Cloud Infrastructure Week#4

Emrah Mutlu

January 2024

AGENDA – WEEK#4

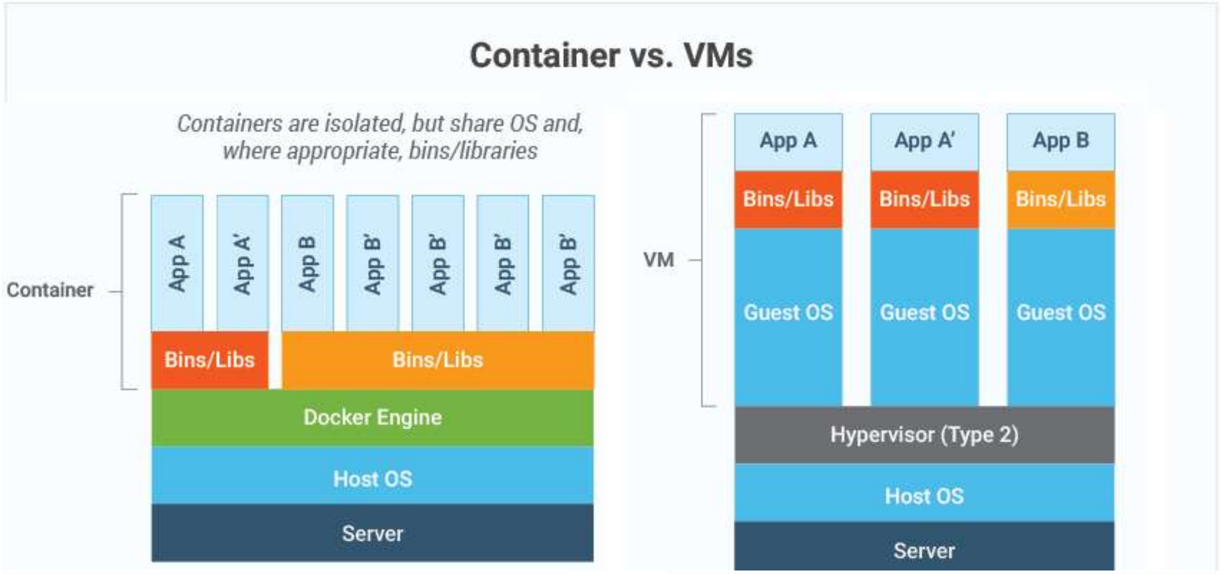
- Cloud Computing Services
  - Kubernetes (K8s) Services / AKS
  - Container Apps
  - Container Instances
  - Azure Functions
  - How to Choose a Compute Service

- Lab Sessions:
  - Lab: Testing compute instances on Cloud

Orion Innovation™

Computing Services

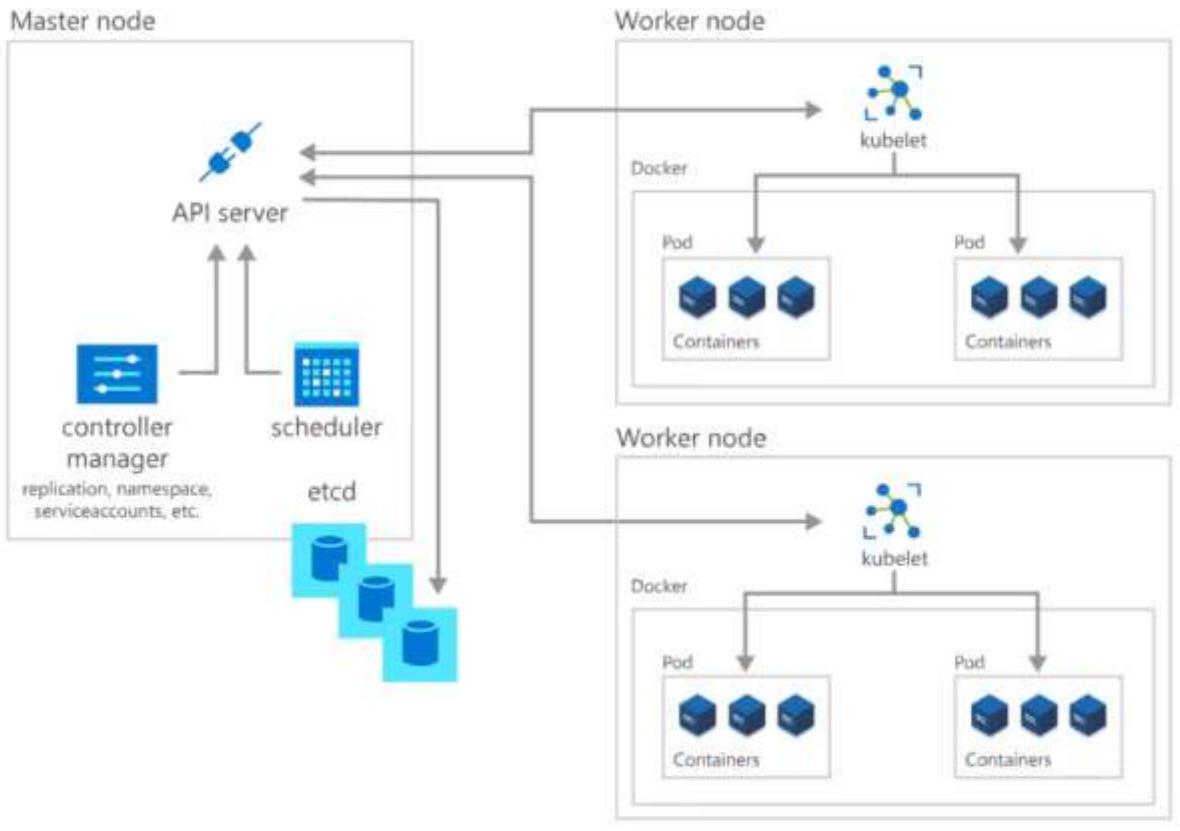
DIFFERENCES BETWEEN THE VM AND CONTAINERS

