



Serverless orchestration

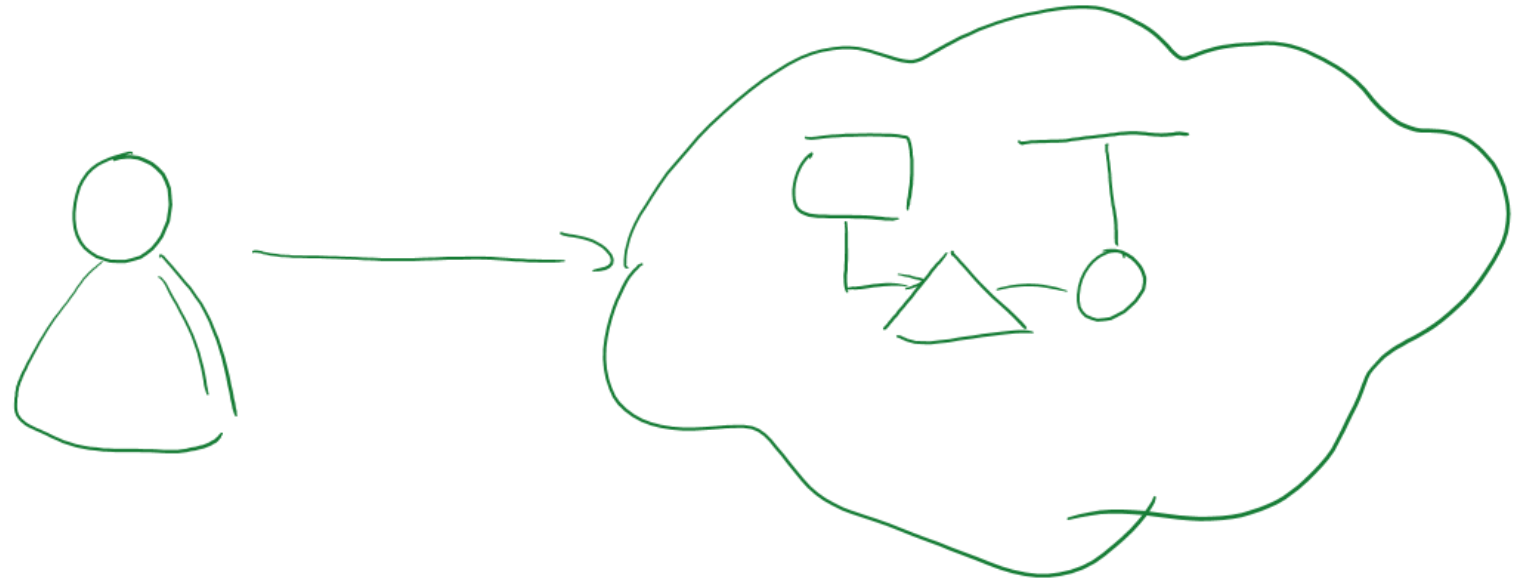
How I develop a membership administration



Thomas Tomow

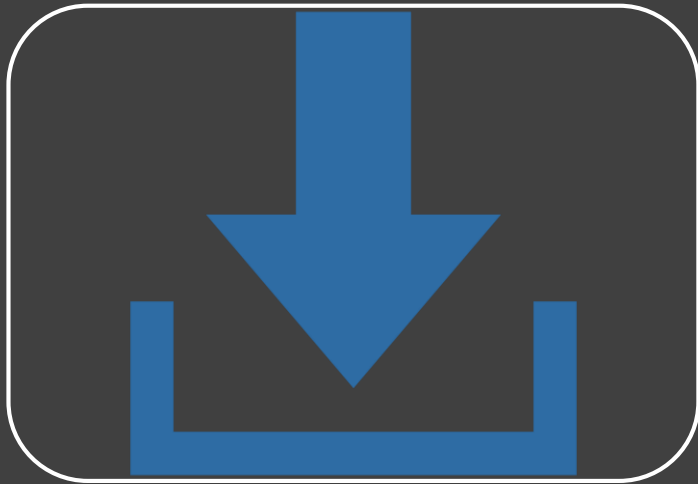
- Azure MVP since 2017
- Working at  devoteam | **Alegri**
- Focusing on IoT, Intelligent Application, ...
- Love community [#communityrocks]
 - Owner of Azure Meetup Konstanz & Region
- Doing Sports
 - Karate (4. DAN)

Intro



What we're going to get

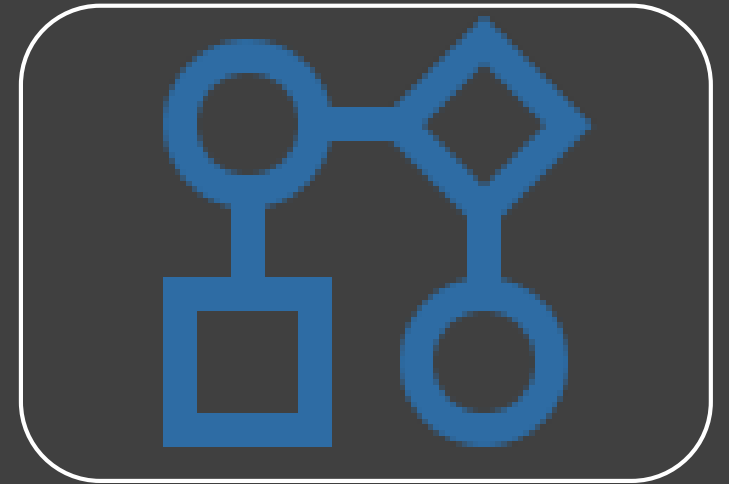
What is a „Durable Function“?



