

Service Fabric Training Blueprint

Created by: Service Fabric PM team.

Blog - <https://blogs.msdn.microsoft.com/azureservicefabric/?orderby=date>

Github - <https://github.com/Microsoft/service-fabric>

Linkedin - <https://www.linkedin.com/groups/8526708/profile>

Stackoverflow - <https://stackoverflow.com/questions/tagged/azure-service-fabric>

Twitter – https://twitter.com/servicefabric_

Contents

Service Fabric Training Blueprint	1
Overview	4
Audience.....	4
Why use Service Fabric?	4
Service Fabric - WHEN you need it	5
Prerequisites.....	6
Tools and Software	6
Level 100 - Beginner.....	7
Overview:.....	7
Concepts:	7
Setup the local development environment.....	7
Simple Container/Application Solution.....	7
Create Azure Service Fabric Cluster	7
Concepts of SF clusters	8
Deploy .Net Application to Azure SF Party Cluster	8
Scaling Concepts and depth of scaling features	8
Success!	8
Exercise.....	9
LEVEL 200 – Diver – Dig deeper into Service Fabric	9
Overview:.....	9
Guest Executables	9
Containers.....	9
How to communicate with Services:.....	10
Why we need Service Fabric Programming Models	10
Let's Introduce Service Fabric Programming Models – Reliable Services	11
Walk through of Application with SF Programming Models	11
Full cycle of development.....	11
Concepts	12
Reliable Services	12
Dive into Stateful Services.....	13
SF cluster with ARM templates.....	13

Container Network Details	13
Learn more about PowerShell/CLI for Service Fabric	13
Further readings:	13
Success!	14
Exercise:	14
Level 300 – Application Expertise	14
Overview:	14
Concepts	14
Manage application lifecycle	14
Test application and services	15
Continuous Integration	16
Monitor applications	16
Application for multiple environments	16
Application Availability:	16
Application Scaling:	16
Application Security - https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-application-and-service-security	17
Success!	17
Exercise	17
Level 400 – All about planning and cluster infrastructure	17
Overview	17
Cluster Creation –	17
Cluster Runtime –	17
Api Management –	17
Cluster Security –	17
Scale Cluster -	17
Services Distribution on node and Constraints placement to add rules for distribution-	18
Cluster Health -	19
Cluster Monitoring -	19
Cluster Resource Manager- Key to understand resource consumption and management of resources on clusters	19
Plan and Prepare	20
Best Practices and Production Readiness -	20
Success!	20

Exercise:.....	20
Optional Further reading	21
Migration from cloud services -	21
Additional Data for Industry Verticals.....	21
Customer Stories	21

Overview

The document provides Service fabric structured learning guide from a beginner to an expert level. An individual can gain expertise in building large-scale microservices/container solutions in a distributed environment, taking advantage of out of box scale, availability, reliability, and service orchestration. The document will guide you to use key Service Fabric components for building, deploying, and operating services/containers on the cloud. You get an expert -advice, hands on training, videos from the SF team and extensively documented core concepts.

Audience

We expect our audience to have a cloud concept and development stack/dev ops understanding. Knowledge of cloud compute and ecosystem is key for advanced levels. Hands on experience in .Net environment is added advantage to go through hands on labs below.

You don't need to have deep expertise in any one particular language or have advanced expertise in Azure cloud to get through the training.

Why use Service Fabric?

There are lot of reasons to consider Service Fabric. Surprisingly, they originate from multiple perspectives because Service Fabric is beneficial to IT Pros, Developers and above all else the business.

For the Business, Service Fabric means agility and business continuity. Usually, we choose agility first and find effective continuity to be out of reach. Whether it's because of finances or because of technology, these business characteristics are hard to get right. Service Fabric was built from the ground up to tackle these challenges. By leveraging this platform businesses benefit from features like replication, backup, restore and transactions, allowing them to focus on innovation. The increased collaboration between the business, IT Pros and developers fosters a DevOps culture that is required to fulfil the business's SMART objectives. Service Fabric is a platform that supports these initiatives in a way that is scalable, reliable and highly available.