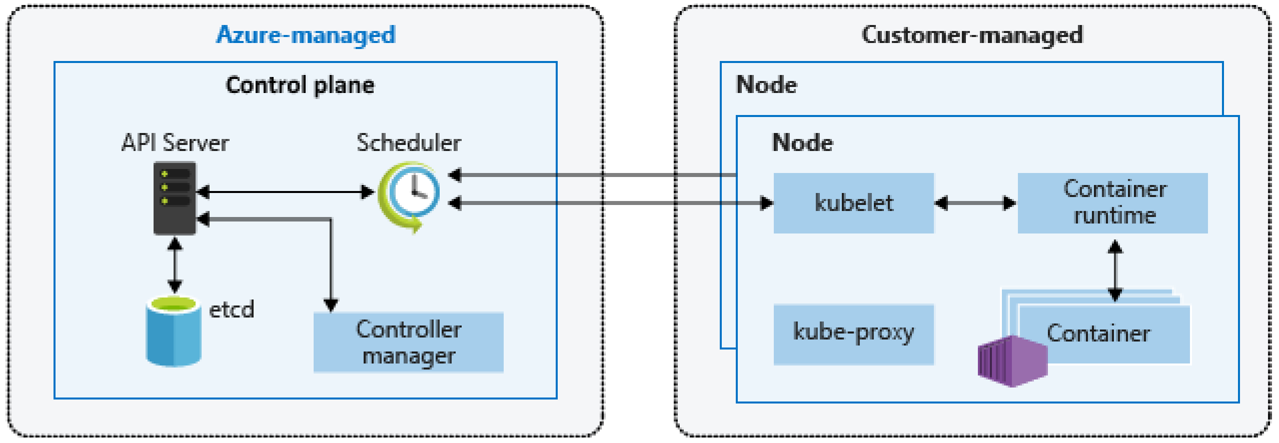


USE CASES	Target developer and application management; also ideal for development and testing	Target developer and application management; also ideal for development and testing	USE CASES
SECURITY	Sharing OS requires root access, which may lead to data access vulnerability	Entire OS is emulated in each VM, thereby offering better data security and protection	SECURITY
BUSINESS BENEFIT	Agile development speeds up go-to-market of products	Automation, easy maintenance, availability, and recovery	BUSINESS BENEFIT
COSTS BENEFITS	Efficient use of resources maximizes use of infrastructure, reducing overall TCO	Less hardware to manage and upgrade; energy savings from fewer servers lower operating costs	COSTS BENEFITS
SHIFT TO CLOUD	Easily scale and provide elastic cloud services	Fast duplication, scale, and workload mobility make it easy to host VMs in the cloud	SHIFT TO CLOUD

AZURE K8S SERVICES (ARCHITECTURE)

