# END To END ETL Implementation Project

A screenshot of a computer

Description automatically generated

Contents

[***END To END ETL Implementation Project*** 1](#_Toc146496855)

[1.Set Up Environment 3](#_Toc146496856)

[2.Data Ingestion using ADF 6](#_Toc146496857)

[3.Data Transformation With Databricks 15](#_Toc146496858)

[4.Creating Views Using Synapse 17](#_Toc146496859)

[5. Publishing Data Analytics Reports – Power BI 22](#_Toc146496860)

[6. Security and Governance 24](#_Toc146496861)

[7. End to End Pipeline Testing 25](#_Toc146496862)

[8. References 27](#_Toc146496863)

Scenario :   
Performing a migration from On-Premise PostgreSQL server to Azure cloud and doing a complete End to end implementation which includes Migration, Extraction, Transformation ,Loading and lastly Analytics .

## 1.Set Up Environment

* 1. For the set up environment I created a new resource group for the project and have the following resources up and running.

These Include : **Azure Databricks, Azure Data Factory, Azure Storage Account, Azure Synapse Analytics Workspace and Azure Key Vault**

A screenshot of a computer

Description automatically generated

* 1. In my On-Premise Microsoft SQL Server/PostgreSQL, Tables which are being migrated :

A screenshot of a computer

Description automatically generated

* 1. Created Azure Key Vault secrets Which comprise secrets created for the SQL server user Login Id and Password

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Also, create a linked service for the key vault, as this needs to be linked whenever key vault is used.

A screenshot of a computer

Description automatically generated

* 1. Storage account and containers

A screenshot of a computer

Description automatically generated

## 2.Data Ingestion using ADF

A diagram of data ingestion

Description automatically generated

* 1. Creating a Self-Hosted Integration Runtime in Azure Data Factory as we are connecting to the on-premises server.

A screenshot of a computer

Description automatically generated

* 1. Create a dataset and a linked for the source of Copy data pipeline

Linked Service for Postgresql

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* 1. Creating a pipeline in ADF with Lookup activity .

A screenshot of a computer

Description automatically generated

**Query** : select table\_name from information\_Schema.tables

where table\_schema = 'public'

Validated the activity and debugged the activity to get the output as follows

A screenshot of a computer

Description automatically generated

The JSON format output includes the list of tables in my PostgreSQL database.

* 1. Now, with the for each activity, we configure the output from the look up tables ; A screenshot of a computer

     Description automatically generated

In the items, we add dynamic content and take the output from lookup activity with expression

A screenshot of a computer program

Description automatically generated

* 1. Configure a **copy activity** as a sub activity in the for each activity

A screenshot of a computer

Description automatically generated

Created a new dataset with the existing self hosted Integration runtime

A screenshot of a computer

Description automatically generated

A simple Concat query with the dynamic content from the previous activity  
 @{concat('Select \* FROM ', item().table\_name)}

Sink Dataset with parquet format , with table name being parameterised with dynamic content , as we have multiple tables being migrated.

A screenshot of a computer

Description automatically generated

Configured the dataset to store the dataset in parquet and store it in a hierarchical fashion.

A screenshot of a computer

Description automatically generated

**Container\_name/OrganisationLT/Table\_name/Tablename.parquet**

**For Direcotry :** @{concat('OrganisationLT/',dataset().tablename)}

For File name : @{concat(dataset().tablename,'.parquet')}

* 1. Validated and published the pipeline and triggered the pipeline. Ran Successfully.

