

Reactor

MS Learn 學堂
建立 Azure App Service Web
應用程式

Friday 18th February | 3:30 PM HK



Kinfey Lo

Cloud Advocate | Microsoft



Hello , All !



Kinfey Lo – (盧建暉)

Microsoft Cloud Advocate

Former Microsoft MVP , Xamarin MVP and Microsoft RD, with more than 10 years of experience in cloud native, artificial intelligence, and mobile applications, providing application solutions for education, finance, and healthcare. Microsoft Iginte, Teched conference lecturer, Microsoft AI hackathon coach, currently at Microsoft, preaching technology and related application scenarios for technicians and different industries.



I love programming (Python , C# , TypeScript , Swift , Rust , Go)

Focus on artificial intelligence, mobile applications, cloud native

Github : <https://github.com/lokinfey> **Open Source Project :** <https://github.com/SciSharp/TensorFlow.NET>

Email : kinfeylo@microsoft.com **Blog :** <https://blog.csdn.net/kinfey>

Twitter : @Ljh8304

140 available in
140 countries

50 regions
worldwide



Azure 擁有比任何其他雲提供商更多的全球區域 — 提供所需的規模以使應用程序更接近世界各地的用戶、保留數據駐留並為客戶提供全面的合規性和彈性選項。

Azure 資源索取



Free Azure Plan <https://azure.com/free>



Free Azure Student Plan <https://aka.ms/StudentGetAzure>

在 Microsoft Azure 平台上託管應用程序的三種方式



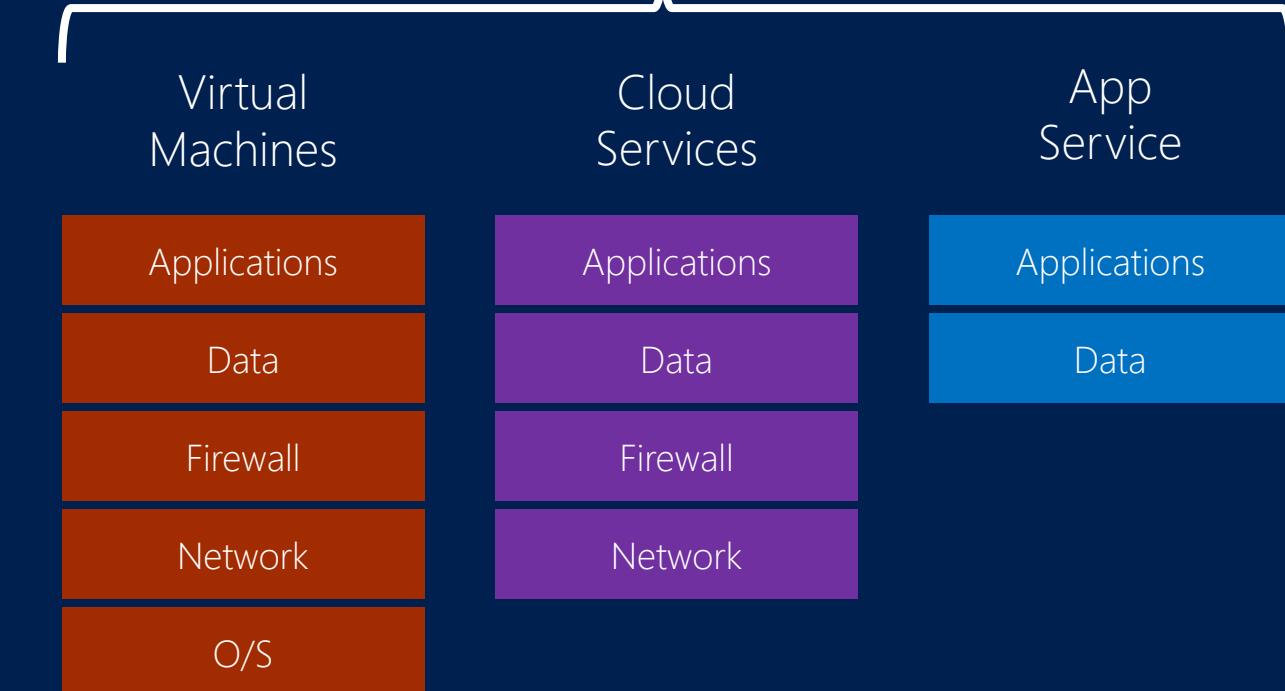
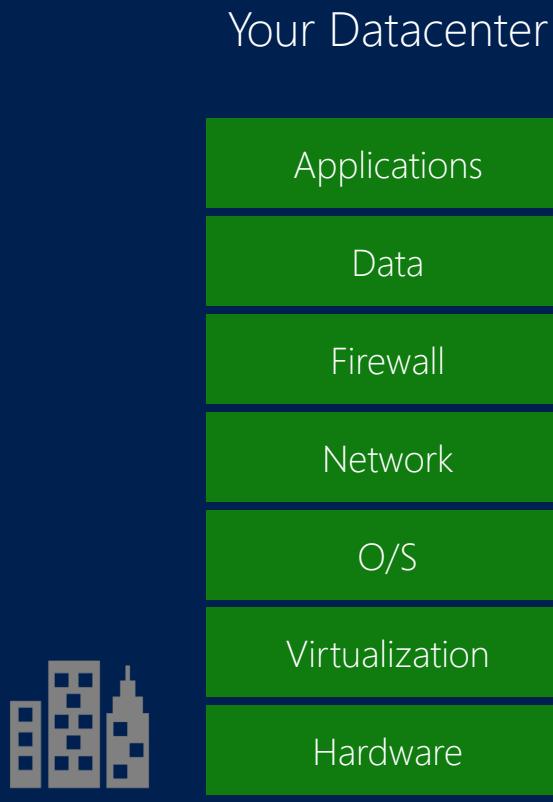
Infrastructure as a Service
IaaS



