
Microsoft Azure

MICROSERVICES WITH SERVICE FABRIC

Making dev's life easier

@g_arora

Gaurav Arora

Gaurav is Microsoft Most Valuable Professional , an author, Mentor and a technology enthusiast with more than 19 years industry experience.

Trends: #GAB2018 #GlobalAzure #IndiaMentor #Azure

AGENDA

- What, why, when of dev's life
- Lets make our life easier – here are microservices
- Service Fabric – makes dev happy
- Microservices in Service Fabric – dev's life is so easy
- Demo

INTRODUCTION

WHAT, WHY, WHEN OF DEV'S LIFE

Alex Soto who is a medical device analyst has defined developer's life in a visual way. Thanks Alex for speaking on the reality. In his article, he tried to visualize: https://dzone.com/articles/reality-developers-life-gifs

- Production environment is always a pain
- Bug, bug and bug
- When you tried to be a Singham (an Indian movie's hero)
- Senior took the credit of work

And many more.

MICROSERVICES

LET'S MAKE OUR LIFE EASIER

We, devs are working in a manner that could be located in different geographical zones, in different teams for same project or any other combination of work. Sometime this is very tough to coordinate with each other or say there's a sync problem chances between devs.

WHAT IS MICROSERVICE?

Microservices is an architectural style is new buzz world that has following characteristics:

- A big project/application can be divided into small pieces of independent services
- Services are independent and loosely coupled
- Every service project is purely independent from other service project – as it contains its own configuration, styles etc. and can be manageable by small teams
- Services can be built on different framework or language e.g. if Service A is written in .NET Core that does not mean that Service B should be written on .NET Core
- These services communicate with each other

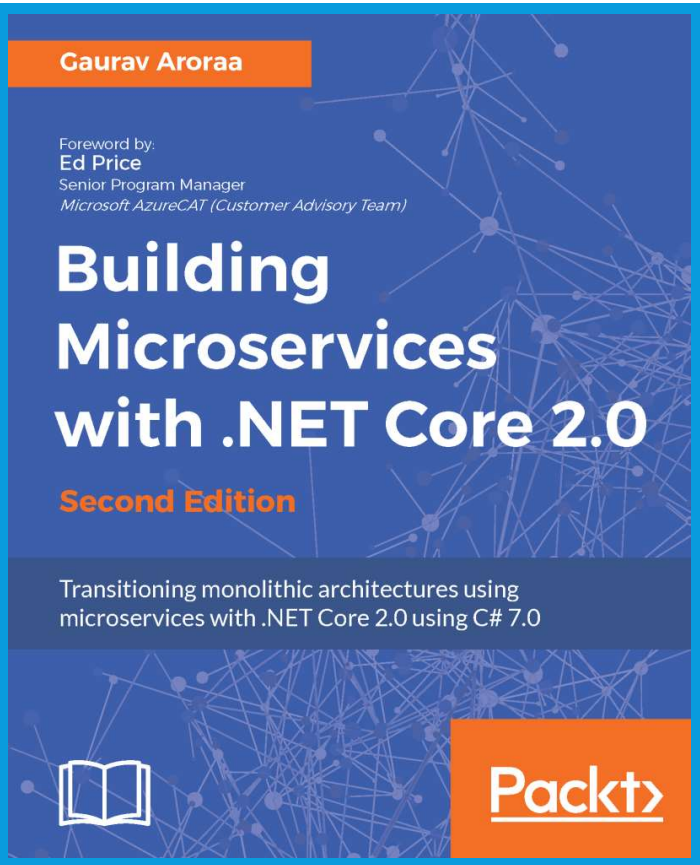
MICROSERVICES: BENEFITS FOR DEV

There will be more benefits apart from mentioned below:

- Knowledge Transfer
- Code/Components
- Deployment
- Technology Choice
- Scaling

BUILDING MICROSERVICES

I have authored a book and you would like to read it.



