

Azure Data Factory DM Project

Date: 30.06.2024
Version: 1.00
Written By: Tomer

1. General

1.1 Project Goals

The purpose of this project is to provide a complete Data Mart on the cloud that updated with new data every 24 hours at 01:00:00 AM.

The project will be scalable in the cloud and cost effective.

1.2 Project Content

Data mart for product that customers ordered.

1. Data cleaning and preparation: prior to analysis, we will perform data cleaning and preparation to ensure their quality and consistency.
2. Main summary tables to be built for the Data Mart need:

Fact_Orders – Contain the order details of which product and quantity.

Dim_Product - Information about the products divided into categories and subcategories.

Dim_Customers - Information about the customers.

Source to target document in this link: S2T.

2. Work plan

Design the project, check if the design answers the requirements of the end users.

3. Technical requirements

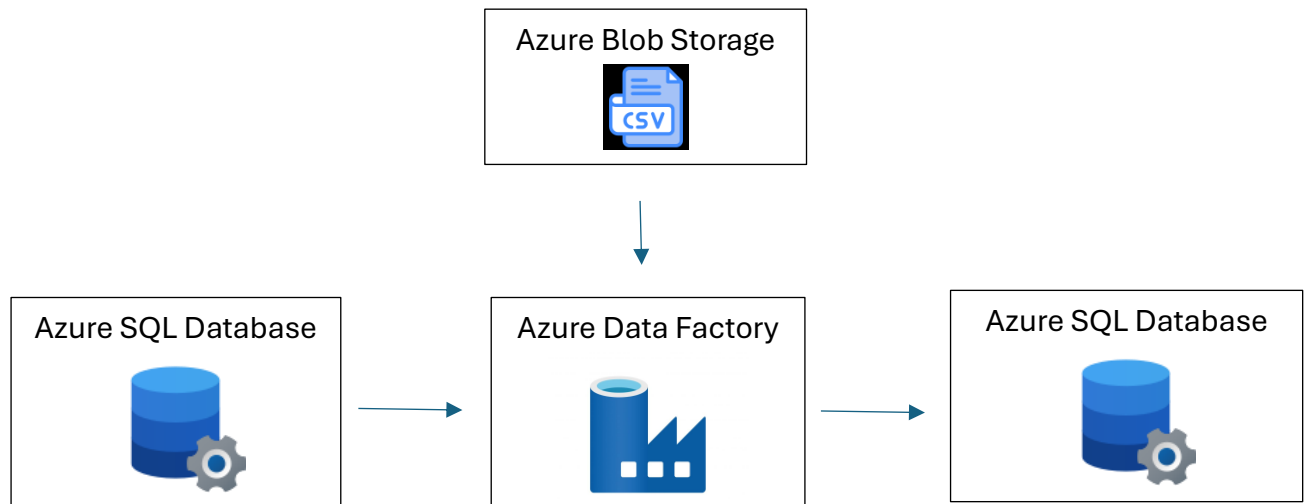
3.1 Prerequisites

- Blob storage container for CSV file that maps the Tables.
- Azure SQL Database with sample data – Source
- Azure SQL Database - Destination
- Azure Data Factory - ETL processes

- Azure Data Factory Triggers – run every 24 hours to add new data to Data Mart.

3.2 Solution Architecture

HLD:



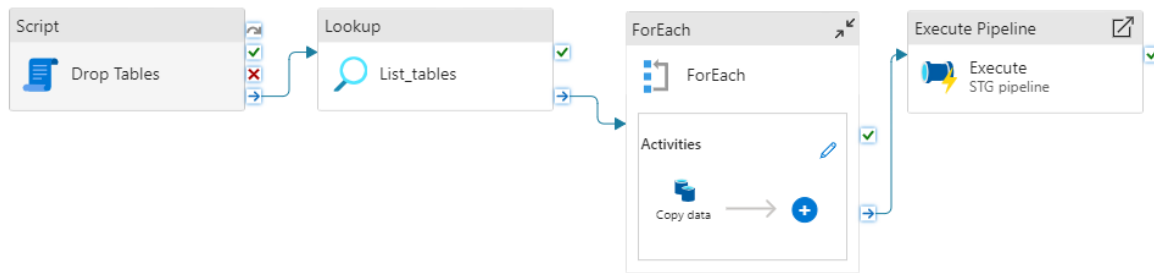
The source is Azure SQL Server Database contain sample data, the Azure Blob storage is a mapping file for tables and columns both go to Azure Data Factory for ETL processing and arrangement into Data Mart.

4. Functional Specification

4.1 ETL processes

1. MRR Pipeline:

Create MRR table in the destination database and copy the relevant tables and columns to the MRR tables.



Script drop tables – is designed to drop all the MRR tables in the destination database.

Lookup list tables – Get from Blob storage CSV file with names of the source and target tables and list of columns.

