



# Beyond Hello World:



Getting Deeper into Azure Functions

# THANK YOU TO OUR SPONSORS!

PLATINUM



GOLD



SILVER



BRONZE

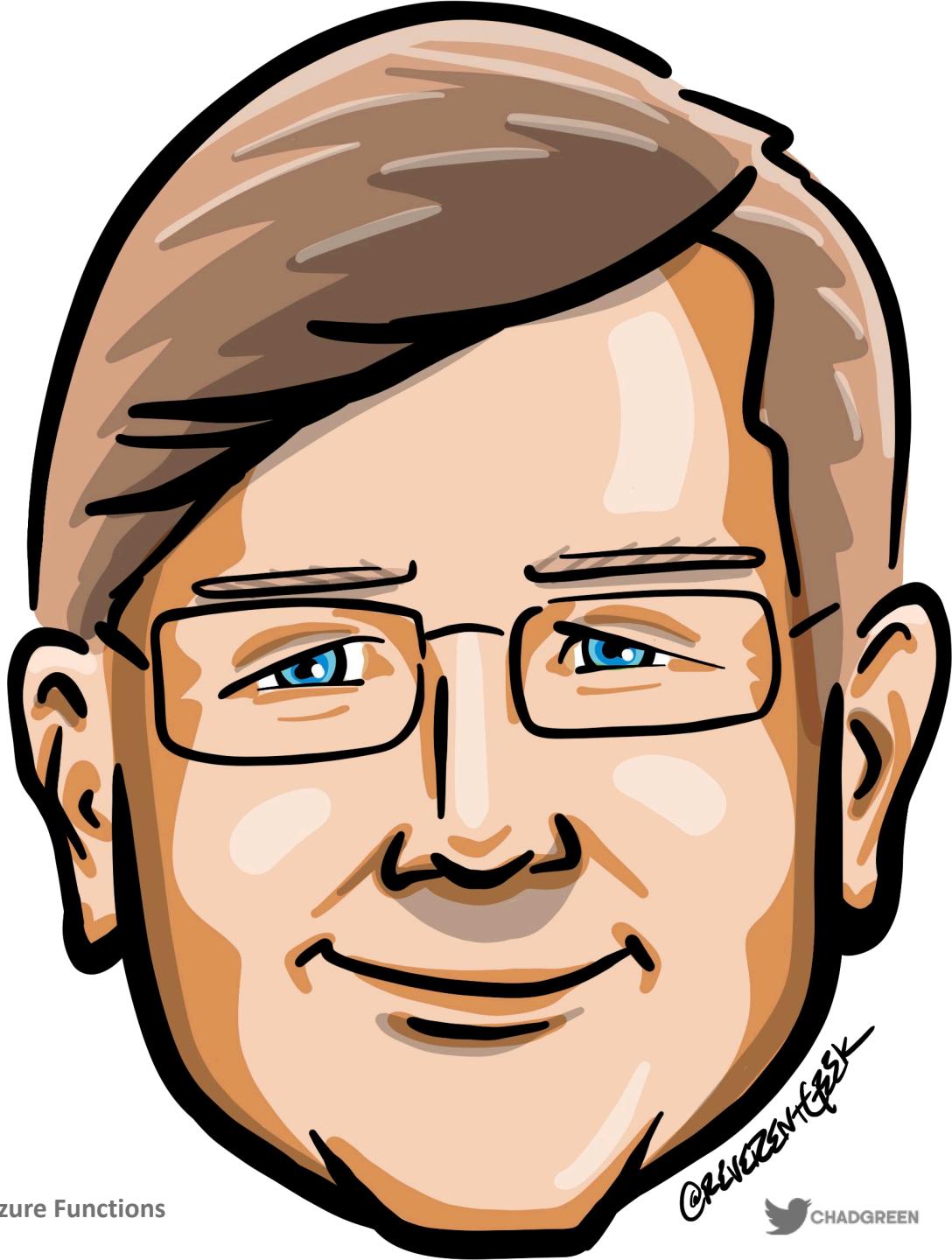


# Who is Chad Green?

- ✉️ chadgreen@chadgreen.com
- ✳️ TaleLearnCode
- 🌐 ChadGreen.com
- 🐦 ChadGreen & TaleLearnCode
- linkedin ChadwickEGreen