Platform as a Service
PaaS



AZURE



Focus on the Application

SECURITY & MANAGEMENT



Compute



App Service



Developer Services



HYBRID OPERATIONS



Integration



Media & CDN



Analytics & IoT



Data



INFRASTRUCTURE SERVICES

Compute



Storage



Networking





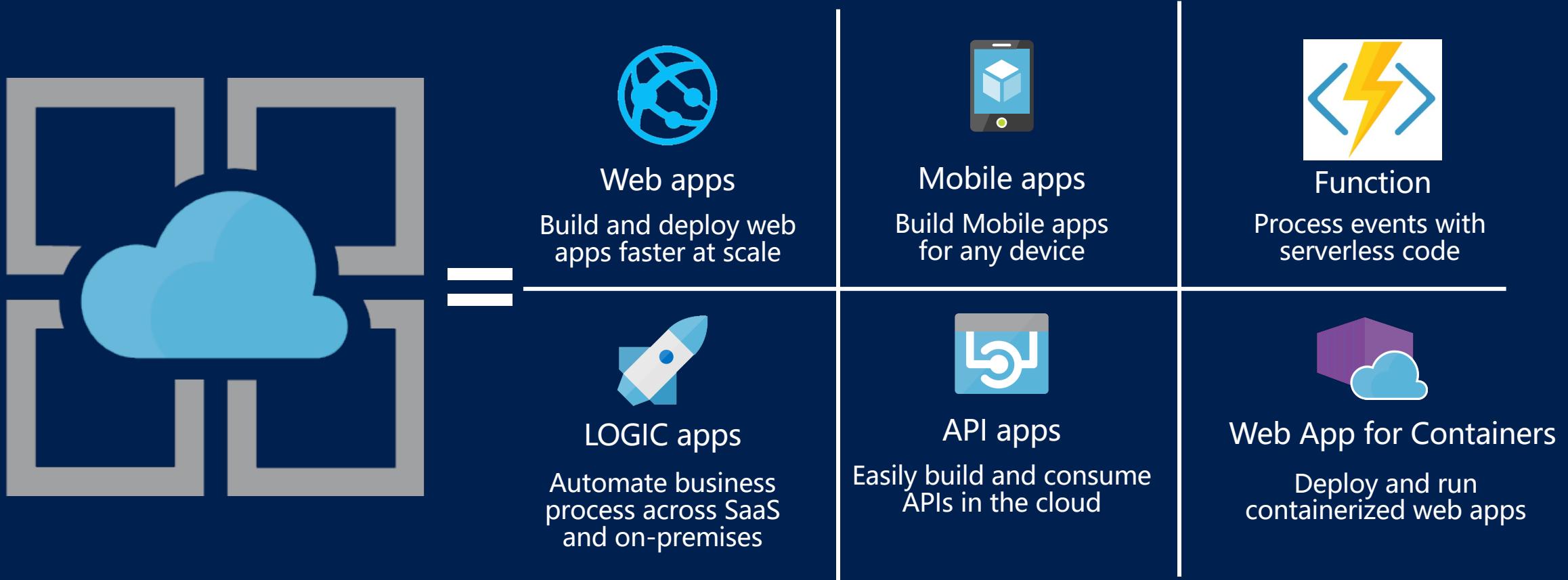
Azure App Service

Microsoft 用於快速創建功能強大的 Web 和移動應用程序的解決方案。

獨一無二的雲服務，可統一您快速輕鬆地創建可在任何平台或任何設備上運行的企業級應用程序所需的一切



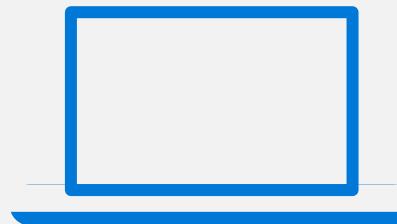
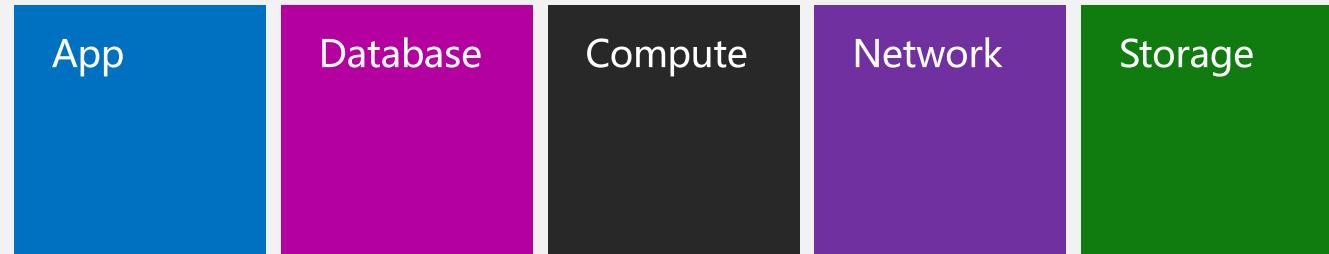
Azure App Service – 多功能集成



Azure Resource Manager (ARM)

