Microsoft

Reactor

MS Learn 學堂
Azure Container Apps 學習

Friday 1st April | 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

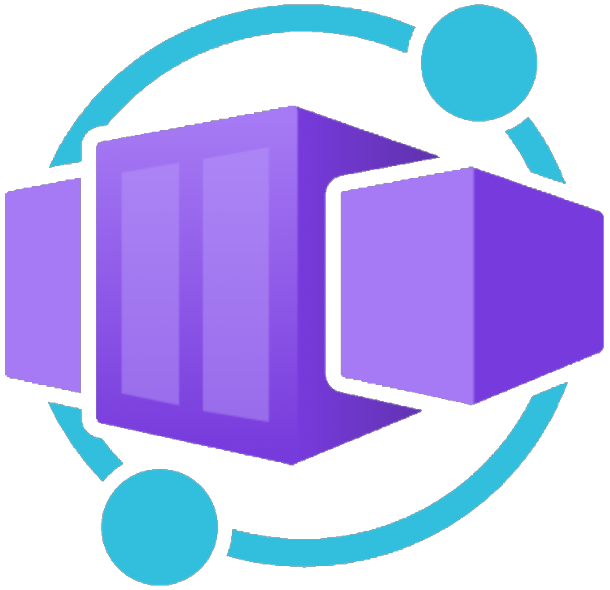
Azure 資源索取

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

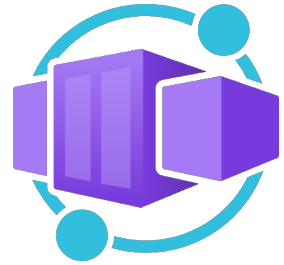
Free Azure Student Plan https://aka.ms/StudentGetAzure

Azure Container Apps

用於構建現代應用程序和微服務的新無服務器容器平台



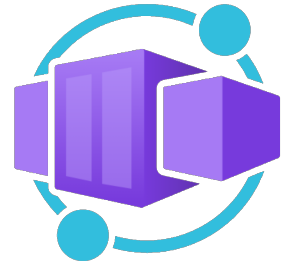
Azure Container Apps (public preview)



“Azure Container Apps 可以執行打包在任何容器中的應用程序代碼，並且不受運行時或編程模型的影響.”

- 享受運行容器帶來的好處，同時拋開管理雲基礎設施和復雜容器編排器的顧慮。
- 無服務器（支持縮放到零）
- 擴展 HTTP 請求、事件或運行始終在線的後台作業
- 入口和服務到服務通信的自動加密
- 建立在 AKS、KEDA、Dapr 和 Envoy 的基礎上

Azure Container Apps



部署容器化應用程式，而不需要管理複雜的基礎結構。使用您慣用的程式設計語言或架構撰寫程式碼，並在分散式應用程式執行階段 (Dapr) 的完整支援下建置微服務。根據 HTTP 流量或 Kubernetes 事件驅動自動調整 (KEDA) 提供的事件，以動態方式縮放。

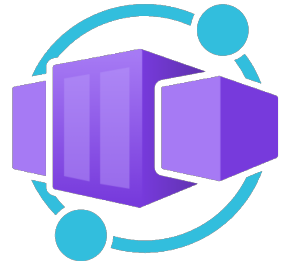
支援各種應用程式類型，包括 HTTP API、微服務、事件處理和背景工作

使用自選語言、架構或 SDK 撰寫程式碼的彈性

以 HTTP 流量或事件觸發程序為基礎的可靠自動縮放功能

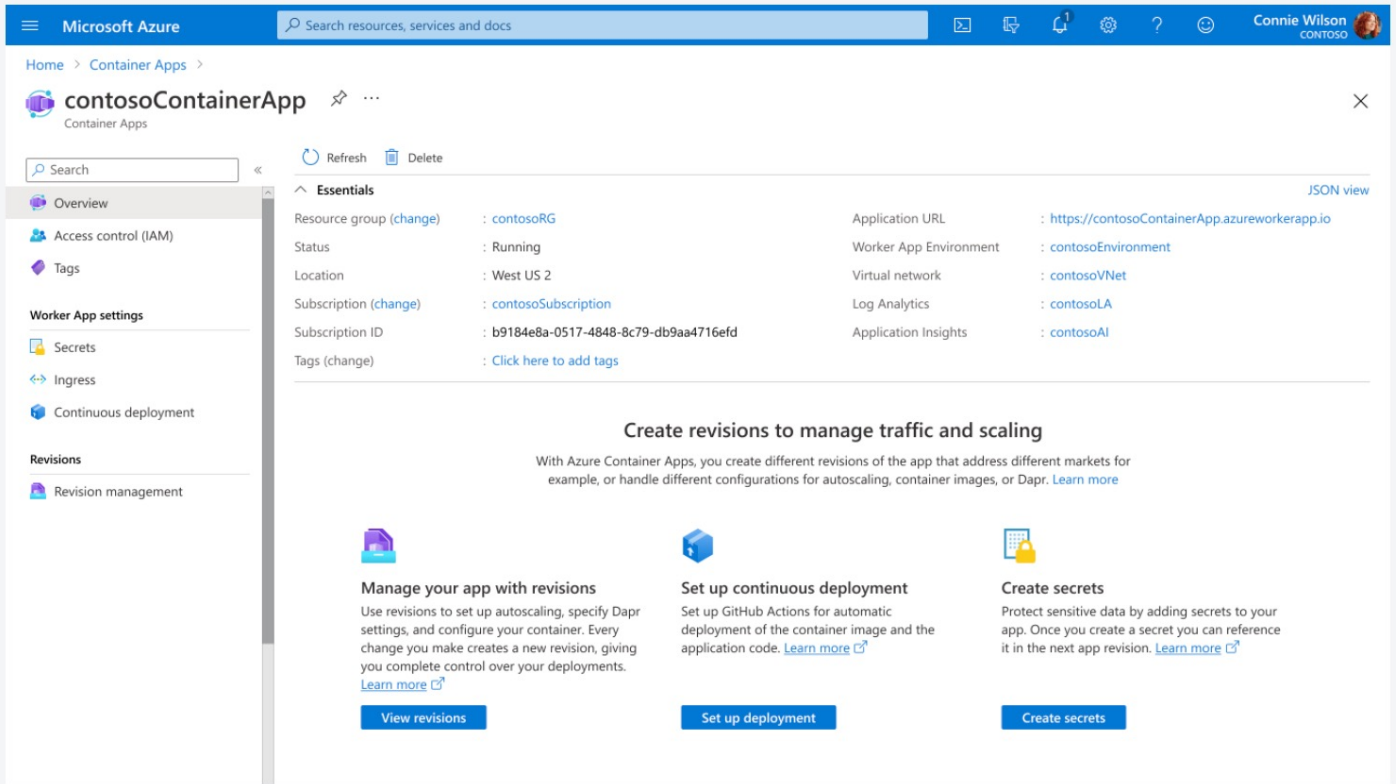
執行現代化應用程式生命週期工作的簡單設定

Azure Container Apps

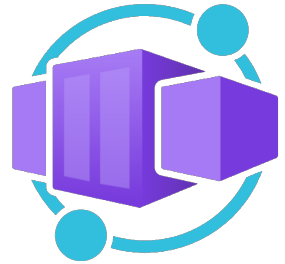


以彈性的無伺服器容器進行縮放

透過為微服務應用程式打造的無伺服器容器服務獲得您需要的彈性，以及可靠的自動調整功能，同時免除管理複雜基礎結構的額外負荷。執行容器並因應 HTTP 流量縮放，或依據 KEDA 持續增加支援的縮放觸發程序清單，包括 Azure 事件中樞、Apache Kafka、RabbitMQ Queue、MongoDB、MySQL 和 PostgreSQL。縮放為零，並只為您使用的部分依秒數付費。