For developers, Service Fabric is a breath of fresh air because Service Fabric dissociates the application from its infrastructure. This means that developers can focus on the business and bring value to customers at a faster pace. Spending a lot of time trying to solve complex infrastructure problems, is counterproductive and can result in rigidity. Service Fabric offers two programming models that simplify code while still providing the advantages of scalability, high availability, integrity and reliability guarantees offered by Service Fabric.

For IT Pros, Service Fabric is a way to bring agility to resource management for mission-critical workloads. Through automation and tooling, it addresses the significant challenges in managing Internet-scale services built using microservices. Facilitating continuous improvement, Service Fabric can automatically rollback unhealthy deployments. It thereby reduces fear and uncertainty that can arise from high-impact deployments. Service Fabric allows code and application state to be independently versioned, deployed and scaled. Consequently, enabling scenarios like A/B-style testing, where users experience distinct versions of a service, Service Fabric helps businesses validate their initiatives and adjust to an ever-changing market.

Service Fabric - WHEN you need it

- Highly available services: Service Fabric services provide fast failover by creating multiple secondary service replicas. If a Virtual Machine, process, or individual service goes down due to hardware or other failure, one of the secondary replicas is promoted to a primary replica with minimal loss of service
- Scalable services: Individual services can be partitioned, allowing for state to be scaled out across the cluster (Group of Virtual Machines). In addition, individual services can be created and removed on the fly. Services can be quickly and easily scaled out from a few instances on a few Virtual Machines to thousands of instances on many Virtual Machines, and then scaled down again, depending on your resource needs. You can use Service Fabric to build these services and manage their complete lifecycles
- Computation on non-static data: Service Fabric enables you to build data, input/output, and compute intensive stateful applications. Service Fabric allows the collocation of processing (computation) and data in applications. Normally, when your application requires access to data, there is network latency associated with an external data cache or storage tier. With stateful Service Fabric services, that latency is eliminated, enabling more performant reads and writes. Say for example that you have an application that performs near real-time recommendation selections for customers, with a round-trip time requirement of less than 100 milliseconds. The latency and performance characteristics of Service Fabric services (where the computation of recommendation selection is collocated with the data and rules) provides a responsive experience to the user compared with the standard implementation model of having to fetch the necessary data from remote storage
- Session-based interactive applications: Service Fabric is useful if your applications, such as online gaming or instant messaging, require low latency reads and writes. Service Fabric enables you to build these interactive, stateful applications without having to create a separate store or cache, as required for stateless apps. (This increases latency and potentially introduces consistency issues.)
- Distributed graph processing: The growth of social networks has greatly increased the need to analyze large-scale graphs in parallel. Fast scaling and parallel load processing make Service Fabric a natural platform for processing large-scale graphs. Service Fabric enables you to build highly scalable services for groups such as social networking, business intelligence, and scientific research.

- Data analytics and workflows: The fast reads and writes of Service Fabric enable applications that must reliably process events or streams of data. Service Fabric also enables applications that describe processing pipelines, where results must be reliable and passed on to the next processing stage without loss. These include transactional and financial systems, where data consistency and computation guarantees are essential.

Prerequisites

Containers - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-containers-overview>

Microservice Architecture - <http://microservices.io/patterns/microservices.html>

Setup - Learning Resources - Pluralsight – Free Access from Azure

Gain access to this resource as we use this content in our training.

Go to <https://azure.microsoft.com/en-us/training/free-online-courses/>

Select Microsoft Azure for .Net Developers – Building Secure Services and application. You will have to setup an account. After setting up the account, you will be given access to the tutorial. You don't have to go through this tutorial now. I have selected relevant videos for you in our training below.

But go through free Azure learning resources in your spare time as they are great place to start.

