Sri Lanka Institute of Information Technology

Data Wearhouse and Business Intelligence

Assignment 01

Compiled by -

IT19955896 – T.M.S.N Tennakoon

Contents

1. Introduction to the project

In corporate environment data is an essential aspect. In modern days data has become one of the most valuable assets. Since millions of data are generated each day, it has become a challenge to handle enormous amount of data and obtain fruitful insights from them. The Datawarehouse concept is crucial for processing enormous amounts of corporate data. This article discusses the architecture, implementation, and ETL procedure of the “Instacart dataset”. This main objective of this project is to develop and deliver a data warehouse for business usage. This will aid in the storage and processing of company data.

2. Data Selection

2.1 Overview

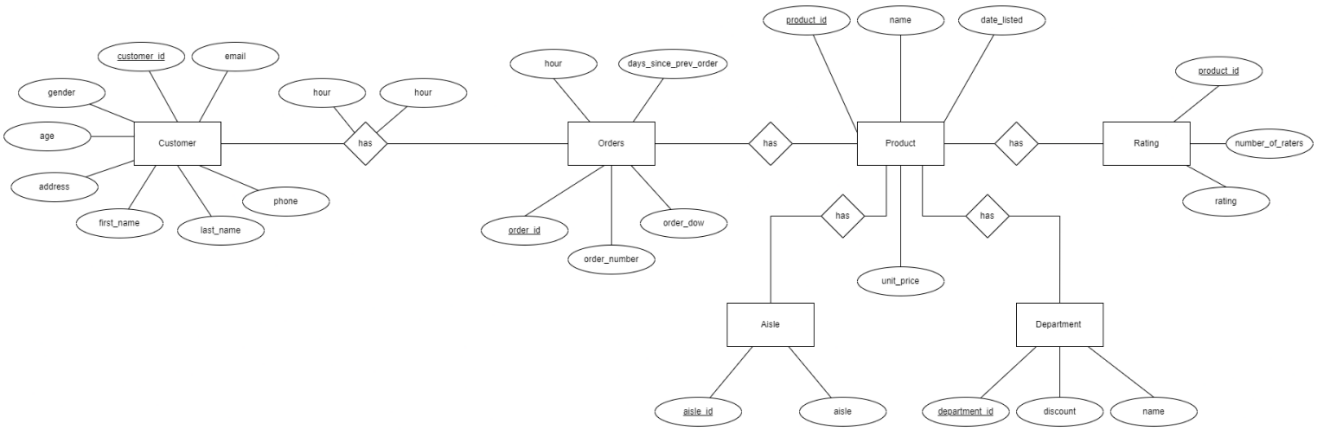
Data is the foundation of all decision-making techniques in the commercial world. In order to obtain reliable results, businesses must have sufficient data to process. The Instacart DataMart Analysis dataset found on Kaggle is used in this project. To adapt to the given context, I added additional datasets to increase the complexity of the Datawarehouse.

- Main Dataset – https://www.kaggle.com/c/instacart-market-basket-analysis/data
- Customer Data generated from – https://www.onlinedatagenerator.com/
- Ratings dataset was developed using Excel

2.2 Data Quality

Since the dataset represents actual transaction data from the Instacart web store in 2017, there were few null values in the chosen dataset. Furthermore, the dataset has sufficient data to be utilized in a BI solution. This dataset is appropriate for a BI implementation because of its high data quality and quantity. There were over 100,000 records in some data files. However, for this project only a part of the data set is utilized to improve the processing time.

The ER diagram for the specified dataset is shown below.



3. Data Preparation

Different types of data sources, including as CSV and Txt files, were utilized to extract data from the dataset. The Datawarehouse extraction procedure is complicated by the variety of data files.

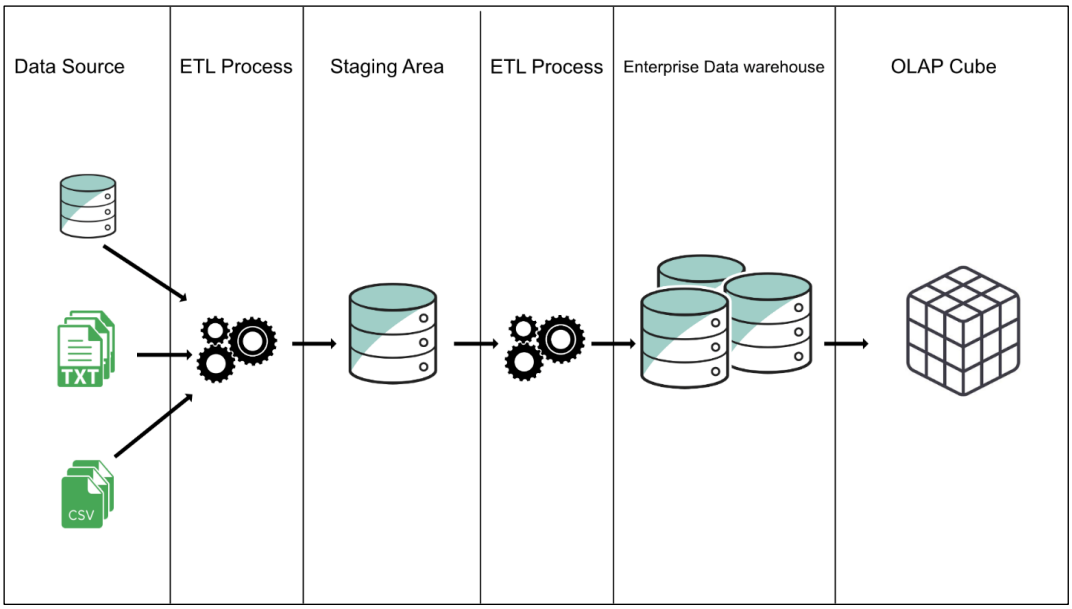
- **CSV files** – Aisle, Orders, Order details, Department
- **Txt files** - Ratings
- **Databases** – Customer
- **Excel file** – Product