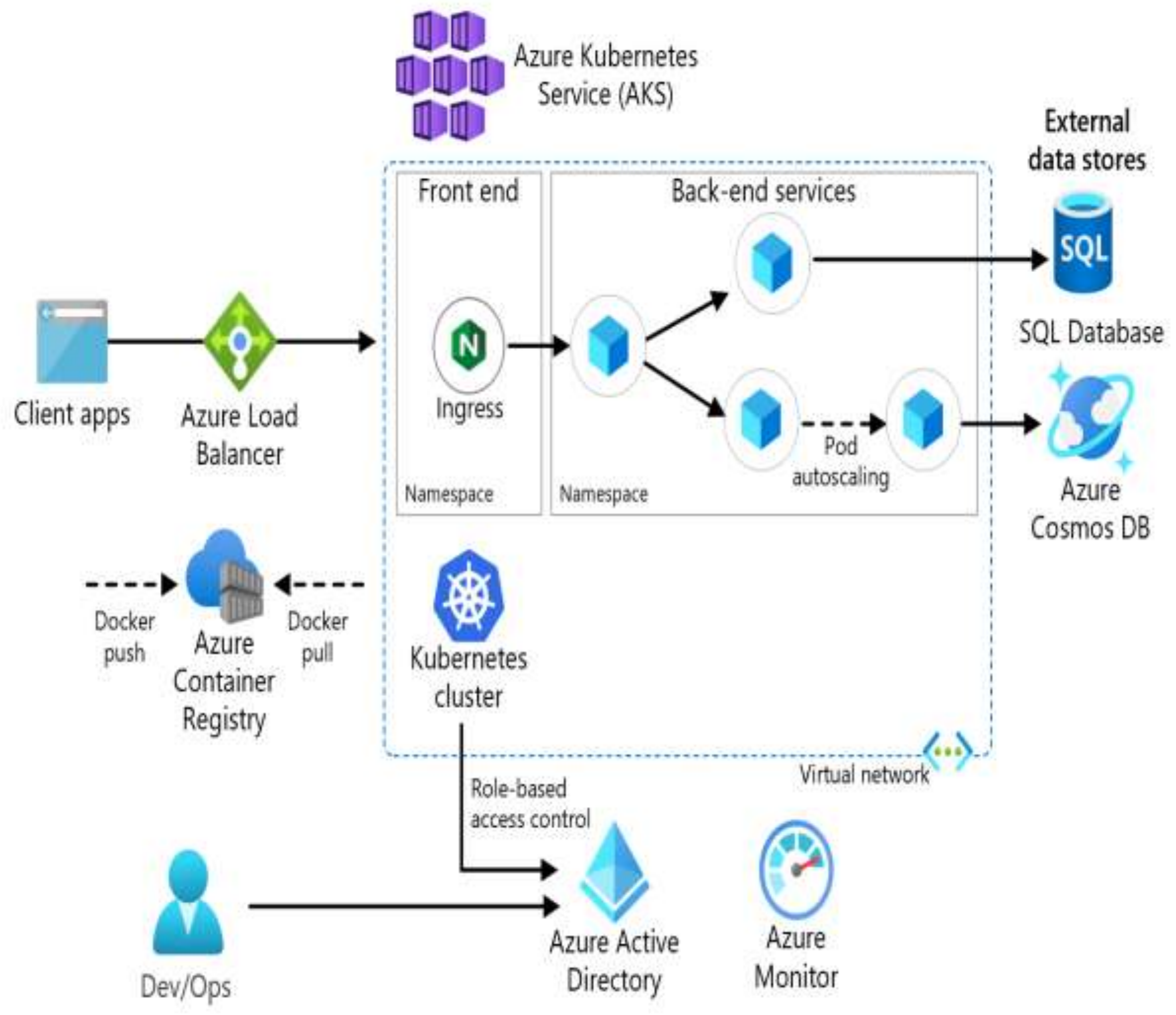
- **Kubernetes** is a cluster of virtual or on-premises machines. These machines—called **nodes**—share compute, network, and storage resources. Each cluster has one **master node** connected to one or more worker nodes. The **worker nodes** are responsible for running groups of containerized applications and workloads, known as **pods**, and the master node manages which pods run on which worker nodes.
- Kubernetes cluster is divided into two components:
  - **Control plane**: provides the core Kubernetes services and orchestration of application workloads.
  - **Nodes**: run your application workloads.
- A **Kubernetes cluster** contains at least one main plane and one or more nodes. Both the control planes and node instances can be physical devices, virtual machines, or instances in the cloud. The default host OS in Kubernetes is Linux, with default support for Linux-based workloads.
- **The Kubernetes control plane** in a Kubernetes cluster runs a collection of services that manage the orchestration functionality in Kubernetes.
- **A node** in a Kubernetes cluster is where your compute workloads run. Each node communicates with the control plane via the API server to inform it about state changes on the node.
- Developers and operators **interact with the cluster** primarily through the master node by using **kubectl**, a command-line interface that installs on their local OS. Commands issued to the cluster through kubectl are sent to the **kube-apiserver**, the Kubernetes API that resides on the master node. The kube-apiserver then communicates requests to the **kube-controller-manager** in the master node, which is in turn responsible for handling worker node operations. Commands from the master node are sent to the **kubelet** on the worker nodes.
- The **master node** maintains the **current state of the Kubernetes cluster** and configuration in the **etcd**, a **key value store database**. To run pods with your containerized apps and workloads, you must describe a new desired state to the cluster in the form of a **YAML file**. The **kube-controller-manager** takes the YAML file and tasks the **kube-scheduler** with deciding which worker nodes the app or workload should run based on predetermined constraints. Working in concert with each worker node's **kubelet**, the **kube-scheduler starts the pods**, watches the state of the machines, and is **overall responsible for the resource management**.





