Build containerized application using Docker and Azure

Hamida Rebaï
Cloud Solution Architect & Microsoft MVP

@rebaihamida

Microsoft MVP & MCT

Microsoft MVP in Developer Technologies

Member and Speaker at dotnetfoundation

Blogger and Technical writer

Book Author

Rebai Hamida – Medium

Hamida Rebai Trabelsi | LinkedIn

Rebaï Hamida – YouTube

@rebaihamida

Microsoft MVP

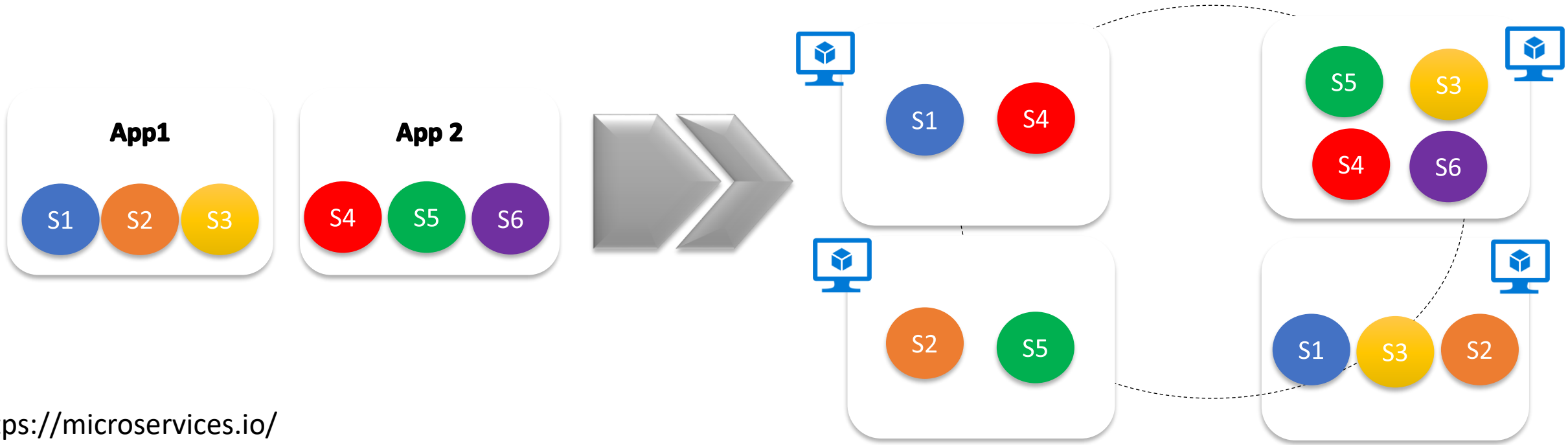
