-core?tabs=core5x

https://docs.microsoft.com/en-us/azure/azure-app-configuration/howto-integrate-azure-managed-service-identity?tabs=core2x

Azure App Configuration

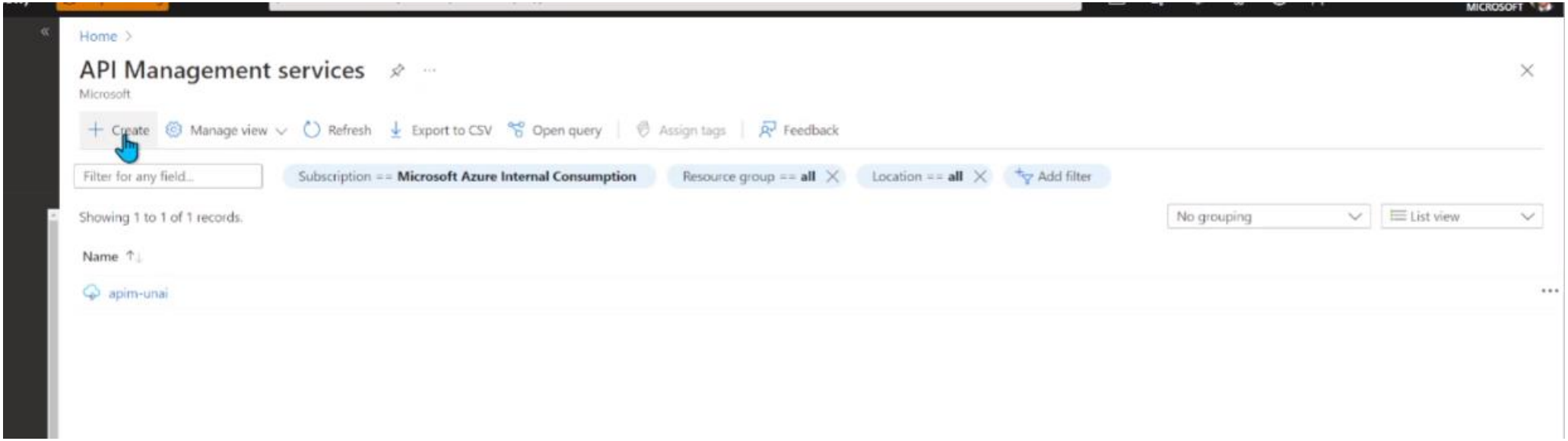

API
APIM
(Front-end)
App service
Azure Function
Cognitive Srvics
AI-102
ON-PREMS
Cache
AUTH
MONITORING
POLICIES

Cache

Auth.

Terminology

- API:
  - A **HTTP service** that you implement with your business **logic**
- Product:
  - One or more APIs, linked to → **usage quota and terms of use**
- Operation:
  - A specific operation in the front-end API that correlates to a **specific request/response** from the backend API


Home
API Management services
Microsoft
Create
Manage view
Refresh
Export to CSV
Open query
Assign tags
Feedback
Filter for any field...
Subscription == Microsoft Azure Internal Consumption
Resource group == all
Location == all
Add filter
Showing 1 to 1 of 1 records.
No grouping
List view
Name
apim-unai

Create API Management

Select the subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * — Microsoft Azure Internal Consumption

Resource group * —

Create new

Instance details

Region * — West US

Resource name *

The value must not be empty.

Organization name *

Administrator email *

Pricing tier

API Management pricing tiers vary in computing capacity per unit and the offered feature set - for example, support for virtual networks, multi-regional deployments, or self-hosted gateways. To accommodate more API requests, consider adding API Management service units instead.
Learn more

Pricing tier — Developer (no SLA)

- Developer (no SLA)
- Basic (99.95% SLA)
- Standard (99.95% SLA)
- Premium (99.95% or 99.99% SLA)
- Consumption (99.95% SLA)

The Developer tier of API Management service may experience intermittent out

Review + create | < Previous | Next : Monitoring >

Understand the full billing model

As you use Azure resources with API Management, you incur costs, or billable meters. Azure resource usage unit costs vary by:

- Time intervals (seconds, minutes, hours, and days)
- Unit usage (bytes, megabytes, and so on)
- Number of transactions

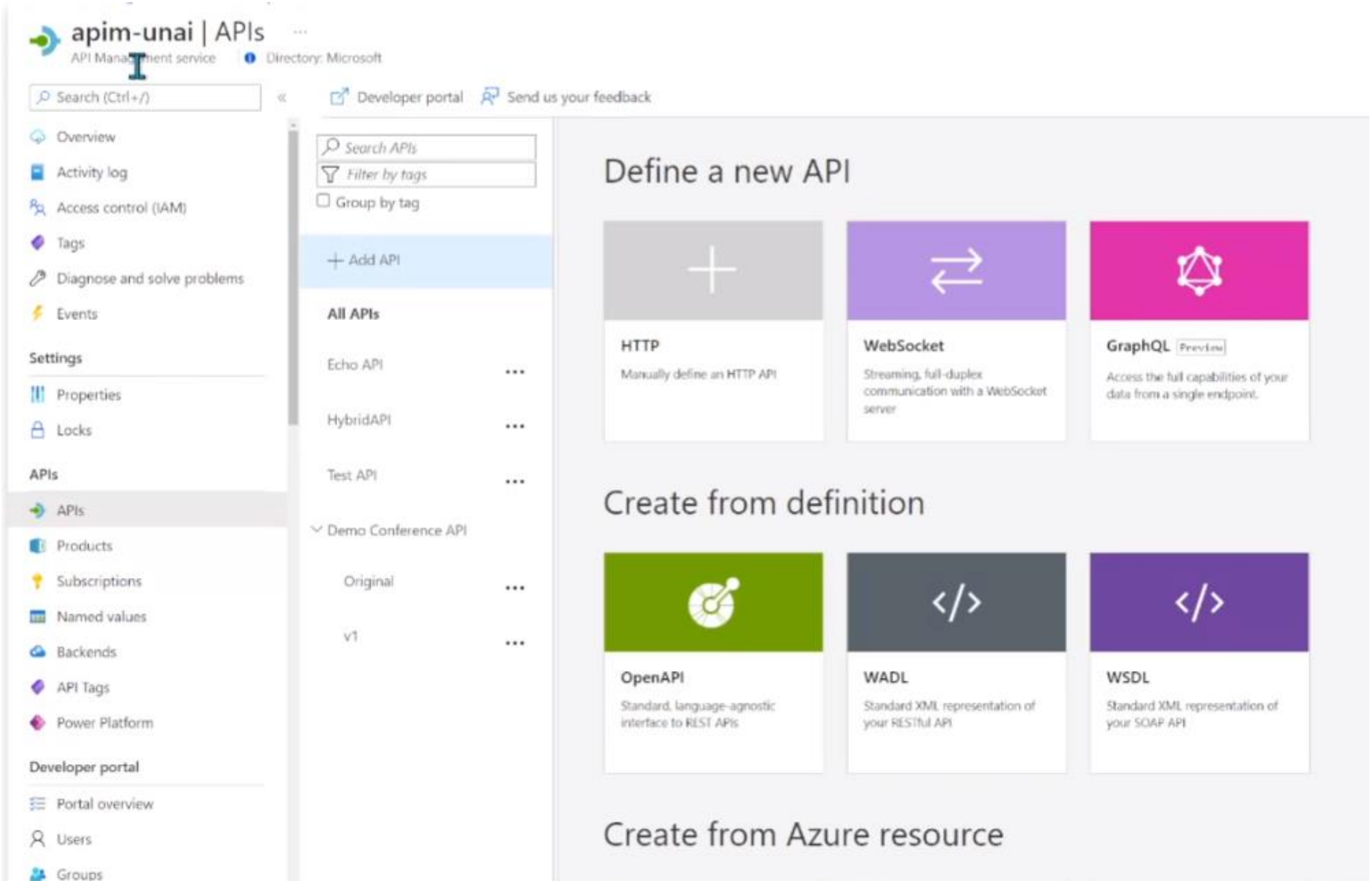
How you're charged for API Management

When you create or use Azure resources with API Management, you'll get charged based on tiers you're working in. Learn more about choosing the best tier for you.

Tiers	Description
Consumption	Incurs no fixed costs. You are billed based on the number of API calls to the service above a certain threshold.
Developer, Basic, Standard, and Premium	Incur monthly costs, based on the number of units and self-hosted gateways. Self-hosted gateways are free for the Developer tier. Upgrade to a different service tier at any time.

You may also incur additional charges when you use other Azure resources with API Management, like virtual networks, availability zones, and multi-region writes. At the end of your billing cycle, the charges for each meter are summed. Your bill or invoice shows a section for all API Management costs. There's a separate line item for each meter.

Self-Dokümantasyonu var.

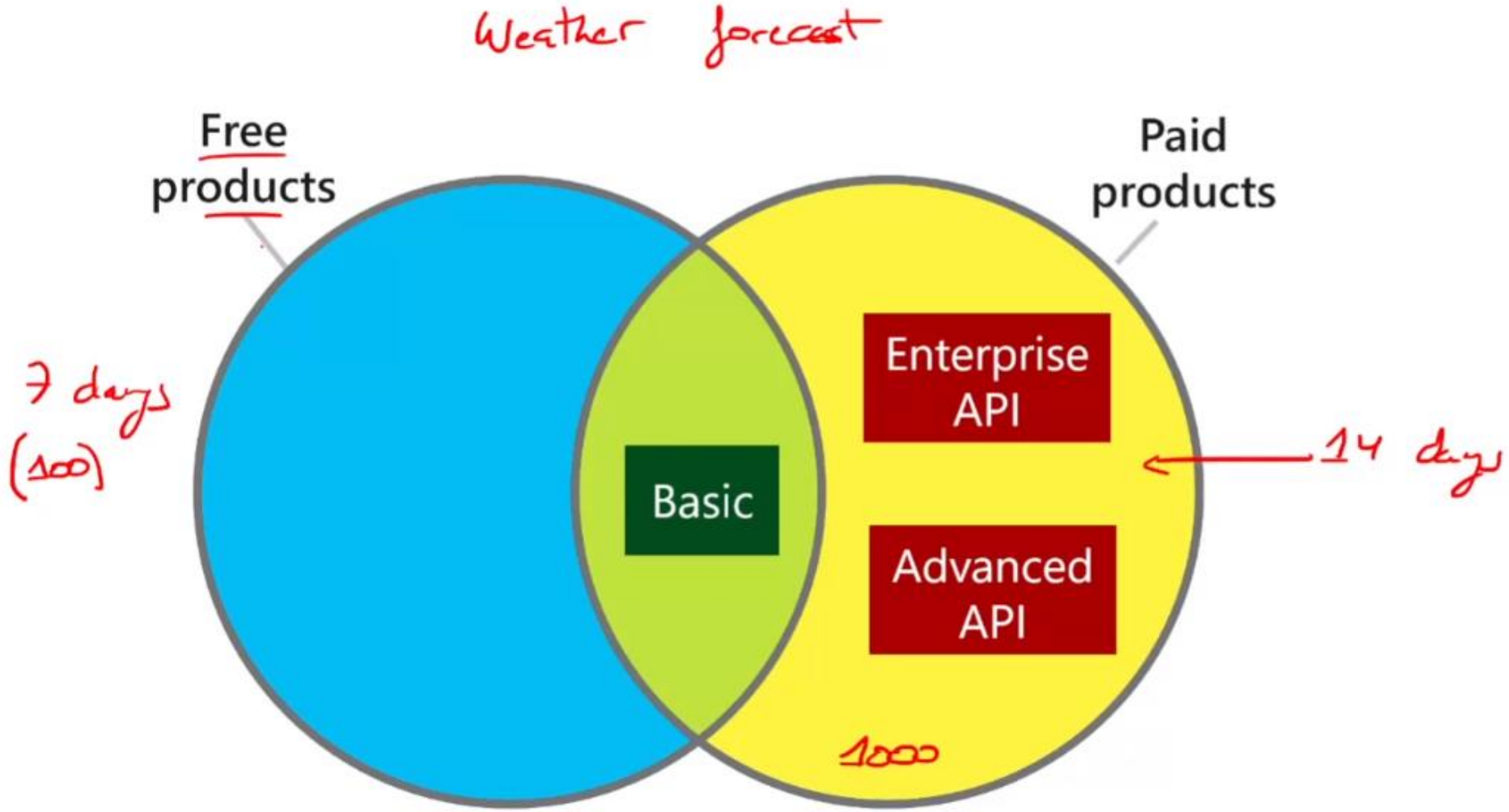


REV de URL aynı, farklı bir versiyon çalışıyor sadece

Products

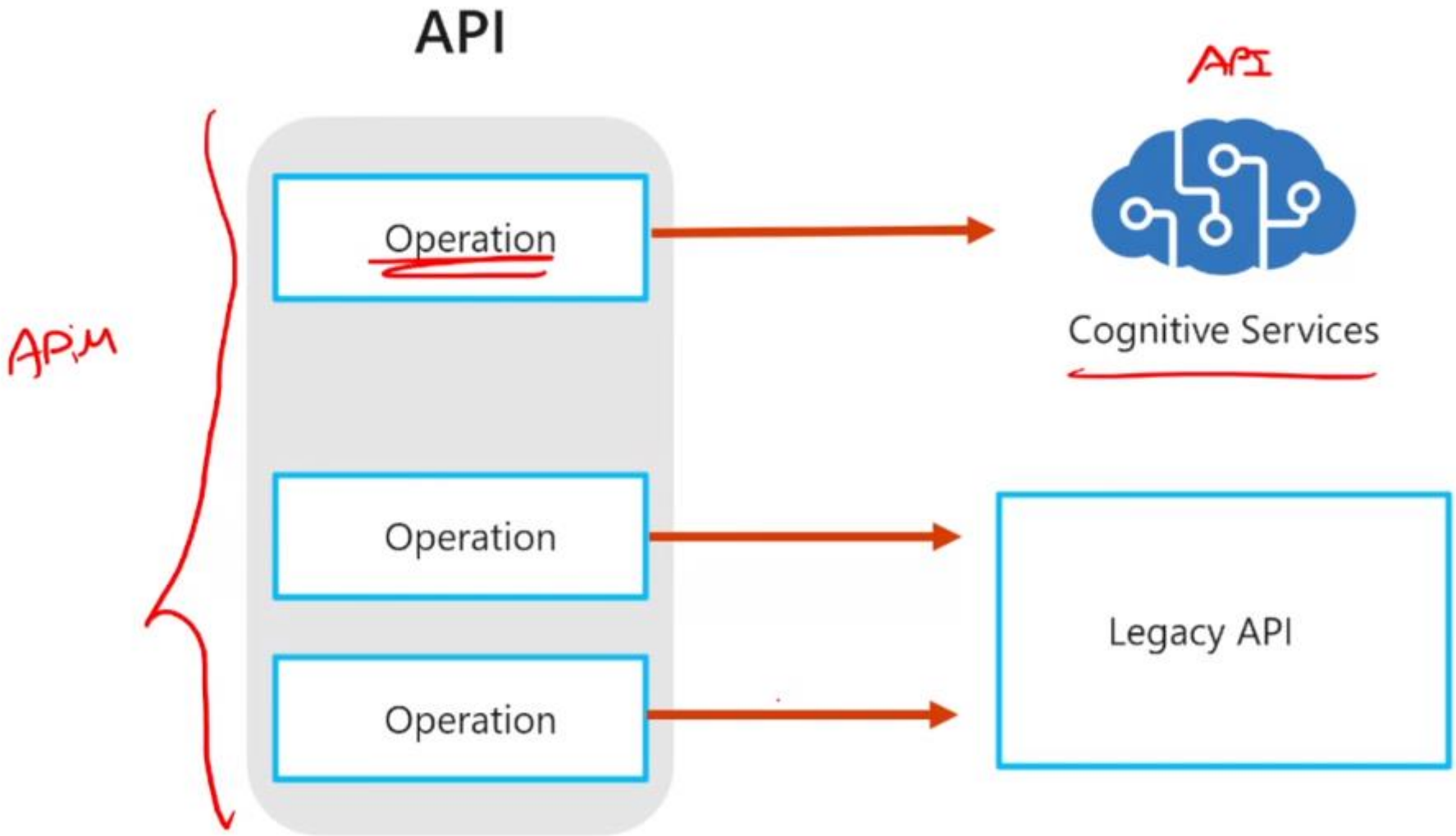
- Contains **one or more APIs** in a package
- Products can be **open or protected**:
  - Open products are free to use without any subscription
  - Protected products must be subscribed to before use
- When a product **is ready** for developers, it can be **published** for use

Products and APIs

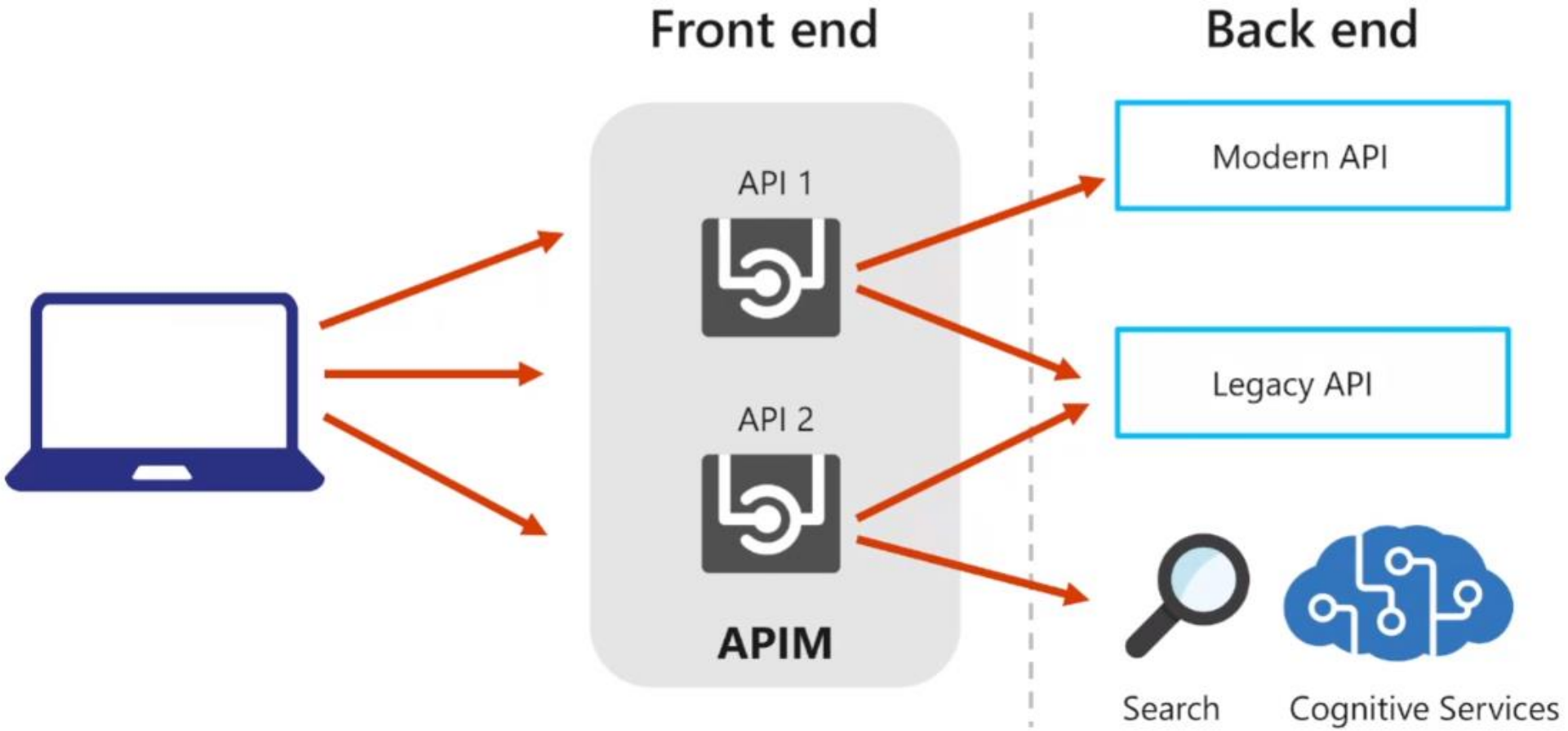


https://docs.microsoft.com/en-us/azure/api-management/api-management-howto-protect-backend-with-aad