Beyond Hello World: Getting Deeper into Azure Functions



# Public Service Announcement



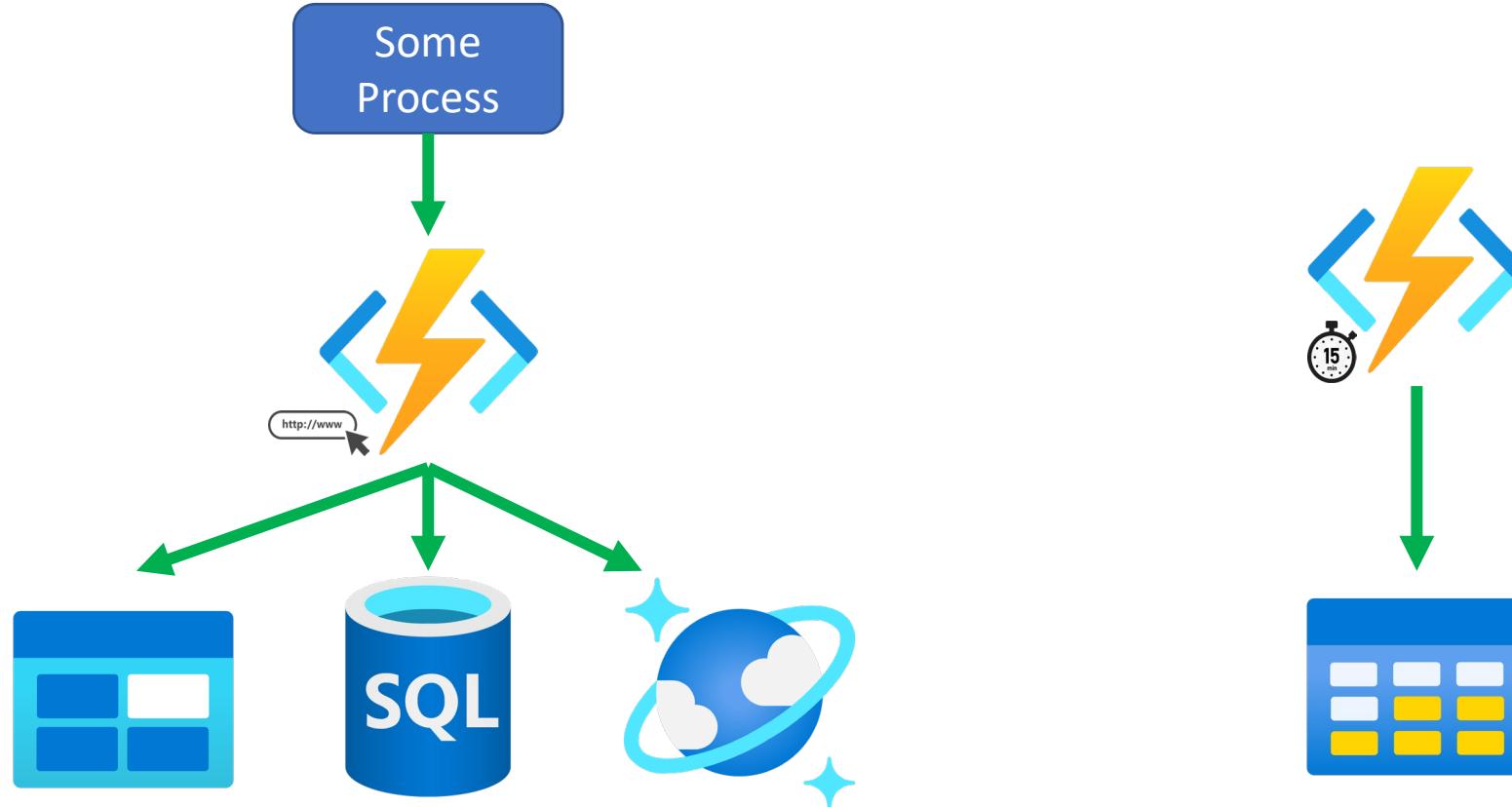
# Listen to your body





Listen to your body

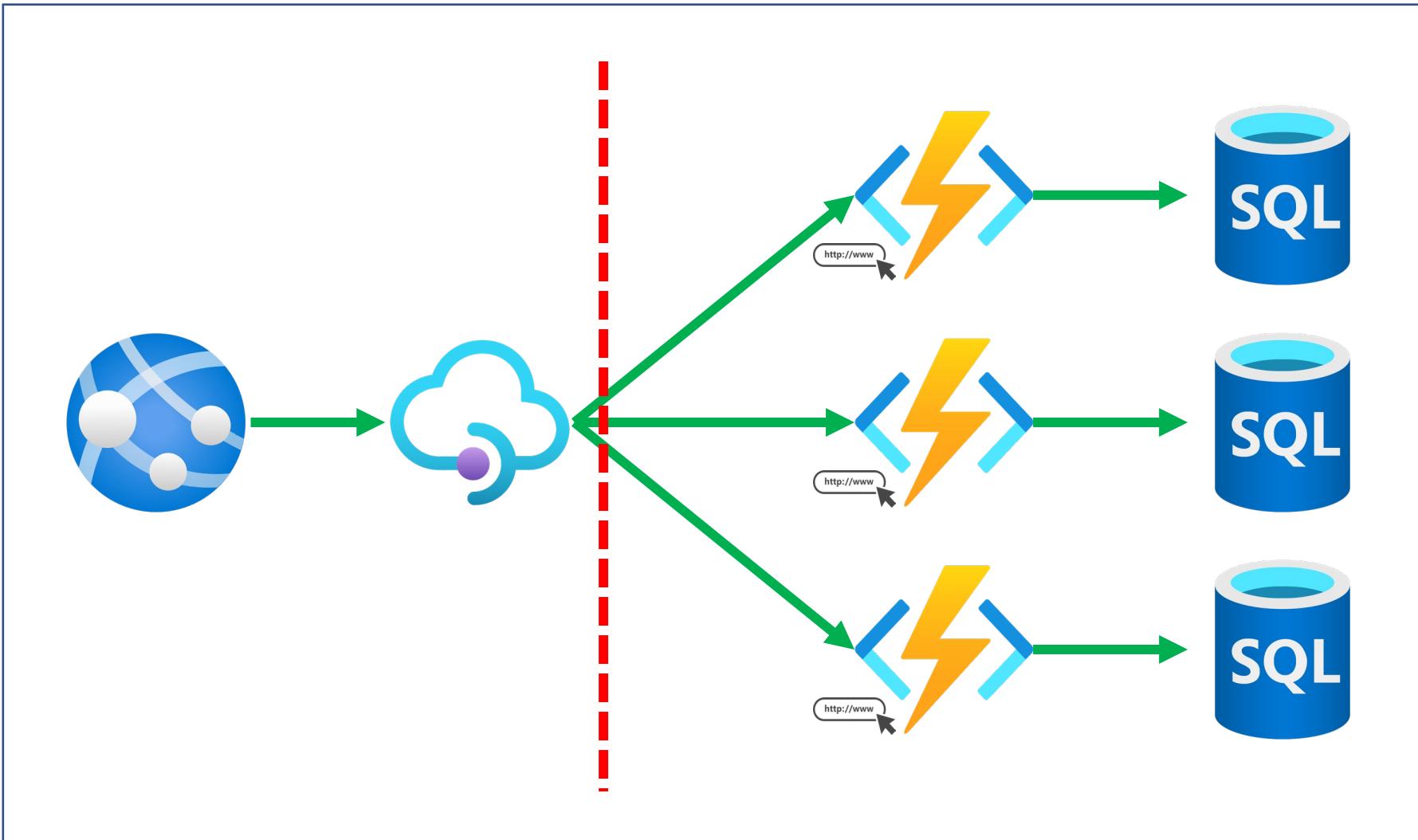
# Most Common Azure Function Patterns



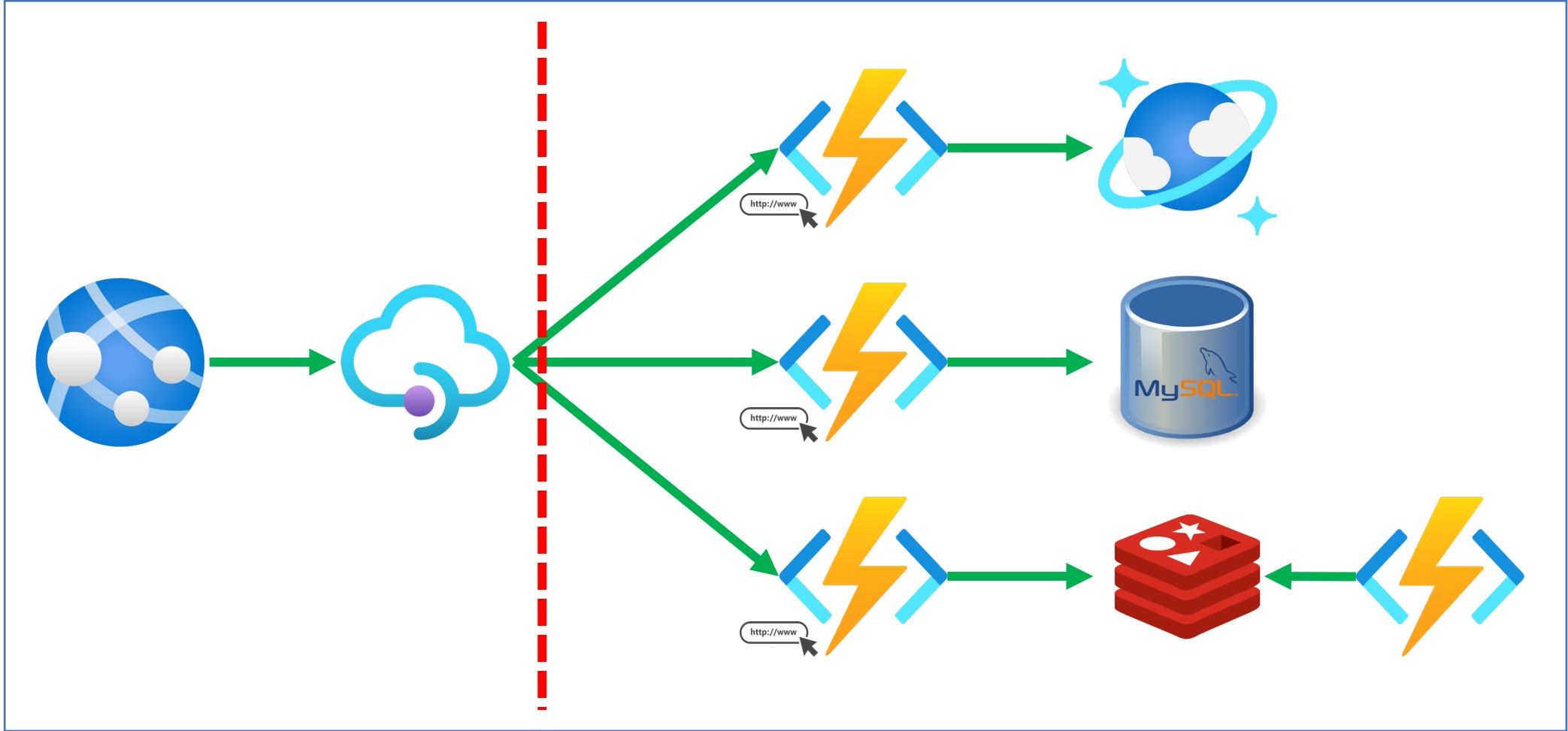
# Real World Scenarios

Beyond Hello World: Getting Deeper into Azure Functions

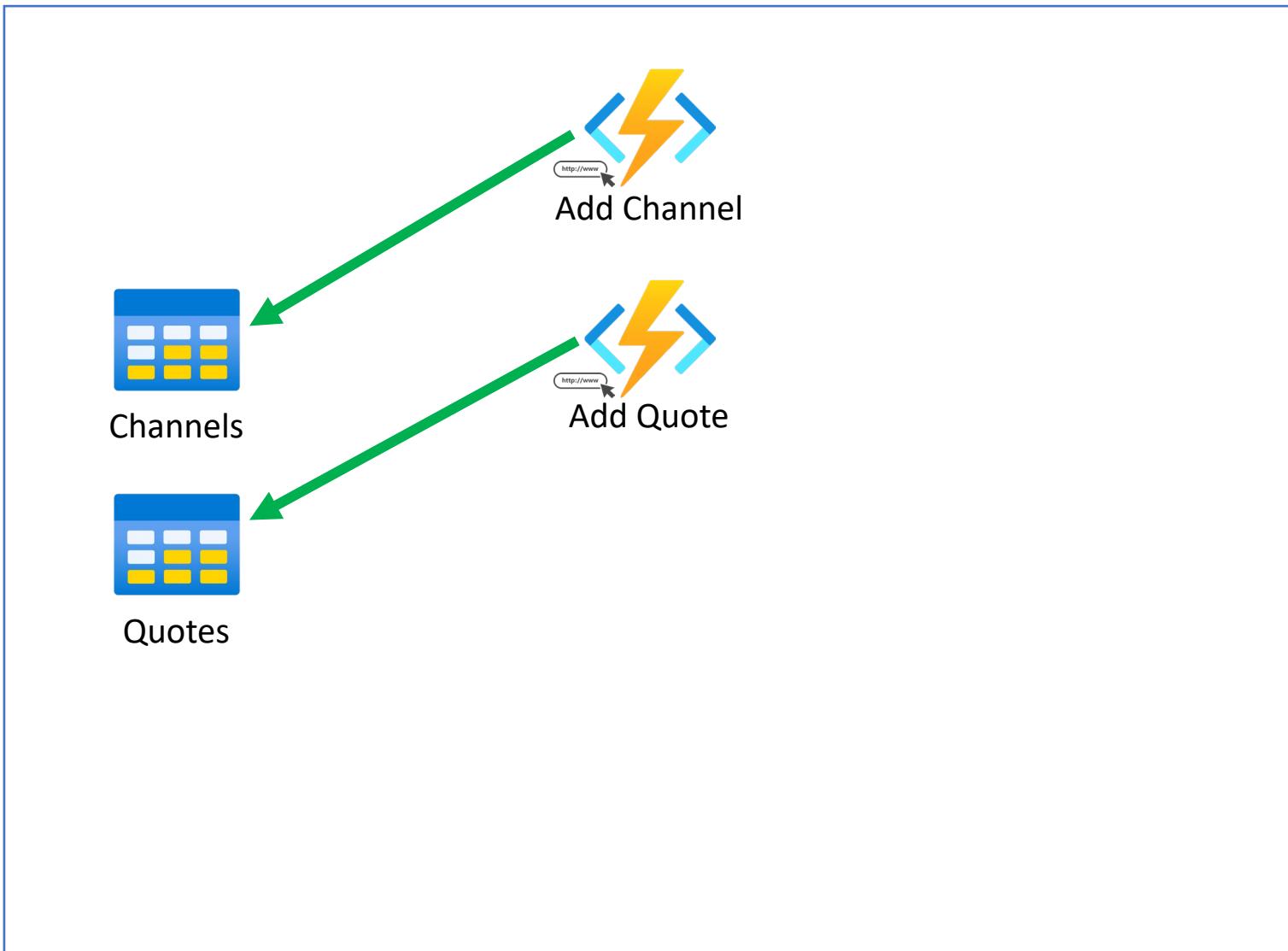
# CRUD Microservice



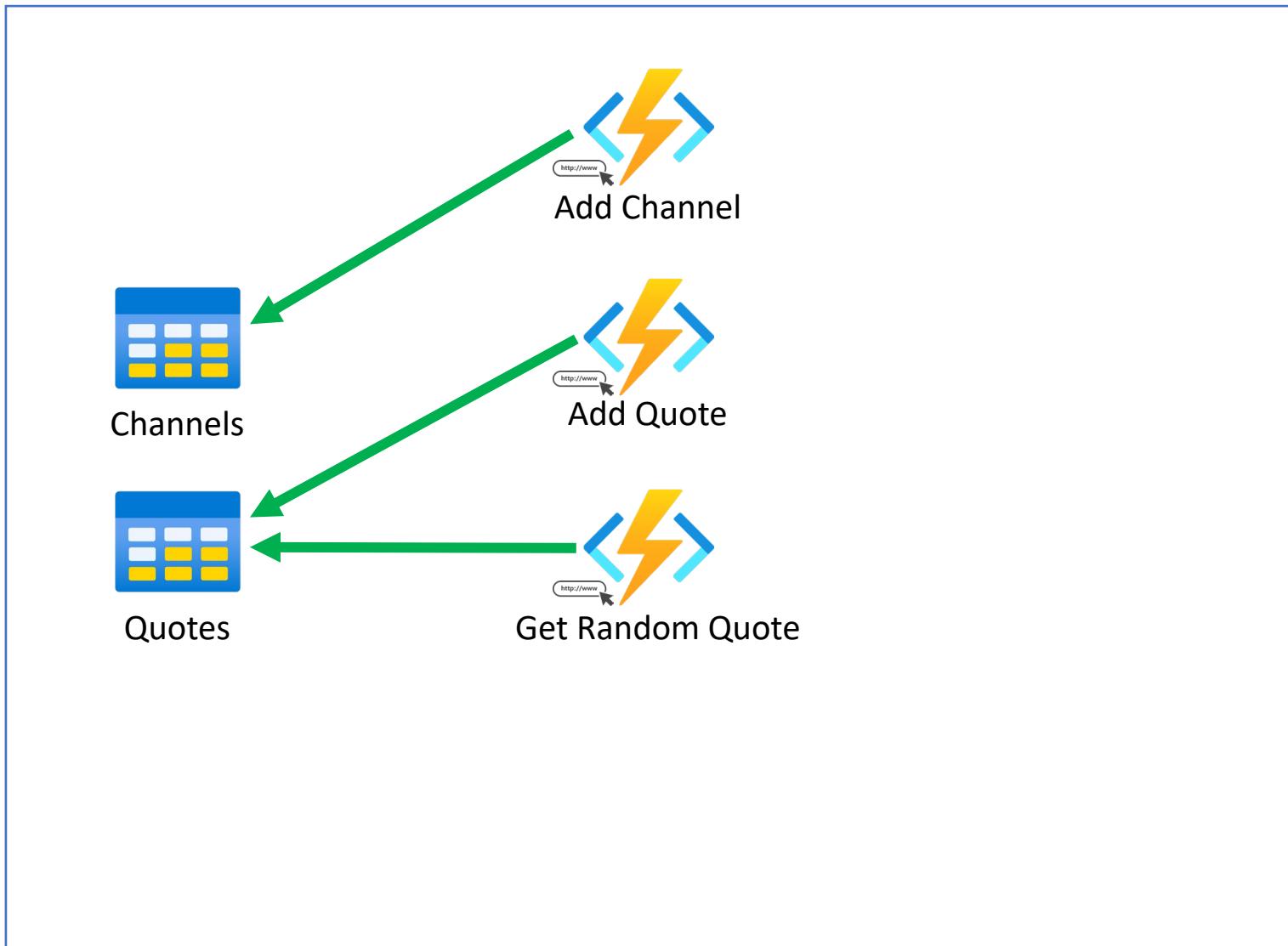
# CRUD Microservice



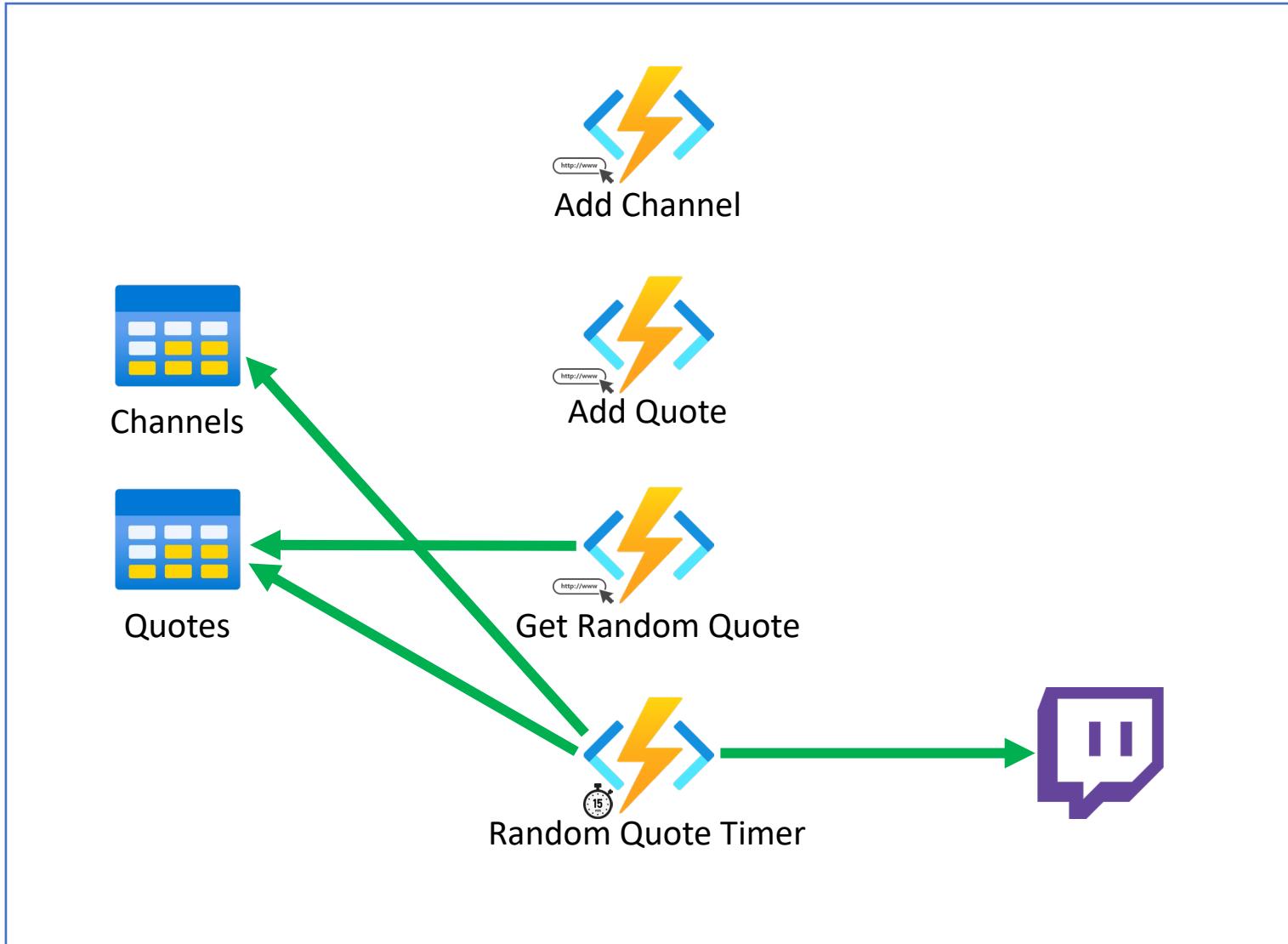
# Random Quote Generator



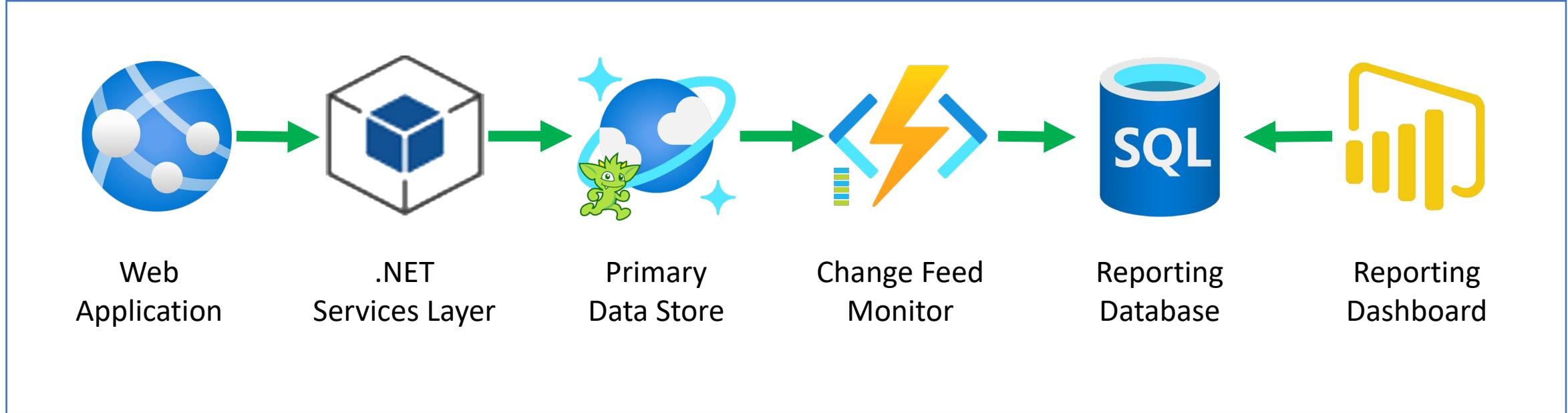
# Random Quote Generator



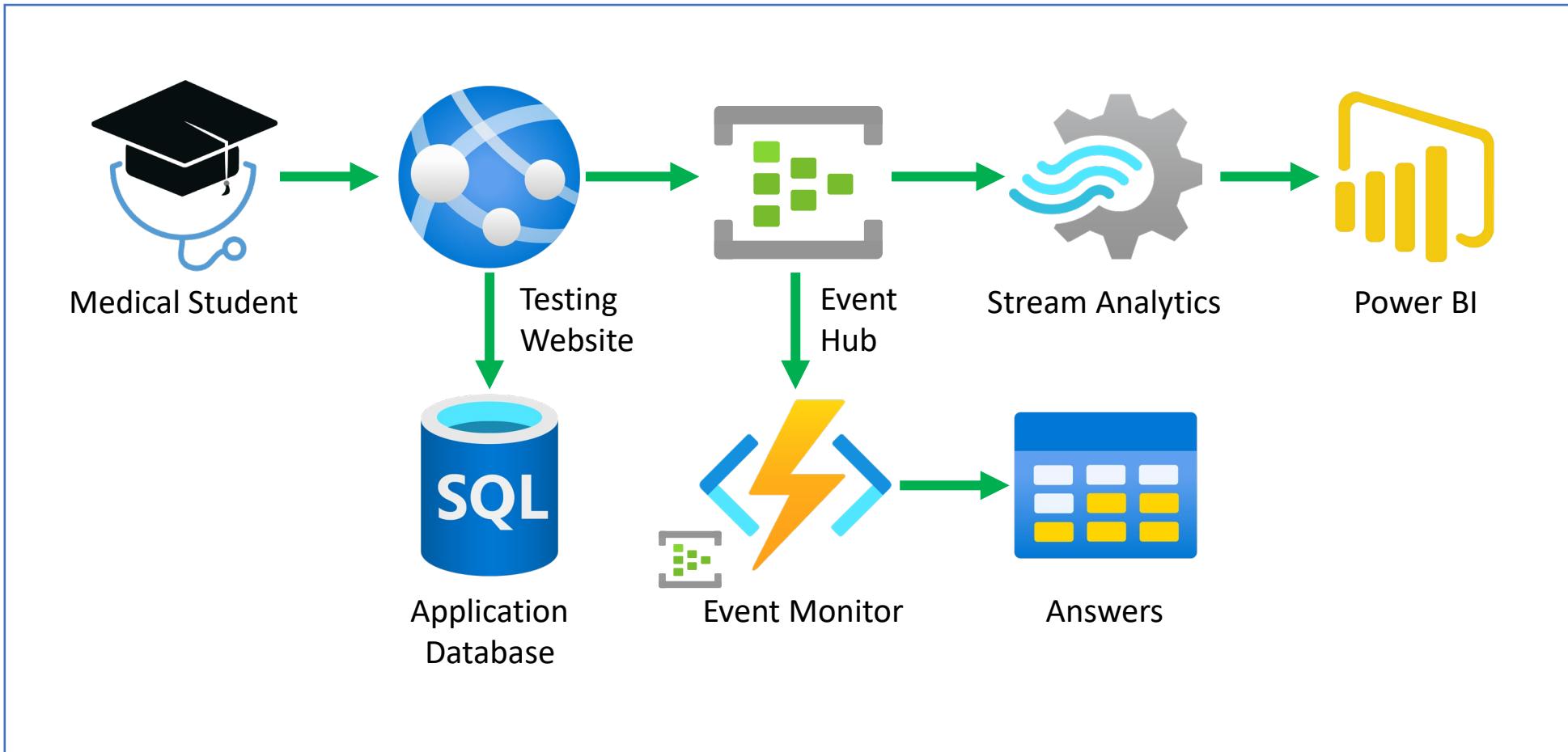
# Random Quote Generator



# Replicating Data



# Real-Time Reporting



# Glennis Platform



Portfolio



Sales



Leasing &  
Billing



Resident  
Care



eMAR



Resident  
Experience



Quality



Financials



Insights

ALINE

# Glennis Platform



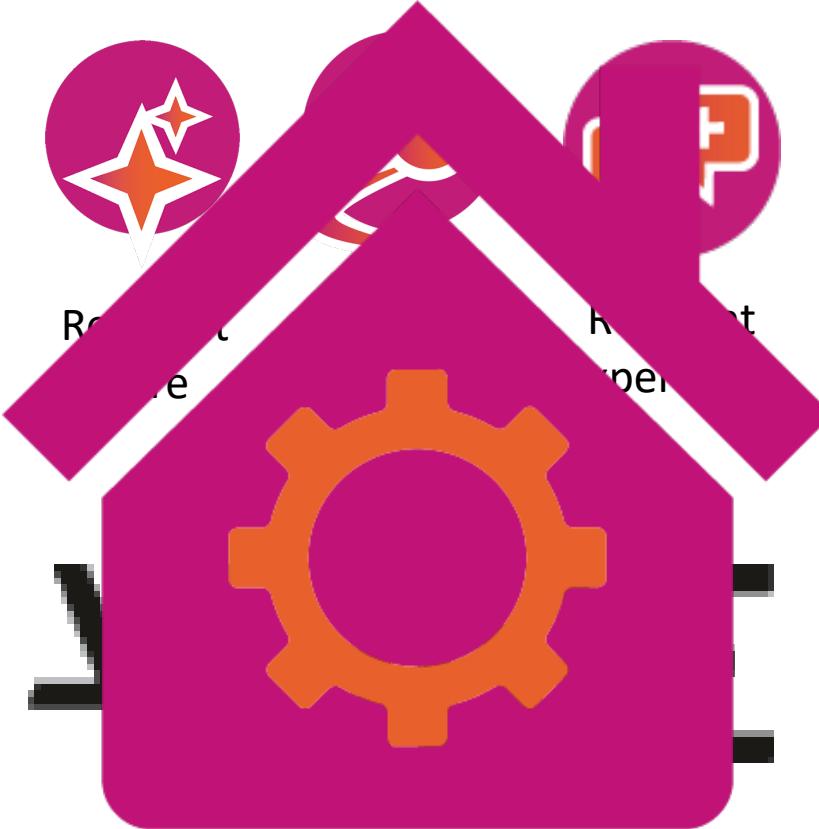
Portfolio



Sales



Leasing &  
Billing



Risk &  
Compliance



Recruitment  
Operations



