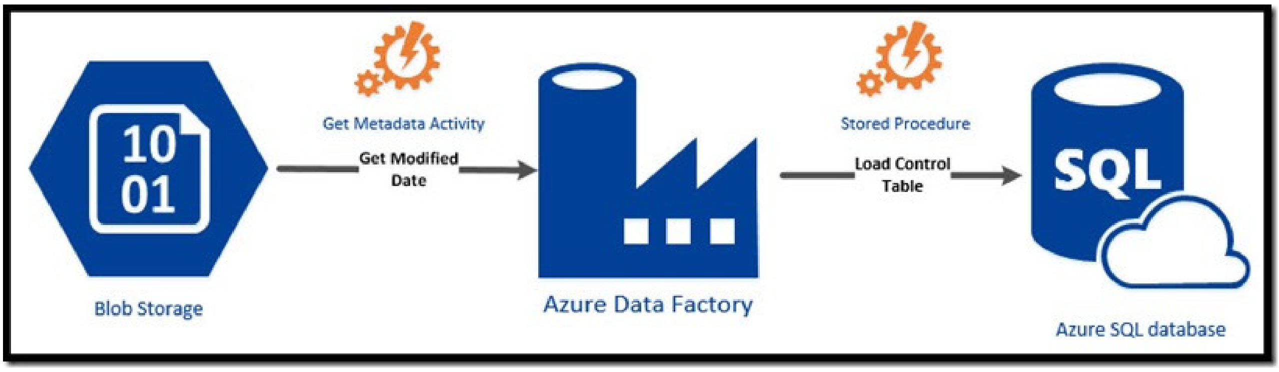
Azure Data Factory - Get Metadata Activity

blog.pragmaticworks.com/azure-data-factory-get-metadata-activity

Welcome to part one of a new blog series I am beginning on ***Azure Data Factory***. In this first post I am going to discuss the Get Metadata activity in Azure Data Factory. In this post you are going to see how to use the get metadata activity to retrieve metadata about a file stored in Azure Blob storage and how to reference the output parameters of that activity. In part two of this blog series I'll show you how to use the stored procedure activity to load the metadata information into a table in Azure SQL Database. Take a look at the below design pattern:

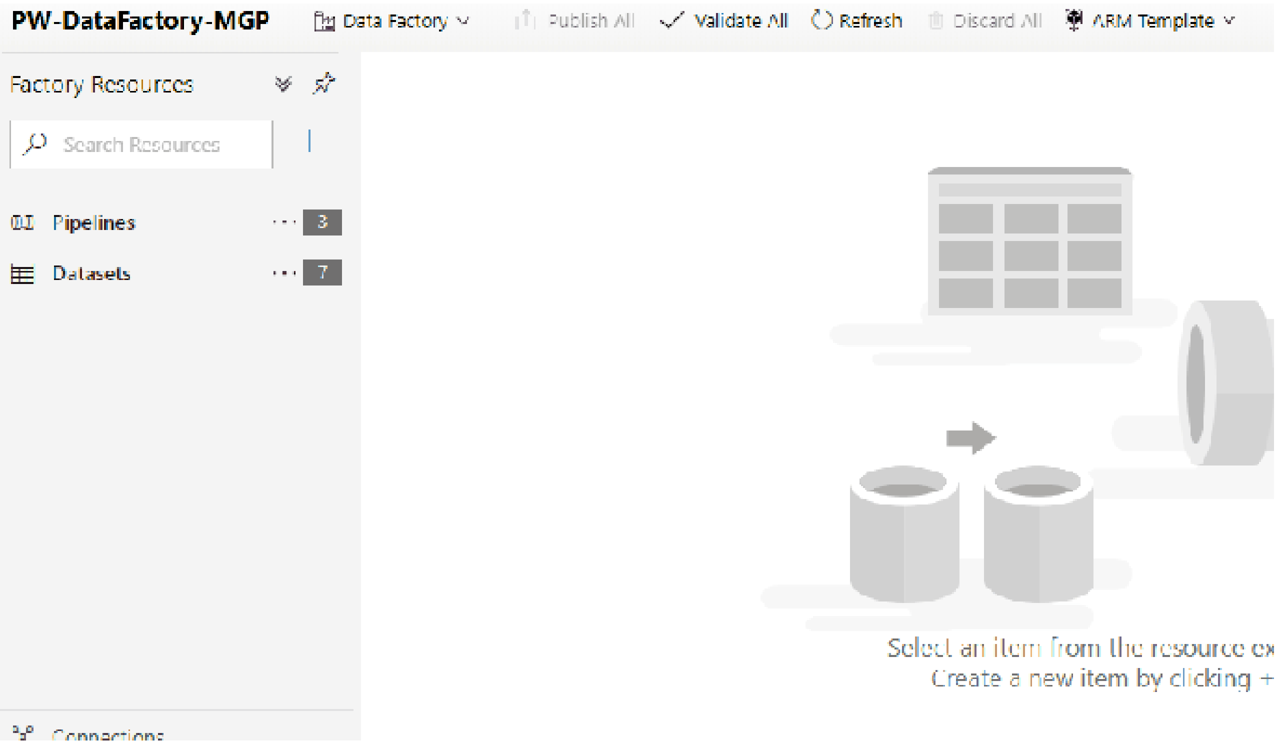


In this post you are specifically going to learn the following three items:

1. Set up and configuration of the activity
2. How to read the outputs
3. How to reference output parameters from the Get Metadata activity

Part 1: Setting up the Get Metadata activity

First, I am going to create a new pipeline and then add the Get Metadata activity to the pipeline.

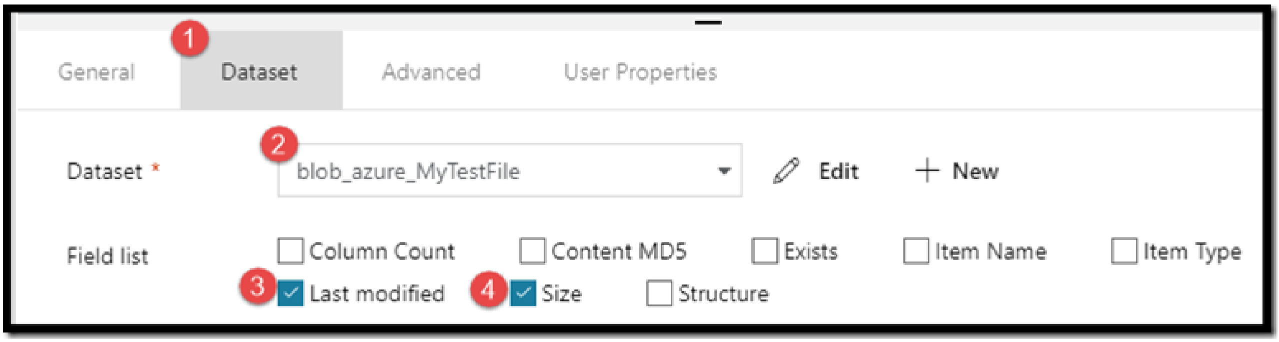


Next, I am going to set up and configure the activity to read from a file I have in Azure Blob storage.

Please note, for this post I assume you know how to create a dataset in Azure Data Factory. If you do not, kindly let me know and I can throw together a quick blog on how that is done!

With the Get Metadata activity selected, complete the following tasks:

1. Click on **Dataset** in the property window.
2. Select your dataset from the dropdown, or create a new one that points to your file.
3. Select the property **Last Modified** from the fields list.
4. Select the property **Size** from the fields list.
5. Select any other properties you would like to get information about.

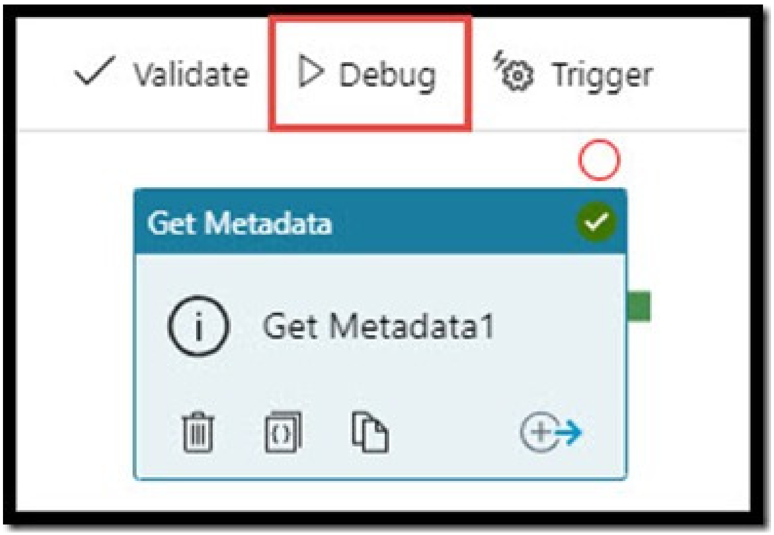


Dataset * blob_azure_MyTestFile

Field list □ Column Count □ Content MD5 □ Exists □ Item Name □ Item Type ☑ Last modified ☑ Size □ Structure