聲明式模板驅動方法

- 用於協調跨不同資源的部署
- 將配置映射到資源設置
- 中央端點：
 - 為雲服務場景部署模型
 - 管理雲服務的生命週期和全局屬性



Azure Resource Manager



Describe 描述

Where

Resource inventory

What

Component relationships

How

Tags + links + groups

Provision 條款

Where

Across Regions

What

Across resources

How

In service and in guest



Control 控制

Who

Access control

What

Changes

How

RBAC, subscriptions and locks

Azure Resource Manager

應用程序生命週期容器

部署和管理您認為合適的應用程序（或資源）

用於供應和配置的聲明式解決方案

通過 JSON 模板簡化應用程序的多個實例化部署

一致的管理

來自 Portal、PowerShell、Xplat CLI 和 Visual Studio 的相同部署和管理體驗

應用程序資源管理器功能

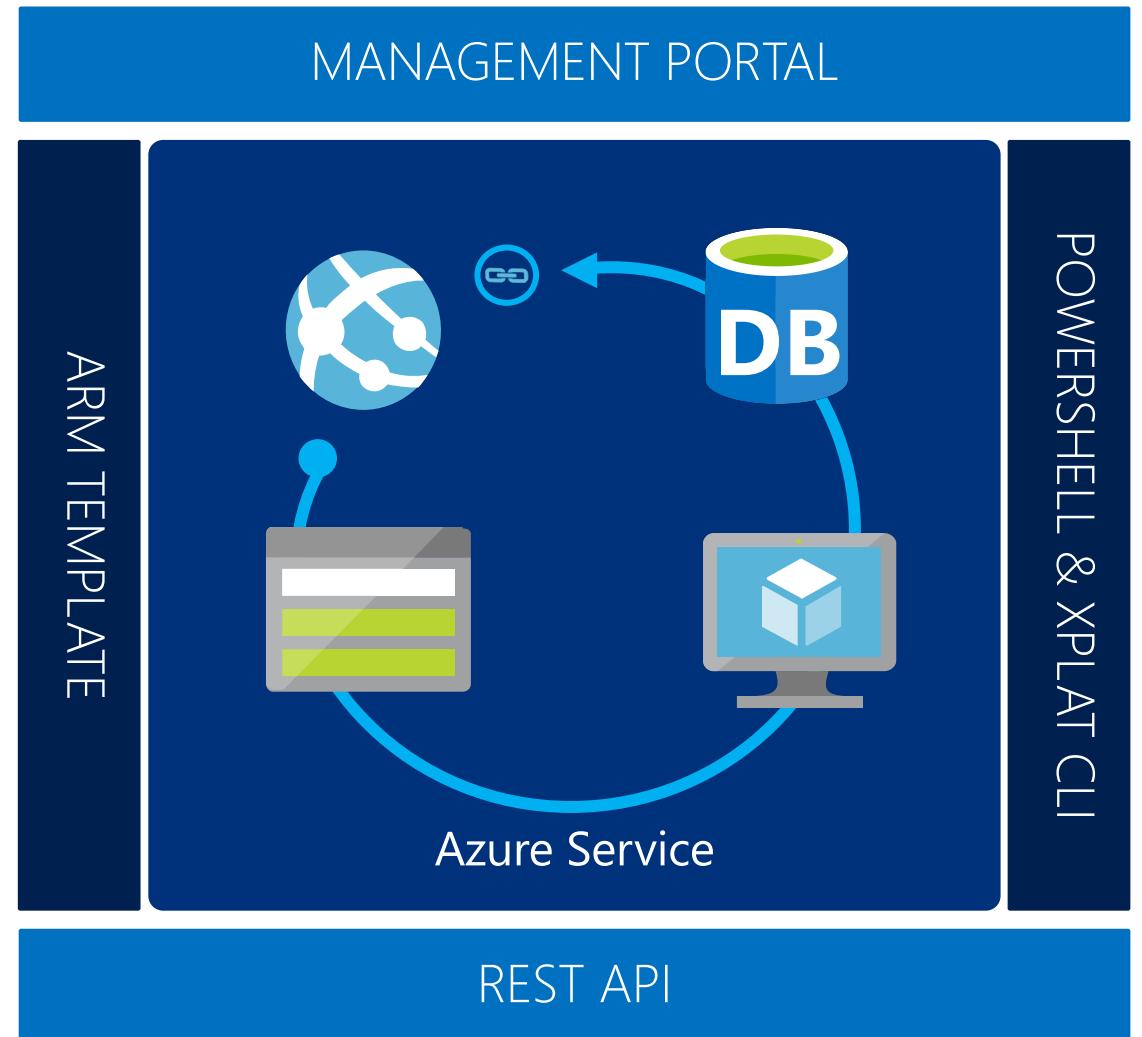
集中審計、簡單的標記和分組以及 RBAC

Management using ARM

Management and provisioning of ARM resources can be performed using

- REST API
- Management Portal
- PowerShell
- Xplat CLI
- ARM Templates

Each mechanism uses the ARM API



