

Azure Container Apps

공혜원

Azure Container Apps 소개

- 서비스 규모에 맞는 최신 애플리케이션을 사용할 수 있는 완벽하게 관리되는 서버리스 컨테이너 서비스
 - 오픈소스에 최신 어플리케이션 구축
 - 인프라가 아닌 애플리케이션에 집중
 - 순조로운 쿠버네티스로 업그레이드
-
- “모든 컨테이너에 패키징된 애플리케이션 코드를 실행할 수 있으며 런타임 또는 프로그래밍 모델에 제약 받지 않는다.”

일정

- Microsoft 2021 Ignite 소개
- 퍼블릭 프리뷰
- 지원하는 리전
 - 북유럽
 - 캐나다 중부

다른 애저 서버리스or컨테이너 서비스와 차이점

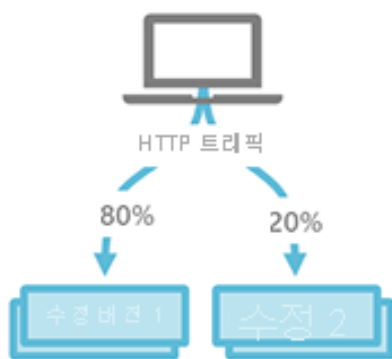
- Azure Container Instances : 한번에 1개의 컨테이너 인스턴스
- Azure Kubernetes Service : 쿠버네티스는 애저에, 제어는 사용자가
- Azure Functions : FaaS
- Web Apps : 웹앱, 모바일 백 엔드 및 RESTful API를 빌드하고 호스트

어떨때 쓰는지?



Azure Container Apps: 시나리오 예

공개 API 엔드포인트



HTTP 요청은 둘로 나뉩니다.
처리하는 프로세스가 데이터베이스의
애플리케이션
새 버전은 나머지
20%를 받습니다.

자동 스케일링 기준

스케일링은 숫자로 결정됩니다.
HTTP 요청을 확장할 수 있습니다.

백그라운드 처리



지속적으로 실행되는 배경
80%를 차지하는 컨테이너 앱의
입니다.

자동 스케일링 기준

스케일링은 수준에 따라 결정됩니다.
CPU 또는 메모리 로드

이벤트 기반 처리



첫 번째 개정에 대한 옵션이 있는
마이크로서비스 버전을 배포 및
큐입니다.

자동 스케일링 기준

스케일링은 숫자로 결정됩니다.
큐에서 메시지

마이크로서비스



아키텍처에 도착할 때 메시지를
관리하는 대기열 읽기 권한자
Dapr. 트래픽과 통합하고

자동 스케일링 기준

개별 마이크로서비스는 동시
KEDA 스케일링 트리거에 따라.