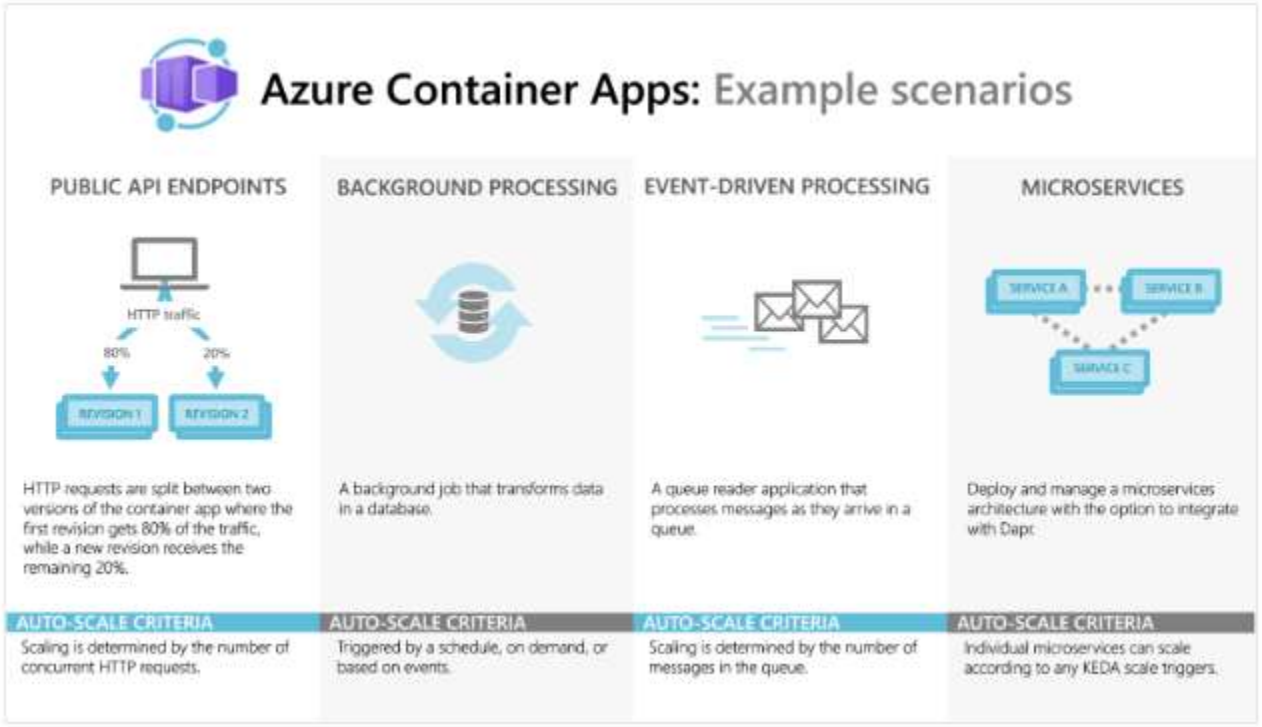
AZURE K8S SERVICES (AKS) IN CHARGE

- **Azure Kubernetes Service** (AKS). AKS is a managed Kubernetes cluster hosted in the Azure cloud. Azure manages the Kubernetes API service, and you only need to manage the agent nodes.
- **Virtual network**. By default, AKS creates a virtual network into which agent nodes are connected. You can create the virtual network first for more advanced scenarios, which lets you control things like subnet configuration, on-premises connectivity, and IP addressing.
- **Ingress**. An ingress server exposes HTTP(S) routes to services inside the cluster.
- **Azure Load Balancer**. Once the AKS cluster is created, the cluster is ready to use the load balancer. Then, once the NGINX service is deployed, the load balancer is configured with a new public IP that will front your ingress controller. This way, the load balancer routes internet traffic to the ingress.
- **External data stores**. Microservices are typically stateless and write state to external data stores, such as Azure SQL Database or Azure Cosmos DB.
- **Azure Active Directory**. AKS uses an Azure Active Directory (Azure AD) identity to create and manage other Azure resources such as Azure load balancers. Azure AD is also recommended for user authentication in client applications.
- **Azure Container Registry**. Use Container Registry to store private Docker images, which are deployed to the cluster. AKS can authenticate with Container Registry using its Azure AD identity. AKS doesn't require Azure Container Registry. You can use other container registries, such as Docker Hub.
- **Azure Monitor**. Azure Monitor collects and stores metrics and logs, application telemetry, and platform metrics for the Azure services. Use this data to monitor the application, set up alerts, dashboards, and perform root cause analysis of failures. Azure Monitor integrates with AKS to collect metrics from controllers, nodes, and containers.



CONTAINER APPS

- Azure Container Apps enables you to build serverless microservices and jobs based on containers.
  - Container Apps is optimized for running general purpose containers, especially for applications that span many microservices deployed in containers.
  - Container Apps is powered by Kubernetes and open-source technologies like Dapr, KEDA, and envoy.
  - Container Apps supports Kubernetes-style apps and microservices with features like service discovery and traffic splitting.
  - Container Apps enables event-driven application architectures by supporting scale based on traffic and pulling from event sources like queues, including scale to zero.
  - Container Apps supports running on demand, scheduled, and event-driven jobs.
- Azure Container Apps doesn't provide direct access to the underlying Kubernetes APIs. If you require access to the Kubernetes APIs and control plane, you should use Azure Kubernetes Service.



HTTP requests are split between two versions of the container app where the first revision gets 80% of the traffic, while a new revision receives the remaining 20%.

A background job that transforms data in a database.

A queue reader application that processes messages as they arrive in a queue.

Deploy and manage a microservices architecture with the option to integrate with Dapr.

Scaling is determined by the number of concurrent HTTP requests.

