Monitoring Workflows and Handling Errors

Mark Heath
MICROSOFT MVP

@mark_heath   www.markheath.net

Overview

Durable Functions REST API

- Check on workflow progress

Storage Account contents

- Understand “Task Hubs”

Handling Errors

- Exceptions in activity functions
- Retrying activities
- Cancelling workflows

Durable Functions REST API

```
{
   id: "4d2b7bb4d7724733982eb0070ff9f54c",
   statusQueryGetUri: "http://localhost:7071/admin/extensions/DurableT
askExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c?taskHub=Durabl
eFunctionsHub&connection=Storage&code=anZaCbawHp/ufQ2EBukjiml1P1s480nP
ImkQ/r5rN2V5XDJ/l/F7lA==",
   sendEventPostUri: "http://localhost:7071/admin/extensions/DurableTa
skExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c/raiseEvent/{eve
ntName}?taskHub=DurableFunctionsHub&connection=Storage&code=anZaCbawHp
/ufQ2EBukjiml1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA==",
   terminatePostUri: "http://localhost:7071/admin/extensions/DurableTa
skExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c/terminate?reaso
n={text}&taskHub=DurableFunctionsHub&connection=Storage&code=anZaCbawH
p/ufQ2EBukjiml1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA=="
}
```

Durable Functions REST API

```
{
   id: "4d2b7bb4d7724733982eb0070ff9f54c",
   statusQueryGetUri: "http://localhost:7071/admin/extensions/DurableT
askExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c?taskHub=Durabl
eFunctionsHub&connection=Storage&code=anZaCbawHp/ufQ2EBukjiml1P1s480nP
ImkQ/r5rN2V5XDJ/l/F7lA==",
   sendEventPostUri: "http://localhost:7071/admin/extensions/DurableTa
skExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c/raiseEvent/{eve
ntName}?taskHub=DurableFunctionsHub&connection=Storage&code=anZaCbawHp
/ufQ2EBukjiml1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA==",
   terminatePostUri: "http://localhost:7071/admin/extensions/DurableTa
skExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c/terminate?reaso
n={text}&taskHub=DurableFunctionsHub&connection=Storage&code=anZaCbawH
p/ufQ2EBukjiml1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA=="
}
```

Durable Functions REST API

```
{
   id: "4d2b7bb4d7724733982eb0070ff9f54c",
   statusQueryGetUri: "http://localhost:7071/admin/extensions/DurableT
askExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c?taskHub=Durabl
eFunctionsHub&connection=Storage&code=anZaCbawHp/ufQ2EBukjiml1P1s480nP
ImkQ/r5rN2V5XDJ/l/F7lA==",
   sendEventPostUri: "http://localhost:7071/admin/extensions/DurableTa
skExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c/raiseEvent/{eve
ntName}?taskHub=DurableFunctionsHub&connection=Storage&code=anZaCbawHp
/ufQ2EBukjiml1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA==",
   terminatePostUri: "http://localhost:7071/admin/extensions/DurableTa
skExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c/terminate?reaso
n={text}&taskHub=DurableFunctionsHub&connection=Storage&code=anZaCbawH
p/ufQ2EBukjiml1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA=="
}
```

Durable Functions REST API

```
{
   id: "4d2b7bb4d7724733982eb0070ff9f54c",
   statusQueryGetUri: "http://localhost:7071/admin/extensions/DurableT
askExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c?taskHub=Durabl
eFunctionsHub&connection=Storage&code=anZaCbawHp/ufQ2EBukjiml1P1s480nP
ImkQ/r5rN2V5XDJ/l/F7lA==",
   sendEventPostUri: "http://localhost:7071/admin/extensions/DurableTa
skExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c/raiseEvent/{eve
ntName}?taskHub=DurableFunctionsHub&connection=Storage&code=anZaCbawHp
/ufQ2EBukjiml1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA==",
   terminatePostUri: "http://localhost:7071/admin/extensions/DurableTa
skExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c/terminate?reaso
n={text}&taskHub=DurableFunctionsHub&connection=Storage&code=anZaCbawH
p/ufQ2EBukjiml1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA=="
}
```