SERVICE FABRIC

MAKING DEV'S LIFE EASY

Service Fabric is a distributing system platform that is an open source under MIT License. On March 14, 2018 (on PI day) , Service Fabric team blogged about this news: https://blogs.msdn.microsoft.com/azureservicefabric/2018/03/14/service-fabric-is-going-open-source/

Main highlight of Service Fabric is that it can run on Windows and Linux or on your laptop or on cloud besides any geographical locations.

It enables devs to build and manage scalable applications. It empowered many more services viz. Azure SQL DB, Azure Cosmos DB, Azure Event hub etc.

MAKING BUSINESS EFFECTIVE

With Cloud – things have changed. And in some ways – cloud was born out of the need for this change.

1. Time To Market
2. High Availability
3. Scalability

These are all the more important and define your business and development strategy. Frameworks and Databases are secondary now.

WHY MICROSOFT FABRIC

As discussed earlier about Microservices - first step is converting humongous applications into:

1. fine grained ,
2. loosely coupled services
3. that can individually

- scaled and
- updated

as required.

Question – If all is so easy then why is Fabric required?

WHY MICROSOFT FABRIC

There is plenty of pain to be dealt with when writing Microservices. The most critical areas are

1. Inter Service communication
2. Failure Handling

Just imagine a set of 15 Microservices – running 15 instances each and each maintaining its data set.

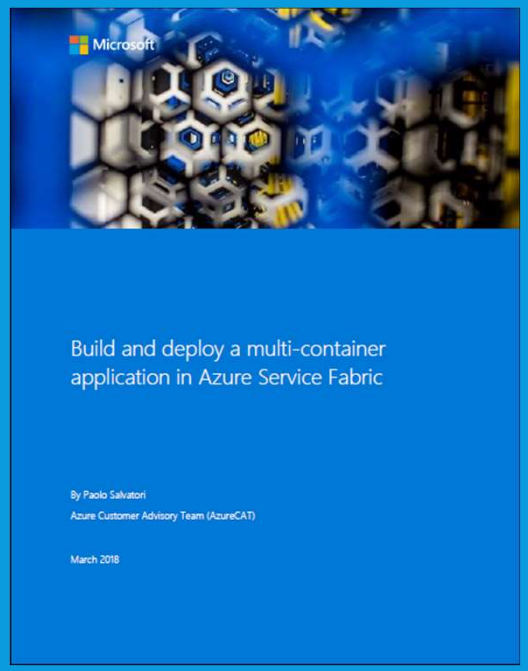
Question - What are the scenarios that you feel are possible?

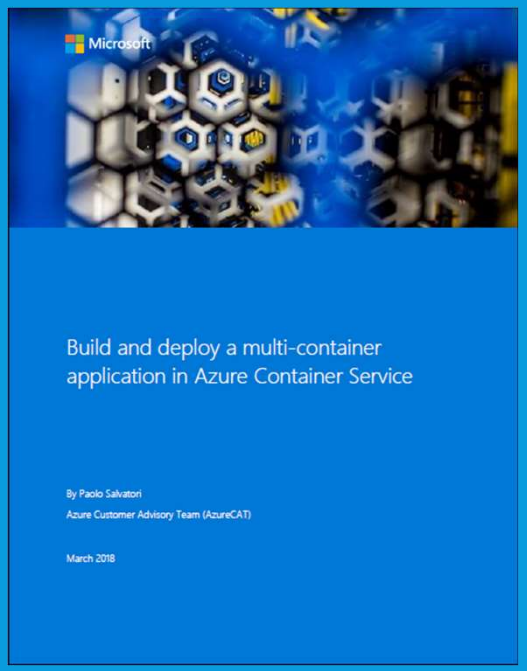
WHY MICROSOFT FABRIC

Microsoft Fabric lets the developers to focus on core business logic rather than trying to figure out solutions to other set of problems around communication or error handling.