Build on top of
Azure Functions



Adding „State“ to
„serverless“
environment



Define and control
Workflow in code

Azure Functions – an overview?

Functions „as a Service“



Pure function is:

- For same Input the same output
- There are no sideeffects (logging, io,...)



Azure Functions – an overview?



Triggers/Input

- Timer
- Queue messages
- HTTP Requests



Languages

- C#
- F#
- JavaScript
- Java

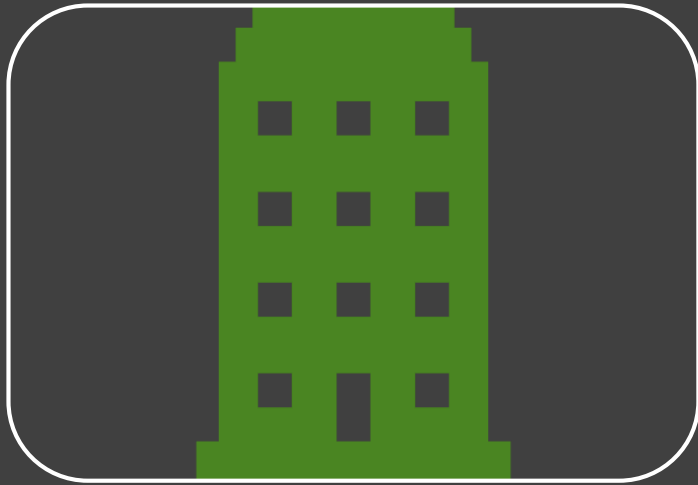


Outputs

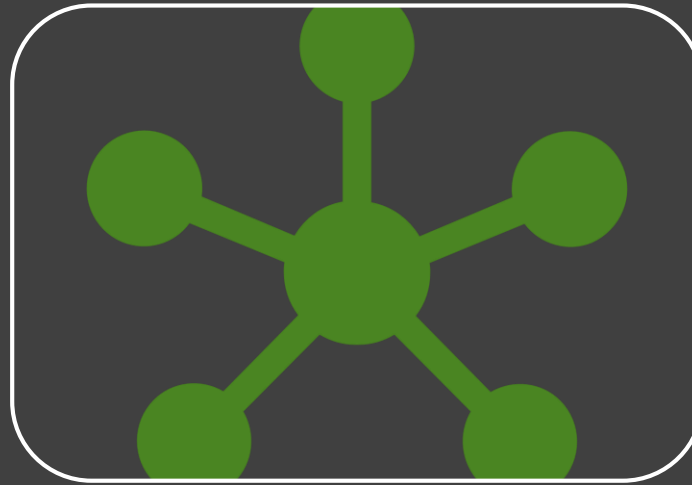
- Blob storage
- Queue
- Event Hub

What means „Serverless“?

