Lab 5 – Transform and Merge Data with ADF and HDInsight

Technologies showcased: ADF GUI, HDInsight, Hive, Azure Blob Storage

Author: Joshuha Owen

Date: 12/14/2017

Table of Contents

[Summary 3](#_Toc502319934)

[Pre-requisites 4](#_Toc502319935)

[Scenario 4](#_Toc502319936)

[Part 1 – Creating the Pipeline for the Hive Activity 4](#_Toc502319937)

[Part 2 – Calling a Execute Pipeline from another Pipeline 9](#_Toc502319938)

## Summary

This tutorial walks through creating a pipeline Hive activity to merge the FAAmaster and FAAaircraft data together into one file. This is to show how we can leverage Hive to do some transform activities.

In this lab we will:

* Show the Hive activity to run Hive scripts against an HDInsight cluster
* Configure the Hive activity
* Chain one pipeline to another using the Execute Pipeline activity

The Hive scripts we will using were deployed as part of lab module 1 to your Azure Blog Storage container Input folder and is named FAAMerge.hql.

The FAA Master and FAA Aircraft folders were deployed as part of lab module 1 to your Azure Blob Storage container Input folder.

These can be looked at locally in the Deployment folder under the Files -> input directory.

## Pre-requisites

* Azure Subscription with rights to use/deploy Azure services, and X of Azure credit
* Azure Blob storage container
* Azure HDInsight cluster
* Azure Data Factory
* Hive scripts
* Application account with contributor rights to your resource group (created in Lab module 1)

## Scenario

|  |  |  |
| --- | --- | --- |
| Part 1 – Creating the Pipeline for the Hive Activity | | |
| **Scenario** | | |
| We will be setting up a new pipeline to run a Hive query against a HDInsight cluster and output that data to the /output folder in our Azure Blob storage account. | | |
| **Commentary / Notes** | **Click Steps & ‘Bits’** | **Screenshots** |
| We will be using the Azure Data Factory we created in Lab 01 called adflab-adf.. | 1. Navigate to the Azure portal within your web browser and navigate to <https://portal.azure.com> 2. Open the Azure Data Factory blade [adflab-adf], pinned from a previous lab but if not navigate to it using the All Resources menu item. |  |
|  | 1. In the Overview blade you should see the following Quick Links: |  |
|  | 1. Click the Pipeline Editor button and you should see graphical user interface Overview page. |  |
|  | 1. Click the Create Pipeline button under Let’s Get Started. |  |
|  | 1. You should now see Pipeline GUI editor. |  |
|  | 1. Fill out the pipeline name. We will be using Merge FAA Files as our pipeline name. |  |
|  | 1. In the Pipeline GUI, drag the Hive activity to the empty pane above General. 2. Rename the activity to Run FAA Hive Merge. |  |
|  | 1. Click the HDI Cluster tab. 2. Click the +New button next to the HDInsight Linked Service dropdown. |  |
|  | 1. You should now see the HDI Linked Service configuration pane. 2. Fill out the following information: Name: HDI-ADFLab Azure Storage Linked Service: AzStorage-Staging Service principal id: Setup in Lab 1 Service principal key: Setup in Lab 1 Subscription: Your Azure Subscription   Resource Group: Your Azure RG for the lab   1. Expand OS Type. Fill out a Cluster SSH user name and password. 2. Click the Script Tab. |  |
|  | 1. For the Script Linked Service choose AzStorage-Staging. 2. Click the Browse Storage button and browse to the input/FAAmerge.hql file. 3. Click the Publish button to save your Pipeline. |  |
|  | 1. Click the Add/Edit Script button. |  |
|  | 1. Find the 3 lines containing the Azure Storage location wasb://output@adflabstaging 2. Replace adflabstaging with the name of your Azure Storage account created in Lab Module 1. 3. Click Finish. |  |
| Note that after this process runs the HQL file will output the results into wasb://output@adflabstaging.blob.core.windows.net/FAAMerge or in other words our adflabstaging storage account under the blob container output in the FAAMerge folder. Hive will name the file 000000\_0. We will later user this in the Azure DW load as an external table. |  |  |

|  |  |  |
| --- | --- | --- |
| Part 2 – Calling a Execute Pipeline from another Pipeline | | |
| **Scenario** | | |
| We will be using the pipeline we created in Lab 03 (and added a pipeline in Lab 04) and call an execute pipeline to call the pipeline we just created. This will then create a chain of pipelines. | | |
| **Commentary / Notes** | **Click Steps & ‘Bits’** | **Screenshots** |
| This concept is similar to an Execute Package task in SSIS if you are familiar and would allow the creation of complex controller pipeline logic if you need that. | 1. You should now be back at the Pipeline editor screen. If not click on the Pencil icon in the left menu and click the S3 to Blob Copy Pipeline we created in Lab 03. |  |
|  | 1. In the Pipeline GUI, drag the Execute Pipeline activity to the right of the Exec Weather to Blob Copy activity. 2. Rename the activity to Exec Merge FAA Files |  |
|  | 1. Drag from the green square of the Exec Weather to Blob Copy activity to the Exec Merge FAA Files activity. 2. Click the Exec Merge FAA Files activity. |  |
|  | 1. Click the Settings tab. 2. In the Invoked Pipeline dropdown select the Merge FAA Files pipeline we created in Part 1 of this lab. 3. Click the Publish button to save your Pipeline. |  |

**IMPORTANT: AVOID INCURRING EXTRA CHARGES BY PAUSING YOUR SUBSCRIPTION RESOURCES**