Source Type	Table Name	Colum Name	Data type	Description
CSV fille	dbo.Aisle	aisle_id aisle	numeric(18, 0) nvarchar(50)	Unique ID Aisle name
CSV fille	dbo.Department	department_id department discount	numeric(18, 0) nvarchar(50) int	Unique ID Department name Discount by department
CSV fille	dbo.Order	order_id user_id order_number order_dow order_hour_of_day days_since_prior_order order_date	numeric(18, 0) numeric(18, 0) numeric(18, 0) numeric(18, 0) numeric(18, 0) numeric(18, 0) datetime	Unique ID User ID Number assigned to order Day of the week Hour of the day Days since last order Orde date
CSV fille	dbo.Order_details	order_id product_id add_to_cart_order reordered	numeric(18, 0) numeric(18, 0) int int	Unique Order ID Product ID Add to cart or not Reordered or not
TXT file	dbo.Rating	product_id num_of_raters rating	numeric(18, 0) numeric(18, 0) numeric(18, 0)	Unique ID Rater count Rating

Database	dbo.Customer	customer_id first_name last_name phone_number email gender age street_address city country	numeric(18, 0) nvarchar(500) nvarchar(500) nvarchar(500) nvarchar(500) nvarchar(500) int nvarchar(500) nvarchar(500)	Unique ID First name Last name Phone Email Gender Age Street address City Country
Excel file	dbo.Product	product_id product_name unit_price aisle_id department_id date_listed	numeric(18, 0) nvarchar(255) float numeric(18, 0) numeric(18, 0) datetime	Unique ID Product name Price of the product Aisle Department Listed date

4. Solution Architecture

Below diagram shows the Datawarehouse architecture.



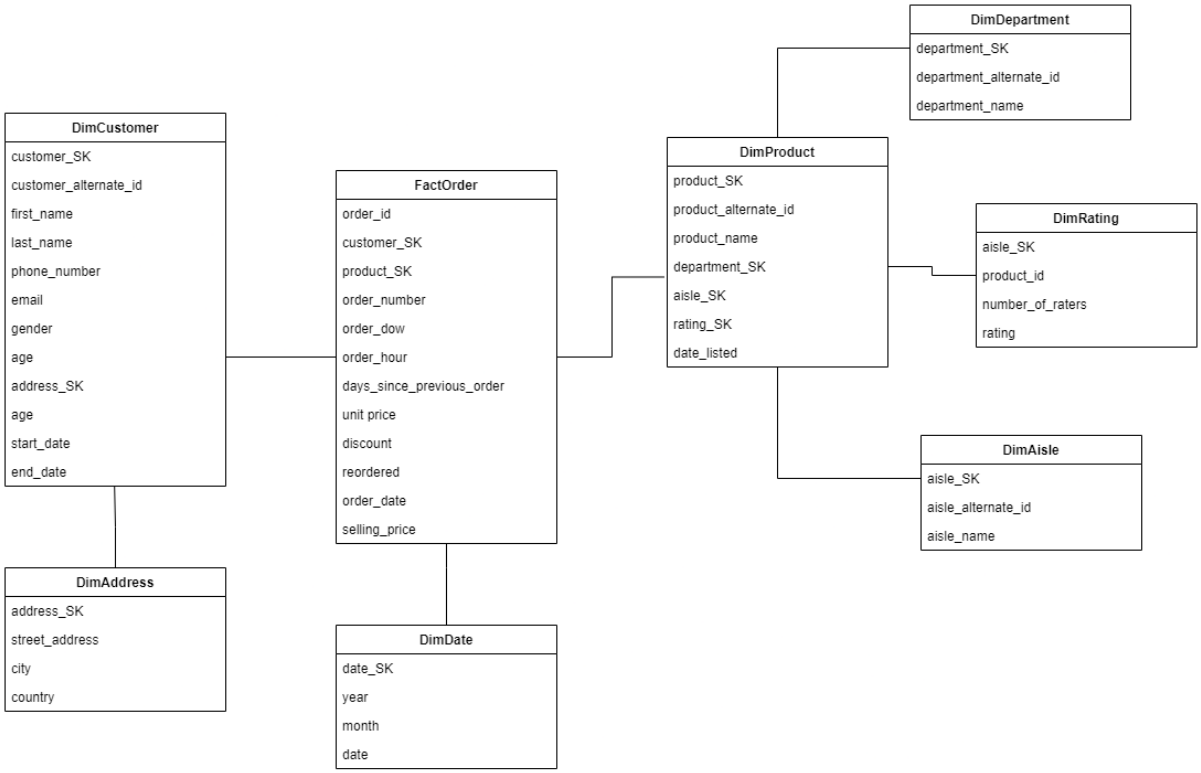
5. Design and Development

5.1 Dimensional Schema

Initially facts and dimensions were identified before implementing the warehouse. Snowflake schema was used to avoid data redundancy. The order table was taken as the fact table, which contains data from retail transactions.

After identifying the facts and dimensions there are 8 tables in the data warehouse. They are,

- Order – Fact table
- Product – Dimension table
- Department - Dimension table
- Aisle - Dimension table
- Rating - Dimension table
- Customer - Dimension table
- Date - Dimension table
- Address - Dimension table



5.2 Tools used

The main IDE in this solution is Visual Studio 2017, the main database server is SQL Server, and the server administration tool is SQL Server Management Studio.

