WebJobs & Azure Functions in modern and Serverless applications

Paris Polyzos
Software Engineer at ZuluTrade Inc
Microsoft Azure MVP

AzureHeads

The ZuluTrade Group

AAAFx

skufler

KYON

Paris Polyzos
Senior Software Engineer
Microsoft Azure MVP

@ppolyzos
ppolyzos.com
AzureHeads.gr

Microsoft CERTIFIED Professional

MVP Microsoft Most Valuable Professional

Goals

AzureHeads

- What, when and how to use the “Functions” programming model
- WebJobs Overview
- Serverless Computing
- Azure Functions Overview

What is the Functions programming model?

AzureHeads

- Function as the unit of work
- Functions start, execute and finish
- Functions have inputs and outputs

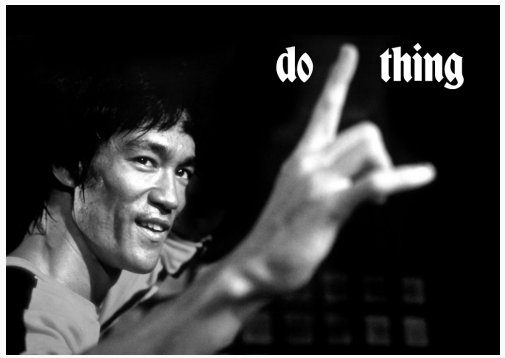
```
public async static Task ProcessQueueMessageAsyncCancellationToken(
    [QueueTrigger("blobcopyqueue")] string blobName,
    [Blob("textblobs/{queueTrigger}",FileAccess.Read)] Stream blobInput,
    [Blob("textblobs/{queueTrigger}-new",FileAccess.Write)] Stream blobOutput,
    CancellationToken token)
{
    await blobInput.CopyToAsync(blobOutput, 4096, token);
}
```

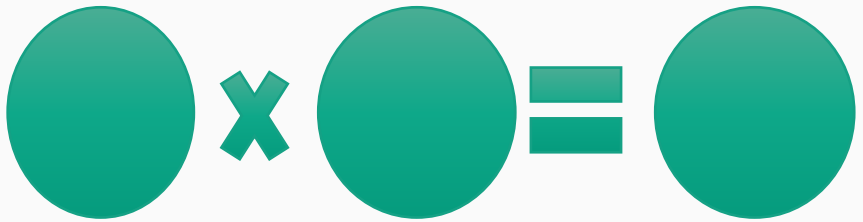
Best Practices

AzureHeads

Functions *should:*

- do one thing
- be idempotent
- finish as quickly as possible





How and when to use them?

AzureHeads

Questions:

1. What is that you want to do?
2. What triggers the function?
3. Do I need additional data?
4. Is there output I should produce?

Conclude:

“When __, get __, do __, and output __”.

Azure WebJobs

AzureHeads

"Azure WebJobs provide an easy way to run scripts or programs, on demand, continuously or on a schedule, as background processes on App Service Web Apps."

WebJobs SDK provide bindings and trigger system which works with Storage Blobs, Queues, Tables, Service Bus.

WebJobs Extensions

WebJobs Demo

Serverless Computing

AzureHeads

“Serverless architectures refer to applications that significantly depend on third-party services (knows as Backend as a Service or "BaaS") or on custom code that's run in ephemeral containers (Function as a Service or "FaaS")”

Martin Fowler

Benefits

- Reduced Operational & Development Costs
- Fast and easy Scaling
- Reduced Packaging & Deployment Complexity
- Easy scheduling & event processing
- Reduced time to market / Experimentation

