

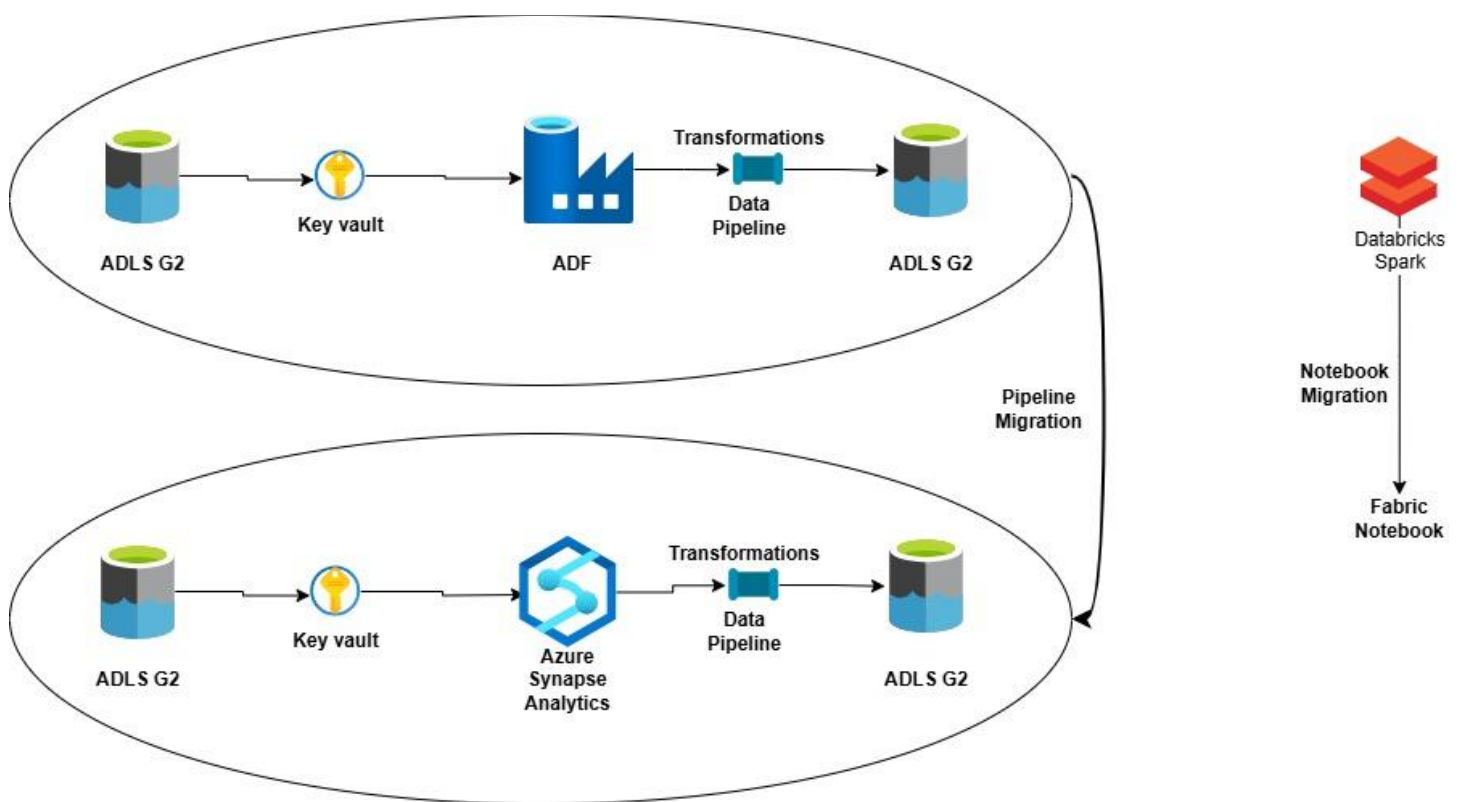
Bootcamp Project – 5

Migrating pipelines from ADF to Synapse

Problem statement:

- The purpose of this project is to learn and implement and try different methods to migrate the pipeline from ADF to Synapse
- In the next phase, we are going to migrate the notebooks from data bricks to Fabric environment.