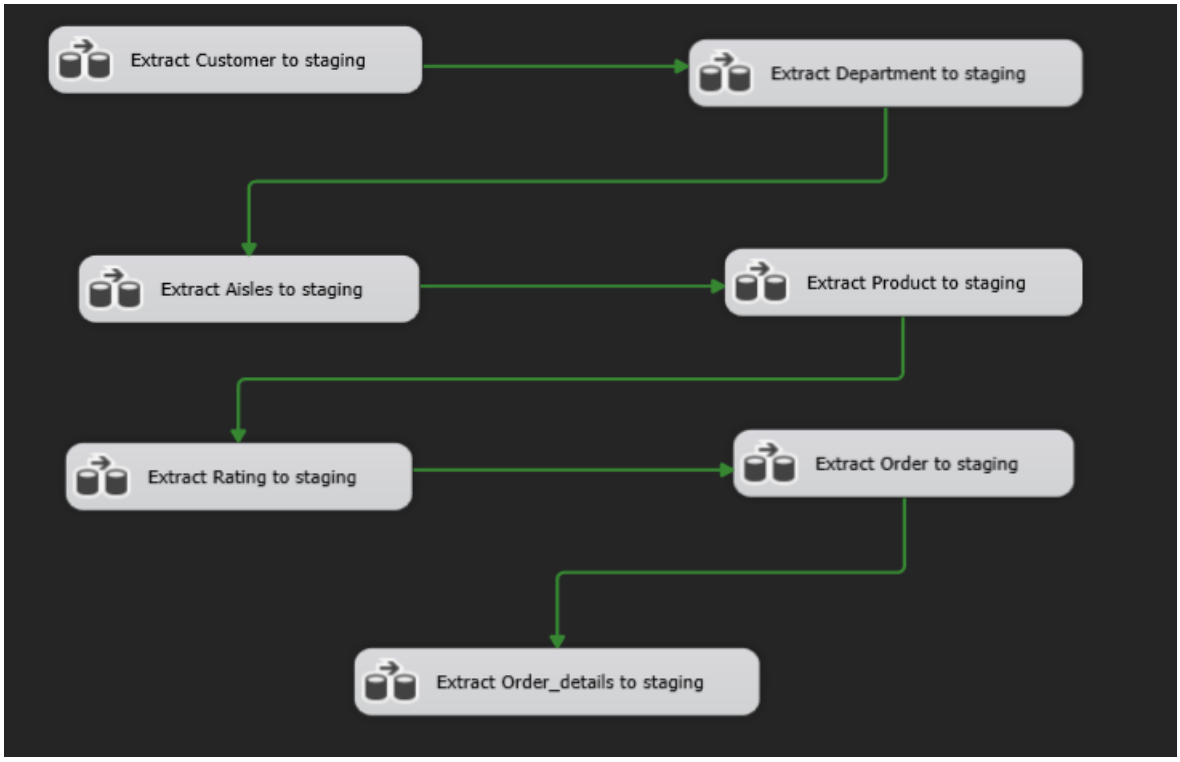
5.3 Assumptions

- Customer dimension and Rating dimension are slowly changing dimensions and they keep history records.
- Product names can't be updated.
- Address dimension contains all possible addresses.
- Extended string data types were used to suite future data.

6. ETL Development

6.1 Extraction

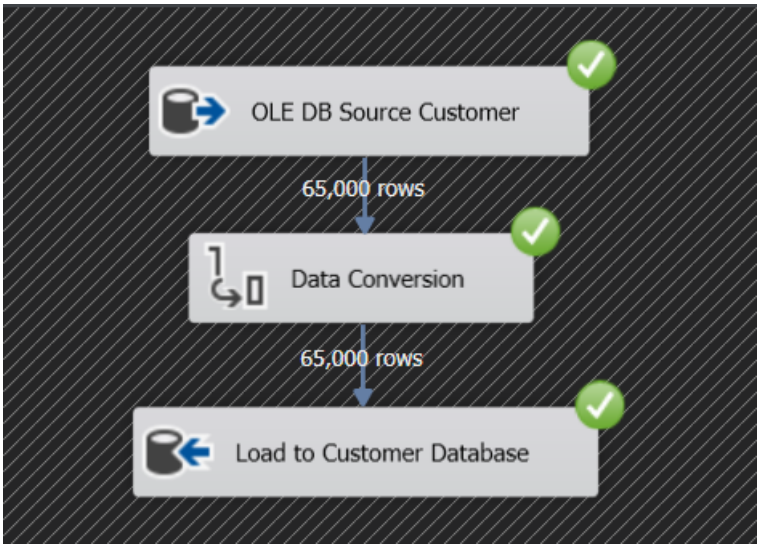
Initially all the data was extracted from all the data sources into separate staging tables. SQL server Integration service was sued for this process. Following is the image of the control flow designed in the SSIS to extract data into staging tables.



SQL Task component was used to truncate each staging table data to prevent data duplication when new data is inserted.

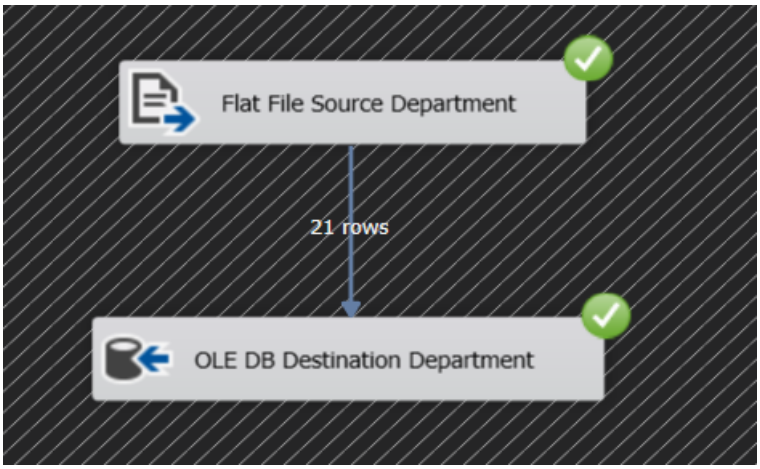
6.1.1 Data flow of extracting Customer Data to staging

Customer table in the Source database is taken as the data source and data conversion is used to convert data to match that of the staging tables.