Azure Container Apps



Select any container image using any language or framework

Choose vCPU cores, memory, and scale settings based on events or HTTP requests

Enable service-to-service communication, configure ingress, and event sources

Create and deploy your application

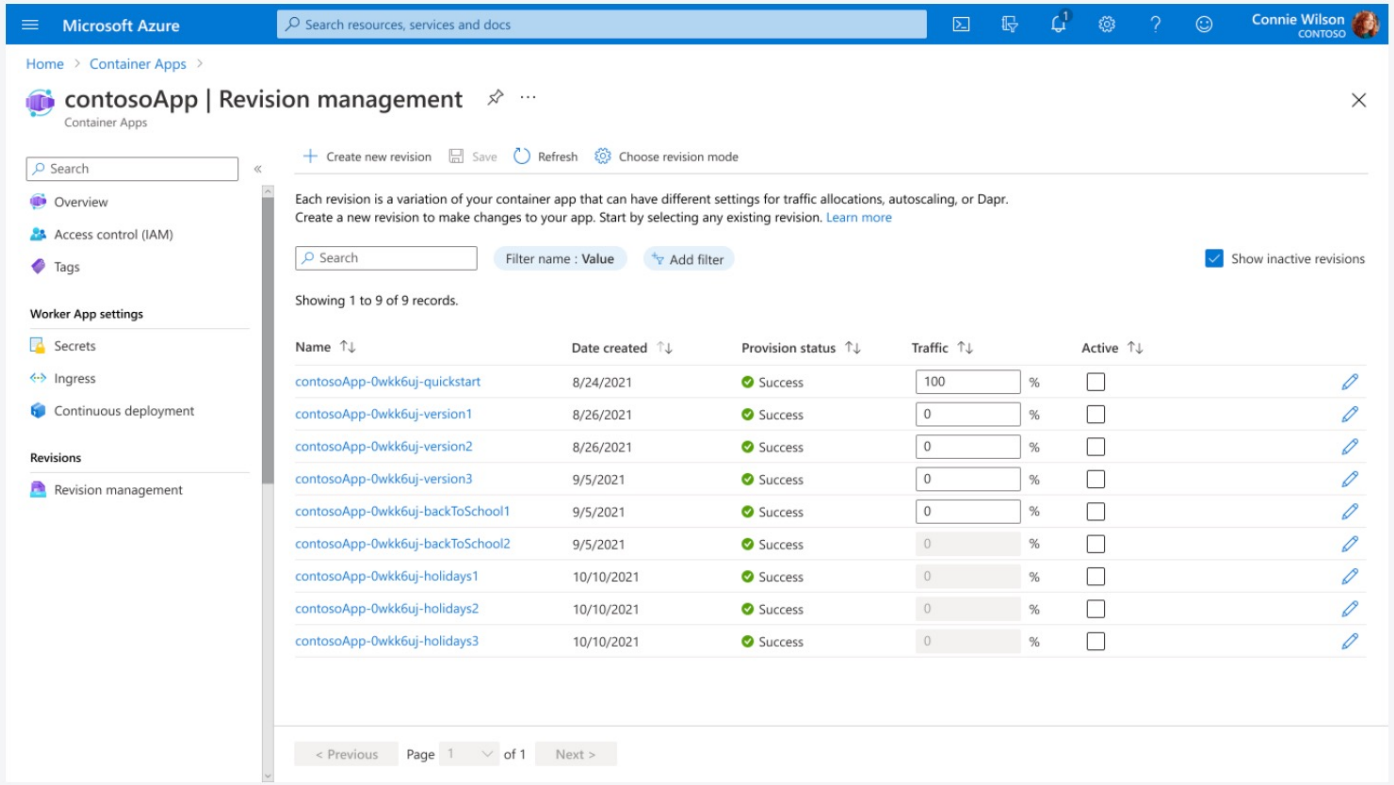
加速開發人員生產力

使用容器建置微服務、API、事件處理背景工作和背景作業。以您慣用的語言撰寫程式碼，並利用內建的分散式應用程式執行階段 (Dapr) 整合來加速開發，以簡化事件處理、發行/訂閱和服務叫用等一般工作。使用 GitHub Actions 設定程式碼到雲端的管線。

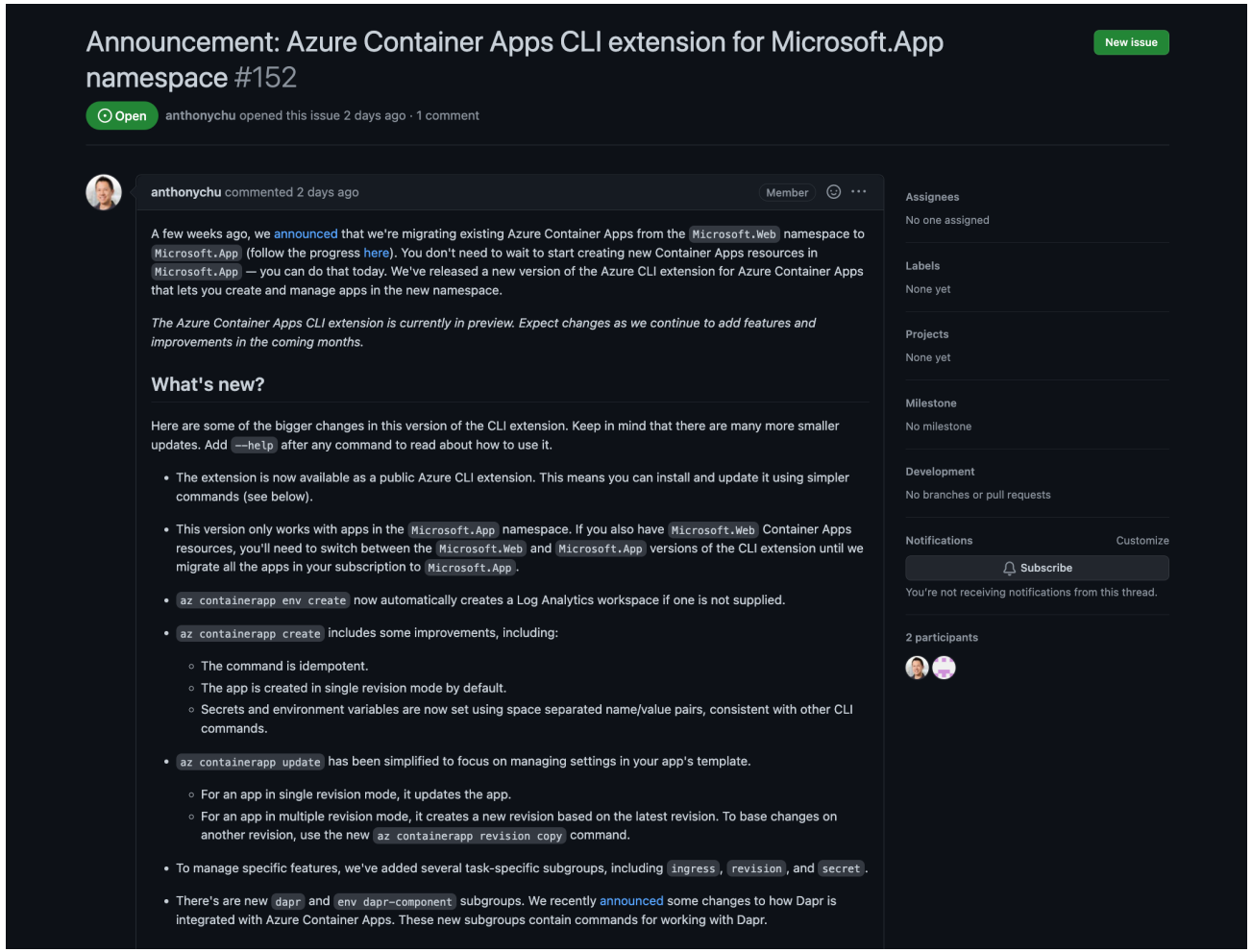
Azure Container Apps

以開放原始碼為基礎建置現代化應用程式

以 Kubernetes 為基礎，利用開放標準建立現代化應用程式。執行應用程式生命週期工作，例如應用程式升級、流量轉移及具有設定清楚直接的版本控制。仰賴內建的服務探索選項來進行微服務通訊、以 Envoy 為基礎的完全受控 HTTP 和 HTTP/2 輸入端點，以及整合式負載平衡、記錄和監視功能。



Azure CLI – Container Apps Extensions



Announcement: Azure Container Apps CLI extension for Microsoft.App namespace #152

New issue

Open anthonychu opened this issue 2 days ago · 1 comment

anthonychu commented 2 days ago — Member

A few weeks ago, we announced that we're migrating existing Azure Container Apps from the `Microsoft.Web` namespace to `Microsoft.App` (follow the progress here). You don't need to wait to start creating new Container Apps resources in `Microsoft.App` — you can do that today. We've released a new version of the Azure CLI extension for Azure Container Apps that lets you create and manage apps in the new namespace.

The Azure Container Apps CLI extension is currently in preview. Expect changes as we continue to add features and improvements in the coming months.

What's new?

Here are some of the bigger changes in this version of the CLI extension. Keep in mind that there are many more smaller updates. Add `--help` after any command to read about how to use it.

- The extension is now available as a public Azure CLI extension. This means you can install and update it using simpler commands (see below).
- This version only works with apps in the `Microsoft.App` namespace. If you also have `Microsoft.Web` Container Apps resources, you'll need to switch between the `Microsoft.Web` and `Microsoft.App` versions of the CLI extension until we migrate all the apps in your subscription to `Microsoft.App`.
- `az containerapp env create` now automatically creates a Log Analytics workspace if one is not supplied.
- `az containerapp create` includes some improvements, including:
  - The command is idempotent.
  - The app is created in single revision mode by default.
  - Secrets and environment variables are now set using space separated name/value pairs, consistent with other CLI commands.
- `az containerapp update` has been simplified to focus on managing settings in your app's template.
  - For an app in single revision mode, it updates the app.
  - For an app in multiple revision mode, it creates a new revision based on the latest revision. To base changes on another revision, use the new `az containerapp revision copy` command.
- To manage specific features, we've added several task-specific subgroups, including `ingress`, `revision`, and `secret`.
- There's are new `dapr` and `env dapr-component` subgroups. We recently announced some changes to how Dapr is integrated with Azure Container Apps. These new subgroups contain commands for working with Dapr.

Assignees — No one assigned

Labels — None yet

Projects — None yet

Milestone — No milestone

Development — No branches or pull requests

Notifications — Customize

Subscribe

You're not receiving notifications from this thread.