Project Architecture :



FIRST PHASE :

- The best way to migrate the data pipeline from ADF to synapse or vice versa :
 - Manually created all linked services and datasets in target service and copy the Json file of the pipeline.
 - We will try this method first :

The screenshot displays the Azure Data Factory (ADF) console for a pipeline named 'pipeline-5-adf'. The left sidebar shows the 'Activities' menu with categories like 'Move and transform', 'Synapse', 'Azure Data Explorer', 'Azure Function', 'Batch Service', 'Databricks', 'Data Lake Analytics', 'General', 'HDInsight', 'Iteration & conditionals', 'Machine Learning', and 'Power Query'. The main canvas shows a 'Copy data' activity being added to the pipeline. Below the canvas, the 'General' tab of the activity configuration is visible, showing fields for 'Name' (Copy data), 'Description', 'Activity state' (Activated), and 'Timeout' (0.12:00:00). A 'Learn more' link is also present.

In this phase, we have created pipeline with copy activity, and we are trying to copy the csv files from ADLS G2 to different container in ADLS in parquet format.

Name ↑↓	Type ↑↓	Related ↑↓	Annotations ↑↓
adls_ls	Azure Data Lake Storage Gen2	1	
adls_ls_sink	Azure Data Lake Storage Gen2	1	
keyvault_ls	Azure Key Vault	2	

These are the linked services created for source and sink and for key vault respectively.

Datasets 2

- ds_csv
- ds_par
- Data flows 0
- Power Query 0

Connection Schema Parameters

Linked service * adls_ls [Test connection](#) [Edit](#) [+ New](#) [Learn more](#)

File path raw / user / CustomerServiceInteraction... [Browse](#) [Preview data](#) [Detect forr](#)

Compression type No compression

Column delimiter Comma (,)

Row delimiter Default (\r\n, or \r\n)

Encoding Default(UTF-8)

Quote character Double quote (")

Escape character Backslash (\)

These are the datasets for source and sink files respectively.

Here I am accessing ADLS G2 by account key and it is being stored in key vault secret.

Key vaults Default Directory

[+ Create](#) [Group by none](#)

[You are viewing a new version of Browse experience. Some features may be missing. Click here to access the old experience.](#)

☐ Name ↑

☐ mann-key-vault



mann-key-vault | Secrets ☆ ...


[Search](#) [Generate/Import](#) [Refresh](#) [Restore Backup](#) [Manage deleted secrets](#) [View sample code](#)

[Overview](#) [Activity log](#) [Access control \(IAM\)](#) [Tags](#) [Diagnose and solve problems](#) [Access policies](#) [Resource visualizer](#) [Events](#) [Objects](#) [Keys](#) **Secrets**

Name	Type	Status	Expiration date
adls-account-key		✓ Enabled	

Edit linked service


 Azure Data Lake Storage Gen2 [Learn more](#) 

Connect via integration runtime * 

 AutoResolveIntegrationRuntime 

Authentication type

Account key 

Account selection method 


☐ From Azure subscription ☒ Enter manually


URL *


https://rawmann.dfs.core.windows.net/

Storage account key

Azure Key Vault

AKV linked service * 

keyvault_Is 

Secret name * 

adls-account-key

☒ Edit


Secret version 

Latest version 

☐ Edit

Save

Cancel

 Test connection

I am using key vault to access account key in the linked service of source and sink.

Activities

- Move and transform
- Synapse
- Azure Data Explorer
- Azure Function
- Batch Service
- Databricks
- Data Lake Analytics
- General
- HDInsight
- Iteration & conditionals
- Machine Learning
- Power Query

Copy data

Copy data

Pipeline run ID: d390766c-de3d-41c1-a75c-43081d1df85a **Pipeline status:** Succeeded

Output

Showing 1 - 1 of 1 items

Activity name	Activity status	Activity name	Run start	Duration	Integration runtime
Copy data	Succeeded	Copy data	5/11/2025, 1:31:27 PM	18s	AutoResolveIntegrationRuntime (East US)

Here is the result of adf pipeline.

Migrating to Synapse:

For this we have to first create linked services and data sets with the same name as given in ADF.