APIs and operations



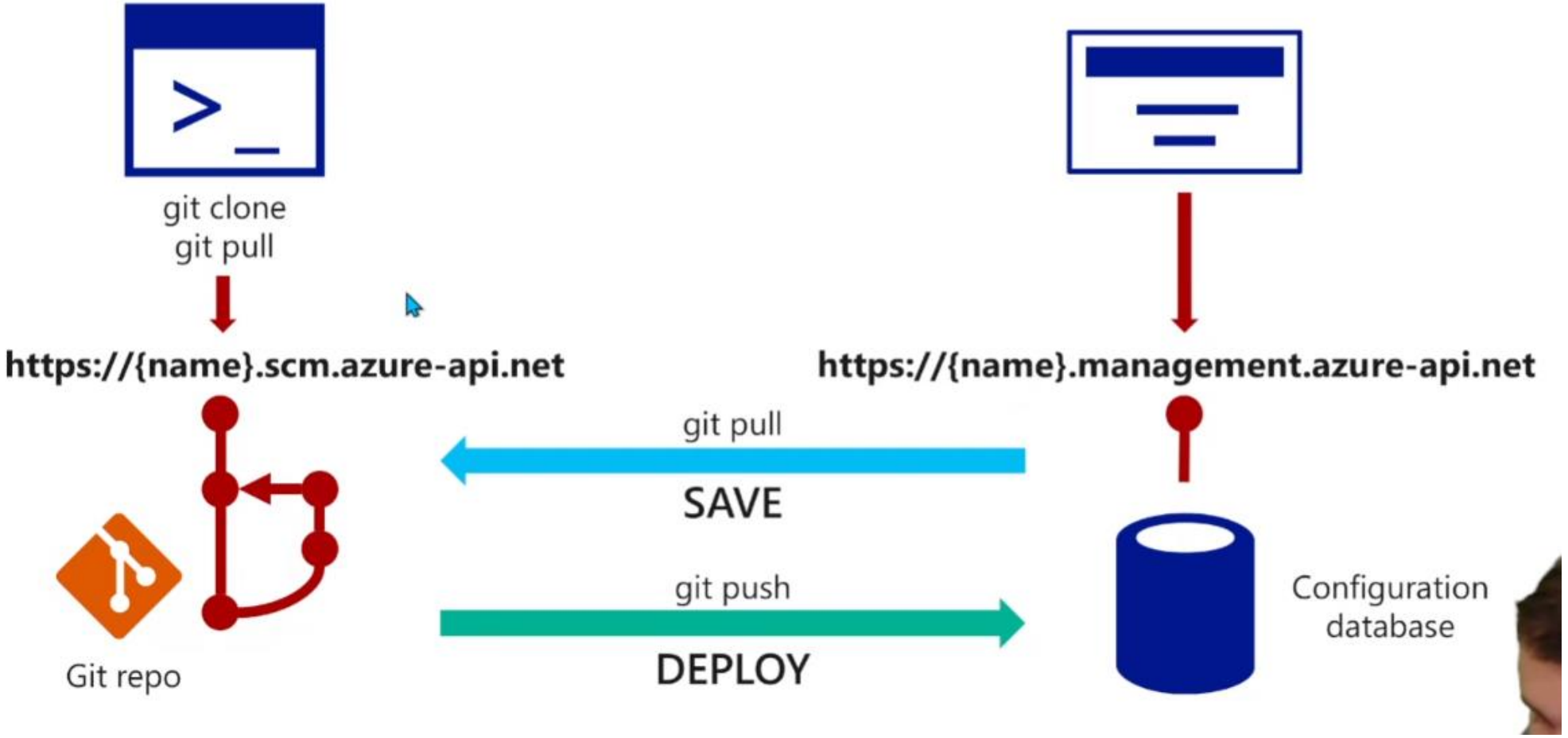
Back-end and front-end APIs



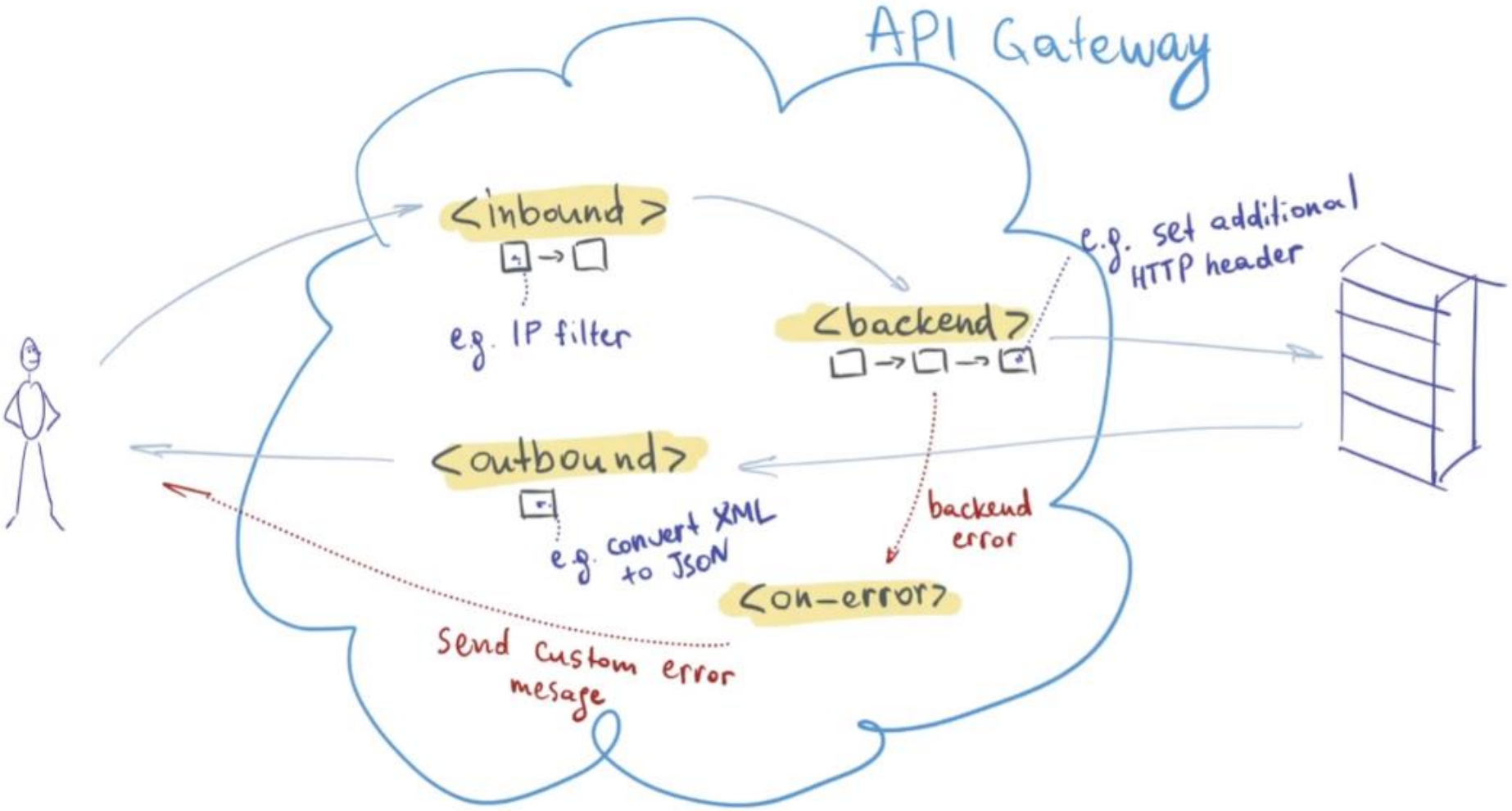
API Management instance overview

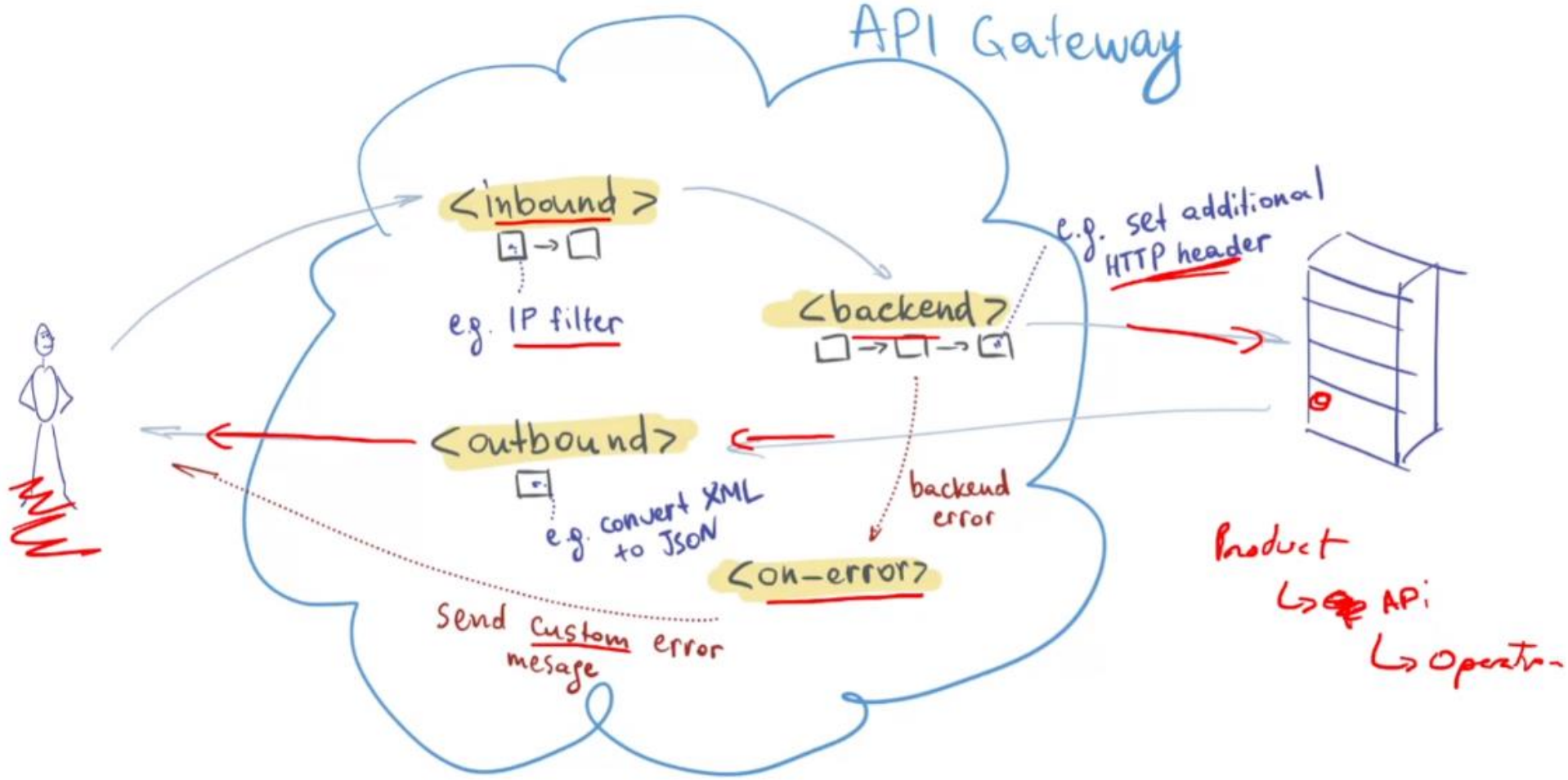
- Create and manage APIs
- Each API contains one or more sets of **operations**
- Operations are configurable, granting **control** over:
  - **URL mapping**
  - Query and path parameters
  - **Request and response content**
  - Operation response **caching**

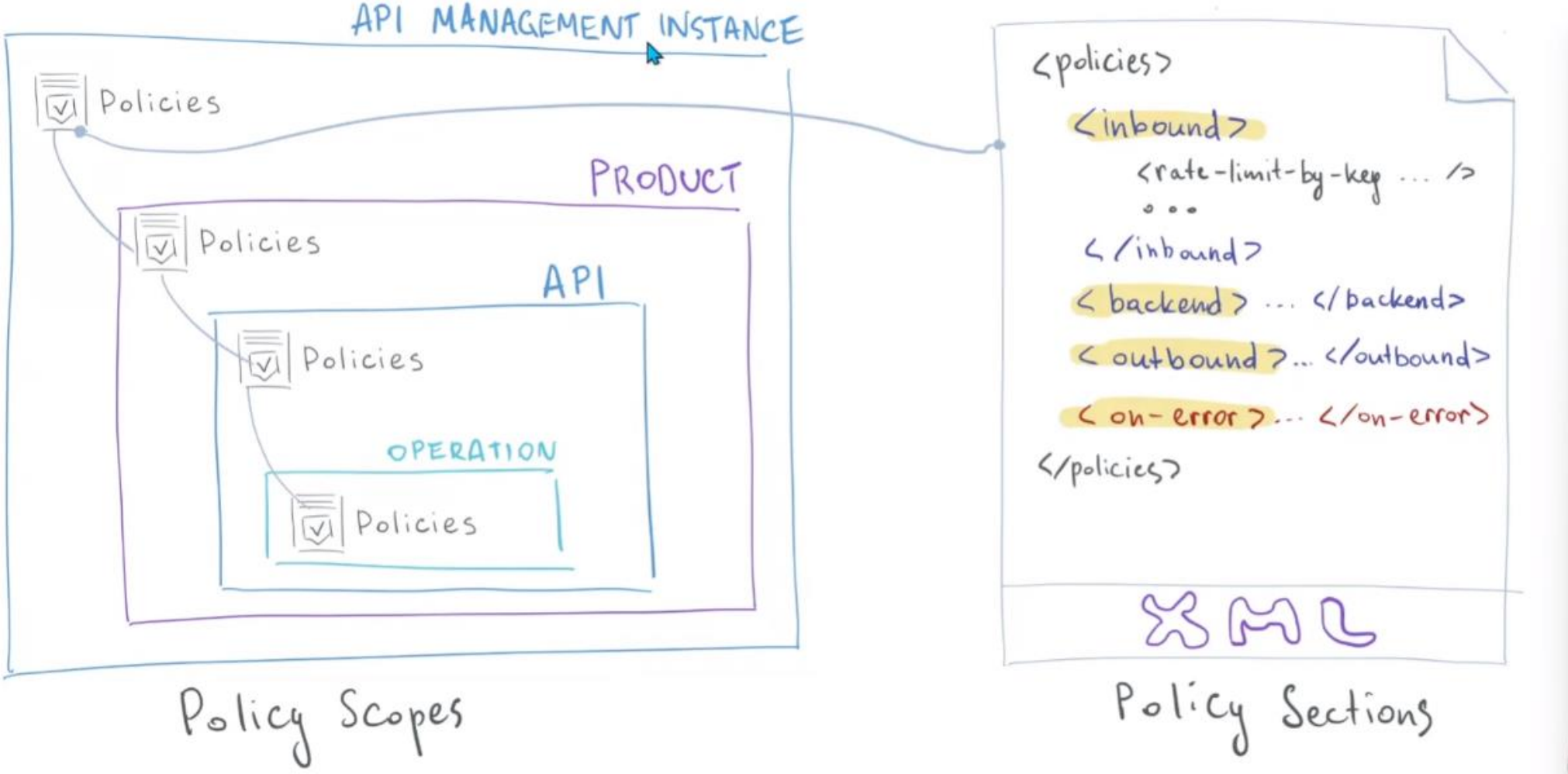
Manage using Git



https://www.maxivanov.io/azure-api-management-visually-explained/




API Gateway
<inbound>
e.g. IP filter
<backend>
e.g. set additional HTTP header
<outbound>
e.g. convert XML to JSON
backend error
<on-error>
Send custom error mesage
Product
API
Operation

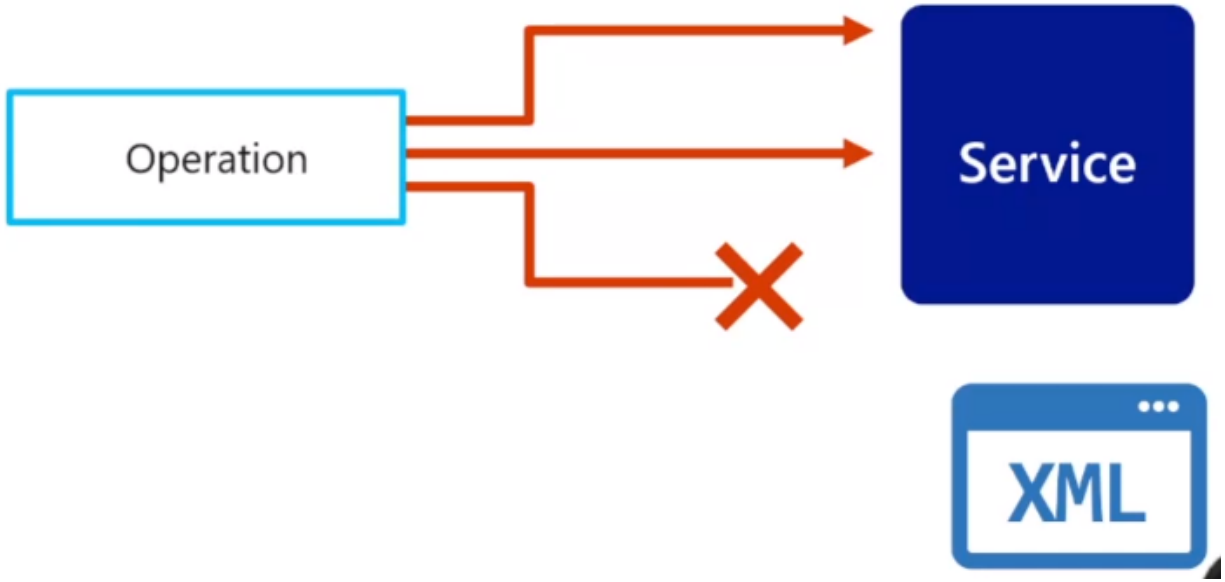


https://www.maxivanov.io/azure-api-management-visually-explained/

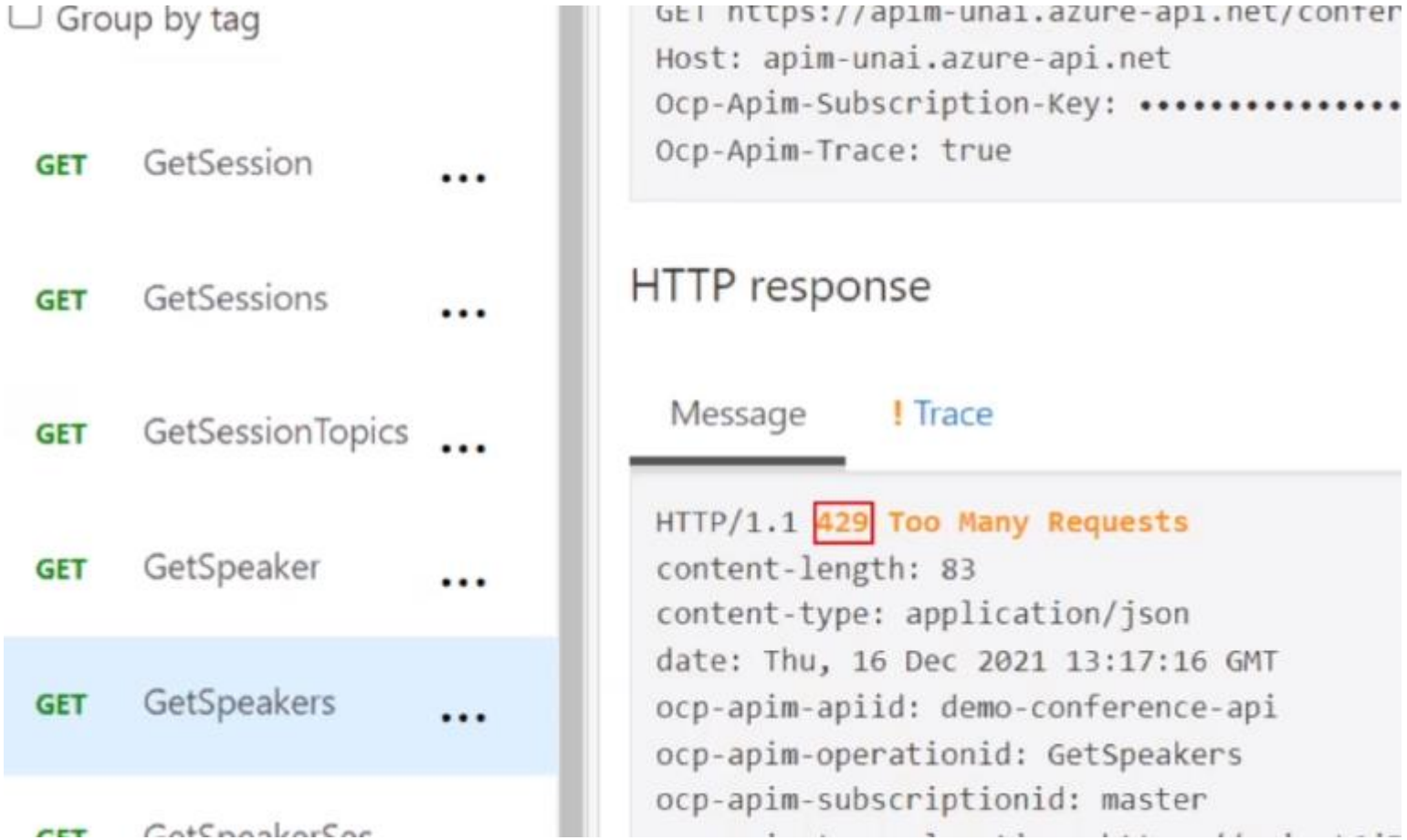
```
<policies>
    <inbound>
        <cross-domain />
        <base />
        <find-and-replace from="xyz" to="abc" />
    </inbound>
</policies>
```