2 participants

az extension add -n containerapp

https://github.com/microsoft/azure-container-apps/issues/152

Azure Container Apps CLI 構建

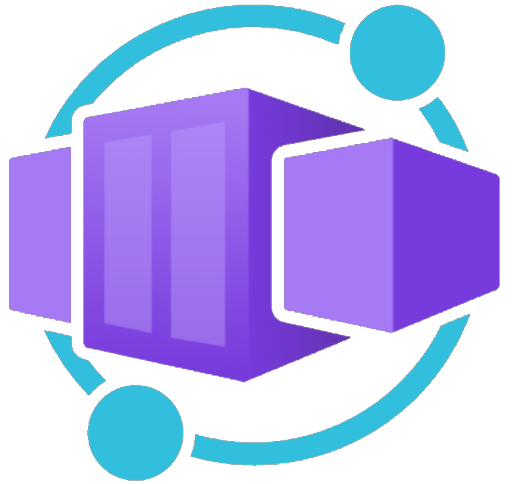
1. az provider register --namespace Microsoft.App

2. RESOURCE_GROUP="my-container-apps"
   LOCATION="canadacentral"
   CONTAINERAPPS_ENVIRONMENT="my-environment"

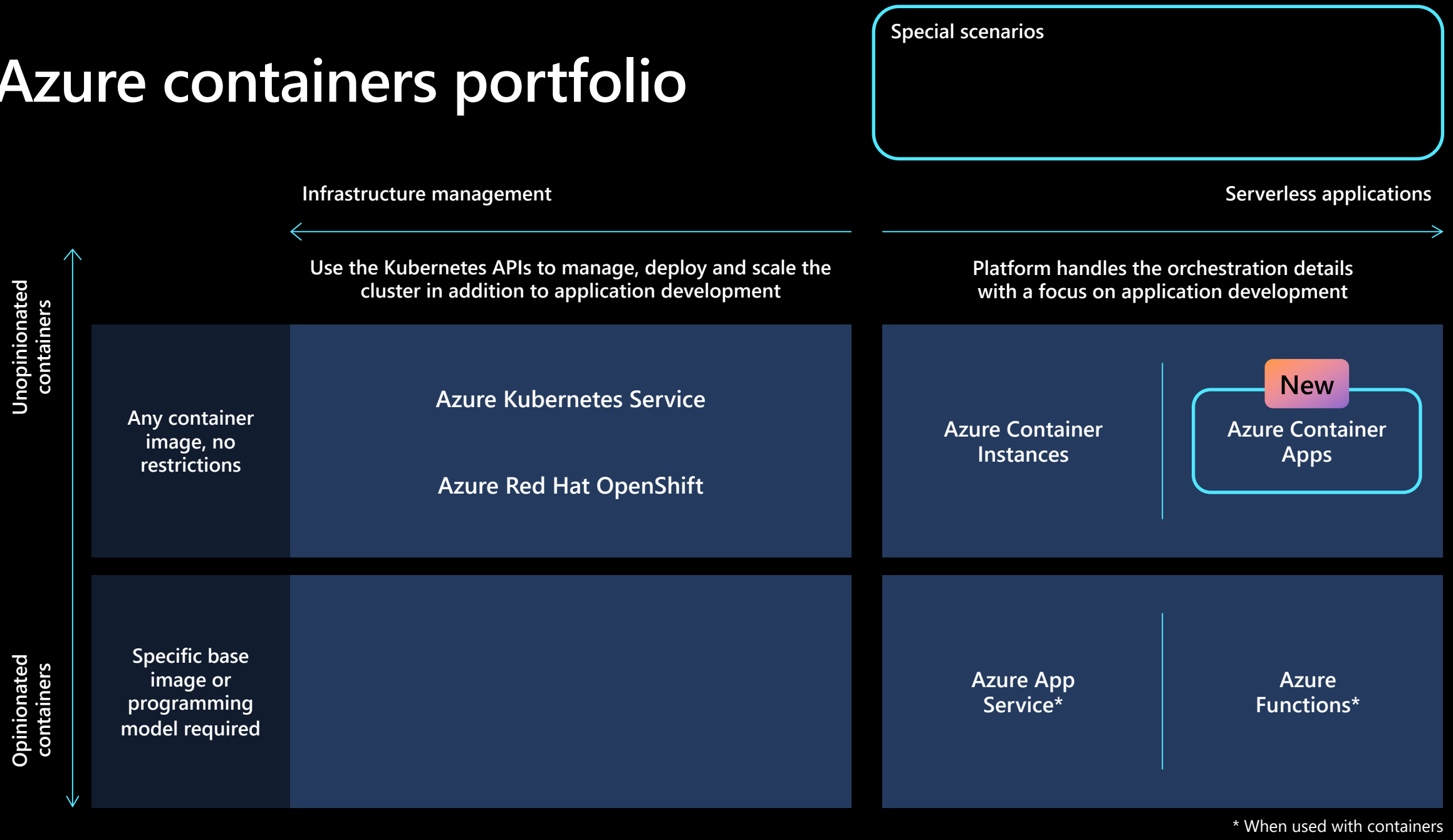
3. az group create \
  --name $RESOURCE_GROUP \
  --location $LOCATION

4. az containerapp env create \
  --name $CONTAINERAPPS_ENVIRONMENT \
  --resource-group $RESOURCE_GROUP \
  --location "$LOCATION"

5. az containerapp create \
   --name my-container-app \
   --resource-group $RESOURCE_GROUP \
   --environment $CONTAINERAPPS_ENVIRONMENT \
   --image mcr.microsoft.com/azuredocs/containerapps-helloworld:latest \
   --target-port 80 \
   --ingress 'external' \
   --query properties.configuration.ingress.fqdn



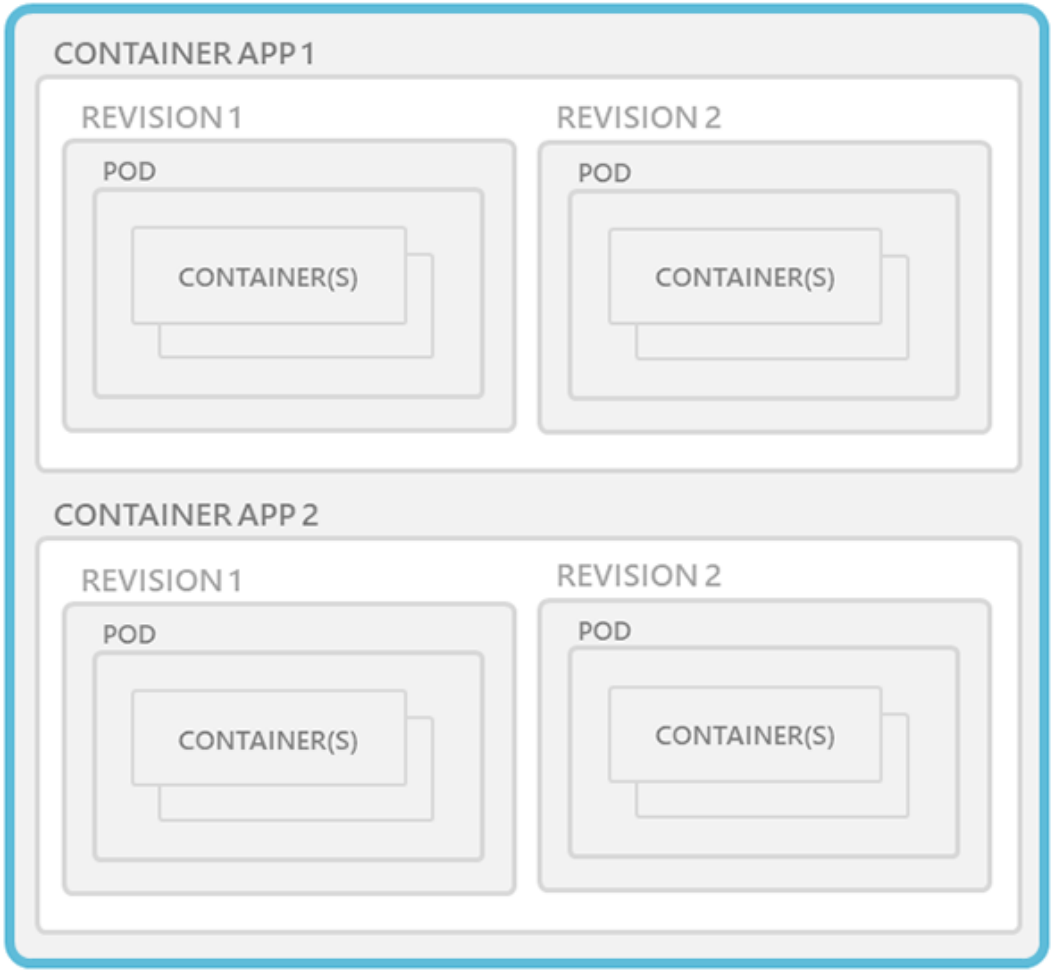
Azure containers portfolio



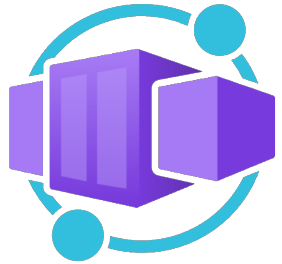
* When used with containers

Azure Container Apps

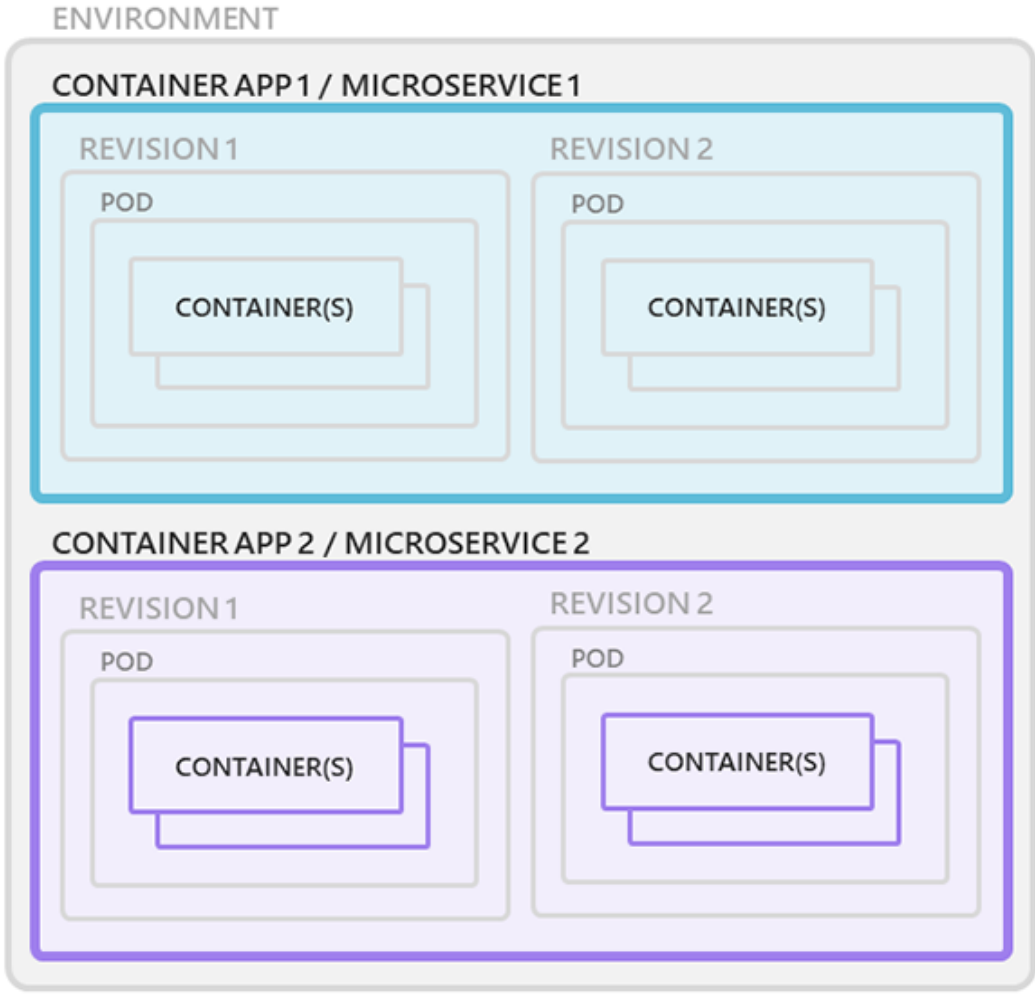
Environments are an isolation boundary around a collection of container apps.



