

Sponsors and Partners

Strategic Sponsors



Gold Sponsors



Silver Sponsors





(400!)



.NET

Deep dive into Service Fabric after 2 years

Tomasz Kopacz

60+ min

(VERY FAST INTRO – WHAT IS
SERVICE FABRIC)

Service Fabric advertisement ☺

Scale

- Scale from small departmental apps to internet scale
- Utilize hardware efficiently at scale

SINGLE CODE BASE!

High Availability/Reliability

- Quickly react to failures of application code, machines, and networks
 - Do it without complexity/cost (Don't buy a SAN)
 - Do it in the cloud (without controlling the hardware)

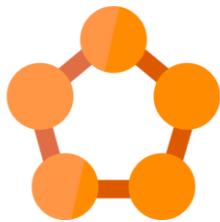
Performance and Latency

- **Millisecond response times for reads**
- **High throughput for writes**

Management

- Perform online upgrades and patches
- Monitor and maintain health information of your distributed application
- Get real-time information about topology etc.

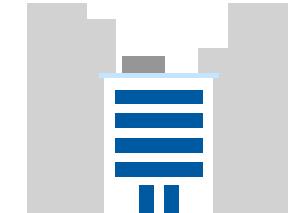
Scalable AND Fast
(not OR – as usual)



Azure Service Fabric



Public Cloud
(Azure)



On Premises
(Own machines)



Other Clouds
(Amazon AWS, etc.)



Service Fabric 101

Cluster: OS Instances
(PHY/VIRT/DOCKER). Pool of Resources. Nodes. Environment Independent Abstraction Layer

Fault Domain & Update Domain

Active-Passive ("Primary" and "Secondary") with fast failover.

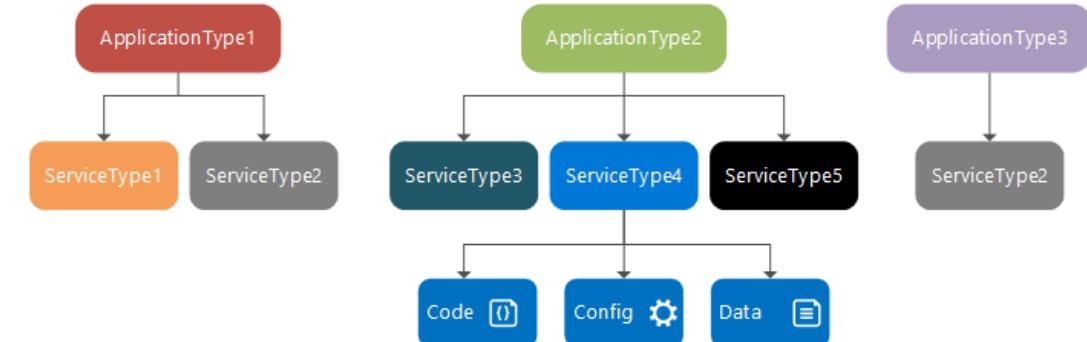
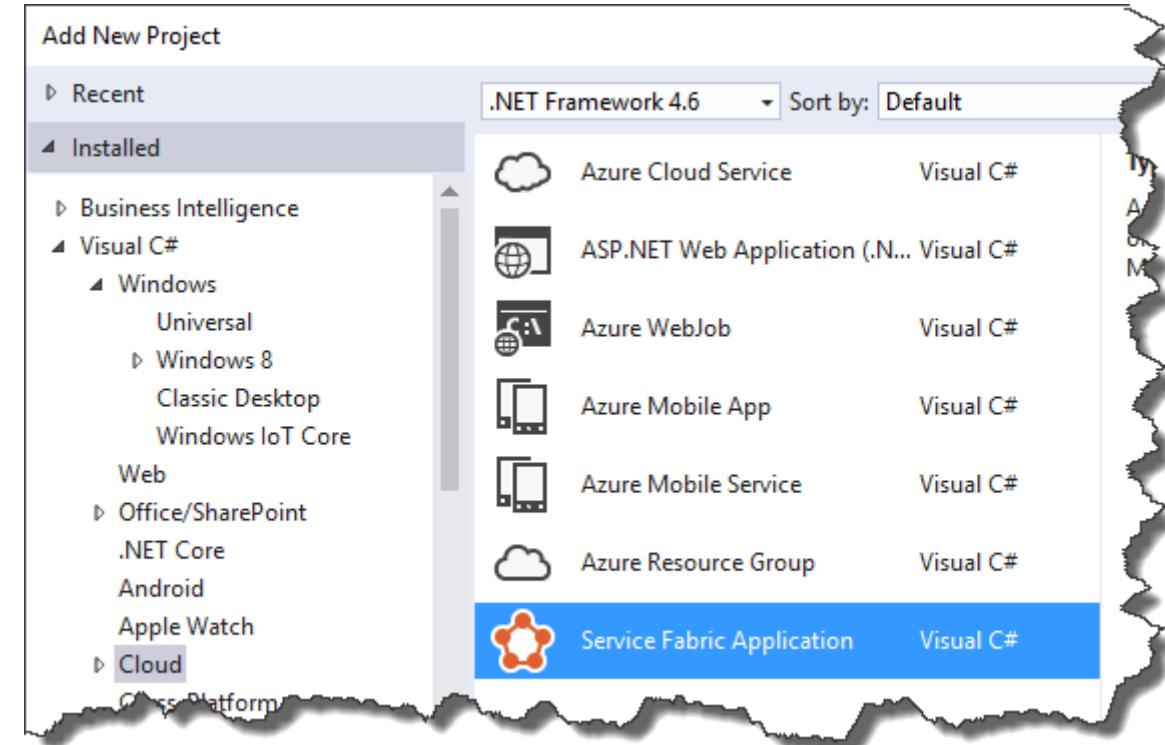
Service: code and state that Fabric manages.

Service **Instance** can communicate with another using any protocol

Service can have **partitions** (scale out unit)

Replicas in statefull services (usually one primary and two secondary, quorum)

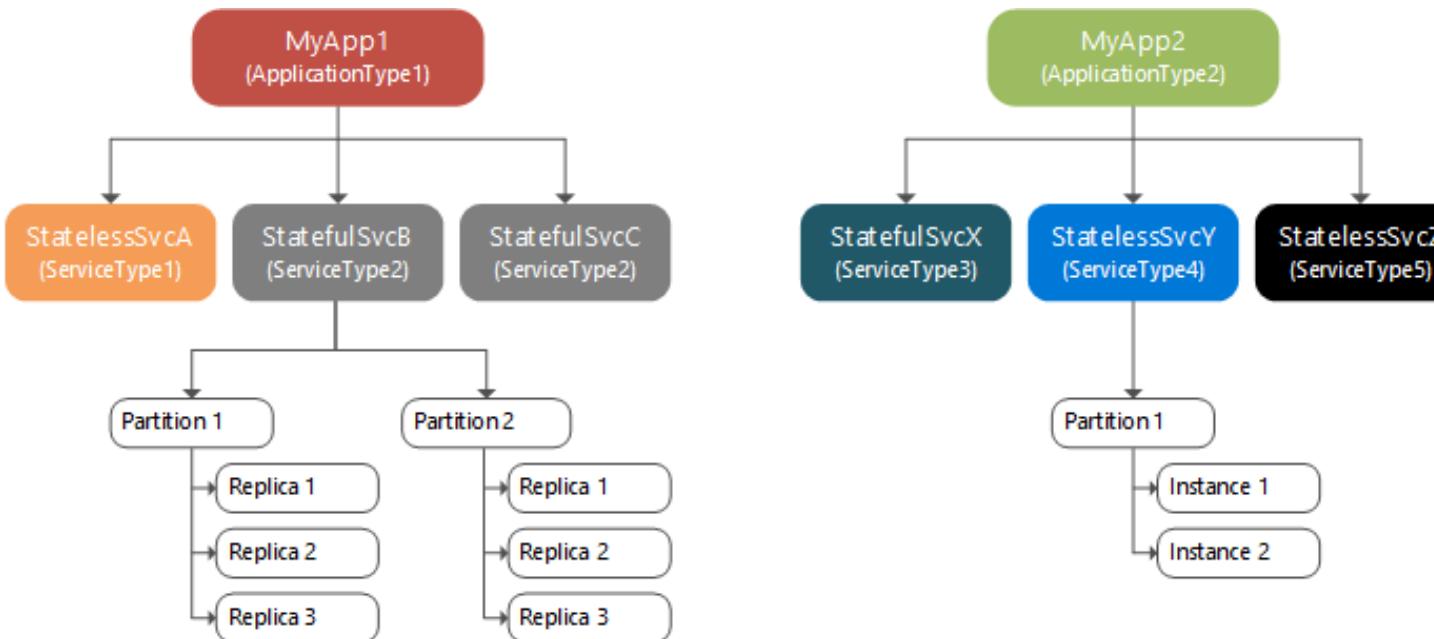
Application is a collection of services.
Upgradeable / downgradeable



Service Fabric API

Guest Executables	Stateless Services	Reliable Services	Reliable Actors
<ul style="list-style-type: none">• Bring any exe• Any language• Any programming model• Packaged as Application• Gets versioning, upgrade, monitoring, health, etc.• Historical: similar to the old „Cloud Services”• Containers (as a way of deploying „guests”)	<ul style="list-style-type: none">• No state or it can be retrieved from an external store• There can be N instances, e.g. web frontends, protocol gateways, utility services• Technically, can be partitioned – but there is no reason to do that ☺	<ul style="list-style-type: none">• Statefull services• Concurrent, granular state changes• Use of the Reliable Collections• Transactions across collections• Full platform integration• N consistent copies (QUORUM) achieved through replication and local persistence• Great for manual control!	<ul style="list-style-type: none">• Stateless & statefull actor objects• Simplified programming model• Single Threaded model• Great for scaled out compute and state• Many ready to use automations!• Do not fight with “defaults”• If necessary – use reliable services!