Quality



Financials



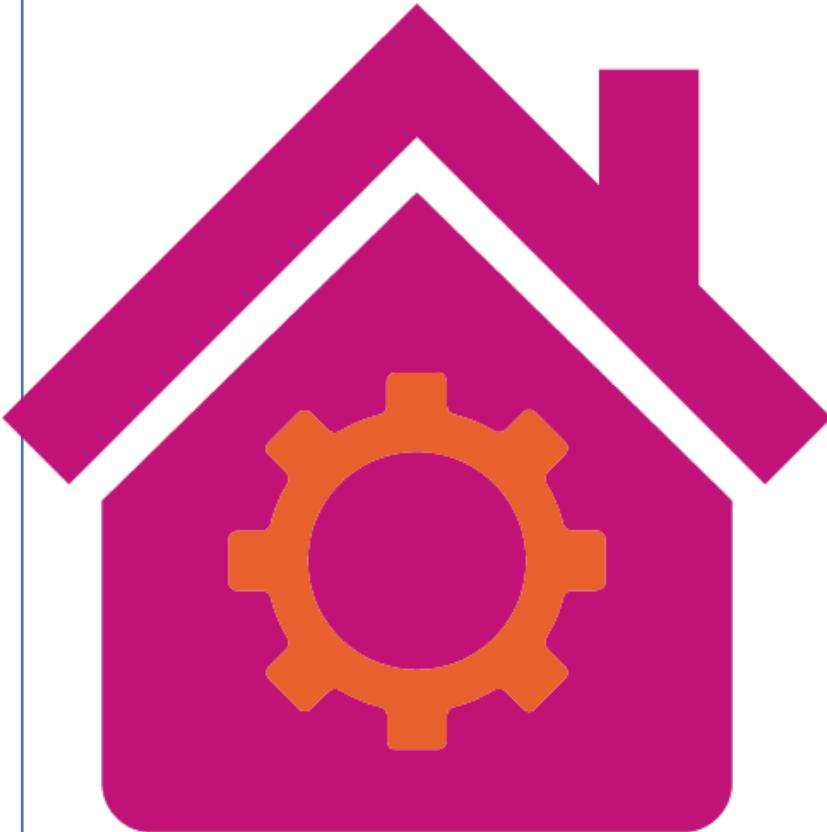
Insights

# Glennis Platform

## Glennis Portfolio

- Portfolio Management
- Care and Services Management
- Pricing Management
- Marketing Data Management
- Marketing APIs

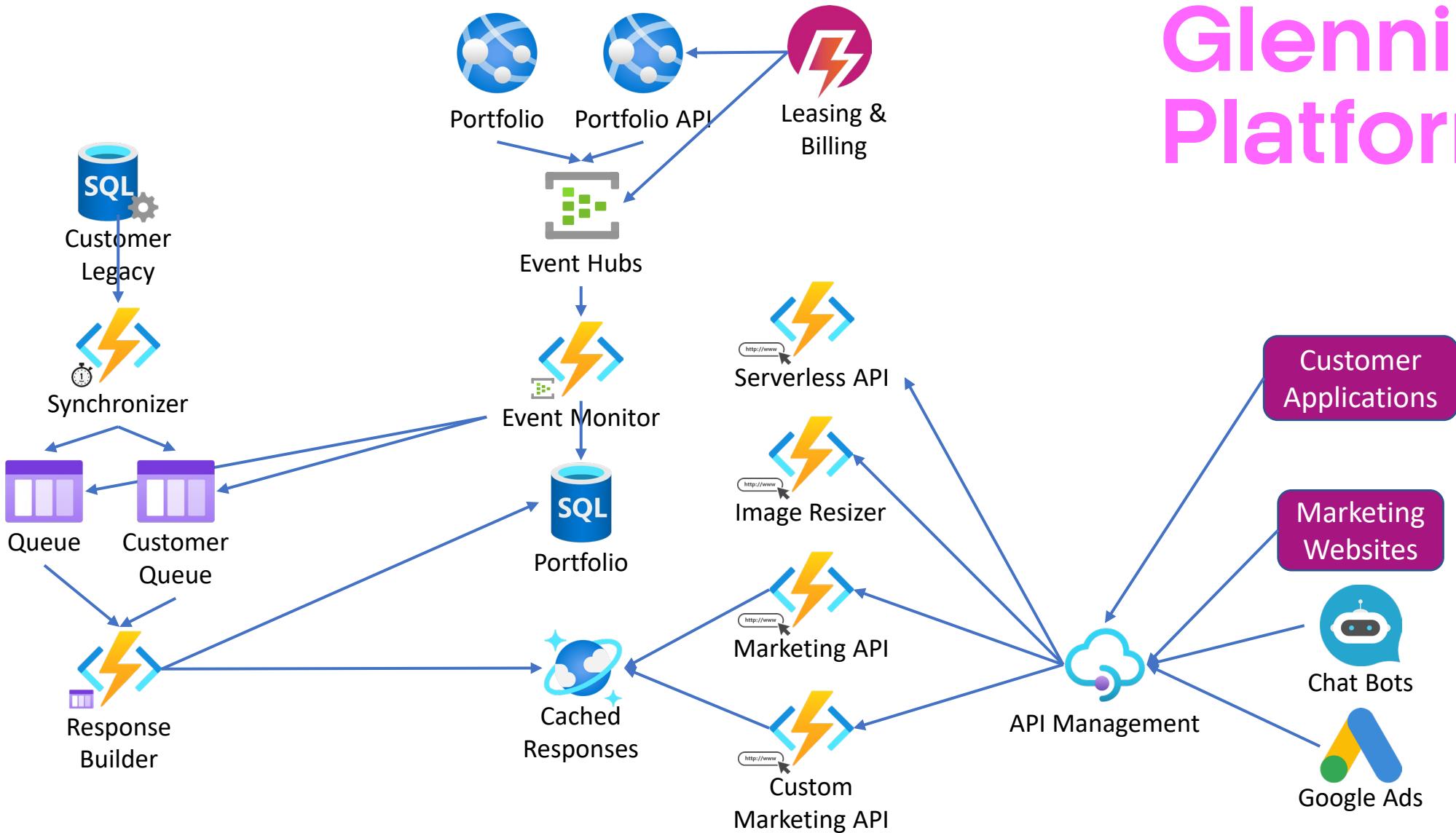
# Glennis Platform



## Glennis Portfolio

- Portfolio Management
- Care and Services Management
- Pricing Management
- Marketing Data Management
- Marketing APIs

# Glennis Platform

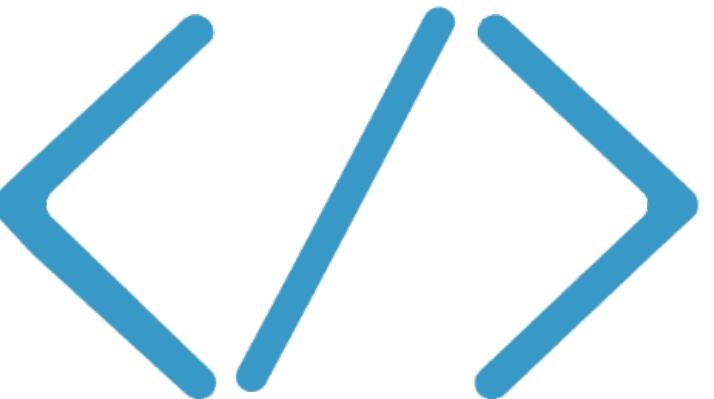


# Azure Functions Basics

Beyond Hello World: Getting Deeper into Azure Functions

# Azure Functions

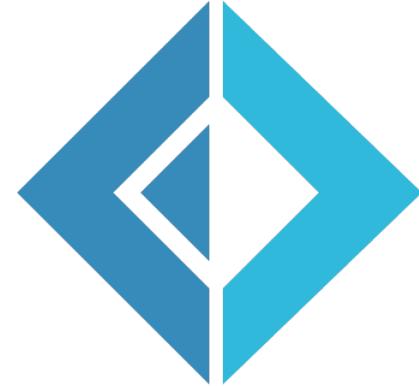
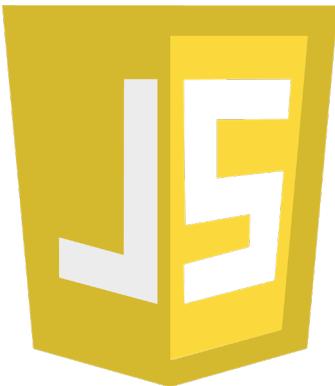
Code



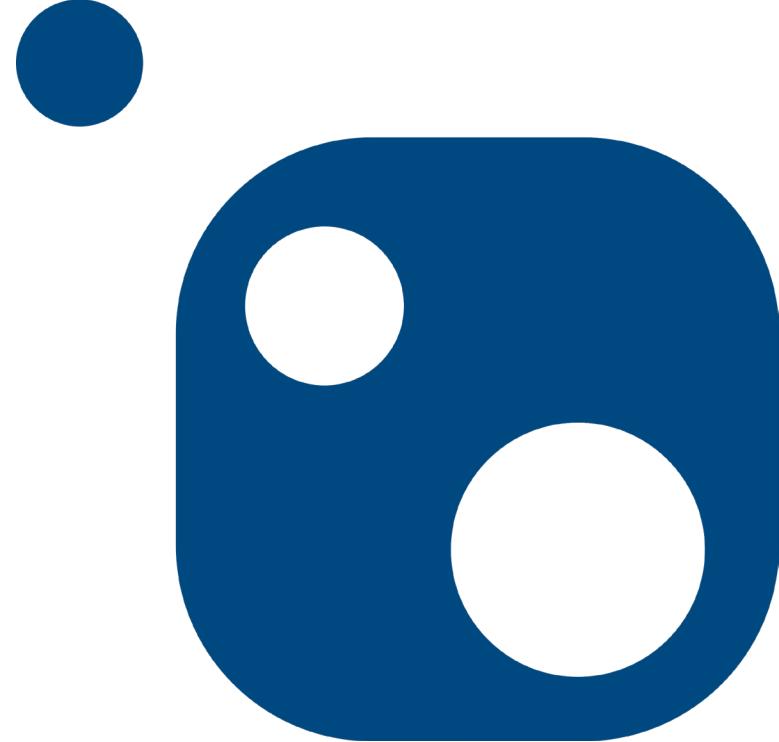
Events + data



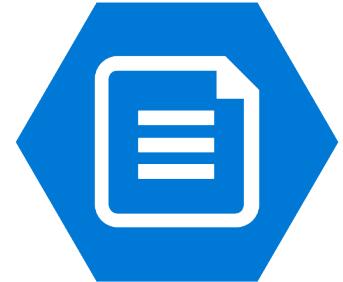
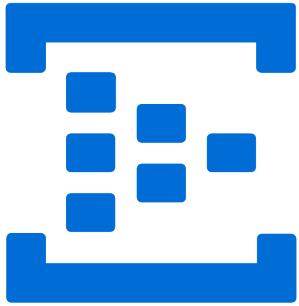
# Choice of Language



# Bring your own dependencies



# Simplified Integration

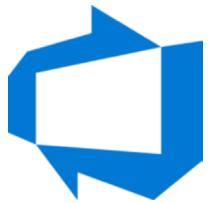
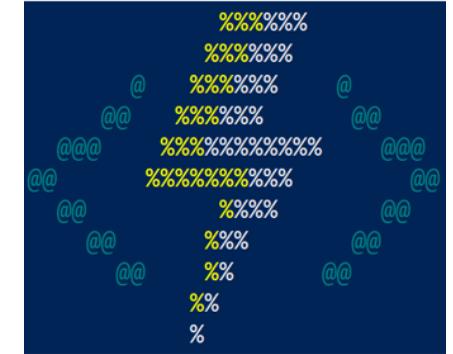
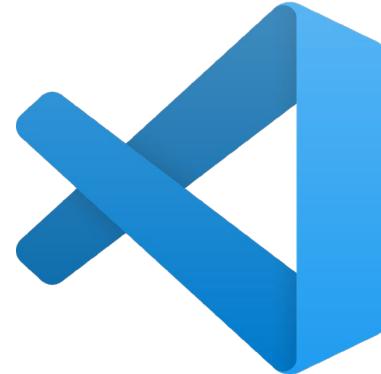
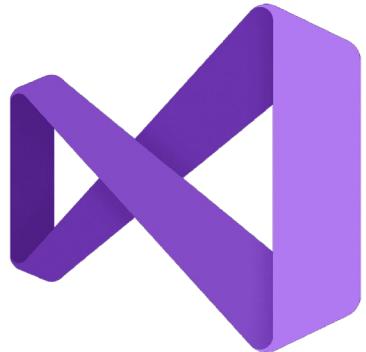