Linked services

Linked services are much like connection strings, which define the connection information needed for Azure Synapse Analytics to connect to external resources.

Filter by name **Annotations: Any**

Showing 1 - 5 of 5 items

Name	Type	Related	Annotations
adls_ls	Azure Data Lake Storage Gen2	1	
adls_ls_sink	Azure Data Lake Storage Gen2	1	
dev-synapse-workspace-1-WorkspaceDefaultSqlServer	Azure Synapse Analytics	0	
dev-synapse-workspace-1-WorkspaceDefaultStorage	Azure Data Lake Storage Gen2	0	
keyvault_ls	Azure Key Vault	2	

Here I have created Linked services in the synapse and test the connections.

Microsoft Azure | Synapse Analytics ▶ dev-synapse-workspace-1

>> Synapse live ▾ ✓ Validate all ⬆ Publish all

Data

Workspace Linked

Filter resources by name

- ▲ Azure Data Lake Storage Gen2 2
 - ▶ dev-synapse-workspace-1 (Primary...
 - ▶ (Attached Containers)
- ▲ Integration datasets 2
 - ds_csv
 - ds_par

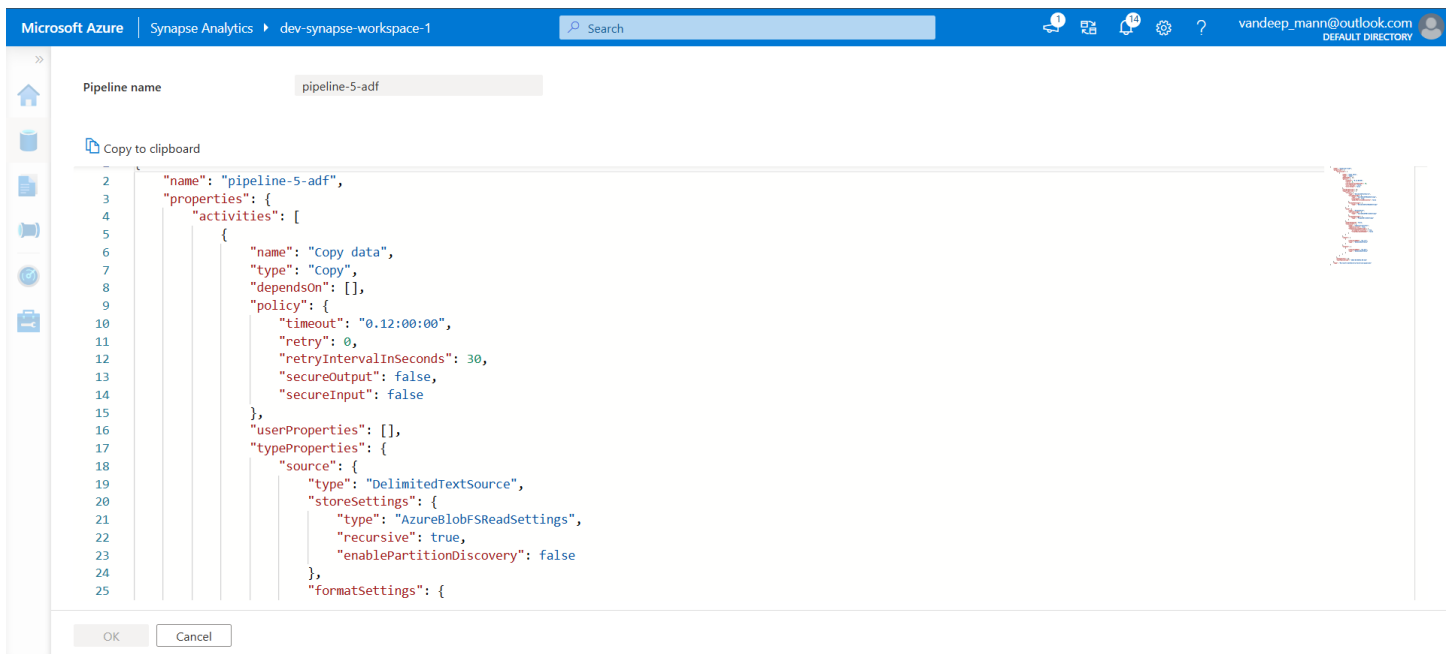
Activities

Search activities

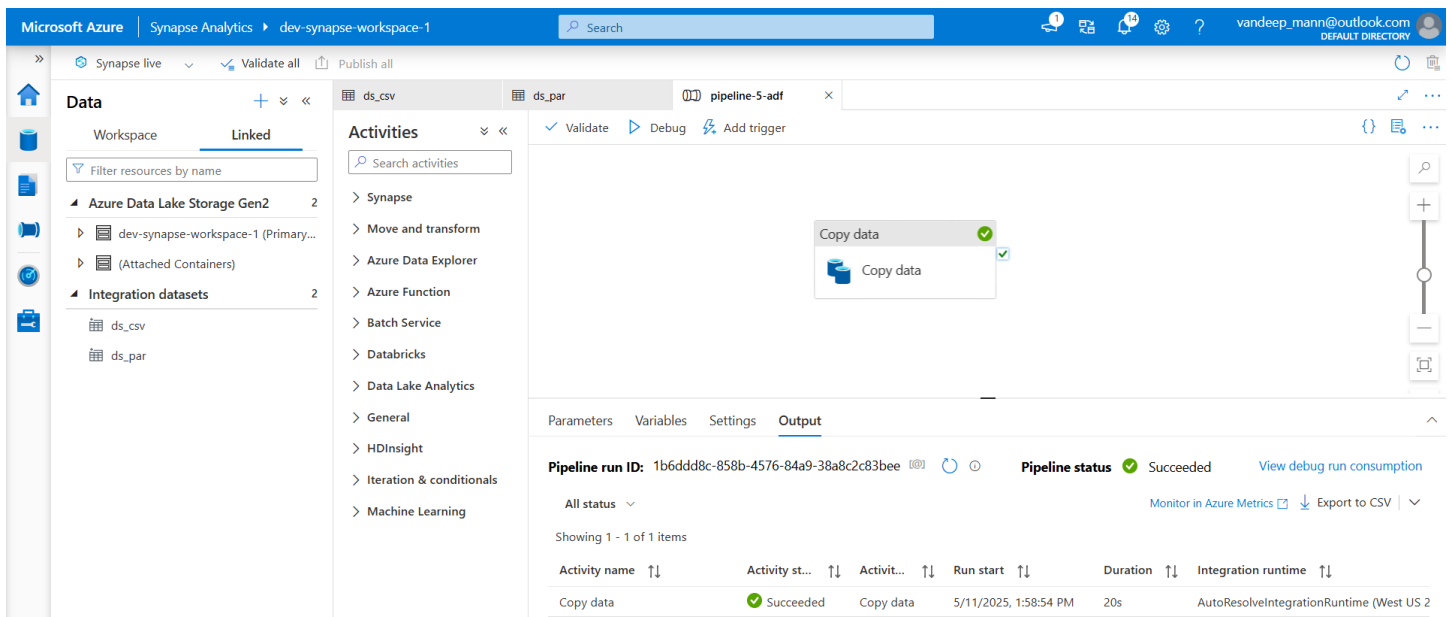
- > Synapse
- > Move and transform
- > Azure Data Explorer
- > Azure Function
- > Batch Service
- > Databricks
- > Data Lake Analytics
- > General
- > HDInsight
- > Iteration & conditionals
- > Machine Learning

I have also created Datasets with same names in synapse too.

Then. Create a pipeline in synapse and name it same as given in adf and go to it's json code and paste the whole json code in it.



Here pipeline has been created in synapse working successfully



2ND PHASE OF THE PROJECT:

To migrate notebook from databricks to Fabric.

