Sri Lanka Institute of Information Technology

Data Warehouse and Business Intelligence – Assignment 1 Submission

Name: Subachandran Andrew Asher

Reg-No: IT20220624

Batch: Year 03 Semester 01

Contents

1.DATA SET SELECTION

Data Set Name: Online Shopping

Provided by: kaggle.com

Source link: - https://www.kaggle.com/datasets/tanyadayanand/online-shopping?select=product.csv

About Dataset:

The selected data source is a collection of transactional data.

Online Shopping is the demanding business model widely used nowadays also it is a rapidly increasing business which creates new entrepreneurs also the customer side numbers also heavily increasing.

In this dataset about online shopping, it is mainly deals with the interaction with customer and product also it gives the data of orders and shipping and mainly it gives the market data where we can get all the information.

Dataset contains five csv files and one text file with information about customers, orders, Shipping, Product, Subproduct and Market. Modifications were done accordingly to the data set derived from the source This data set reflects combinations between customer transactions and product orders.

- Customer.csv: Customer data containing the details of customers their region, province, and the Customer segment according to their orders.
- Order.csv: The order details and the priority given to their orders.
- Product.csv: The details of products and the specific category of products
- Subproduct.csv: Gives the detailed explanation of products with the respective product ID
- Shipping.csv: Shipping Details of the products and the mode of shipping and Date of shipping
- Market.csv: The table which links all the table and gives the values of product margins, profits, Discounts and Sales.

2.PREPARATION OF DATA SOURCE

All the data sources are provided in csv format by the web site. In preparation of data sources, some changes have done for the source format (some tables were created, some columns were added and removed, some tables are merged) of the given files as converting into text files and importing csv files into a source database.

Final State of Preparation of the source data formats before Transforming data =>

- Shipping.txt
- OnlineShopping_SourceDB (Source Database) Tables: -
- dbo.Customer
- dbo.Order
- dbo.Product
- dbo.SubProduct
- dbo.Market

- To do Datawarehouse following changes are made to satisfy the complexity of the project and to show cast all the types
- Merged Order table and shipping table text file with the common attribute Order ID
- Subproduct is the sub table of Product table - it is used to create merge in data warehousing by merge subproduct and product dimension
- Some Alterations done in the columns because of some validation errors