Azure Resource Manager (ARM) tools

Available tools and technologies

- Command Line (CLI)
- JSON-based templates
- IDE: includes Visual Studio and Eclipse
- Scripting Support: PowerShell and Python

Azure Resource Providers

A resource provider is a service that supplies the resources you can deploy and manage through Resource Manager .

Example:

If you want to deploy an Azure Key Vault for storing keys and secrets, you will work with the **Microsoft.KeyVault** resource provider.

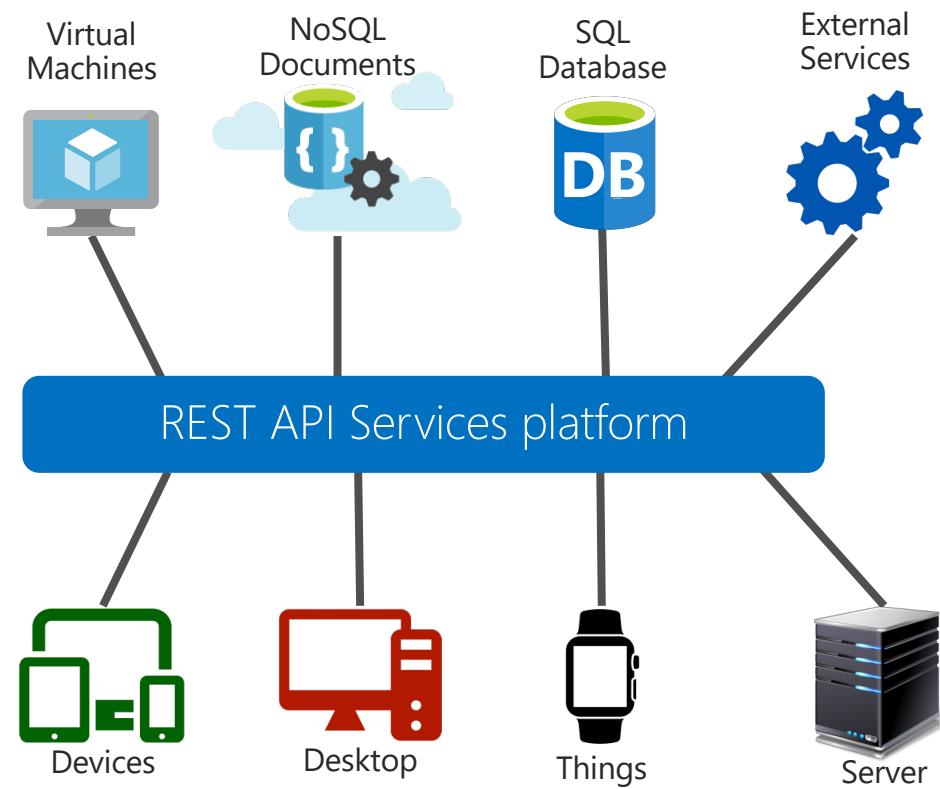
To deploy and manager your infrastructure, you will need to know details about the resource providers:

- What resource types it offers
- Version numbers of the REST API operations
- Operations the resource provider supports
- Scheme to use when setting the values of the resource type to create