RabbitMQ



twilio

# Flexible Development



**Maven™**



# Many Deployment Options

## Consumption



Serverless

# Many Deployment Options

Consumption



Serverless

**App Service Plan**



Free, Basic,  
Standard,  
Premium

# Many Deployment Options

Consumption



Serverless

App Service Plan



Free, Basic,  
Standard,  
Premium

App Service Environment



Network  
Isolation

# Many Deployment Options

Consumption



Serverless

App Service Plan



Free, Basic,  
Standard,  
Premium

App Service Environment



Network  
Isolation

Azure Stack



On Premises

# Many Deployment Options

## Consumption



Serverless

## App Service Plan



Free, Basic,  
Standard,  
Premium

## App Service Environment



Network  
Isolation

## Azure Stack



On Premises

## Azure Functions Runtime



Functions on  
Your Server

# Many Deployment Options

## Consumption



Serverless

## App Service Plan



Free, Basic,  
Standard,  
Premium

## App Service Environment



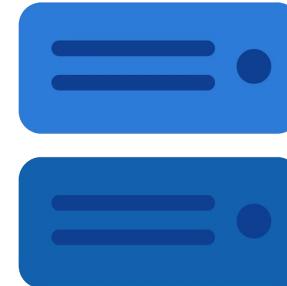
Network  
Isolation

## Azure Stack



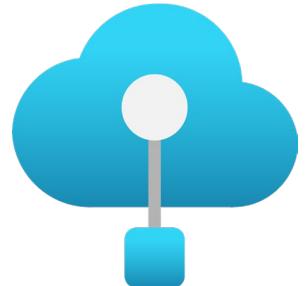
On Premises

## Azure Functions Runtime



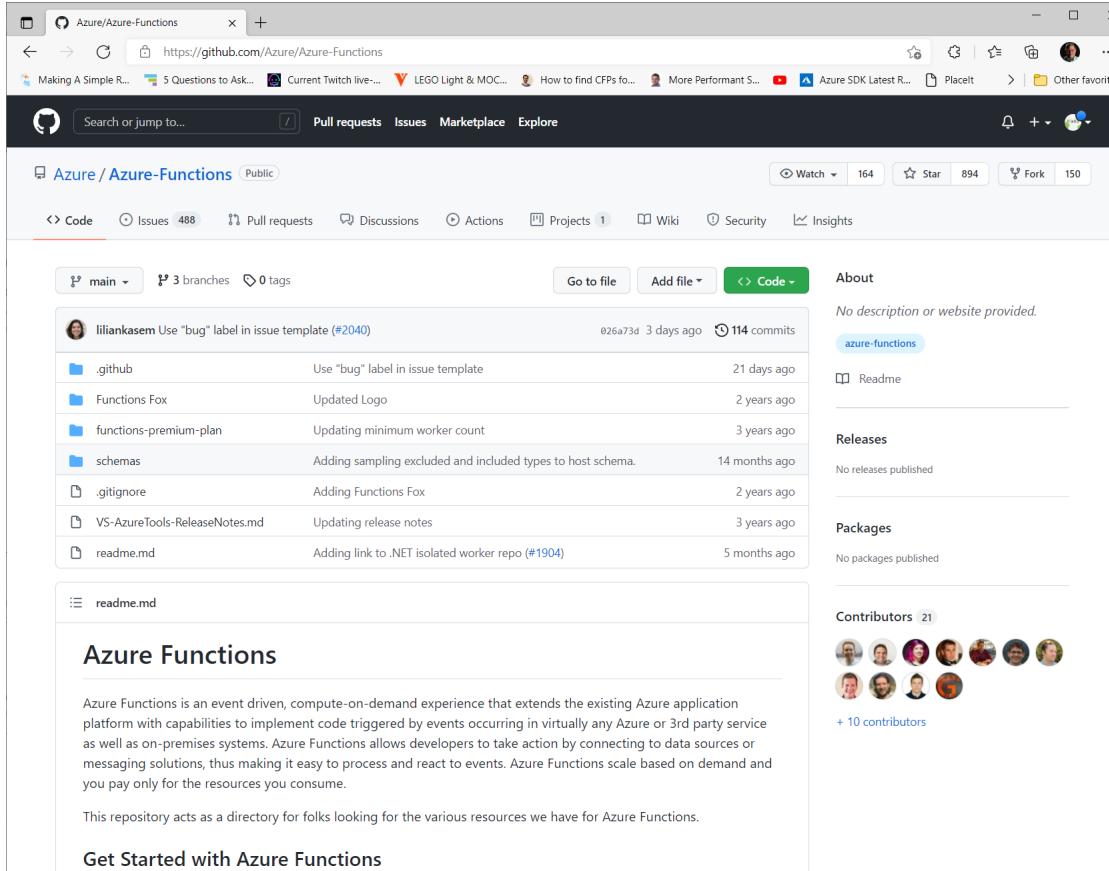
Functions on  
Your Server

## Azure IoT Edge



On Devices

# Open Source



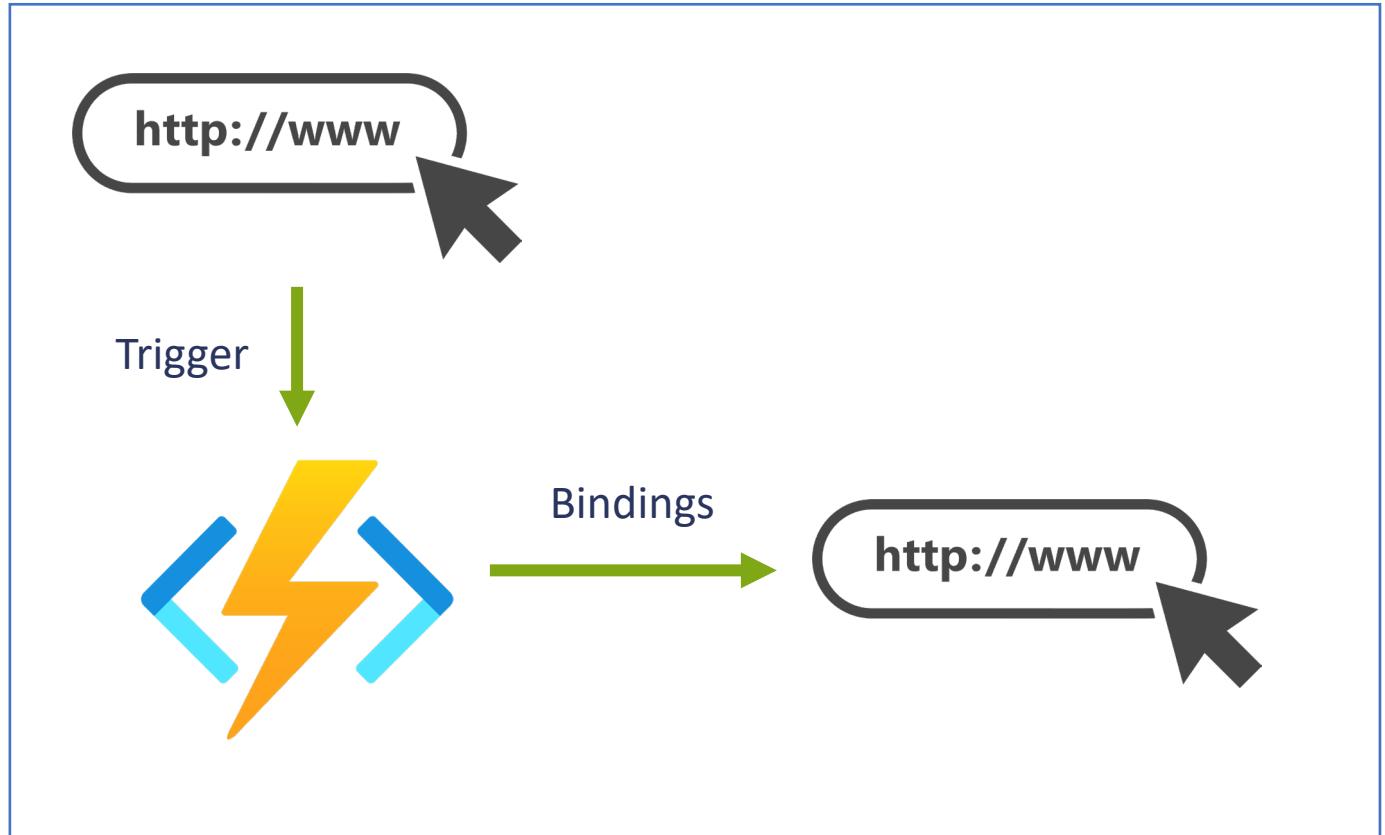
Beyond Hello World: Getting Deeper into Azure Functions

# Supported Triggers and Bindings

Beyond Hello World: Getting Deeper into Azure Functions

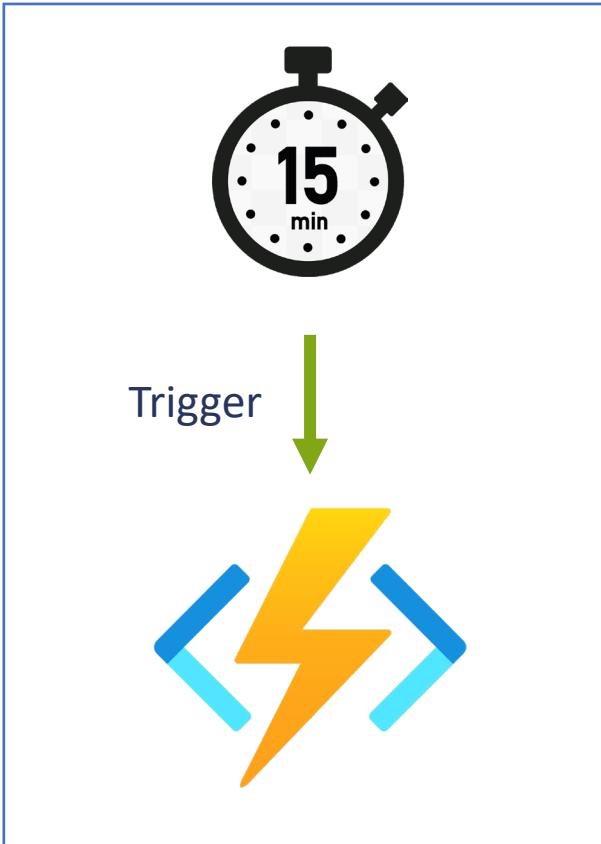
# HTTP

Trigger	✓
Input	✗
Output	✓
Consumption	✓
Premium	✓
Dedicated	✓
C# In-Process	✓
C# Isolated	✓
Java	✓
JavaScript	✓
PowerShell	✓
Python	✓



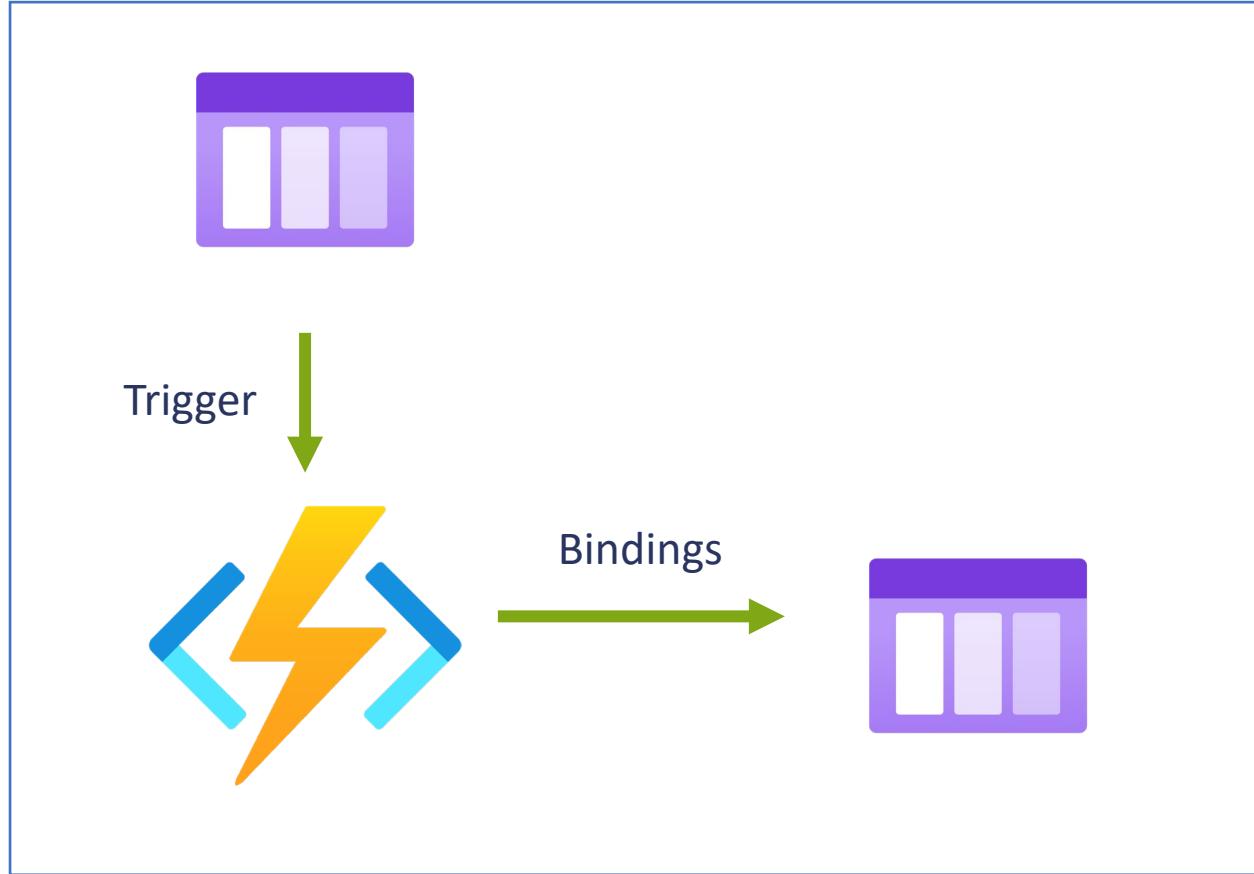
# Timer

Trigger	✓
Input	✗
Output	✗
Consumption	✓
Premium	✓
Dedicated	✓
C# In-Process	✓
C# Isolated	✓
Java	✓
JavaScript	✓
PowerShell	✓
Python	✓



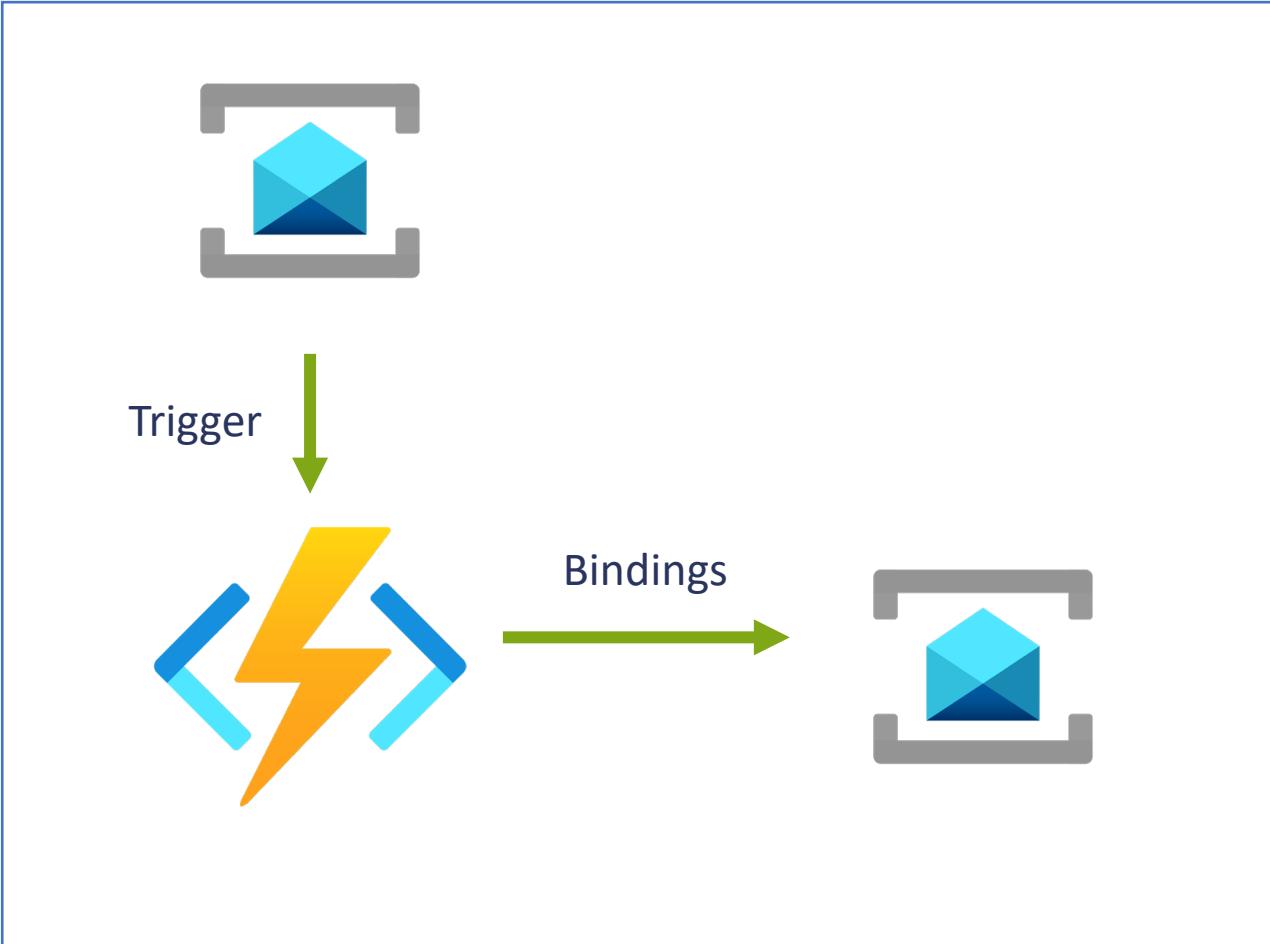
# Queue Storage

Trigger	✓
Input	✗
Output	✓
Consumption	✓
Premium	✓
Dedicated	✓
C# In-Process	✓
C# Isolated	✓
Java	✓
JavaScript	✓
PowerShell	✓
Python	✓