MVP

Microsoft Most Valuable Professional

Table of content

Microservices architecture

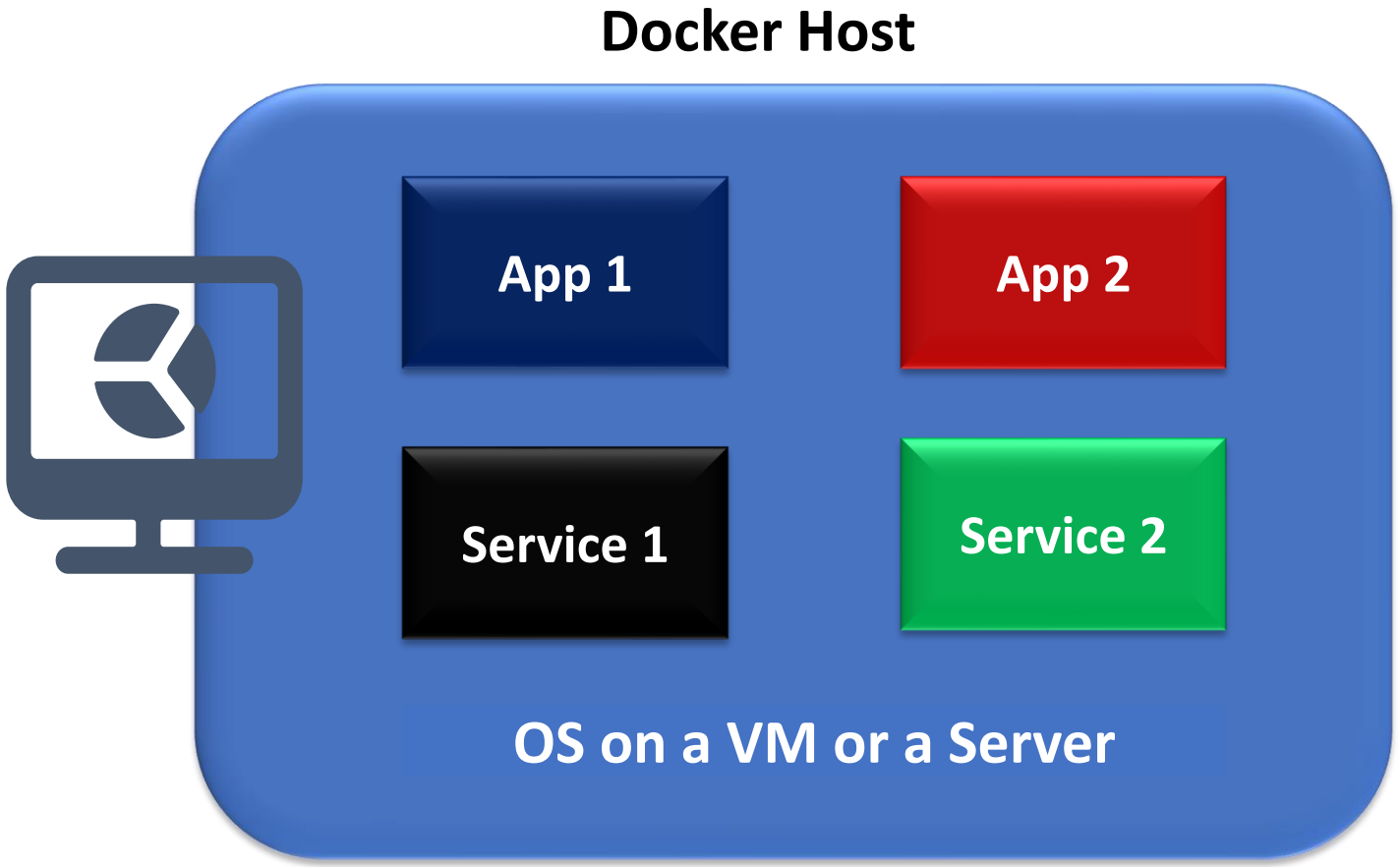
- Model approach for distributed and large or complex critical applications that are based on multiple, independent subsystems in the form of autonomous services.
- Application is built as a collection of services car be developed, tested, versioned, deployed, and scaled.