Partitions, replicas etc – logical view



Application Name	fabric;TK_2016MainSFFunctions
Partition Description	
Partition Scheme	Int64Range
Count	3
Low Key	1
High Key	3
Target Replica Set Size	
3	
Min Replica Set Size	
3	
Replica Restart Wait Duration Seconds	
1800	
Quorum Loss Wait Duration Seconds	
3214202341	
Stand By Replica Keep Duration Seconds	
604800	
Move Cost	
Zero	

Physical: Fault Domain and Update Domain

Statefull A

- Primary: 0, Secondary: 1,2

Statefull B

- Partition 0
 - Primary: 1, Secondary: 2,3
- Partition 1
 - Primary: 2, Secondary: 3,4

Statefull C

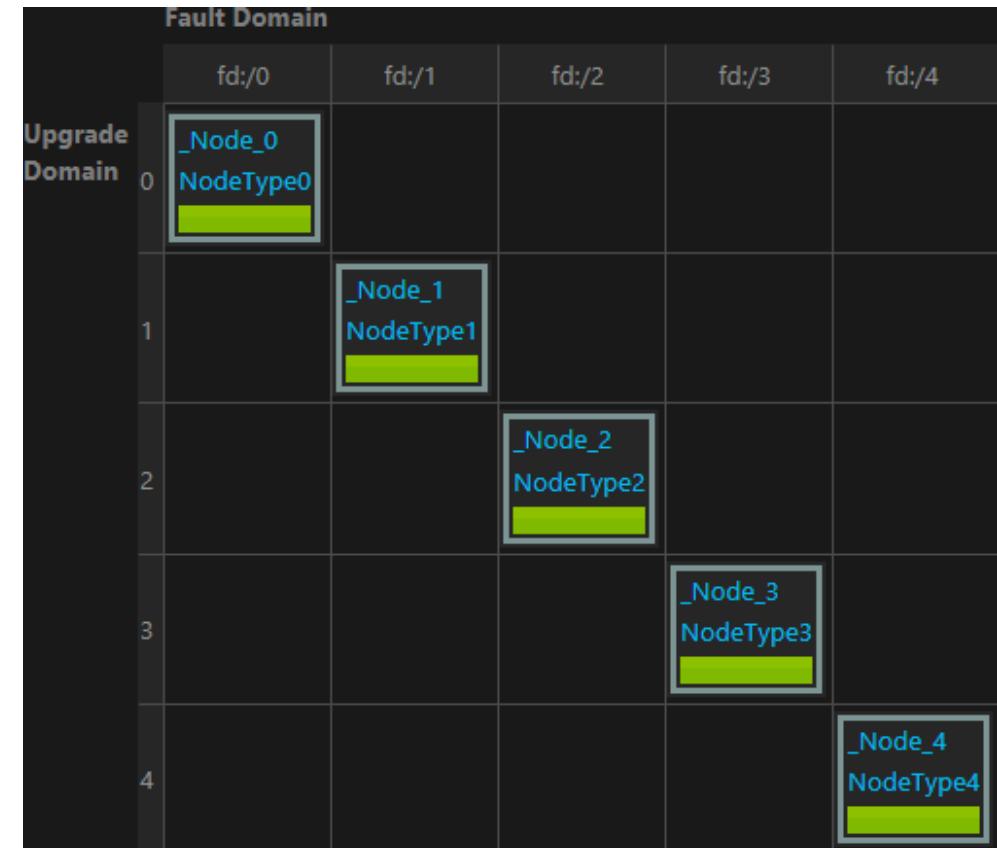
- Primary 0 (for some reasons!), Secondary 4,1

Stateless D

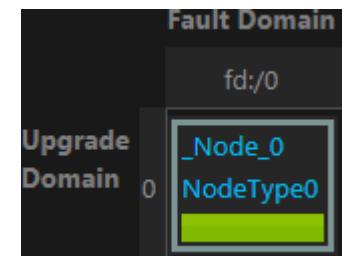
- Node 0

Web E

- Load Balancer, Node 0,1,2,3,4



We can also have a
“single node”
Service Fabric!



Demo

First, slightly more complicated than Hello World
Service Fabric App

Fast Demo

Guest services

Great, but - how to design Application

Architecture

- Any helpful patterns?
- How to design microservices architecture?



Other talks:

[Hands-on Experience: What it Means to Design a Domain Model](#)
[CQRS for... everyone!](#)

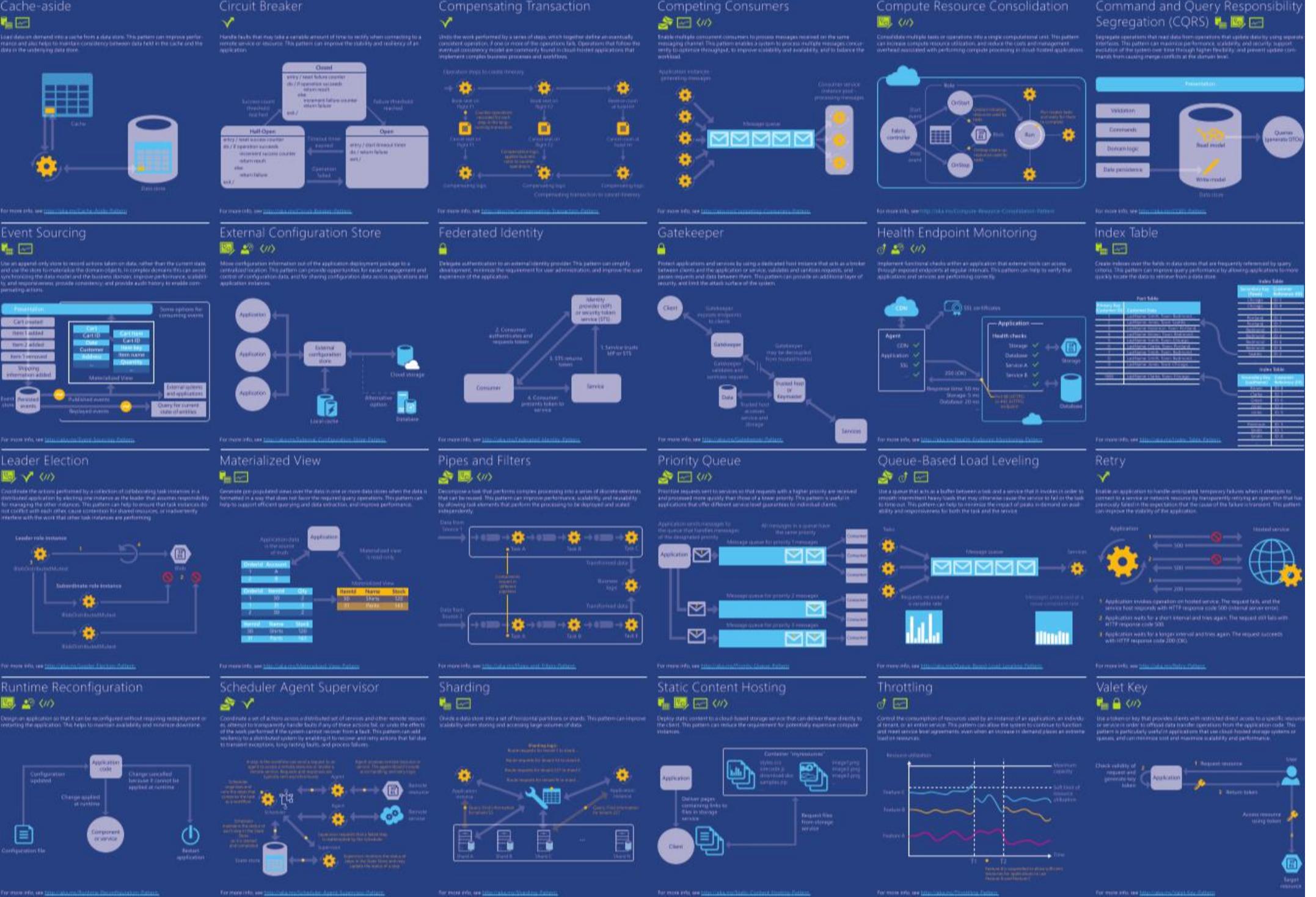
[DDD: Where's the Value and What's in It for Me?](#)

(And materials at the end of this PPTX ☺)