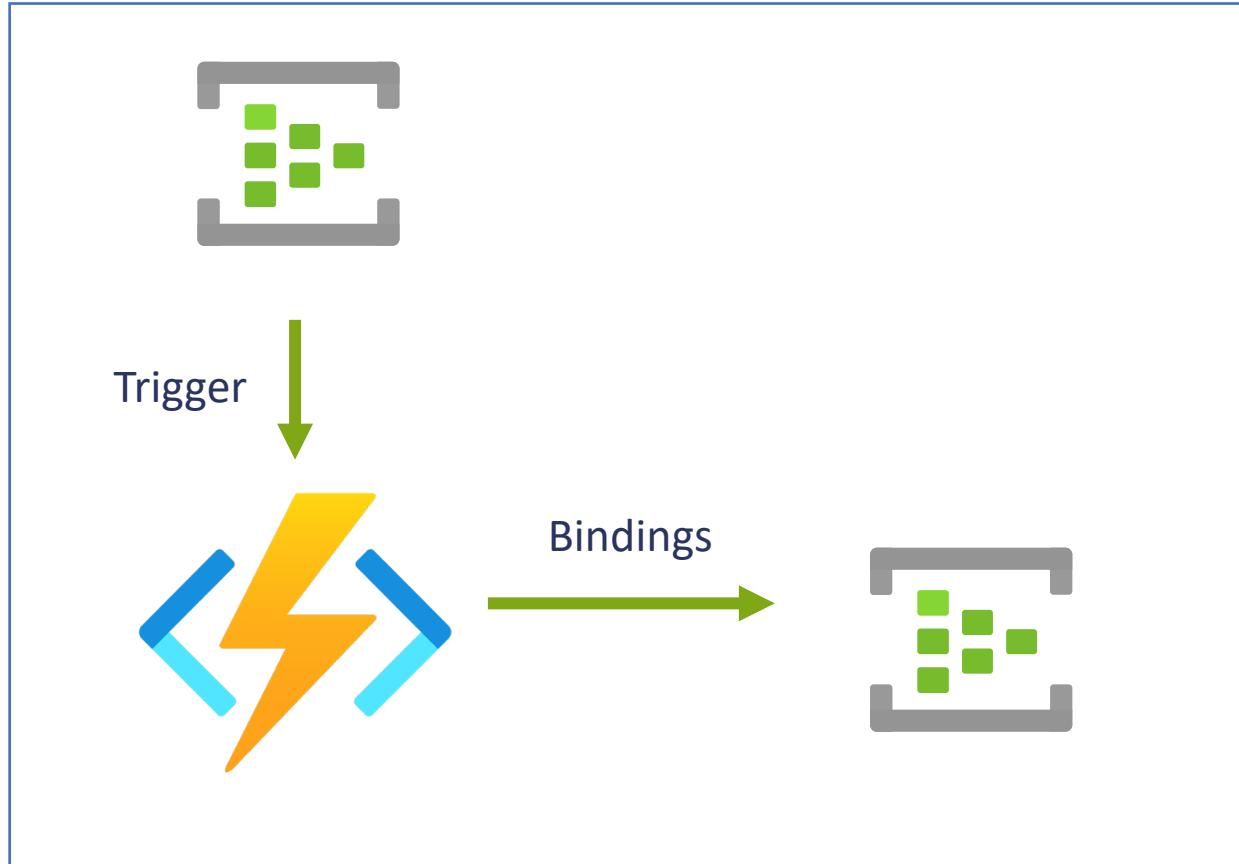
# Service Bus

Trigger	✓
Input	✗
Output	✓
Consumption	✓
Premium	✓
Dedicated	✓
C# In-Process	✓
C# Isolated	✓
Java	✓
JavaScript	✓
PowerShell	✓
Python	✓



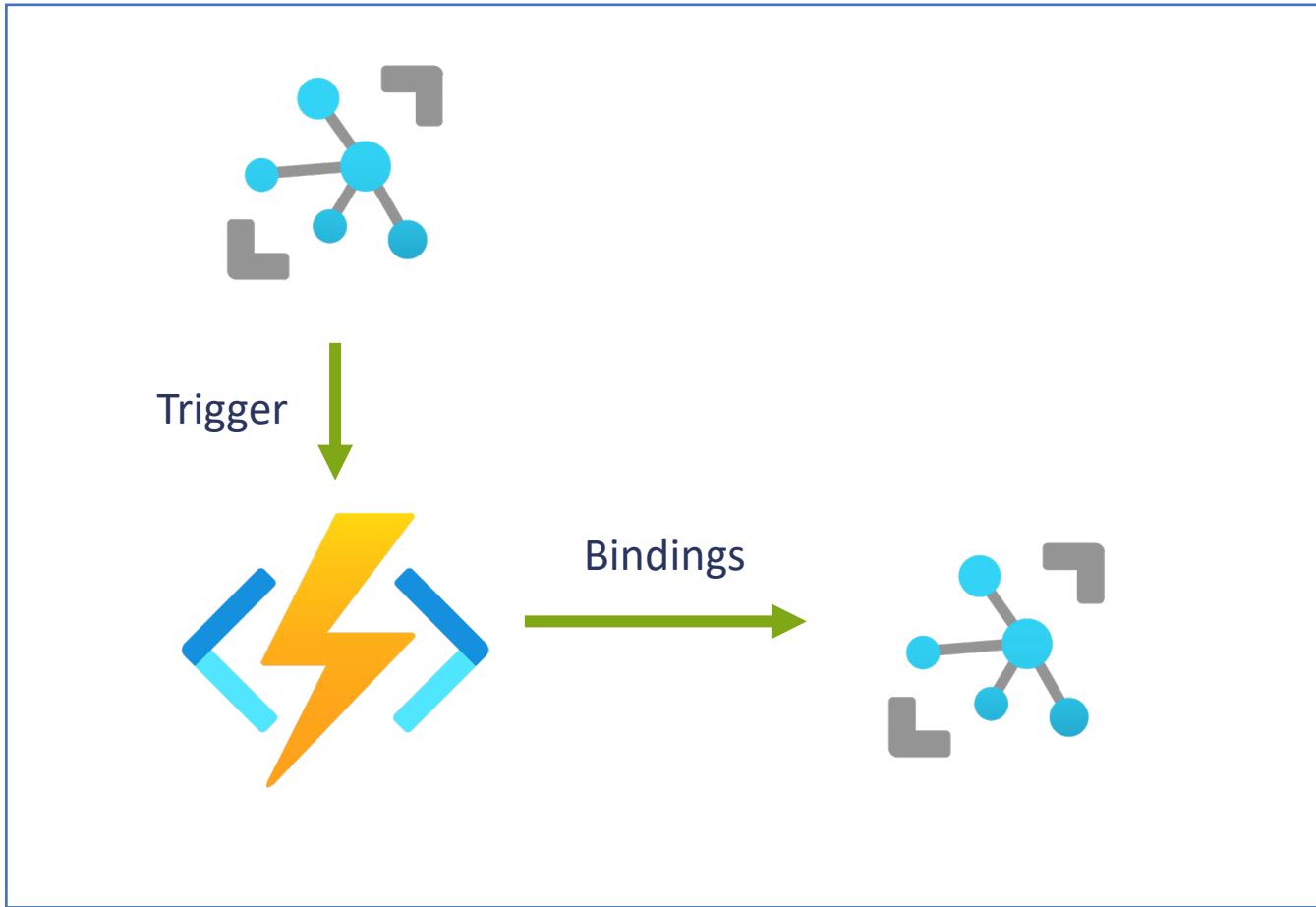
# Event Hub

Trigger	✓
Input	✗
Output	✓
Consumption	✓
Premium	✓
Dedicated	✓
C# In-Process	✓
C# Isolated	✓
Java	✓
JavaScript	✓
PowerShell	✓
Python	✓



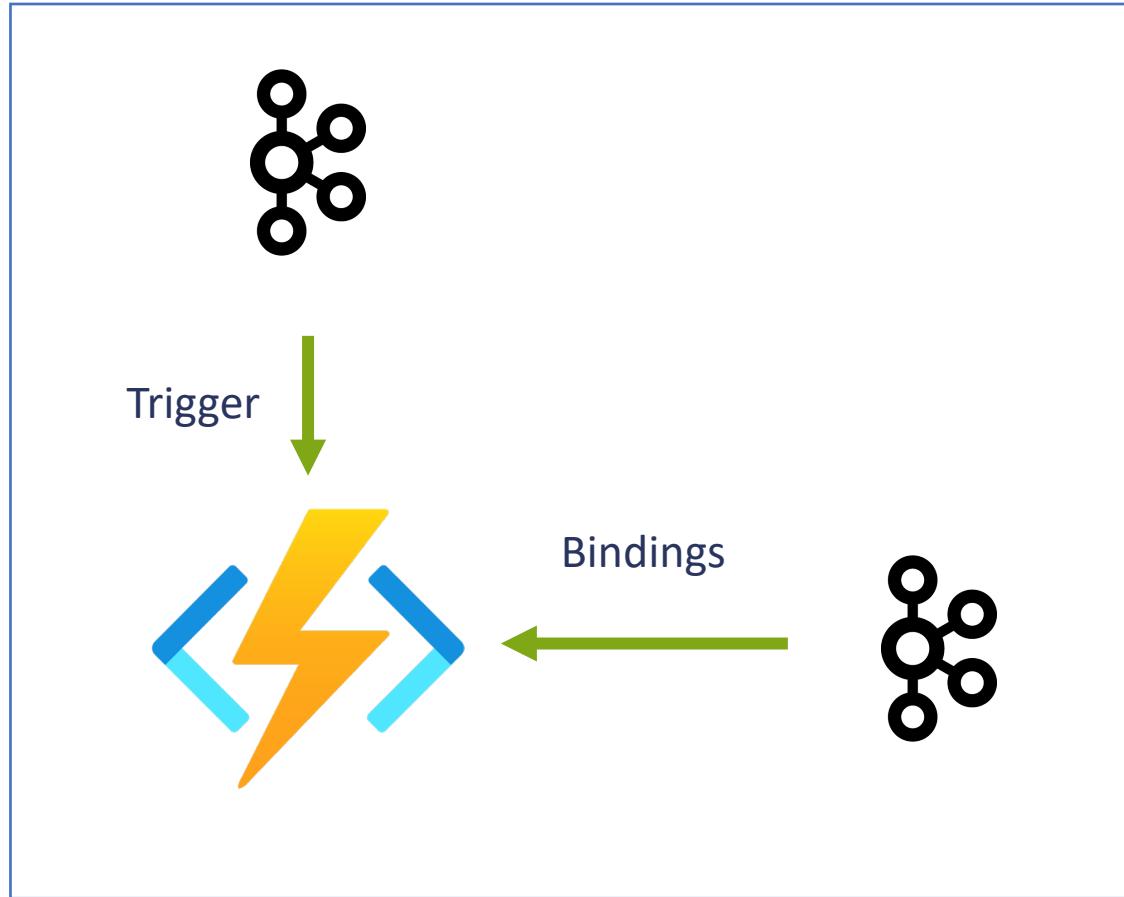
# IoT Hub

Trigger	✓
Input	✗
Output	✓
Consumption	✓
Premium	✓
Dedicated	✓
C# In-Process	✓
C# Isolated	✓
Java	✓
JavaScript	✓
PowerShell	✓
Python	✓



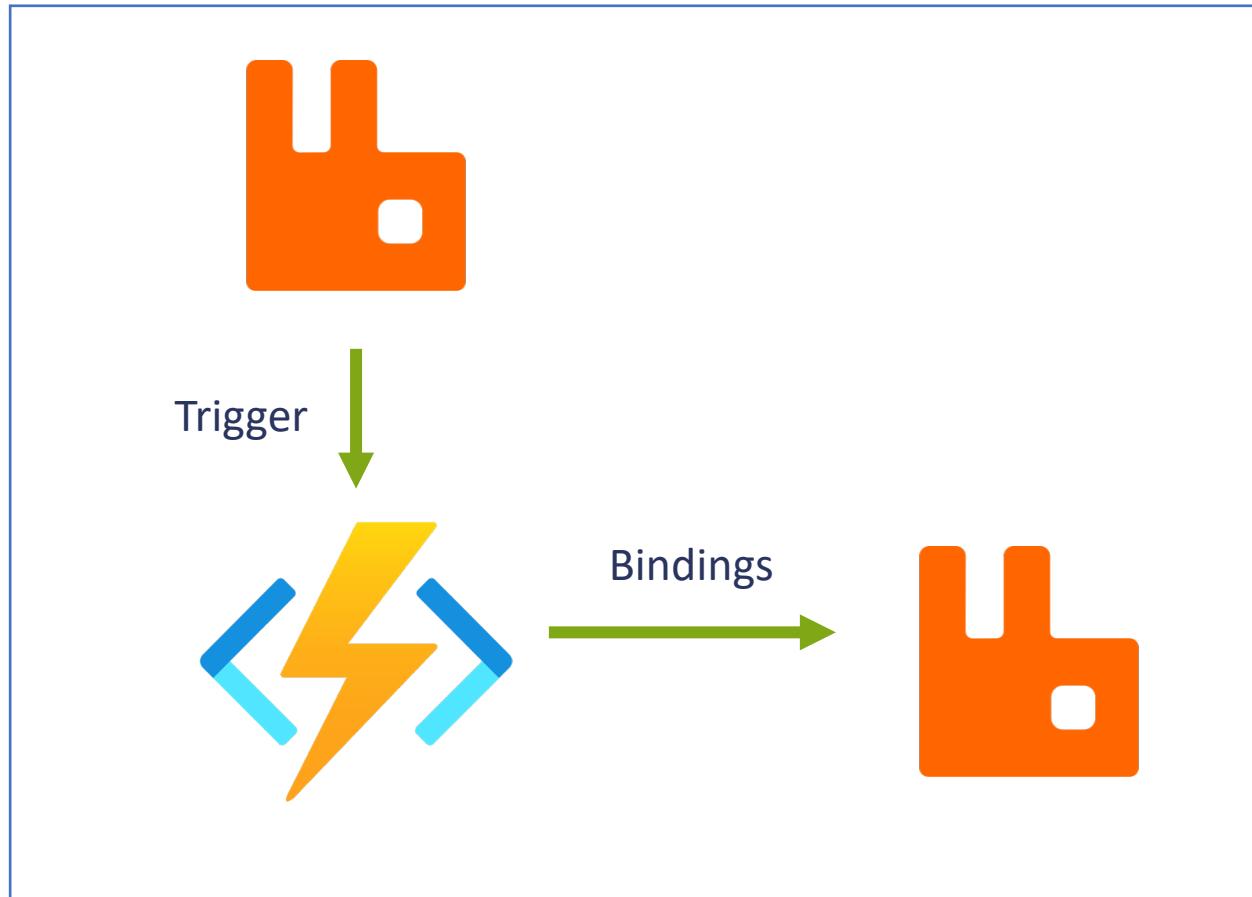
# Kafka

Trigger	✓
Input	✗
Output	✓
Consumption	✗
Premium	✓
Dedicated	✓
C# In-Process	✓
C# Isolated	✓
Java	✓
JavaScript	✓
PowerShell	✓
Python	✓



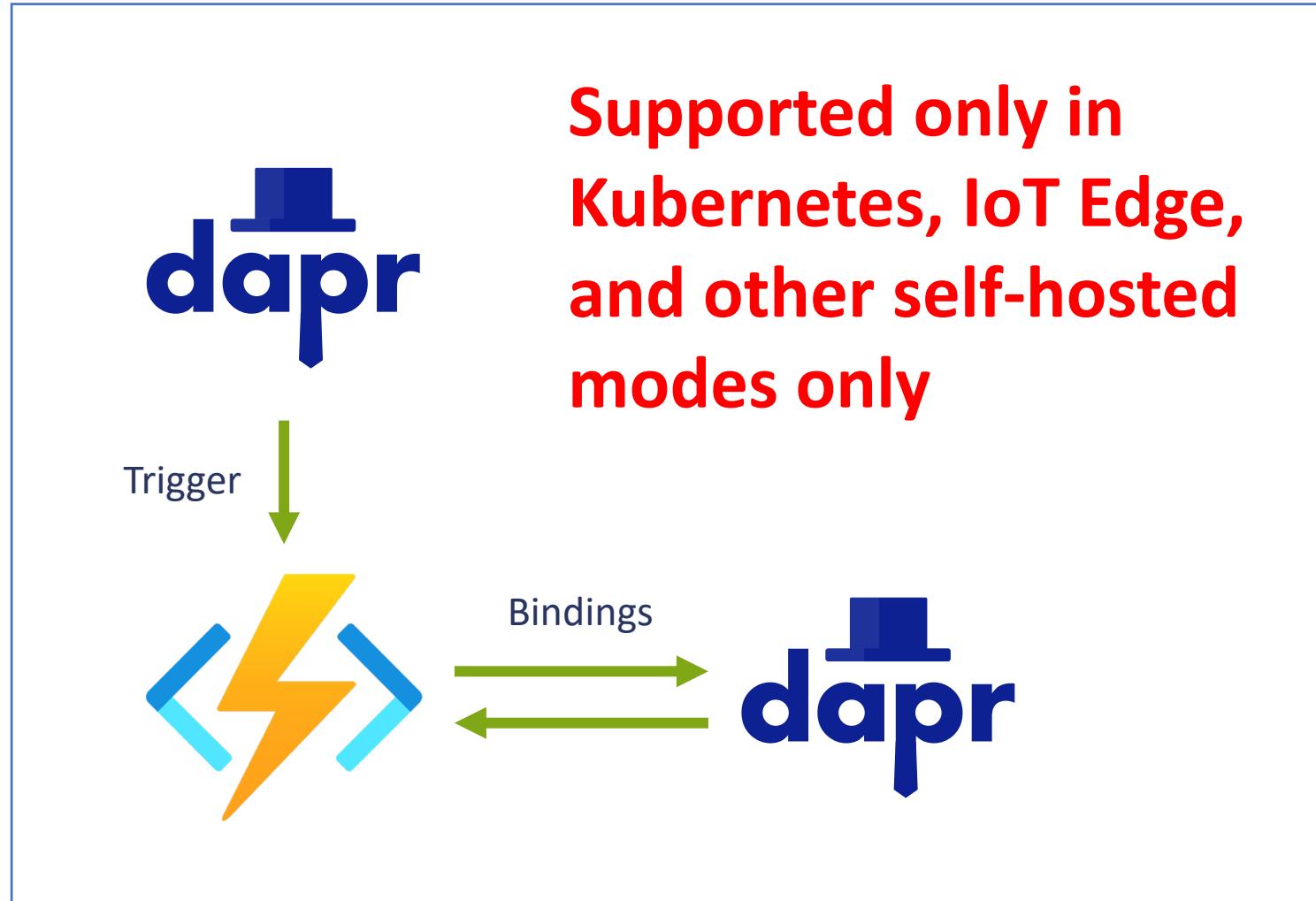
# RabbitMQ

Trigger	✓
Input	✗
Output	✓
Consumption	✗
Premium	✓
Dedicated	✓
C# In-Process	✓
C# Isolated	✓
Java	✓
JavaScript	✓
PowerShell	✓
Python	✓