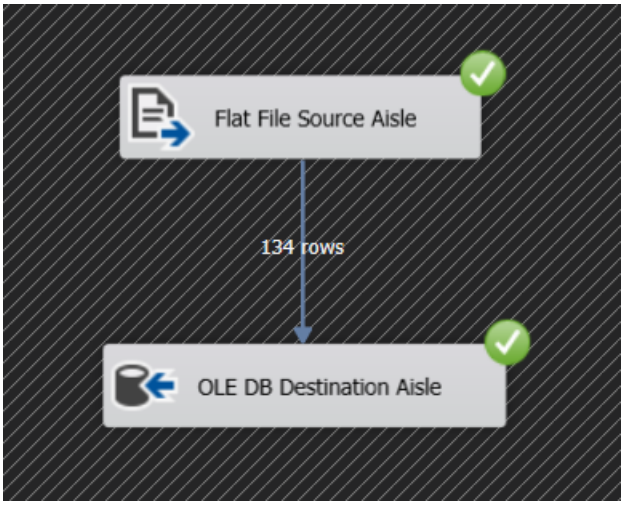
6.1.2 Data flow of extracting Department Data to staging

Data from Department CSV file was directly extracted into staging table.



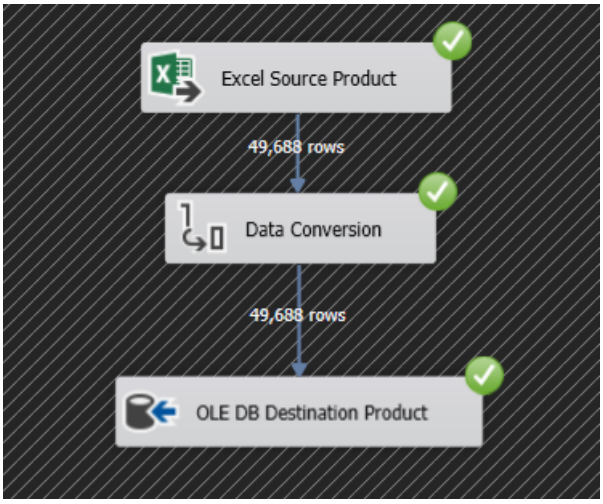
6.1.3 Data flow of extracting Aisle Data to staging

Data from Aisle CSV file was directly extracted into staging table.



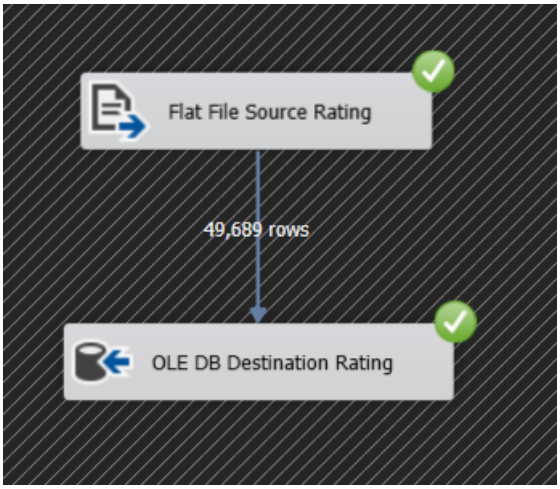
6.1.4 Data flow of extracting Product Data to staging

Data from Product excel file was converted to suitable data types before loading into the staging table.



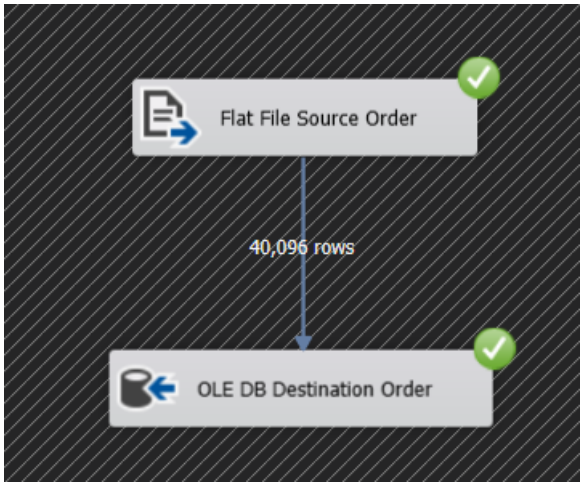
6.1.4 Data flow of extracting Rating Data to staging

Data from Rating Txt file was directly extracted into staging table.



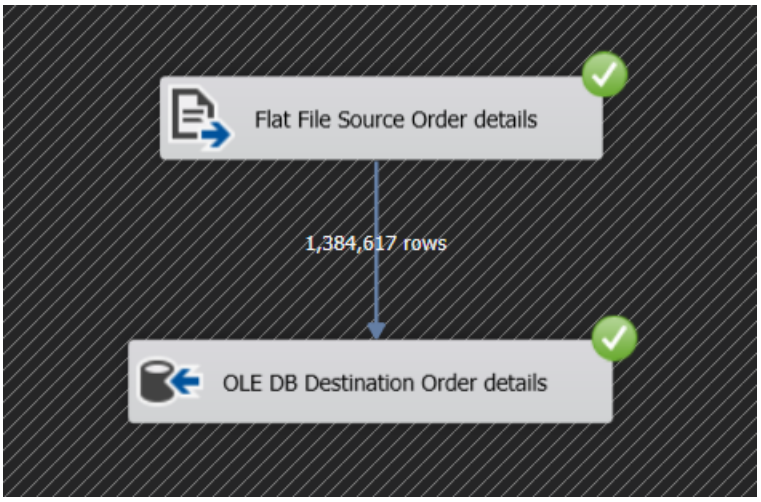
6.1.5 Data flow of extracting Order Data to staging

Data from Order CSV file was directly extracted into staging table.