Technology

- **When to use Reliable Services and when – Actors**
- **How to combine Service Fabric and App Services**
- **How to use reliable collections**
- **Technical: garbage collector, scale, upgrade, ...**

Cloud Design Patterns



Going back to technology

Few minutes on Reliable Collections

Remember:
Immutable object

Demo

Reliable collection

„Shopping cart“

Shopping cart, long invoice, ERP form, State per customer, Web session

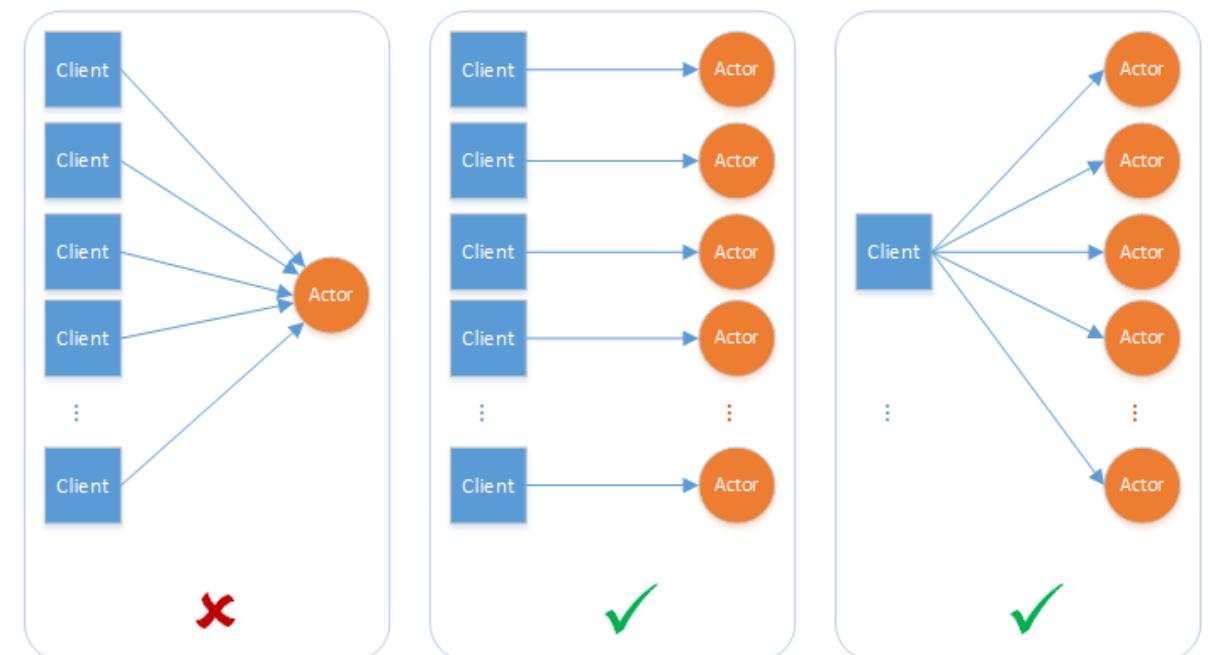
Processing jobs, Gaming session, User accounts

2 separate processes (Actors) – for “managing” cart and (optionally) for storing it in external storage

Communication between actors: direct (bigger “set of” baskets)

Actor – Stateless service (single): queue based

Customer is responsible
for storing ID



Demo

Shopping Cart

Actors: Concurrency, timers,
reminders and garbage
collector

Concurrency, timers and reminders

Turn based concurrency

Internal: counter 😊

No more than one thread can be active inside the actor code at any time.

ActorServiceSettings

ActorConcurrencySettings

A -> B -> C, then C -> A reentrant, logical context

ActorGarbageCollectionSettings

(Many performance counters!)

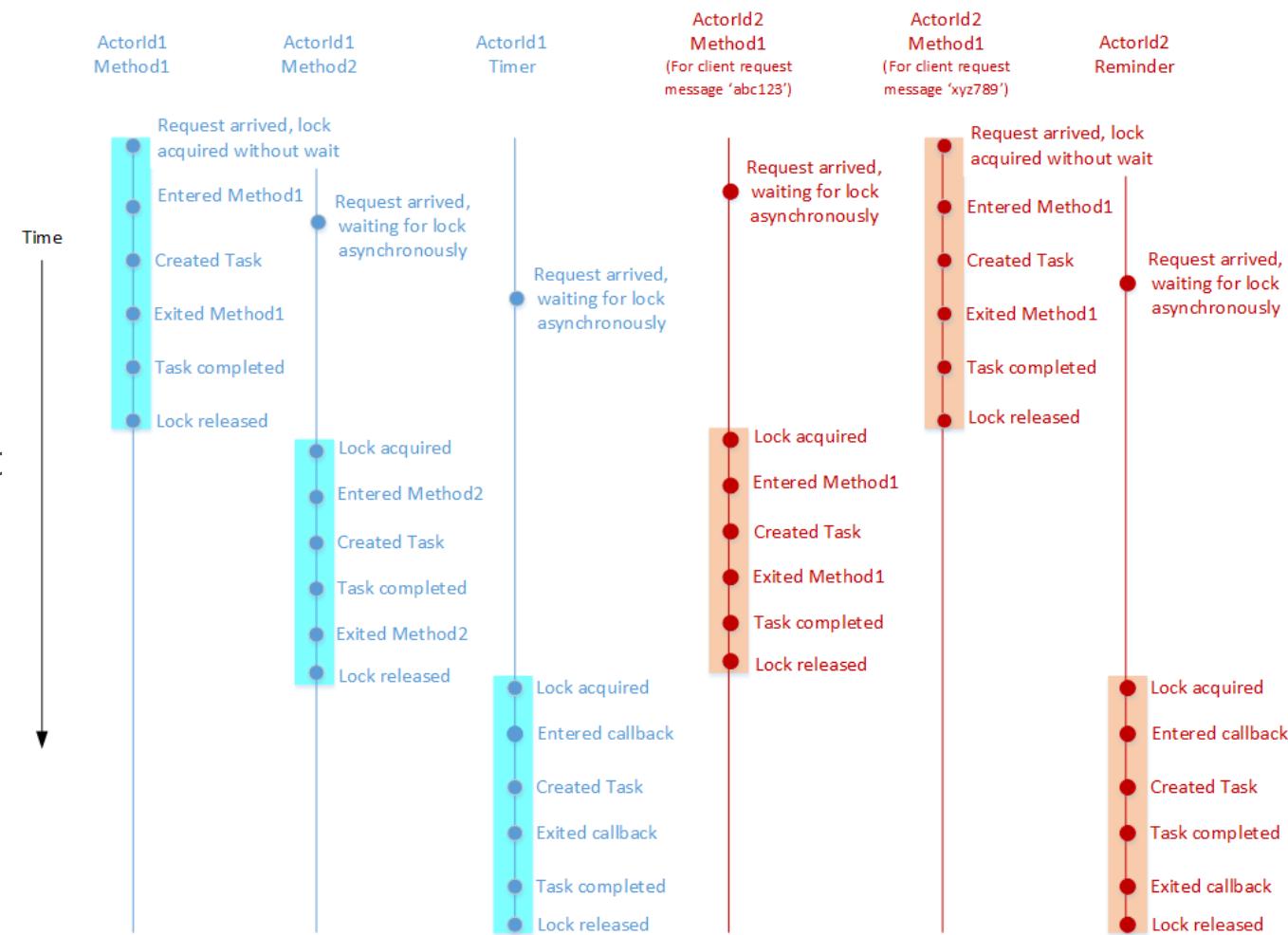
Schedule:

Timer

Callback when Actor is alive

Reminder

Callback and can wake up (restore state) for actor



Garbage Collector

Actor is alive when:

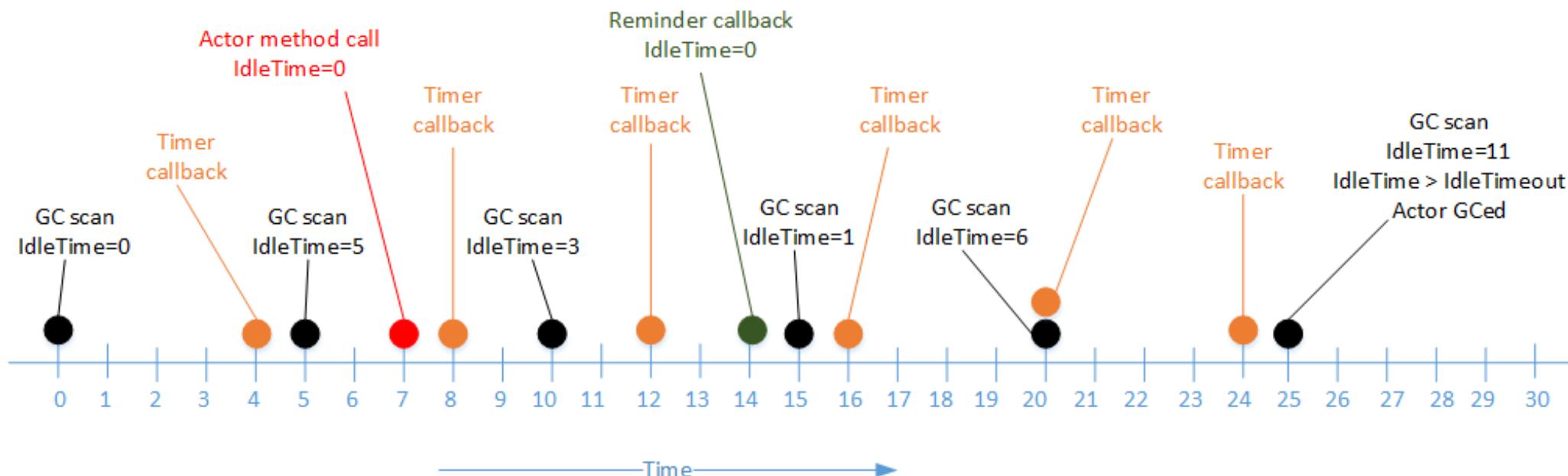
Receiving a call

IRemindable.ReceiveReminderAsync

ActorGarbageCollectionSetting

IdleTimeout, ScanInterval

ScanInterval = 5, IdleTimeout = 10



```
ActorRuntime.RegisterActorAsync<Reminder1>(
    (context, actorType) => new ActorService(context, actorType,
    null, null, null,
    new ActorServiceSettings())
{
    ActorConcurrencySettings = new ActorConcurrencySettings()
    { ReentrancyMode = ActorReentrancyMode.Disallowed },
    ActorGarbageCollectionSettings =
    //new ActorGarbageCollectionSettings(60 * 60, 60) //Default
    new ActorGarbageCollectionSettings(5 * 60, 60) // 5 min
}
```

Demo (ok, code browse)

Timers and Reminders

Timers

10/8/2016 10:56:05 AM - CUSTOM EVENT

OnDeactivateAsync - P1, Timers Stop
Device type: PC Application version: 4.0

10/8/2016 10:56:01 AM - CUSTOM EVENT

OnDeactivateAsync - P2, Timers Stop
Device type: PC Application version: 4.0

10/8/2016 10:55:57 AM - CUSTOM EVENT

OnDeactivateAsync - P3, Timers Stop
Device type: PC Application version: 4.0

10/8/2016 10:55:06 AM - CUSTOM EVENT

doWorkInTimerAsync P2, 68
Device type: PC Application version: 4.0

10/8/2016 10:54:06 AM - CUSTOM EVENT

doWorkInTimerAsync P2, 68
Device type: PC Application version: 4.0

10/8/2016 10:53:06 AM - CUSTOM EVENT

doWorkInTimerAsync P2, 68
Device type: PC Application version: 4.0

10/8/2016 10:53:04 AM - CUSTOM EVENT

doWorkInTimerAsync P3, 188
Device type: PC Application version: 4.0

10/8/2016 10:52:05 AM - CUSTOM EVENT

doWorkInTimerAsync P2, 68
Device type: PC Application version: 4.0

10/8/2016 10:51:04 AM - CUSTOM EVENT

doWorkInTimerAsync P2, 68
Device type: PC Application version: 4.0

10/8/2016 10:50:04 AM - CUSTOM EVENT

RegisterTimerAsync - P3, 188
Device type: PC Application version: 4.0

10/8/2016 10:50:04 AM - CUSTOM EVENT

OnActivateAsync B - P3
Device type: PC Application version: 4.0

10/8/2016 10:50:04 AM - CUSTOM EVENT

Timer1 - P3
Device type: PC Application version: 4.0

10/8/2016 10:50:03 AM - CUSTOM EVENT

RegisterTimerAsync - P2, 68
Device type: PC Application version: 4.0

10/8/2016 10:50:03 AM - CUSTOM EVENT

OnActivateAsync B - P2
Device type: PC Application version: 4.0

10/8/2016 10:50:03 AM - CUSTOM EVENT

Timer1 - P2
Device type: PC Application version: 4.0

10/8/2016 10:50:02 AM - CUSTOM EVENT

RegisterTimerAsync - P1, 4800
Device type: PC Application version: 4.0

10/8/2016 10:50:02 AM - CUSTOM EVENT

OnActivateAsync B - P1
Device type: PC Application version: 4.0

10/8/2016 10:50:02 AM - CUSTOM EVENT

Timer1 - P1
Device type: PC Application version: 4.0

10/8/2016 10:50:00 AM - CUSTOM EVENT

RunAsync
Device type: PC Application version: 4.0

Reminders

Deallocate (GC)

Reminder

Activation

Constructor

10/8/2016 2:51:00 PM - CUSTOM EVENT

ReceiveReminderAsync - PR2, MyReminder, 164
Device type: PC Application version: Reminder 4.0

10/8/2016 2:50:42 PM - CUSTOM EVENT

OnDeactivateAsync - PR1
Device type: PC Application version: Reminder 4.0

10/8/2016 2:50:17 PM - CUSTOM EVENT

ReceiveReminderAsync - PR3, MyReminder, 54
Device type: PC Application version: Reminder 4.0

10/8/2016 2:49:59 PM - CUSTOM EVENT

ReceiveReminderAsync - PR2, MyReminder, 163
Device type: PC Application version: Reminder 4.0

10/8/2016 2:48:59 PM - CUSTOM EVENT

ReceiveReminderAsync - PR2, MyReminder, 162
Device type: PC Application version: Reminder 4.0

10/8/2016 2:47:59 PM - CUSTOM EVENT

ReceiveReminderAsync - PR2, MyReminder, 161
Device type: PC Application version: Reminder 4.0

10/8/2016 2:47:17 PM - CUSTOM EVENT

ReceiveReminderAsync - PR3, MyReminder, 53
Device type: PC Application version: Reminder 4.0

10/8/2016 2:46:58 PM - CUSTOM EVENT

ReceiveReminderAsync - PR2, MyReminder, 160
Device type: PC Application version: Reminder 4.0

10/8/2016 2:45:58 PM - CUSTOM EVENT

ReceiveReminderAsync - PR2, MyReminder, 159
Device type: PC Application version: Reminder 4.0

10/8/2016 2:44:58 PM - CUSTOM EVENT

ReceiveReminderAsync - PR2, MyReminder, 158
Device type: PC Application version: Reminder 4.0

10/8/2016 2:44:48 PM - CUSTOM EVENT

ReceiveReminderAsync - PR1, MyReminder, 1
Device type: PC Application version: Reminder 4.0

10/8/2016 2:44:48 PM - CUSTOM EVENT

OnActivateAsync - PR1
Device type: PC Application version: Reminder 4.0

10/8/2016 2:44:47 PM - CUSTOM EVENT

Reminder1 - PR1
Device type: PC Application version: Reminder 4.0

10/8/2016 12:08:49 PM - CUSTOM EVENT

ReceiveReminderAsync - PR2, MyReminder, 3
Device type: PC Application version: Reminder 4.0

10/8/2016 12:07:49 PM - CUSTOM EVENT

ReceiveReminderAsync - PR3, MyReminder, 0
Device type: PC Application version: Reminder 4.0

10/8/2016 12:07:48 PM - CUSTOM EVENT

ReceiveReminderAsync - PR2, MyReminder, 2
Device type: PC Application version: Reminder 4.0

10/8/2016 12:06:48 PM - CUSTOM EVENT

ReceiveReminderAsync - PR2, MyReminder, 1
Device type: PC Application version: Reminder 4.0

10/8/2016 12:05:47 PM - CUSTOM EVENT

ReceiveReminderAsync - PR2, MyReminder, 0
Device type: PC Application version: Reminder 4.0

10/8/2016 12:05:02 PM - CUSTOM EVENT

OnDeactivateAsync - PR2
Device type: PC Application version: Reminder 4.0

10/8/2016 12:04:48 PM - CUSTOM EVENT

TKRegisterReminderAsync - PR3, 180
Device type: PC Application version: Reminder 4.0

10/8/2016 12:04:48 PM - CUSTOM EVENT

OnActivateAsync - PR3
Device type: PC Application version: Reminder 4.0

10/8/2016 12:04:47 PM - CUSTOM EVENT

Reminder1 - PR3
Device type: PC Application version: Reminder 4.0

10/8/2016 12:04:47 PM - CUSTOM EVENT

TKRegisterReminderAsync - PR2, 60
Device type: PC Application version: Reminder 4.0

Events: actor to client communication

Not a tool to communicate BETWEEN actors. **Best effort only – no guaranteed delivery!**

Can be used between services (statefull / stateless) and actors. Example: Actor send event to Web

Actor:

```
public interface ITKProgressEvents : IActorEvents {  
    2 references  
    void ProgressUpdated(string message);  
}  
  
3 references  
public interface ITKActorEventSource : IActor, IActorEventPublisher<ITKProgressEvents> {  
    2 references  
    Task<int> StartLongCalculationAsync(int param);  
}
```

```
var ev = GetEvent<ITKProgressEvents>();  
ev.ProgressUpdated($"START");  
for (int i = 0; i < param; i++) {  
    ev.ProgressUpdated($"Progress: {i} / {param}");  
    await Task.Delay(1000);  
}
```

Client (WebApi in this case):

```
ITKActorEventSource actor = ActorProxy.Create<ITKActorEventSource>  
(new ActorId(id), "fabric:/TK_EventsAndService");  
await actor.SubscribeAsync(new TKProgressEvents());  
await ...
```

```
internal class TKProgressEvents : ITKProgressEvents {  
    3 references  
    public void ProgressUpdated(string message) {  
        ServiceEventSource.Current.Message($"ProgressUpda...")  
    }  
}
```

Demo

„Actor to client”

Debugging + Monitoring +
Diagnostics + Health ...

