Incubation Lab – Pepsico Project Simulation

**Project Documentation of Milestones**

**Tajinder Singh**

**Milestone 1:**

**Goal:** Understand the **business requirement** and create a high-level **architecture** on Azure

You need to **perform** the following **steps**:

* Please go through the business requirement
* Identify the appropriate Azure services
* Understand the Sales **schema** of the Adventure Works dataset. Refer this [link](https://i0.wp.com/improveandrepeat.com/wp-content/uploads/2018/12/AdvWorksOLTPSchemaVisio.png?ssl=1)
* Create a high-level architectural diagram using [draw.io](https://www.draw.io/)

The business requirement is to design and implement an end-to-end solution using Microsoft Azure platform. The source data will be in a SQL DB from where it will be fetched in Azure Storage and it will undergo various transformations to prepare the final output, in the form of visualization and curated data in warehouse.

We will need following **Azure services**:

* Azure SQL DB with server
* Azure Data Lake Storage Gen 2
* Azure Data Factory
* Azure Databricks
* Azure Synapse Workspace
* Azure Key Vault

**Architectural Diagram** of the pipeline:

A picture containing diagram

Description automatically generated

**Milestone 2:**

**Goal**: Set up source database & create data dictionary

You need to **perform** the following **steps**:

* **Import** Adventure Works sample database in **Azure SQL**
* **Analyse** this source database (Sales Schema)
* Create a data dictionary in Excel (all the tables, columns, data type, profiling statistics).

1. **Create a Resource Group.**

Text, application

Description automatically generated with medium confidence

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, chat or text message

Description automatically generated

Finish creating steps by pressing ‘Review+Create’ button. Resource Group **IncubationLab** is created.

1. **Create an Azure SQL Database.**
2. Click on Create a resource and search for SQL Database.
3. Give Subscription and Resource Group as the one that we created in for this project.
4. Give the Database name as **AdventureWorks** and create a new server **incubation** for this specific project and resource group as **IncubationLab** and sql authentication.

A picture containing graphical user interface

Description automatically generated

1. Enable Sample data which has AdventureWorks Database.

A screenshot of a computer

Description automatically generated with medium confidence

1. Enable these features so that we can use SQL DB with Data Factory and access database in SQL Server Management Studio.

Graphical user interface, text, application, chat or text message

Description automatically generated

1. Analyse the data in SSMS with SQL authentication. Connect to DB with server endpoint, username and password.

Graphical user interface, application

Description automatically generated

1. Data Dictionary of tables:

<IncubationLabDataDictionary.xlsx>

**Milestone 3:**

**Goal:** Use ADF to copy data from Azure SQL to Landing Zone to bronze zone

You need to **perform** the following **steps**:

* Create required resources in the data lake for landing and bronze zone.
* Using ADF, **import** data from Azure SQL to Landing Zone in Data Lake
* Import only the **Sales** schema tables in Data Lake.
* Then data from the landing zone will be copied to the bronze zone without any change.
* Test the above pipeline.

1. **Create ADLS storage with following structure:**

Graphical user interface, application

Description automatically generated

1. **Set up Azure Key Vault Service for storing AZ SQL DB connection string and ADLS Account Key as Secrets.**
2. Create a key vault instance with resource group and subscription with following details

Graphical user interface, text, application, email

Description automatically generated

1. Go to the instance and under secrets add new secret for ADLS and AZ SQL DB.

Graphical user interface, text, application

Description automatically generated

1. Here give the name of secret and value as Account Key for ADLS and connection string for AZ SQL DB.

Graphical user interface, text, application

Description automatically generated

1. We created 2 secrets as follows:

Table

Description automatically generated with medium confidence

1. **Create a data factory**
2. Click on Create a resource and search for Data Factories on next page click on create and give name as **IncubationProjectDF.**
3. Under Author section, click to create a pipeline.

Graphical user interface, text, application

Description automatically generated

1. Create the pipeline with following activities:
2. Lookup Activity – It will list all tables in the AdventureWorks DB with a query
3. ForEach Activity – It will copy each table to ADLS Gen to landingzone folder as it is.
4. CopyDataActivity – It will copy data from landingzone to raw/bronze zone in ADLS.
5. Create all linked services to access AZ SQL DB and ADLS through the Key Vault Linked Service, use **secret name and secret version**.