A screenshot of a computer

Description automatically generated

* 1. Validated for the files in the Storage account

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

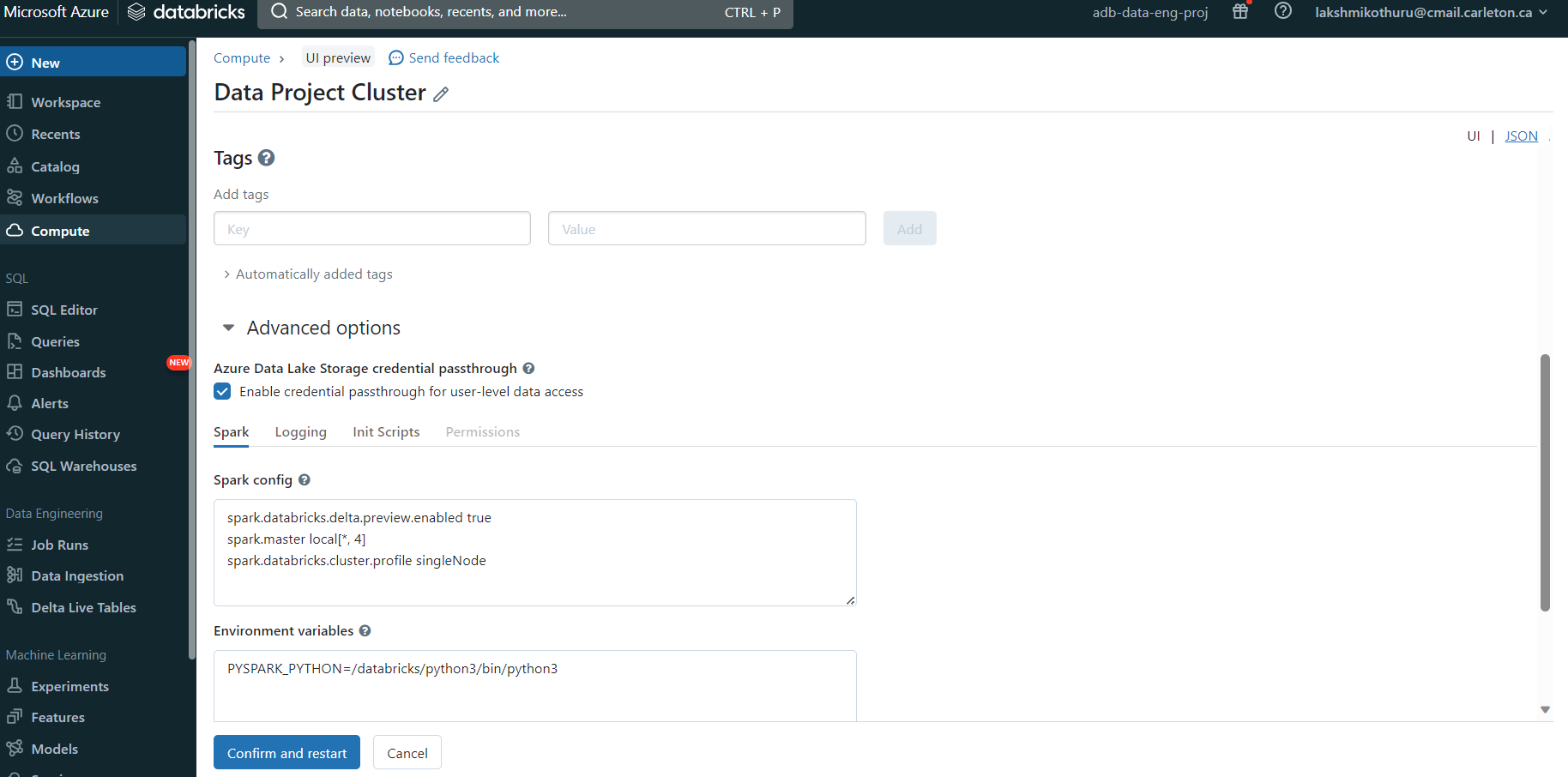
Description automatically generated

A screenshot of a computer

Description automatically generated

## 3.Data Transformation With Databricks

3.1 Create a databricks single-node cluster while enabling credential passthrough for user-level data access



[Access Azure Data Lake Storage using Azure Active Directory credential passthrough (legacy) - Azure Databricks | Microsoft Learn](https://learn.microsoft.com/en-us/azure/databricks/data-governance/credential-passthrough/adls-passthrough)

I did mounting through credential passthrough method, which surprisingly worked well, unlike in my other projects where I couldn’t make this work.