Tools and Software

Azure Monitor - <https://azure.microsoft.com/en-us/services/monitor/>

Azure App Insights - <https://azure.microsoft.com/en-us/services/application-insights/>

Azure Log Analytics - <https://azure.microsoft.com/en-us/services/log-analytics/>

Azure Key Vault - <https://azure.microsoft.com/en-us/services/key-vault/>

Azure Container Registry - <https://azure.microsoft.com/en-us/services/container-registry/>

Docker Containers - <https://www.docker.com/what-docker>

Windows Containers - <https://docs.microsoft.com/en-us/virtualization/windowscontainers/about/>

Dev Ops –

VSTS - <https://azure.microsoft.com/en-us/services/visual-studio-team-services/>

Jenkins - <https://azuremarketplace.microsoft.com/en-us/marketplace/apps/bitnami.production-jenkins>

Visual Studio 2017 - <https://azure.microsoft.com/en-us/products/visual-studio/>

Azure API Management (<https://azure.microsoft.com/en-us/services/api-management/>)

Application Gateway (<https://azure.microsoft.com/en-us/services/application-gateway/>)

Level 100 - Beginner

Overview:

In this section, you will gain insights into basic concepts of Service Fabric (SF). The training is designed to give a hands-on experience in creating SF clusters and deploying existing applications into the cluster. Along the way, you will also learn about the key SF services and components.

By the end of this session, you have learned to deploy asp.net core/services/containerized apps to Service Fabric and take advantage of scaling provided by SF.

Estimated time to complete this level: 4- 6 hours.

Concepts:

1. What is a service Fabric Cluster?
 1. <https://app.pluralsight.com/player?course=microsoft-azure-dotnet-secure-services-applications&author=scott-allen&name=microsoft-azure-dotnet-secure-services-applications-m3&clip=2&mode=live>
 2. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-overview>
 3. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-architecture>
- b. Basic concepts of Service Fabric
 - i. https://mva.microsoft.com/en-US/training-courses/building-microservices-applications-on-azure-service-fabric-16747?l=tbuZM46yC_5206218965

Setup the local development environment

Windows : <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-get-started>

Simple Container/Application Solution

- c. Container solution – Move application in a container into SF cluster
 - i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-quickstart-containers>
- d. Push .Net application to party cluster
 - i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-quickstart-dotnet>

Create Azure Service Fabric Cluster

- i. Create Cluster using Portal
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-creation-via-portal>
 2. Note: The certificate you download does not have a password. You can additionally give a password or leave it blank. Download the certificate as pfx

and click on the certificate to install it on the machine. Follow the default path, password is blank. Now you have the certificate in your local store.

- ii. Access the cluster
 1. Reach the cluster endpoint
<https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-connect-to-secure-cluster#connect-to-a-cluster-using-powershell>
 2. Reach the SF explorer
 - a. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-connect-to-secure-cluster#connect-to-a-secure-cluster-using-service-fabric-explorer>
 3. Note: As client certificate generated by SF cluster creation is self-signed (ie not authorized by any CA), you will see the message whether you want to trust the site, please select advance and continue. In Chrome, you can select the certificate from the options it provides. IE will automatically read the certificate from your local store/ personal store. This is the certificate you downloaded at the end of cluster creation from Azure portal and installed it.
 4. Explore the SF explorer - At this point, there is no application deployed to SF. But you will see the node type and nodes created - as mentioned in your cluster creation. You can also see SF system services.
 1. Service Fabric Explorer
https://mva.microsoft.com/en-US/training-courses/building-microservices-applications-on-azure-service-fabric-16747?l=bBTFg46yC_9806218965

Concepts of SF clusters

- i. System Services:
<https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-technical-overview#system-services>
- ii. Node Types:
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-nodetypes>
- iii. Security
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-security>
 2. Note: One you downloaded after Service Fabric cluster creation was admin client - i.e. to access SF explorer and endpoints.

Deploy .Net Application to Azure SF Party Cluster

- i. Note: Party clusters are free, limited-time Service Fabric clusters hosted on Azure and run by the Service Fabric team where anyone can deploy applications and learn about the platform. Party clusters are time bound ie your application on the clusters are only available for few hours.
- ii. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-tutorial-deploy-app-to-party-cluster>

Scaling Concepts and depth of scaling features

- i. https://sec.ch9.ms/ch9/8eff/2d449d8a-322e-4613-b9ff-630ba0448eff/CONN17T116_mid.mp4

Success!

Congratulations, you are one among few who can create service fabric applications and cluster. Take advantage of microservice architecture and redesign your application or containerize it and push to service fabric.