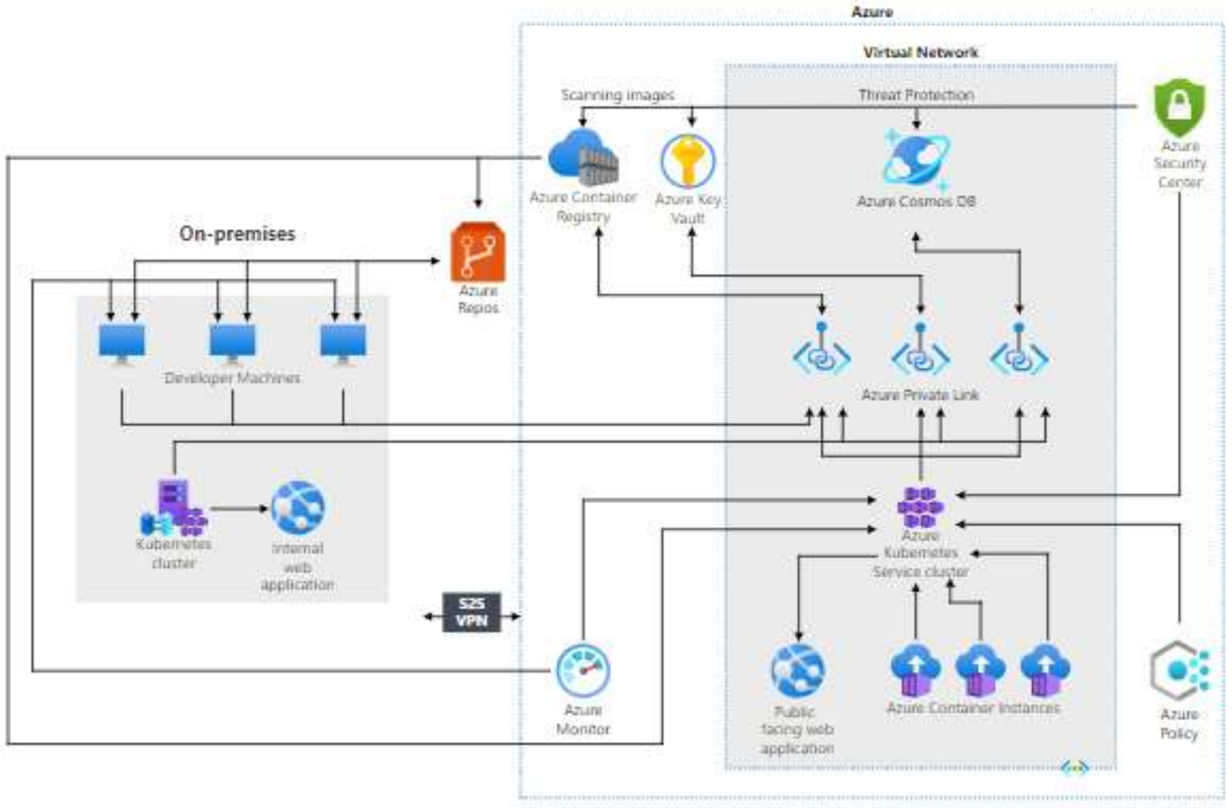
Triggered by a schedule, on demand, or based on events.

Scaling is determined by the number of messages in the queue.

Individual microservices can scale according to any KEDA scale triggers.

CONTAINER INSTANCES

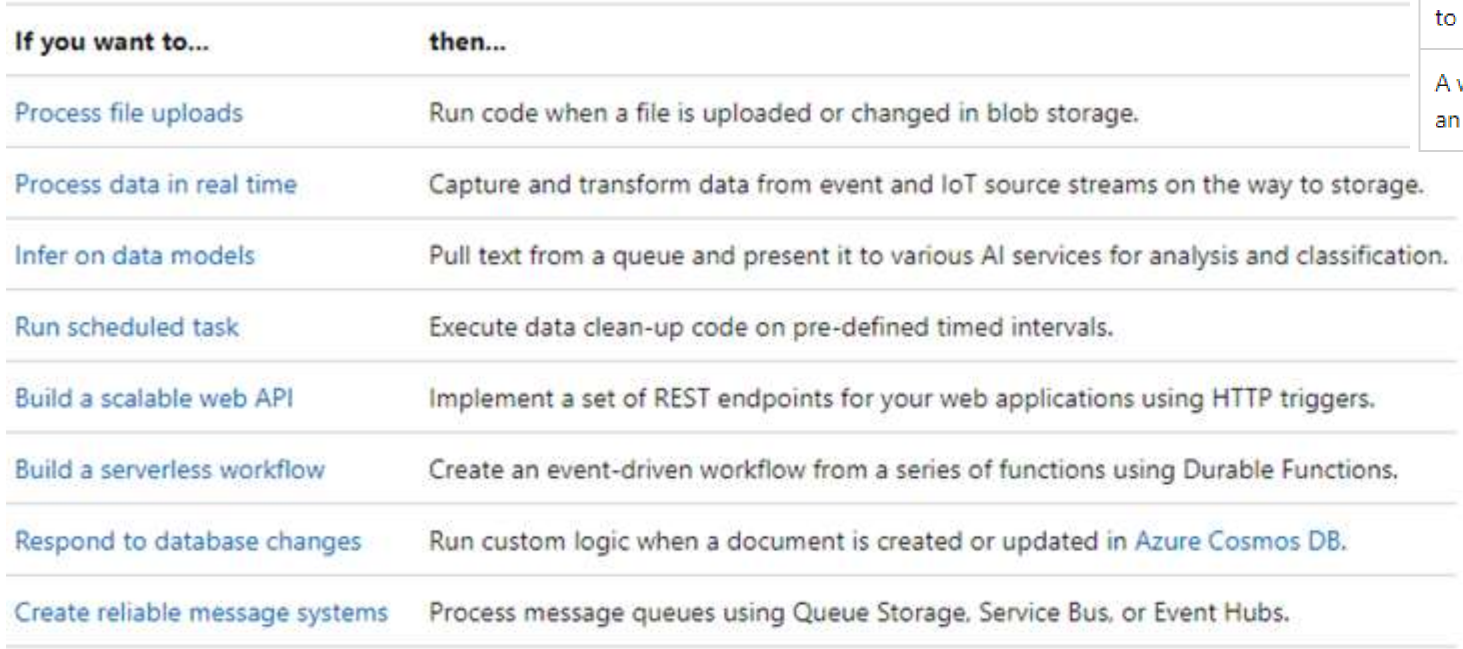
- Azure Container Instances (ACI) provides a single pod of Hyper-V isolated containers on demand.
- It's a simpler and more flexible option than Container Apps.
- Concepts like scale, load balancing, and certificates aren't provided with ACI containers.
- Users often interact with Azure Container Instances through other services.
  - For example, Azure Kubernetes Service can layer orchestration and scale on top of ACI through virtual nodes.
  - If you need a less "opinionated" building block that doesn't align with the scenarios Azure Container Apps is optimizing for, Azure Container Instances is an ideal option.



