

Azure Functions Overview

Kyle McMaster

@KyleMcMaster

kyle.mcmaster@nimblepros.com | NimblePros.com





What are Azure Functions?

- Event-driven, serverless compute platform
- Supports functions, small pieces of code that do single things
- Develop simple deployables for technology-specific microservices

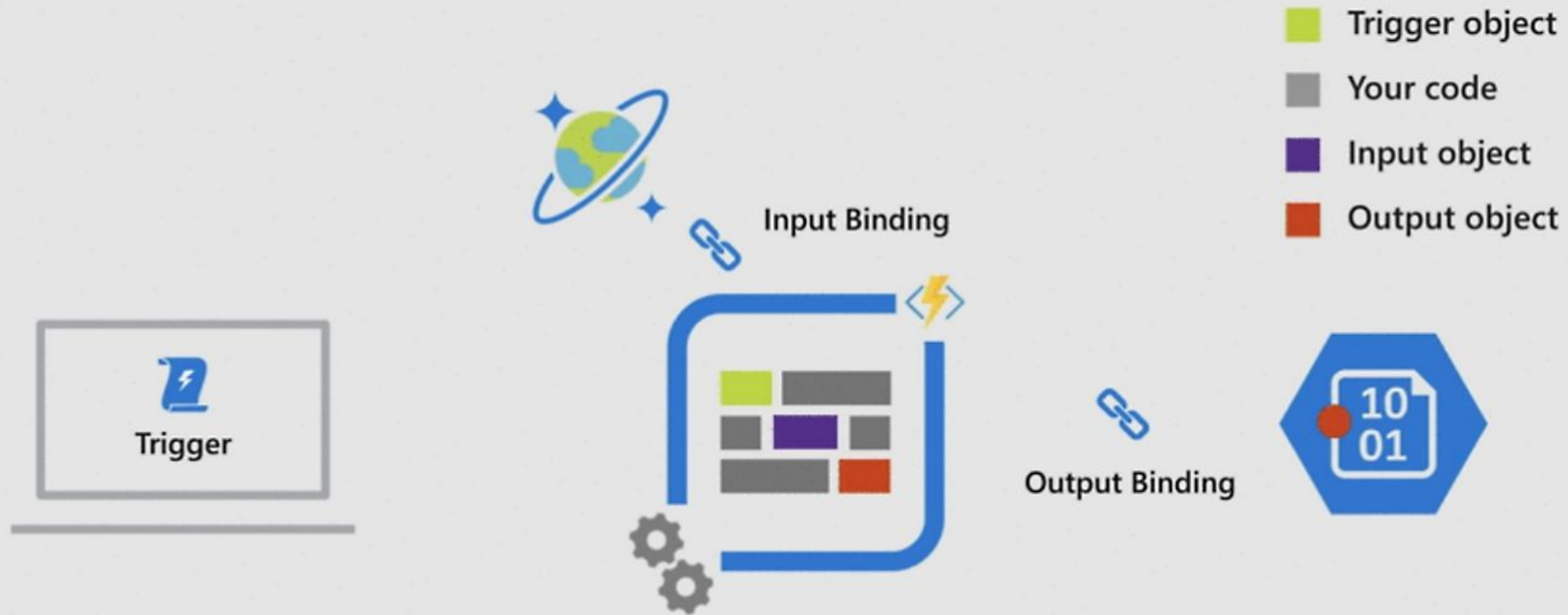


Triggering Azure Functions

- Timer Triggers
- HTTP Triggers
- Cosmos DB Triggers
- Events from Event Hub, Event Grid, or Azure Service Bus
- SignalR and Notification Hub messages
- Blob Storage Triggers



Azure Functions Bindings





Use Cases

- Respond to HTTP Requests
- Process file uploads from Azure Blob Storage
- Process messages from Azure Storage Queues
- Run a process on a scheduled interval
- Create serverless workflows With Durable Functions



Runtimes and Models

Azure Functions are built on top of the Azure Web Jobs SDK

Supported Runtimes

- v1 (.NET Full Framework)
- v4 (Current, .NET 6+) ★

Models

- In Process
- Isolated Process



Hosting and Scaling

- Consumption Plan
 - Scale out, configurable timeout options
- Premium Plan
 - Avoid cold starts, longer timeouts, more resources, consistency
- Dedicated Plan
 - App Service
- Azure Container Apps (Preview) 🌟



Resources

Bindings

- <https://learn.microsoft.com/en-us/azure/azure-functions/functions-triggers-bindings?tabs=isolated-process%2Cpython-v2&pivots=programming-language-csharp#supported-bindings>

Samples

- <https://learn.microsoft.com/en-us/azure/azure-functions/functions-dotnet-class-library?tabs=v4%2Ccmd>

Use cases

- <https://learn.microsoft.com/en-us/azure/azure-functions/functions-scenarios?pivots=programming-language-csharp>

Hosting plans

- <https://learn.microsoft.com/en-us/azure/azure-functions/functions-scale#overview-of-plans>

Durable Orchestrations

Now we are going to talk about this...





Durable Orchestrations

- Asynchronous workflow orchestration for Azure Functions
- Enables stateful Functions in serverless compute environment
- Built on Durable Task Framework and Web Jobs SDK
- Error Handling, Retries, Fan In/out, concepts baked in



Samples

Backup site content

- https://learn.microsoft.com/en-us/azure/azure-functions/durable/durable-functions-cloud-backup?WT.mc_id=dotnet-sample-marouill&tabs=csharp#e2_backupsitecontent-orchestrator-function

Waiting for external user input

- <https://learn.microsoft.com/en-us/azure/azure-functions/durable/durable-functions-phone-verification?tabs=csharp>



Considerations

- Orchestrations can be replayed
- Orchestrators should be idempotent
- Activities can contain async or state dependent mutations
- State is serialized and persisted to Azure Storage Queues/Tables
- Implications of long running “stateful” processes

<https://learn.microsoft.com/en-us/azure/azure-functions/functions-overview?pivots=programming-language-csharp>



Summary

-



Extra Fun

Functional Testing

<https://github.com/KyleMcMaster/azure-function-functional-tests>