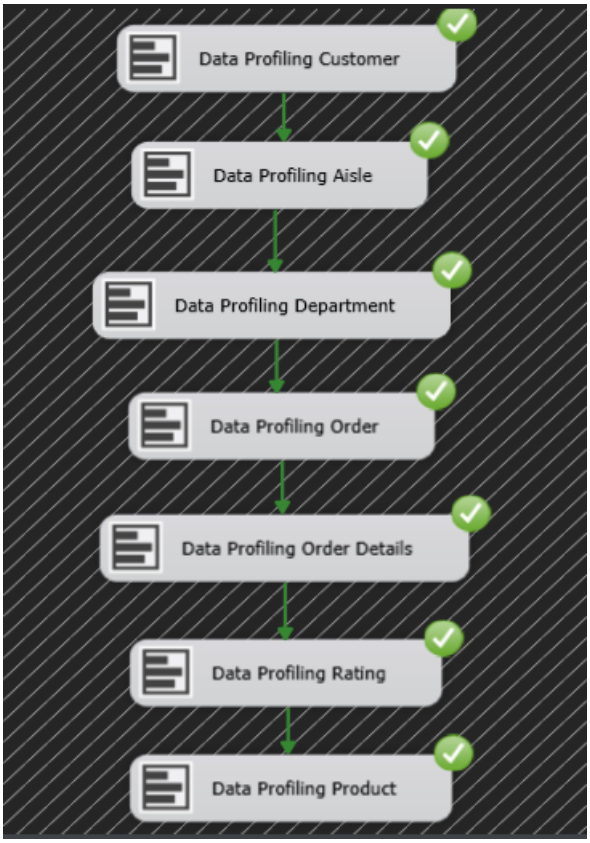
6.1.6 Data flow of extracting Order details Data to staging

Data from Order details CSV file was directly extracted into staging table.



6.1.7 Data Profiling

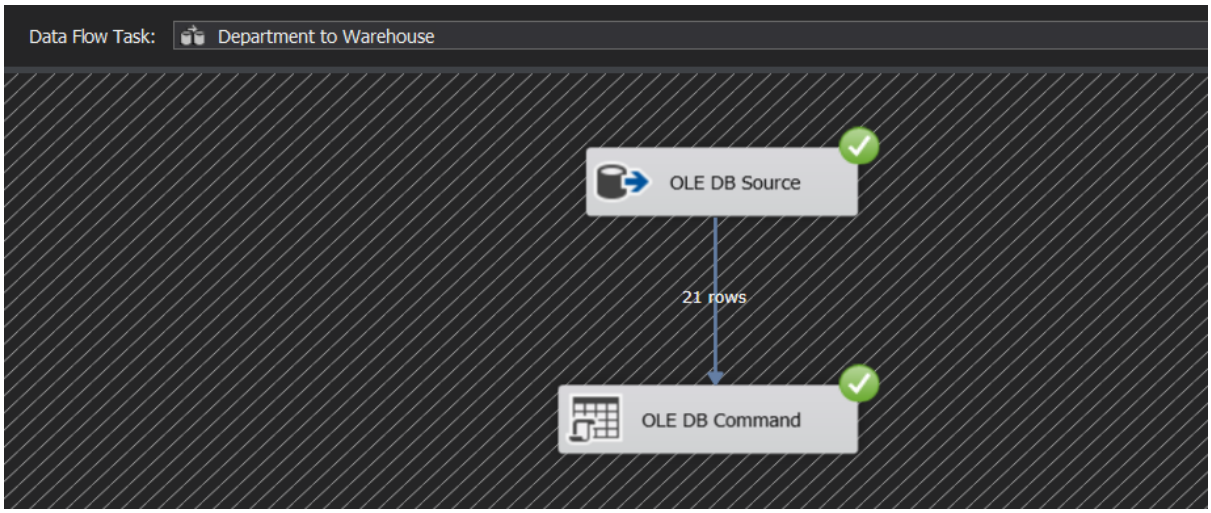
Data profiling was done to obtain an overview of the data. Data types, count and null values were identified from this process.



