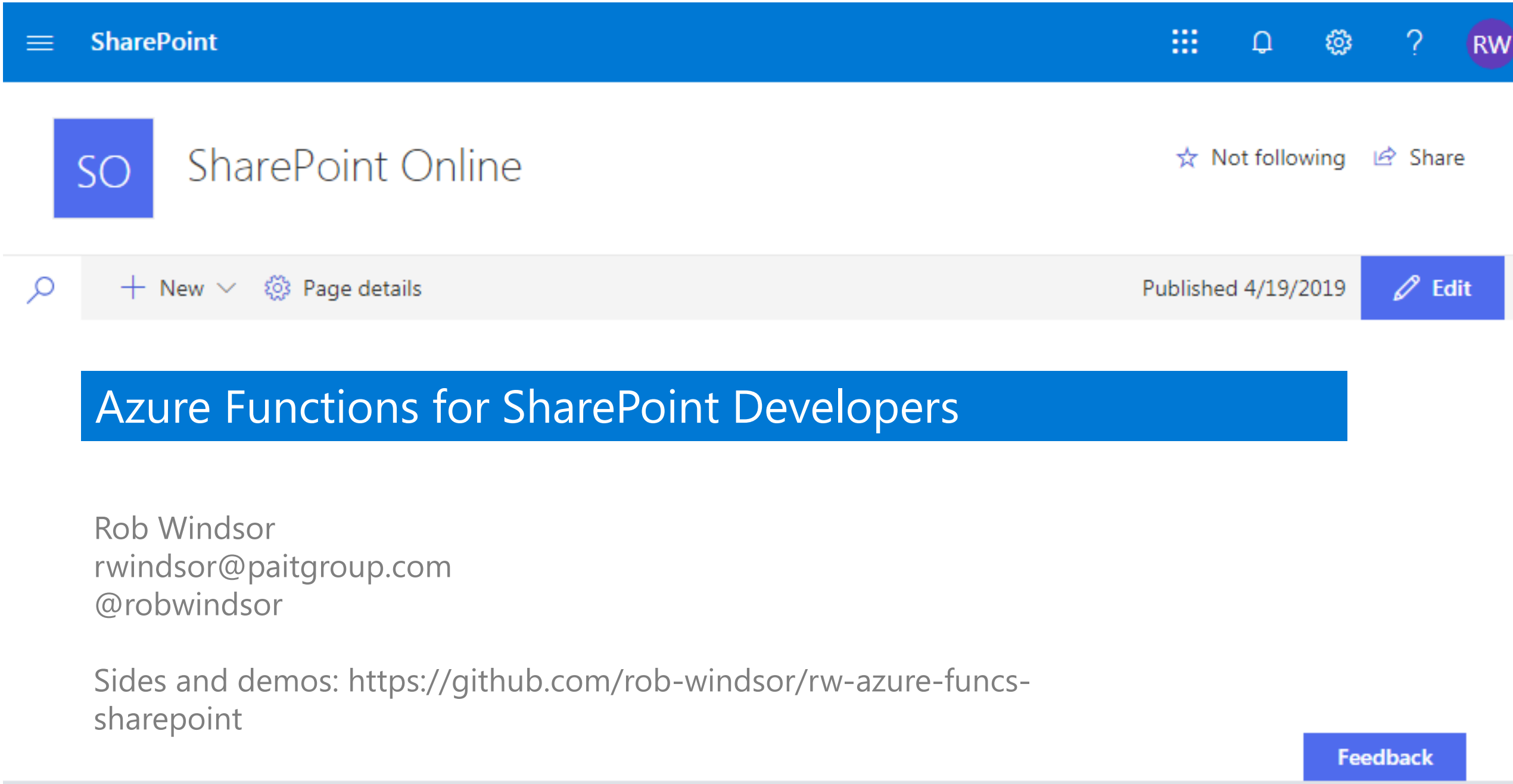
SharePoint
RW
SO
SharePoint Online
Not following
Share
New
Page details
Published 4/19/2019
Edit
Azure Functions for SharePoint Developers
Rob Windsor
rwindsor@paitgroup.com
@robwindsor
Sides and demos: https://github.com/rob-windsor/rw-azure-funcs-sharepoint
Feedback

THANK YOU SPONSORS!