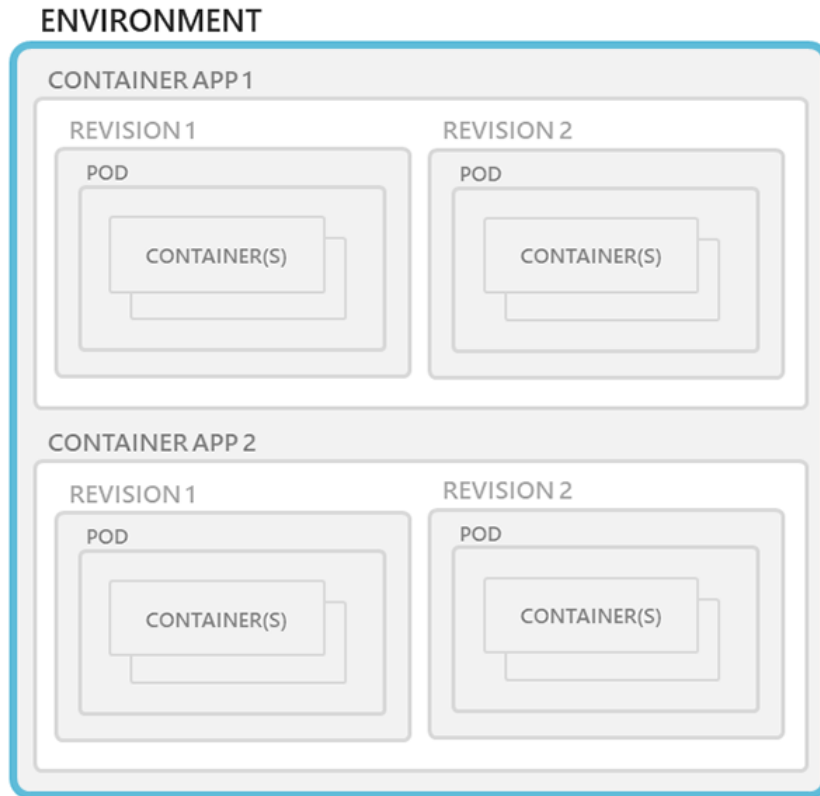
제한 사항

- 리눅스만 지원
- Root 권한 지원 안함
 - 프로그램이 root access가 필요한 프로세스를 실행하려고 하면 런타임 오류 발생
- 할당량(구독 단위)
 - 2개의 Environment
 - 1개 Environment당 20개 컨테이너 앱
 - 1개 Environment 당 50코어
 - 컨테이너 앱 당 최대 2코어/4GB메모리

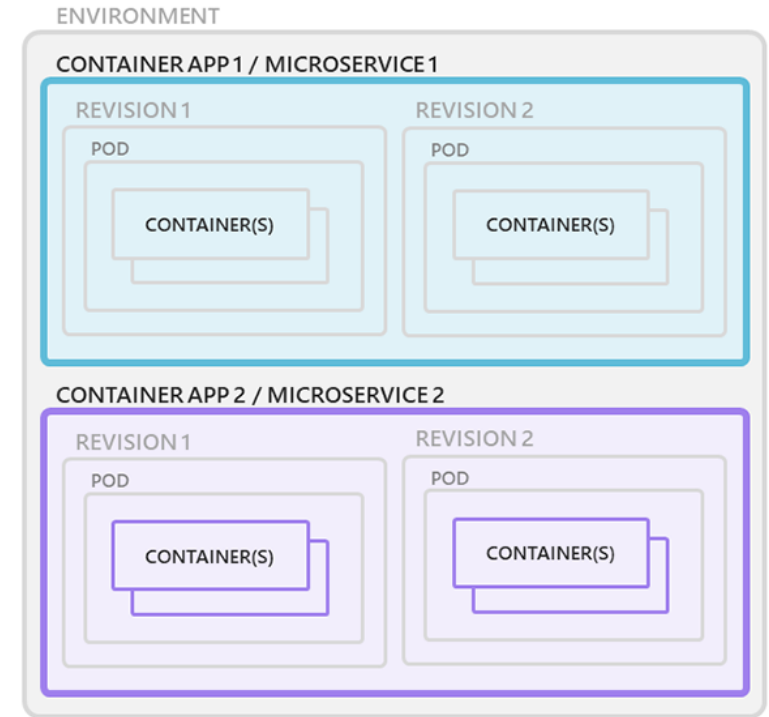
Container Apps 구조 설명



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



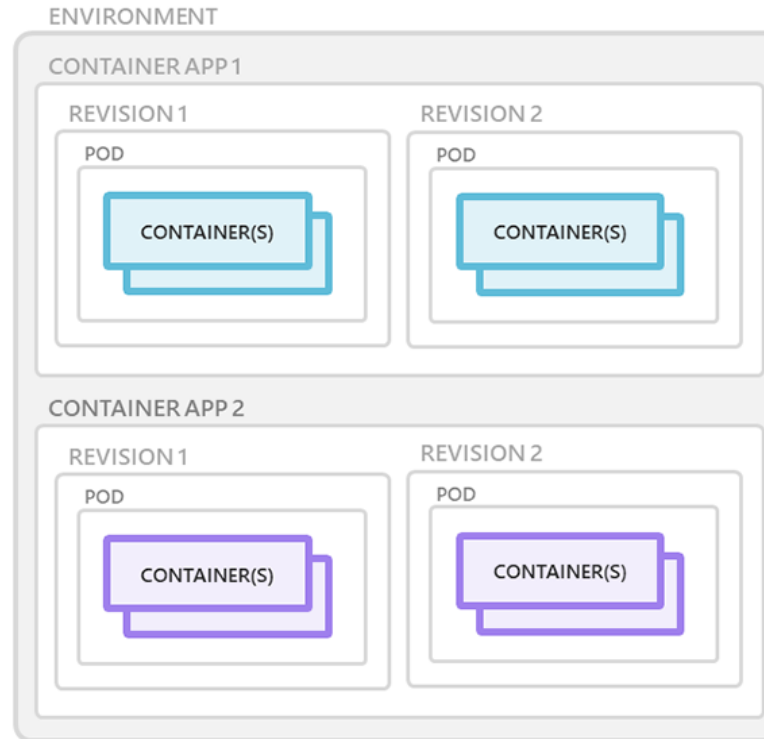
Container apps are deployed as **microservices**.