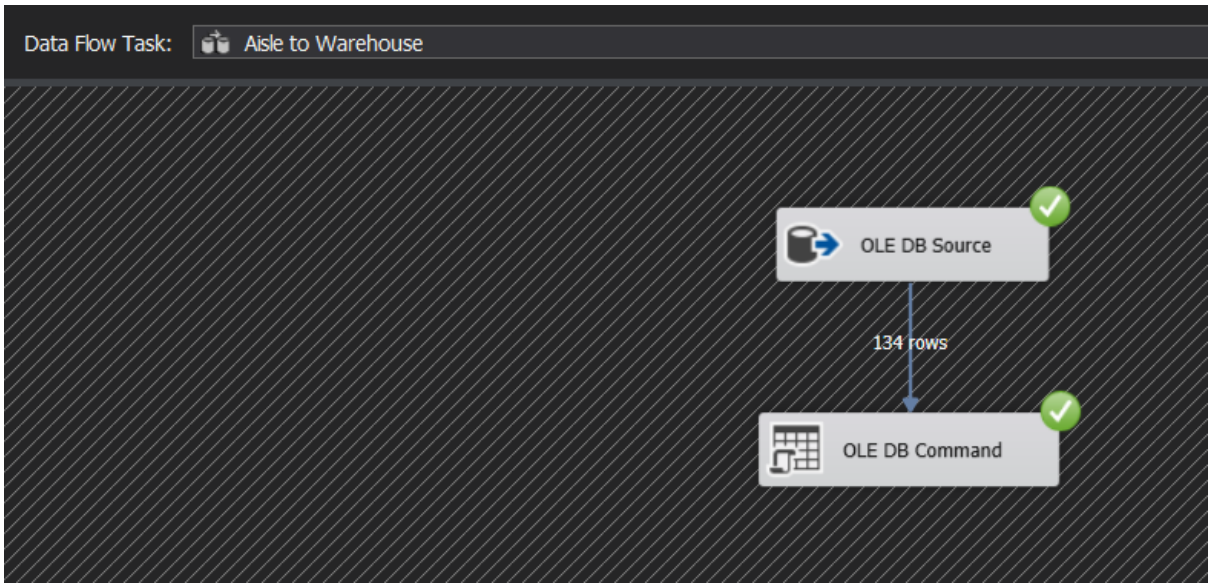
6.2 Transform and load

The data in the staging tables are then transformed and loaded into the warehouse. Mainly 6-dimensional table and a fact table was created using the snowflake schema. Initially the tables were created in the database and stored procedures were used to prevent the data redundancy. Date dimension was populated using a SQL procedure.

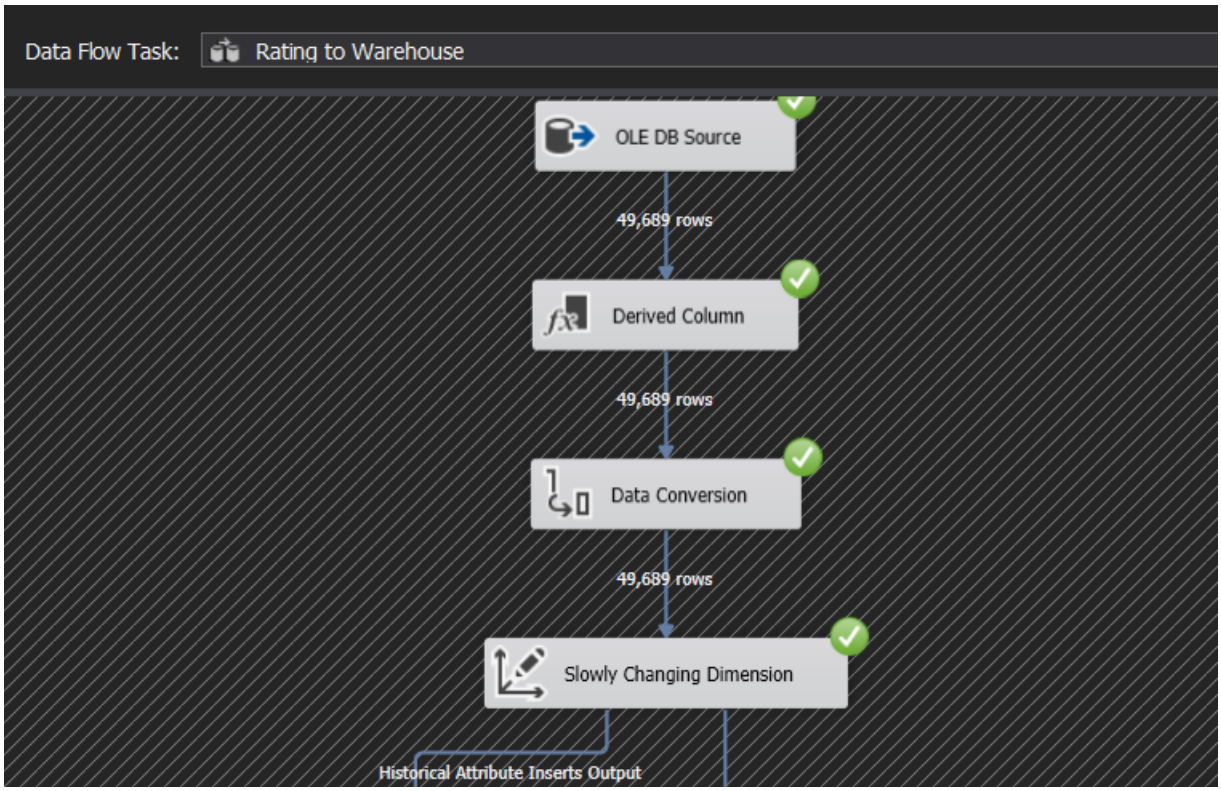
6.2.1 Transform and load into Department Dimension