Debugging – also remotely

The image shows two instances of the Cloud Explorer interface and an 'Attach to Process' dialog box.

Cloud Explorer (Left): Shows the 'Microsoft Azure' resource type for 'SFTools'. Under 'Service Fabric', there are several options: Refresh, Open in Portal, Enable Debugging (selected), Disable Debugging, Attach Debugger (highlighted), Open Service Fabric Explorer, Enable Streaming Traces, and Disable Streaming Traces.

Cloud Explorer (Middle): Similar to the left, but the 'Attach Debugger' option is selected instead of 'Enable Debugging'.

Attach to Process Dialog: This dialog is titled 'Attach to Process' and specifies a 'Debug scope: Azure (5 machines)'. It includes a 'Scope Description' section stating: 'Attach the debugger by selecting one or more processes currently running under this role. Multiple instances may be currently running the processes with the same name, in which case breakpoints will be hit for each instance that receives traffic.' The 'Attach to' dropdown is set to 'Automatic'. The 'Available Processes' table lists various processes:

Process	ID	Title	Type	User Name	Session
FileStoreService.exe (5)				NETWORK SERVICE	0
LogonUI.exe (5)			x64	SYSTEM	1
lsass.exe (5)			x64	SYSTEM	0
Microsoft.VisualStudio.WindowsAz...			Managed, x64	SYSTEM	0
MonAgentHost.exe (5)			x64	SYSTEM	0
MonAgentManager.exe (5)			x64	SYSTEM	0
msdtc.exe (5)			x64	NETWORK SERVICE	0
MyStatefulService.exe (5)			Managed, x64	NETWORK SERVICE	0
MyStatelessService.exe (5)			Managed, x64	NETWORK SERVICE	0
rdpclip.exe (5)			x64		2
rundll32.exe (15)			x64	SYSTEM	0
ServerManager.exe	5732		Managed, x64	SFT-3\sftoolsuser	2
ServiceFabricNodeBootstrapAgent....			Managed, x64	SYSTEM	0

At the bottom right of the dialog are 'Refresh', 'Attach', and 'Cancel' buttons.

Better use bigger machines! (D2, D3 etc)

Diagnostics and Troubleshooting

Detailed System Optics

- Repair suggestions. Examples: Slow RunAsync cancellations, RunAsync failures
- All important events logged. Examples: App creation, deploy and upgrade records. All Actor method calls.

Custom Application Tracing

- ETW == Fast Industry Standard Logging Technology
- Works across environments. Same tracing code runs on devbox and also on production clusters on Azure.
- Easy to add and system appends all the needed metadata such as node, app, service, and partition.
- Application Insight – why not
- Custom service health

Choice of Tools

- Visual Studio Diagnostics Events Viewer
- Windows Event Viewer
- Windows Azure Diagnostics + Operational Insights
- Easy to plug in your preferred tools: Kibana, Elasticsearch and more

Custom Service Health

System.Fabric.Health

```
HealthInformation healthInformation =  
    new HealthInformation("ServiceCode", "StateDictionary", HealthState.Error);
```

Statefull:

```
this.Partition.ReportReplicaHealth(healthInformation);
```

Stateless:

```
this.Partition.ReportInstanceHealth(healthInformation);
```

Application, DeployedApplication, DeployedServicePackage:

```
var activationContext = FabricRuntime.GetActivationContext();  
activationContext.ReportApplicationHealth(healthInformation);
```

The screenshot shows the Microsoft Azure Service Fabric Explorer interface. At the top, there are three status buttons: 'OK' (green), 'Warning' (yellow), and 'Error' (red). Below them is a search bar labeled 'Search Cluster'. The main pane displays a hierarchical tree of cluster health. The 'Cluster' node is marked as 'Error'. Under 'Applications', the 'System' application is marked as 'Error'. Within 'System', the 'TestApplicationType' application is marked as 'Error'. Under 'TestApplicationType', the 'fabric/TestApplication' service is marked as 'Error'. Within 'fabric/TestApplication', the 'Stateful1' instance is marked as 'Error'. The 'Nodes' node under 'System' has three children: '_Node_0 (ActiveSecondary)', '_Node_1 (ActiveSecondary)', and '_Node_4 (Primary)', which is highlighted with a blue selection bar. To the right of the tree, there are tabs for 'ESSENTIALS' and 'DETAILS'. Under 'ESSENTIALS', the 'Id' is listed as '131027240552591593', 'Health State' is 'Error', and 'Role' is 'Primary'. Under 'DETAILS', the 'Node Name' is '_Node_4' and 'Status' is 'Ready'. At the bottom right, there is a section titled 'UNHEALTHY EVALUATIONS' with a table showing one event: 'Event' is 'Error', 'Kind' is 'Health State', 'Description' is 'Error event: SourceId='ServiceCode', Property='StateDictionary'', and 'Event' is 'Error'.

Service Placements & balancing & Metrics

```
<StatefulServiceType ServiceTypeName="MiscTelemetryType" HasPersistedState="true">
  <LoadMetrics>
    <!--Default: PrimaryCount, ReplicaCount, Count -->
    <!--Custom-->
    <LoadMetric Name="LARGEMEM" Weight="High" DefaultLoad="100"/>
    <!--Dynamic-->
    <LoadMetric Name="BUSINESSOPCOUNT" Weight="Low" DefaultLoad="1" PrimaryCount="1" ReplicaCount="1"/>
  </LoadMetrics>
</StatefulServiceType>
```

```
<Section Name="MetricBalancingThresholds">
  <Parameter Name="MetricName1" Value="2"/>
  <Parameter Name="MetricName2" Value="3.5"/>
</Section>
```

```
<StatelessServiceType ServiceTypeName="PublicGatewayType">
  <PlacementConstraints>NodeType == NodeType1 || NodeType == NodeType2</PlacementConstraints>
</StatelessServiceType>
```

```
<Section Name="MetricActivityThresholds">
  <Parameter Name="Memory" Value="1536"/>
</Section>
```

See:
[Here](#), [here](#), [here](#), [here](#) and [here](#)
(Obligatory lecture!)

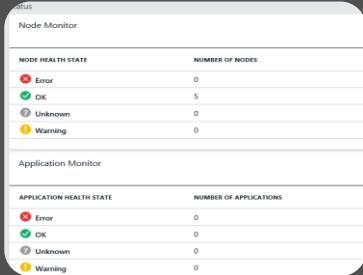


Demo

ETW, Monitoring, custom metrics, health etc.

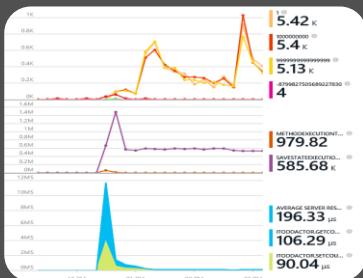
Monitoring your Services

Visibility into how your services are doing when running in production



Health status monitoring

- Built-in health status for cluster and services
- Flexible and extensible health store for custom app health reporting
- Allows continuous monitoring for real-time alerting on problems in production



Performance and stress response

- Rich built-in metrics for Actors and Services programming models
- Easy to add custom application performance metrics

[Show dashboard in OMS](#)

Demo

Cluster info

Application updates,
Configuration, Scale, Backup

App Lifecycle

```
Connect-ServiceFabricCluster localhost:19000
```

```
Copy-ServiceFabricApplicationPackage <folder> `
```

```
    -ImageStoreConnectionString
```

```
file:C:\SfDevCluster\Data\ImageStoreShare -
```

```
ApplicationPackagePathInImageStore TK_ETWAppInsightMetrics
```

```
Register-ServiceFabricApplicationType TK_ETWAppInsightMetrics
```

```
New-ServiceFabricApplication fabric:/TK_ETWAppInsightMetricsType
```

```
TK_ETWAppInsightMetricsType 1.0.0
```

```
<deploy another version>
```