Durable Functions REST API

```
{
   id: "4d2b7bb4d7724733982eb0070ff9f54c",
   statusQueryGetUri: "http://localhost:7071/admin/extensions/DurableT
askExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c?taskHub=Durabl
eFunctionsHub&connection=Storage&code=anZaCbawHp/ufQ2EBukjiml1P1s480nP
ImkQ/r5rN2V5XDJ/l/F7lA==",
   sendEventPostUri: "http://localhost:7071/admin/extensions/DurableTa
skExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c/raiseEvent/{eve
ntName}?taskHub=DurableFunctionsHub&connection=Storage&code=anZaCbawHp
/ufQ2EBukjiml1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA==",
   terminatePostUri: "http://localhost:7071/admin/extensions/DurableTa
skExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c/terminate?reaso
n={text}&taskHub=DurableFunctionsHub&connection=Storage&code=anZaCbawH
p/ufQ2EBukjiml1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA=="
}
```

Durable Functions REST API

```
{
   id: "4d2b7bb4d7724733982eb0070ff9f54c",
   statusQueryGetUri: "http://localhost:7071/admin/extensions/DurableT
askExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c?taskHub=Durabl
eFunctionsHub&connection=Storage&code=anZaCbawHp/ufQ2EBukjiml1P1s480nP
ImkQ/r5rN2V5XDJ/l/F7lA==",
   sendEventPostUri: "http://localhost:7071/admin/extensions/DurableTa
skExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c/raiseEvent/{eve