ForEach – loop over the CSV rows and populate the activity copy data
Copy Data:

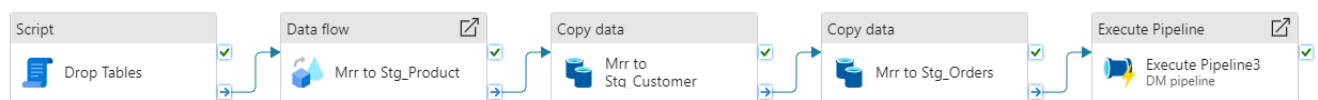
source – Contain SELECT query with dynamic variables to get the table with the desired columns.

Sink – Contain dynamic MRR table name to create.

Execute pipeline – execute the next pipeline STG.

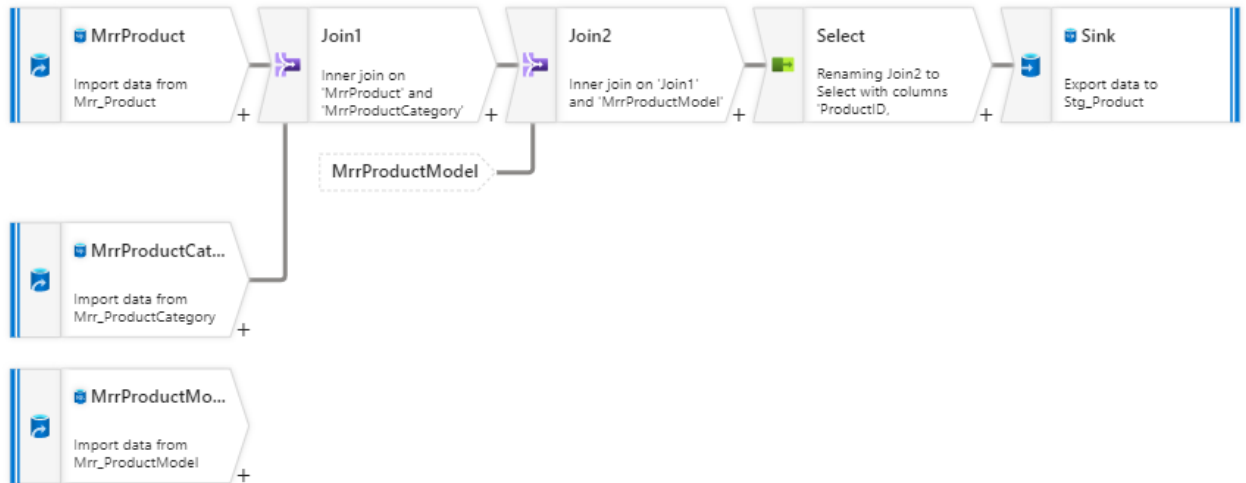
2. STG Pipeline

Join the MRR tables to STG tables



Script drop tables – is designed to drop Stg_Customer, Stg_Orders and truncate Stg_Product in the destination database.

MRR to STG product data flow



Join1 – join Mrr_Product with Mrr_ProductCategory

Join2 – join Join1 with Mrr_ProductModel

Select – Choose the relevant columns to pass to the Sink

Sink – Add the data to the Stg_Product table

MRR to STG Customer copy data

Source – Query that join tables: Mrr_Customer, Mrr_CustomerAddress, Mrr_Address.

Sink – Create Stg_Customer and populate with the copied data

MRR to STG Orders copy data

Source – Query that join tables: Mrr_Customer, Mrr_CustomerAddress, Mrr_Address.

Sink – Create Stg_Customer and populate with the copied data

Execute pipeline – execute the next pipeline DM.

3. DM Pipeline

Transform the data before coping to DM tables



Script drop tables – is designed to drop all the DM tables in the destination database.

Dim_Customer data flow



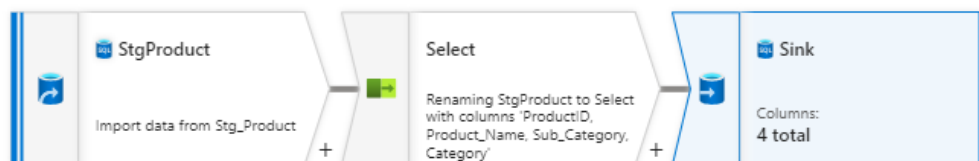
StgCustomer – the source dataset.

Derived Column – add new column FullName concat of FristName and LastName.

Select – Choose the relevant columns to pass to the Sink

Sink – Add the data to the Dim_Customer table

Dim_Product data flow



StgProduct – the source dataset.

Select – choose the relevant columns to pass to the sink.

Sink – Add the data to the Dim_Product.

Fact_Orders data flow



StgOrder – the source dataset.

Derived Column – Total calculation $Qty * Price * (1 - Discount) * 1.17$

Select – choose the relevant columns to pass to the sink.

Sink – Add the data to the Dim_Product.

4. Trigger

Trigger schedule to execute all the pipeline every day at 1 AM,
It starts date is 30/06/2024 and there is no end date.

5. Monitor Dashboard

Show if the trigger, pipeline and activity complete successfully.
There is option to create alert if not succussed to send email on failure.