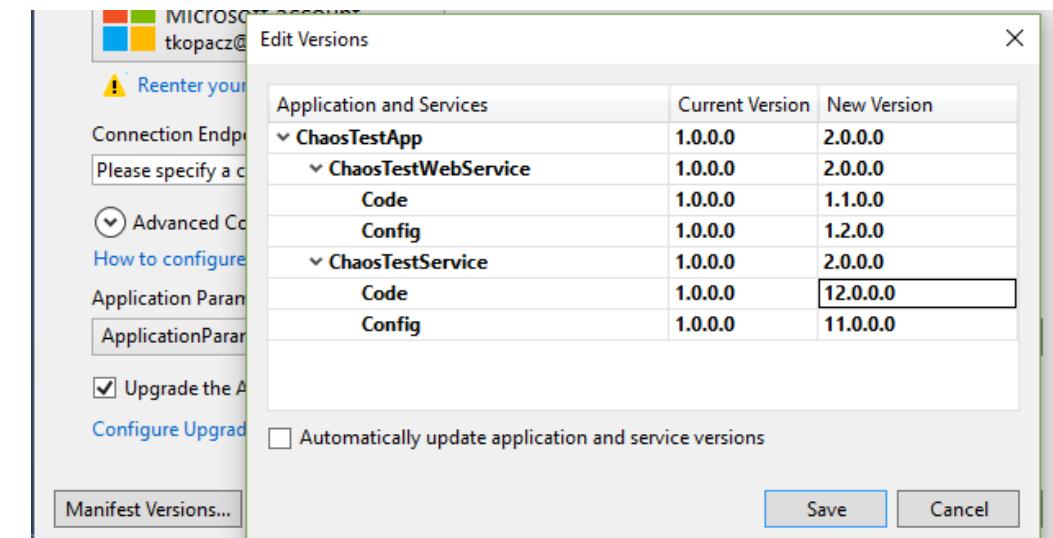
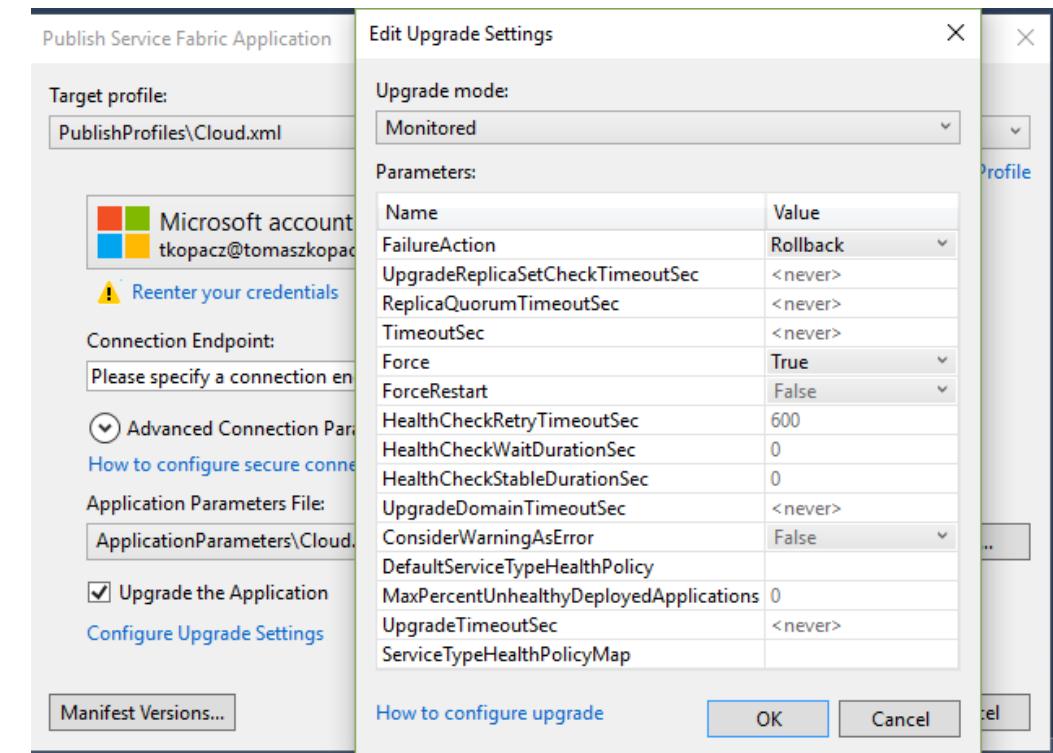
```
Start-ServiceFabricApplicationUpgrade -ApplicationName  
fabric:/TK_ETWAppInsightMetricsType -ApplicationTypeVersion 2.0.0  
-HealthCheckStableDurationSec 60 -UpgradeDomainTimeoutSec 1200 -  
UpgradeTimeout 3000 -FailureAction Rollback -Monitored
```

```
Remove-ServiceFabricApplication
```

```
fabric:/TK_ETWAppInsightMetricsType -Force
```

```
Unregister-ServiceFabricApplicationType
```

```
TK_ETWAppInsightMetricsType -ApplicationTypeVersion 1.0.0 -Force
```



Configuration

This is also independent
“upgrade” element!

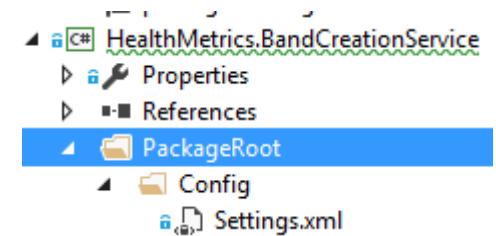
Code:

```
ConfigurationSettings configSettings =  
FabricRuntime.GetActivationContext().  
GetConfigurationPackageObject("Config").Settings;
```

```
KeyedCollection<string, ConfigurationProperty>  
serviceParameters = configSettings.Sections["HealthMetrics.BandCreationService.Settings"].Parameters
```

```
this.NumberOfCreationThreads = int.Parse(serviceParameters["NumberOfCreationThreads"].Value);
```

Can be also JSON etc – XML is build-in



```
<?xml version="1.0" encoding="utf-8" ?>  
<Settings xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmlns:xsi="http://www.w3.org/2001/  
    <Section Name="HealthMetrics.BandCreationService.Settings">  
        <Parameter Name="BandActorServiceName" Value="BandActorService" />  
        <Parameter Name="DoctorActorServiceName" Value="DoctorActorService" />  
        <Parameter Name="NumberOfCreationThreads" Value="" MustOverride="true"/>  
        <Parameter Name="MaxBandsToCreatePerServiceInstance" Value="" MustOverride="true" />  
        <Parameter Name="CountyFileName" Value="HealthMetrics.Common.csv" />  
        <Parameter Name="PeopleFileName" Value="Names.csv" />
```

Scale (Azure)

To remove

[Disable-ServiceFabricNode](#) & [Get-ServiceFabricNode](#) – to remove all reliable services / actors from VM
Reduce number of VM in VMSS (usually using ARM template). Important – no less than FD / UD (5)
Run [Remove-ServiceFabricNodeState](#) – unregister in Service Fabric Explorer (FM Service to be precise)

To add

Increase number of VM in VMSS

Duration – for example 2 hours 49 minutes 1 second, depends on many factors!

Autoscale – the same mechanism as in other VMSS

Remember! Reliability: one machine at the time
(bronze, 3 replicas)

Or more in Silver (5 replicas), Gold (7 replicas), Platinum (9 replicas); keep in mind – QUORUM!

Btw: durability: Bronze – no extra privileges, Gold – Azure infrastructure jobs can be paused for max 2h

Backup API (for statefull services)

To do a backup:

Invoke backup process with callback and type of backup (Full / Incremental):

```
var backupDescription = new BackupDescription(BackupOption.Full, this.BackupCallbackAsync);
await this.BackupAsync(backupDescription, TimeSpan.FromHours(1), cancellationToken);
```

BackupCallbackAsync will receive BackupInfo with folder containing ready to save data from selected node / partition

(C:\SfDevCluster\Data_\App_\Node_3\TK_2016MainSFFunctionsType_App8\work\157ba701-f94e-4a68-85d0-e6c50e172c03\131212795996136678\trbackup)

To restore

Override OnDataLossAsync(RestoreContext restoreCtx...)

Trigger restore

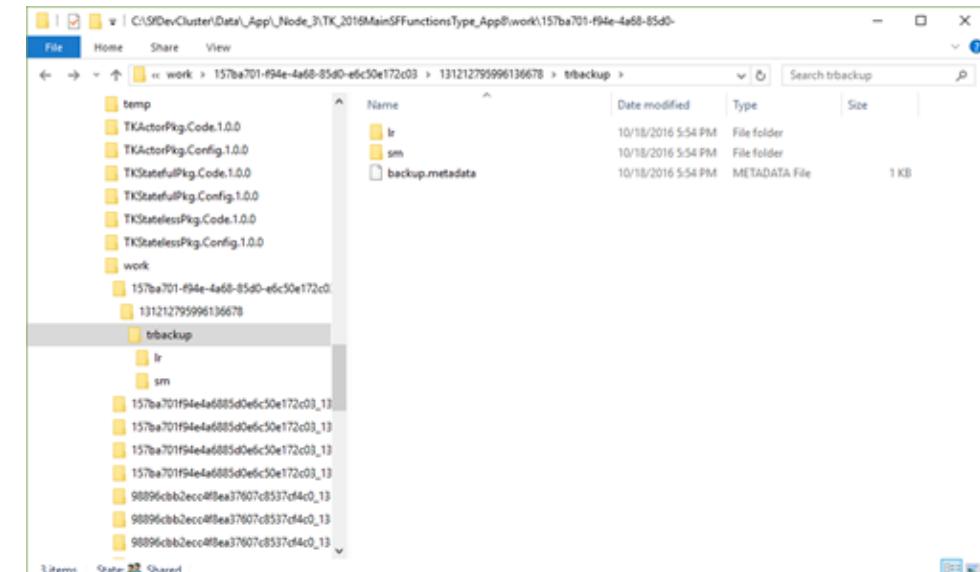
Partition data loss – automatically

Deleted or lost/corrupted service: restore service, then for each partition:

```
FabricClient.TestManagementClient.StartPartitionDataLossAsync
```

Test: Invoke-ServiceFabricPartitionDataLoss

Actors: see IActorStateProvider



Demo

Backup. Look at the code!
Scale – video, sorry (too long!)