Exercise:

We would like to give you a simple exercise to implement the concepts you have learned above. Create a simple .NET web services, push it to Service Fabric's Azure cluster by containerizing your web services. We are sure you have paid attention above! 😊

LEVEL 200 – Diver – Dig deeper into Service Fabric

Overview:

Our objective with this section is to get you deep expertise in SF applications and its programming model. Now that you are in Level2, you are comfortable programming in SF environment and utilizing cloud and basic features of SF.

By end of this session, you can create, debug, deploy new services with SF programming models. Containerize any .net application, debug and deploy into clusters with CI/CD mechanisms. Also, gain knowledge in logging and monitoring of applications on Service Fabric cluster. We will also introduce you to PowerShell commands to create and operate cluster, helpful in continuous deployments.

Estimated time to complete – 9 - 13 hours

Guest Executables

Note: Guest Executable can be anything, your .net web services, java, c++ etc. Guest executables do not call the Service Fabric SDK APIs directly. However, they still benefit from features the platform offers, such as service discoverability, scale, custom health and load reporting by calling REST APIs exposed by Service Fabric. They also have full application lifecycle support. You can always go this route to deploy your services to SF. You don't need SF programming model always. We will discuss when to use SDK API in the next section.

- i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-guest-executables-introduction>
- ii. Build Guest executables - This sample shows Node.js app but you can use your typical Asp.net web services.
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/quickstart-guest-app>

Containers

Note: This section is extension of the container hands on from Level100 – Here you will containerize your existing .net application, debug, and deploy to cloud.

- i. Create a .net application as containers - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-host-app-in-a-container>
- ii. Monitor - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-tutorial-monitoring-wincontainers>
- iii. Build services and containerize –

- i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-get-started-containers>
- iv. Samples - <https://github.com/Azure-Samples/service-fabric-containers>
- v. Debug in Containers - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-how-to-debug-windows-containers>
- vi. Deeper into Containers
 - i. Volumes - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-containers-volume-logging-drivers>
 - ii. View Container Logs - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-containers-view-logs>

How to communicate with Services:

Note: This is an important section where you will explore effective ways of communicating with your services from another service within SF or externally from a browser/system.

- i. Service Communication
 - 1. DNS service:
 - i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-dnsservice>
 - 2. Reverse Proxy –
 - i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-reverseproxy>
 - 3. Reverse proxy config –
 - i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-reverseproxy-configure-secure-communication>
 - 4. Https for application –
 - i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-tutorial-dotnet-app-enable-https-endpoint>

Why we need Service Fabric Programming Models

Services built on SF programming model is called Reliable services. By plugging your services to lightweight programming model, you now get access to Sf runtime and its capabilities. Unlike Service Fabric Services modeled as Guest Executables, Reliable Services get to use the rest of the Service Fabric APIs directly. This allows services to:

- 1) Query the system
- 2) Report health about entities in the cluster
- 3) Receive notifications about configuration and code changes
- 4) Find and communicate with other services,

...and giving them access to many other capabilities, all from a first-class programming model in several programming languages.

Another reason to use SF programming model is to use reliable collections. With reliable collections, you can store data locally in the service instance with high availability and reliability. The services that use reliable collections to store data locally is called Stateful services.

You would want to use stateful services when you need hot data residing in the same VM to reduce latency. If you don't have this requirement, we would suggest using Cosmos DB or SQL Db to store your data persistently. You will learn more about these services in the sections below.

Let's Introduce Service Fabric Programming Models – Reliable Services

Note: Reliable services is one of the programming models, other one being Reliable Actor model. We don't need to go through the latter as it is a very niche case.