Container Apps 구조 설명



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



인그레스

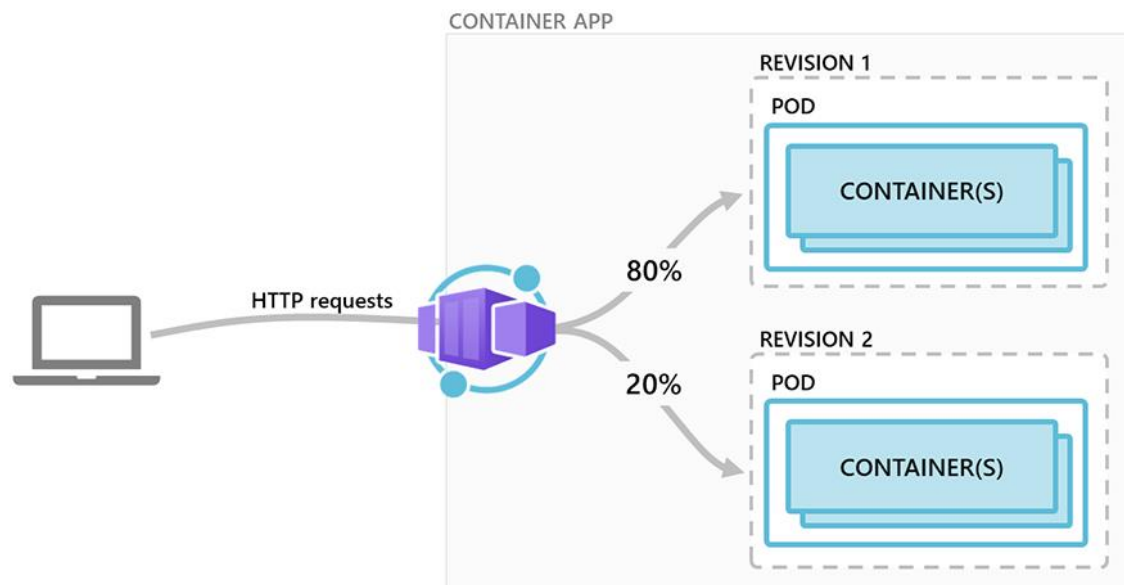
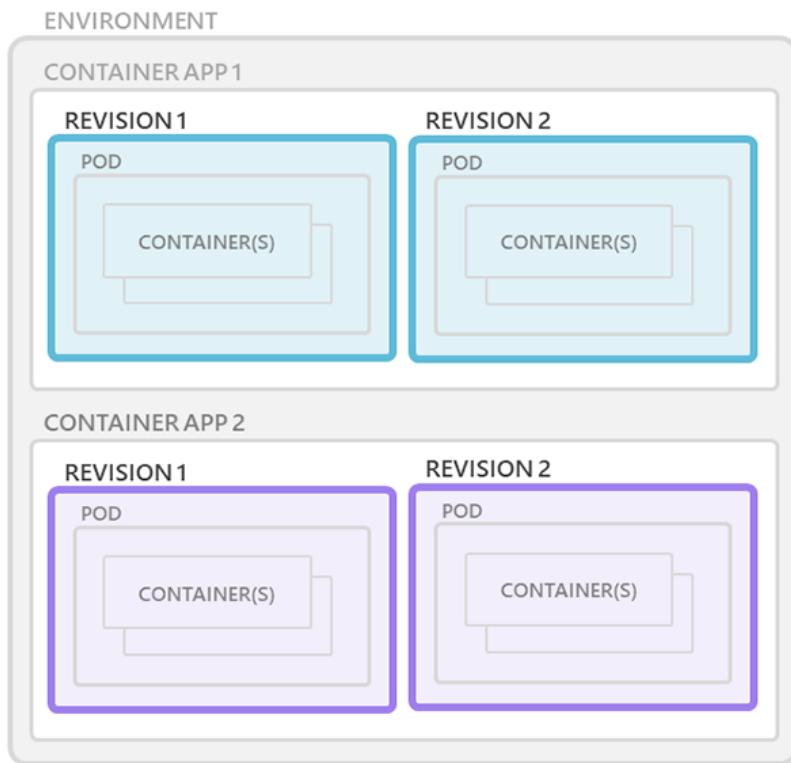
- Replica간에 트래픽을 자동으로 로드 밸런싱
- Public or VNET(private)
- HTTP/1.1 및 HTTP/2 지원
- 엔드포인트는 항상 포트 80(HTTP의 경우) 및 443(HTTPS의 경우)을 노출
 - 기본적으로 포트 80에 대한 HTTP 요청은 443에서 HTTPS로 자동으로 리디렉션.