File Edit Session View Remote Desktops Tools Help

TKRDG

- Azure
 - ADFS
 - BTS
 - NetCorePerf
 - pltkw3 - standalone
 - pltkw3devstand.northeurope
 - pltkw3 - roslyn in netcore
 - pltkw3eastus2 - datalake up
 - pltkw3w16.westeurope.cloudapp.azure.com
 - TKW
 - TKW2 (TKC.LOCAL domain)
 - TKW3 (TKW3.LOCAL)
 - VNET_DEMO
 - DOM2
 - MISC

TK_2016MainSFFunctions - Microsoft Visual Studio (Administrator)

File Edit View Project Build Debug Team Tools Architecture Test Load Test Analyze Window Help

Debug Any CPU TK_2016MainSFFunctions Start

Cloud Explorer Reenter your credentials Microsoft Azure Resource Types Search for resources

ActorVsStatefullReadOnly.webtest ActorVsStatefullReadOnly.loadtest

ActorVsStatefullReadOnly

- http://pltkw3sfcluster.westeurope.cloudapp.azure.com:8080/api/sf/ActorGet
- http://pltkw3sfcluster.westeurope.cloudapp.azure.com:8080/api/sf/TKStatefulGet

Validation Rules

- Response URL
- Response Time Goal

Solution Explorer

Search Solution Explorer (Ctrl+Shift+F)

Solution 'TK_2016MainSFFunctions' (7 projects)

- Solution Items
 - Local.testsettings
 - MainSample.txt
- TestMainSF
 - Properties
 - References
 - ActorVsStatefull.loadtest
 - ActorVsStatefull.webtest
 - ActorVsStatefullReadOnly.loadtest
 - ActorVsStatefullReadOnly.webtest
- TK_2016MainSFFunctions
 - Properties
 - Services
 - ApplicationPackageRoot
 - ApplicationParameters
 - PublishProfiles
 - Scripts
 - packages.config
- TKActor
 - Properties
 - References
 - PackageRoot
 - ActorEventSource.cs
 - App.config

Solution Explorer Team Explorer

Properties

Microsoft.VisualStudio.TestTools.WebStress.Request

Cache Control	False
Encoding	utf-8
Expected HTTP Status	0
Expected Response URL	
Follow Redirects	True
Ignore HTTP Status Codes	False
Method	GET
Parse Dependent Requests	True
Record Results	True
Reporting Name	

Cache Control
True indicates that the requests are cached. False indicates that the requests are not cached.

Test Results

tkopacz@PLTKW3DEVSTAND 201

Test run aborted Results: 0/1 passed; Item(s) checked: 1

Action	Result	Test Name	ID	Error Message
Aborted	Aborted	ActorVsStatefullReadonly	z:\s\azurefy15tk\0:	

Actions Properties

What do you like about this tool?
What don't you like or feel is missing?

Package Manager Console Error List Output Find Results 1 Test Results

Ready Publish

Demo

Chaos testing service

ChaosTestApplication - Microsoft Visual Studio

File Edit View Project Build Debug Team XML Data Lake Tools Architecture Test Web Essentials R Tools Analyze Window Help

Quick Launch (Ctrl+Q) Tomasz Kopacz (PRV)

Cloud Explorer DefaultController.cs ServiceManifest.xml

Microsoft Azure Resource Types Search for resources

(Local) HACKATON01 (tkopacz@microsoft.com) TK - HDInsight PW (tkopacz@microsoft.com) TK pltkw3 (tkopacz@microsoft.com)

ServiceManifest.xml

```
10 <StatelessServiceType ServiceTypeName="ChaosTestWebServiceType" />
11 </ServiceTypes>
12
13 <!-- Code package is your service executable. -->
14 <CodePackage Name="Code" Version="1.0.0.0">
15   <EntryPoint>
16     <ExeHost>
17       <Program>ChaosTest.WebService.exe</Program>
18     </ExeHost>
19   </EntryPoint>
20 </CodePackage>
21
22 <!-- Config pac
independen
23 <ConfigPackage
24 <Resources>
25   <Endpoints>
26     <!-- This e
listen
replic
27       <Endpoint N
28         <Endpoint N
29       </Endpoints>
30     </Resources>
31 </ServiceManifest
32
33
34
```

Publish Service Fabric Application

Target profile: PublishProfiles\Cloud.xml

Microsoft tkopacz@microsoft.com

Connection Endpoint: pltkw3sfcluster.westeurope.cloudapp.azure.com:19000

Advanced Connection Parameters

How to configure secure connections

Application Parameters File: ApplicationParameters\Cloud.xml

Upgrade the Application

Configure Upgrade Settings

Manifest Versions... Publish Cancel

Output

Show output from: General

Actions Properties

What do you like about this tool?
What don't you like or feel is missing?

SQL S... Test E... Toolbar... Server... Perform... JSON... Cloud... Task Runner Explorer Web Publish Activity Error List Output Find Results 1 Immediate Window Bookmarks Test Results

Solution Explorer Solution 'ChaosTestApplication' (4 projects)

- ChaosTest.ChaosService
- ChaosTest.Common
- ChaosTest.WebService
 - Properties
 - References
 - App_Start
 - Controllers
 - DefaultController.cs
 - Extensions
 - PackageRoot
 - Config
 - ServiceManifest.xml
 - wwwroot
 - App.config
 - ChaosTestWebServiceConfigSettings.cs
 - packages.config
 - Program.cs
 - ServiceEventSource.cs
 - Startup.cs
 - WebService.cs
- ChaosTestApplication

Syntax Visualizer Solution Explorer Team Explorer

Properties ChaosTestApplication Project Properties

Application Debug Mode Remove Application

Application URL

Project File ChaosTestApplication.sproj

Project Folder C:\AzureFY15TK\05PaaS_Service

Application Debug Mode

Indicates behavior of application deployment as part of debugging. Remove Application - The application will be re...

Similar action - PowerShell

```
$timeToRun = 60  
$maxStabilizationTimeSecs = 180  
$concurrentFaults = 3  
$waitTimeBetweenIterationsSec = 60
```

```
Invoke-ServiceFabricChaosTestScenario -TimeToRunMinute $timeToRun -  
MaxClusterStabilizationTimeoutSec $maxStabilizationTimeSecs -MaxConcurrentFaults  
$concurrentFaults -EnableMoveReplicaFaults -WaitTimeBetweenIterationsSec  
$waitTimeBetweenIterationsSec
```

```
$serviceName = "fabric:/WordCount/WordCountService"
```

```
Invoke-ServiceFabricFailoverTestScenario -TimeToRunMinute  
$timeToRun -MaxServiceStabilizationTimeoutSec $maxStabilizationTimeSecs -  
WaitTimeBetweenFaultsSec $waitTimeBetweenFaultsSec -ServiceName $serviceName -  
PartitionKindUniformInt64 -PartitionKey 1
```

Short: Service Fabric on
premise, for free

Options

I am excited to announce today that Azure Service Fabric for Windows Server will be generally available for download at no cost. With today's announcement, customers can now provision Service Fabric clusters in their own data centers or other cloud providers and run production workloads with the option to purchase support for ultimate confidence. One such customer is [Owners.com](#), an online platform that gives consumers a convenient

Same code can be run on:

Service Fabric as a part of Azure Stack

Service Fabric on single VM on anything (Hyper-V,
physical, XEN)

3 VMs, 5 VMs, etc...

Materials here

```
PS C:\Microsoft.Azure.ServiceFabric.WindowsServer.5.3.204.9494> .\CreateServiceFabricCluster.ps1 -ClusterConfigFilePath ..\ClusterConfig.json -AcceptEULA
Creating Service Fabric Cluster...
If it's taking too long, please check in Task Manager details and see if Fabric.exe for each node is running. If not, please look at: 1. traces in DeploymentTraces directory and 2. traces in FabricLogRoot configured in ClusterConfig.json. Trace folder already exists. Traces will be written to existing trace folder: C:\Microsoft.Azure.ServiceFabric.WindowsServer.5.3.204.9494\DeploymentTraces
Running Best Practices Analyzer...
Best Practices Analyzer completed successfully.
Creating Service Fabric Cluster...
Processing and validating cluster config.
Adding Section=Setup with 2 parameters.
Adding Section=Security with 6 parameters.
Adding Section=FileStoreService with 5 parameters.
Adding Section=FailoverManager with 4 parameters.
Adding Section=Diagnostics with 5 parameters.
Adding Section=WinFabEtlFile with 3 parameters.
Adding Section=WinFabCrashDump with 4 parameters.
Adding Section=WinFabPerfCtrFolder with 4 parameters.
Adding Section=FileShareWinFabEtw with 5 parameters.
Adding Section=FileShareWinFabCrashDump with 5 parameters.
Adding Section=FileShareWinFabPerfCtr with 5 parameters.
Adding Section=UpgradeOrchestrationService with 5 parameters.
Adding Section=ClusterManager with 3 parameters.
Adding Section=NamingService with 3 parameters.
Adding Section=ImageStoreService with 3 parameters.
Adding Section=FaultAnalysisService with 3 parameters.
Adding Section=Management with 1 parameters.
Adding Section=Federation with 1 parameters.
Adding Section=Hosting with 3 parameters.
Adding Section=HttpGateway with 1 parameters.
Adding Section=Trace/Etw with 1 parameters.
Adding Section=ReconfigurationAgent with 1 parameters.
Baseline upgrade started.
Configuring nodes.
Default installation directory chosen based on system drive of machine pltkw3srvfab0.TKW3.LOCAL
Copying installer to all machines.
Configuring machine pltkw3srvfab0.TKW3.LOCAL
```