- i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-choose-framework>
- ii. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-reliable-services-introduction>
- iii. Application Design with Stateless Services-
 - 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-application-scenarios#an-application-built-using-stateless-services>
- iv. Application Design with Stateful Services:
<https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-application-scenarios#an-application-built-using-stateful-services>

Walk through of Application with SF Programming Models

- i. Create SF App - <https://app.pluralsight.com/player?course=microsoft-azure-dotnet-secure-services-applications&author=scott-allen&name=microsoft-azure-dotnet-secure-services-applications-m3&clip=4&mode=live>
- ii. Exploring App - <https://app.pluralsight.com/player?course=microsoft-azure-dotnet-secure-services-applications&author=scott-allen&name=microsoft-azure-dotnet-secure-services-applications-m3&clip=5&mode=live>
- iii. Asp.net Core and SF Programming Model -
<https://app.pluralsight.com/player?course=microsoft-azure-dotnet-secure-services-applications&author=scott-allen&name=microsoft-azure-dotnet-secure-services-applications-m3&clip=6&mode=live>
- iv. Adding Stateful Services - <https://app.pluralsight.com/player?course=microsoft-azure-dotnet-secure-services-applications&author=scott-allen&name=microsoft-azure-dotnet-secure-services-applications-m3&clip=7&mode=live>

NOTE: We are omitting next order of the Pluralsight tutorial purposefully. The reason being, we will cover the latest design for communication from front end to backend stateful services and deploying to azure in "Full cycle of development" section below.

Full cycle of development

- i. This is same application you downloaded (Voting app) and deployed to Service Fabric cluster.
- vii. Create with new SF programming model-
 - i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-tutorial-create-dotnet-app>
- viii. Deploy –
 - i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-tutorial-deploy-app-to-party-cluster>

- ix. Enable HTTPS –
 - i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-tutorial-dotnet-app-enable-https-endpoint>
- x. CI/CD –
 - i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-tutorial-deploy-app-with-cicd-vsts>
- xi. Monitor –
 - i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-tutorial-monitoring-aspnet>
- xii. Manifest for the application and its terminology - This manifest talk of Voting Services you used in Level100
 - i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-manifest-example-reliable-services-app>

Concepts

- i. Design time Concepts -
 - 1. https://mva.microsoft.com/en-US/training-courses/building-microservices-applications-on-azure-service-fabric-16747?l=tlklo46yC_29o6218965
 - 2. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-content-roadmap#design-time-application-type-service-type-application-package-and-manifest-service-package-and-manifest>

Reliable Services

- i. Lifecycle -
 - 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-reliable-services-lifecycle>
- iv. Communicate with reliable services - We show where you can plug the listener in reliable services and give some examples of communication listeners in this section. Service Fabric ships nuget packages for Asp.net core (Kestrel,httpsys). You can also write your own custom listeners.
 - i. ASP.net core in Service Fabric
 - i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-reliable-services-communication-aspnetcore>
- v. Migrating WCF services and plug into SF programming model

You can alternatively use containers without using SF programming model to move to SF. However, if you chose to go through programming model route, then you can use the following communication listener for communication.

 - i. WCF Communication - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-reliable-services-communication-wcf>
 - ii. Secure WCF communication - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-reliable-services-secure-communication-wcf>

- vi. Notification in Reliable Services:
 - i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-reliable-services-notifications>

Dive into Stateful Services

- 1. Reliable Collections -
 - a. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-reliable-services-reliable-collections>
 - b. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-work-with-reliable-collections> - Please read the Next sections to get a concept clear on data contracting and programming concepts used in reliable collections.
 - c. Reliable Collection Serialization - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-reliable-services-reliable-collections-serialization>
- i. Partitioning of Stateful Services - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-concepts-partitioning>

SF cluster with ARM templates

You have created Azure cluster through portal, but this section will let you create SF cluster through ARM templates. ARM templates help you manage and update your clusters.

- i. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-creation-via-arm>
- ii. Connect to a cluster with CLI
<https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-connect-to-secure-cluster#connect-to-a-secure-cluster-using-azure-service-fabric-cli-sfctl>
- iii. Connect to a cluster with Powershell
<https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-connect-to-secure-cluster#connect-to-a-cluster-using-powershell>
- iv. Connect to service fabric explorer: <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-connect-to-secure-cluster#connect-to-a-secure-cluster-using-service-fabric-explorer>
- v. Setup client certificate - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-connect-to-secure-cluster#set-up-a-client-certificate-on-the-remote-computer>

Container Network Details

- i. Network Mode - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-networking-modes#next-steps>

Learn more about PowerShell/CLI for Service Fabric

- i. <https://docs.microsoft.com/en-us/powershell/module/azurerm.servicefabric/?view=azurerm-ps-6.5.0>
- ii. Service Fabric powershell - <https://docs.microsoft.com/en-us/powershell/module/servicefabric/?view=azureservicefabric-ps>
- iii. CLI - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-sfctl>

Further readings:

Monitor with Visual Studio -

<https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-diagnostics-how-to-monitor-and-diagnose-services-locally>

Success!