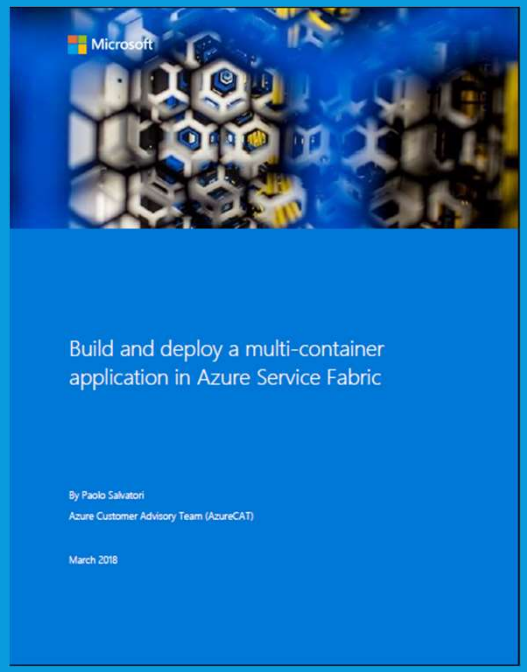
1. Issue Tracking and Mitigation is part of the Fabric.
2. Beauty is that apart from the Stateless Microservices it also allows to run Stateful Microservices.
3. Data and computing is collocated to reduce latency.
4. Dynamic Scaling
5. Easy Scaling Up / Down.
6. State management, Application Life Cycle Management – without Downtime
7. Diagnostics
8. Run Any code.

HTTPS://BLOGS.MSDN.MICROSOFT.COM/AZURECAT/2018/04/17/TWO-NEW-AZURECAT-EBOOKS-BUILD-AND-DEPLOY-A-MULTI-CONTAINER-APPLICATION-SERVICE-FABRIC-AZURE-CONTAINER-SERVICE/

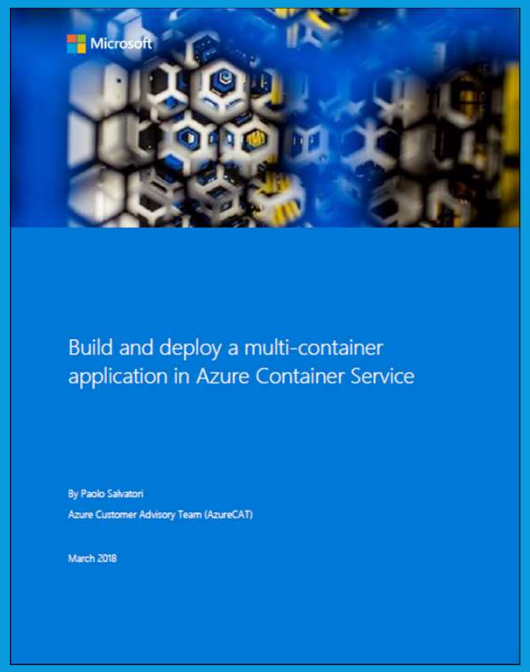




BUILD AND DEPLOY A MULTI-CONTAINER APPLICATION IN AZURE SERVICE FABRIC – A BOOK FROM AZURE CAT



BUILD AND DEPLOY A MULTI-CONTAINER APPLICATION IN AZURE CONTAINER SERVICE – A BOOK FROM AZURE CAT



DEMO-1

https://github.com/garora/talks

WHAT WILL WE BUILD HERE?

1. Application using .NET and Service Fabric
2. ASP.NET Core for web interface
3. A small voting system

PREREQUISITES

1. Visual Studio 2017
2. Azure Development, ASP.NET Web development
3. Microsoft Service Fabric

DEMO-2

https://github.com/garora/talks

WHAT WILL WE BUILD HERE?

1. Building microservices using .NET Core with Service Fabric
2. ASP.NET Core for web interface
3. A small ticketing system

PREREQUISITES

1. Visual Studio 2017 Update 3 or later
2. Azure Development, ASP.NET Core
3. Microsoft Service Fabric

DEMO-3

https://github.com/garora/talks

WHAT WILL WE BUILD HERE?

1. Building microservices using .NET Core with Service Fabric
2. ASP.NET Core for web interface
3. A small ecommerce system – (checkout only)
4. Mapping DevOps (using VSTS)
5. Deployment using containers