Azure Container Apps

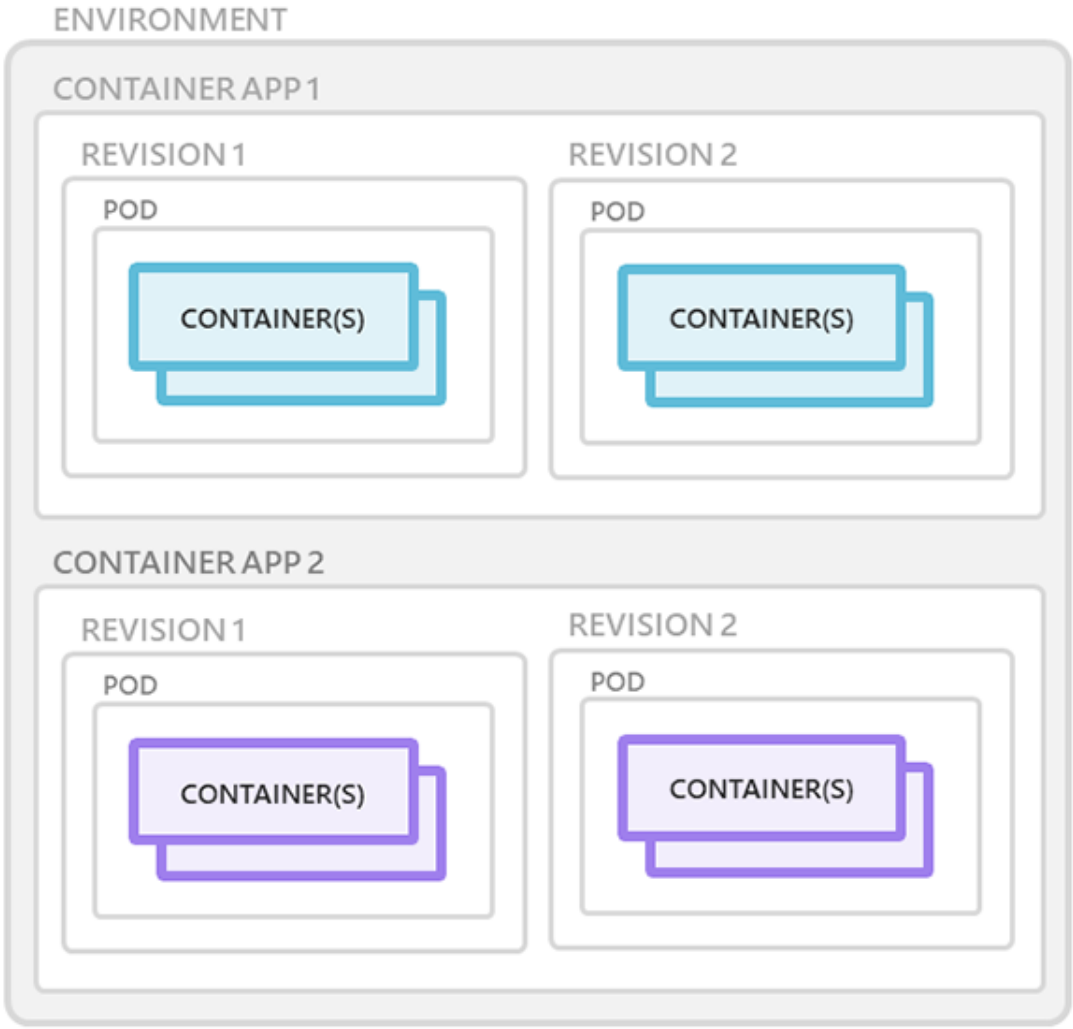


Container apps are deployed as **microservices**.

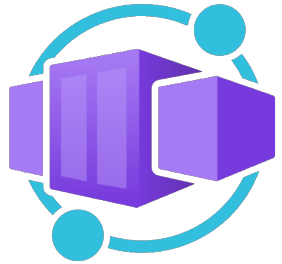


Azure Container Apps

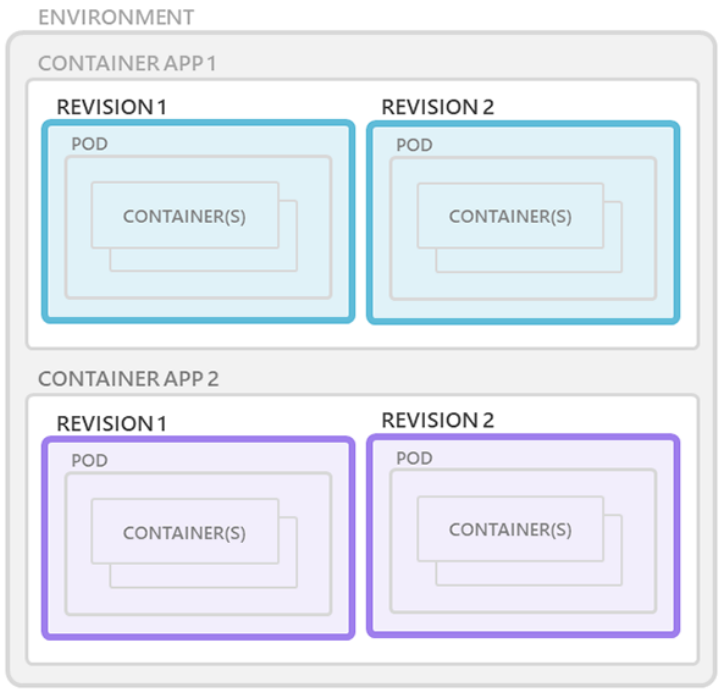
Containers for an Azure Container App are grouped together in pods inside revision snapshots.