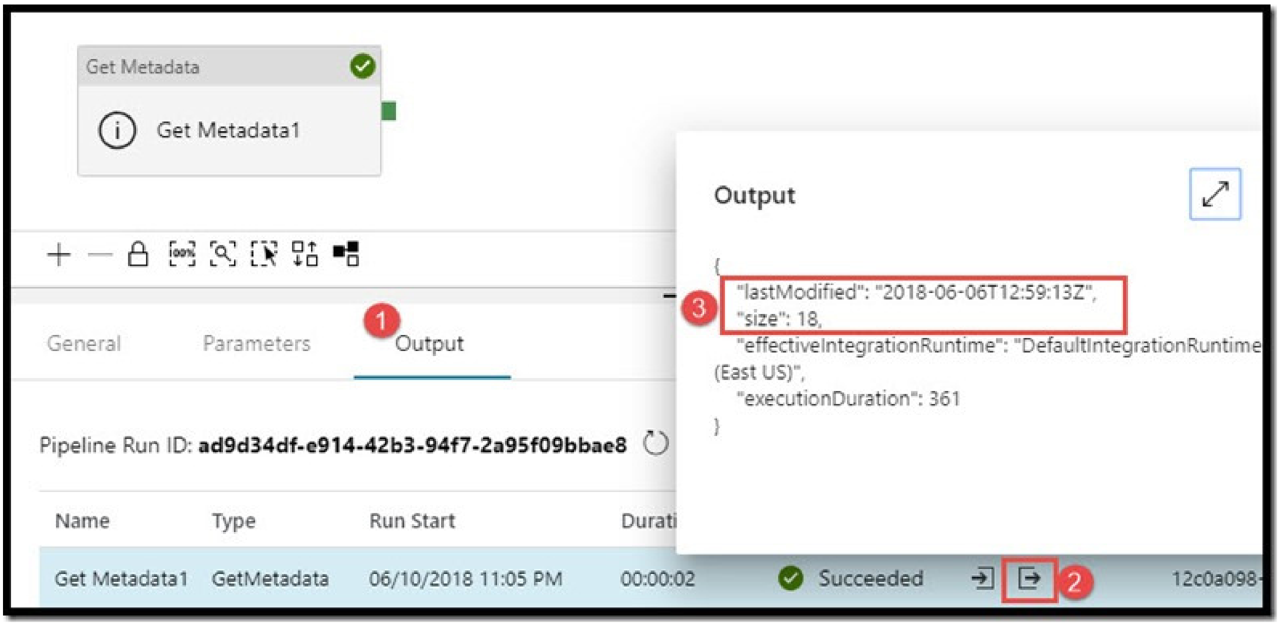
Part two: How to read the outputs from Get Metadata activity

Now that the activity has been configured, it's time to run it in debug mode to validate the output parameters.



Once debug completes you can now take a look at the output of the debug execution for any of the activities in your pipeline. We only have the one activity in this example. Click on the output to see the output values for the items selected:

***Tip**: If you don't see the output of the debug operation, click in the background of the pipeline to deselect any activities that may be selected. The output of the debug operation is a property on the pipeline, not any particular activity.*



```
{
    "lastModified": "2018-06-06T12:59:13Z",
    "size": 18,
    "effectiveIntegrationRuntime": "DefaultIntegrationRuntime
(East US)",
    "executionDuration": 361
}
```

Pipeline Run ID: **ad9d34df-e914-42b3-94f7-2a95f09bbae8**

Name	Type	Run Start	Durati		
Get Metadata1	GetMetadata	06/10/2018 11:05 PM	00:00:02	Succeeded	12c0a098-

Part 3: How to reference the output parameters

For me, this was the hard part - I discovered early on that there is no "Output Parameter" option defined on any of the activities, this is something I just expected since I come from a background of SQL and SSIS. However, not all is lost, you will find that referencing these output parameters is not that difficult and they have a basic pattern you can work with. But that will come after we cover the basics!