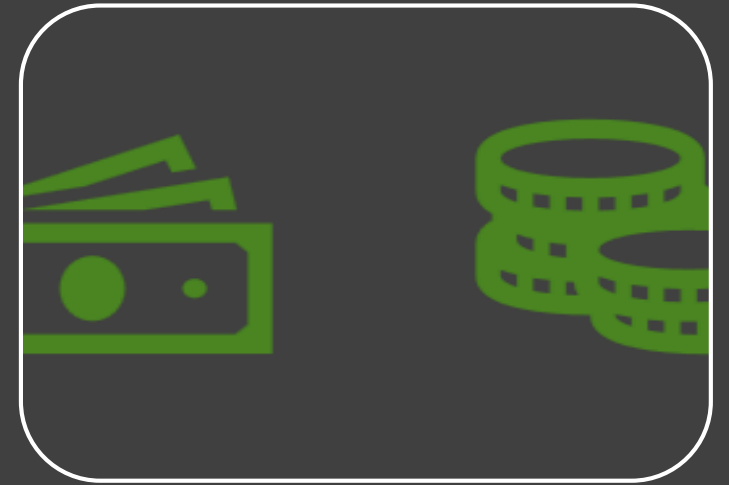
Cloud-computing execution model



Provider acts as server
managing allocation of
machine resources



Scaling managed

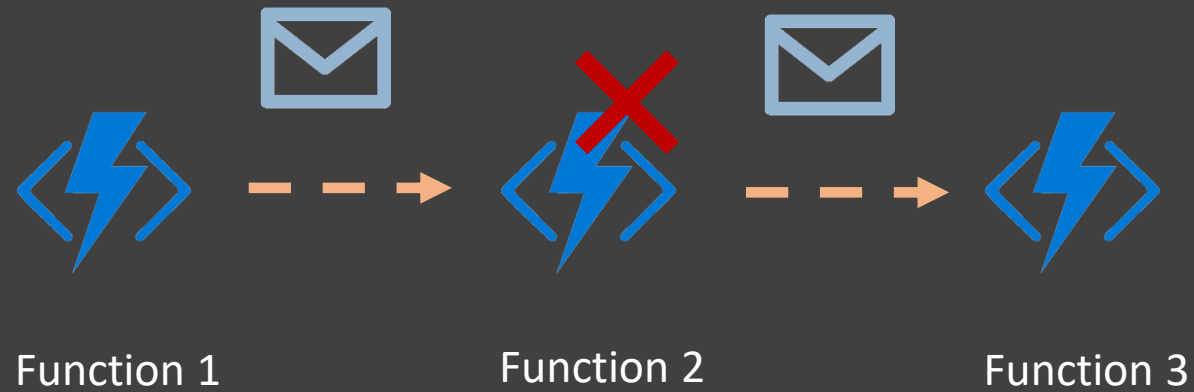


Pricing based on usage

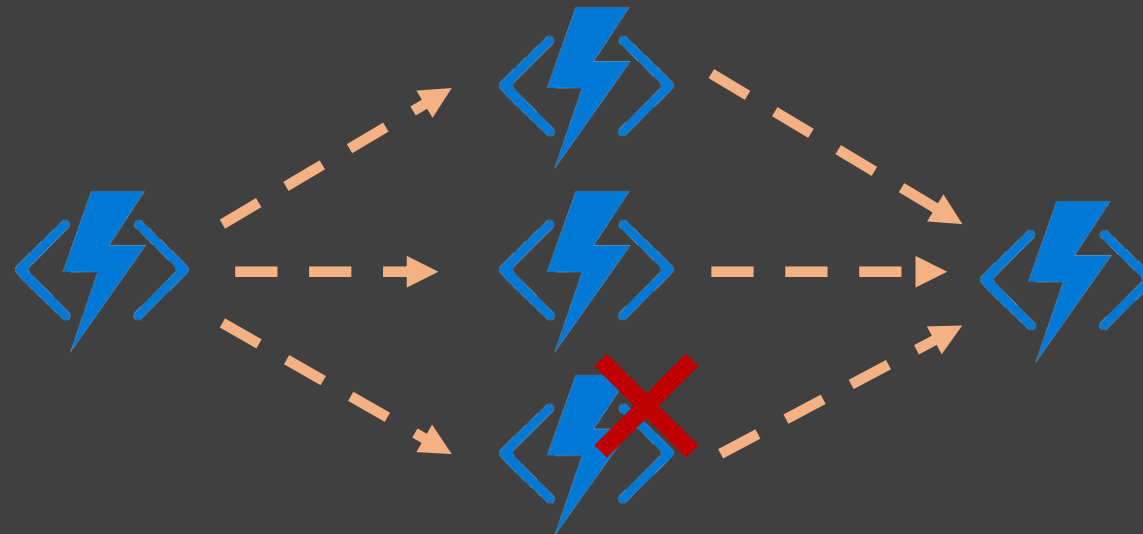
- Consumption plan
- App Service Plan

Why Durable?

Function Chaining
Workflow



Fan-Out
Fan-In
Workflow



What are the benefits of durable functions?



- Workflow defined by code
- Easy to implement
- Handling Exception in one point
- Tracking progress or cancel orchestration
- Managed state-handling

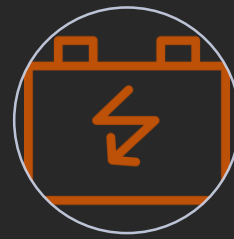
Key Concepts

- defines workflow in code
- Does actions like
 - Waiting, calling activities, sub-Orchestration
- Uses `OrchestrationClientBinding`
- Could be Azure Function
- Triggered by `ActivityTrigger` from Orchestrator
- Gets `DurableActivityContext` passed

Orchestrator



Starter



Action



Orchestrator Rules – Important to know !



... code must be deterministic

- every time same result
- e.g. No calls to get Date/Time

... code should be non-blocking

- No I/O, Thread.Sleep,...

... code must never initiate any async operation

- No Task.Run, HttpClient.SendAsync,....

... Infinite loops should be avoided

- Can cause OutOfMemory

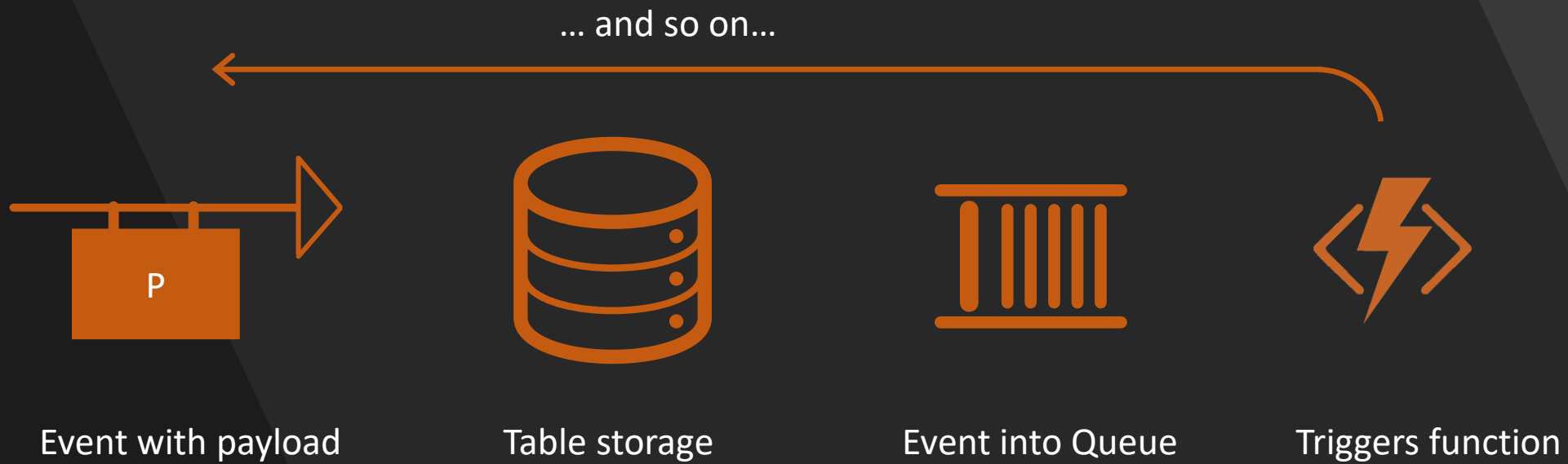
Keeping states



- Using Azure Storage for keeping states
- For communication between functions
 - Queues, by sending messages
- For keeping track of state
 - Tables, by appending
- Called: **Event Sourcing**