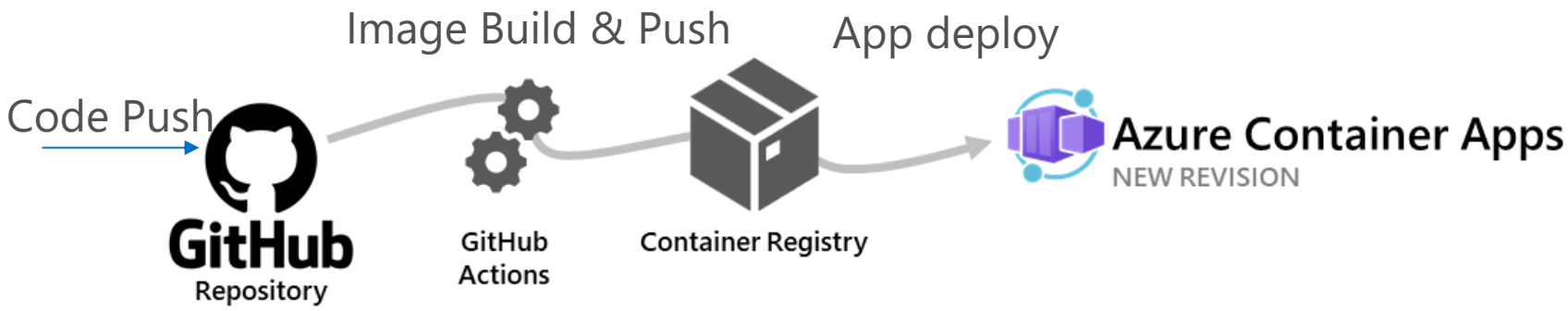


Azure Container Apps



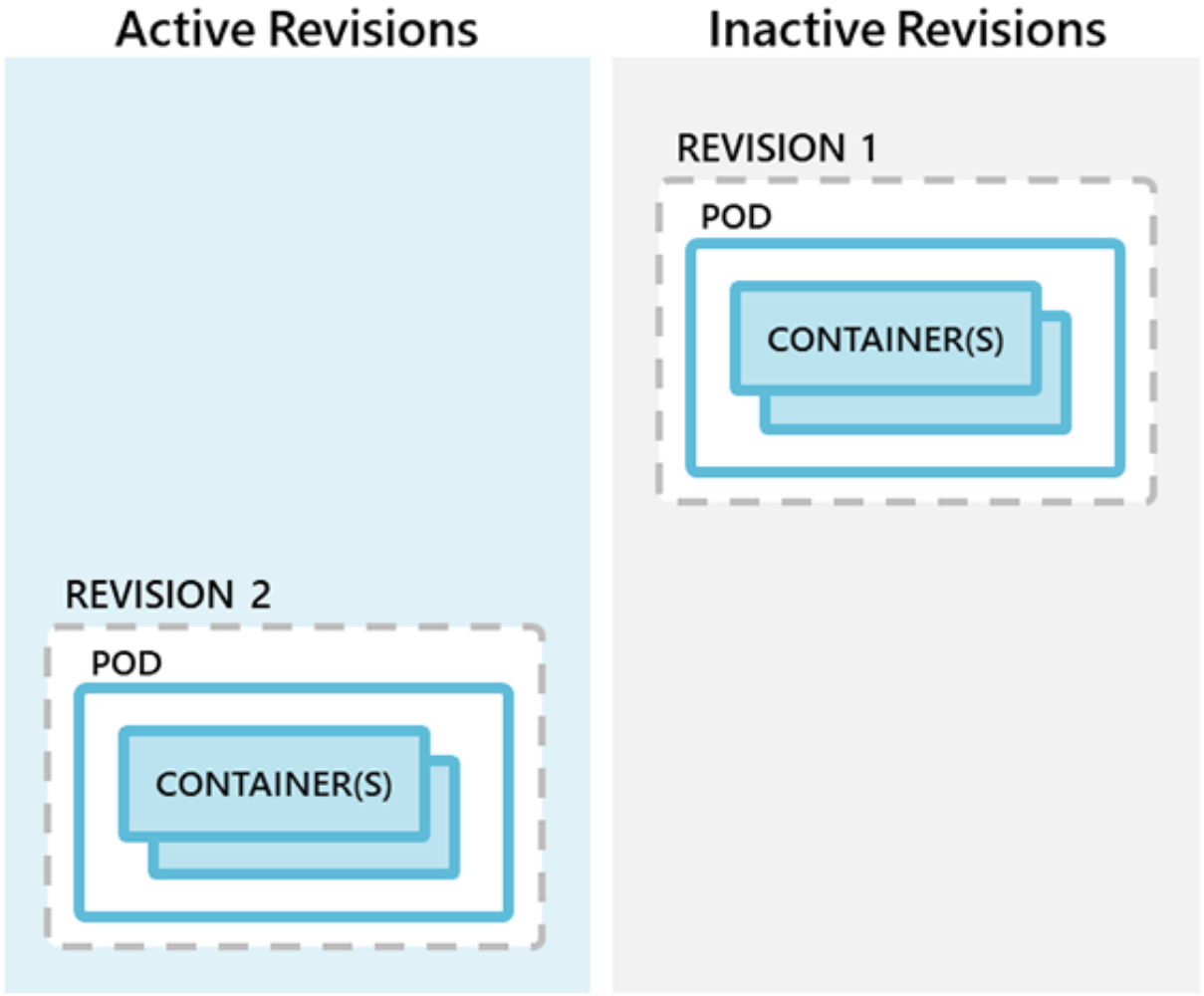
Revisions are immutable snapshots of a container app.



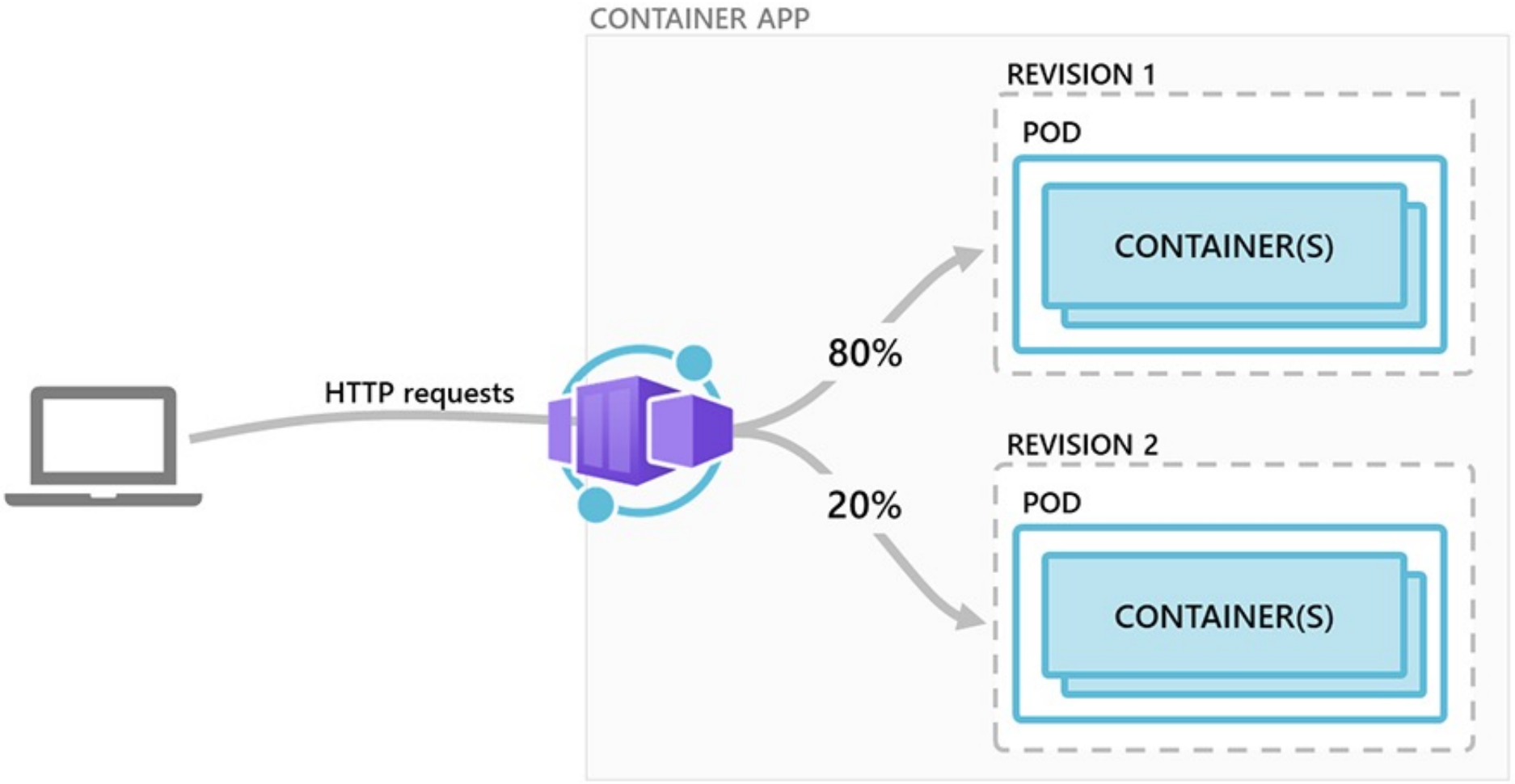


Azure Container Apps

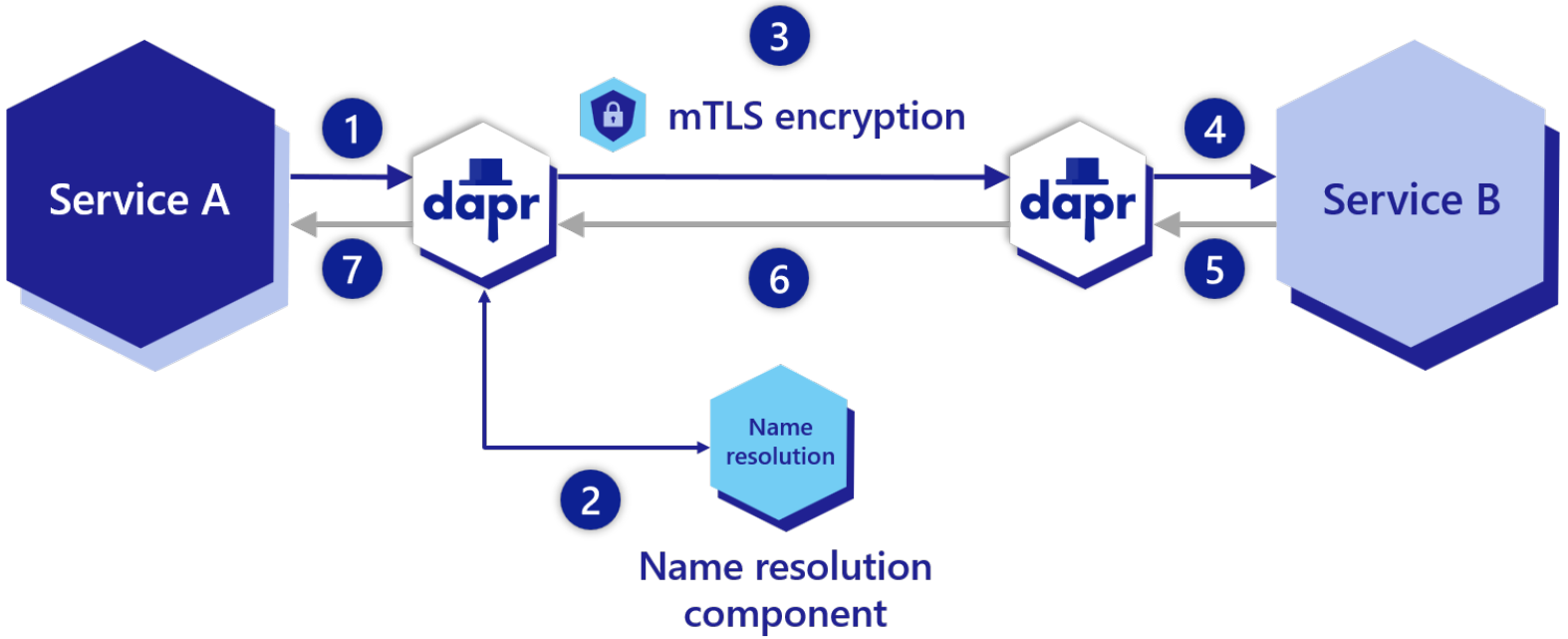
Once a revision is no longer needed, you can **deactivate** individual revisions, or choose to automatically deactivate old revisions.