리비전

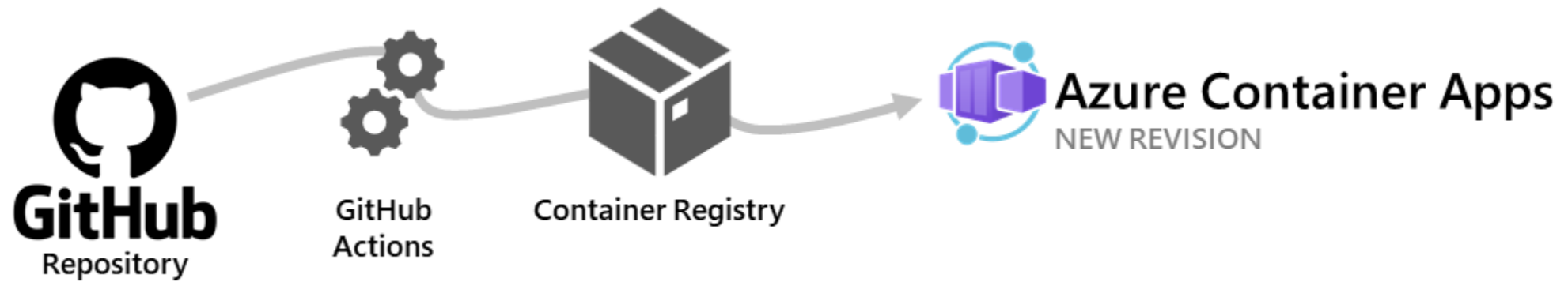
- 변하지 않는 스냅샷



Revisions are immutable snapshots of a container app.