Event Sourcing

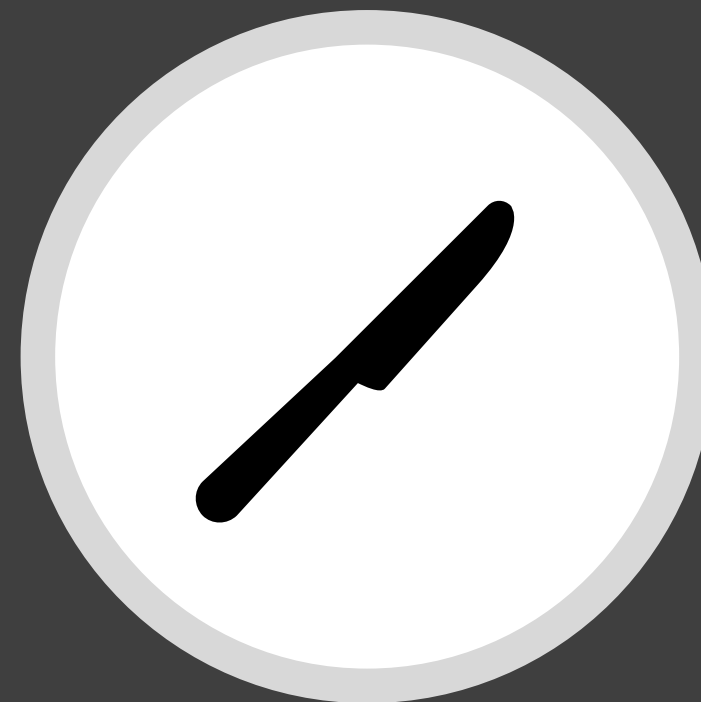
Sidenote:
Highlevel for
Durable Function



Storage



- Provide connection string
- Dfx associate assigned storage with naming „Task Hub“
 - Contains states and history
 - Multiples Task Hubs
 - Can share a single storage account



First steps

Demo

Maintaining orchestration



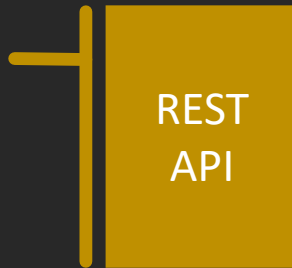
- How to do logging
- How to monitor
- How to get details from orchestration instances
- How to handle errors

Logging orchestration



- Logging: ILogger
 - `Log.LogInformation(„whatever...!“)`
- But..., Orchestrator does replay
 - Use
`if(!ctx.IsReplaying)`
`{ log.LogInformation(„whatever....!“); }`
- Also use CustomStatus
 - `Ctx.SetCustomStatus(„{`
`percentage: „90%“,`
`message: „Updating DB“`
`}“);`

REST API



HTTP/1.1 202 Accepted

Content-Length: 923

Content-Type: application/json; charset=utf-8

Location:

<https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2?taskHub=DurableFunctionsHub&connection=Storage&code=XXX>

{

"id": "34ce9a28a6834d8492ce6a295f1a80e2",

"statusQueryGetUri": "https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2?taskHub=DurableFunctionsHub&connection=Storage&code=XXX",

"sendEventPostUri": "https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2/raiseEvent/{eventName}?taskHub=DurableFunctionsHub&connection=Storage&code=XXX",

"terminatePostUri": "https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2/terminate?reason={text}&taskHub=DurableFunctionsHub&connection=Storage&code=XXX",

"rewindPostUri": "https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2/rewind?reason={text}&taskHub=DurableFunctionsHub&connection=Storage&code=XXX"

}

REST API



REST
API

HTTP/1.1 202 Accepted

Content-Length: 923

Content-Type: application/json; charset=utf-8

Location:

<https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2?taskHub=DurableFunctionsHub&connection=Storage&code=XXX>

{

"id": "34ce9a28a6834d8492ce6a295f1a80e2",

"statusQueryGetUri": "<https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2?taskHub=DurableFunctionsHub&connection=Storage&code=XXX>",

"sendEventPostUri": "<https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2/raiseEvent/{eventName}?taskHub=DurableFunctionsHub&connection=Storage&code=XXX>",

"terminatePostUri": "<https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2/terminate?reason={text}&taskHub=DurableFunctionsHub&connection=Storage&code=XXX>",

"rewindPostUri": "<https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2/rewind?reason={text}&taskHub=DurableFunctionsHub&connection=Storage&code=XXX>"

}

REST API

REST
API

HTTP/1.1 202 Accepted

Content-Length: 923

Content-Type: application/json; charset=utf-8

Location:

<https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2?taskHub=DurableFunctionsHub&connection=Storage&code=XXX>