ntName}?taskHub=DurableFunctionsHub&connection=Storage&code=anZaCbawHp
/ufQ2EBukjiml1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA==",
   terminatePostUri: "http://localhost:7071/admin/extensions/DurableTa
skExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c/terminate?reaso
n={text}&taskHub=DurableFunctionsHub&connection=Storage&code=anZaCbawH
p/ufQ2EBukjiml1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA=="
}
```

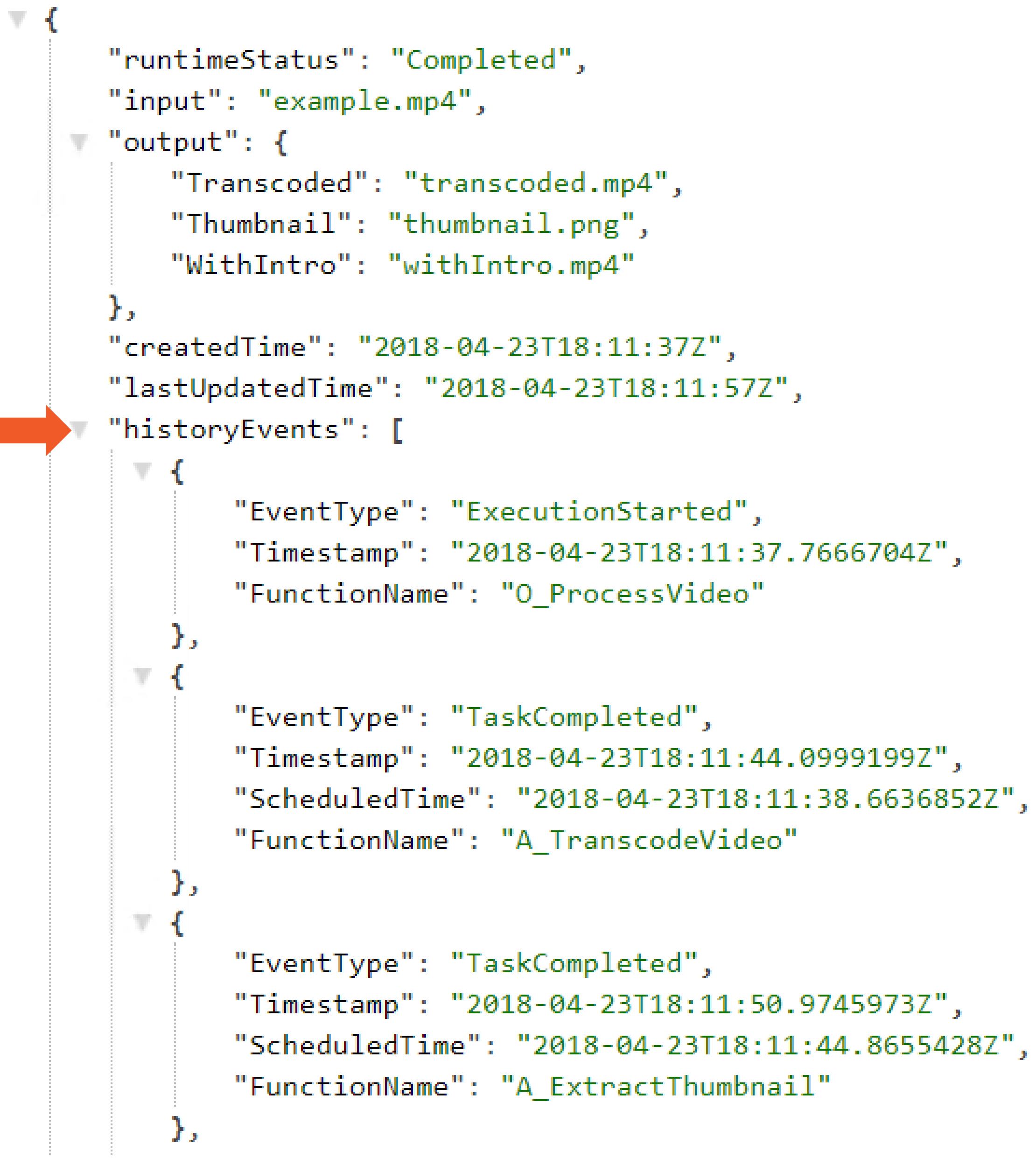
Durable Functions REST API

```
{
   id: "4d2b7bb4d7724733982eb0070ff9f54c",
   statusQueryGetUri: "http://localhost:7071/admin/extensions/DurableT
askExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c?taskHub=Durabl
eFunctionsHub&connection=Storage&code=anZaCbawHp/ufQ2EBukjiml1P1s480nP
ImkQ/r5rN2V5XDJ/l/F7lA==",
   sendEventPostUri: "http://localhost:7071/admin/extensions/DurableTa
skExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c/raiseEvent/{eve
ntName}?taskHub=DurableFunctionsHub&connection=Storage&code=anZaCbawHp
/ufQ2EBukjiml1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA==",
   terminatePostUri: "http://localhost:7071/admin/extensions/DurableTa
skExtension/instances/4d2b7bb4d7724733982eb0070ff9f54c/terminate?reaso
n={text}&taskHub=DurableFunctionsHub&connection=Storage&code=anZaCbawH
p/ufQ2EBukjiml1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA=="
}
```