https://microservices.io/

Microservices ≠ Container

Containers and Docker



Overview Docker container and image

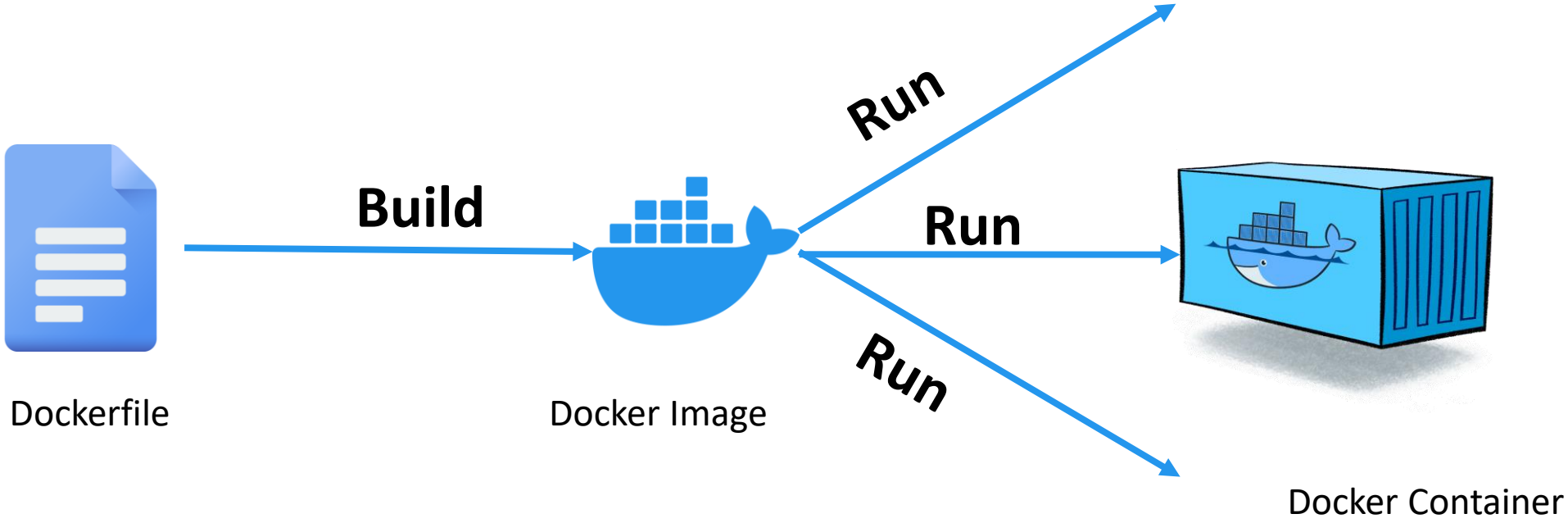
Docker Container

- Virtualized runtime environment used in application development.
- Can use just one machine, share its kernel and virtualize the OS to run more isolated processes.
- Docker containers are **lightweight.**

Docker Images

- Snapshot in other types of VM environments.
- Record of a Docker container at a specific point in time.
- Image can’t be changed, it can be duplicated, shared, or deleted.

Container and image



Dockerfile

Specify image

```
FROM mcr.microsoft.com/dotnet/sdk:7.0 AS build
WORKDIR /src
```

1

Copy project file

```
COPY ["mysolution.csproj", « mysolution/"]
RUN dotnet restore « mysolution.csproj"
```

2

Copy and Build

```
COPY . .
WORKDIR "/src/mysolution"
RUN dotnet build “mysolution.csproj" -c Release -o /app/build
```

3

Build runtime image

```
FROM mcr.microsoft.com/dotnet/aspnet:7.0 AS base
WORKDIR /app
COPY --from=build /app/publish .
```