Github Action

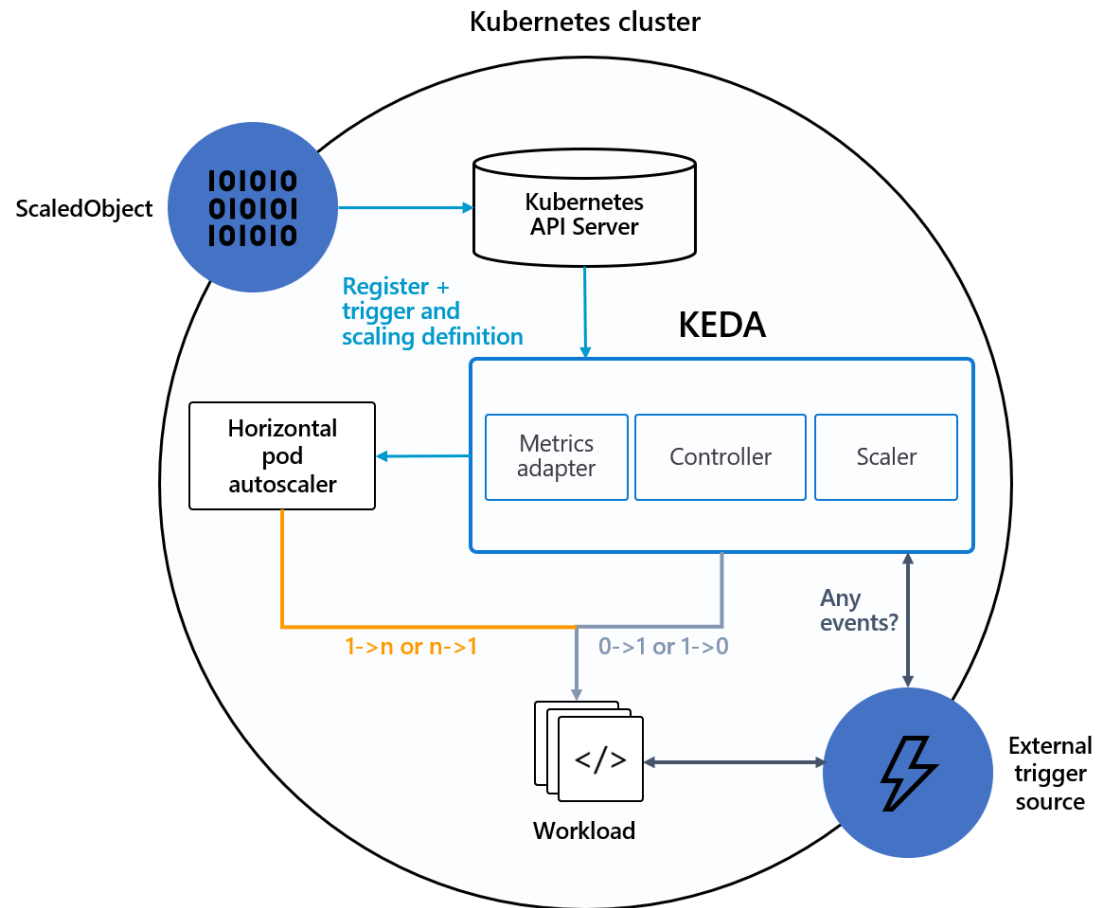


앱 스케일링

- KEDA Scalers
- HTTP – concurrentRequests
- Event driven
- CPU/Memory

KEDA 지원