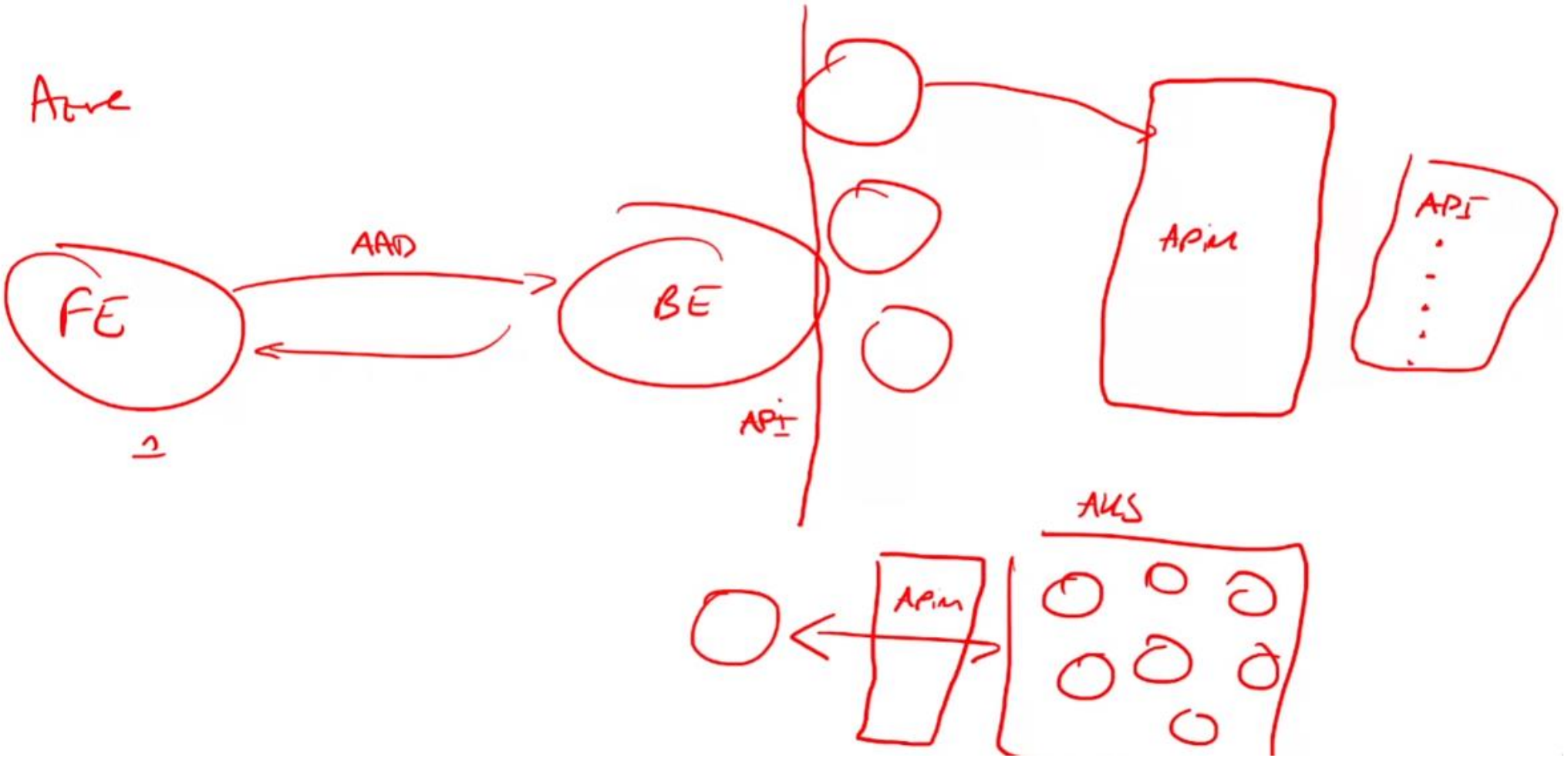
Advanced policy scenarios – limit concurrency

```
<limit-concurrency key="expression" max-count="number">
    <!-- nested policy statements -->
</limit-concurrency>
```



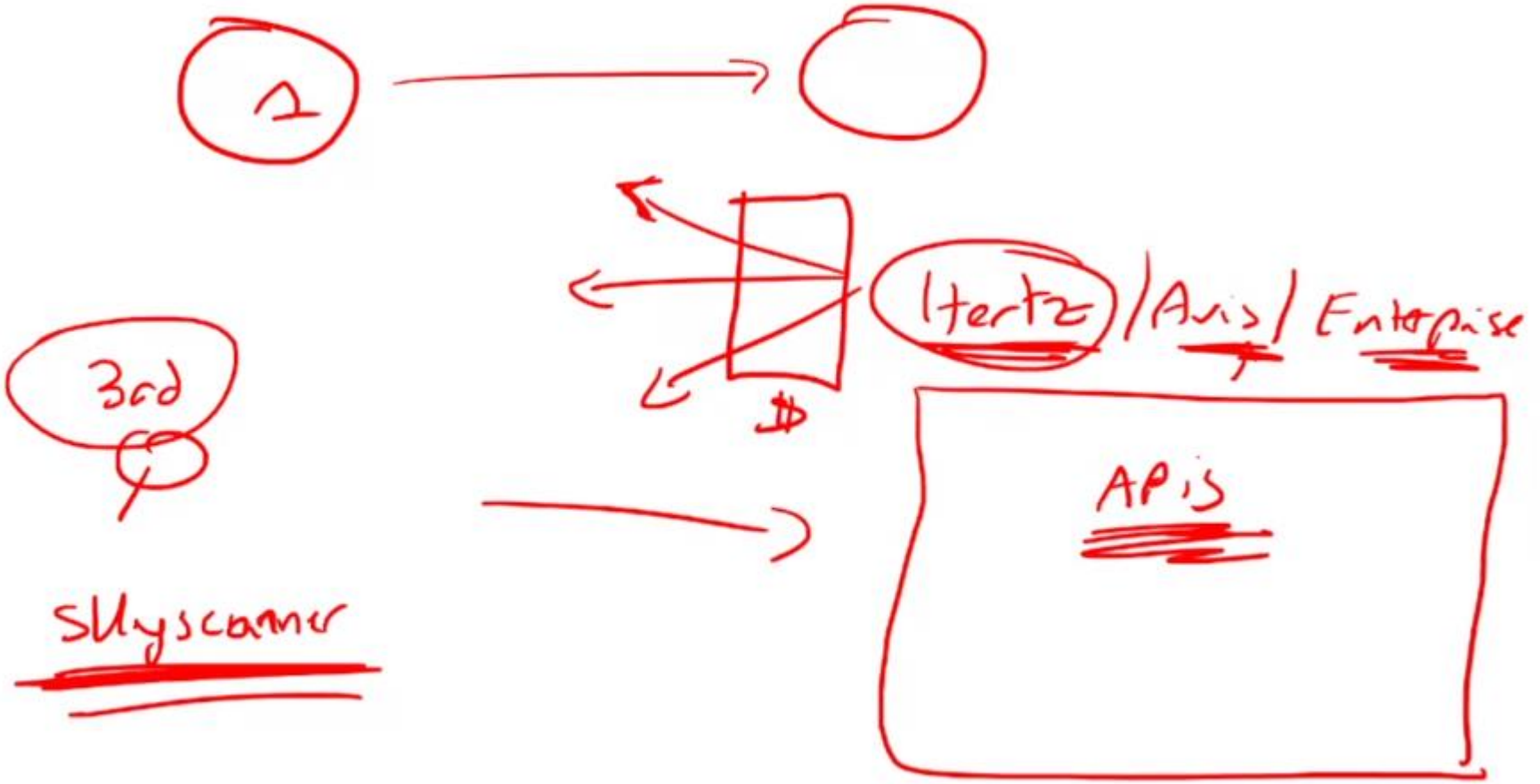
https://docs.microsoft.com/en-us/biztalk/core/connect-to-azure-api-management



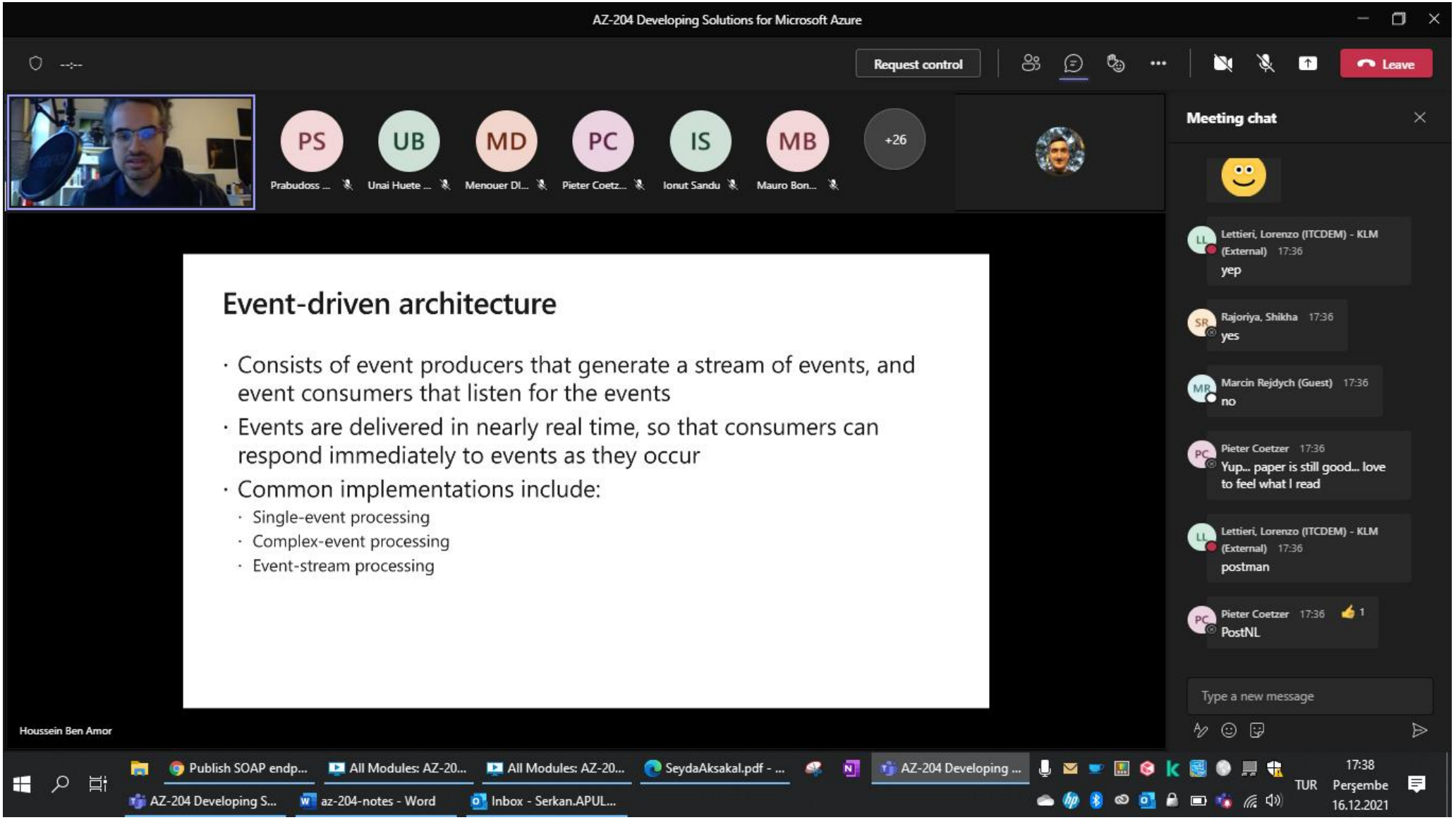


URL -> browse azure architectures

Browse Azure Architectures - Azure Architecture Center | Microsoft Docs


Hertz
Avis
Enterprise
3rd
APIs
Skyscanner

Module 9: Develop event-based solutions

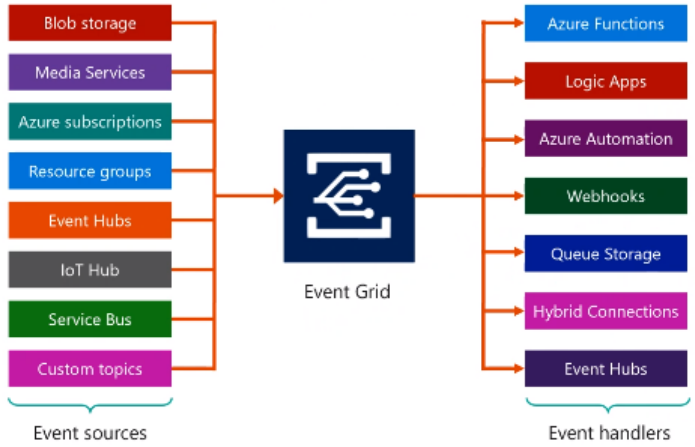

AZ-204 Developing Solutions for Microsoft Azure
Request control
Leave
PS
UB
MD
PC
IS
MB
+26
Prabudoss ...
Unai Huete ...
Menouer Dl...
Pieter Coetz...
Ionut Sandu
Mauro Bon...
Event-driven architecture
· Consists of event producers that generate a stream of events, and event consumers that listen for the events
· Events are delivered in nearly real time, so that consumers can respond immediately to events as they occur
· Common implementations include:
· Single-event processing
· Complex-event processing
· Event-stream processing
Houssein Ben Amor
Meeting chat
Lettieri, Lorenzo (ITCDEM) - KLM (External) 17:36
yep
Rajoriya, Shikha 17:36
yes
Marcin Rejdych (Guest) 17:36
no
Pieter Coetzer 17:36
Yup... paper is still good... love to feel what I read
Lettieri, Lorenzo (ITCDEM) - KLM (External) 17:36
postman
Pieter Coetzer 17:36
PostNL
Type a new message
Publish SOAP endp...
All Modules: AZ-20...
All Modules: AZ-20...
SeydaAksakal.pdf - ...
AZ-204 Developing ...
AZ-204 Developing S...
az-204-notes - Word
Inbox - Serkan.APUL...
TUR
17:38
Perşembe
16.12.2021

AZ-204 Developing Solutions for Microsoft Azure

Request control | Leave

PS Prabudoss ... | UB Unai Huete ... | MD Menouer Dl... | PC Pieter Coetz... | IS Ionut Sandu | MB Mauro Bon... | +26

Sources and handlers



Houssein Ben Amor

Meeting chat

Lettieri, Lorenzo (ITCDEM) - KLM (External) 17:36
yep

Rajoriya, Shikha 17:36
yes

Marcin Rejdych (Guest) 17:36
no

Pieter Coetzer 17:36
Yup... paper is still good... love to feel what I read

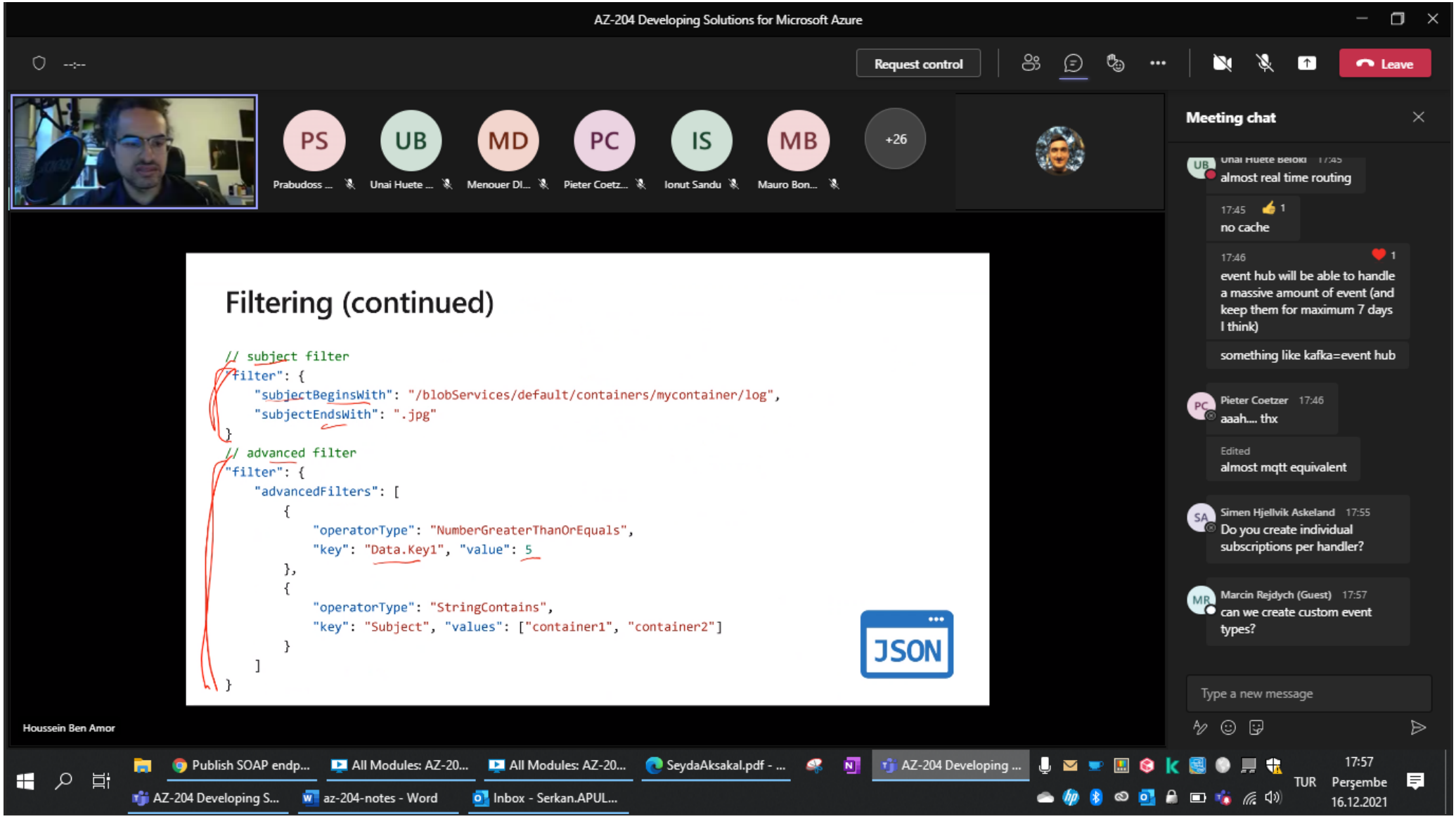
Lettieri, Lorenzo (ITCDEM) - KLM (External) 17:36
postman