4

Dockerfile

Starting the app

```
ENTRYPOINT ["dotnet", « yousolution.dll"]
```

5

To optimize publish

Use : -no-restore and p:PublishTrimmed, p:PublishReadyToRun, p:PublishSingleFile

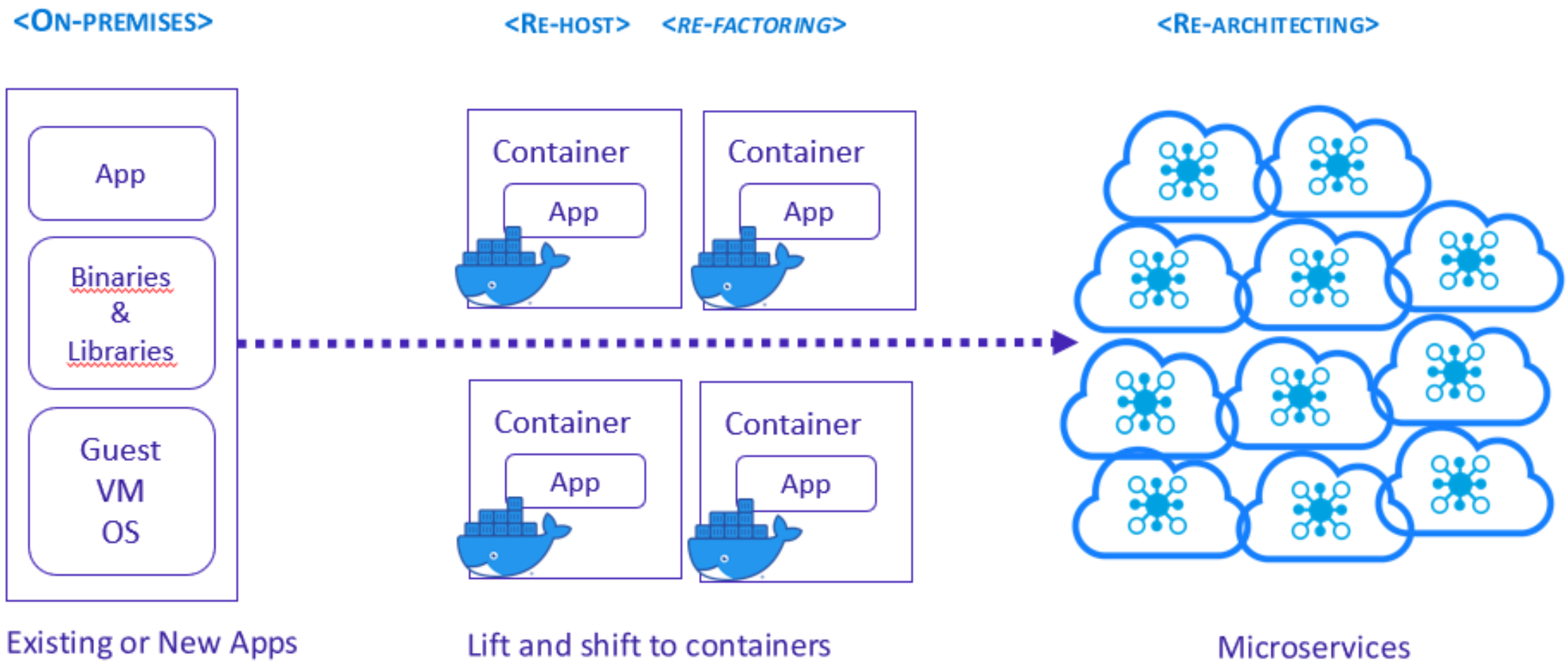
Build our image & start container

docker build -t yoursolutionimage:1.0 .