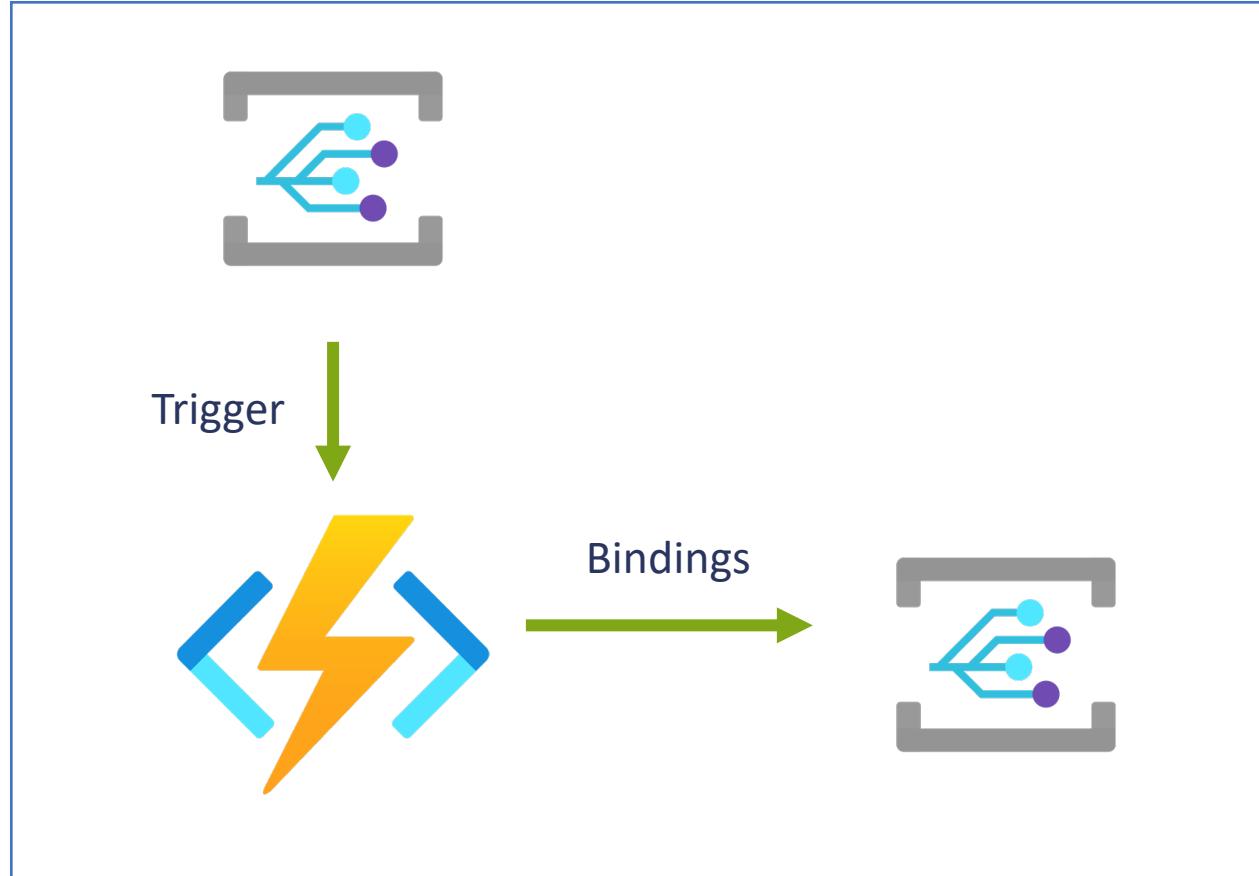
# Dapr

Tigger	✓
Input	✓
Output	✓
Consumption	✗
Premium	✗
Dedicated	✓
C# In-Process	✓
C# Isolated	✗
Java	✗
JavaScript	✓
PowerShell	✗
Python	✓



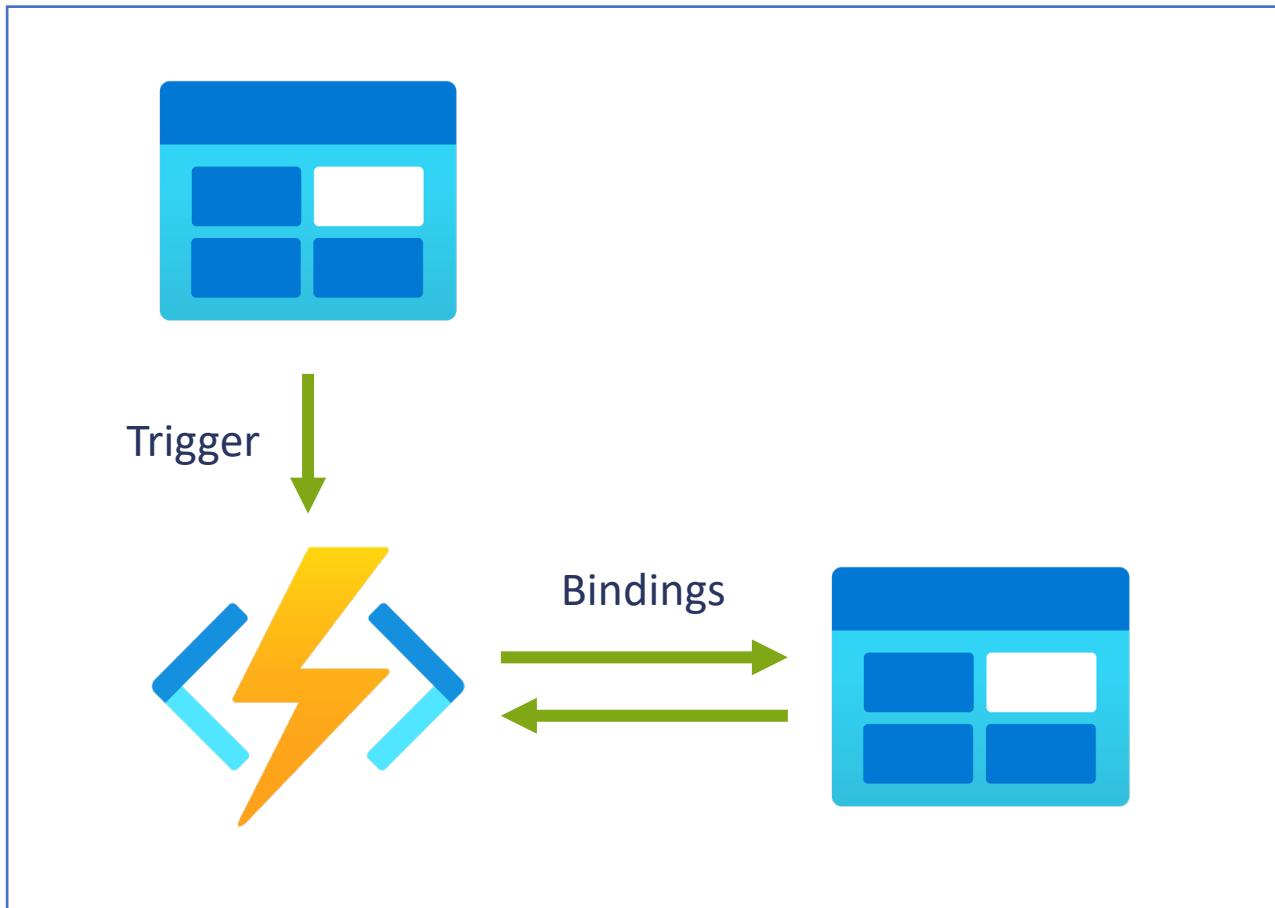
# Event Grid

Trigger	✓
Input	✗
Output	✓
Consumption	✓
Premium	✓
Dedicated	✓
C# In-Process	✓
C# Isolated	✓
Java	✓
JavaScript	✓
PowerShell	✓
Python	✓



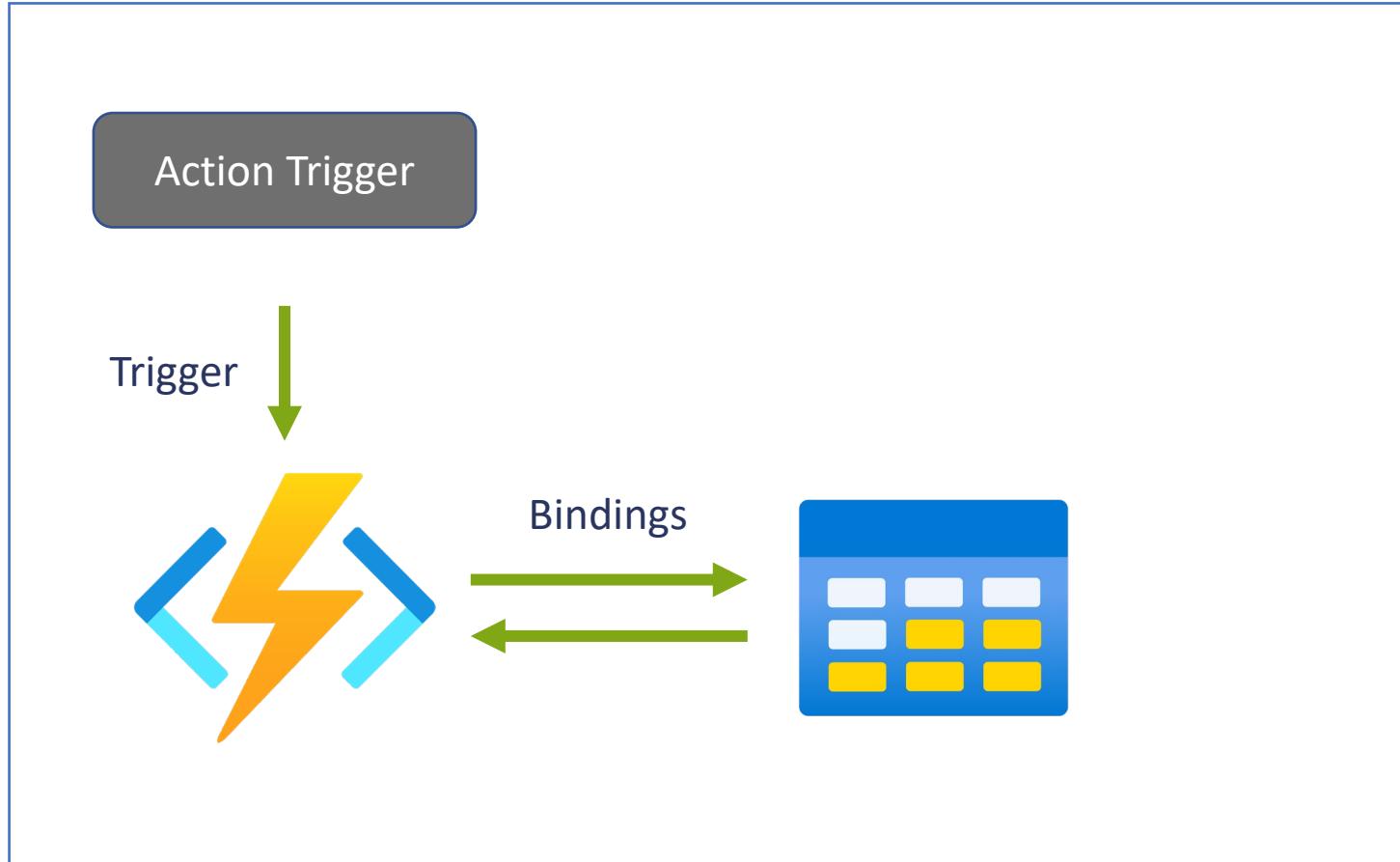
# Blob

Trigger	✓
Input	✓
Output	✓
Consumption	✓
Premium	✓
Dedicated	✓
C# In-Process	✓
C# Isolated	✓
Java	✓
JavaScript	✓
PowerShell	✓
Python	✓



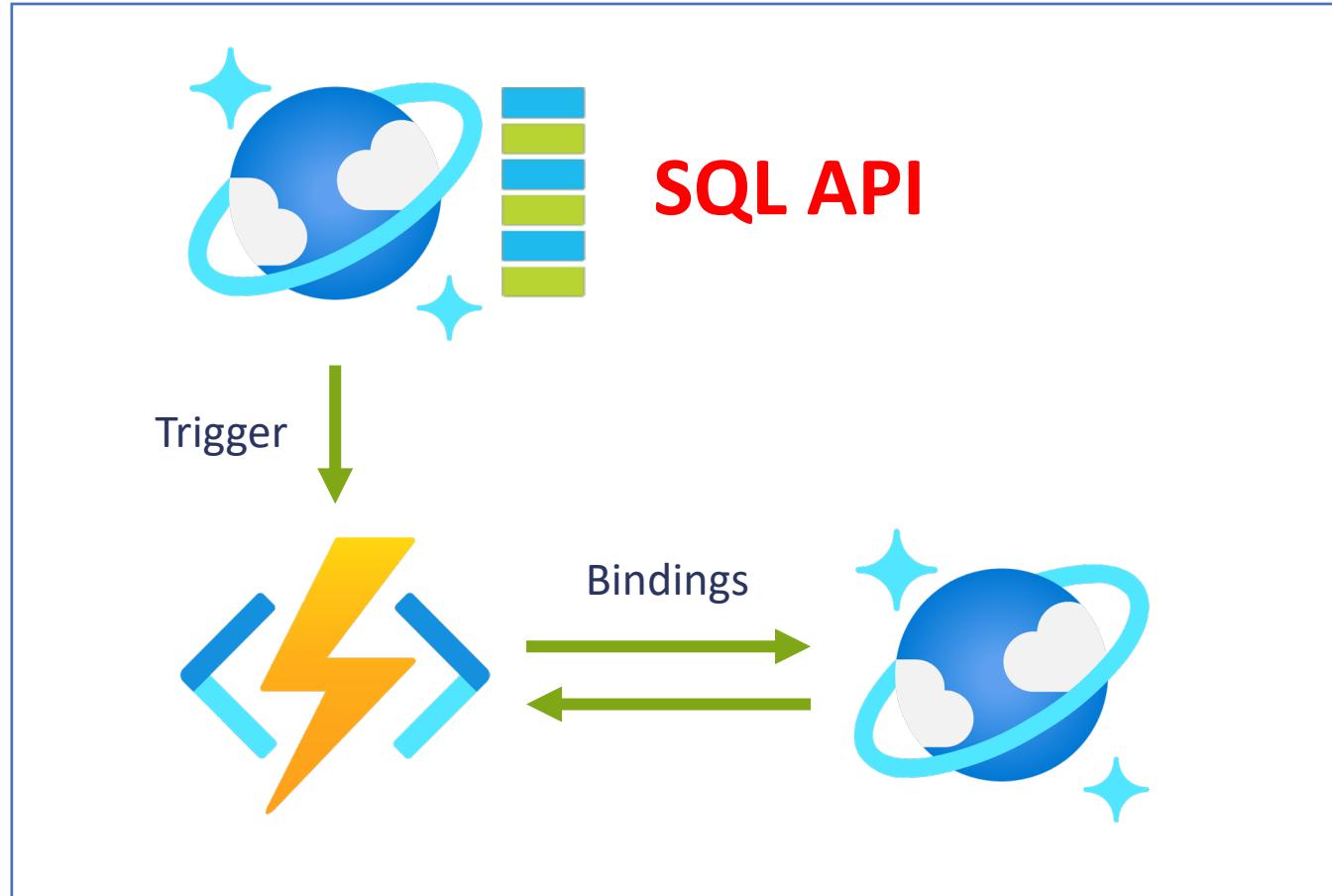
# Table Storage

Trigger	
Input	
Output	
Consumption	
Premium	
Dedicated	
C# In-Process	
C# Isolated	
Java	
JavaScript	
PowerShell	
Python	