Get Orchestration Status API

```
http://localhost:7071/admin/extensions/DurableTaskExtension
/instances/4d2b7bb4d7724733982eb0070ff9f54c?taskHub=Durable
FunctionsHub&connection=Storage&code=anZaCbawHp/ufQ2EBukjim
l1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA==&showHistory=true
```

showHistory=true



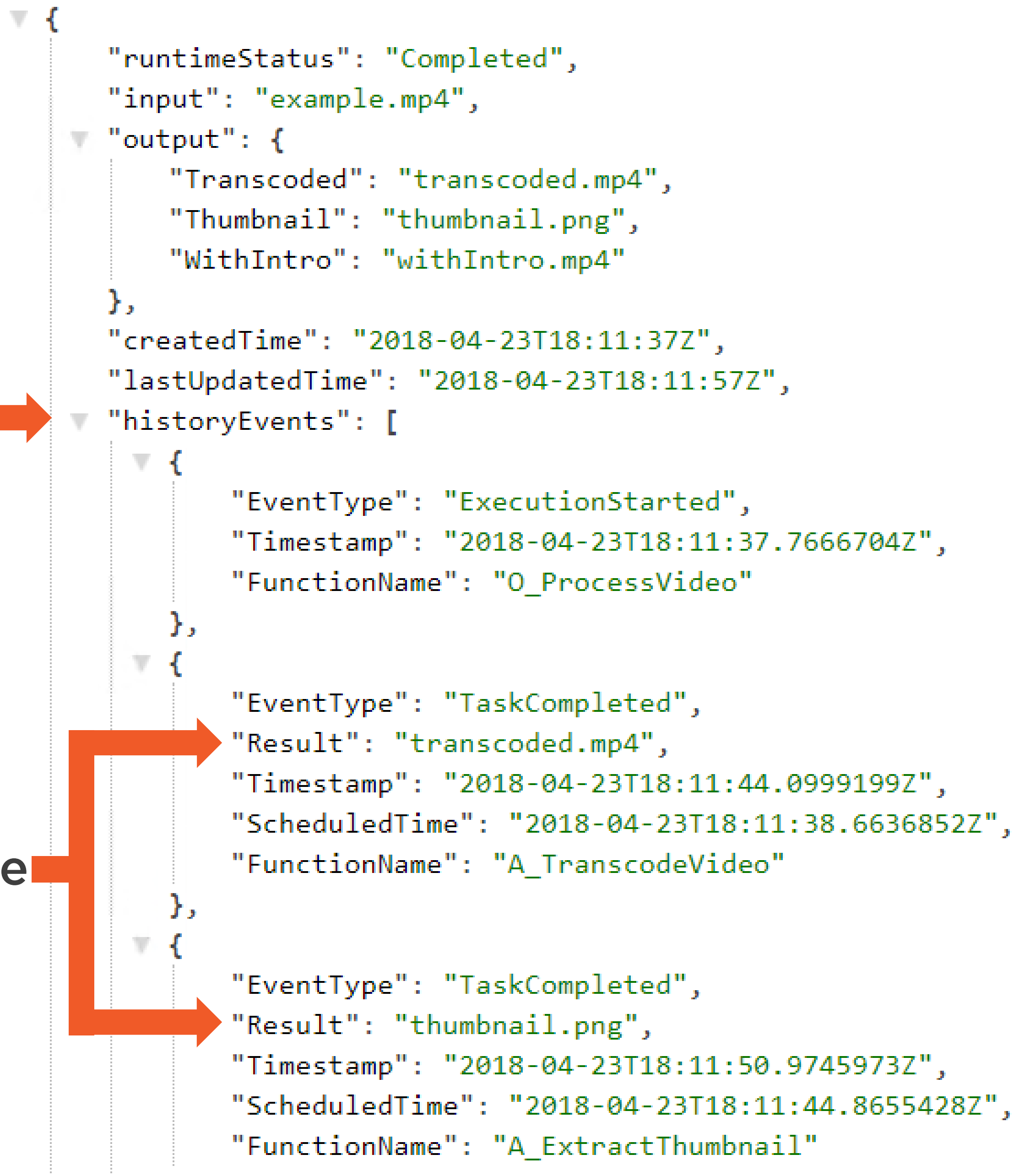
```
{
    "runtimeStatus": "Completed",
    "input": "example.mp4",
    "output": {
        "Transcoded": "transcoded.mp4",
        "Thumbnail": "thumbnail.png",
        "WithIntro": "withIntro.mp4"
    },
    "createdTime": "2018-04-23T18:11:37Z",
    "lastUpdatedTime": "2018-04-23T18:11:57Z",
    "historyEvents": [
        {
            "EventType": "ExecutionStarted",
            "Timestamp": "2018-04-23T18:11:37.7666704Z",
            "FunctionName": "O_ProcessVideo"
        },
        {
            "EventType": "TaskCompleted",
            "Timestamp": "2018-04-23T18:11:44.0999199Z",
            "ScheduledTime": "2018-04-23T18:11:38.6636852Z",
            "FunctionName": "A_TranscodeVideo"
        },
        {
            "EventType": "TaskCompleted",
            "Timestamp": "2018-04-23T18:11:50.9745973Z",
            "ScheduledTime": "2018-04-23T18:11:44.8655428Z",
            "FunctionName": "A_ExtractThumbnail"
        },
```

Get Orchestration Status API

```
http://localhost:7071/admin/extensions/DurableTaskExtension
/instances/4d2b7bb4d7724733982eb0070ff9f54c?taskHub=Durable
FunctionsHub&connection=Storage&code=anZaCbawHp/ufQ2EBukjim
l1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA==&showHistory=true&showHis
toryOutput=true
```