{

"id": "34ce9a28a6834d8492ce6a295f1a80e2",

"statusQueryGetUri": "https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2?taskHub=DurableFunctionsHub&connection=Storage&code=XXX",

"sendEventPostUri": "https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2/raiseEvent/{eventName}?taskHub=DurableFunctionsHub&connection=Storage&code=XXX",

"terminatePostUri": "https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2/terminate?reason={text}&taskHub=DurableFunctionsHub&connection=Storage&code=XXX",

"rewindPostUri": "https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2/rewind?reason={text}&taskHub=DurableFunctionsHub&connection=Storage&code=XXX"

}

REST API

REST
API

HTTP/1.1 202 Accepted

Content-Length: 923

Content-Type: application/json; charset=utf-8

Location:

<https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2?taskHub=DurableFunctionsHub&connection=Storage&code=XXX>

{

"id": "34ce9a28a6834d8492ce6a295f1a80e2",

"statusQueryGetUri": "https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2?taskHub=DurableFunctionsHub&connection=Storage&code=XXX",

"sendEventPostUri": "https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2/raiseEvent/{eventName}?taskHub=DurableFunctionsHub&connection=Storage&code=XXX",

"terminatePostUri": "https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2/terminate?reason={text}&taskHub=DurableFunctionsHub&connection=Storage&code=XXX",

"rewindPostUri": "https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2/rewind?reason={text}&taskHub=DurableFunctionsHub&connection=Storage&code=XXX"

}

Get Status

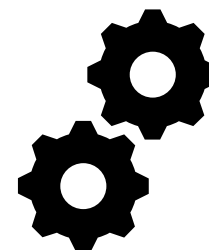
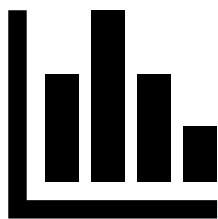
REST
API

Parameter & filter

- showHistory, showHistoryOutput
- createTimeFrom, createTimeTo
- runtimeStatus

<https://{host}/runtime/webhooks/durabletask/instances/34ce9a28a6834d8492ce6a295f1a80e2?taskHub=DurableFunctionsHub&connection=Storage&code=XXX&showHistory=true&showHistoryOutput=true>

```
{ "createdTime": "2018-02-28T05:18:49Z",  
  "historyEvents":  
    [  
      {  
        "EventType": "ExecutionStarted",  
        "FunctionName": "E1_HelloSequence",  
        "Timestamp": "2018-02-28T05:18:49.3452372Z",  
        "Result": "Hello Tokyo!",  
        "ScheduledTime": "2018-02-28T05:18:51.3939873Z",  
      }, ...  
    ]  
}
```



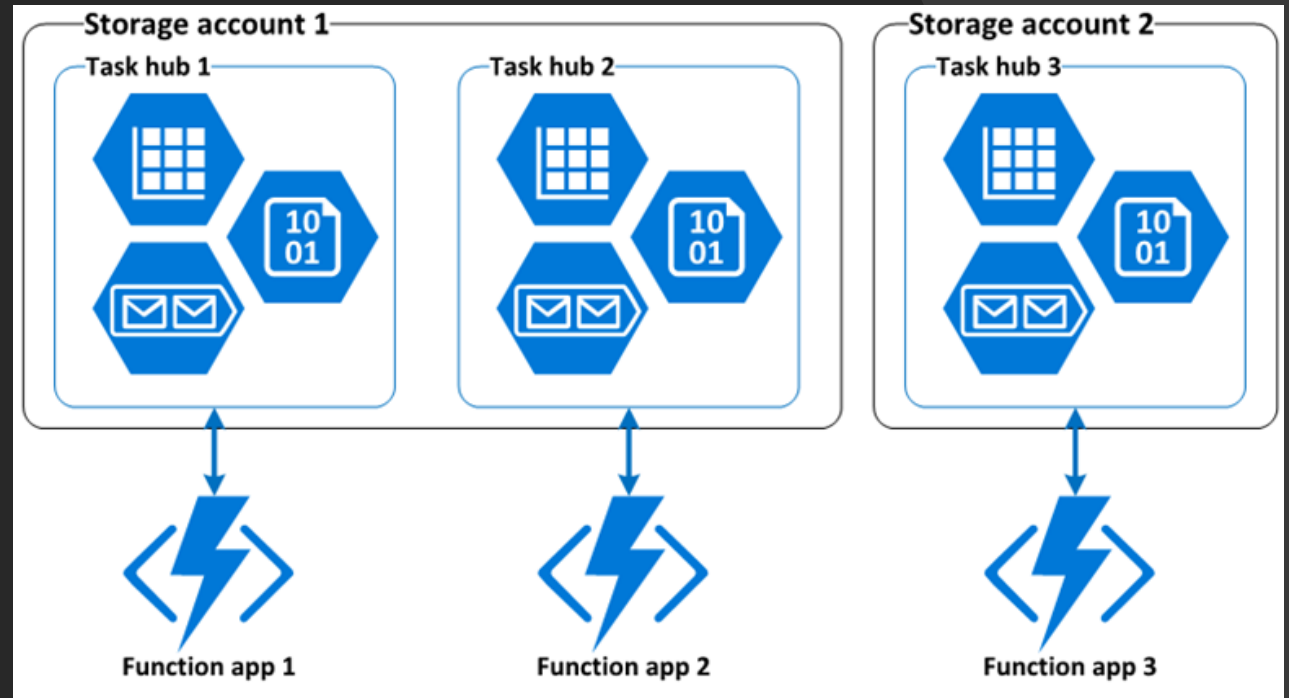
Going into code

Continuing Demo

Insides of Task Hub



- Using Azure Storage Explorer
 - <https://azure.microsoft.com/en-US/features/storage-explorer/>
 - Explore Azure Storage
 - Explore local from Azure Storage Emulator