Create and run our container

docker run –d -p 8080:80 -ti –name myapp --rm yousolutionimage:1.0

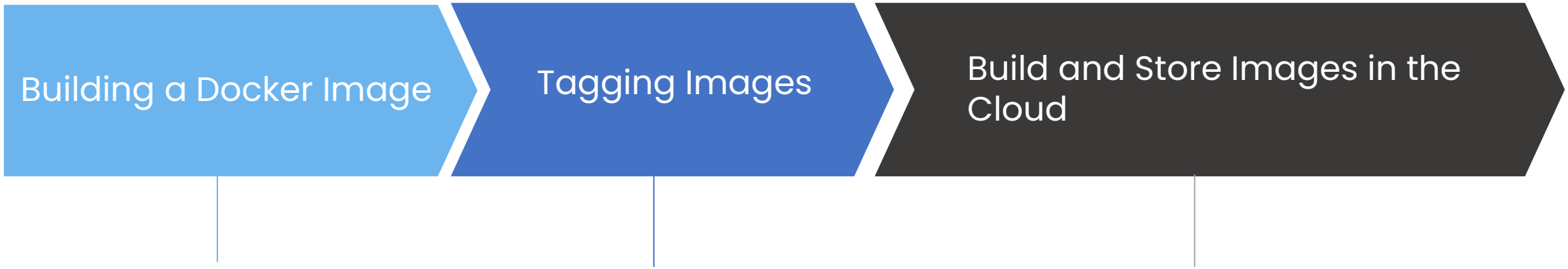
Containers Deployment



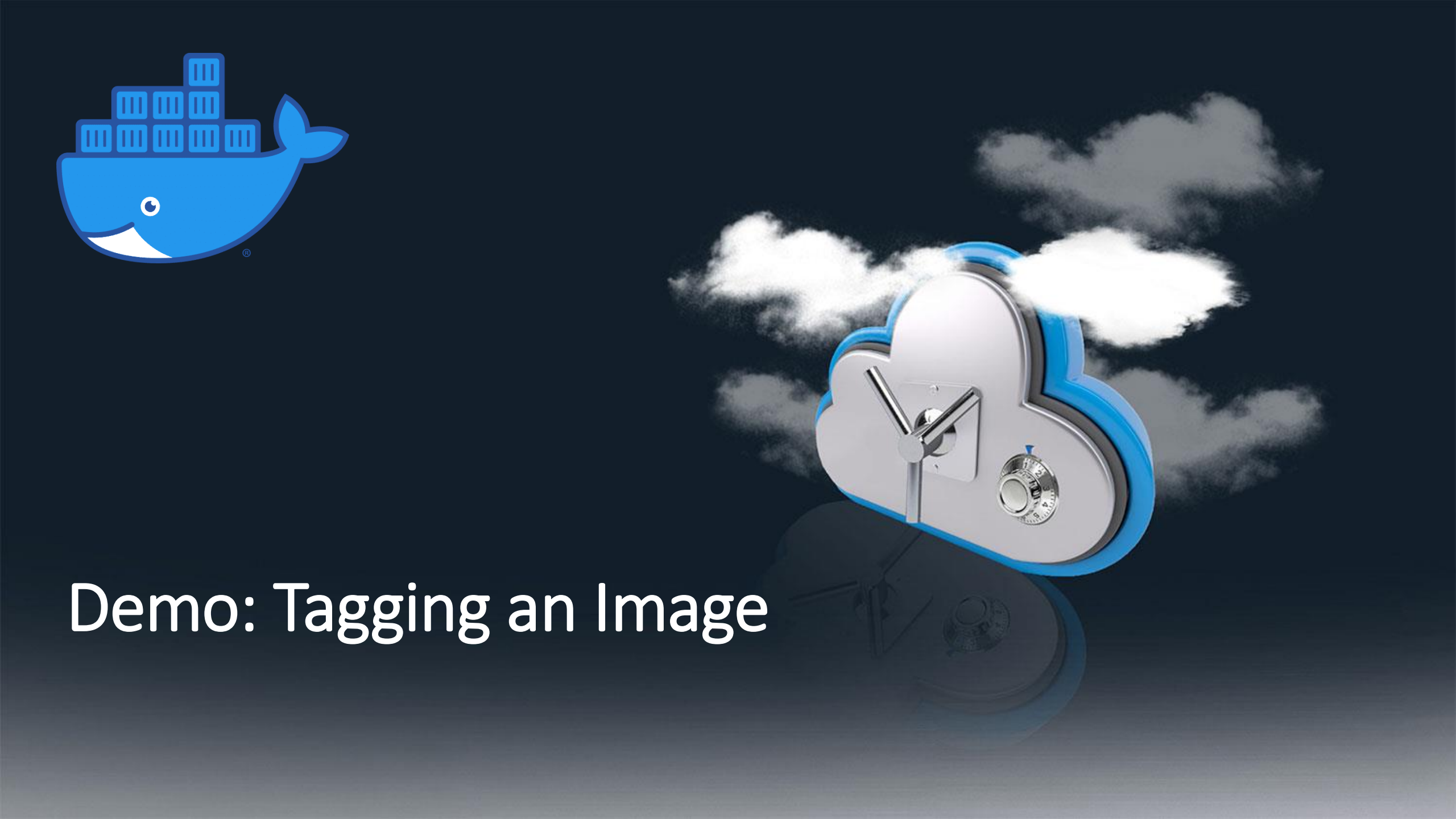
Prerequisites

- Install **Docker Desktop** it is free, it is available for Mac and Windows.
- **Docker Hub** account, it is free.
- Install **Visual Studio 2019 or 2022** or **Visual Studio Code**.
- **Azure** account where we are able to create an azure container registry
- **PowerShell** in Windows or **Azure Cloud Shell**.
- If you are using Visual Studio Code, Microsoft C# for Visual Studio Code, Docker and Azure App Service extensions must be installed.