Azure REST API

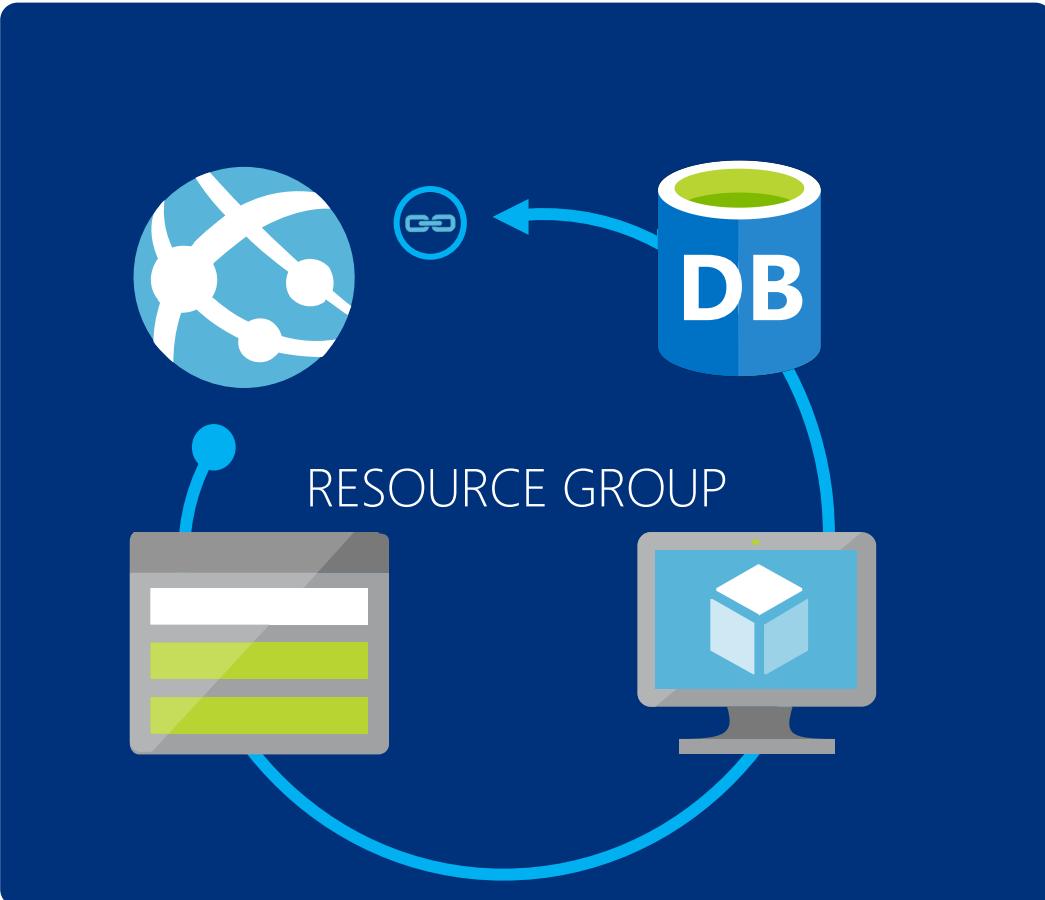
ARM uses REST API to manage resources

- Linked Resources
- Management Locks
- Policy Assignments
- Policy Definitions
- Resources
- Resource Groups
- Resource Providers
- Role-based Access Control
- Subscriptions
- Tags
- Template Deployments
- Tenants



You are responsible for ensuring that these requests are secure (i.e. use service principals)

Resource groups



1. Tightly coupled containers of multiple resources of similar or different types
2. Every resource **must** exist in **one and only one** resource group
3. Resource groups can span regions

Resource group lifecycle

Question:

Should these resources be placed in the same resource group or in separate ones?

Hint:

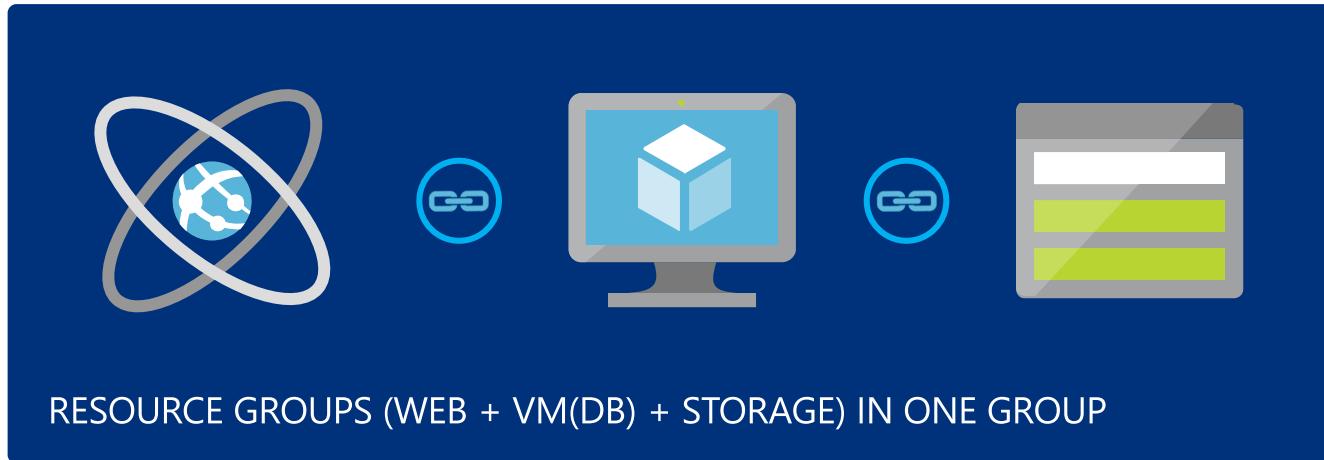
Do they share a common lifecycle and method of management

Answer:

Up to you, or....