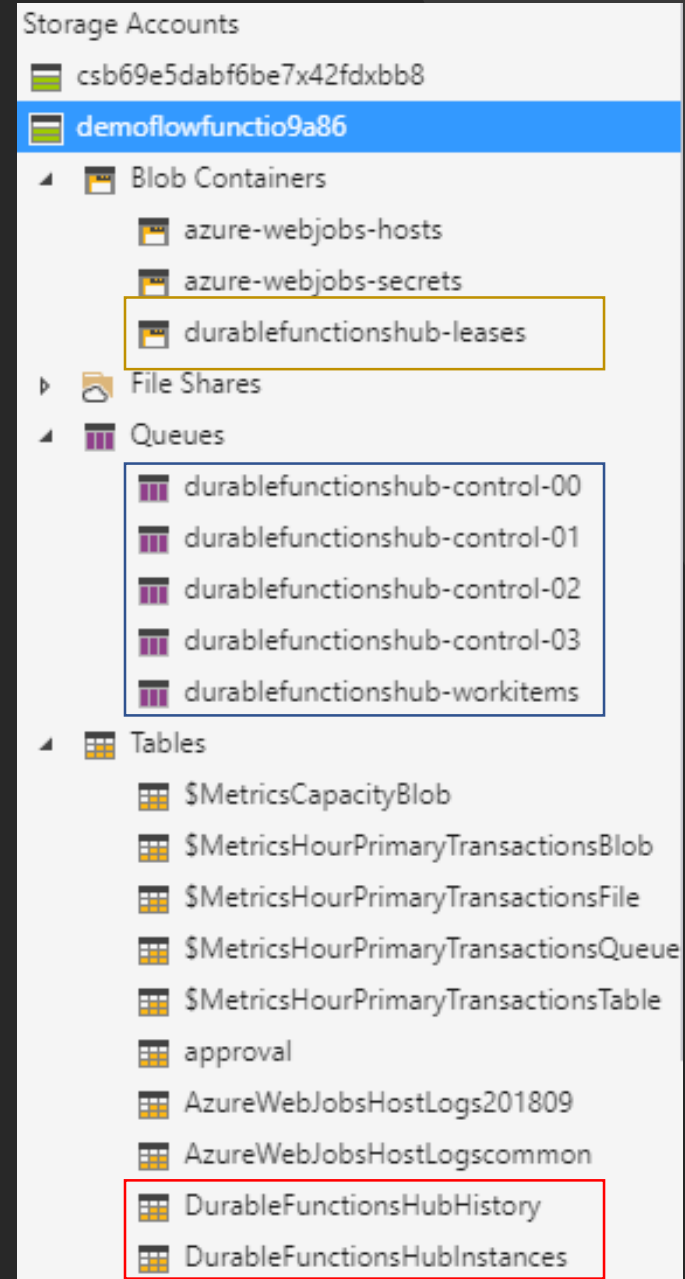


From <https://docs.microsoft.com/en-us/azure/azure-functions/durable-functions-task-hubs>

Insides of Task Hub



- Contains some maintenance queues and
- Message Queue
- History Table
- Instance Table
- Lease blobs



Insides of Task Hub



History Table

- Contains all information to keep track
 - Runtime state, Inputvalues, Activity names,....

EventType	ExecutionId	IsPlayed	_Timestamp	Input	Name	OrchestrationInstance
OrchestratorStarted	1d311ce4ffb944a8be757a7e065b3a81	false	2018-09-09T07:09:13.686Z	null	null	null
ExecutionStarted	1d311ce4ffb944a8be757a7e065b3a81	true	2018-09-09T07:09:08.158Z	" +4915153811045,toto_san@live.com"	E4_SmsPhoneVerification	{"InstanceId":"0c5809f2f620410ea5111724a67e9934"
OrchestratorCompleted	1d311ce4ffb944a8be757a7e065b3a81	false	2018-09-09T07:09:13.688Z	null	null	null
OrchestratorStarted	1d311ce4ffb944a8be757a7e065b3a81	false	2018-09-09T07:09:14.436Z	null	null	null
TaskCompleted	1d311ce4ffb944a8be757a7e065b3a81	true	2018-09-09T07:09:14.361Z	null	null	null
TimerCreated	1d311ce4ffb944a8be757a7e065b3a81	false	2018-09-09T07:09:14.438Z	null	null	null
OrchestratorCompleted	1d311ce4ffb944a8be757a7e065b3a81	false	2018-09-09T07:09:14.438Z	null	null	null
OrchestratorStarted	1d311ce4ffb944a8be757a7e065b3a81	false	2018-09-09T07:10:44.289Z	null	null	null
TimerFired	1d311ce4ffb944a8be757a7e065b3a81	true	2018-09-09T07:09:14.438Z	null	null	null
ExecutionCompleted	1d311ce4ffb944a8be757a7e065b3a81	false	2018-09-09T07:10:44.291Z	null	null	null
OrchestratorCompleted	1d311ce4ffb944a8be757a7e065b3a81	false	2018-09-09T07:10:44.291Z	null	null	null
null	1d311ce4ffb944a8be757a7e065b3a81	null	null	null	null	null
TaskScheduled	1d311ce4ffb944a8be757a7e065b3a81	false	2018-09-09T07:09:13.688Z	null	E4_SendEmailChallenge	null
ExecutionStarted	d1b411d889a14f04aaf95f45d37761ad	true	2018-09-04T14:27:27.408Z	" +4915153811045,toto_san@live.com"	E4_SmsPhoneVerification	{"InstanceId":"15436018ed5544bfa7390fb0d2f41a93"
TaskScheduled	d1b411d889a14f04aaf95f45d37761ad	false	2018-09-04T14:27:28.609Z	null	E4_SendEmailChallenge	null
OrchestratorStarted	d1b411d889a14f04aaf95f45d37761ad	false	2018-09-04T14:27:30.780Z	null	null	null
TaskFailed	d1b411d889a14f04aaf95f45d37761ad	true	2018-09-04T14:27:30.529Z	null	null	null