Hurray! You have now gained expertise in SF programming models, development lifecycle and managing containers.

Exercise:

Now create a business application or side fun project that uses SF programming models and deploy to clusters. Take your containerized applications created in previous section's exercise and hook up logging and monitoring. You will continue to work on applications with SF concepts throughout the sections.

If you have managed to create a great use case and application, you can email it to our team. If our team likes it, your application will be highlighted in one of our blogs and you will have a chance to explain and write about it.

Level 300 – Application Expertise

Overview:

In this section, you will learn how to manage your application with Service Fabric. You will learn to manage the application lifecycle as well as upgrade and scale with SF features. By the end of this session, you will have gained enough knowledge to deploy and manage your application into production. You should have a thorough understanding of SF application manifests, upgrading your application after deployment, testing, deploying, monitoring, security and finally scaling.

Estimate time to complete – 5 to 7 hours

Concepts

- i. Models and Hosting
 - Application Model - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-application-model>
 - Learn more about Manifests - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-application-and-service-manifests>
- ii. Examples –
 - 1. Reliable Services - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-manifest-example-reliable-services-app>
 - 2. Containers - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-manifest-example-container-app>
- iii. Hosting Model - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-hosting-model>

Manage application lifecycle

- iv. Application Lifecycle

1. - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-application-lifecycle>
- v. Application Upgrade –
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-application-upgrade>
- vi. Configuration –
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-visualstudio-configure-upgrade>
- vii. Application Upgrade parameters –
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-visualstudio-configure-upgrade>
- viii. Data Serialization in App Upgrade –
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-application-upgrade-data-serialization>
- ix. Upgrade Application -Advanced –
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-application-upgrade-advanced>
- x. Change/Update Service Manifests –
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-service-manifest-resources>
- xi. Upgrade with Powershell –
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-application-upgrade-tutorial-powershell>
- xii. Upgrade with Visual Studio –
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-application-upgrade-tutorial>
- xiii. Troubleshoot Applications –
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-application-upgrade-troubleshooting>
- xiv. Run Time concepts -
 1. https://mva.microsoft.com/en-US/training-courses/building-microservices-applications-on-azure-service-fabric-16747?l=x7CVH56yC_1406218965
 2. https://mva.microsoft.com/en-US/training-courses/building-microservices-applications-on-azure-service-fabric-16747?l=x7CVH56yC_1406218965

Test application and services

- xv. Test Communication - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-testability-scenarios-service-communication>
- xvi.
- xvii. Simulate Failures:
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-controlled-chaos>
 2. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-testability-actions>
 3. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-testability-workload-tests>
 4. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-testability-scenarios>
 5. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-node-transition-apis>

- xviii. Test with Fault analysis - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-testability-overview>

Continuous Integration

- xix. CI\CD with VSTS –
 - 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-tutorial-deploy-app-with-cicd-vsts> - You have gone through this tutorial at the Level 100 and 200.
- xx. CI\CD Jenkins –
 - 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cicd-your-linux-applications-with-jenkins>

Monitor applications

- xxi. Monitoring with App Insights –
 - 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-diagnostics-event-analysis-appinsights>
- xxii. Logging –
 - 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-how-to-diagnostics-log>
- xxiii. Aggregate Events with EventFlow –
 - 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-how-to-diagnostics-log>
- xxiv. Container logging -
 - 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-diagnostics-oms-containers>

Application for multiple environments

<https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-manage-multiple-environment-app-configuration>

Application Availability:

Availability with replication of instances - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-availability-services>

Reconfiguring Replication – <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-concepts-reconfiguration>

Refresher on service instances and replicas - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-concepts-replica-lifecycle>

Application Scaling:

- xxv. Scale Applications –
 - 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-concepts-scalability>
- xxvi. Availability of Services –
 - 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-availability-services>
- xxvii. Partition of Services –
 - 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-concepts-partitioning>

Application Security - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-application-and-service-security>

Further Reading – Key Vault usage for application

Success!