In Visual Studio:

- For Devs → it's an app
- For DevOps → it's an environment



OR



Organizational concepts

Resource groups

- Tightly couple to a container of resources
- Follows RBAC rules
- 1 resource group at a time
- Scope for billing controls

Resource tags

- Loosely coupled user or system-defined categorization
- Arbitrary boundaries
- 15 tags to use as needed
- Defined by platform and user

資源組和標籤是定義應用程序的構建塊

Azure App Service

App Service Plan

An **App Service plan** defines a set of compute resources for a web app to run. One or more apps can be configured to run in the same **App Service plan**.

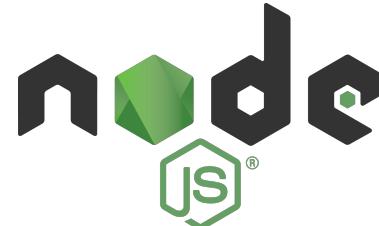
Each App Service plan defines:

- Region (West US, East US, etc.)
- Number of VM instances
- Size of VM instances (**Small, Medium, Large, Extra-Large**)
- Pricing tier (**Free, Shared, Basic, Standard, Premium, PremiumV2, Isolated, Consumption**)



Supported platforms on Azure App Service

Mainstream support:



Community support:



Deployment Options

The basics



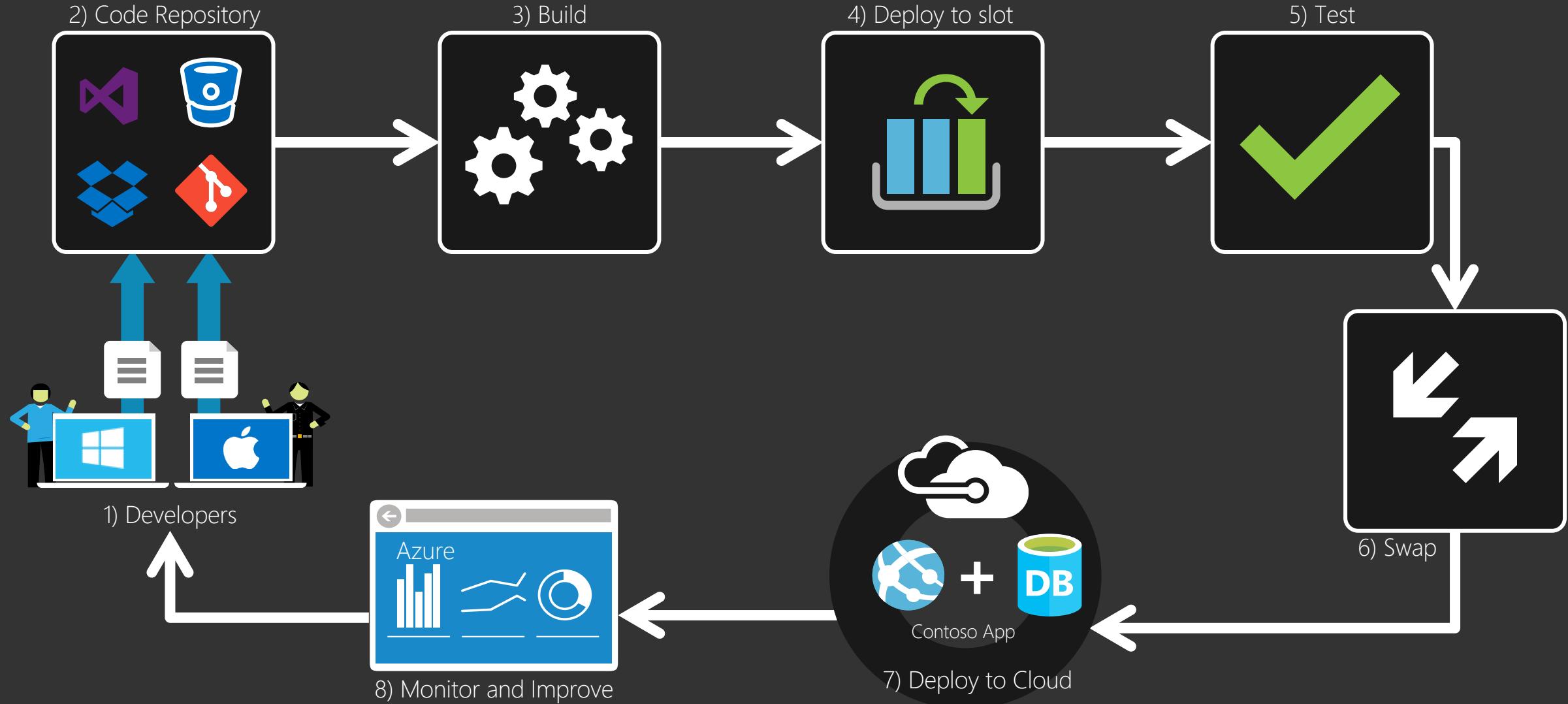
Source control/continuous deployment integration