showHistory=true

showHistoryOutput=true



```
{
    "runtimeStatus": "Completed",
    "input": "example.mp4",
    "output": {
        "Transcoded": "transcoded.mp4",
        "Thumbnail": "thumbnail.png",
        "WithIntro": "withIntro.mp4"
    },
    "createdTime": "2018-04-23T18:11:37Z",
    "lastUpdatedTime": "2018-04-23T18:11:57Z",
    "historyEvents": [
        {
            "EventType": "ExecutionStarted",
            "Timestamp": "2018-04-23T18:11:37.7666704Z",
            "FunctionName": "O_ProcessVideo"
        },
        {
            "EventType": "TaskCompleted",
            "Result": "transcoded.mp4",
            "Timestamp": "2018-04-23T18:11:44.0999199Z",
            "ScheduledTime": "2018-04-23T18:11:38.6636852Z",
            "FunctionName": "A_TranscodeVideo"
        },
        {
            "EventType": "TaskCompleted",
            "Result": "thumbnail.png",
            "Timestamp": "2018-04-23T18:11:50.9745973Z",
            "ScheduledTime": "2018-04-23T18:11:44.8655428Z",
            "FunctionName": "A_ExtractThumbnail"
```

Looking Inside Task Hubs

Durable Functions stores state in Azure Storage

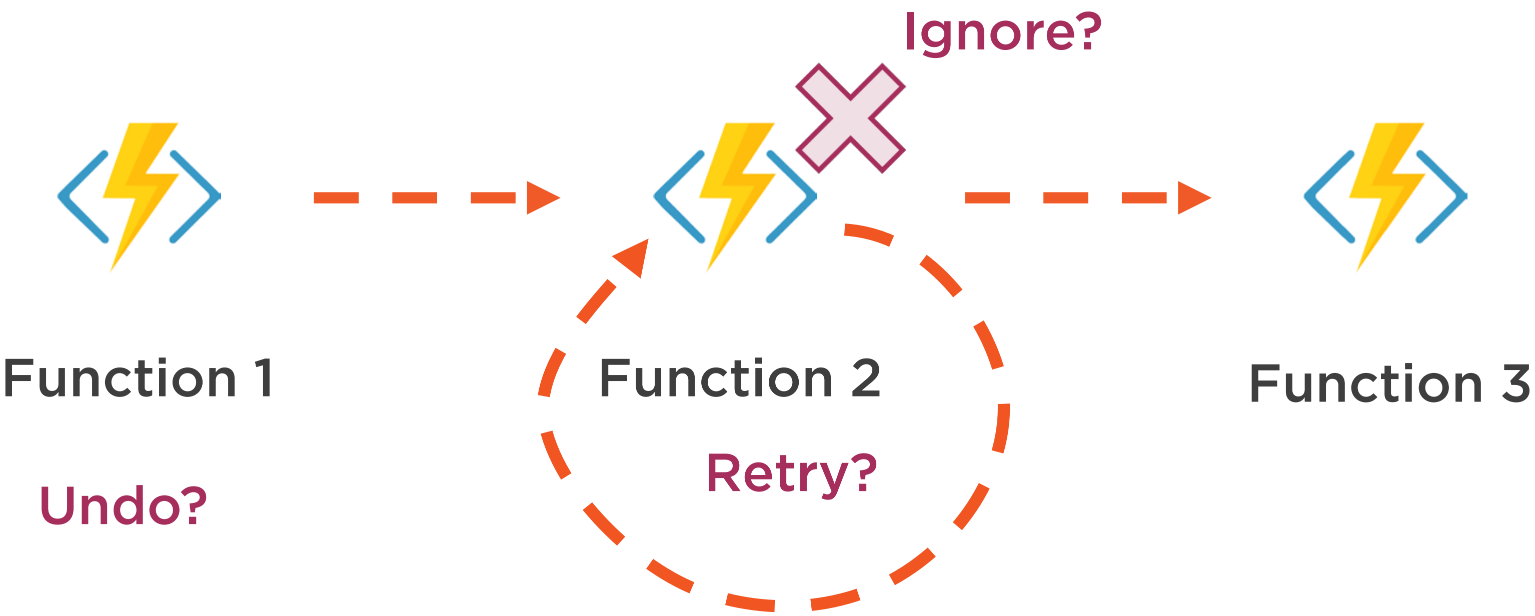
Azure Storage Explorer

- https://azure.microsoft.com/en-us/features/storage-explorer/
- Explore contents of online storage accounts
- Explore local storage emulator contents

Demo

Exploring Task Hub contents with Azure Storage Explorer

Handling Errors



Exceptions in Activity Functions

```
public static object TranscodeVideo([ActivityTrigger] string input)
{
   try
   {
      var output = PerformTranscode(input);
      return new { Success = true,
                   Output = output };
   }
   catch (Exception e)
   {
      return new { Success = false,
                   Error = e.Message };
   }
}
```