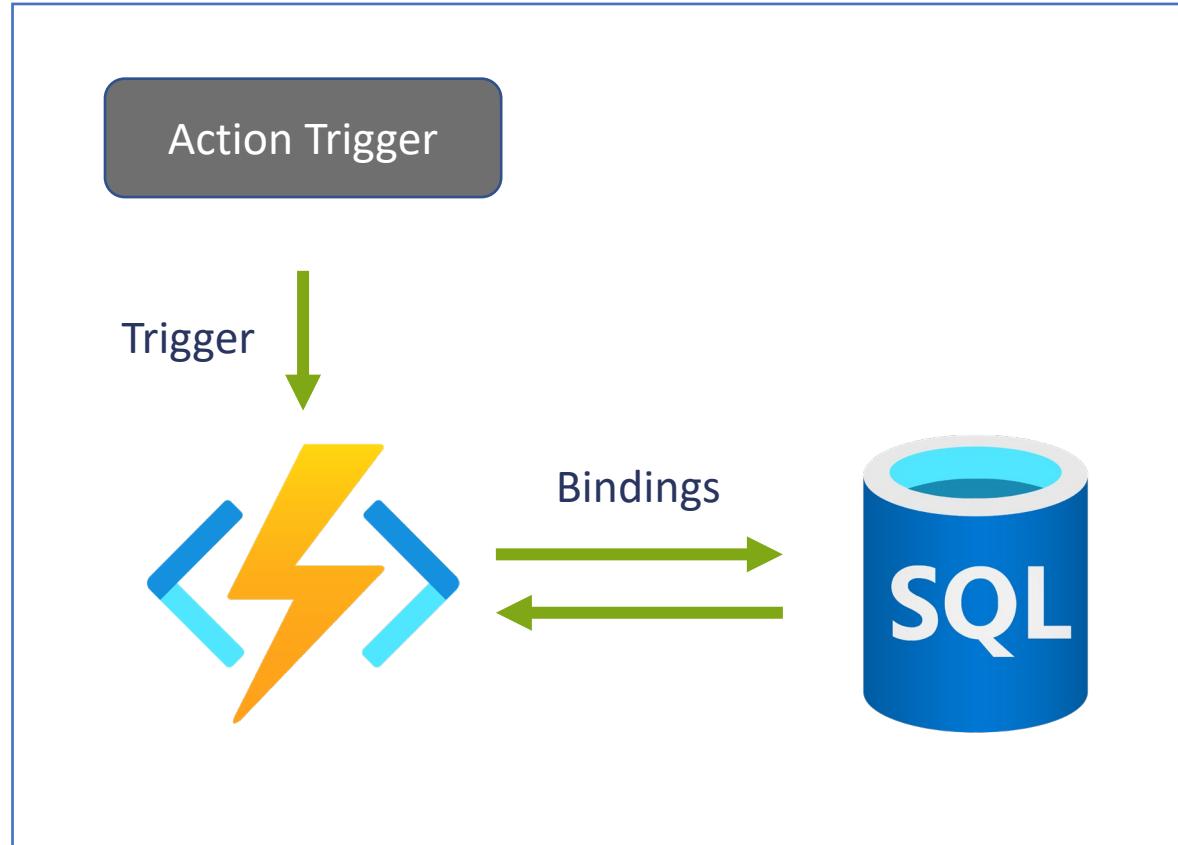
# Cosmos DB

Trigger	✓
Input	✓
Output	✓
Consumption	✓
Premium	✓
Dedicated	✓
C# In-Process	✓
C# Isolated	✓
Java	✓
JavaScript	✓
PowerShell	✓
Python	✓



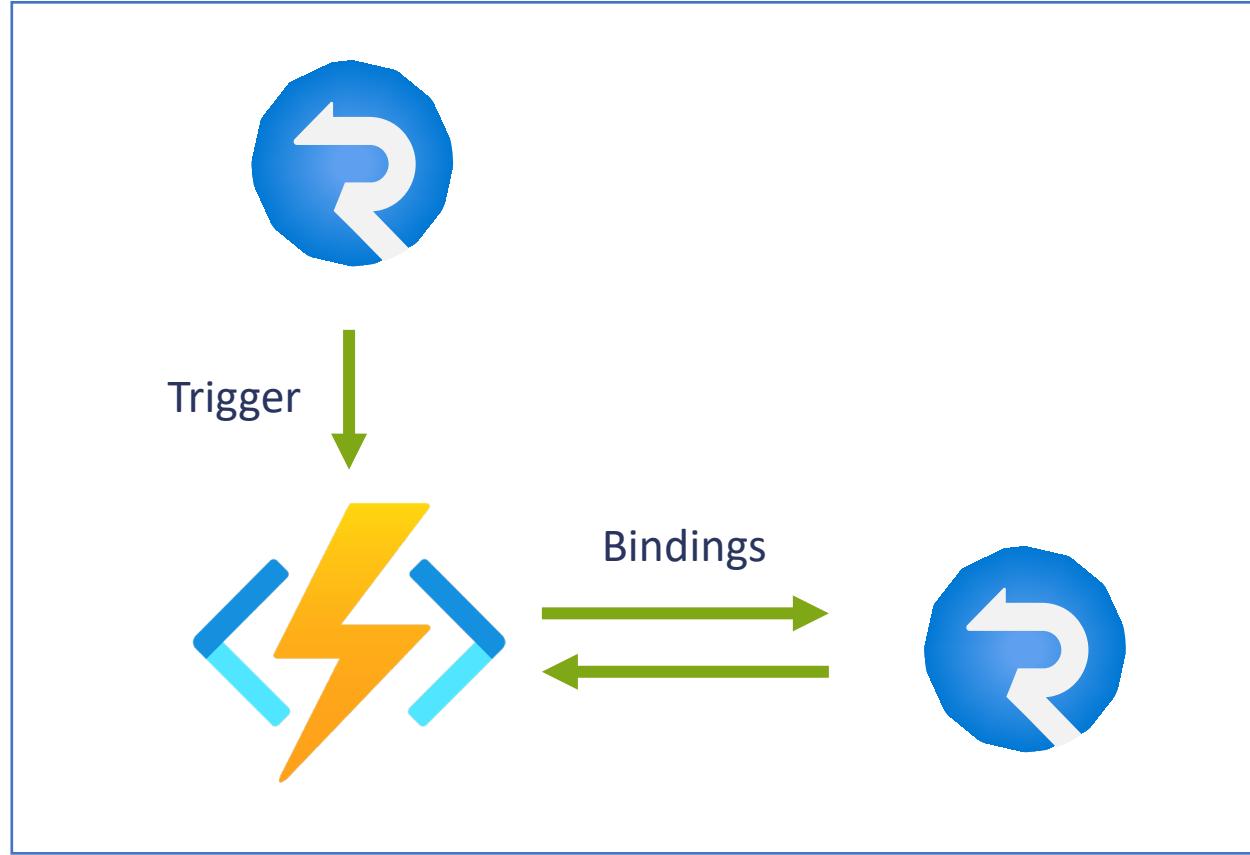
# Azure SQL

Trigger	✗
Input	✓
Output	✓
Consumption	✗
Premium	✓
Dedicated	✓
C# In-Process	✓
C# Isolated	✓
Java	✗
JavaScript	✓
PowerShell	✓
Python	✓



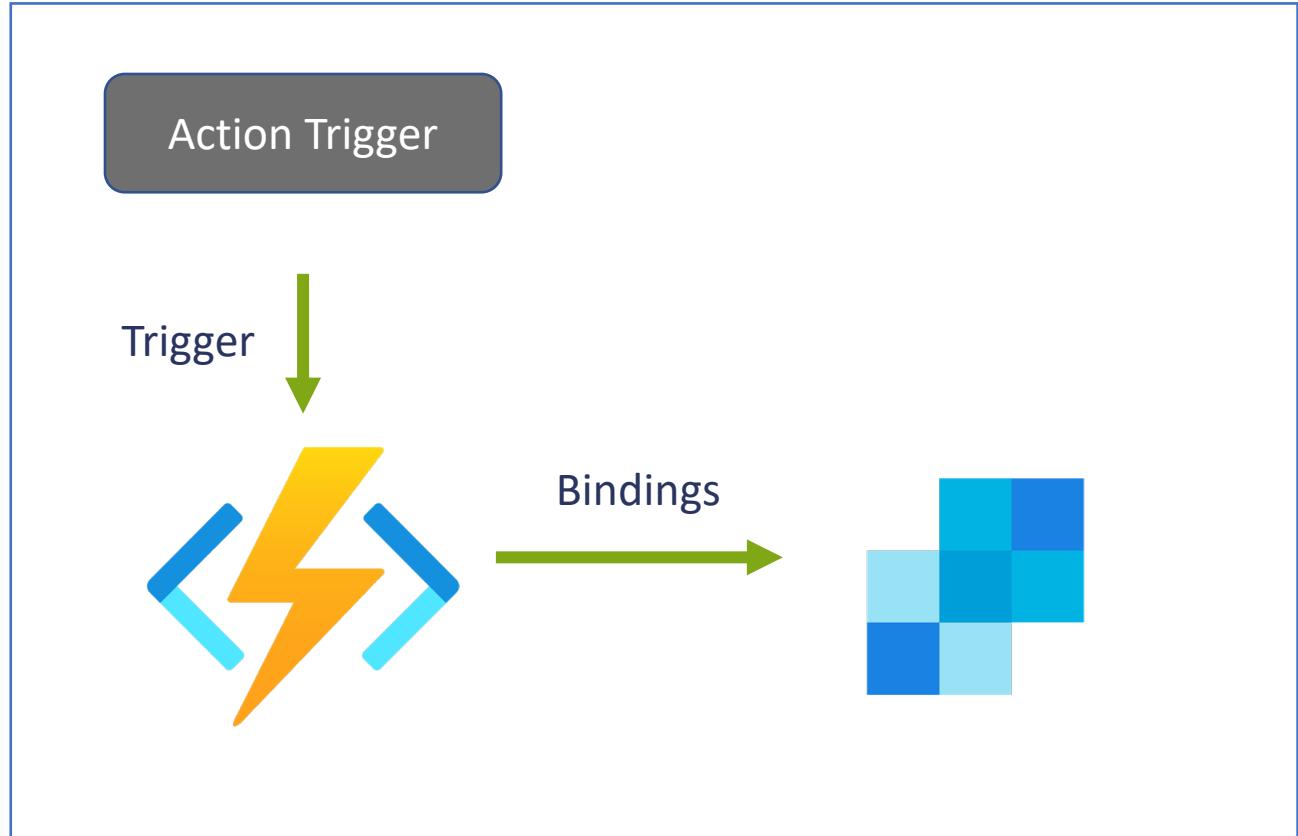
# SignalR

Trigger	✓
Input	✓
Output	✓
Consumption	✓
Premium	✓
Dedicated	✓
C# In-Process	✓
C# Isolated	✓
Java	✓
JavaScript	✓
PowerShell	✓
Python	✓



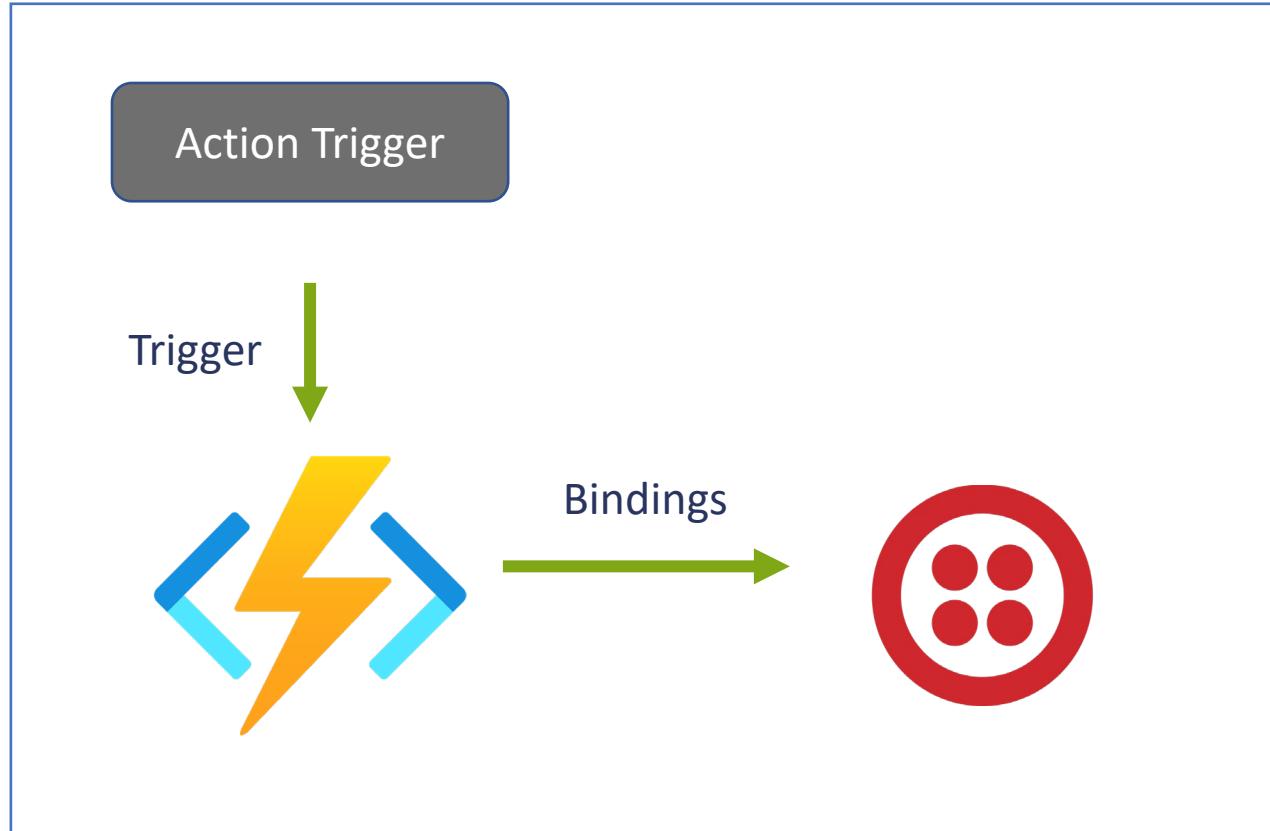
# SendGrid

Trigger	
Input	
Output	
Consumption	
Premium	
Dedicated	
C# In-Process	
C# Isolated	
Java	
JavaScript	
PowerShell	
Python	



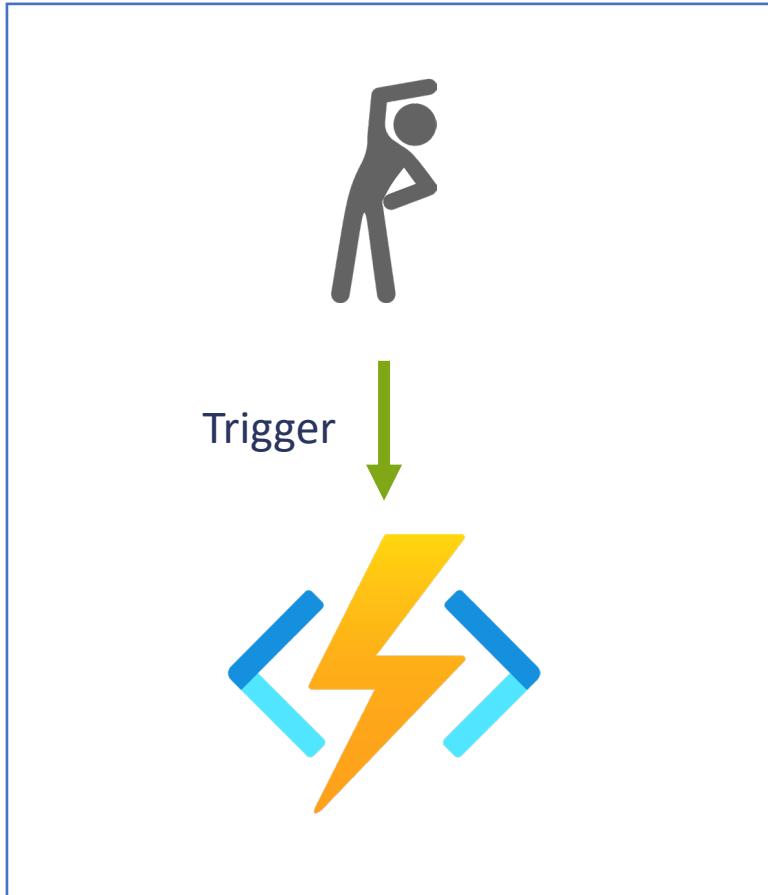
# Twilio

Trigger	✗
Input	✗
Output	✓
Consumption	✓
Premium	✓
Dedicated	✓
C# In-Process	✓
C# Isolated	✓
Java	✓
JavaScript	✓
PowerShell	✓
Python	✓

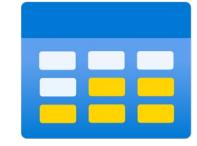
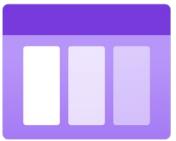
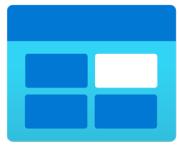