3.2 Created a databricks notebook activity for bronze to silver notebook, created linked service and tried utilizing Keyvault.

Since My account is not a premium account I’m unable to load the key vault secret name, even after applying role assignments.

A computer screen with a message box

Description automatically generated

So I directly used the Access token , without Key Vault ,which is not a preferred approach

A screenshot of a computer

Description automatically generated

3.3 Pipeline Run Successfully

A screenshot of a computer

Description automatically generated

* 1. Verifying for data in the gold container

A screenshot of a computer

Description automatically generated

## 4.Creating Views Using Synapse

Created a Synapse Workspace,a database **gold\_db** and used the linked feature to look at the storage containers which also link the tables

A screenshot of a computer

Description automatically generated

Created a procedure for parameterising and creating the views for all the tables in the gold container

A screenshot of a computer

Description automatically generated

Now I’m creating a pipeline for automating this task , creating a new linked service for the SQL database(serverless) which has the stored procedure.

A screenshot of a computer

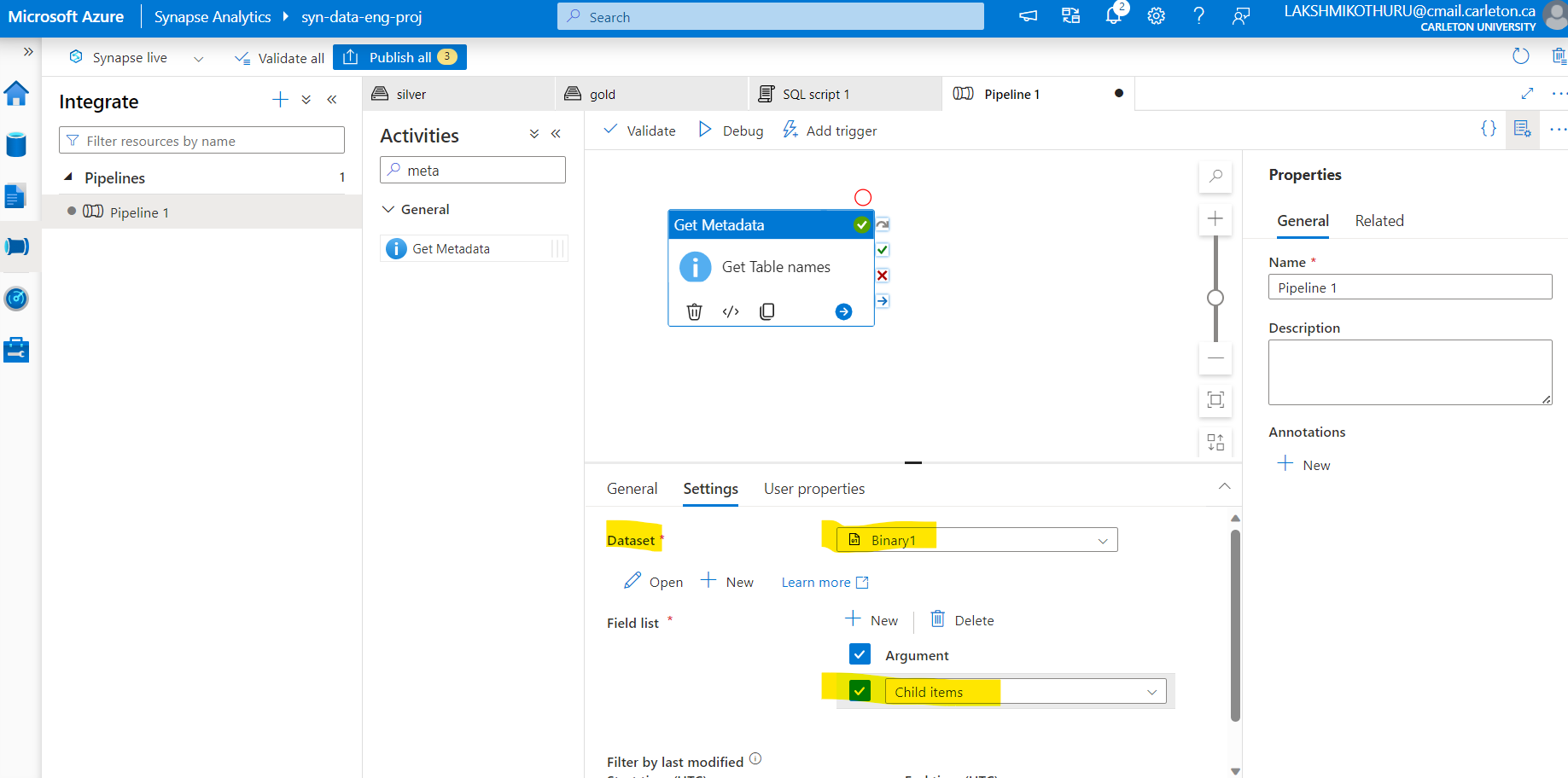
Description automatically generated

Entered Connection string details manually as we can’t populate the database name .

Used the domain name which can be found under **syn-data-eng-proj** **| Properties| Serverless SQL endpoint**

Create a pipeline in the synapse studio and use the activity GET METADATA and create a dataset in binary format as we need only table names.

Pass the Field list argument as Child Items as we need table name .



Next, I’m using ForEach, passing the childnames to the Stored procedure I created earlier.

A screenshot of a computer

Description automatically generated

Expression to get table name from the GET METADATA activity

A computer screen with a blue screen

Description automatically generated

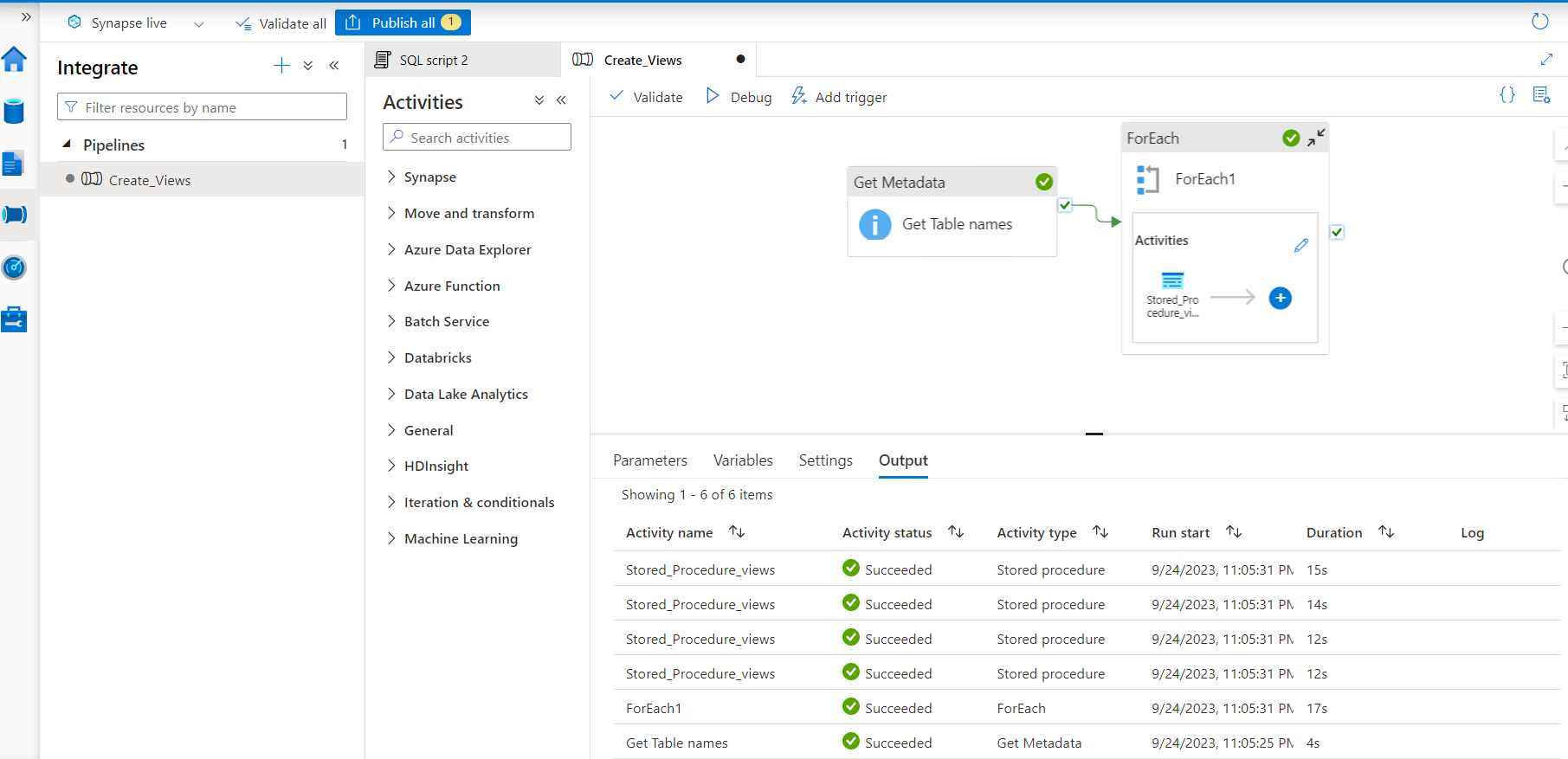
In the For each activity, I’m now calling a stored procedure activity and linking the stored procedure created in SQL Pool.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