Notebook1 Python ▾ Tabs: OFF ▾ ☆

File Edit View Run Help [Last edit was 3 minutes ago](#)

▶ Run all demo cluster ▾ Schedule Sha

```
from pyspark.sql import SparkSession
from pyspark.sql import Row

# Sample data
data = [("Alice", 34), ("Bob", 45), ("Cathy", 29)]

# Create DataFrame
df = spark.createDataFrame(data, ["Name", "Age"])

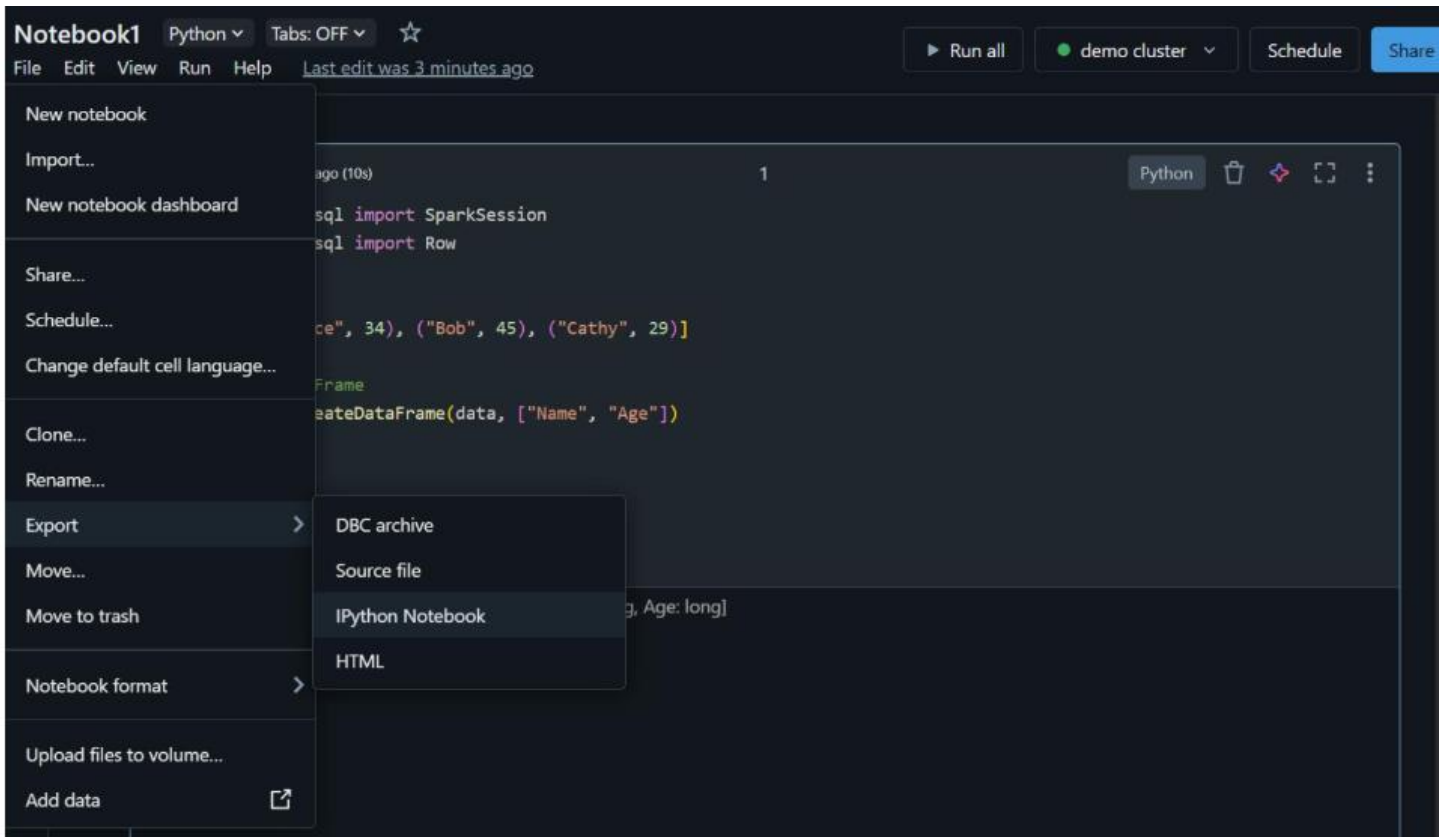
# Display
df.show()
```

▶ (2) Spark Jobs

▶ df: pyspark.sql.dataframe.DataFrame = [Name: string, Age: long]

Name	Age
Alice	34
Bob	45
Cathy	29

Export the notebook in ipynb



Now we can import those notebooks in fabric.