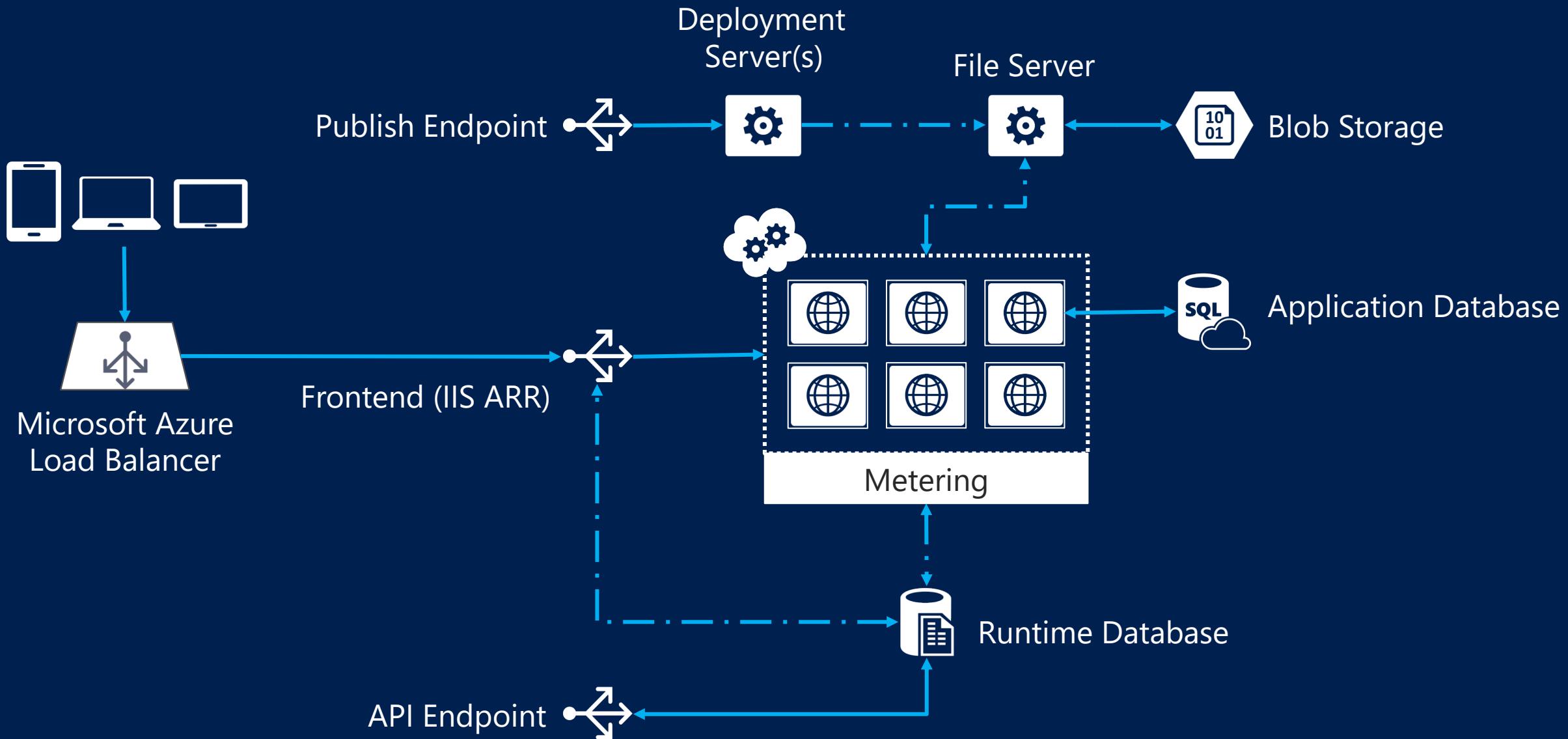
Cool



Continuous Integration cycle



App Service Web App Architecture



Deployment Slots

Staging



Staging



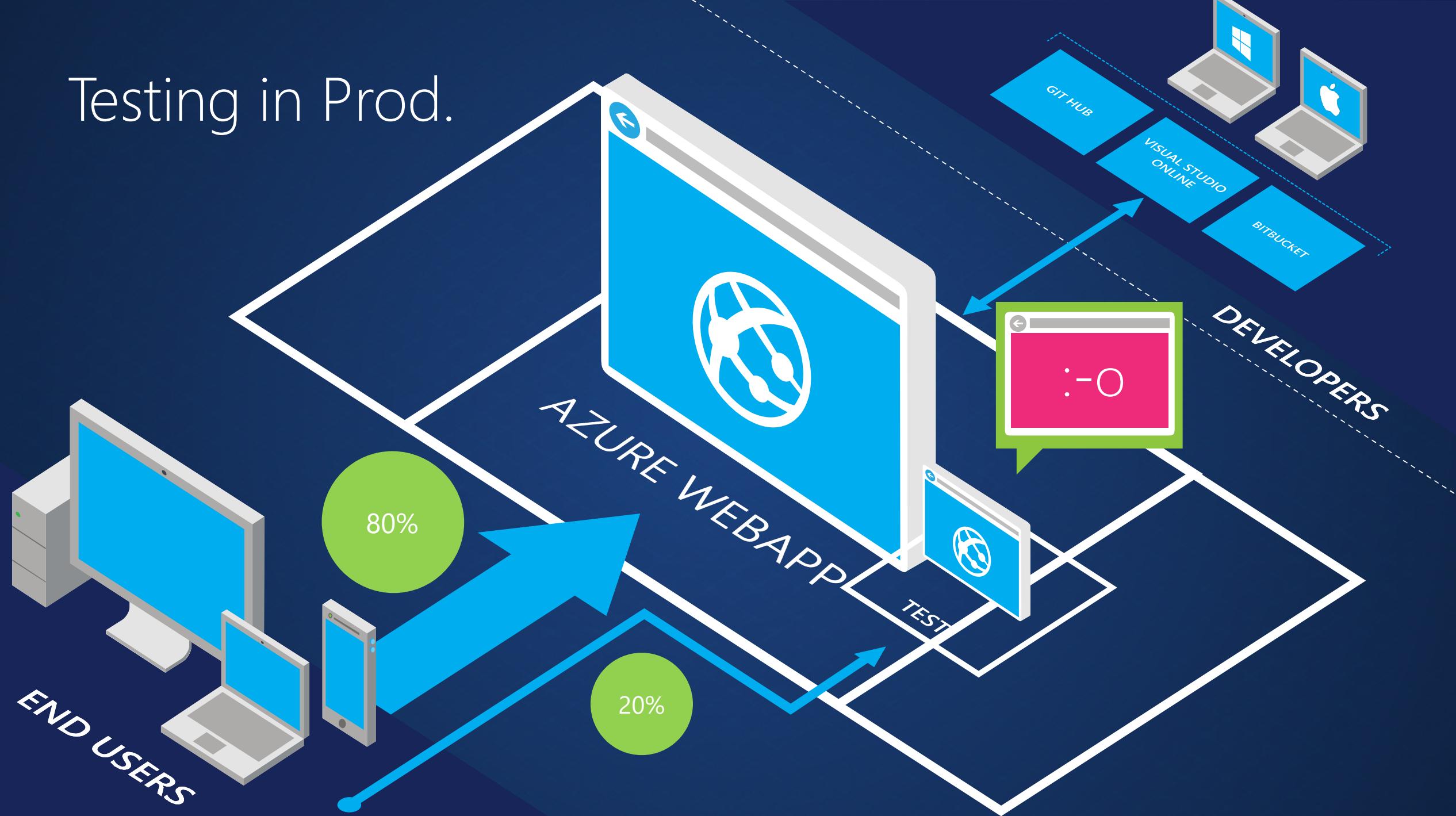
Staging



Staging



Testing in Prod.