- Kubernetes Event-Driven Autoscaling



Currently available scalers for KEDA

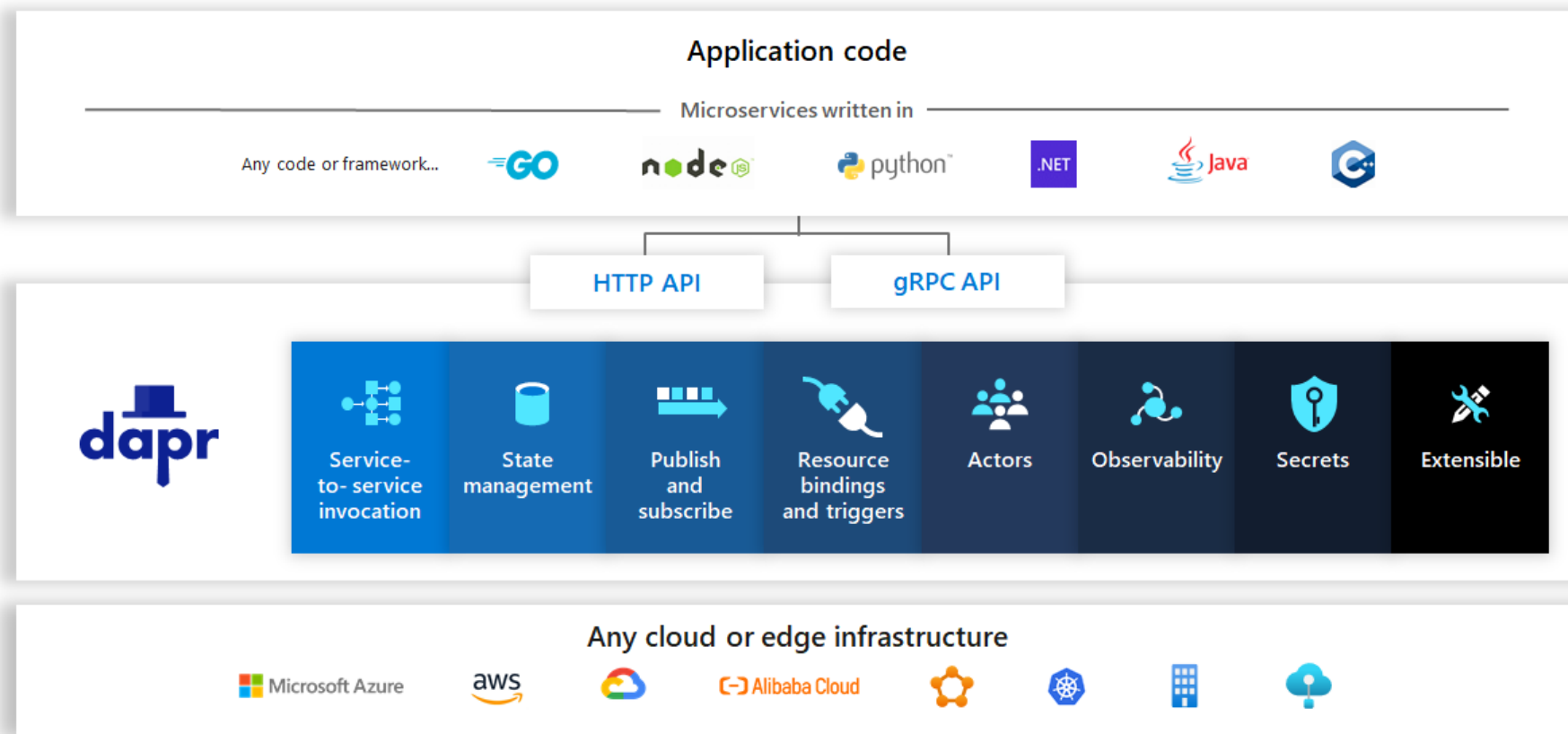
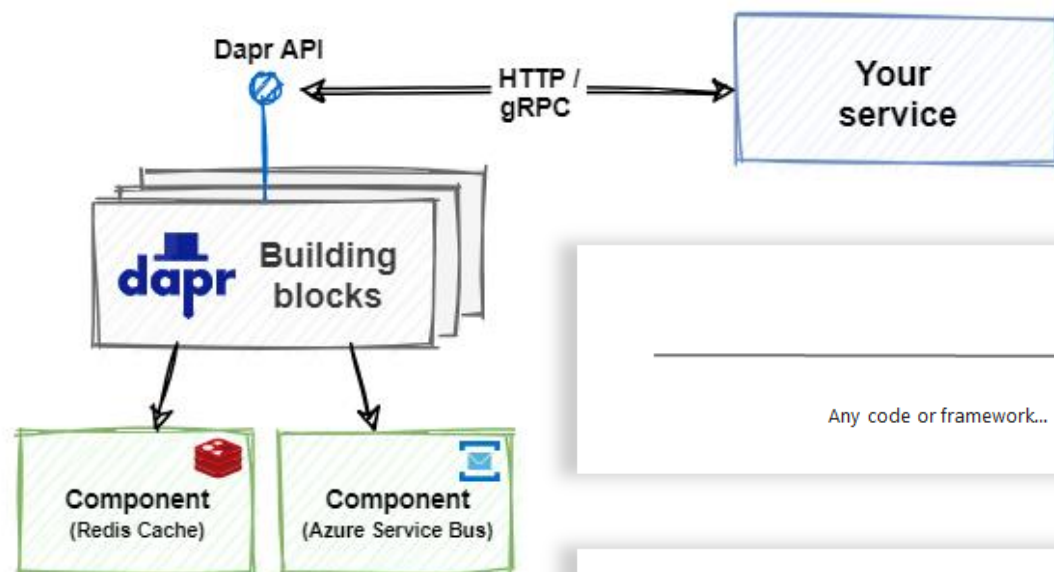