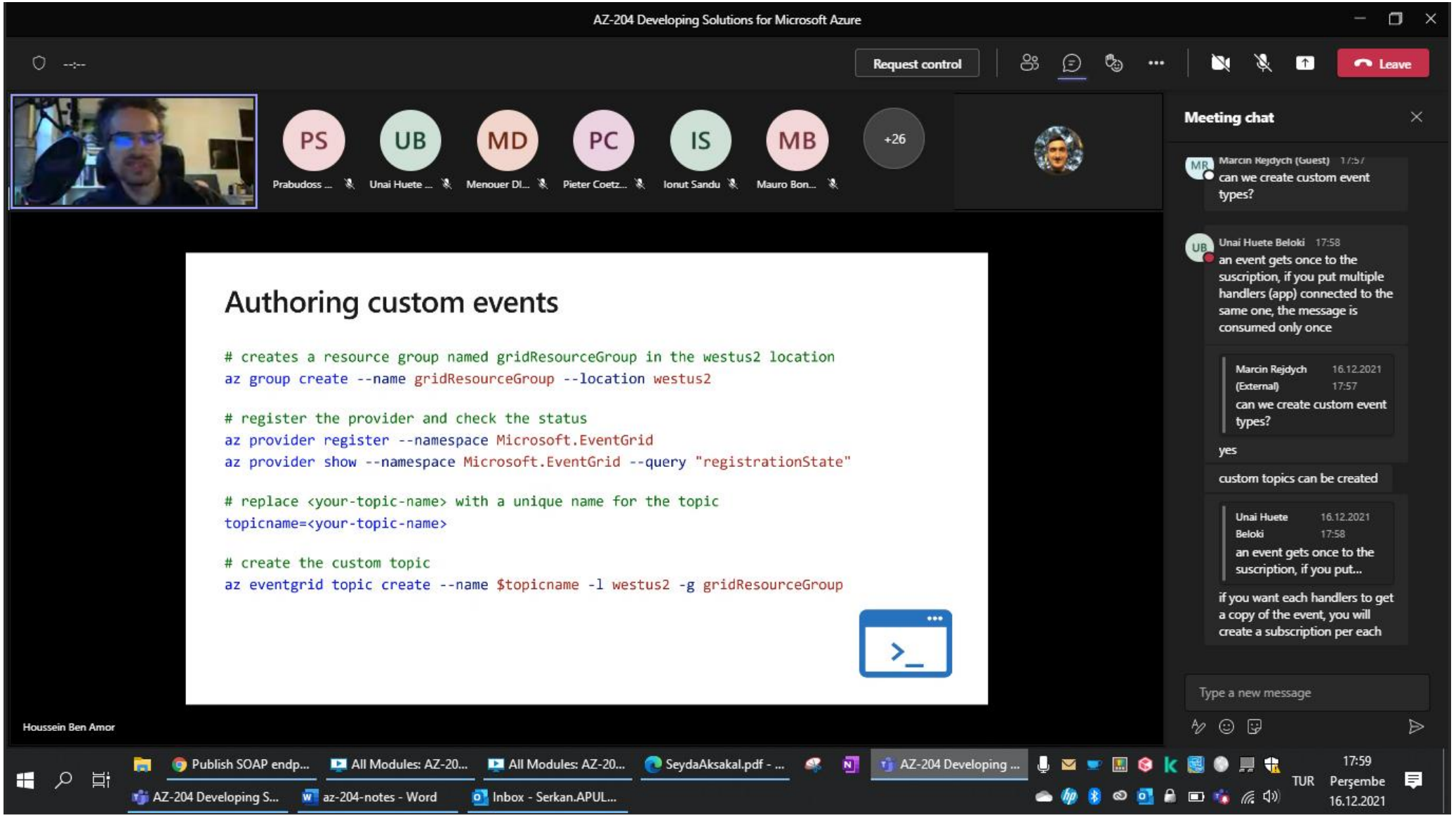
Pieter Coetzer 17:36 👍 1
PostNL

Type a new message

Publish SOAP endp... | All Modules: AZ-20... | All Modules: AZ-20... | SeydaAksakal.pdf - ... | AZ-204 Developing ... | AZ-204 Developing S... | az-204-notes - Word | Inbox - Serkan.APUL...

TUR 17:42 Perşembe 16.12.2021


AZ-204 Developing Solutions for Microsoft Azure
Request control
Leave
PS
UB
MD
PC
IS
MB
+26
Prabudoss ...
Unai Huete ...
Menouer Dl...
Pieter Coetz...
Ionut Sandu
Mauro Bon...
Filtering (continued)
// subject filter
"filter": {
"subjectBeginsWith": "/blobServices/default/containers/mycontainer/log",
"subjectEndsWith": ".jpg"
}
// advanced filter
"filter": {
"advancedFilters": [
{
"operatorType": "NumberGreaterThanOrEquals",
"key": "Data.Key1", "value": 5
},
{
"operatorType": "StringContains",
"key": "Subject", "values": ["container1", "container2"]
}
]
}
JSON
Houssein Ben Amor
Meeting chat
almost real time routing
17:45
no cache
17:46
event hub will be able to handle a massive amount of event (and keep them for maximum 7 days I think)
something like kafka=event hub
Pieter Coetzer 17:46
aaah.... thx
Edited
almost mqtt equivalent
Simen Hjellvik Askeland 17:55
Do you create individual subscriptions per handler?
Marcin Rejdych (Guest) 17:57
can we create custom event types?
Type a new message
Publish SOAP endp...
All Modules: AZ-20...
All Modules: AZ-20...
SeydaAksakal.pdf - ...
AZ-204 Developing ...
AZ-204 Developing S...
az-204-notes - Word
Inbox - Serkan.APUL...
TUR
17:57
Perşembe
16.12.2021

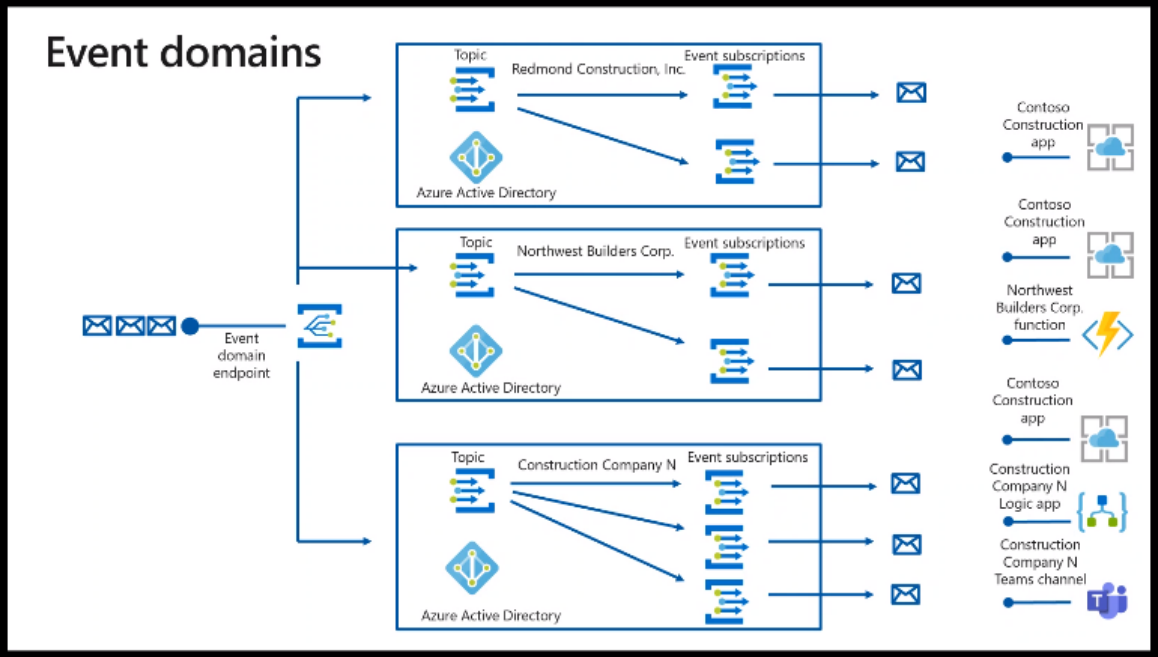

AZ-204 Developing Solutions for Microsoft Azure
Request control
Leave
PS
UB
MD
PC
IS
MB
+26
Prabudoss ...
Unai Huete ...
Menouer Dl...
Pieter Coetz...
Ionut Sandu
Mauro Bon...
Authoring custom events
creates a resource group named gridResourceGroup in the westus2 location
az group create --name gridResourceGroup --location westus2
register the provider and check the status
az provider register --namespace Microsoft.EventGrid
az provider show --namespace Microsoft.EventGrid --query "registrationState"
# replace <your-topic-name> with a unique name for the topic
topicname=<your-topic-name>
# create the custom topic
az eventgrid topic create --name $topicname -l westus2 -g gridResourceGroup
Houssein Ben Amor
Meeting chat
can we create custom event types?
an event gets once to the suscription, if you put multiple handlers (app) connected to the same one, the message is consumed only once
Marcin Rejdych 16.12.2021 (External) 17:57
can we create custom event types?
yes
custom topics can be created
Unai Huete Beloki 16.12.2021 17:58
an event gets once to the suscription, if you put...
if you want each handlers to get a copy of the event, you will create a subscription per each
Type a new message
Publish SOAP endp...
All Modules: AZ-20...
All Modules: AZ-20...
SeydaAksakal.pdf - ...
AZ-204 Developing ...
AZ-204 Developing S...
az-204-notes - Word
Inbox - Serkan.APUL...
TUR
17:59
Perşembe
16.12.2021

AZ-204 Developing Solutions for Microsoft Azure

Request control

Leave

PS Prabudoss ... UB Unai Huete ... MD Menouer Dl... PC Pieter Coetz... IS Ionut Sandu MB Mauro Bon... +26



Houssein Ben Amor

Meeting chat

can we create custom event types?

Unai Huete Beloki 17:58
an event gets once to the suscription, if you put multiple handlers (app) connected to the same one, the message is consumed only once

> Marcin Rejdych (External) 16.12.2021 17:57
> can we create custom event types?

yes

custom topics can be created

17:59 1

> Unai Huete Beloki 16.12.2021 17:58
> an event gets once to the suscription, if you put...

if you want each handlers to get a copy of the event, you will create a subscription per each

Type a new message

Publish SOAP endp... All Modules: AZ-20... All Modules: AZ-20... SeydaAksakal.pdf - ... AZ-204 Developing ... AZ-204 Developing S... az-204-notes - Word Inbox - Serkan.APUL...

TUR 18:02 Perşembe 16.12.2021

AZ-204 Developing Solutions for Microsoft Azure

Request control

Leave

PS Prabudoss ... UB Unai Huete ... MD Menouer Dl... PC Pieter Coetz... IS Ionut Sandu MB Mauro Bon... +27

Lesson 02: Azure Event Hubs

Houssein Ben Amor

Meeting chat

types?

Unai Huete Beloki 17:58
an event gets once to the suscription, if you put multiple handlers (app) connected to the same one, the message is consumed only once

> Marcin Rejdych (External) 16.12.2021 17:57
> can we create custom event types?

yes

custom topics can be created

17:59 1

> Unai Huete Beloki 16.12.2021 17:58
> an event gets once to the suscription, if you put...

if you want each handlers to get a copy of the event, you will create a subscription per each

Type a new message

Publish SOAP endp... All Modules: AZ-20... All Modules: AZ-20... SeydaAksakal.pdf - ... AZ-204 Developing ...

AZ-204 Developing S... az-204-notes - Word Inbox - Serkan.APUL...

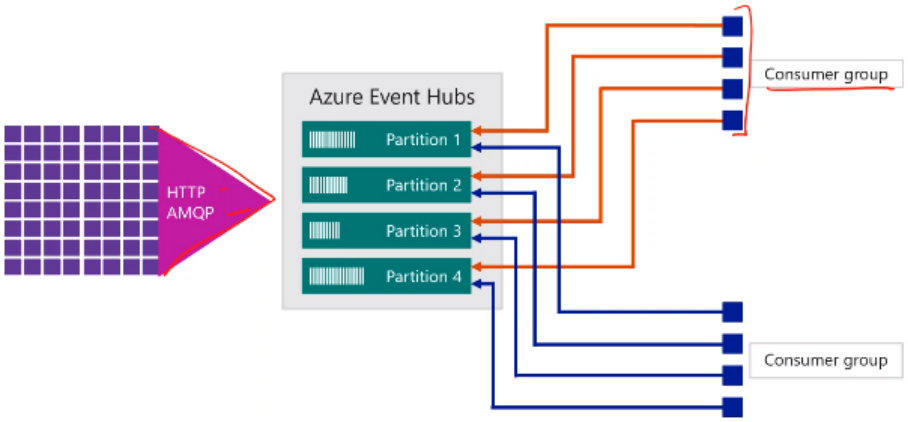
TUR 18:05 Perşembe 16.12.2021

AZ-204 Developing Solutions for Microsoft Azure

Request control | Leave

PS Prabudoss ... | UB Unai Huete ... | MD Menouer Dl... | PC Pieter Coetz... | IS Ionut Sandu | MB Mauro Bon... | +27

Consumer groups



Houssein Ben Amor

Meeting chat

if you want each handlers to get a copy of the event, you will create a subscription per each

Pieter Coetzer 18:05
Would you compare this to AMQP (Rabbit MQ) ?

18:06
like iOT (event producers? ?

Unai Huete Beloki 18:07 1
I would say kafka=event hub

18:07 1
rab

18:09 1

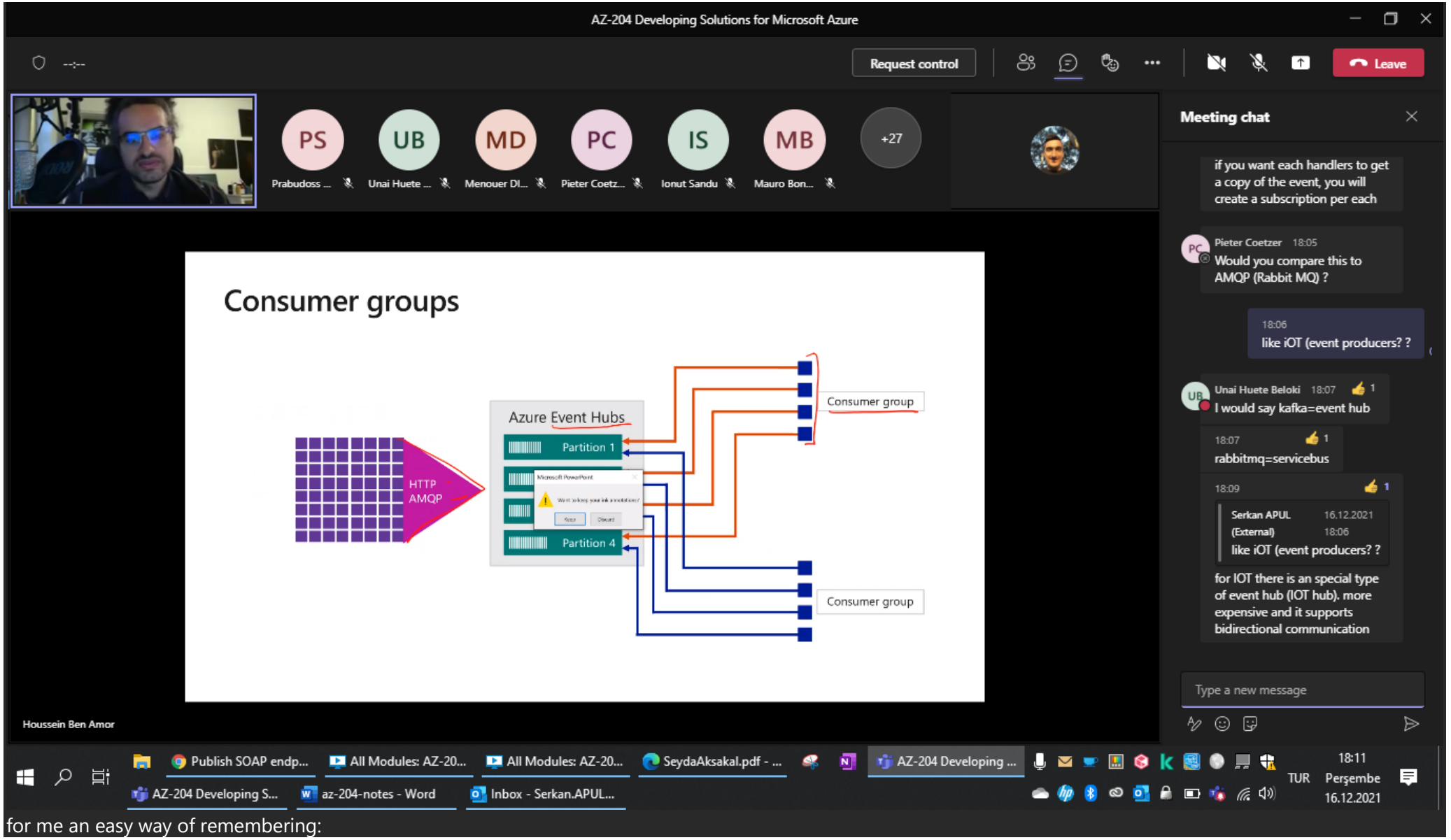
> Serkan APUL (External) 16.12.2021 18:06
> like iOT (event producers? ?

for IOT there is an special type of event hub (IOT hub). more expensive and it supports bidirectional communication

Type a new message

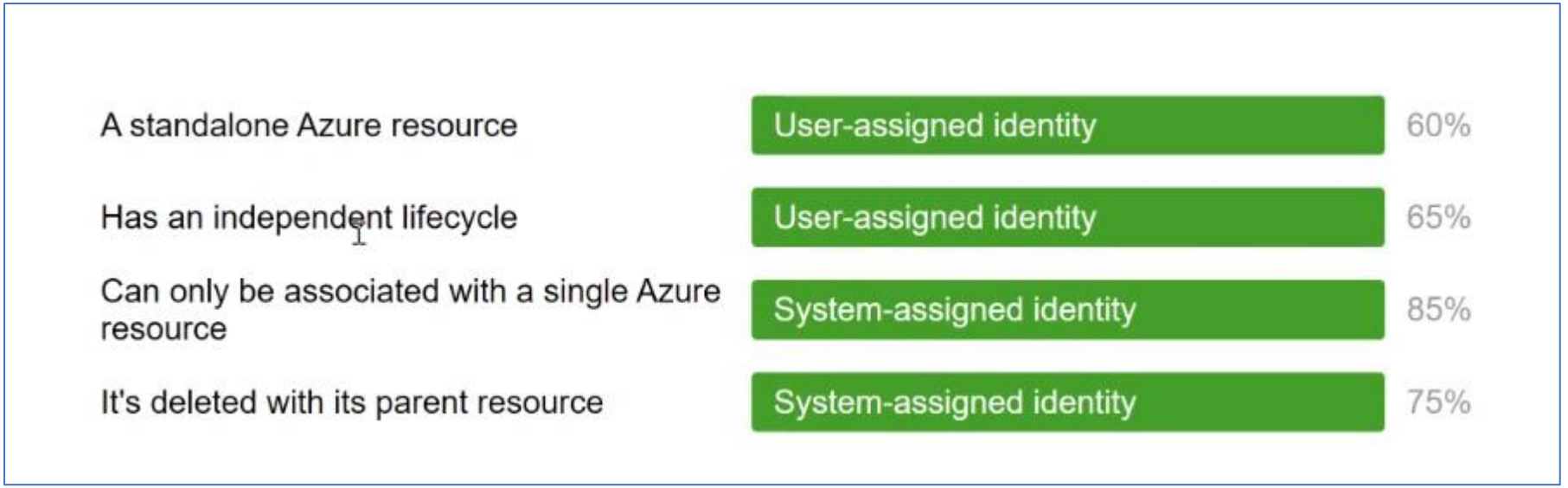
Publish SOAP endp... | All Modules: AZ-20... | All Modules: AZ-20... | SeydaAksakal.pdf - ... | AZ-204 Developing ... | AZ-204 Developing S... | az-204-notes - Word | Inbox - Serkan.APUL...