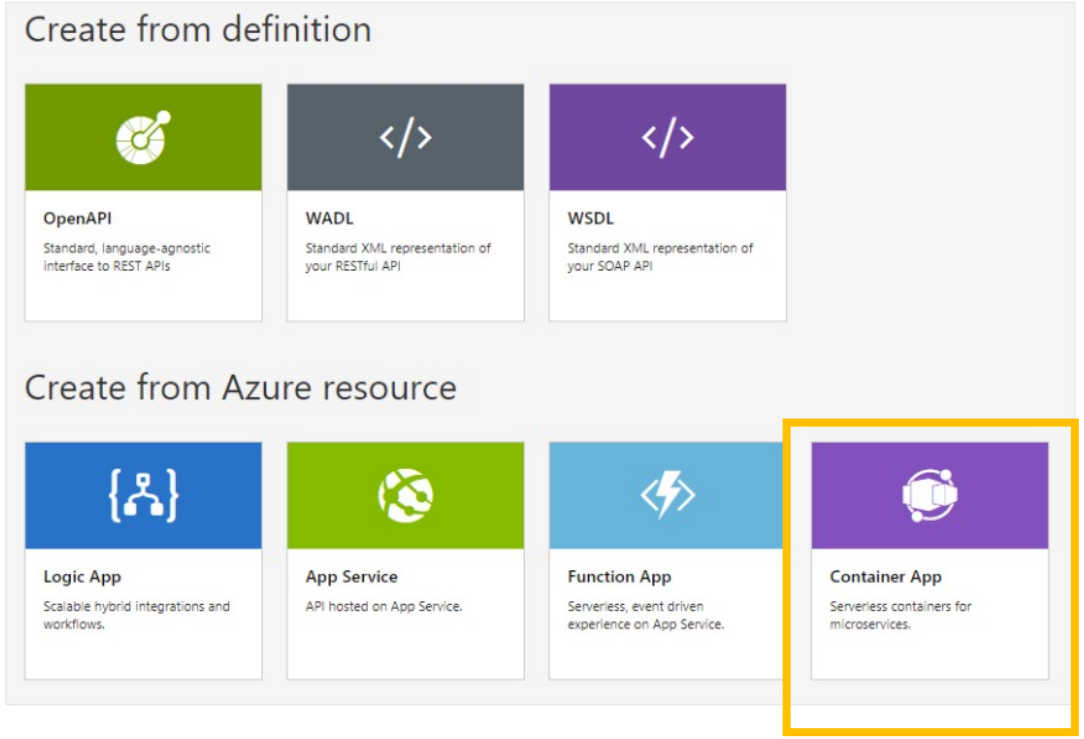
Ingress traffic splitting 入口流量拆分



Dapr integration (mTLS, service discovery, tracing, etc.)



API Management Import



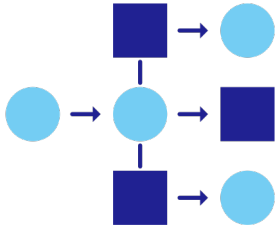
API Management 將在多個位置查找 OpenAPI 規範：

- The Container App configuration
- /openapi.json
- /openapi.yml
- /swagger/v1/swagger.json

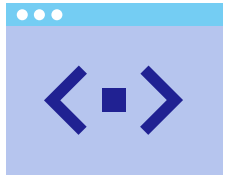
https://docs.microsoft.com/en-us/azure/api-management/import-container-app-with-oas

企業開發者現狀

部署橫向擴展應用程序以提高靈活性、成本和效率

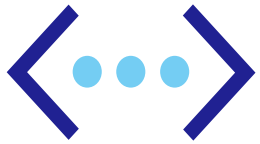


開發與服務交互的彈性、可擴展、基於微服務的應用程序



構建應用程序，而不是基礎設施

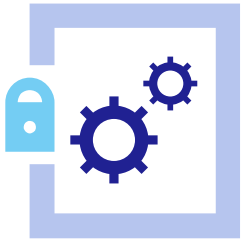
趨向於使用簡單代碼到雲管道的無服務器平台



在開發過程中使用多種語言和框架

是什麼阻礙了微服務的發展?