Microsoft Azure

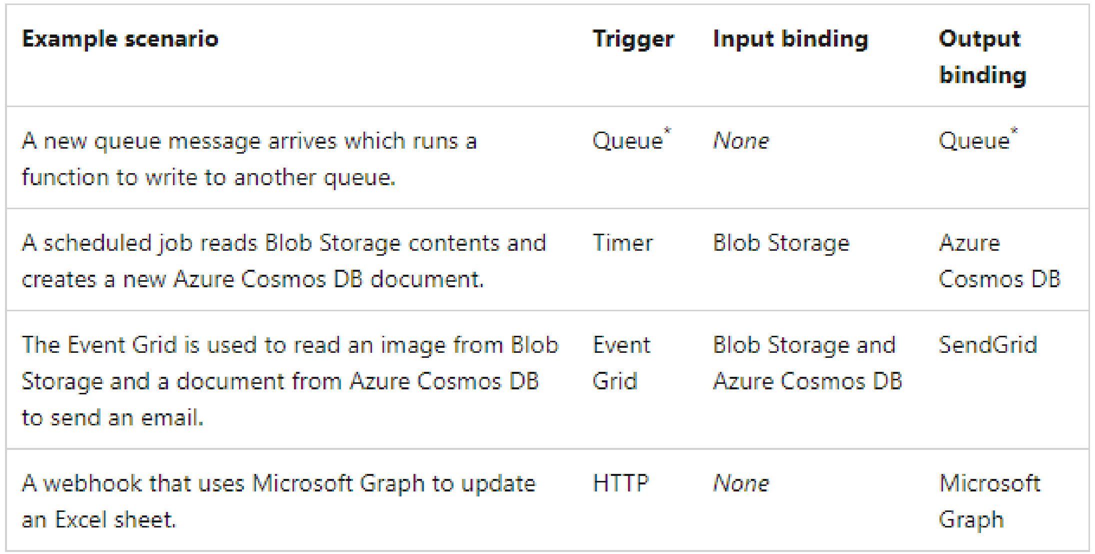
AZURE FUNCTIONS

- **Azure Functions** is a **serverless** solution that allows you to write less code, maintain less infrastructure, and save on costs.
- **Functions provides** a comprehensive set of **event-driven *triggers and bindings*** that connect your functions to other services without having to write extra code.
- The following are a common set of **integrated scenarios** that feature Functions.



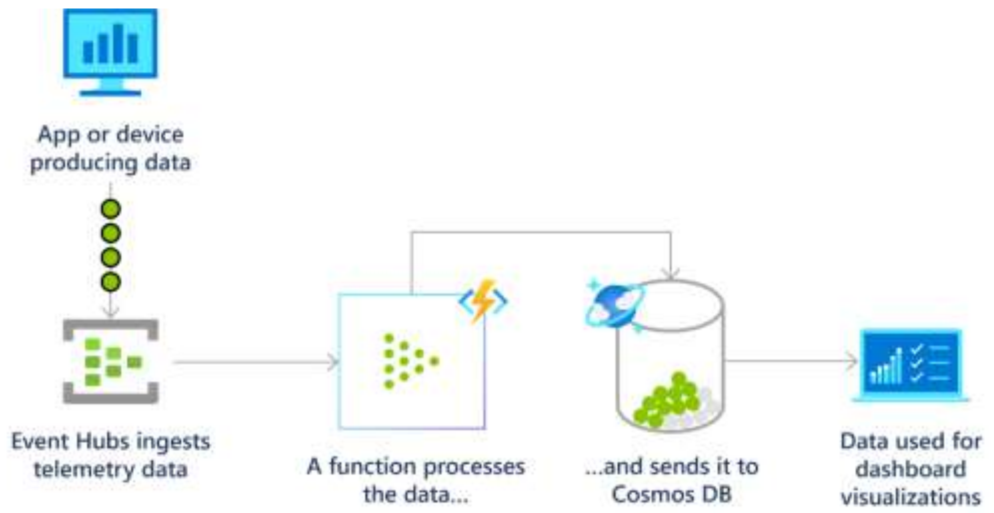
If you want to...	then...
Process file uploads	Run code when a file is uploaded or changed in blob storage.
Process data in real time	Capture and transform data from event and IoT source streams on the way to storage.
Infer on data models	Pull text from a queue and present it to various AI services for analysis and classification.
Run scheduled task	Execute data clean-up code on pre-defined timed intervals.
Build a scalable web API	Implement a set of REST endpoints for your web applications using HTTP triggers.
Build a serverless workflow	Create an event-driven workflow from a series of functions using Durable Functions.
Respond to database changes	Run custom logic when a document is created or updated in Azure Cosmos DB.
Create reliable message systems	Process message queues using Queue Storage, Service Bus, or Event Hubs.

- Functions support C#, Java, JavaScript, PowerShell, Python, TypeScript, Go and Rust programming languages.
- Functions support **Consumption** (pay by each function run) and **Premium** (scale on demand, pre-warmed resources for better performance) plans to save costs.