TUR 18:10 Perşembe 16.12.2021



for me an easy way of remembering:

- event grid: reactive programming, react almost in real time to events
- event hub (like kafka): massive ingestions of events/logs per second, after ingestion data analytic tools can be connected to it
- service bus (like rabbitmq): premium messaging solution, really used in microservices based architecture for async comm


A standalone Azure resource
User-assigned identity
60%
Has an independent lifecycle
User-assigned identity
65%
Can only be associated with a single Azure resource
System-assigned identity
85%
It's deleted with its parent resource
System-assigned identity
75%

1 2 3 4 5

In API Management, which of the below options is used to set up policies like quotas?

single choice **33%** group success rate

"A security identity that user-created apps, services, and automation tools use to access specific Azure resources." What is this?

single choice **91%** group success rate

1 2 3 4 5

In API Management, which of the options below accurately reflects how policies are applied?

single choice **87%** group success rate

Which component of the API Management system is the endpoint that accepts API calls and routes them to your backends? ?

Correct!

- Product definition
- Azure portal
- API gateway
- Developer portal

Which of the following statements are true with regard to API policies? Choose all that apply.

multiple choice **61%** group success rate

1 2 3 4

Fill the gap to create a secret in a Key Vault?

fill the gaps **21%** group success rate

az keyvault secret set --vault-name myvault --name dbpsw --value 'Pa5w.rd'

set

21%

Sort following API management elements from high to low

sort into order **73%** group success rate

1	Product	78%
2	API	83%
3	Operation	83%

Regarding Azure Event Grid, which of the below represents the smallest amount of information that fully describes something happening in the system?

single choice **69%** group success rate

2 / 3

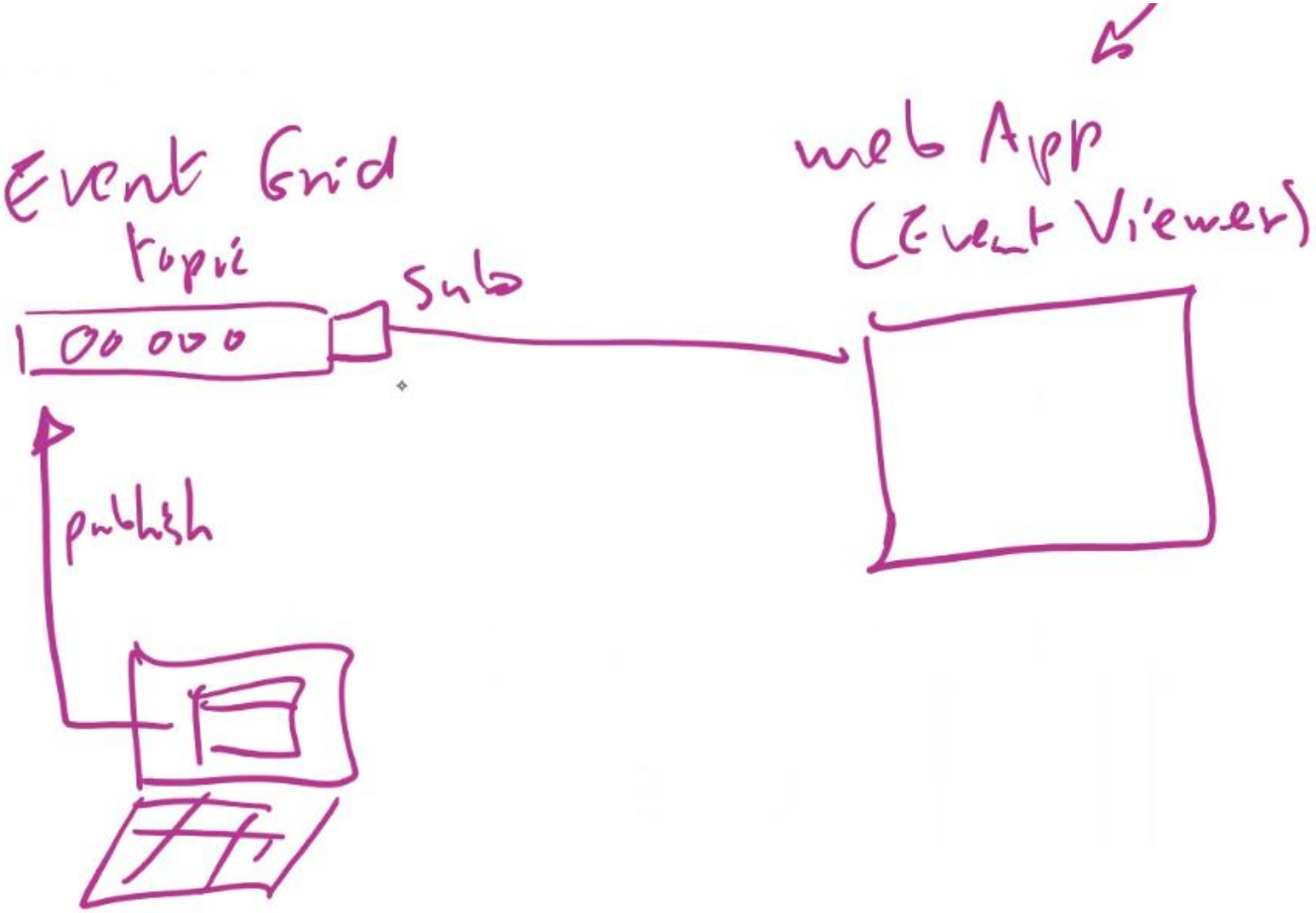
Select the service corresponding to each statement

Correct!

Designed to handle streams of events	Event Hubs ✓
Designed for communication between apps when status changes	Event Grid ✓
Designed to handle individual message	Event Grid ✓
Designed for Big data pipelines	Event Hubs ✓

Azure Event Hubs is a big data streaming platform and event ingestion service. What type of service is it?

single choice **34%** group success rate

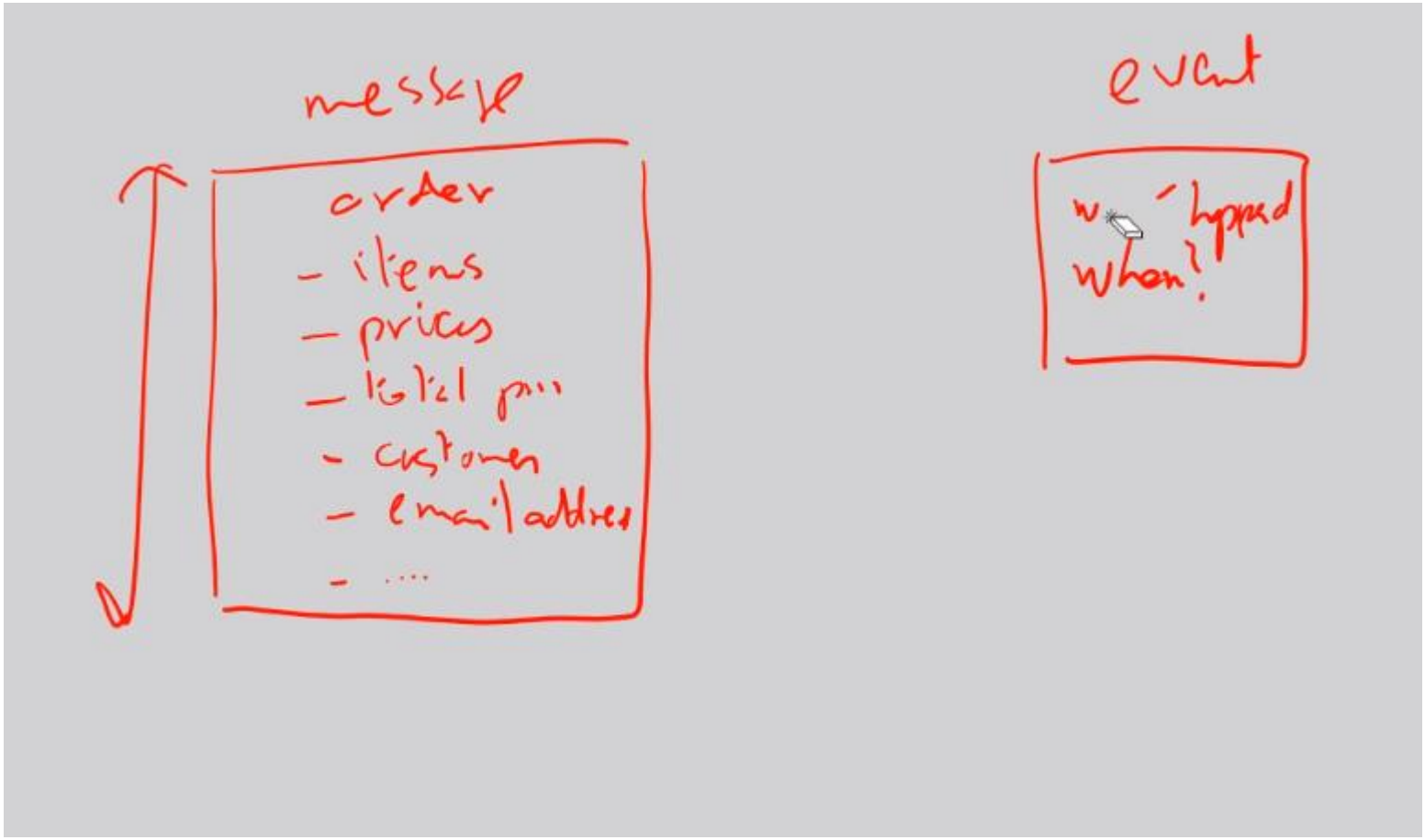

Event Grid
topic
sub
web App
(Event Viewer)
publish

Microsoft

Module 10: Develop message-based solutions

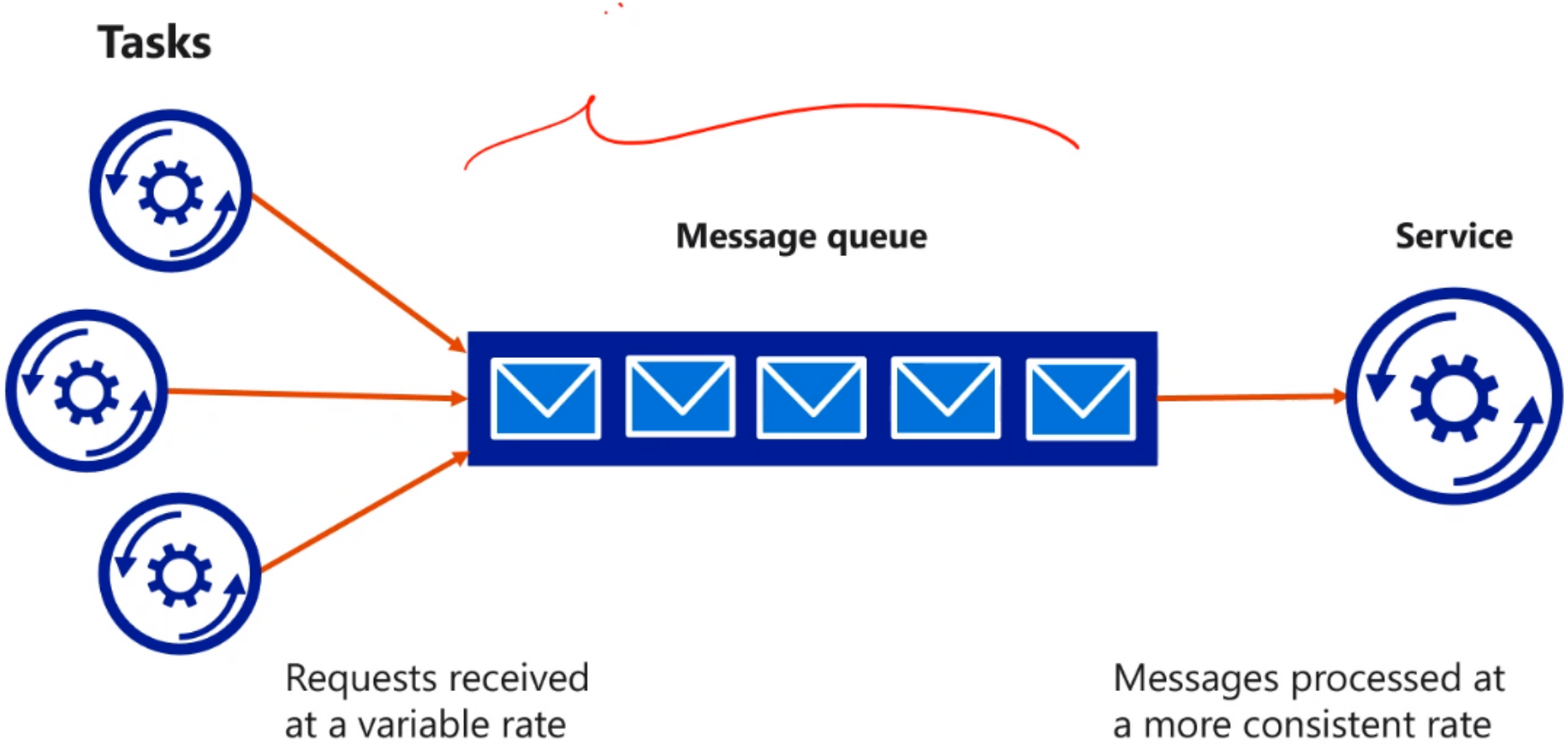
Topics

- Azure Service Bus
- Azure Queue Storage


message
order
- items
- prices
- total price
- customer
- email address
- ....
event
what happened
when?

Lesson 01: Azure Service Bus

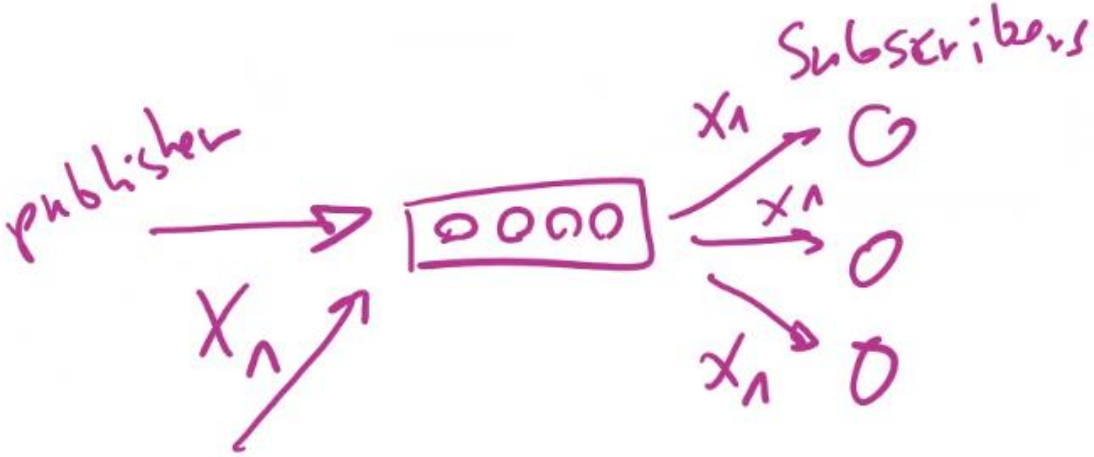
Queue-based load leveling



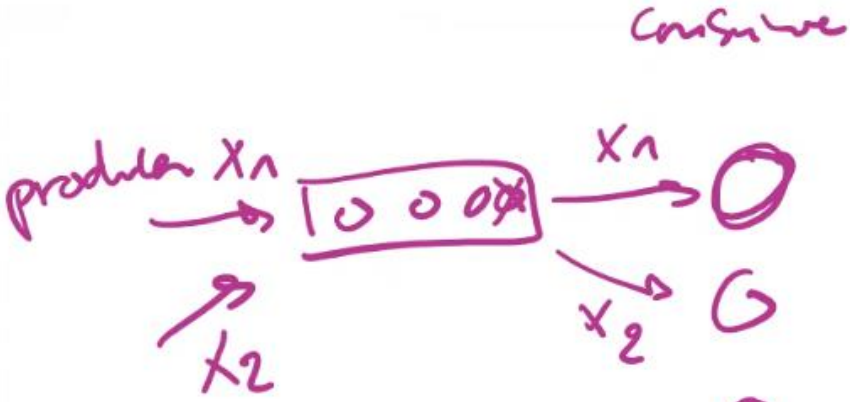
Queues

- Messages are sent to and received from queues
- Enables you to store messages until the receiving application is available to receive and process them
- Supports a brokered messaging communication model
- A general-purpose technology that can be used for a wide variety of scenarios

Topic
(pubsub)