Insides of Task Hub



Instances Table

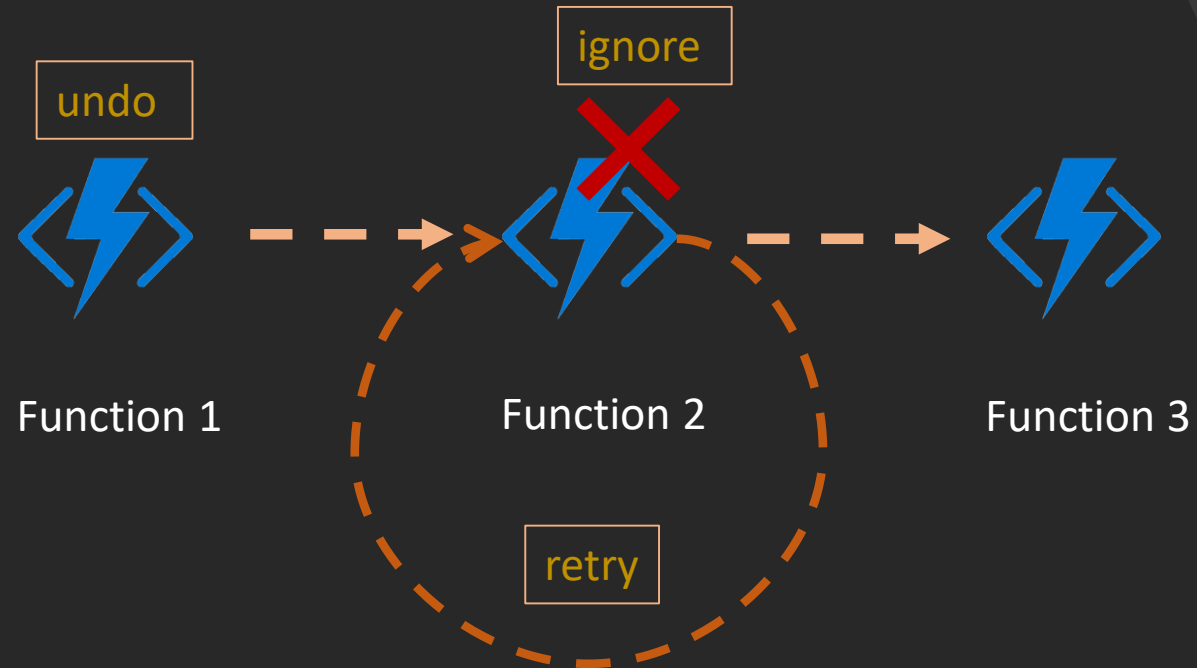
- Contains all information to all running orchestration instances