Drawbacks

- Vendor control
- Execution Duration
- Unit Testing
- Limited Tooling

Azure Functions

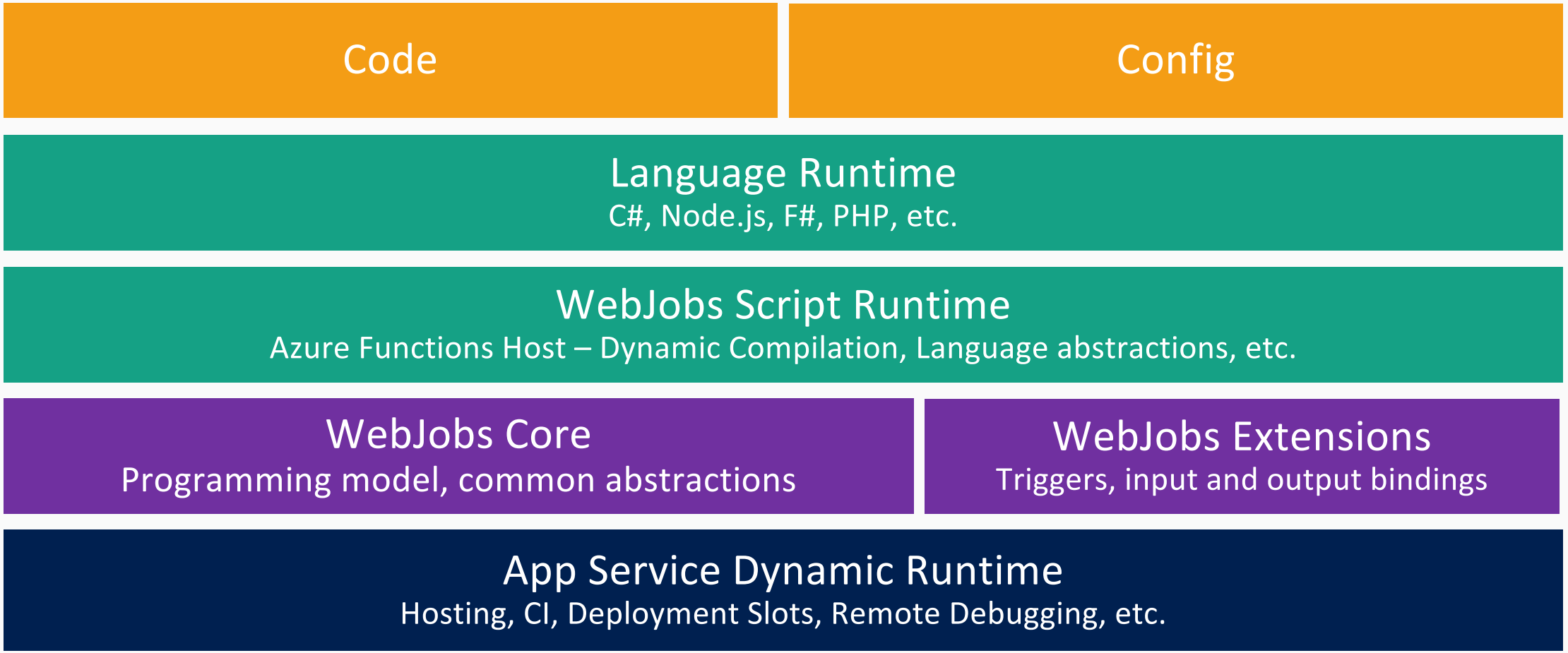
Process events with serverless code

- Develop in any language C#, node.js, F#, Python…
- Schedule event-driven tasks easily
- Expose Functions as Http API endpoints
- Pay only for what you use
- Scale Functions based on demand

Azure Functions Architecture

AzureHeads

Built on top of App Service & WebJobs SDK



Azure Functions Platform & Pricing

AzureHeads

Dedicated & Dynamic Tiers

Dedicated

- Basic, Standard, Premium App Service Tiers
- Pay based on # of **reserved VMs**
- **You** are responsible for scaling

Dynamic

- Pay based on # of **executions**
- **Platform** responsible for scaling
- Limits

Pricing

(Reserved Memory × Duration) + Number Of Executions = xxx (GB-s)

Execution Time	=	(1,536/1,024	*	2,000,000 - 400,000)	*	€0.000007	=	€17.54
		Function Memory Size converted to GB's		Total Execution time – 400,000 Free Grant		Regional Cost		Monthly Execution Time Cost
Requests	=	(2,000,000-1,000,000)	/	1,000,000	*	€0.16866	=	€0.17
		Total Execution count minus free grant				Price per million Executions		Monthly Execution Cost

Azure Functions

WebJobs vs Azure Functions

AzureHeads

Common Points

- Same “Functions” Programming model
- Support bindings for triggers/inputs/outputs
- WebJobs SDK extensions model
- External libraries support
- Can run locally and be debugged
- Both provide runtime telemetry via Dashboard

WebJobs vs Azure Functions

AzureHeads

Programming model differences

WebJobs

- Attributes for configuring bindings
- Traditional .NET developer experience (Visual Studio, NuGet, MSBuild)
- Many functions per class
- Can access and manipulate many core SDK features
- Can’t listen for HTTP requests*

Azure Functions

- C#, Node.js, F#, Python
- Config files for bindings
- Diverse development (Azure Portal, VS Code)
- Supports HTTP

WebJobs vs Azure Functions

AzureHeads

Hosting model differences

WebJobs

- Supports Host configuration
- Build a console app
- **You** manage scaling
- Runs the service in the background of Web/Mobile/API app
- Runs any console app (not just SDK based ones)

Azure Functions

- Limited control over the host
- Just give it your code/config
- Function app owns the whole host
- Scale is managed **for you**
- Only runs Azure Functions stuff – no other things

AzureHeads

Thank You

https://azure.microsoft.com/en-us/documentation/articles/websites-webjobs-resources/
https://azure.microsoft.com/en-gb/services/functions/
www.azureheads.gr
www.ppolyzos.com

Supported Bindings

AzureHeads

Type	Service	Trigger	Input	Output
Schedule	Azure Functions	✔		
HTTP (REST or WebHook)	Azure Functions	✔		✔
Blob Storage	Azure Storage	✔	✔	✔
Queues	Azure Storage	✔		✔
Tables	Azure Storage		✔	✔
Tables	Azure Mobile Apps Easy Tables		✔	✔
No-SQL DB	Azure DocumentDB		✔	✔
Streams	Azure EventHubs	✔		✔
Push Notifications	Azure Notification Hubs			✔
SaaS	Twillio, SendGrid (experimental)			✔