Congrats! You have now gained the expertise to manage your application on SF clusters. You are completely ready to write production ready application.

Exercise

Take your application from previous section's exercise and upgrade your application deployed on SF clusters. Try simulating test failures for your application. Set up scaling abilities to your application.

Level 400 – All about planning and cluster infrastructure

Overview

This section will guide you through all the scenarios of setup, manage and maintenance of your application on SF clusters. Gives you in depth understanding of scaling, loading, optimization of resources in Service Fabric. Deep dive into infrastructure management of SF.

After this section, you should be completely ready to deploy any applications/containers into various environments, including production, and take advantage of SF features to its full capacity.

Estimate time to completion – 8 – 9 hours

Cluster Creation –

<https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-tutorial-create-vnet-and-windows-cluster>

Cluster Runtime –

<https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-tutorial-upgrade-cluster>

Api Management –

xxviii. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-tutorial-deploy-api-management>

Cluster Security –

xxix. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-security>

Scale Cluster -

xxx. Long Video - Watch to gain perspective on

1. - Managing, securing, and scaling Azure Service Fabric clusters and applications - BRK3209
- xxxi. Key Concepts before we talk more about Scaling -
 1. Durability Tier - reduces risk of data loss in a stateful services due to infrastructure upgrades, scale in and out. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-capacity#the-durability-characteristics-of-the-cluster>
 2. Reliability Tier -Remember Reliability tier is guidance on providing reliability to your services <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-capacity#the-reliability-characteristics-of-the-cluster>
 3. Concepts - Watch the Video - a good one.
 - a. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-concepts-scalability>
- xxxii. Scaling
 1. Overview- <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-scaling>
 2. Commands to Scale - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-tutorial-scale-cluster>
 3. Scale programmatically - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-programmatic-scaling>
 4. Auto Scaling - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-resource-manager-autoscaling>
 5. Manual Scale out - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-scale-up-down>
 6. Scale up of Primary Node - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-upgrade-primary-nodetype-vm>

Container Scale -

All the above concepts stick to containers. You would do additional resource governance - Valid this? and fill this section

Services Distribution on node and Constraints placement to add rules for distribution-

- xxxiii. Placement constraint - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-resource-manager-cluster-description#node-properties-and-placement-constraints>
- xxxiv. Resource governance - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-resource-governance>
- xxxv.
- xxxvi. Service Placement Constraints –
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-resource-manager-advanced-placement-rules-placement-policies>
- xxxvii. Defragmented Load & Metrics –
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-resource-manager-defragmentation-metrics>
- xxxviii. Cluster Balancing –
 1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-resource-manager-balancing>
- xxxix. Throttle Resources –

1. <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-resource-manager-advanced-throttling>

Cluster Health -

View Health report - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-view-entities-aggregated-health>

Cluster Monitoring -

- xl. Log Analytics - Monitor the cluster and infrastructure
 1. Setup the log analytics - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-diagnostics-oms-setup>
 2. Monitor - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-diagnostics-event-analysis-oms>
- xli. Monitor performance counters - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-diagnostics-oms-agent>
- xl. Diagnostics - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-diagnostics-event-aggregation-wad>
 1. Events -
 2. SF Events - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-diagnostics-events>
 3. Event store - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-diagnostics-eventstore>
 4. Reliable Service Events - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-reliable-services-diagnostics>

Cluster Resource Manager- Key to understand resource consumption and management of resources on clusters.