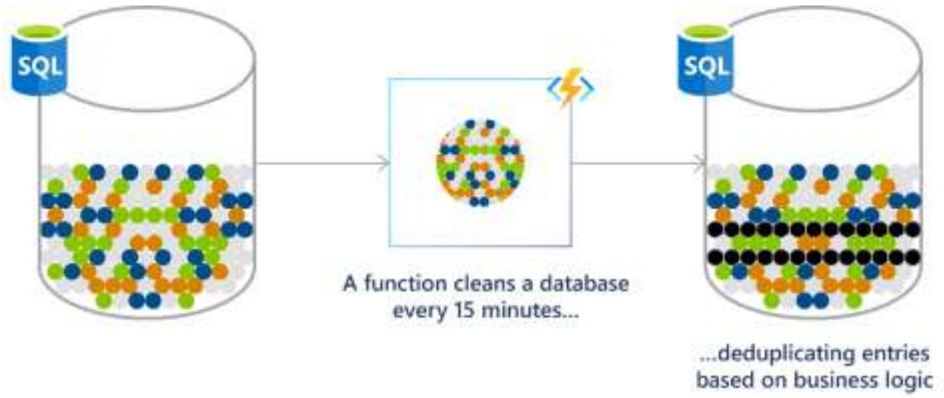


Example scenario	Trigger	Input binding	Output binding
A new queue message arrives which runs a function to write to another queue.	Queue*	*None*	Queue*
A scheduled job reads Blob Storage contents and creates a new Azure Cosmos DB document.	Timer	Blob Storage	Azure Cosmos DB
The Event Grid is used to read an image from Blob Storage and a document from Azure Cosmos DB to send an email.	Event Grid	Blob Storage and Azure Cosmos DB	SendGrid
A webhook that uses Microsoft Graph to update an Excel sheet.	HTTP	*None*	Microsoft Graph

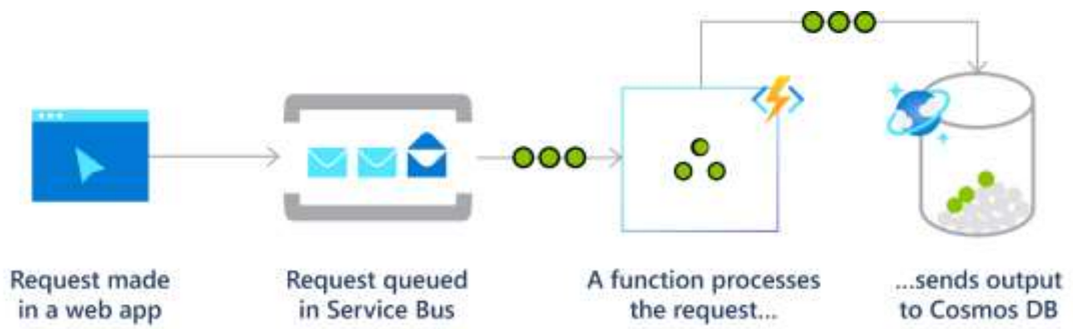
- Real-time stream and event processing


App or device producing data
Event Hubs ingests telemetry data
A function processes the data...
...and sends it to Cosmos DB
Data used for dashboard visualizations

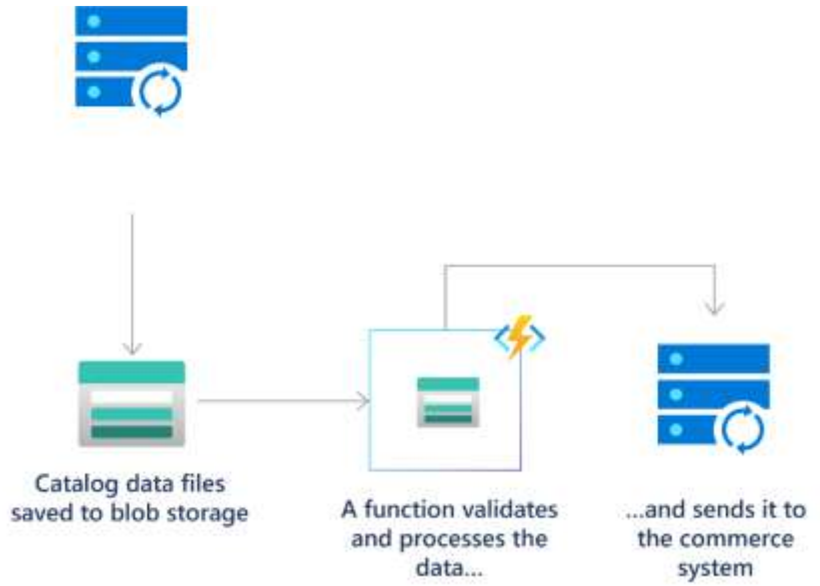
- Scheduled job: DB Clean Up


SQL
A function cleans a database every 15 minutes...
SQL
...deduplicating entries based on business logic