Diamond Sponsors

backupify
a datto company

tyGraph

Microsoft

AvePoint

Platinum Sponsors

Quest

NiCE
IT Management Solutions
www.nice.de

SC

KnowledgeLake
Intelligent Content Automation

WEBCON
LOW-CODE, BUT BETTER.

Gold Sponsors

MARTELLO

ScriptRunner®
The #1 for PowerShell Management

Hosted by:

PAIT GROUP

Be sure to visit each sponsor's channel in Microsoft Teams for prizes and extra entries in our big raffle prize drawing Wednesday!

Join us for our end of day reception Wednesday starting at 3:45 in Taneycomo A for free food, drinks and raffle prizes!

NORTH AMERICAN COLLABORATION SUMMIT
2021
BRANSON MISSOURI

*OPEN BAR
*Beer, wine, & soda

WIN PRIZES
WIN PRIZES
WIN PRIZES

About Me

- Lead SharePoint Consultant at PAIT Group
- Microsoft MVP, MCPD, MCT Alumni
- Founder and Past-president of the North Toronto .NET UG

Azure Functions Overview

Azure Functions is a serverless solution that allows you to write less code, maintain less infrastructure, and save on costs. Instead of worrying about deploying and maintaining servers, the cloud infrastructure provides all the up-to-date resources needed to keep your applications running.

You focus on the pieces of code that matter most to you, and Azure Functions handles the rest.

Azure Function and Function Apps

- Functions contained in a Function App
- Function configuration
  - Trigger
  - Bindings
- Function app configuration
  - Programming language
  - Runtime version
  - Hosting plan
  - Authentication

Triggers

Azure Functions support a wide range of trigger types. Here are some of the most common types:

Type	Purpose
Timer	Execute a function at a set interval.
HTTP	Execute a function when an HTTP request is received.
Blob	Execute a function when a file is uploaded or updated in Azure Blob storage.
Queue	Execute a function when a message is added to an Azure Storage queue.
Azure Cosmos DB	Execute a function when a document changes in a collection.
Event Hub	Execute a function when an event hub receives a new event.

https://docs.microsoft.com/en-us/learn/modules/execute-azure-function-with-triggers/2-determine-best-trigger

Bindings

This table shows the bindings that are supported in the major versions of the Azure Functions runtime:

Type	1.x	2.x and higher[1]	Trigger	Input	Output
Blob storage	✓	✓	✓	✓	✓
Azure Cosmos DB	✓	✓	✓	✓	✓
Event Grid	✓	✓	✓		✓
Event Hubs	✓	✓	✓		✓
HTTP & webhooks	✓	✓	✓		✓
IoT Hub	✓	✓	✓		✓
Mobile Apps	✓			✓	✓
Notification Hubs	✓				✓
Queue storage	✓	✓	✓		✓
SendGrid	✓	✓			✓
Service Bus	✓	✓	✓		✓
SignalR		✓		✓	✓
Table storage	✓	✓		✓	✓

https://docs.microsoft.com/en-us/azure/azure-functions/functions-triggers-bindings?tabs=csharp#supported-bindings

Languages and Runtimes

Language	1.x	2.x	3.x
C#[1]	GA (.NET Framework 4.8)	GA (.NET Core 2.1[2])	GA (.NET Core 3.1) GA (.NET 5.0)
JavaScript	GA (Node 6)	GA (Node 10 & 8)	GA (Node 14, 12, & 10)
F#	GA (.NET Framework 4.8)	GA (.NET Core 2.1[2])	GA (.NET Core 3.1)
Java	N/A	GA (Java 8)	GA (Java 11 & 8)
PowerShell	N/A	GA (PowerShell Core 6)	GA (PowerShell 7.0 & Core 6)
Python	N/A	GA (Python 3.7 & 3.6)	GA (Python 3.8, 3.7, & 3.6) Preview (Python 3.9)
TypeScript	N/A	GA[3]	GA[3]

https://docs.microsoft.com/en-us/azure/azure-functions/functions-versions

Isolated Function Apps

- Support for .NET functions has been tightly integrated with runtime
  - Run as class library in same process as runtime
  - Must use .NET version supported by runtime
- Functions may now be run in isolated process
  - Enables use of .NET versions not supported by runtime (e.g., .NET 5)
- Guide for running functions on .NET 5.0 in Azure
  - https://docs.microsoft.com/en-us/azure/azure-functions/dotnet-isolated-process-guide