Steps:

1. Setup ClusterConfig.json (nodeTypes, nodes, faultDomain, upgradeDomain, reliabilityLevel)
2. Allow communication between nodes (file sharing – simplest)
3. .\CreateServiceFabricCluster.ps1 -ClusterConfigFilePath ..\ClusterConfig.json –AcceptEULA
4. **THOSE ARE REALLY ALL STEPS!**

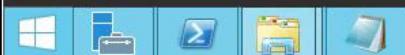
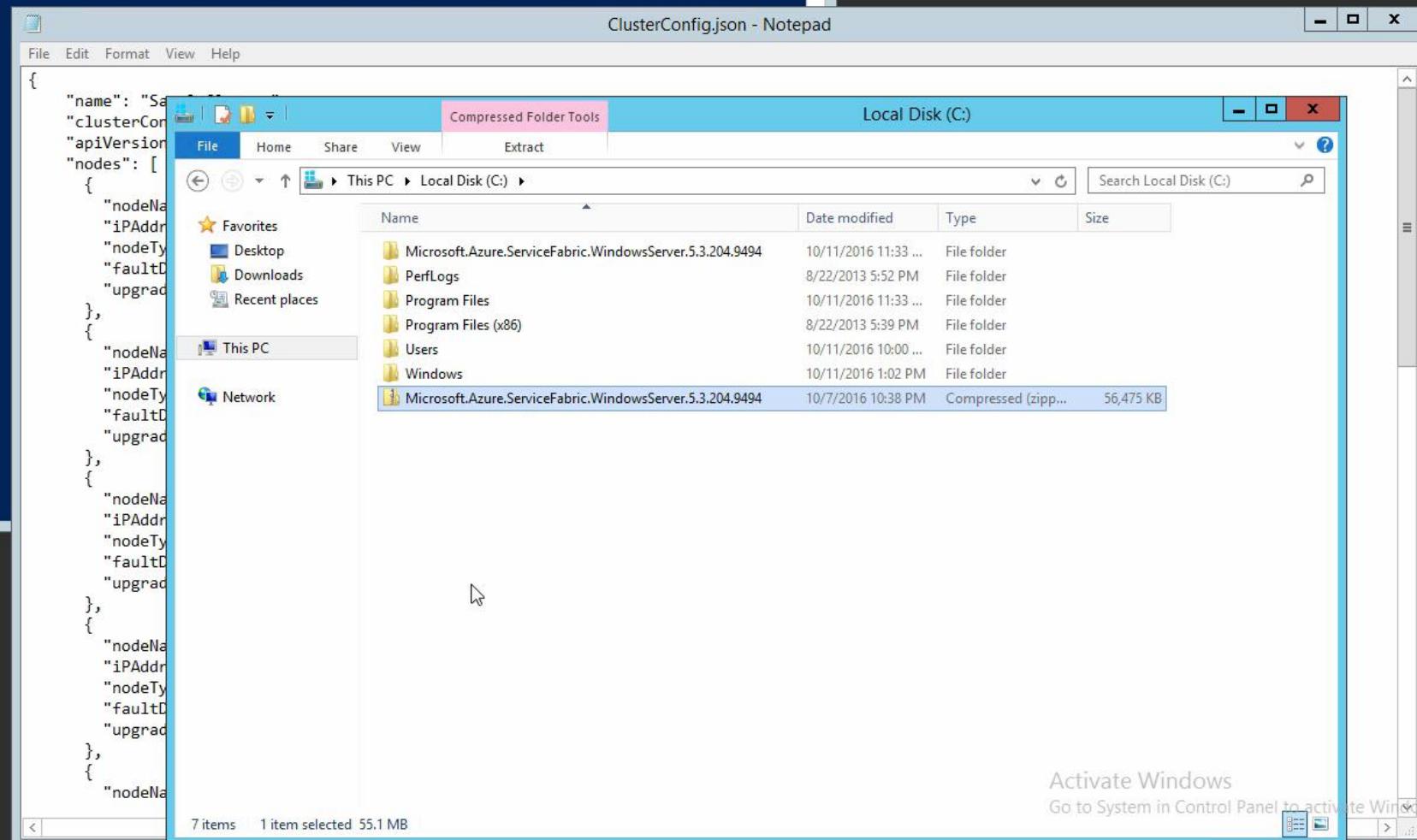
„Demo“



Recycle Bin

Administrator: Windows PowerShell

```
PS C:\Microsoft.Azure.ServiceFabric.WindowsServer.5.3.204.9494> .\CreateServiceFabricCluster.ps1 -ClusterConfigFilePath .\ClusterConfig.json -AcceptEULA
```



Security

Options

Two aspects:

Cluster Authentication (node to node)

Server Authentication (management endpoint)

[First: Internal Gateway in VNET \(SF not visible\)](#)

[Host: Azure](#)

Key vault for storing certificates

Azure Active Directory for client Authentication

Service Fabric Management URL need to be in Replay URL – as in normal Web App registered in AAD

[Doc and procedures here](#)

[Host: On Premise](#)

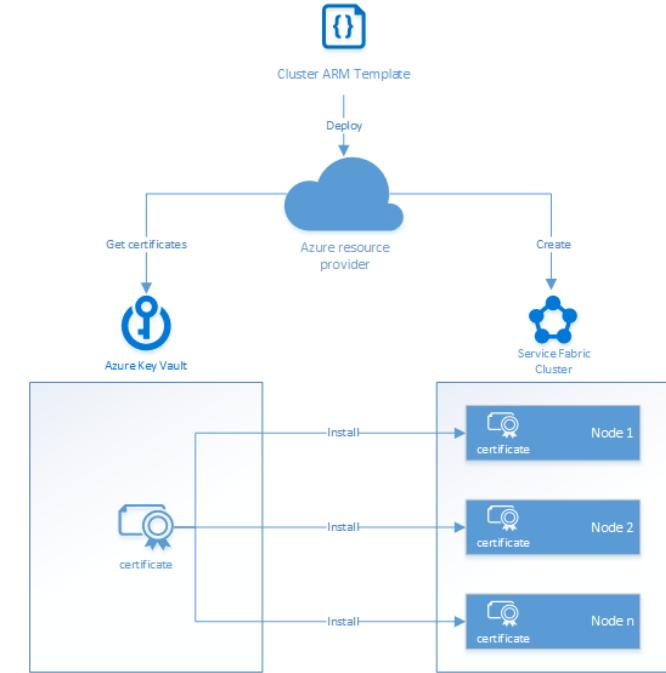
(Unsecure)

Windows Authentication (domain required!)

X509 certificate (I think that this is best options – similar to solution used in Azure)

[Remember there is also app in SF \(and app need to be secured as well!\)](#)

Web Apps is much more convenient for UI. Service Fabric can host backend (.NET Core, API, ...)



Summary

„Summary”

Service Fabric is a great platform for building applications

And for hosting microservices as well ☺

Not limited to REST-based communication model

Many internal capabilities

(and – please do not „fight” with them!)

But – DDD, CQRS, correct architecture – as usual!

WebApi as a frontend

Materials:

<https://github.com/tkopacz/2016DeveloperDays>

Additional links – mainly architecture

„Required“ Reading

[Domain-Driven Design, Tackling Complexity in the Heart of Software](#)
[Implementing Domain-Driven Design](#)

Online Resources

[Design Patterns, CQRS](#)

<http://martinfowler.com/>, [dddcommunity.org](#)

[Eric Evans on DDD: Strategic Design](#), [Domain Driven Design Quickly](#), [DDD: putting the model to work](#)

[Patterns and Practices: CQRS](#), [CQRS Journey](#), [Introduction to CQRS](#), [Azure guidance \(general\)](#)

[Greg Young on CQRS \(YouTube\)](#), [7 Hours of Greg Young \(YouTube\)](#)

Courses

[DDD fundamentals](#), [DDD in practice](#), [Modern architecture: CQRS + ES](#)

Samples (bigger one!)

[IoT samples](#), [Java sample](#), [Web Reference App](#)

<https://github.com/CESARDELATORRE/MyWorld> (work in progress!)



net.developerdays.pl
@DeveloperDaysPL

Sponsors and Partners

Strategic Sponsors



Gold Sponsors



Silver Sponsors



Fast Reports
Reporting must be Fast!



JETBRAINS THE DRIVE
TO DEVELOP