- xl. Resource Manager Architecture - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-resource-manager-architecture>
- xl. Describer a cluster - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-resource-manager-cluster-description>
- xl. Application groups overview - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-resource-manager-application-groups>
- xl. Configure Cluster Resource Manager - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-resource-manager-configure-services>
- xl. Resource Consumption - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-resource-manager-metrics>
- xl. Health Integration - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-resource-manager-management-integration#health-integration>
- xl. Blocklisting Nodes - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-resource-manager-management-integration#blocklisting-nodes>

Service Movement - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-resource-manager-movement-cost>

Service Affinity - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-resource-manager-metrics>

Plan and Prepare

Disaster Recovery - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-disaster-recovery>

Capacity Planning - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-capacity>

Application Capacity - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-capacity-planning>

Standalone Cluster: Follow this entire section

<https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cluster-creation-for-windows-server>

Periodic Backup - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-backuprestore-service-configure-periodic-backup>

Best Practices and Production Readiness -

Cluster Capacity and Management - https://mva.microsoft.com/en-US/training-courses/service-fabric-patterns-and-practices-16925?l=cyDYZcSGD_2805167344

Hyperscale Web - https://mva.microsoft.com/en-US/training-courses/service-fabric-patterns-and-practices-16925?l=NgldAdSGD_405167344

Production Readiness Checklist - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-production-readiness-checklist>

Success!

You are an achiever. Wow! You gained deep perspective and hands on experience to develop, design, deploy and maintain applications on SF. You are now ready for production scale applications that you can monitor, scale, load, plan to capacity, manage consumption and check for readiness.

Exercise:

Add placement constraints, capacity constraints, dynamic capacity reporting, and balancing thresholds. See how nodes and service instances are balanced and managed with the change of these parameters. Devise auto scale scenarios and observe scale in and scale out and decide when to use different durability and reliability tiers.

Design cluster for your enterprise application and make decisions on different node types, capacity for each node type, any placement constraint required for distribution of your applications, partitions for your stateful services and resource governance for your applications.

If you think you have nailed it, please send your enterprise application and cluster design for your microservices or containers to us. We will try to showcase it on our blog.

Optional Further reading

Migration from cloud services -

https://mva.microsoft.com/en-US/training-courses/service-fabric-patterns-and-practices-16925?l=hd1cMgSGD_5605167344

Migrate from cloud services - <https://docs.microsoft.com/en-us/azure/architecture/service-fabric/migrate-from-cloud-services>

From Web and Worker Role to SF - <https://docs.microsoft.com/en-us/azure/service-fabric/service-fabric-cloud-services-migration-worker-role-stateless-service>

Refactor App - <https://docs.microsoft.com/en-us/azure/architecture/service-fabric/refactor-migrated-app>

Additional Data for Industry Verticals

Service Fabric Team blog -

<https://blogs.msdn.microsoft.com/azureservicefabric/>

Use Cases:

IOT - https://mva.microsoft.com/en-US/training-courses/service-fabric-patterns-and-practices-16925?l=6wECzeSGD_3805167344

Gaming - https://mva.microsoft.com/en-US/training-courses/service-fabric-patterns-and-practices-16925?l=6wECzeSGD_3805167344

Customer Stories

IOT/Manufacturing -

BMW - <https://blogs.msdn.microsoft.com/azureservicefabric/2016/08/24/service-fabric-customer-profile-bmw-technology-corporation/>

Honeywell - <https://blogs.msdn.microsoft.com/azureservicefabric/2018/03/20/service-fabric-customer-profile-honeywell/>

Schneider Electric - <https://blogs.msdn.microsoft.com/azureservicefabric/2016/08/05/service-fabric-customer-profile-schneider-electric/>

Mesh Systems - <https://blogs.msdn.microsoft.com/azureservicefabric/2016/06/20/service-fabric-customer-profile-mesh-systems/>

Consumer products -

Alaskan Airlines -

<https://blogs.msdn.microsoft.com/azureservicefabric/2017/10/18/service-fabric-customer-profile-alaska-airlines/>

Talk Talk TV - <https://blogs.msdn.microsoft.com/azureservicefabric/2016/03/15/service-fabric-customer-profile-talktalk-tv/>

Finance -

D+H tech - <https://blogs.msdn.microsoft.com/azureservicefabric/2017/05/09/service-fabric-customer-profile-dh/>

Tax/Accounting - Wolters Kluwer -

<https://blogs.msdn.microsoft.com/azureservicefabric/2017/05/08/service-fabric-customer-profile-wolters-kluwer-cch/>

Quorum Business - <https://blogs.msdn.microsoft.com/azureservicefabric/2016/11/15/service-fabric-customer-profile-quorum-business-solutions/>