Queue

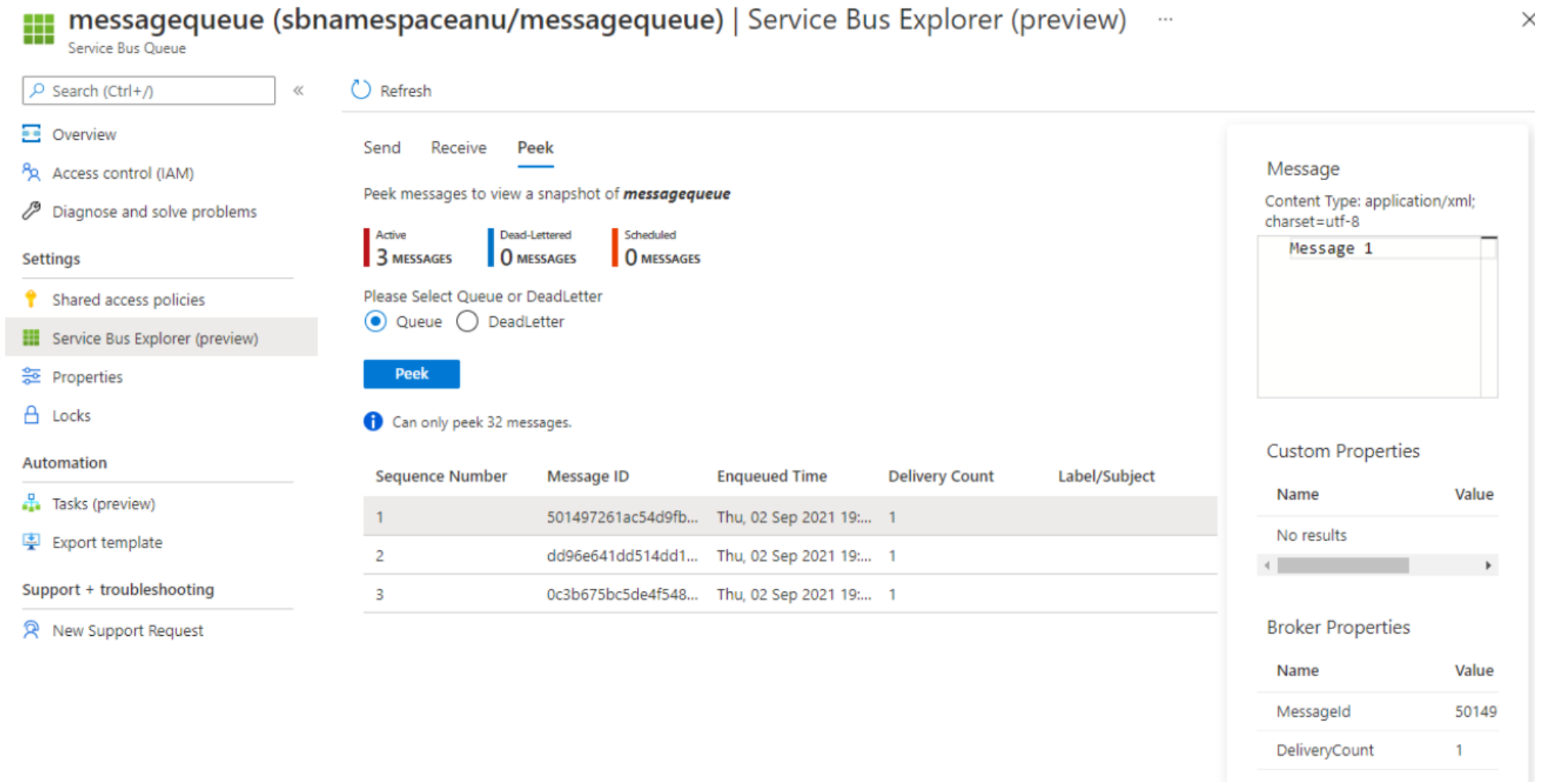


Lesson 02: Azure Queue Storage

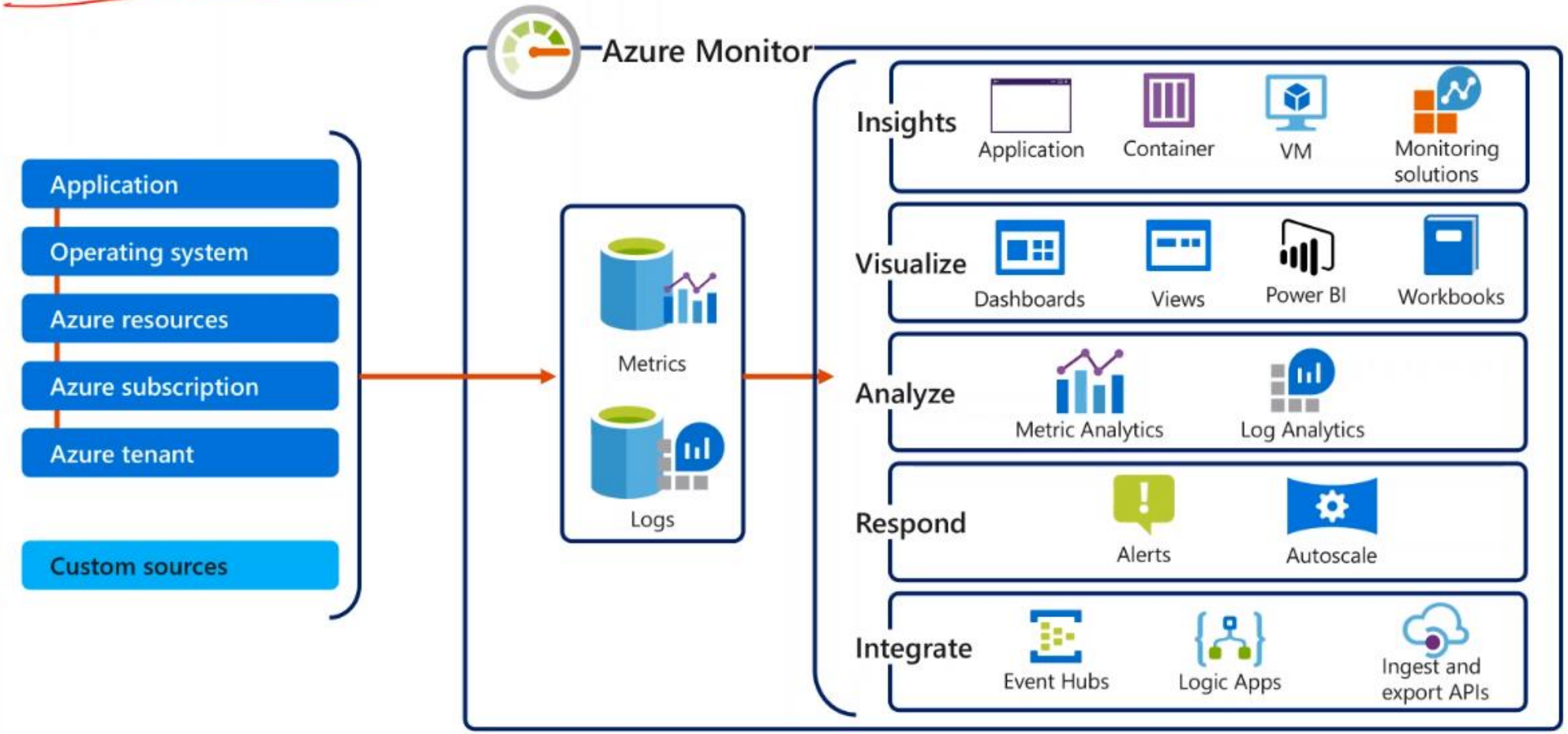
Comparing cloud messaging options

Requirement	Simple queuing	Eventing and PubSub	Big data streaming	Enterprise messaging
Product	Queue storage	Event Grid	Event Hubs	**Service Bus**
Supported advantages	• Communication within an app • Individual message • Queue semantics / polling buffer • Simple and easy to use • Pay as you go	• Communication between apps / orgs • Individual message • Push semantics • Filtering and routing • Pay as you go • Fan out	• Many messages in a Stream (think in MBs) • Ease of use and operation • Low cost • Fan in • Strict ordering • Works with other tools	• Instantaneous consistency • Strict ordering • Java Messaging Service • Non-repudiation and security • Geo-replication and availability • Rich features (such as deduplication and scheduling)
Weaknesses	• Ordering of messaging • Instantaneous consistency	• Ordering of messaging • Instantaneous consistency	• Server-side cursor • Only once	• Cost • Simplicity
Type	Serverless	Serverless	Big data	Enterprise

service bus --> moving a message that will be processed (the information inside message is important,)
event grid --> react to event (the happening of the event if the important , we want to react ASAP)