KEDA 지원

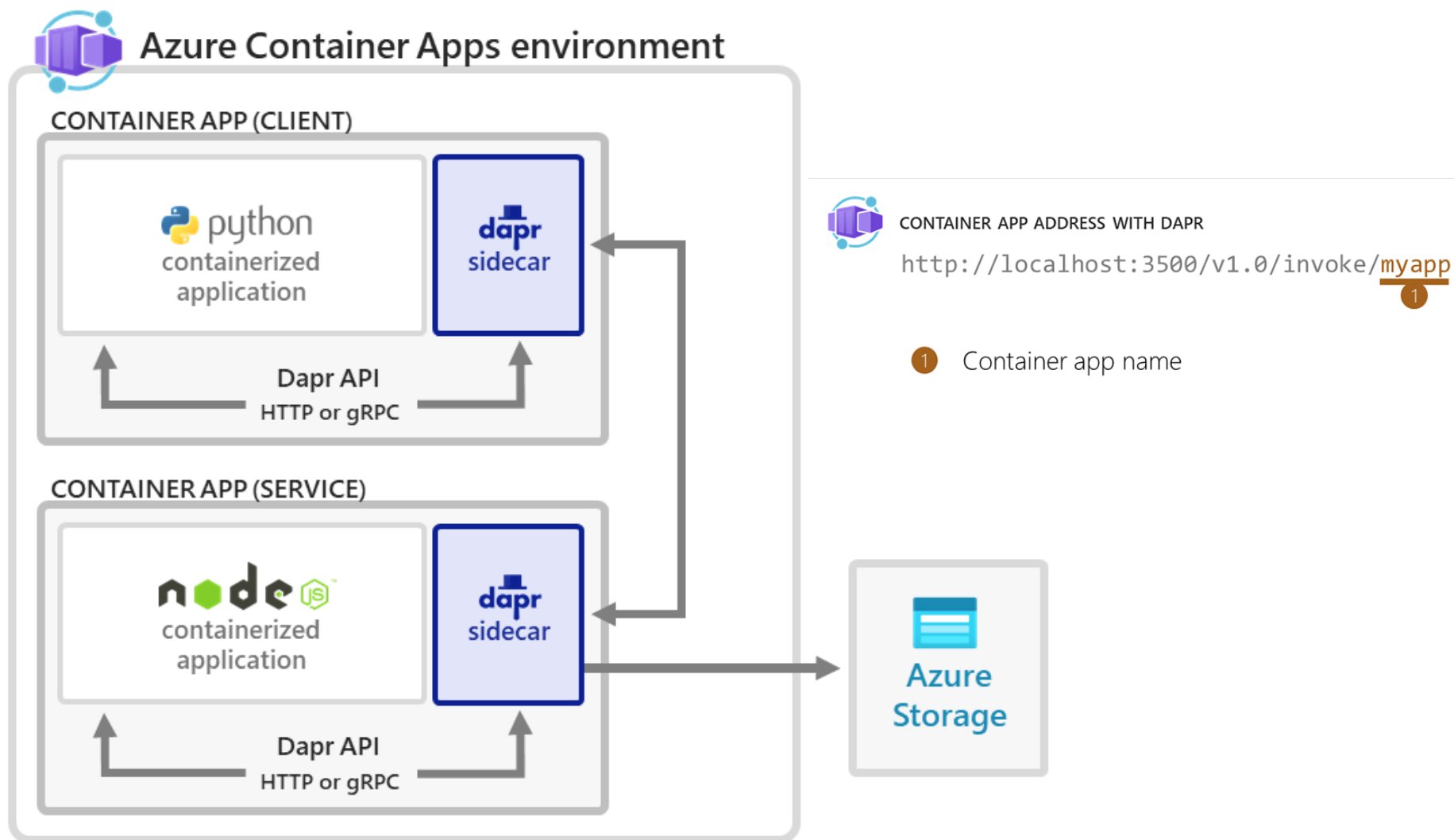
```
apiVersion: keda.sh/v1alpha1
kind: ScaledObject
metadata:
  name: order-processor-scaler
spec:
  scaleTargetRef:
    name: order-processor
  minReplicaCount: 0 # Change to define how many minimum replicas you want
  maxReplicaCount: 50
  # The period to wait after the last trigger reported active before scaling the resource back to 0.
  # By default it's 5 minutes (300 seconds).
  cooldownPeriod: 5
  triggers:
  - type: azure-servicebus
    metadata:
      queueName: orders
      messageCount: '50'
    authenticationRef:
      name: trigger-auth-service-bus-orders
```

```
apiVersion: keda.sh/v1alpha1
kind: TriggerAuthentication
metadata:
  name: trigger-auth-service-bus-orders
spec:
  secretTargetRef:
  - parameter: connection
    name: secrets-order-management
    key: servicebus-order-management-connectionstring
```