Demonstration flow



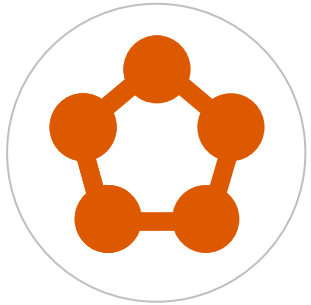
Demo: Building a Docker Image



Demo: Tagging an Image

Containers in Azure: a complete Set of Choices

App Service



Service Fabric

Kubernetes Service

Container Instances

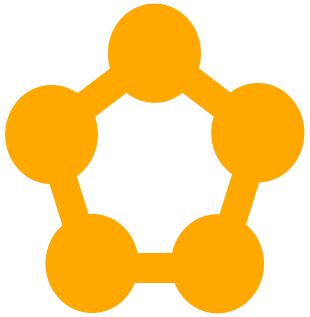
Batch

PaaS vs CaaS vs FaaS

1

PaaS et CaaS

Azure Service Fabric



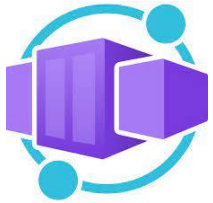
2

CaaS

Azure Container Instance

Azure Container App

Azure Kubernetes Service



3

FaaS

Azure Functions

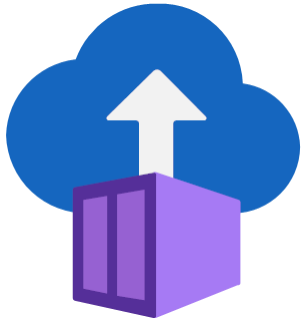
Container and Orchestrator

Store

- Azure Container Registry

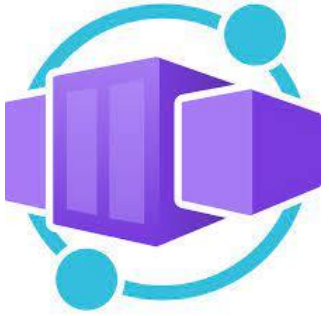
For single container

- Azure Container Instance
- Azure App Service as a Container



For Multiple containers

- Azure Kubernetes Service
- Azure Container App



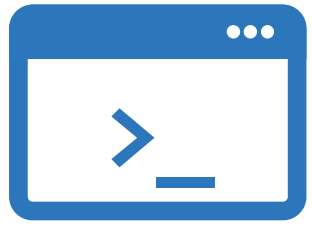

Azure Container Registry

Build and store images by using Azure Container Registry

- What is Container Registry?
  - Azure service that you can use to create your own private Docker registries.
  - Similar to Docker Hub but offers a few unique benefits: Container Registry runs in Azure, Container Registry is highly scalable, providing enhanced throughput for Docker pulls that can span many nodes concurrently.
- Use Container Registry
  - Create a registry by using either the Azure portal or the Azure CLI.
  - Store and host images, and build images.

Create an ACR with Azure CLI

```
az acr create --resource-group dockersamplerg  --
name acrdemoapp --sku Basic --admin-enabled true
```



Build and Store Images by Using Azure Container Registry (ACR)

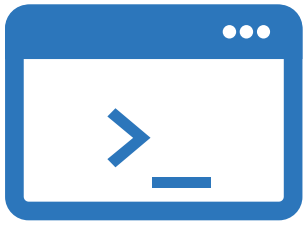

Azure Kubernetes Services

Azure Kubernetes Services

- Azure Kubernetes Service (AKS) is a managed Kubernetes service that lets you quickly deploy and manage clusters.

Create AKS cluster using Azure CLI

az aks create --resource-group aksgr --name myAKSCluster --node-count 1 -- generate-ssh-keys --attach-acr **aksprojectcontainer**