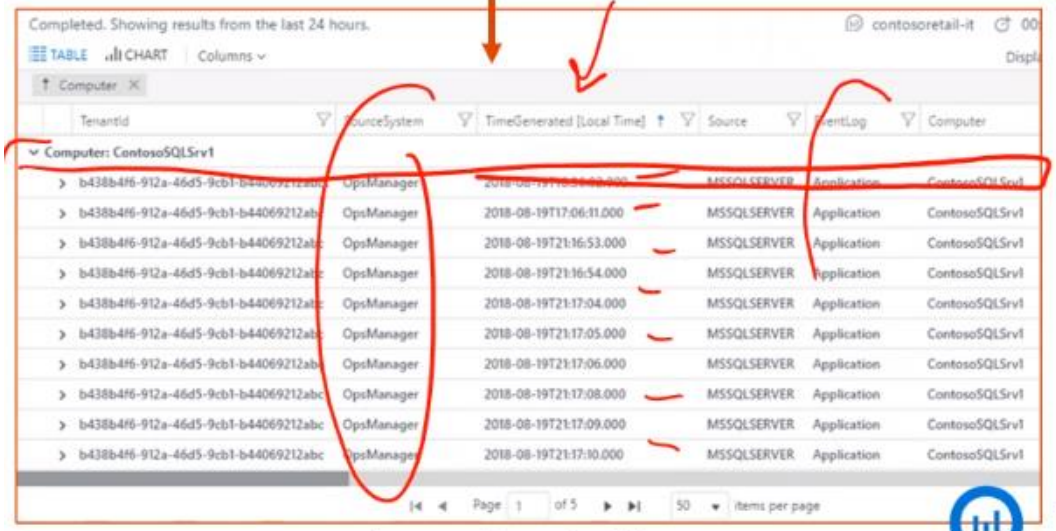
Azure Monitor overview



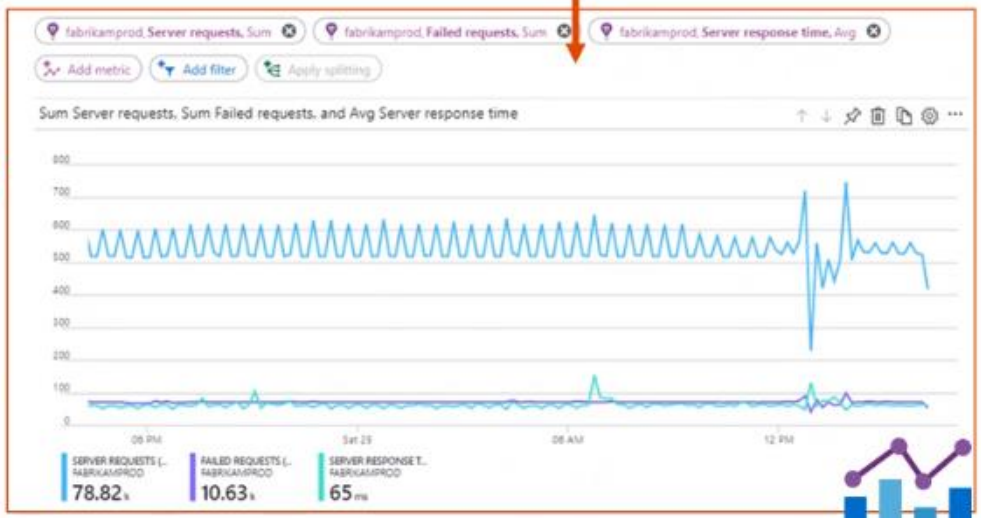
Monitoring data platform

Logs

Metrics



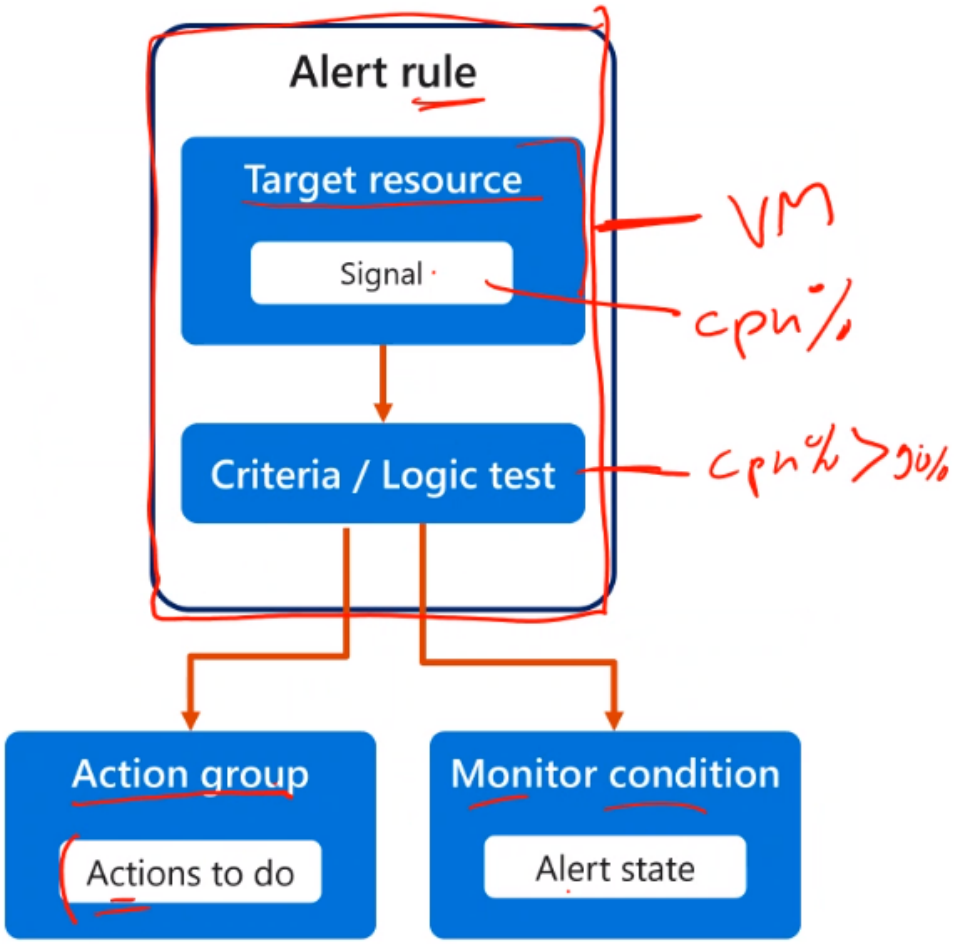
Log Analytics



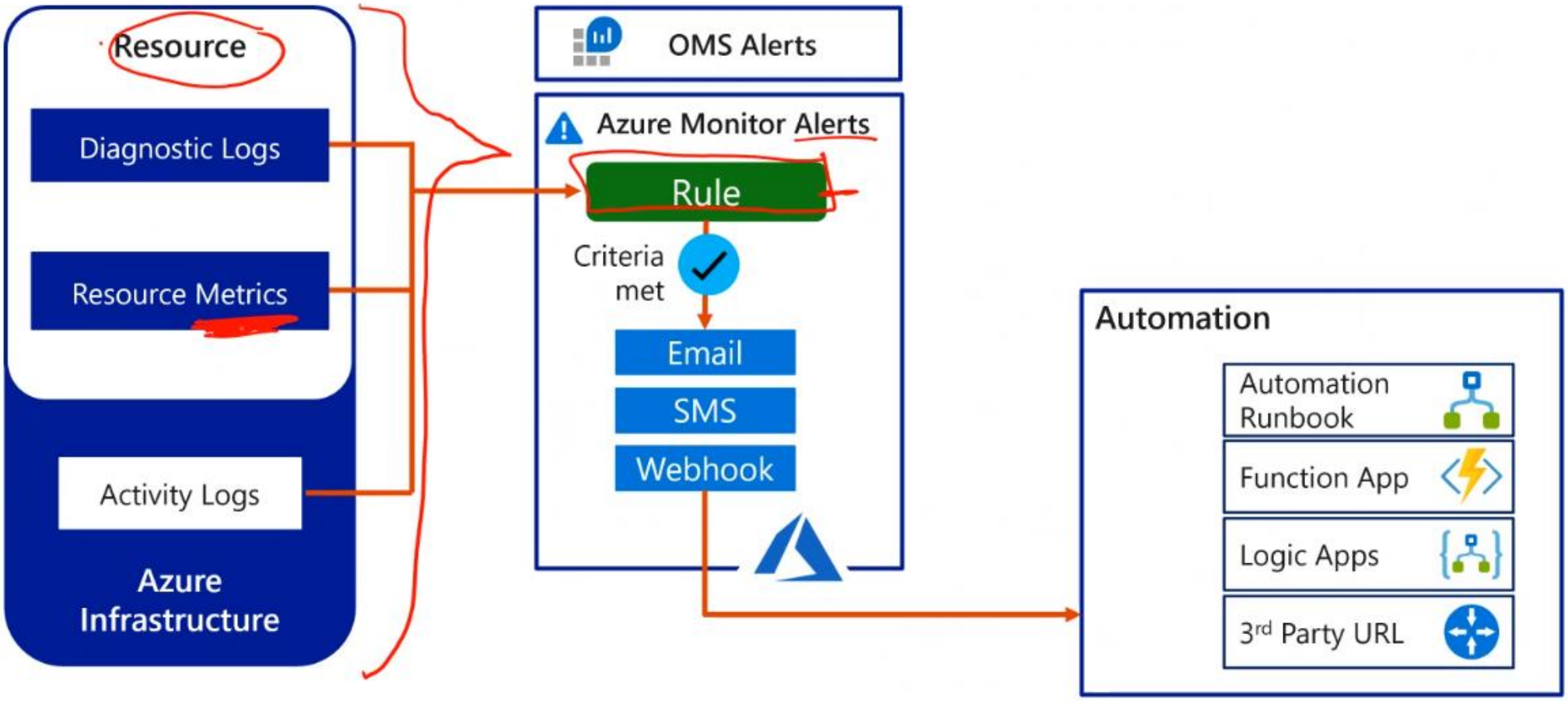
Metrics Explorer

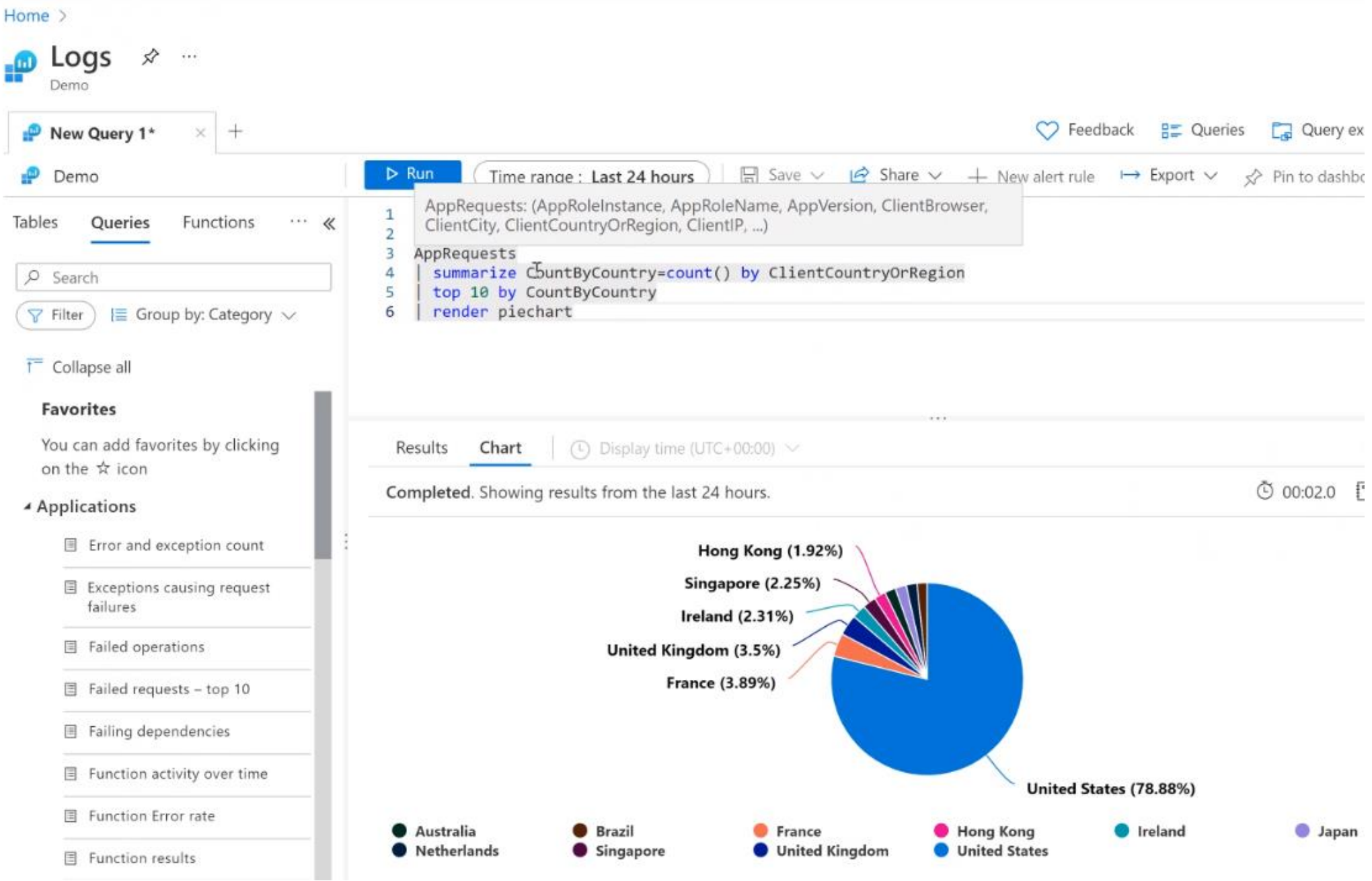
Alerts

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

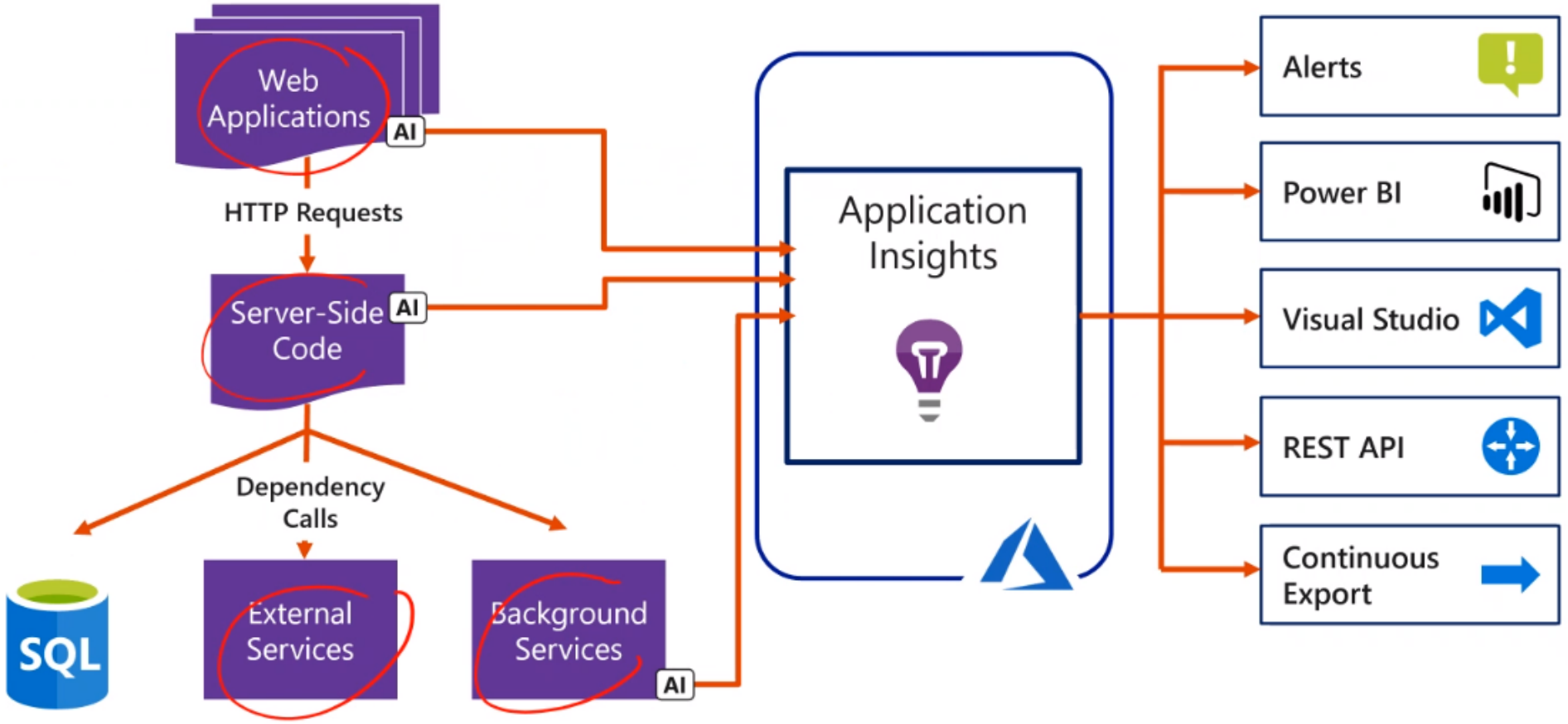


Alerts workflow

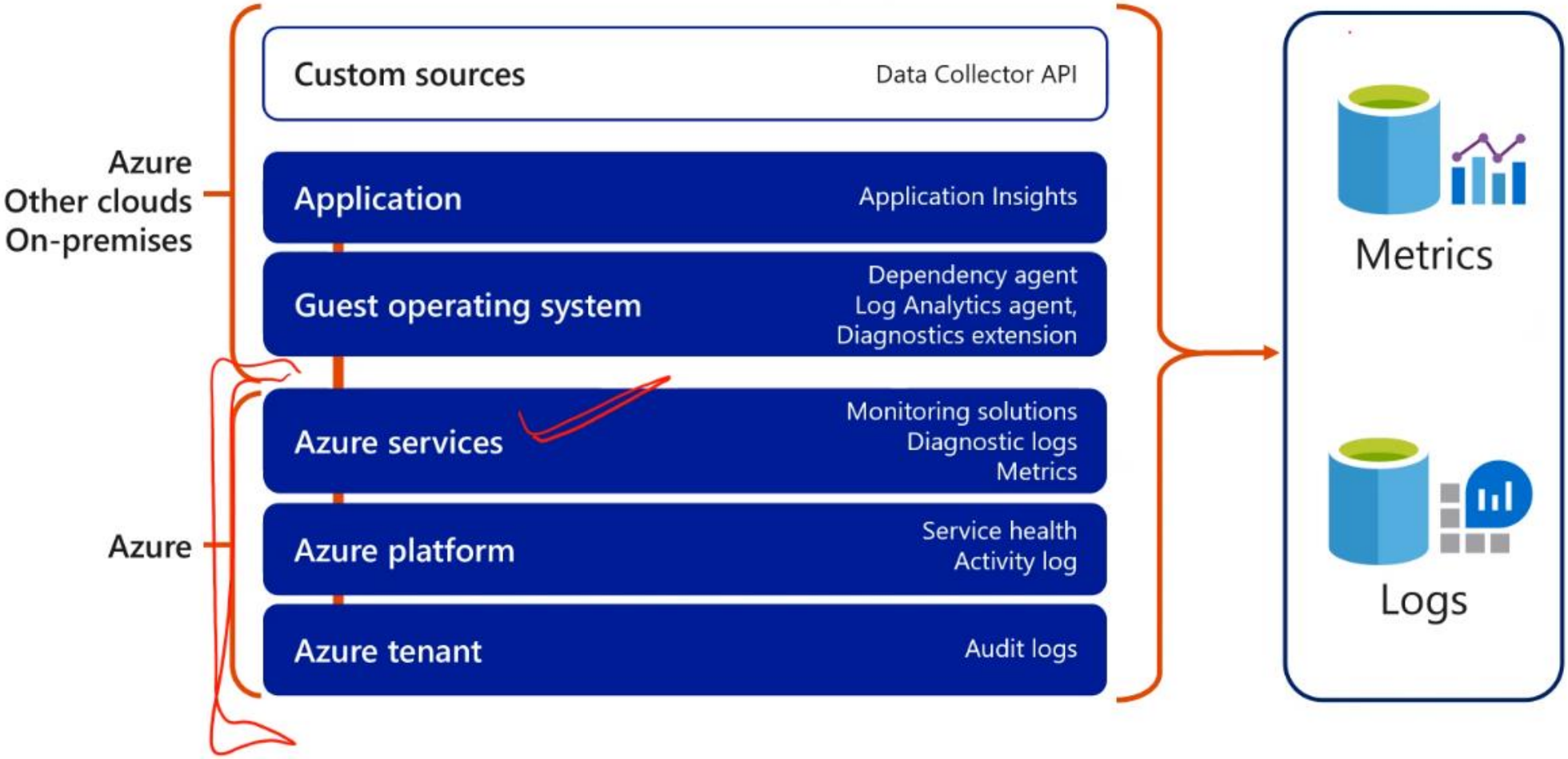



Home
Logs
Demo
New Query 1*
Feedback
Queries
Query ex
Demo
Run
Time range : Last 24 hours
Save
Share
New alert rule
Export
Pin to dashbo
Tables
Queries
Functions
Search
Filter
Group by: Category
Collapse all
Favorites
You can add favorites by clicking on the ☆ icon
Applications
Error and exception count
Exceptions causing request failures
Failed operations
Failed requests – top 10
Failing dependencies
Function activity over time
Function Error rate
Function results
AppRequests: (AppRoleInstance, AppRoleName, AppVersion, ClientBrowser, ClientCity, ClientCountryOrRegion, ClientIP, ...)
1
2
3 AppRequests
4 | summarize CountByCountry=count() by ClientCountryOrRegion
5 | top 10 by CountByCountry
6 | render piechart
Results
Chart
Display time (UTC+00:00)
Completed. Showing results from the last 24 hours.
00:02.0
Hong Kong (1.92%)
Singapore (2.25%)
Ireland (2.31%)
United Kingdom (3.5%)
France (3.89%)
United States (78.88%)
Australia
Brazil
France
Hong Kong
Ireland
Japan
Netherlands
Singapore
United Kingdom
United States

Application Insights architecture



Data sources



https://docs.microsoft.com/en-us/azure/azure-monitor/app/platforms

Transient errors

- Transient faults are **temporary** faults
  - Could be caused by environmental issues
    - Loss of network connectivity
    - Busy hardware components
    - Temporary unavailability of a connected service
    - Server timeouts
  - Typically are self-correcting

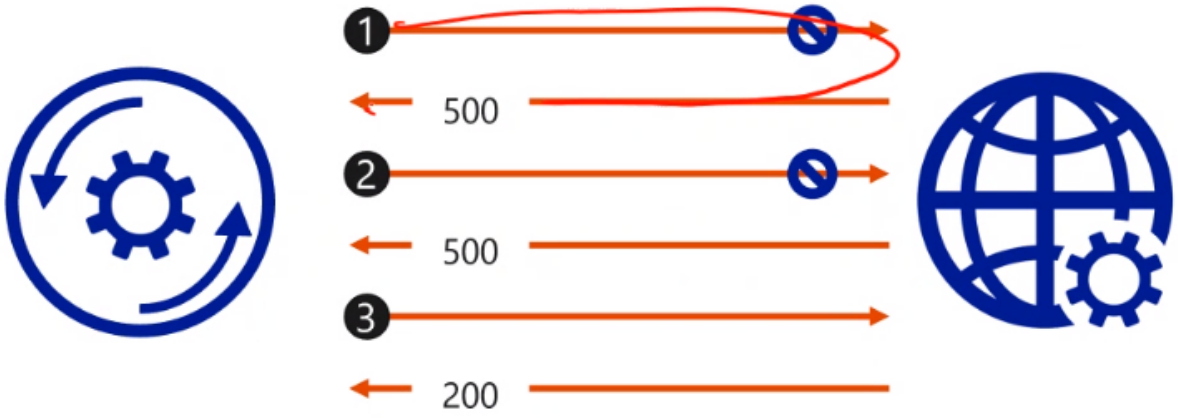
Handling transient errors

If an application detects a failure when it tries to send a request to a remote service, it can handle the failure by using the following strategies:

- Cancel
  - If the fault indicates that the failure isn't transient or is unlikely to be successful if repeated, the application should cancel the operation and report an exception. For example, an authentication failure caused by providing invalid credentials is not likely to succeed no matter how many times it's attempted.
- Retry
  - If the specific fault reported is unusual or rare, it might have been caused by unusual circumstances, such as a network packet becoming corrupted while it was being transmitted. In this case, the application could retry the failing request again immediately, because the same failure is unlikely to be repeated and the request will probably be successful.
- Retry after a delay
  - If the fault is caused by one of the more commonplace connectivity or busy failures, the network or service might need a short period of time while the connectivity issues are corrected or the backlog of work is cleared. The application should wait for a suitable amount of time before retrying the request.

Retrying after a transient error

1. The application invokes an operation on a hosted service. The request fails, and the service host responds with HTTP response code 500 (internal server error).
2. The application waits for a short interval and tries again. The request still fails with HTTP response code 500.
3. The application waits for a longer interval and tries again. The request succeeds with HTTP response code 200 (OK).



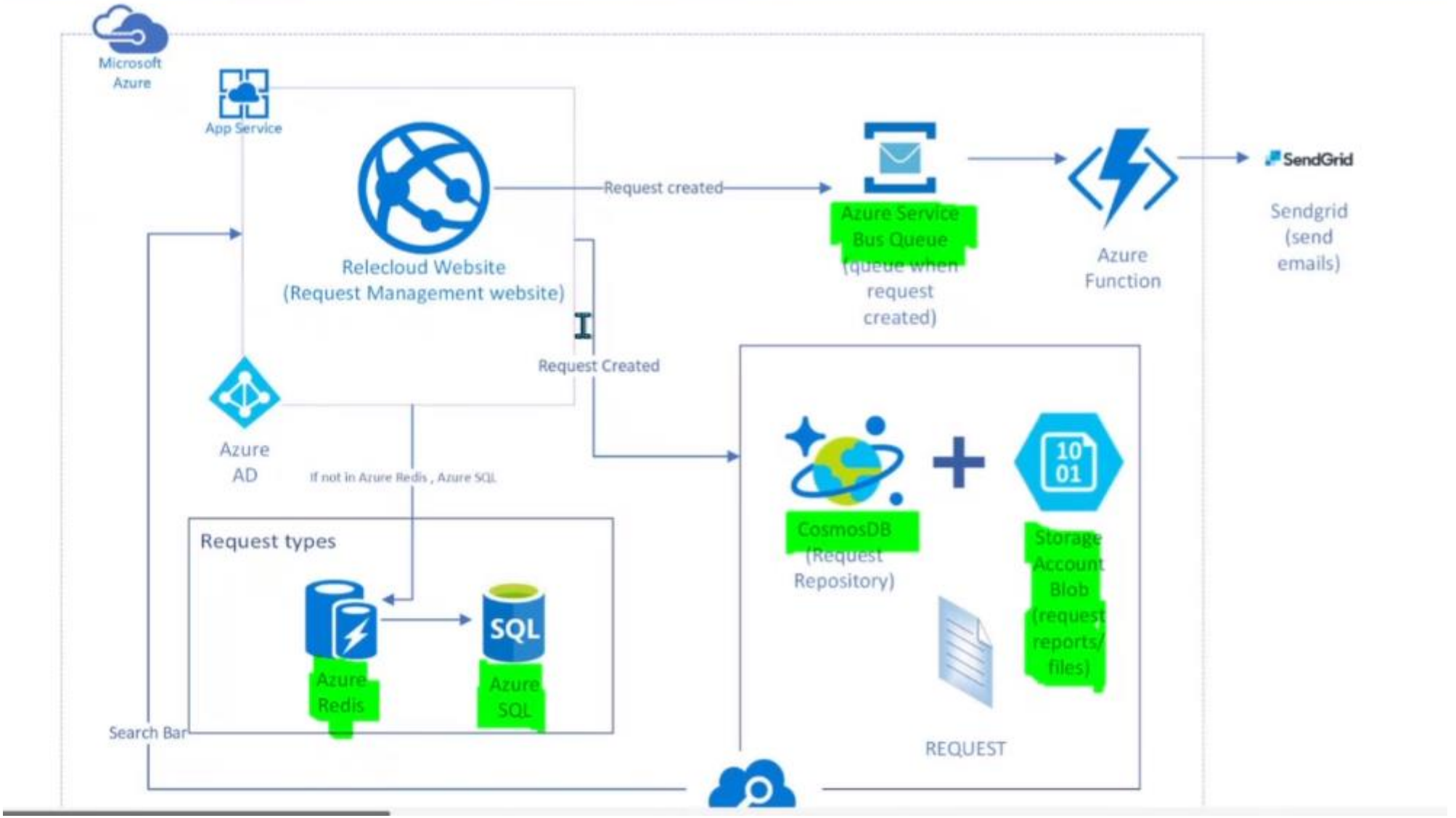
Pricing Tiers

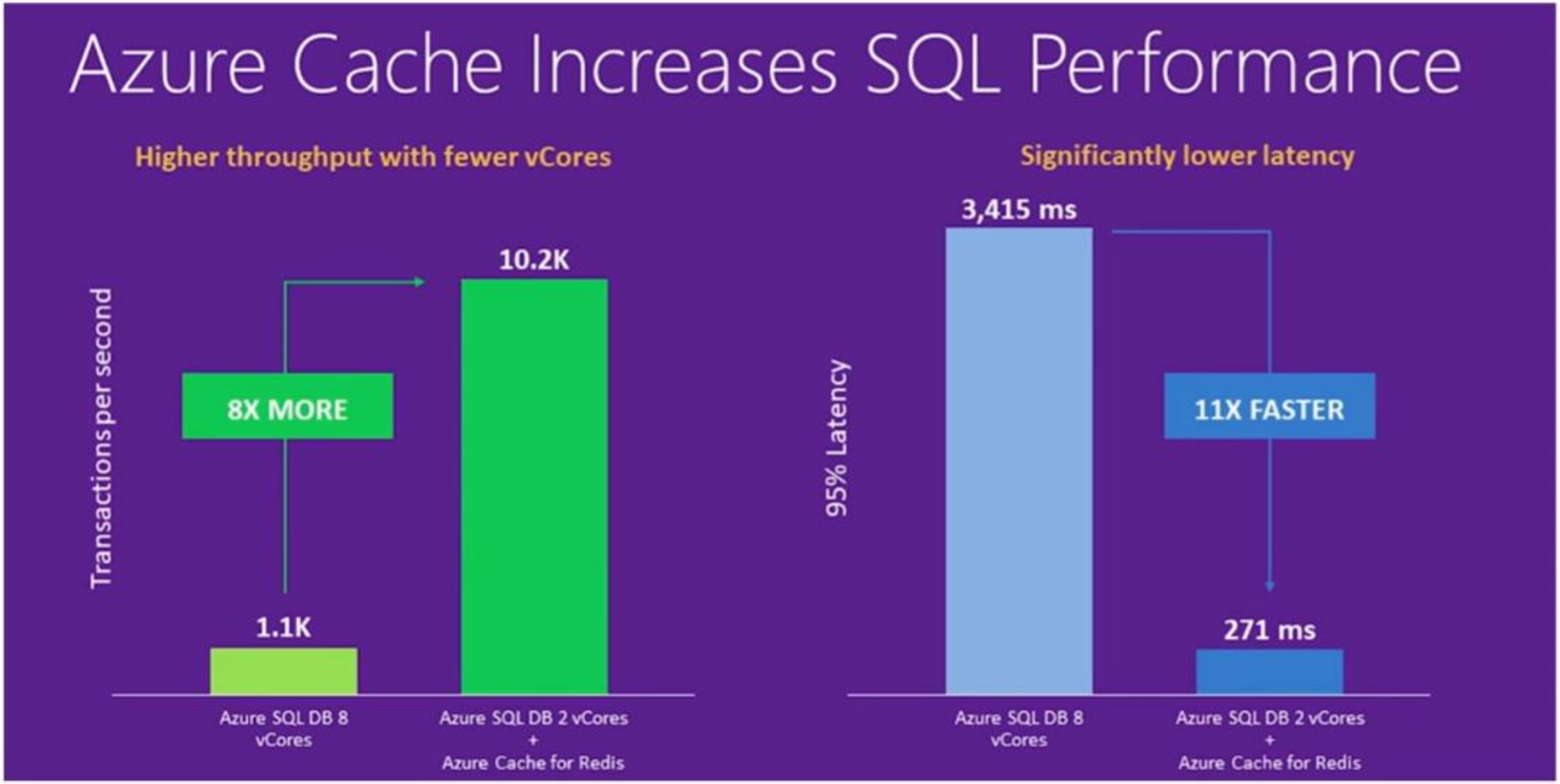
Basic	An OSS Redis cache **running on a single VM**. This tier has no service-level agreement (SLA) and is ideal for **development/test** and non-critical workloads.
Standard	An OSS Redis cache **running on two VMs** in a replicated configuration. SLA
Premium ↑	**High-performance OSS Redis caches**. This tier offers **higher throughput, lower latency, better availability, and more features.** Premium caches are deployed on more powerful VMs compared to those for Basic or Standard caches.
Enterprise +	High-performance caches **powered by Redis Labs' Redis Enterprise software**. This tier supports Redis **modules including RediSearch, RedisBloom, and RedisTimeSeries**. In addition, it offers even **higher availability than the Premium** tier.
Enterprise Flash	Cost-effective large caches powered by Redis Labs' Redis Enterprise software. This tier extends Redis data storage to non-volatile memory, which is cheaper than DRAM, on a VM. It reduces the overall per-GB memory cost.

https://docs.microsoft.com/en-us/azure/azure-cache-for-redis/cache-overview#feature-comparison

Configuration

- There are several parameters that you will need to decide to configure the cache properly for your purposes:
  - Name
  - Resource Group
  - Location
  - Pricing tier — Tiers + Memory
  - Virtual network support (Premium)
  - Clustering support + Premium SLA↑