構建分佈式應用程序的有限工具和運行時

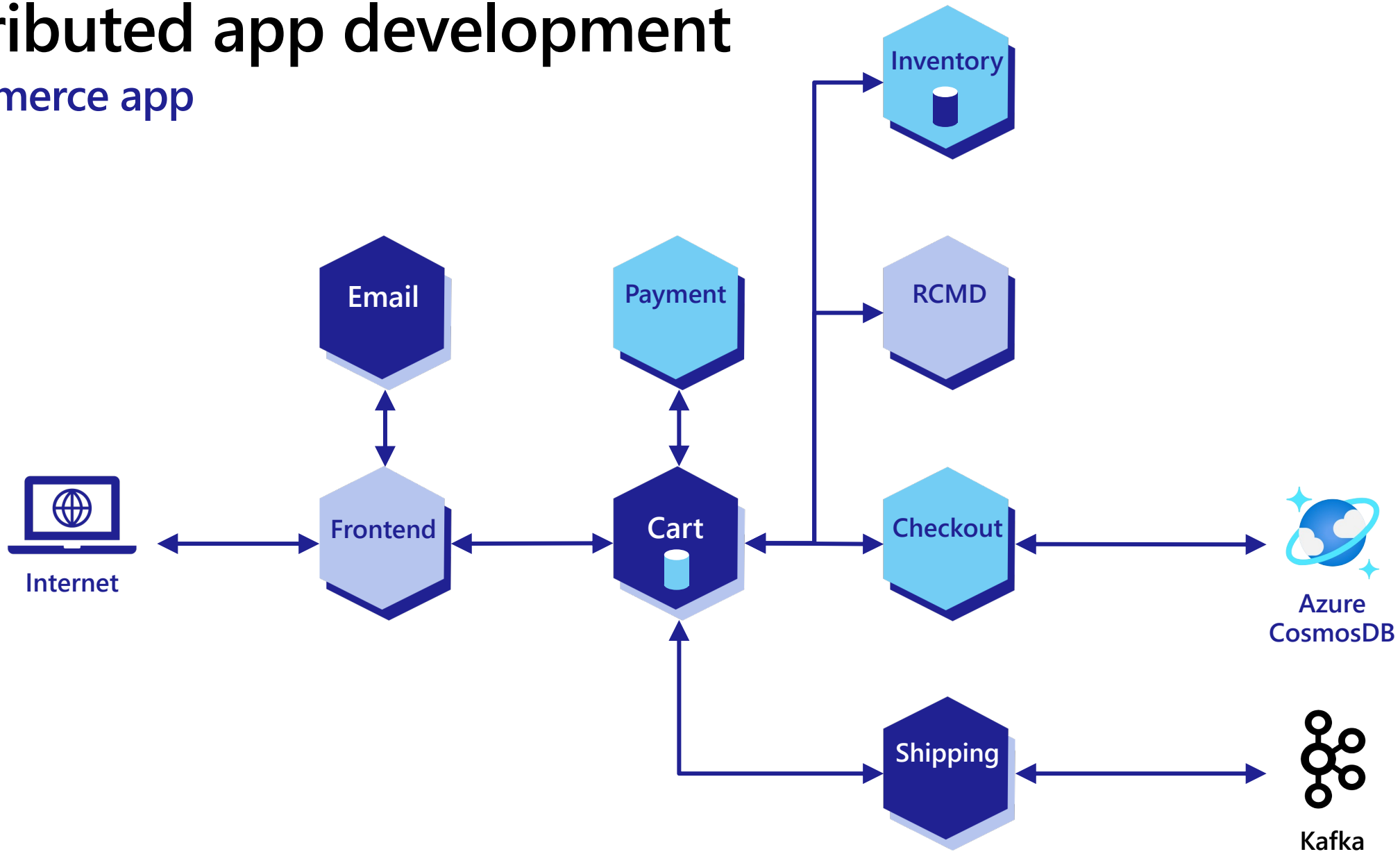


運行時具有有限的語言支持和嚴格控制的功能集

運行時僅針對可移植性有限的特定基礎設施平台

Distributed app development

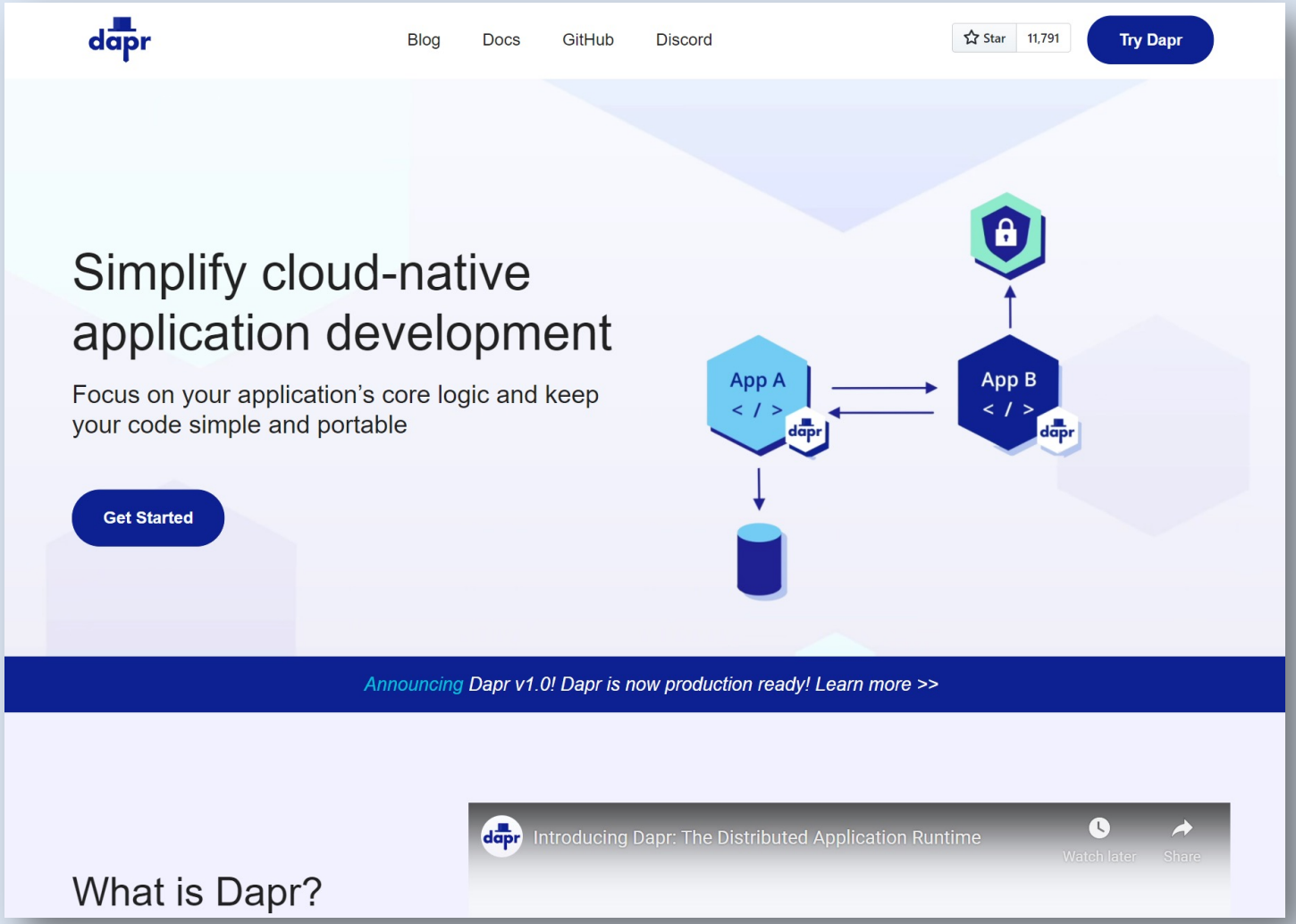
E-commerce app



Distributed Application Runtime

用於構建跨雲和邊緣的分佈式應用程序的便攜式、事件驅動的運行時

dapr.io



Dapr Goals

Best-practices building blocks

Any language or framework

Consistent, portable, open APIs

Adopt standards

Extensible and pluggable components

Platform agnostic cloud + edge

Community driven, vendor neutral

Microservice building blocks

Microservice building blocks

Any cloud or edge infrastructure

Application code

Microservices written in

Any code or framework...

