Contents

Integrate Azure Data Factory with Databricks

Imagine you are part of an analytics team that has recently received a huge assignment of analysing crime data of several metropolitan cities. The dataset that you received has detailed crime information for major cities. However, each dataset is formatted and structured differently and stored in different data stores. Each city uses a different category and terms for similar type of data. Your team is responsible to analyse all the datasets and report aggregated number of crimes per month in each city.

Your team has decided to leverage the capabilities of Azure Data Factory and Azure Databricks to ingest, transform, and aggregate the required data

- In this module, you will:
  - Use ADF to orchestrate data transformations using a Databricks Notebook activity.

1. ADF and Azure Databricks

You can use Azure Data Factory to ingest raw data collected from different sources and work with Azure Databricks to restructure it as per your requirements. The integration of Azure Dataricks with ADF allows you to add Databricks notebooks within an ADF pipeline to leverage the analytical and data transformation capabilities of Databricks. You can add a notebook within your data workflow to structure and transform raw data loaded into ADF from different sources. Once the data is transformed using Databricks, you can then load it to any data warehouse source.

Requirements

Data ingestion and transformation using the collective capabilities of ADF and Azure Databricks essentially Involves the following steps:

1. Create Azure storage account - The first step is to create an Azure storage account to store your ingested and transformed data.
2. Create an Azure Data Factory - Once you have your storage account setup, you need to create your Azure Data Factory using Azure portal.
3. Create data workflow pipeline - After your storage and ADF is up and running, you start by creating a pipeline, where the first step is to copy data from your source using ADF's Copy activity. Copy Activity allows you to copy data from different on-premises and cloud sources.
4. Add Databricks notebook to pipeline - Once your data is copied to ADF, you add your Databricks notebook to the pipeline, after copy activity. This notebook may contain syntax and code to transform and clean raw data as required.
5. Perform analysis on data - Now that your data is cleaned up and structured into the required format, you can use Databricks notebooks to further train or analyse it to output required results.

2. Create Azure Storage Account

1. In the Azure portal, select "**Create a resource**", enter "**storage account**" into the Search the Marketplace box and select Storage account - blob, file, table, queue from the results, and select **Create**.
2. In the Create storage account blade, enter the following:
   - Subscription: Select the subscription you are using for this module.
   - *Resource group*: Select "create new" and enter a unique name.
   - *Storage account name*: Enter a unique name (make sure you see a green checkbox).
   - *Location*: Select the location closest to your physical location.
   - *Performance*: Select Standard.
   - *Account kind*: Select Storage (general purpose v1).
   - *Replication*: Select Locally redundant storage (LRS).
3. Select Next: **Advanced >**.
4. In the Advanced tab, select the following:
   1. *Secure transfer required*: Select Disabled
   2. *Virtual network*: Select None
5. Select **Review + create**.
6. In the Review tab, select **Create**.

2.1 Acquire account name and key

1. Once provisioned, navigate to your storage account.
2. Select Access keys from the left-hand menu and copy the Storage account name and key1 Key value into a text editor, such as Notepad, for later use.

2.2. Acquire account name and key

1. Select Blobs from the left-hand menu, then select **+ container** to create a new container.
2. Enter *dwtemp* for the container name.
3. Leave the public access level selected as *Private*
4. Select **OK**.

You have your storage account and Azure Data Factory up and running, now it's time to switch to your Databricks workspace to complete rest of the workflow. We'll use a sample dataset to create an ADF pipeline and use sample notebooks to transform and analyse the data.

3. Create an Azure Databricks Workspace

1. In the Azure portal, select "**Create a resource**", enter "**Databricks**" into the Search the Marketplace box and select Storage account - blob, file, table, queue from the results, and select **Create**.

3.1. Clone the Databricks archive

1. From the Azure portal, navigate to your Azure Databricks workspace and select **Launch Workspace**.
2. Within the Workspace, using the command bar on the left, select **Workspace**, **Users**, and select your username (the entry with house icon).
3. In the blade that appears, select the downwards pointing chevron next to your name, and select **Import**.
4. On the Import Notebooks dialog, select URL and paste in the following URL:
5. https://github.com/MicrosoftDocs/mslearn-data-ingestion-with-azure-data-factory/blob/master/DBC/03-Data-Ingestion-Via-ADF.dbc?raw=true
6. Select **Import**.
7. A folder named after the archive should appear. Select that folder.
8. The folder will contain one or more notebooks that you'll use in completing this lab.

Complete the following notebooks

1. **01 Getting Started** - This notebook contains instructions for setting up your storage account and Azure Data Factory (ADF). If you've already set up your storage account in the previous unit, you can skip this notebook.

2. **02 Data Ingestion** - In this notebook you create an ADF v2 pipeline to ingest data from a public dataset into your Azure Storage account. Once the data is ingested, you use Databricks notebook function to examine the data.

3. **03 Data Transformation** - This notebook contains instructions to create connectivity between your Azure Data Factory and Databricks workspace. You use a sample notebook to add to your ADF pipeline that will transform and restructure your data. You'll also perform some basic aggregation on the sample dataset to generate required reports.

Module Conclusion

Azure Data Factory allows you to ingest raw unstructured data from different sources. The integration between ADF and Azure Databricks helps you to create end-to-end data workflow to ingest, prepare, transform, and load your data into any destination storage in a required format and structure.