# Warmup

Trigger	✓
Input	✗
Output	✗
Consumption	✗
Premium	✓
Dedicated	✓
C# In-Process	✓
C# Isolated	✓
Java	✓
JavaScript	✓
PowerShell	✓
Python	✓



# Supported Bindings



Third Party Bindings

Custom Bindings

# Durable Functions

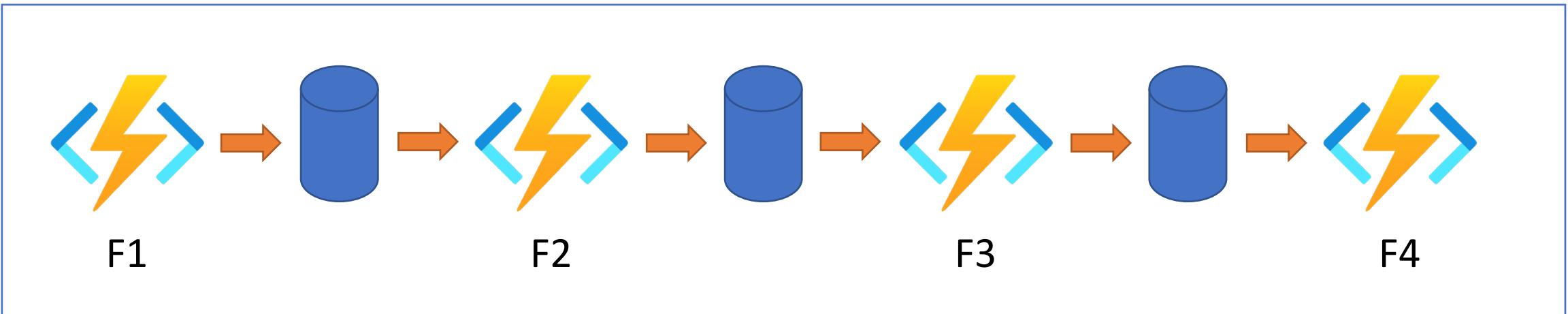
Beyond Hello World: Getting Deeper into Azure Functions

# Durable Functions

Stateful Federated Operations

# Durable Functions

## Patterns – Function Chaining



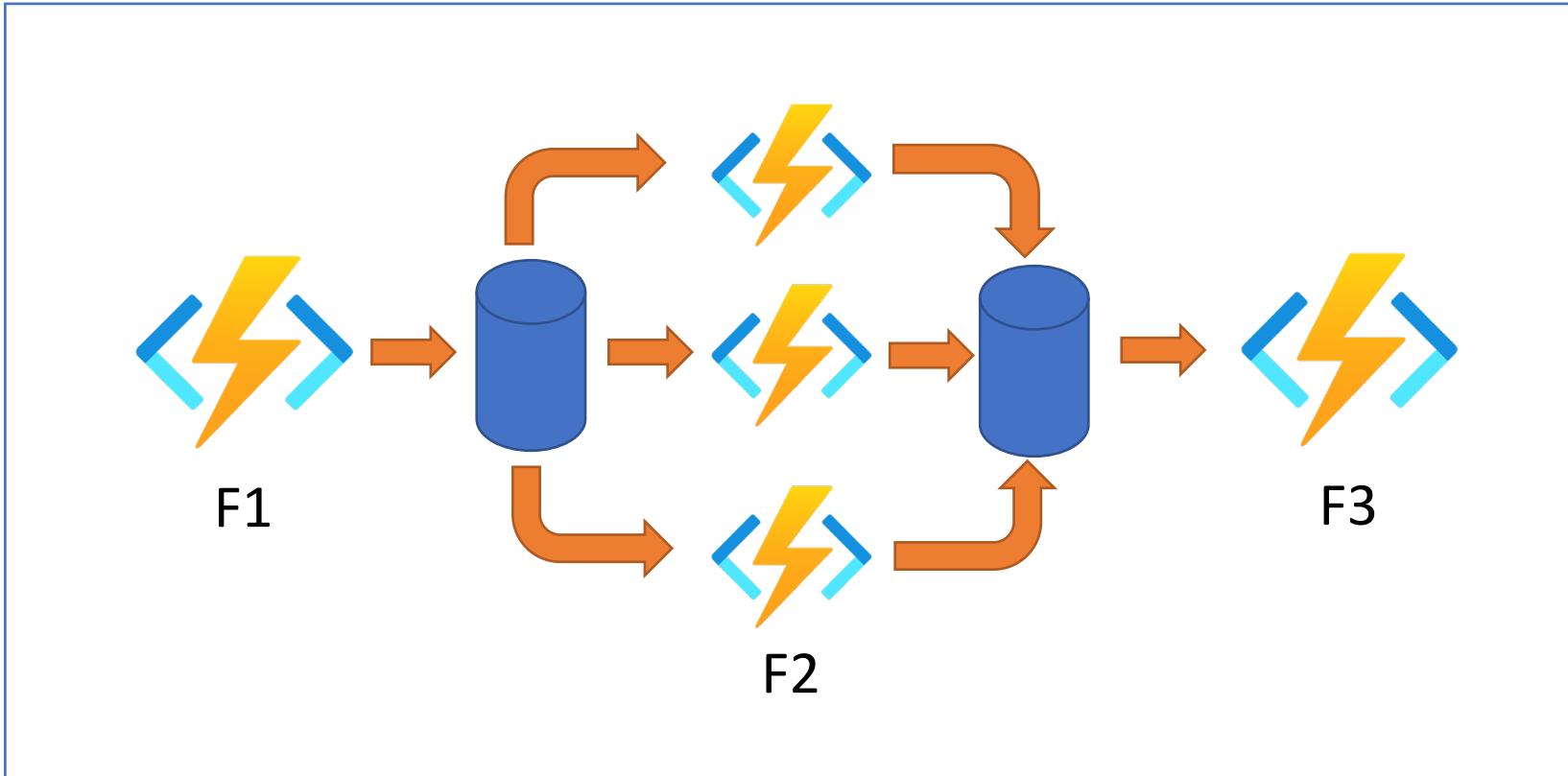
# Durable Functions

## Patterns – Function Chaining

```
[FunctionName("Chaining")]
public static async Task<object> Run(
    [OrchestrationTrigger] IDurableOrchestrationContext context)
{
    try
    {
        var x = await context.CallActivityAsync<object>("F1", null);
        var y = await context.CallActivityAsync<object>("F2", x);
        var z = await context.CallActivityAsync<object>("F3", y);
        return await context.CallActivityAsync<object>("F4", z);
    }
    catch (Exception)
    {
        // Error handling or compensation goes here.
    }
}
```

# Durable Functions

Patterns – Fan out/fan in



# Durable Functions

## Patterns – Fan out/fan in

```
[FunctionName("FanOutFanIn")]
public static async Task Run(
    [OrchestrationTrigger] IDurableOrchestrationContext context)
{
    var parallelTasks = new List<Task<int>>();

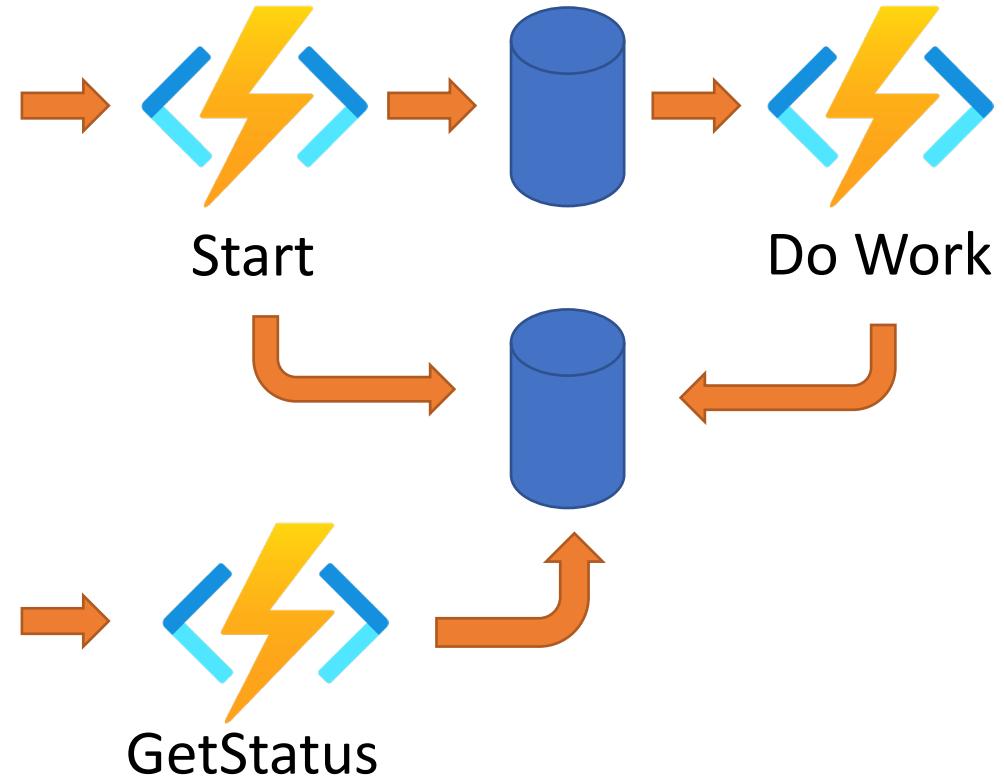
    // Get a list of N work items to process in parallel.
    object[] workBatch = await context.CallActivityAsync<object[]>("F1", null);
    for (int i = 0; i < workBatch.Length; i++)
    {
        Task<int> task = context.CallActivityAsync<int>("F2", workBatch[i]);
        parallelTasks.Add(task);
    }

    await Task.WhenAll(parallelTasks);

    // Aggregate all N outputs and send the result to F3.
    int sum = parallelTasks.Sum(t => t.Result);
    await context.CallActivityAsync("F3", sum);
}
```

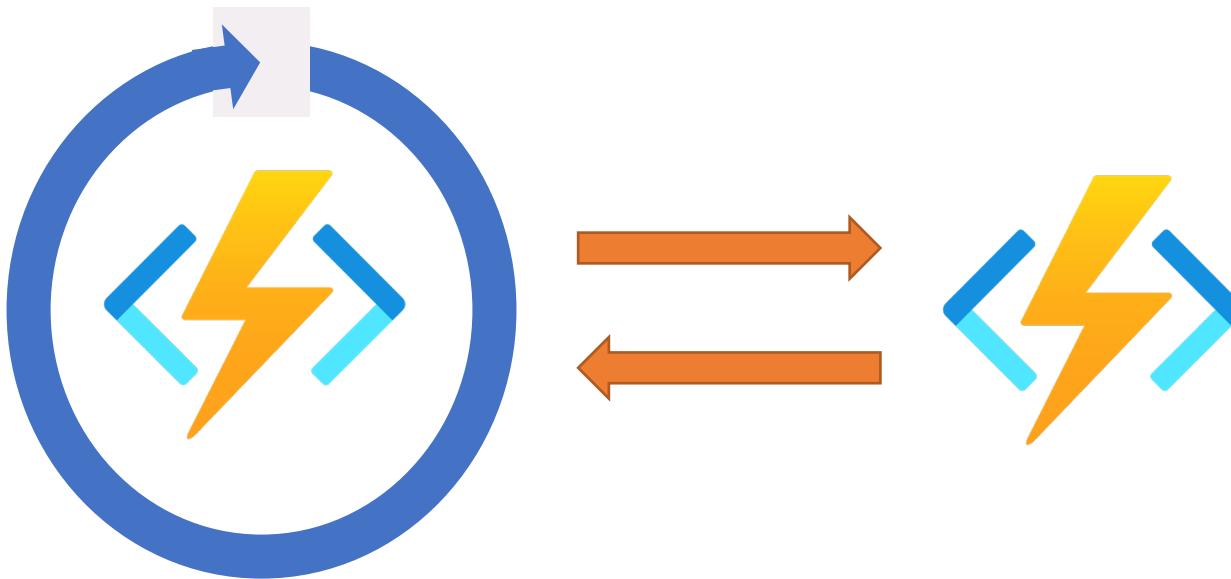
# Durable Functions

## Patterns – Async HTTP APIs



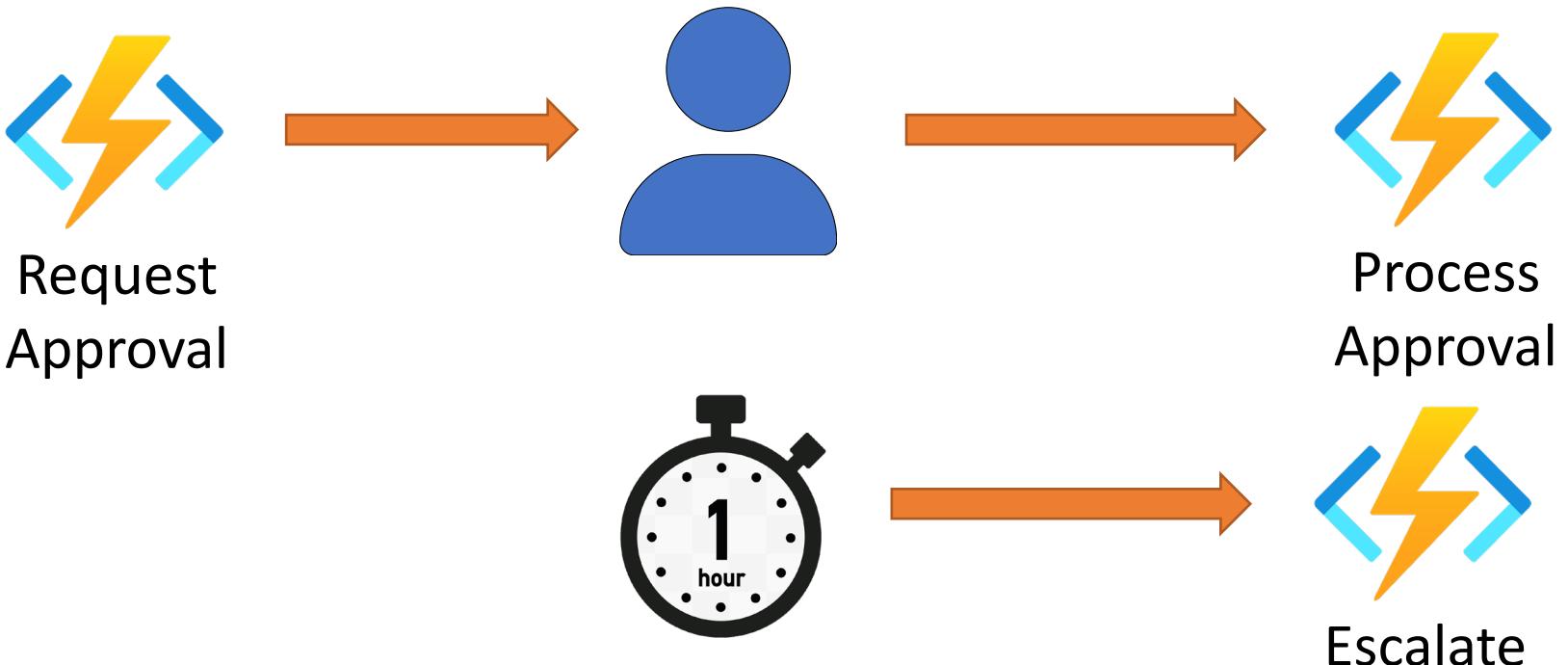
# Durable Functions

## Patterns – Monitor



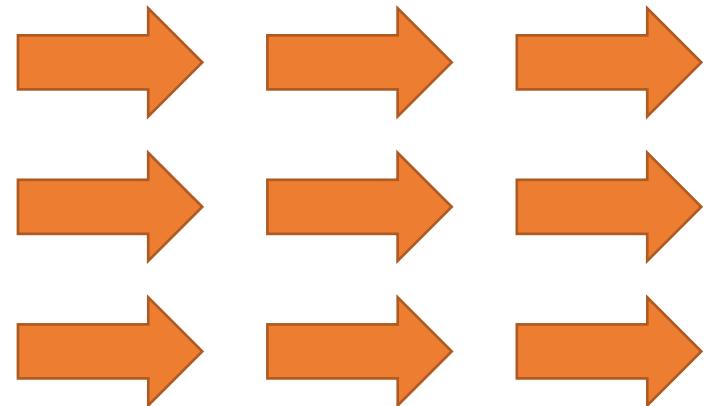
# Durable Functions

## Patterns – Human Interaction



# Durable Functions

Patterns – Aggregator (stateful entities)



# Summary

- Real-World Scenarios
- Azure Functions Triggers and Bindings
- Azure Durable Functions

# Thank You!

✉️ chadgreen@chadgreen.com

✳️ TaleLearnCode

🌐 ChadGreen.com

🐦 ChadGreen & TaleLearnCode

linkedin ChadwickEGreen