For now, let's take a look at the basic pattern:

> **@activity('Get Metadata1').output**

This pattern can be broken down into three basic parts.

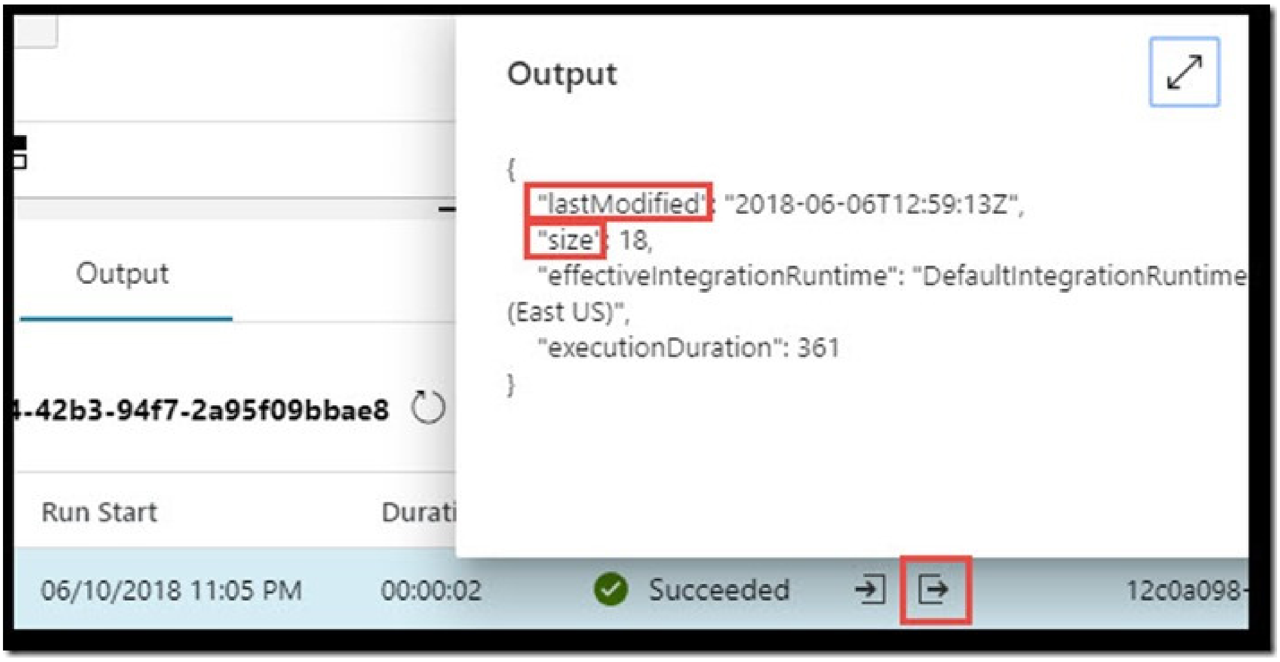
1. **@activity** – specifies you are getting information from an activity in the pipeline.
2. **'Get Metadata1'** – this is the name of the activity that you want to get information from.
3. **.output** – you are looking for an output from the specified activity.

Unfortunately this part is not complete, now you have to specify exactly which output parameter you want, and you have to figure out how to call that parameter. For example, do you want to retrieve the Last Modified date or the Size? What specific name do you use to call that parameter? For example, *@activity('Get Metadata1').output.Last Modified* won't work because Last Modified is the incorrect name of the output parameter. So, the challenge now is figuring out what that output parameter name is so you can use that somewhere else in your pipeline.

So how do you get the specific name of the output parameters?

You can get the specific name of the output parameters by taking a look at the output results of the Debug operation. Take another look at the output results and you will see the exact name needed for the output parameter reference. Last Modified is going to be lastModified and the final code to reference the output parameter will look like the following:

> **@activity('Get Metadata1').output.lastModified**



Output

```
{
   "lastModified": "2018-06-06T12:59:13Z",
   "size": 18,
   "effectiveIntegrationRuntime": "DefaultIntegrationRuntime
(East US)",
   "executionDuration": 361
}
```

-42b3-94f7-2a95f09bbae8

Run Start　Durati

06/10/2018 11:05 PM　00:00:02　Succeeded　12c0a098-

As always, thanks for checking out my blog! Be sure to check out the other blogs in this series to get a better understanding of how to use use the output parameter in other activities.