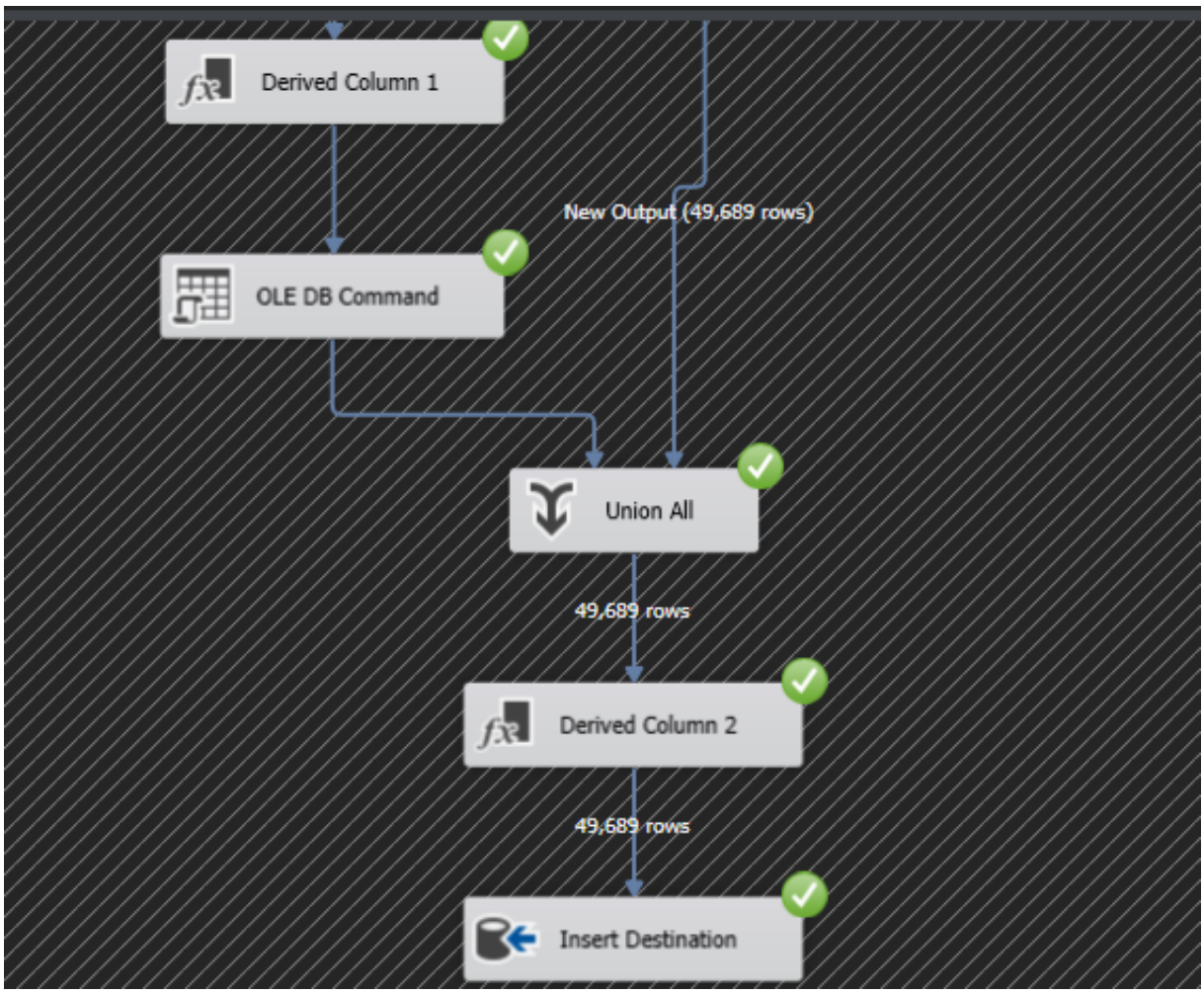


6.2.2 Transform and load into Aisle Dimension



6.2.3 Transform and load into Rating Dimension

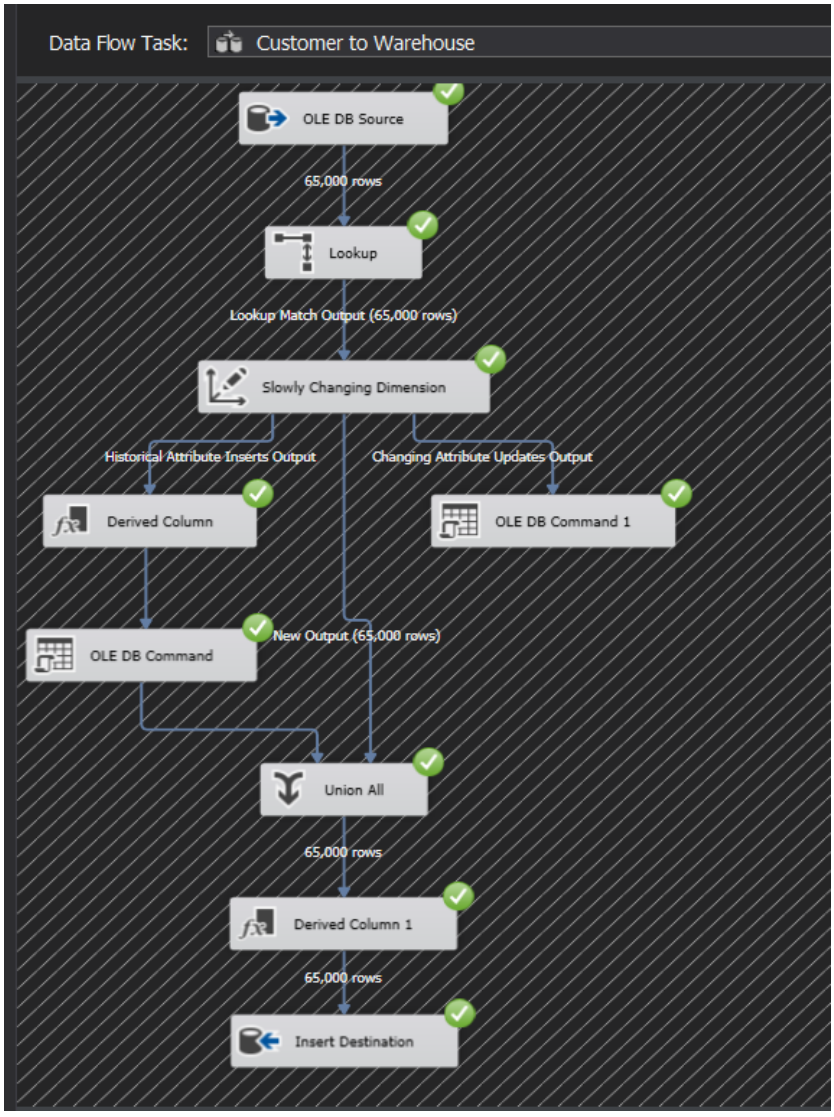




6.2.4 Transform and load into Address Dimension

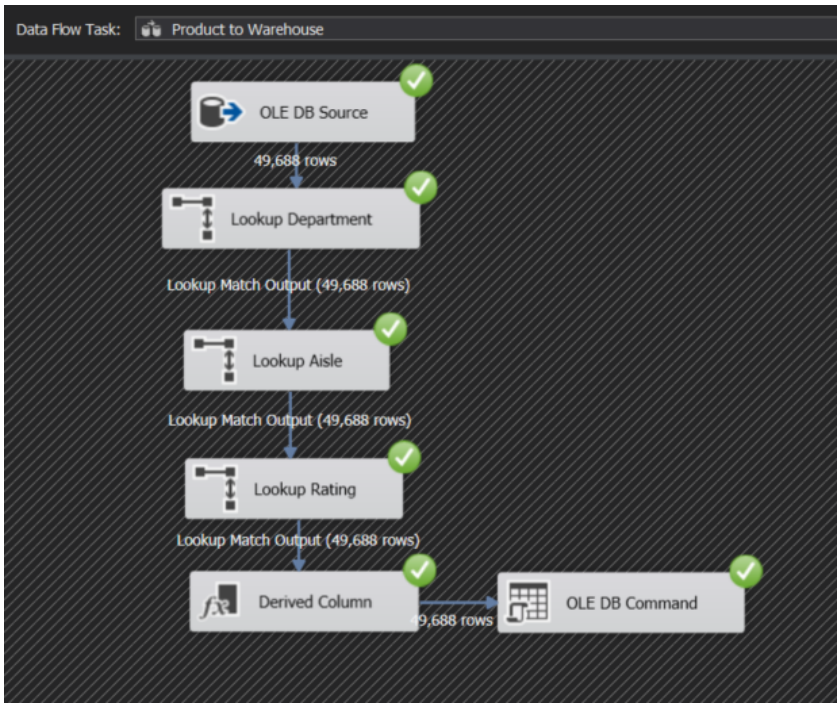
Address data from the customer table in the staging area was loaded into a separate dimension.

6.2.5 Transform and load into Customer Dimension



6.2.6 Transform and load into Product Dimension

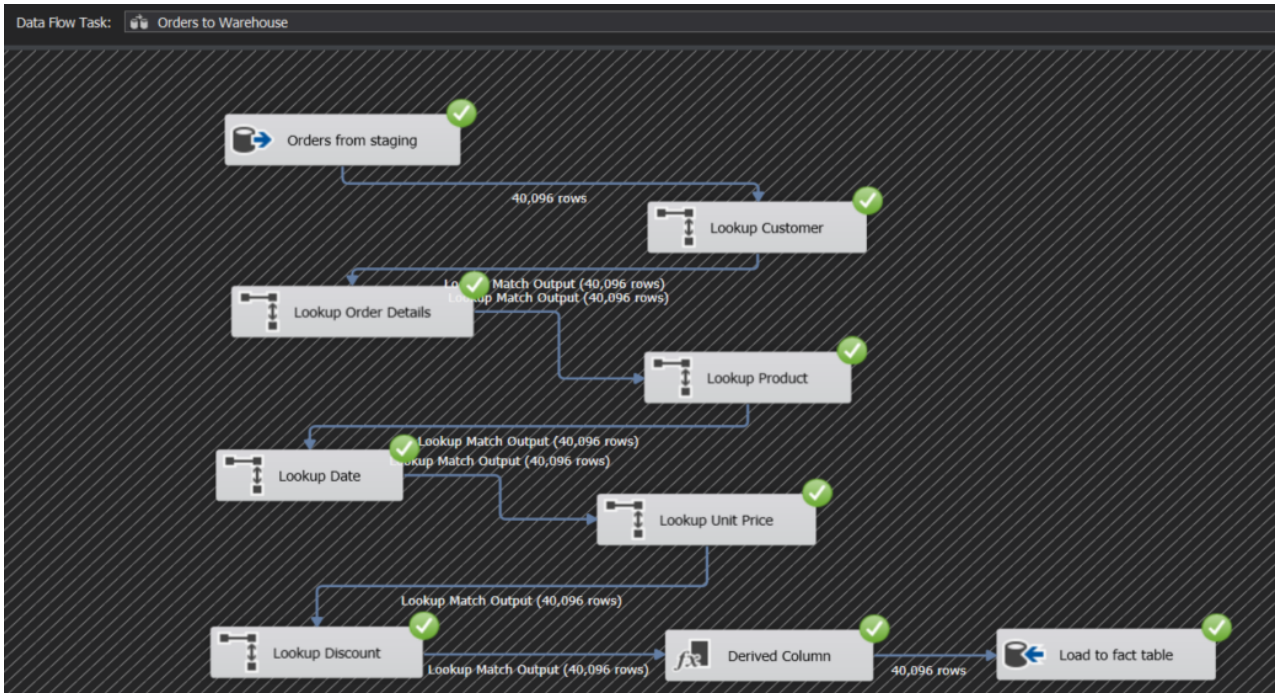
Product name contained null values; they were replaced with “NA”. Surrogate keys from Aisle, Department and Rating tables were inserted into the Product dimension.



6.2.7 Transform and load into Order fact table

Surrogate keys from the Customer table and Product table were inserted into the FactOrder table. Discount from the Department table and Unit price from the Product table were also inserted into this table. Selling price of the product was calculated using the following formula,

*Selling_price = unit_price – (unit price * (discount/100))*



7. Accumulating fact table

Accumulating fact table are used to summarizes the measurement events occurring at predictable steps between the beginning and the end of a process. DATEDIFF SQL function was performed on accm_txn_complete_time coming from the csv file and accm_txn_complete_time in the fact table to obtain the date difference in hours.