Exceptions in Activity Functions

```
public static object TranscodeVideo([ActivityTrigger] string input)
{
    // don’t catch exceptions, allow them to propagate up to
    // the orchestrator function
    return PerformTranscode(input);
}
```

Handling Exceptions in Orchestrator Functions

```
public static async Task<object> MyOrchestrator(
      [OrchestrationTrigger] DurableOrchestrationContext ctx)
{
   try
   {
      var a = await ctx.CallSubOrchestratorAsync<string>("A_1");
      var b = await ctx.CallSubOrchestratorAsync<string>("A_2", a);
      var c = await ctx.CallSubOrchestratorAsync<string>("A_3", b);
      return new { Success = true, Output = c };
   }
   catch (Exception e)
   {
      // log exception, perform cleanup
      return new { Success = false, Error = e.Message };
   }
}
```

Handling Exceptions in Orchestrator Functions

```
public static async Task<object> MyOrchestrator(
      [OrchestrationTrigger] DurableOrchestrationContext ctx)
{
   try
   {
      var a = await ctx.CallSubOrchestratorAsync<string>("A_1");
      var b = await ctx.CallSubOrchestratorAsync<string>("A_2", a);
      var c = await ctx.CallSubOrchestratorAsync<string>("A_3", b);
      return new { Success = true, Output = c };
   }
   catch (Exception e)
   {
      await ctx.CallSubOrchestratorAsync<string>("A_Cleanup");
      return new { Success = false, Error = e.Message };
   }
}
```

Demo

Handling Errors

- Throw an exception from an activity
- Catch it in our orchestrator

Retrying Activities in Orchestrator Functions

```
public static async Task<object> MyOrchestrator(
      [OrchestrationTrigger] DurableOrchestrationContext ctx)
{
   for (var attempt = 0; attempt < 3; attempt++)
   {
      try
      {
         await ctx.CallSubOrchestratorAsync<string>("A_1");
         break; // success
      }
      catch (Exception e)
      {
         await Task.Delay(5000); // NO – DON’T DO THIS!!!
      }
   }
   // ... continue if successful or cleanup
}
```

Retrying Activities in Orchestrator Functions

```
public static async Task<object> MyOrchestrator(
      [OrchestrationTrigger] DurableOrchestrationContext ctx)
{
    // retry four up to 4 times
    // with 5 second delay between each attempt...
    var a = await ctx.CallActivityWithRetryAsync<string>(
             "A_ExtractThumbnail",
             new RetryOptions(TimeSpan.FromSeconds(5), 4),
             transcodedLocation);
    // if we get here ExtractThumbnail was called successfully
}
```

Retrying Activities in Orchestrator Functions

```
public static async Task<object> MyOrchestrator(
      [OrchestrationTrigger] DurableOrchestrationContext ctx)
{
    // retry four up to 4 times
    // with 5 second delay between each attempt...
    var a = await ctx.CallActivityWithRetryAsync<string>(
             "A_ExtractThumbnail",
             new RetryOptions(TimeSpan.FromSeconds(5), 4)
               {Handle = ex => ex is InvalidOperationException},
             transcodedLocation);
    // if we get here ExtractThumbnail was called successfully
}
```

Cancelling Workflows

Call REST API to terminate orchestration

- No longer needed
- Running too long
- System maintenance

POST to terminate API

```
terminatePostUri: "http://localhost:7071/admin/exte
nsions/DurableTaskExtension/instances/4d2b7bb4d7724
733982eb0070ff9f54c/terminate?reason={text}&taskHub
=DurableFunctionsHub&connection=Storage&code=anZaCb
awHp/ufQ2EBukjiml1P1s480nPImkQ/r5rN2V5XDJ/l/F7lA=="
```

Demo

Cancelling Workflows

Summary

Durable Functions REST API

- Check on the status of orchestrations
- Cancel orchestrations

Task Hub Storage

- Azure Storage Explorer

Handling Errors

- Catch exceptions in orchestrator functions
- Retrying activities with delay
- CallActivityWithRetryAsync
- Performing clean-up

Up next …

Implementing the “fan-out fan-in” pattern