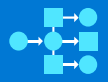






HTTP API

gRPC API


Service-to-service invocation


State management


Publish and subscribe


Resource bindings and triggers


Actors


Observability


Secrets


Extensible

Hosting infrastructure


Microsoft Azure


Azure Arc


aws


Google Cloud


Alibaba Cloud


kubernetes

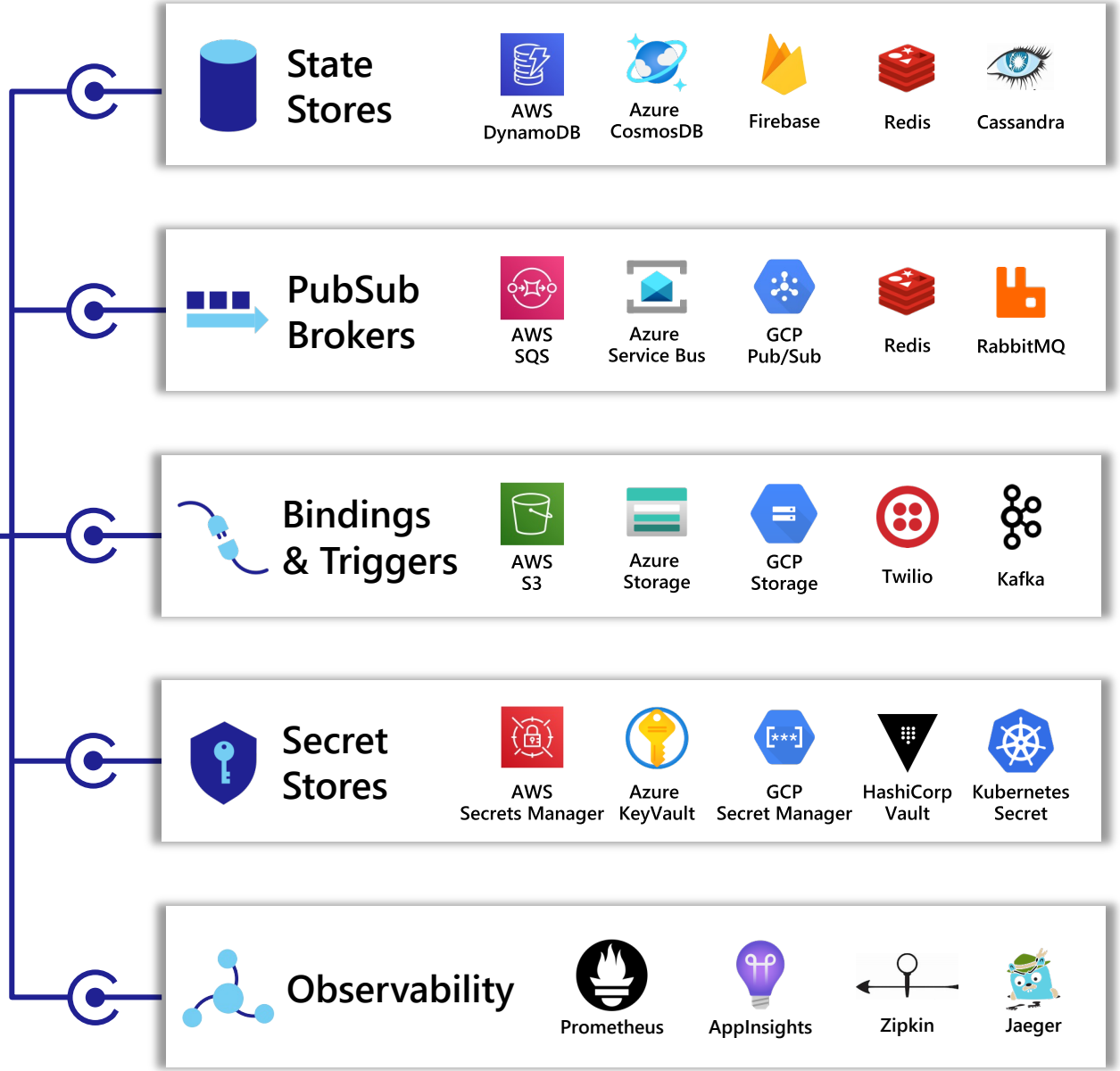

On-Premises

Dapr components

Swappable YAML files with resource connection details

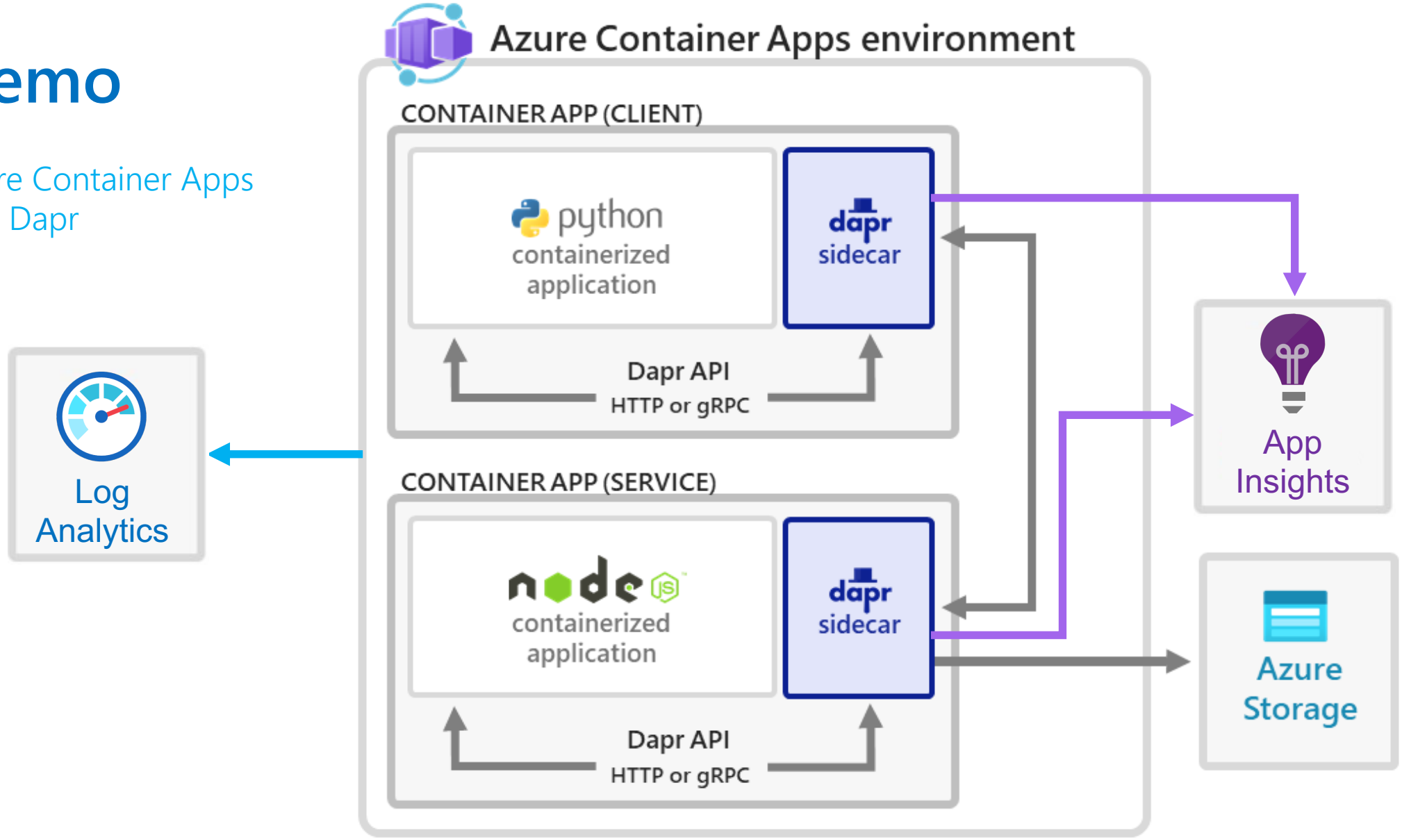
Over 70 components available

Create components for your resource at: github.com/dapr/components-contrib



Demo

Azure Container Apps with Dapr



https://github.com/clarenceb/tutorial-dapr-cli

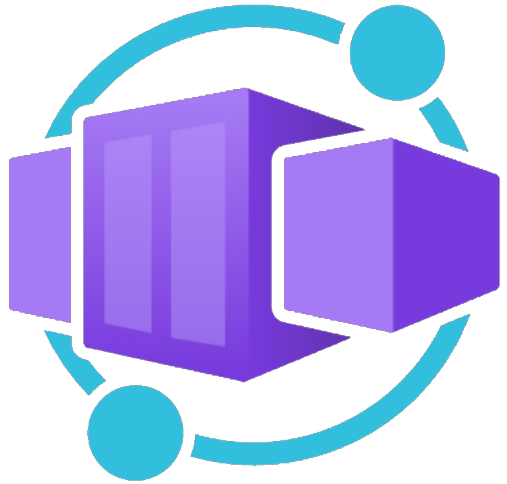
Azure Container Apps CLI 構建

1. az provider register --namespace Microsoft.App

2. RESOURCE_GROUP="my-container-apps"
   LOCATION="canadacentral"
   CONTAINERAPPS_ENVIRONMENT="my-environment"

3. az group create \
   --name $RESOURCE_GROUP \
   --location $LOCATION

4. az containerapp env create \
   --name $CONTAINERAPPS_ENVIRONMENT \
   --resource-group $RESOURCE_GROUP \
   --location "$LOCATION"

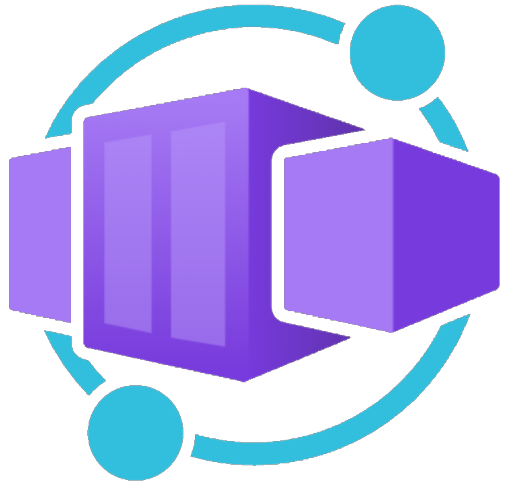


5. STORAGE_ACCOUNT="<storage account name>"

6. STORAGE_ACCOUNT_CONTAINER="mycontainer"

7. az storage account create \
  --name $STORAGE_ACCOUNT \
  --resource-group $RESOURCE_GROUP \
  --location "$LOCATION" \
  --sku Standard_RAGRS \
  --kind StorageV2

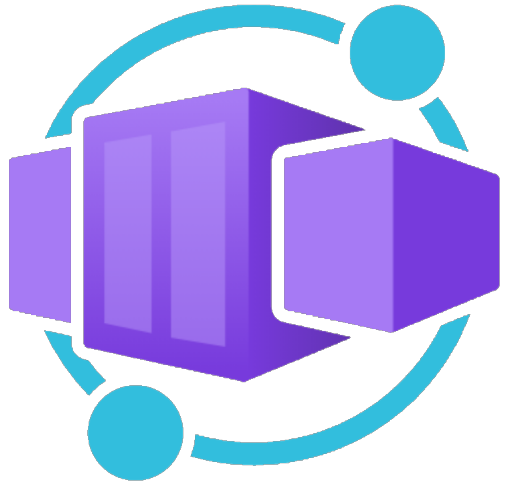
8. STORAGE_ACCOUNT_KEY=`az storage account keys list --resource-group $RESOURCE_GROUP --account-name $STORAGE_ACCOUNT --query '[0].value' --out tsv`



9. statestore.yaml

```
componentType: state.azure.blobstorage version: v1
metadata:
-   name: accountName
    value: "<STORAGE_ACCOUNT>"
-   name: accountKey
    secretRef: account-key
-   name: containerName
    value: mycontainer
secrets:
-   name: account-key
    value: "<STORAGE_ACCOUNT_KEY>"
scopes:
- nodeapp
```

```
az containerapp env dapr-component set \
    --name $CONTAINERAPPS_ENVIRONMENT --resource-group
$RESOURCE_GROUP \
    --dapr-component-name statestore \
    --yaml statestore.yaml
```



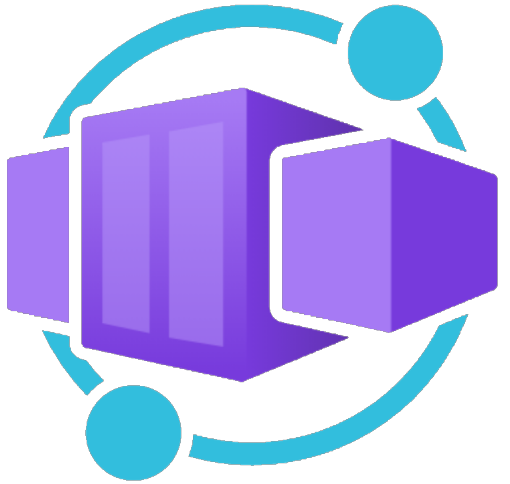
```
echo $STORAGE_ACCOUNT

echo $STORAGE_ACCOUNT_KEY
```

9. Server & Client

```
az containerapp create \
  --name nodeapp \
  --resource-group $RESOURCE_GROUP \
  --environment $CONTAINERAPPS_ENVIRONMENT \
  --image dapriosamples/hello-k8s-node:latest \
  --target-port 3000 \
  --ingress 'external' \
  --min-replicas 1 \
  --max-replicas 1 \
  --enable-dapr \
  --dapr-app-port 3000 \
  --dapr-app-id nodeapp
```

```
az containerapp create \
  --name pythonapp \
  --resource-group $RESOURCE_GROUP \
  --environment $CONTAINERAPPS_ENVIRONMENT \
  --image dapriosamples/hello-k8s-python:latest \
  --min-replicas 1 \
  --max-replicas 1 \
  --enable-dapr \
  --dapr-app-id pythonapp
```



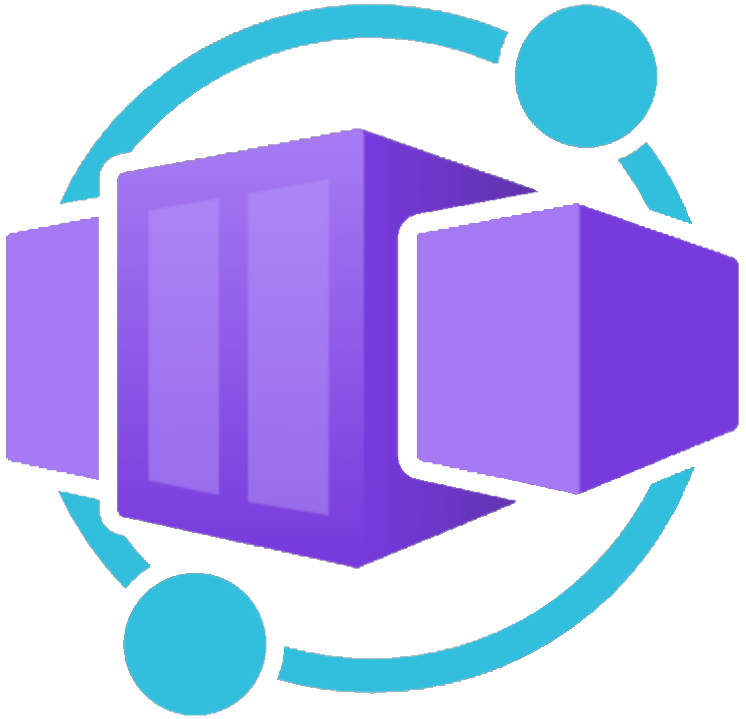
10. 管理相關資源

```
LOG_ANALYTICS_WORKSPACE_CLIENT_ID=`az containerapp env
show --name $CONTAINERAPPS_ENVIRONMENT --resource-
group $RESOURCE_GROUP --query
properties.appLogsConfiguration.logAnalyticsConfiguration.cust
omerId --out tsv`

az monitor log-analytics query \
  --workspace $LOG_ANALYTICS_WORKSPACE_CLIENT_ID \
  --analytics-query "ContainerAppConsoleLogs_CL | where
ContainerAppName_s == 'nodeapp' and (Log_s contains
'persisted' or Log_s contains 'order') | project
ContainerAppName_s, Log_s, TimeGenerated | take 5" \
  --out table
```

```
ContainerAppName_s    Log_s                              TableName      TimeGenerated
--------------------  ---------------------------------  -------------  ----------------------------
nodeapp               Got a new order! Order ID: 60      PrimaryResult  2022-04-01T01:56:49.016Z
nodeapp               Successfully persisted state.      PrimaryResult  2022-04-01T01:56:49.016Z
nodeapp               Got a new order! Order ID: 5       PrimaryResult  2022-04-01T01:55:51.704Z
nodeapp               Successfully persisted state.      PrimaryResult  2022-04-01T01:55:51.704Z
nodeapp               Got a new order! Order ID: 63      PrimaryResult  2022-04-01T01:56:51.696Z
```

MS Learn Module 推薦



https://aka.ms/LearnAzureContainer

Microsoft

Reactor

Thank You!

Q&A