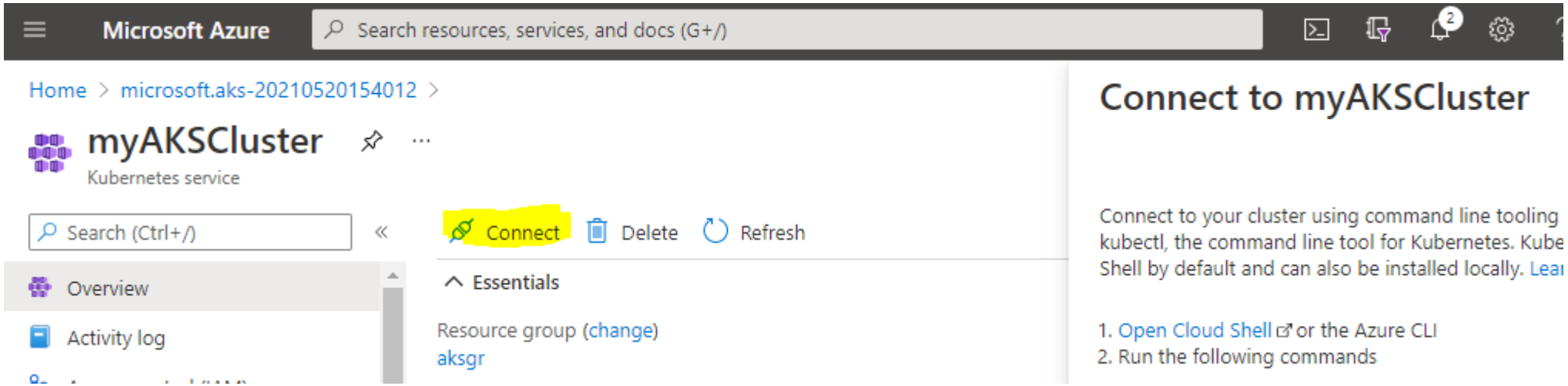
Deploy an Azure Kubernetes Service cluster and run an application using the Azure CLI

1- Open CloudShell My Dashboard — Microsoft Azure and connect to your cluster as bellow:

az account set — subscription yoursubscription

az aks get-credentials — resource-group aksgr — name myAKSCluster

You can find these commands when you open Azure Portal and your cluster:



Deploy an Azure Kubernetes Service cluster and run an application using the Azure CLI

- 2- Create an empty file called: azure-demo-deployment.yaml

- 3- Copy this content to the empty file created, we will describe it after.

Deploy an Azure Kubernetes Service cluster and run an application using the Azure CLI

```
apiVersion: apps/v1
kind: Deployment
metadata:
name: demo-kubernetes-deployment
spec:
selector:
matchLabels:
app: demo-kubernetes-pod
replicas: 1
template:
metadata:
labels:
app: demo-kubernetes-pod
spec:
containers:
-- name: aksprojectcontainer
image:
aksprojectcontainer.azurecr.io
/aksproject:latest
ports:
-- containerPort: 80
```

Deploy an Azure Kubernetes Service cluster and run an application using the Azure CLI

4- Run the application and deploy it in the cluster using the kubectl apply command and specify the name of your YAML manifest.

kubectl apply -f azure-demo-deployment.yaml

We will use kubectl get deployments to verify if the deployment was created or not.

When the application runs, a Kubernetes service exposes the application front end to the internet. This process can take a few minutes to complete.

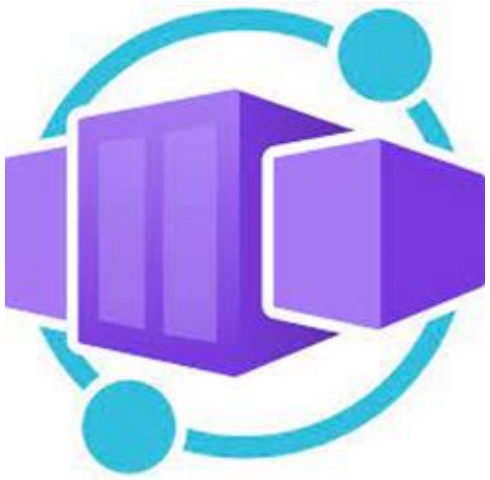
5- we will use kubectl get service command with the --watch argument.