AutoScale

AutoScale



AutoScale

Azure WebApp

END USERS

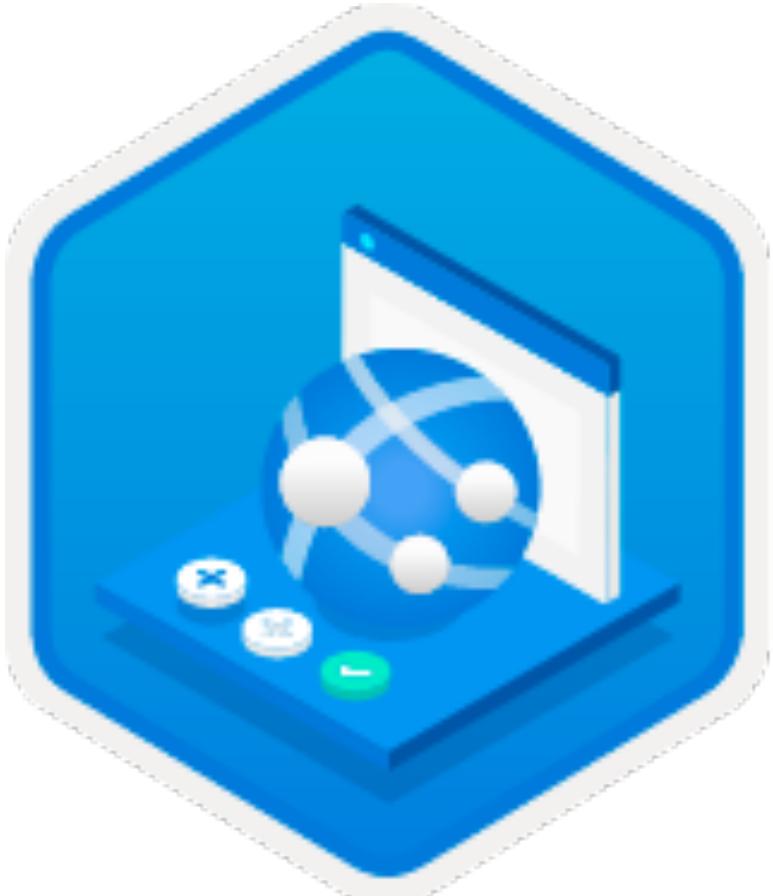
DEVELOPERS

GITHUB

VISUAL STUDIO
ONLINE

BITBUCKET

MS Learn Module 推薦



<https://aka.ms/AzureHKStudy001>

Reactor

Thank You!

Q&A