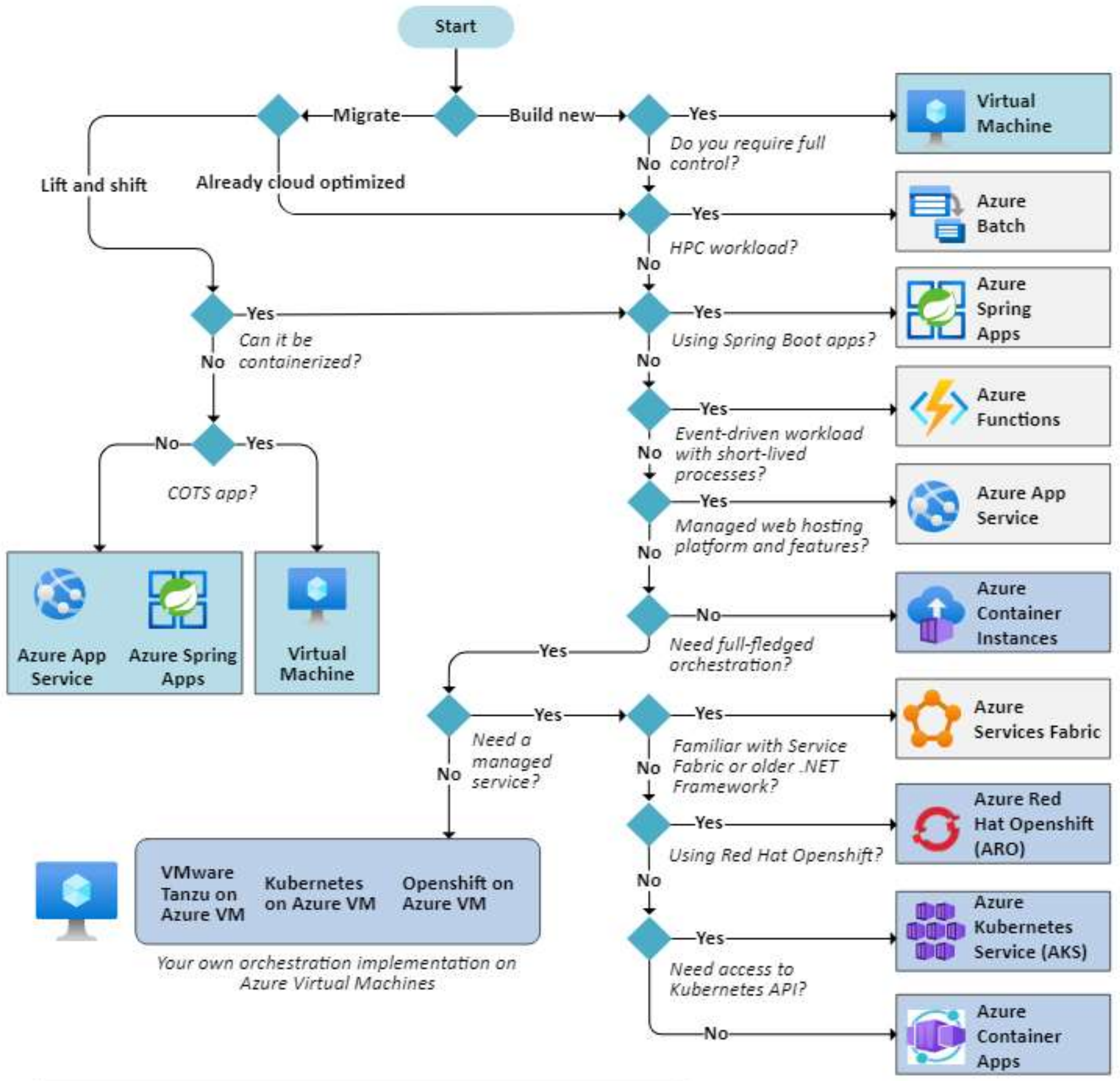
- Reliable message system design


Request made in a web app
Request queued in Service Bus
A function processes the request...
...sends output to Cosmos DB

- Processing file uploads


Catalog data files saved to blob storage
A function validates and processes the data...
...and sends it to the commerce system

HOW TO CHOOSE A COMPUTE SERVICE



Container exclusive services
- Azure Container Instances
- Azure Red Hat Openshift
- Azure Kubernetes Service
- Azure Container Apps
- VMware Tanzu on Azure VM
- Kubernetes on Azure VM
- Openshift on Azure VM

Container compatible services
- Azure Batch
- Azure Functions
- Azure Service Fabric
- Azure Spring Apps
- Azure App Service

Orion Innovation

Lab Session

LAB#9: TESTING COMPUTE INSTANCES ON CLOUD

- Container App:
- Follow the guideline:
  - https://learn.microsoft.com/en-us/azure/container-apps/quickstart-portal
  - https://learn.microsoft.com/en-us/azure/container-apps/get-started?tabs=bash

LAB#10: TESTING COMPUTE INSTANCES ON CLOUD

- AKS Setup & Test
- Follow the instructions:
- https://learn.microsoft.com/en-gb/training/modules/deploy-azure-kubernetes-service-cluster/9-create-azure-kubernetes-service

LAB#11: TESTING COMPUTE INSTANCES ON CLOUD

- Azure Functions setup and test.
- Follow the instructions:
- https://learn.microsoft.com/en-us/training/modules/develop-azure-functions/5-create-function-visual-studio-code

HOMEWORK

- Compare AWS and Azure’s Container & Serverless compute services. Provide a short summary.
- Research and provide a summary about the differences between the Cloud Network and Traditional Network.
- Try to learn more about Network:
  - Subnetting #1: https://www.youtube.com/watch?v=bQ8sdpGQu8c (29 min. watch)
  - Subnetting #2: https://www.youtube.com/watch?v=IGhd-0di0Qo (25 min. watch)

Thank You

orioninc.com