Microsoft Azure
App Service
Relecloud Website
(Request Management website)
Request created
Azure Service Bus Queue (queue when request created)
Azure Function
SendGrid
Sendgrid (send emails)
Request Created
Azure AD
If not in Azure Redis , Azure SQL
Request types
Azure Redis
SQL
Azure SQL
Search Bar
CosmosDB (Request Repository)
10 01
Storage Account Blob (request reports/ files)
REQUEST

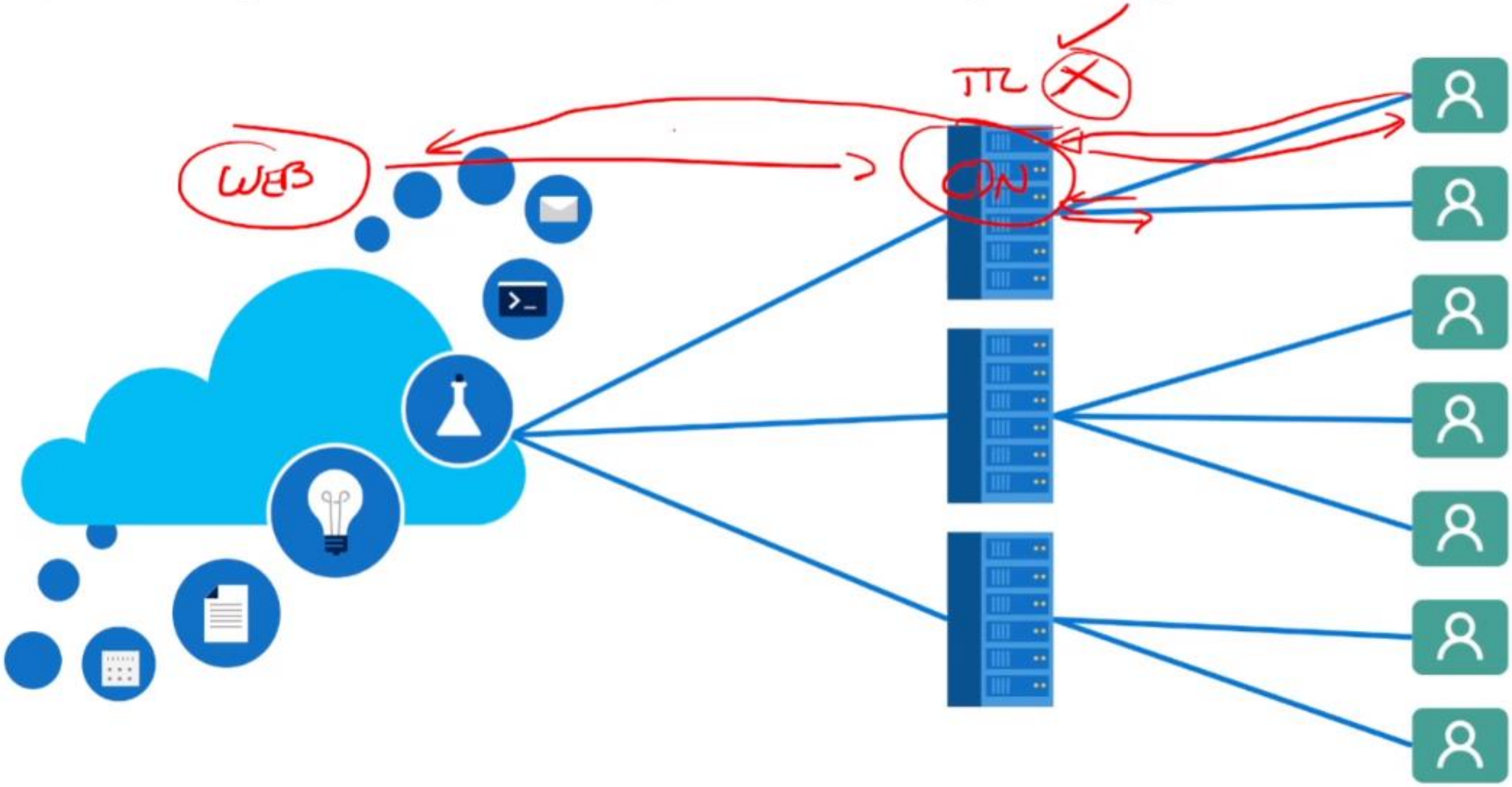


https://channel9.msdn.com/Shows/Azure-Friday/Scale-Your-Cloud-App-with-Azure-Cache-for-Redis?w...DX_880918

https://docs.microsoft.com/en-gb/shows/Azure-Friday/Scale-Your-Cloud-App-with-Azure-Cache-for-Redis?wt.mc_id=DX_880918

```
}

public async Task<IList<ConsultType>> GetConsultTypesAsync()
{
    IList<ConsultType> consultTypes;
    var consultTypesJson = await this.cache.GetStringAsync("ConsultTypes");
    if (consultTypesJson != null)
    {
        // We have cached data, deserialize the JSON data. // //
        consultTypes = JsonConvert.DeserializeObject<IList<ConsultType>>(consultTypesJson);
    }
    else
    {
        // There's nothing in the cache, retrieve it from the repository and cache it.
        consultTypes = await this.database.ConsultTypes.OrderBy(ct => ct.Sequence).ToListAsync();
        consultTypesJson = JsonConvert.SerializeObject(consultTypes);
        await this.cache.SetStringAsync("ConsultTypes", consultTypesJson);
    }
    return consultTypes;
}
```

Improving the client experience by using a CDN



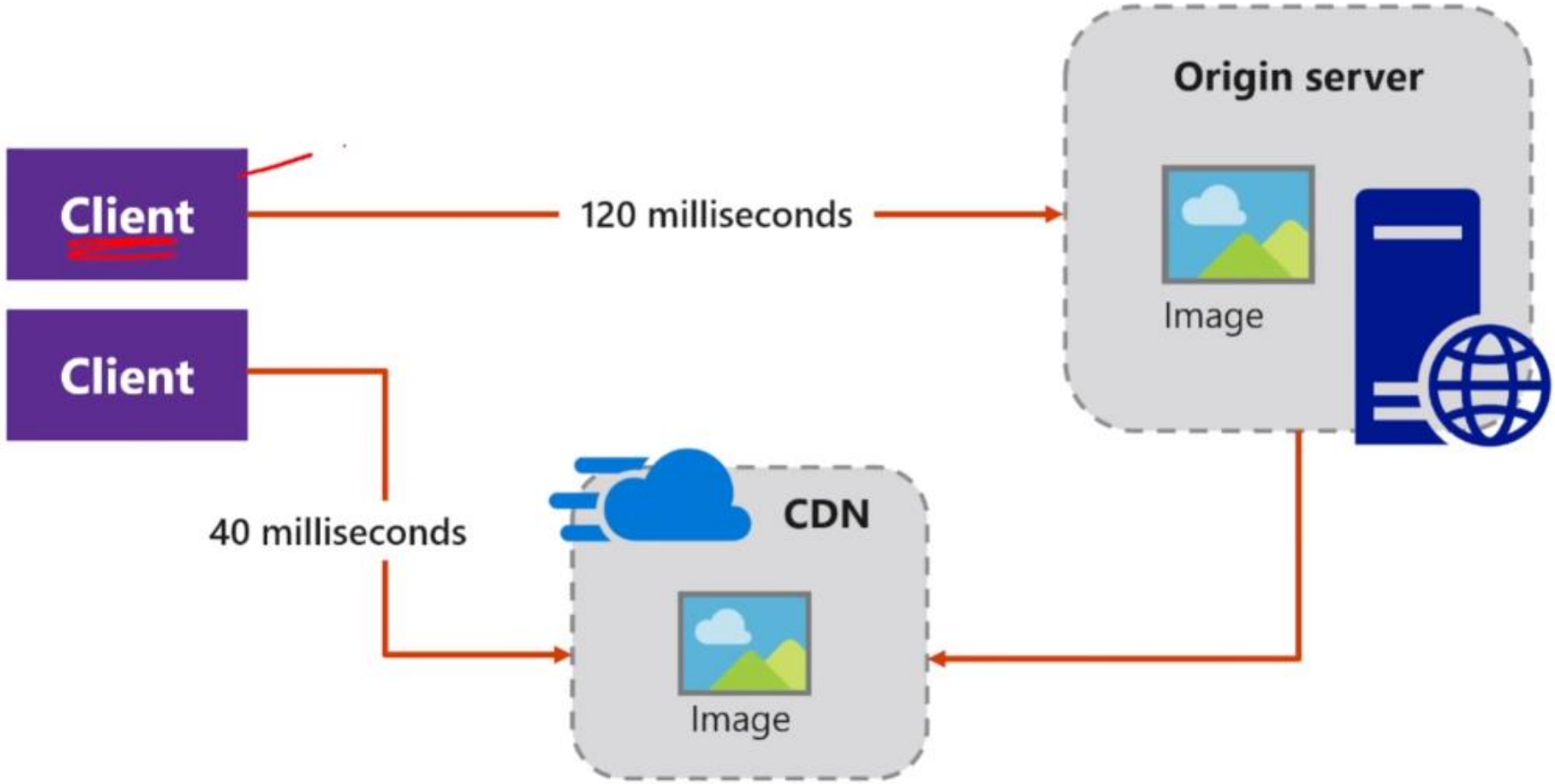
CDN uses

- Delivering static resources, often **from a website, for client applications**
  - Resources can be images, style sheets, documents, files, client-side scripts, HTML pages, HTML fragments, or any other content that the server does not need to modify for each request
- Delivering public static and shared **content to devices such as mobile phones and tablets**
- Serving **entire websites** that consist of **only public static content** to clients
  - Does not require any dedicated compute resources

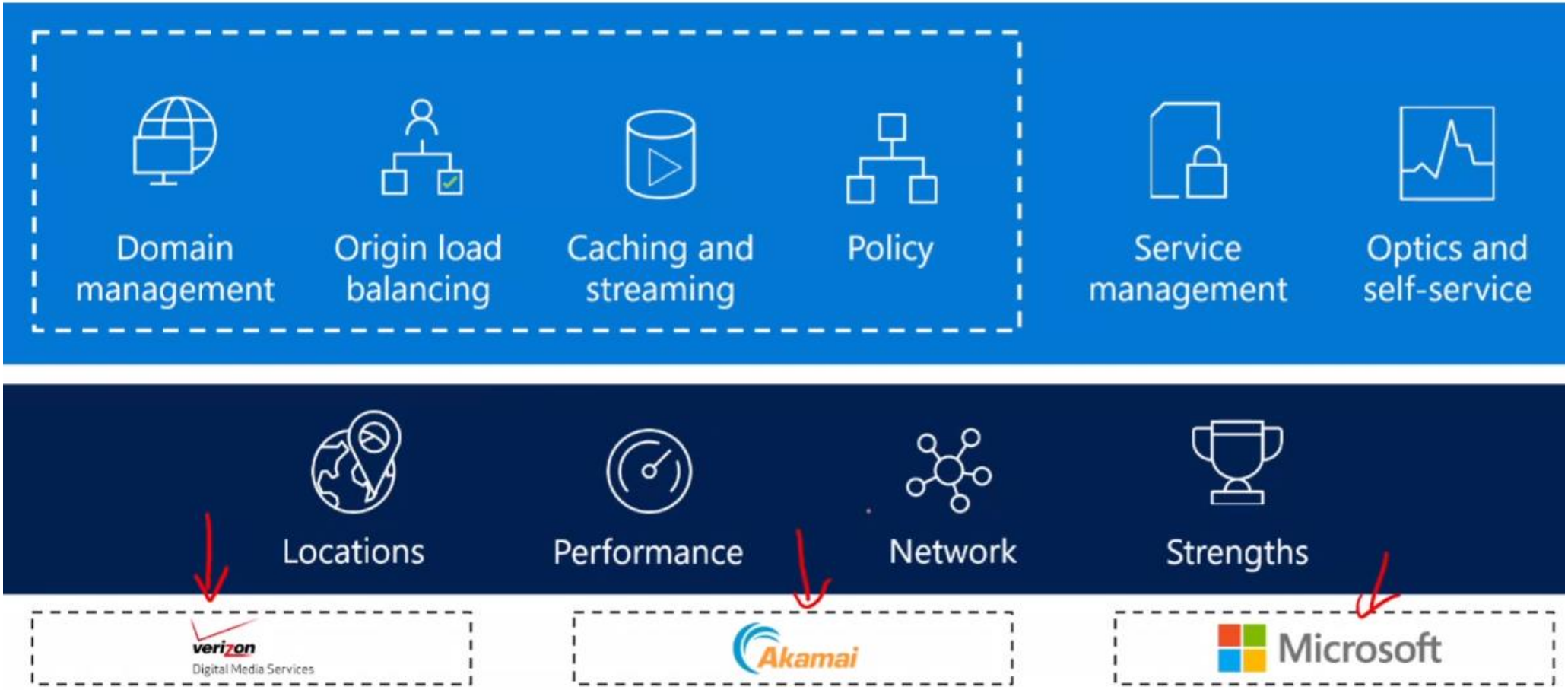
CDN uses (continued)

- **Streaming video files** to client devices on demand
  - **Taking advantage of the low latency and reliable connectivity** available from the globally located datacenters that offer CDN connections
- Supporting **Internet of Things (IoT)** solutions
  - The huge numbers of devices and appliances involved in an IoT solution can easily overwhelm an application if it has to **distribute firmware updates directly to each device**
- Coping with peaks and surges in demand without requiring the application to scale
  - Avoiding the consequent increased running costs associated with scale

CDN usage


Origin server
Client
120 milliseconds
Image
Client
40 milliseconds
CDN
Image

Azure CDN platform

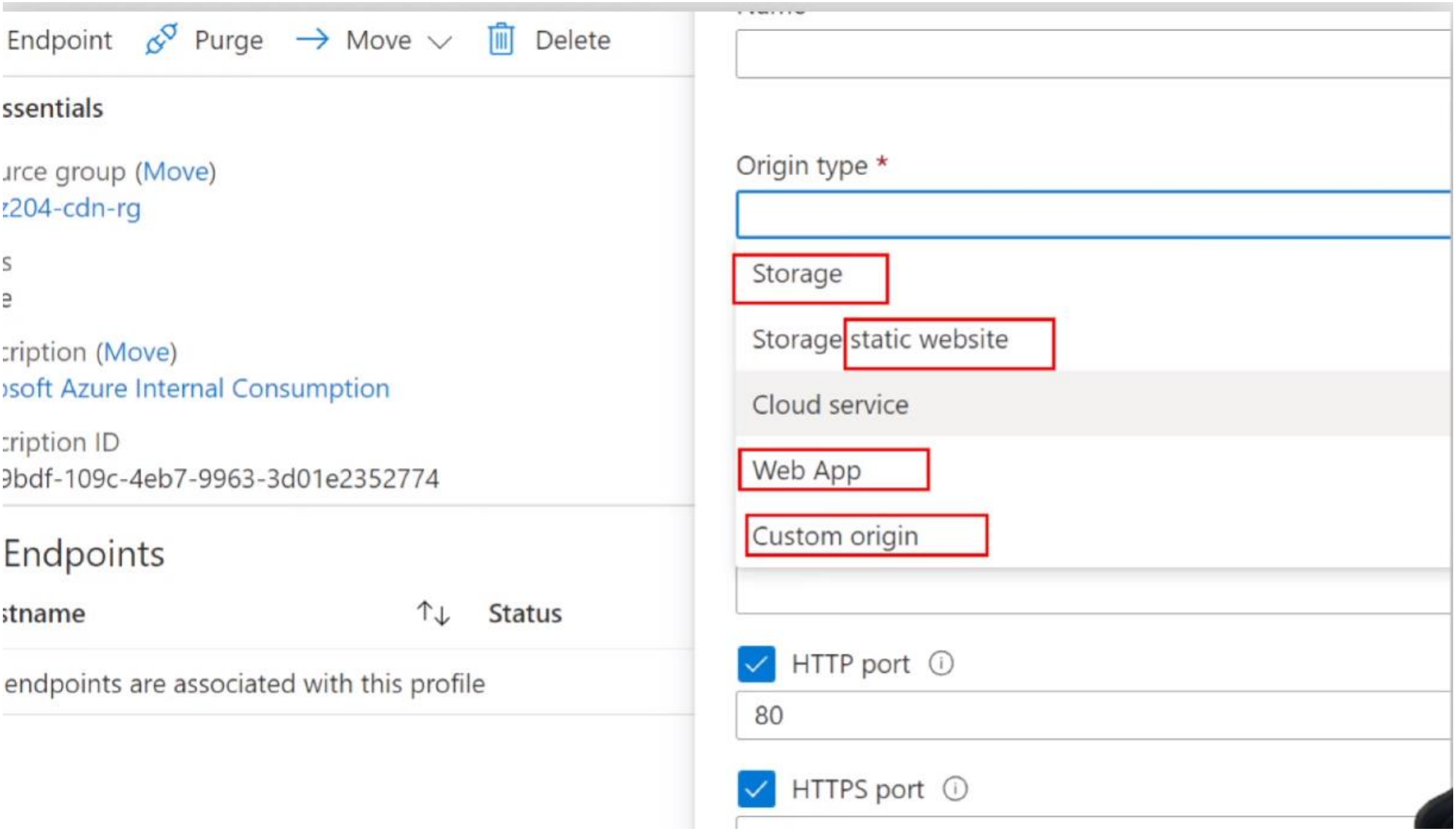


Manage Azure CDN profiles by using Azure CLI

```
az cdn profile list

az cdn profile list --resource-group ExampleGroup

az cdn profile create --name DemoProfile --resource-group ExampleGroup –sku
Standard_Akamai
```

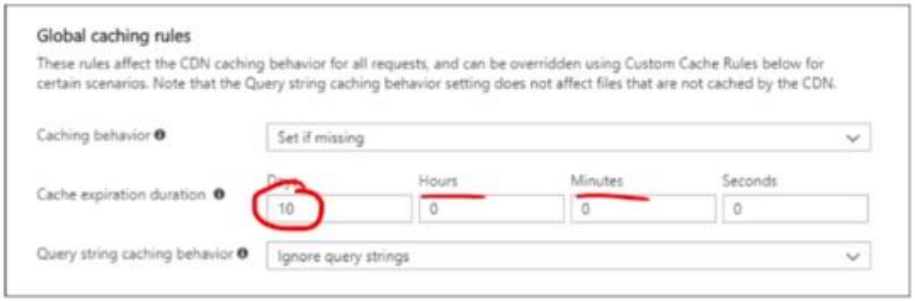
You can customize further by using one of the following options:

Custom_Verizon
Premium_Verizon
Standard_Akamai
Standard_ChinaCdn
Standard_Verizon
Standard_Microsoft


Endpoint
Purge
Move
Delete
ssentials
urce group (Move)
z204-cdn-rg
s
e
cription (Move)
soft Azure Internal Consumption
cription ID
9bdf-109c-4eb7-9963-3d01e2352774
Endpoints
stname
Status
endpoints are associated with this profile
Origin type *
Storage
Storage static website
Cloud service
Web App
Custom origin
HTTP port
80
HTTPS port

Cache expiration in Azure CDN (Akamai/Verizon)

- Azure CDN caching rules specify cache expiration behavior both globally and with custom conditions. There are two types of **caching rules**:
  - **Global caching rules**. affects all requests to the endpoint. The global caching rule overrides any HTTP cache-directive headers, if set.
  - **Custom caching rules for each endpoint** in your profile. Custom caching rules match specific paths and file extensions, are processed in order, and **override the global caching rule, if set**.



Purging and preloading assets by using Azure CLI

Purge assets from an endpoint:

```
az cdn endpoint purge --content-paths '/css/*' '/js/app.js' --name ContosoEndpoint --
profile-name DemoProfile --resource-group ExampleGroup
```

Preload assets into an endpoint:

```
az cdn endpoint load --content-paths '/css/*' '/js/app.js' --name ContosoEndpoint --
profile-name DemoProfile --resource-group ExampleGroup
```