Graphical user interface, text, application

Description automatically generated

Graphical user interface

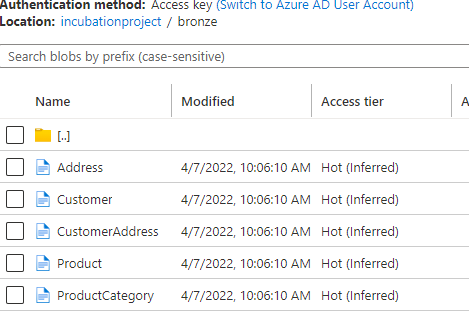
Description automatically generated

1. When the pipeline is triggered:
2. Landing Zone

Graphical user interface, text, application, email

Description automatically generated

1. Bronze Zone



**Milestone 4:**

**Goal:** Extend the above ADF pipeline to copy data from bronze to silver zone after transformation using **Databricks**

* You need to **perform** the following **steps**:
* Extend the pipeline to create one more step
* In this step, the data will be picked from bronze and transformed/processed using data bricks, and then the output will be written to the silver zone.
* Perform basic level transformation on the data.
* Suggest different options to use data bricks in a cost-effective way.
* Test the above pipeline
* Following transformation to be added:
* Ingest ONLY the required columns
* Can we add some NULL handling transformation for all the columns? Below is the common rule for null handling.
  + - * STRING -> NA
      * INT -> -1
      * DATE/TIMESTAMP -> 1900-01-01
* Column Renaming transformation
* Type Casting with the required data type in silver layer
* Filtering Rules to be applied
* Add audit columns
* Error handling (bad records option OR different Mode option such as PERMISSIVE, DROPMAL FORMED, FAILFAST)

1. **Create a Databricks Instance *IncubationProjectDB* and launch workspace to configure the cluster to run notebook in databricks.**

Graphical user interface, text, application

Description automatically generated

1. Set up the cluster with following configuration:

Graphical user interface, text, application

Description automatically generated

1. Go to this link and create a secret scope for databricks:

<https://adb-8136760688538417.17.azuredatabricks.net/#secrets/createScope>

Here add highlighted details from Azure Key Vault Service properties:

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, application

Description automatically generated

1. Now we can ADLS csv files in Databricks notebook for basic transformations(select, rename and typecast column).

Graphical user interface

Description automatically generated

1. Write the output to silver zone as parquet file using df.write.parque(…)
2. Now go to settings in Databricks and generate a token to access notebook from Data Factory.

Graphical user interface

Description automatically generated

Graphical user interface, text, application, chat or text message

Description automatically generated

dapi70a79badcae4c1bfb1537ea723198208-3

dapi1c8a53fc479c47ec44efc94f65be6cde

1. Store the token in AZ Key vault secrets to access it via Linked Service.

Graphical user interface, text, application

Description automatically generated

1. Add a notebook activity in data factory and configure it as follows:

Graphical user interface, text, application

Description automatically generated

1. Now whole pipeline looks like this. Trigger and check the output in silver zone and we will see individual folder with parquet files.

**Pipeline:**

A picture containing diagram

Description automatically generated

**Silver Zone:**

Graphical user interface, text, application, email

Description automatically generated

**Parquet File:**

Graphical user interface, text, application, email

Description automatically generated

**Milestone 5:**

**Goal:** Make the pipeline **configuration driven**

You need to **perform** the following **steps**:

* The above data bricks job should **read parameters** from a configuration file
* Perform basic validations (Column level) using Databricks
* Register Delta Table
* Merge the incoming file into the target (Insert new records and update existing ones)
* Calculate column level profiling stats and store it in ADLS as well (with every run)

1. **Create Config file**
2. Create a config.json file to avoid hard coding of paths and column names. It looks like this.

A picture containing text

Description automatically generated

1. Upload config file in Databricks File System(DBFS).

Graphical user interface, application

Description automatically generated

Access it using following command in notebook.

Text

Description automatically generated

1. Now all parameters can be referred in variables using below method.

Text

Description automatically generated

1. Write transformation on tables using the config file.

Text

Description automatically generated

1. **Register delta tables**
2. First create a Database to store delta tables

Text

Description automatically generated with medium confidence

1. At the end of transformation write the dataframe to delta table in databicks



1. Tables will be created in DBFS and ADLS

Graphical user interface, text, application

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

1. **Merge new data to tables**
2. Create two records (1 to update and 1 to insert) in delta table

Graphical user interface

Description automatically generated

1. Merge this table with original delta table and write to ADLS as well.

Text

Description automatically generated

Text

Description automatically generated

1. **Calculate column level profiling stats**

Text

Description automatically generated

**Milestone 6:**

**Goal:** Createanother ADF pipeline to copy data from silver zone to gold zone after it goes transformation.

You need to **perform** the following **steps**:

* Design & create a simple **star schema** and **aggregate/reporting tables** in Databricks and Azure Synapse Analytics (based on the source data) – Gold zone tables
* Develop and schedule Databricks notebooks to transform the silver zone tables and load data in Gold zone tables (including Synapse tables)
* Test this pipeline.