Hosting Plans

The following is a summary of the benefits of the three main hosting plans for Functions:

Plan	Benefits
Consumption plan	Scale automatically and only pay for compute resources when your functions are running. On the Consumption plan, instances of the Functions host are dynamically added and removed based on the number of incoming events. ✓ Default hosting plan. ✓ Pay only when your functions are running. ✓ Scales automatically, even during periods of high load.
Premium plan	Automatically scales based on demand using pre-warmed workers which run applications with no delay after being idle, runs on more powerful instances, and connects to virtual networks. Consider the Azure Functions Premium plan in the following situations: ✓ Your function apps run continuously, or nearly continuously. ✓ You have a high number of small executions and a high execution bill, but low GB seconds in the Consumption plan. ✓ You need more CPU or memory options than what is provided by the Consumption plan. ✓ Your code needs to run longer than the maximum execution time allowed on the Consumption plan. ✓ You require features that aren't available on the Consumption plan, such as virtual network connectivity.
Dedicated plan	Run your functions within an App Service plan at regular App Service plan rates. Best for long-running scenarios where Durable Functions can't be used. Consider an App Service plan in the following situations: ✓ You have existing, underutilized VMs that are already running other App Service instances. ✓ You want to provide a custom image on which to run your functions. ✓ Predictive scaling and costs are required.

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

Timeouts

Function app timeout duration

The timeout duration of a function app is defined by the `functionTimeout` property in the host.json project file. The following table shows the default and maximum values in minutes for both plans and the different runtime versions:

Plan	Runtime Version	Default	Maximum
Consumption	1.x	5	10
Consumption	2.x	5	10
Consumption	3.x	5	10
Premium	1.x	Unlimited	Unlimited
Premium	2.x	30	Unlimited
Premium	3.x	30	Unlimited
App Service	1.x	Unlimited	Unlimited
App Service	2.x	30	Unlimited
App Service	3.x	30	Unlimited

ⓘ Note

Regardless of the function app timeout setting, 230 seconds is the maximum amount of time that an HTTP triggered function can take to respond to a request. This is because of the **default idle timeout of Azure Load Balancer**. For longer processing times, consider using the **Durable Functions async pattern** or **defer the actual work and return an immediate response**.

https://docs.microsoft.com/en-us/azure/azure-functions/functions-scale#timeout

Resources

- Azure Functions documentation
  - https://docs.microsoft.com/en-us/azure/azure-functions
- Microsoft Learn modules
  - https://docs.microsoft.com/en-us/learn/modules/create-serverless-logic-with-azure-functions/
  - https://docs.microsoft.com/en-us/learn/modules/build-api-azure-functions/
  - https://docs.microsoft.com/en-us/learn/modules/develop-test-deploy-azure-functions-with-visual-studio/
  - https://docs.microsoft.com/en-us/learn/modules/develop-test-deploy-azure-functions-with-core-tools/
- SharePoint, Azure Functions, and Visual Studio blog series
  - https://www.paitgroup.com/blog/sharepoint-azure-functions-and-visual-studio-part-1
- YouTube videos
  - https://www.youtube.com/watch?v=lHtzueua0G4
  - https://www.youtube.com/watch?v=lzrar1x6XuM

Thank You

- Big thanks to the organizers, sponsors, and you for making this event possible
- Please fill out your evaluation
- Please keep in touch


rwindsor@paitgroup.com


@robwindsor


https://blogs.msmvps.com/windsor


https://github.com/rob-windsor/rw-azure-funcs-sharepoint