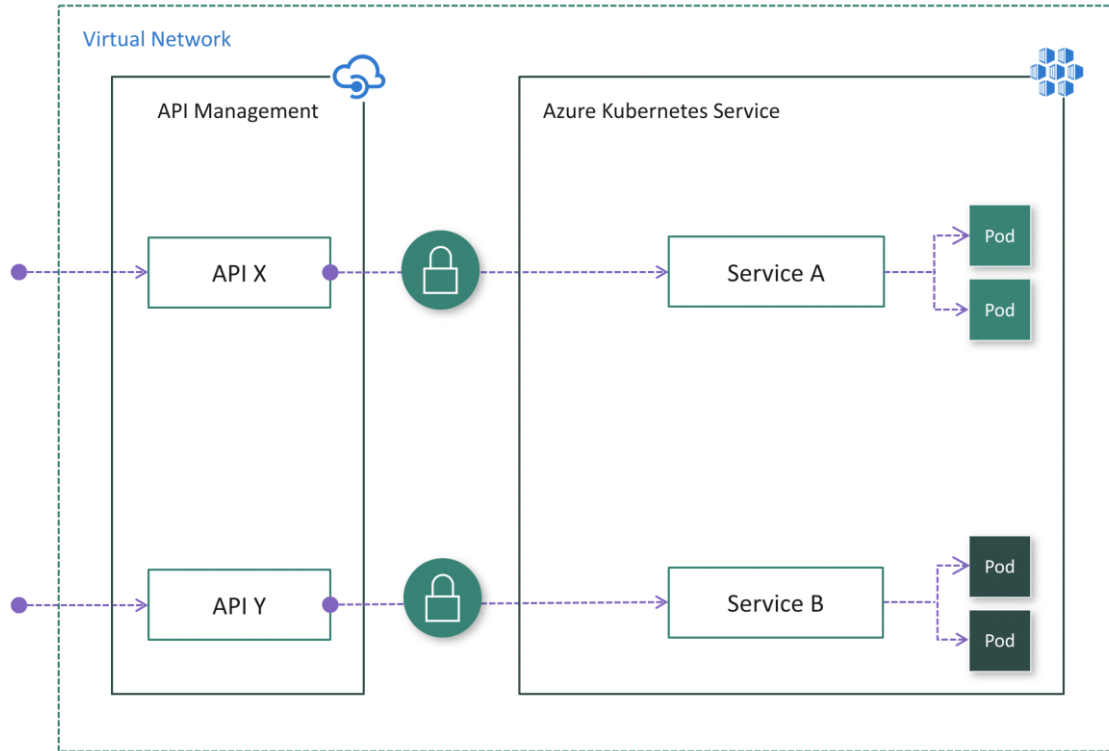
Dapr



Dapr



API Management



Create from definition



OpenAPI

Standard, language-agnostic interface to REST APIs



WADL

Standard XML representation of your RESTful API



WSDL

Standard WSA representation of your SOAP API

Create from Azure resource



Logic App

Scale hybrid integrations and workflows



App Service

API hosted on App Service



Function App

Serverless, event-driven workload on App Service



Container App

Serverless containers for microservices

샘플

Solution Overview