	ExecutionId	Input	LastUpdatedTime	Name	Output
08.158Z	1d311ce4ffb944a8be757a7e065b3a81	" +4915153811045,toto_san@live.com"	2018-09-09T07:10:44.291Z	E4_SmsPhoneVerification	false
27.408Z	d1b411d889a14f04aaf95f45d37761ad	" +4915153811045,toto_san@live.com"	2018-09-04T14:27:30.866Z	E4_SmsPhoneVerification	Orchestrator function 'E4_SmsPhoneVerification' failed: The activity function 'E4_Se
59.393Z	5d8ca6f3f4dc45cd9230212eab6dcd14	" +4915153811045,toto_san@live.com"	2018-09-09T08:30:04.664Z	E4_SmsPhoneVerification	Orchestrator function 'E4_SmsPhoneVerification' failed: The activity function 'E4_Se
43.172Z	15cae781830542ea9fbdcc9c1707b58b	" +4915153811045,toto_san@live.com"	2018-09-12T21:59:16.983Z	E4_SmsPhoneVerification	false
24.640Z	9477f351fe5c426abd8d57891141fd90	" +4915153811045,toto_san@live.com"	2018-09-09T07:10:14.940Z	E4_SmsPhoneVerification	false
08.642Z	2aebd0dc952048d08daea3572705e4ec	" +4915153811045,toto_san@live.com"	2018-09-12T21:27:10.821Z	E4_SmsPhoneVerification	Orchestrator function 'E4_SmsPhoneVerification' failed: The activity function 'E4_Se
20.754Z	1e9a69147bd548b193fcc07c9babfdac	" +4915153811045,toto_san@live.com"	2018-09-04T06:27:37.035Z	E4_SmsPhoneVerification	Orchestrator function 'E4_SmsPhoneVerification' failed: The activity function 'E4_Se
58.950Z	2d8a1d25345246ab9ef6f9bfccfc9f2	" +4915153811045,toto_san@live.com"	2018-09-12T21:25:03.385Z	E4_SmsPhoneVerification	Orchestrator function 'E4_SmsPhoneVerification' failed: The activity function 'E4_Se
04.415Z	431efc8fb6f74b8b900e7fa2ea1f6045	" +4915153811054,toto_san@live.com"	2018-09-02T14:06:32.845Z	E4_SmsPhoneVerification	Orchestrator function 'E4_SmsPhoneVerification' failed: The activity function 'E4_Se
37.059Z	ff20c7c5c8da4dd6b5f0f7c878b36d70	" +4915153811045,toto_san@live.com"	2018-09-09T07:03:38.441Z	E4_SmsPhoneVerification	false
47.811Z	6b99b9e4281743eab7aff696d6985232	" +4915153811054,toto_san@live.com"	2018-09-02T13:42:08.044Z	E4_SmsPhoneVerification	{text}
06.837Z	c79cd0e8f4d741879d372bb1c9d727d2	" +4915153811045,toto_san@live.com"	2018-09-13T07:04:42.914Z	E4_SmsPhoneVerification	false
01.664Z	9981135771e04092968922e9cefd2cb1	" +4915153811045,toto_san@live.com"	2018-09-04T06:21:07.746Z	E4_SmsPhoneVerification	Orchestrator function 'E4_SmsPhoneVerification' failed: The activity function 'E4_Se
06.129Z	3d1ba9ad4f33481289962d46925e07a7	" +4915153811045,toto_san@live.com"	2018-09-13T06:31:05.447Z	E4_SmsPhoneVerification	false
01.574Z	10a25f268d0a40debae696dee56687b3	" +4915153811054,toto_san@live.com"	2018-09-02T13:40:34.932Z	E4_SmsPhoneVerification	{text}
36.869Z	365fc98f964d4cafb64e8cb9fff74858	" +4915153811054,toto_san@live.com"	2018-09-02T18:00:40.945Z	E4_SmsPhoneVerification	Orchestrator function 'E4_SmsPhoneVerification' failed: The activity function 'E4_Se
30.422Z	af14b7d13cb24f0c8e66aca2998cc303	" +4915153811045,toto_san@live.com"	2018-09-06T18:16:41.934Z	E4_SmsPhoneVerification	Orchestrator function 'E4_SmsPhoneVerification' failed: The activity function 'E4_Se

Handling errors



- Undo,
- Retry,
- Ignore,
- What to do?



Handling errors



- Use Try/Catches in activities
 - Return an object to communicate error back to orchestrator

```
public static object Run([ActivityTrigger] string input)
{
    try
    {
        var output = DoPhantasticCalculationHere(input);
        return new {Success = true, Value = output};
    }
    catch (System.Exception ex)
    {
        return new {Success = false, Value = ex.Message};
    }
}
```

Handling errors



- Use retries in orchestrator
 - Instable connections, transient issues, ...
 - Dfx contains method `CallActivityWithRetryAsync` for use with Retry-policy

```
public static async Task<object> Run([OrchestrationTrigger] DurableOrchestrationContext ctx)
{
    var retryOptions = new RetryOptions(
        firstRetryInterval: TimeSpan.FromSeconds(5),
        maxNumberOfAttempts: 3);

    await ctx.CallActivityWithRetryAsync("MyGeniousFunction", retryOptions, null);
    //...
}
```

Handling errors



- Handling with TimeOuts

- Using durable time
- Setting Context-Timer by CreateTimer
- Handling timeout with Task.WhenAny

```
public static async Task<bool> Run(DurableOrchestrationContext context)
{
    TimeSpan timeout = TimeSpan.FromSeconds(30);
    DateTime deadline = context.CurrentUtcDateTime.Add(timeout);
    using (var cts = new CancellationTokenSource())
    {
        Task activityTask = context.CallActivityAsync(„DoGeniousFunction");
        Task timeoutTask = context.CreateTimer(deadline, cts.Token);
        Task winner = await Task.WhenAny(activityTask, timeoutTask);
        if (winner == activityTask)
        {
            // success case
            cts.Cancel();
            return true;
        } else {
            // timeout case
            return false;
        }
    }
}
```

Handling errors



- Cancellation by REST API, if

- Not needed anymore
- Running too long
- Other maintenance reason

```
"terminatePostUri": "https://{host}/runtime/webhooks/durabletask/instances/  
34ce9a28a6834d8492ce6a295f1a80e2/terminate?reason={text}&taskHub=DurableFu  
nctionsHub&connection=Storage&code=XXX"
```


It was a plessure

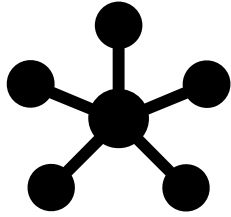
Thomas Tomow | Azure MVP

mail: toto_san@live.com

twitter: [toto_san1](https://twitter.com/toto_san1)

www: tomow.de





Resources

- Microsoft Documentation is a really good start
<https://docs.microsoft.com/en-gb/azure/azure-functions/durable-functions-install>
- See the code in GitHub
<https://github.com/Azure/azure-functions-durable-extension>
- Also notice the Durable Task Framework for development on every other thing then DurableFunctions 😊
<https://github.com/Azure/durabletask>