1. The following tables from the Silver Zone were used to create star schema with Dimension and Fact Table.
2. Customer
3. Address
4. Product
5. ProductCategory
6. SalesOrderHeader
7. SalesOrderDetail

Star Schema of the tables is as follows:

Diagram

Description automatically generated

1. Create another databricks notebook with having code for creating Dimensions and Fact Table. Join multiple Dimension Tables to create Fact table having numerical columns and columns for analysis.

Write the Fact Table to Gold Zone in ADLS.

1. Create a Synapse analytics workspace with 100DWc(Minimum) Dedicated SQL Pool.

Graphical user interface, text, application, email

Description automatically generated

Store the connection string of this dedicated pool in Key Vault so that it can accessed for writing data using Data Factory.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Connect to this pool in SSMS with dedicated endpoint URL and create a FactSales table with same schema of the table that was created in gold zone.

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

1. Add two activities in the Data Factory.
2. For the notebook which generates Dimension and fact Tables and writes it to the Gold zone in ADLS.
3. ii) Copy data activity which copied data from the Gold zone to Synapse table in dedicated pool.

Waterfall chart

Description automatically generated with medium confidence

Output in Synapse when pipeline is run.

A screenshot of a computer

Description automatically generated with medium confidence

**Milestone 7:**

**Goal:** Integration with Power BI

You need to **perform** the following **steps**:

* Connect PowerBI with Synapse and Databricks to run basic queries and create a very simple dashboard (parameterized).

1. Install PowerBI Desktop Edition in PC and open it. To analyze the data in dedicated SQL pool in Azure Synapse Analytics, click on Get Data->More->Azure->Azure Synapse Analytics SQL

Graphical user interface, application

Description automatically generated

In server name give dedicated endpoint URL of the dedicated pool and select Direct Query(if the data changes frequently use it) or import.

Graphical user interface

Description automatically generated

Under File->Options and settings->Data Source Settings->Data sources in current file->Edit Permissions->Edit->Database give username and password for the dedicated pool.

Graphical user interface, application

Description automatically generated

Now different types of the visualizations such as Bar chart, Pie chart, Map etc. can be created to analyse the data. Some charts are as below:

Chart, bar chart

Description automatically generated

Chart, pie chart

Description automatically generated

**Milestone 8:**

**Goal:** Send Email notification alert

You need to **perform** the following **steps**:

* Extend ADF pipeline to send email notification in case of failure or pipeline completion (all tables loaded).

1. Create a logic app instance in Azure with following details:

A screenshot of a computer

Description automatically generated

When logic app is created go to Logic app designer under Development Tools in sidebar and configure the logic app to send the outlook mail when HTTP request is triggered.

A screenshot of a computer

Description automatically generated with medium confidence

HTTP Request Configuration:

A screenshot of a computer

Description automatically generated

Email Configuration:

Here the highlighted parameters are dynamically sent from the data factory when we use Web activity.

Text

Description automatically generated

1. Now in Azure Data Factory following pipeline is created. To execute the main pipeline and send mail based on success or failure.

Graphical user interface, text, application, chat or text message

Description automatically generated

In Web activity we give the URL of our Logic App, method as POST and build the body of the activity dynamically as below:

Graphical user interface, text, application, email

Description automatically generated

Highlighted parts will be taken dynamically from the Data Factory.

Graphical user interface, text, application, email

Description automatically generated

1. Email when pipeline is successful:

Graphical user interface, text, application

Description automatically generated

Email when pipeline fails:

Graphical user interface, text, application

Description automatically generated

**Milestone 9:**

**Goal**: Design and develop a **SQL Metastore**

You need to **perform** the following **steps**:

Design and develop a SQL Metastore to capture and manage:

* Job and Job Run details
* Table Details (which needs to be ingested in the lake)
* Name
* Acquisition strategy (incremental or full)
* Watermark to pull incremental data
* PK column
* merge strategy (truncate & load | Append | PK based upsert.

1. **Creating a stored procedure in the SQL DB**.
2. Create an audit table to capture the pipeline run details with following details.

Graphical user interface, text, application

Description automatically generated

1. Graphical user interface, text, application

   Description automatically generatedFirst create a stored procedure in SQL DB which will insert rows in this table from the data factory.

1. **Updating Audit Table**

For this pipeline I have captured the audits in the first copy data activity, that is, when data is copied from the SQL DB to the landing zone. For this following set up was configured in the pipeline:

Graphical user interface, text, application, chat or text message

Description automatically generated

1. Add a stored procedure activity in for each pipeline and select linked service to the DB where this stored procedure and Audit table is stored. Select the stored procedure name from the drop down.

Graphical user interface, text, application

Description automatically generated

1. Pass following parameters from the pipeline to the stored procedure which will be updated in the Audit table.

Some values are hard coded while others are dynamically updated from the system variables and activities.

Graphical user interface

Description automatically generated

1. When pipeline is run the metrics are stored in the SQL DB.

A screenshot of a computer

Description automatically generated