Microservices, IoT and Azure

README

IMPORTANT

This code repository is a reference implementation that accompanies the book Microservices, IoT and Azure. The most up to date version of the source code is available on line at this location:

http://bobfamiliar.github.io/microservices-iot-azure/

The Reference implementation

A working definition of microservices is contrasted with traditional monolithic layered architecture. A fictitious, home-biomedical startup is used to demonstrate microservice architecture and automation capabilities for cross-cutting and business services as well as connected device scenarios for Internet of Things (IoT). Several Azure PaaS services are detailed including Storage, SQL Database, DocumentDb, Redis Cache, Cloud Services, Web API's, API Management, IoT Hub, IoT Suite, Event Hub, and Stream Analytics. Finally the book looks to the future and examines Service Fabric to see how microservices are becoming the de facto approach to building reliable software in the cloud.

The Reference Implementation will provide PowerShell scripts to automate the Provisioning, Build, Deployment and De-Provisioning of the solution. The Solution consists of several independent Microservices built using C# and ASP.NET Web API along with DocumentDb and Redis. In addition there is an IoT component of the solution that is built using Event Hub, Stream Analytics, Cloud Services and SQL Database. There is a sample real-time data visualization client that demonstrates how to orchestrate the Microservices into a complete solution.

Viewed as a whole, the Reference Implementation demonstrates how to use several Azure PaaS Services together along with custom code and automation scripts to create a modern Software-as-a-Service solution.

What you will learn

- What are Microservices and why are they a compelling architecture pattern for SaaS applications
- How to design, develop and deploy Microservices using Visual Studio, PowerShell and Azure
- Microservice patterns for traditional line of business solutions
- Microservice patterns for Internet of Things and Big Data Analytics solutions
- Techniques for automating Microservice provisioning, build and deployment
- What is Service Fabric and how that is the future direction for Microservices on the Microsoft Azure

Chapters

Chapters 1: From Monolithic to Microservice - Shifting demographics and competitive pressure on business to drive impact at velocity is requiring us to evolve our approach to how we develop, deploy, and support our software products. This chapter lays out a roadmap to evolve not only application architecture but also process and organization.

Chapters 2: What Is a Microservice? - This chapter provides a working definition of microservices and details the benefits as well as the challenges to evolving to this architecture pattern.

Chapters 3: Microservice Architecture - Traditionally, we have used separation of concerns, a design principle for separating implementation into distinct layers in order to define horizontal seams in our application architecture. Microservice architecture applies separation of concern to define vertical seams that define their isolation and autonomous nature. This chapter will compare and contrast microservice architecture with traditional layered architecture.

Chapters 4: Azure – A Microservice Platform - The Azure platform exudes characteristics of microservices. This chapter examines several Azure services to identify common patterns of services that are designed and implemented using microservices. Storage, SQL Database, DocumentDb, Redis Cache, Service Bus, API management, and app containers are reviewed.

Chapters 5: Automation - Automation is the key to being able to evolve to a continuous delivery approach and realize the benefits of SaaS. This chapter outlines a framework for defining and organizing your automation process and takes you through 10 exercises that will provision, build, and deploy the reference implementation using PowerShell.

Chapters 6: Microservice Reference Implementation - The epic story of Home Biomedical, a wholly owned subsidiary of LooksFamiliar, Inc., is detailed, and the implementation details of the reference microservices are revealed. The common libraries for ReST calls and DocumentDb and Redis Cache for data access are reviewed. Designing for both public and management APIs is discussed along with the implementation details for the model, interface, service, API, SDK, and console components.

Chapters 7: IoT and Microservices - IoT is becoming a more common solution pattern as we learn to incorporate streaming data into our solutions. This chapter outlines the capabilities needed to realize an IoT solution and maps them to the Azure IoT stack. IoT Hub, IoT Suite, Event Hub, and Stream Analytics are detailed, as well as how to use Event Hub, Cloud Services, and Notification Hub to support mobile alerts. A working example of data visualization using a JavaScript client along with SignalR, ReST, and SQL Database is reviewed.

Chapters 8: Service Fabric - Service Fabric is the microservice management, runtime, and infrastructure that Microsoft uses to build, deploy, and manage their own first-class cloud services such as SQL Database, DocumentDb, Bing Cortana, Halo Online, Skype for Business, In Tune, Event Hubs, and many others. This chapter provides a primer and demonstrates Service Fabric by migrating one of the Web API microservices to Service Fabric.

Requirements

- Windows 8.1 or Windows 10
- Visual Studio 2015

- .NET 4.5.2
- Azure SDK 2.7.1
- Azure PowerShell 0.9.8

Additional Tools

- SQL Server Management Studio
- Postman
- Azure Management Studio

Setup

The details of how to setup the reference implementation are detailed in Chapter 5. There are 10 exercises that take you step by step through the provision, build and deploy process. Once the process is complete, you will have a complete end-to-end IoT solution running in Azure.