kubectl get service demo-kubernetes-deployment — watch


Azure Container Apps

Azure Container App

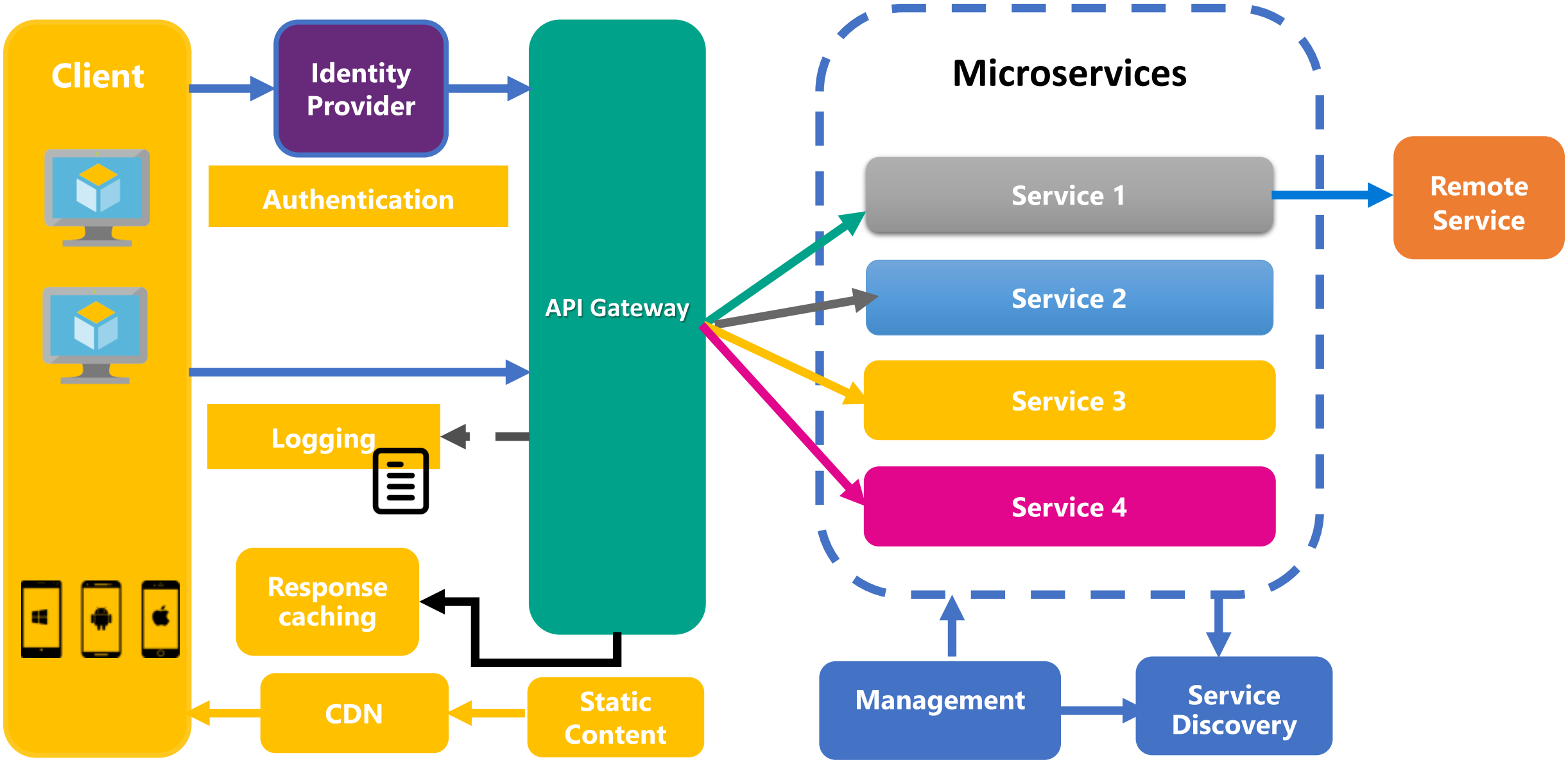
- Provide a serverless hosting service that sits on top of an AKS Service
- Deploy multiple containers
- Azure Container app do not even expose Kubernetes APIs to the users.




packt
A Developer's Guide to Building Resilient Cloud Applications with Azure
Modernize your apps with serverless, event-driven architecture and database-oriented cloud
HAMIDA REBAI TRABELSI


packt
A Developer's Guide to Cloud Apps Using Microsoft Azure
Migrate and modernize your cloud-native applications with containers on Azure using real-world case studies
HAMIDA REBAI TRABELSI

Microservices architecture style



Role of an Architect