Pipeline Ran successfully

Vailidating the views in Serverless SQL POOL

A screenshot of a computer

Description automatically generated

## 5. Publishing Data Analytics Reports – Power BI

Here I’m Importing data from Azure SQL Database. I’ve given the Serverless SQL server endpoint , which can be found **synapse-workspace** | Properties| Serverless SQL endpoint

**A screenshot of a computer

Description automatically generated**

Signed-in with Microsoft account, now able to connect to synapse workspace and import data

A screenshot of a computer

Description automatically generated

Relationship established automatically.

A screenshot of a computer

Description automatically generated

Data Analysed and plotted as a report in Power BI

A screenshot of a graph

Description automatically generated

## 6. Security and Governance

Next I’ll have to create a security group in Azure active directory/Microsoft Entra ID for easy access control in a real-time scenario.

Unfortunately, as mine is a student account I’m unable to access this page.

A screenshot of a computer

Description automatically generated

## 7. End to End Pipeline Testing

This is a end to End Pipeline, meaning any update in the data in the database tables, should automatically be updated everywhere .

To do this we initiate a schedule trigger to run everyday at a specific time.

A screenshot of a computer

Description automatically generated

Before this to see the updated the database with 2 records for testing end to end

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

The Trigger ran successfully, leading to both the pipelines in ADF in Synpase running successfully .

When the Power BI report is refreshed we get the following result.

A screenshot of a computer

Description automatically generated

## 8. References

1. [Azure Data Engineering Project – Mr K Talks](https://www.youtube.com/playlist?list=PLrG_BXEk3kXx6KE4nBmhf6QwSHMbznP2W)
2. [Access Azure Data Lake Storage using Azure Active Directory credential passthrough (legacy) - Azure Databricks | Microsoft Learn](https://learn.microsoft.com/en-us/azure/databricks/data-governance/credential-passthrough/adls-passthrough)
3. [Access Azure Data Lake Storage using Azure Active Directory credential passthrough (legacy) - Azure Databricks | Microsoft Learn](https://learn.microsoft.com/en-us/azure/databricks/data-governance/credential-passthrough/adls-passthrough)
4. [pyspark.sql.DataFrame.join — PySpark master documentation (databricks.com)](https://api-docs.databricks.com/python/pyspark/latest/pyspark.sql/api/pyspark.sql.DataFrame.join.html)