ER-Diagram

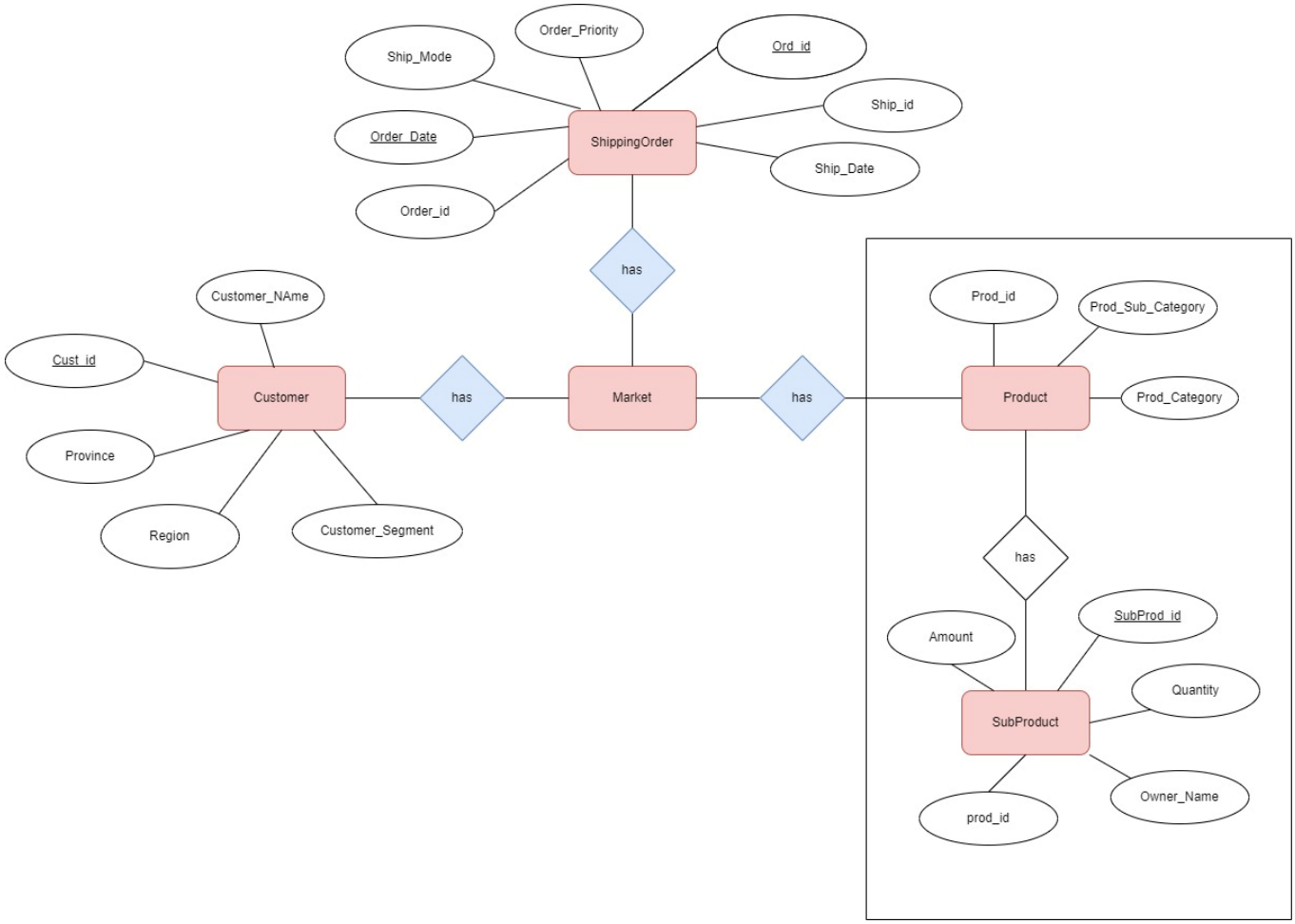


Figure 1: ER-Diagram

- The above diagram shows the connection between the entities in the data set.
- Assumptions:
    - The particular transaction includes list of Products ordered by customers.
    - One summary report (Market) summarizes many transactions takes place.
    - There can be many product data sets in a single summary report.

Description of the data set

<table>
<tr><th>Source Type</th><th>Table Name</th><th>Include</th></tr>
<tr><td>Shipping.txt</td><td>Shipping</td><td>
<table>
<tr><td>Column</td><td>Data type</td><td>Description</td></tr>
<tr><td>Order_ID</td><td>int</td><td>Ord Key</td></tr>
<tr><td>Ship_Mode</td><td>nvarchar(50)</td><td>Shipping Mode</td></tr>
<tr><td>Ship_Date</td><td>datetime</td><td>Shipping Date</td></tr>
<tr><td>Ship_id</td><td>int</td><td>Shipping Key</td></tr>
</table>
</td></tr>
<tr><td rowspan="3">OnlineShopping_SourceDB</td><td>Customer</td><td>
<table>
<tr><td>Column</td><td>Data type</td><td>Description</td></tr>
<tr><td>Customer_Name</td><td>nvarchar(50)</td><td>Customer Name</td></tr>
<tr><td>Province</td><td>nvarchar(50)</td><td>CustomerResidence</td></tr>
<tr><td>Region</td><td>nvarchar(50)</td><td>Customer Region</td></tr>
<tr><td>Customer_Segment</td><td>nvarchar(50)</td><td>Segment</td></tr>
<tr><td>Cust_id</td><td>int</td><td>Customer Key</td></tr>
</table>
</td></tr>
<tr><td>Product</td><td>
<table>
<tr><td>Column</td><td>Data type</td><td>Description</td></tr>
<tr><td>Prod_id</td><td>int</td><td>Prod key</td></tr>
<tr><td>Product_Category</td><td>nvarchar(50)</td><td>Product Category</td></tr>
<tr><td>Product_sub_category</td><td>nvarchar(50)</td><td>Product Sub Category</td></tr>
</table>
</td></tr>
<tr><td>SubProduct</td><td>
<table>
<tr><td>Column</td><td>Data type</td><td>Description</td></tr>
<tr><td>SubProd_id</td><td>int</td><td>SubProd Key</td></tr>
<tr><td>Prod_id</td><td>int</td><td>Product key</td></tr>
<tr><td>Quantity</td><td>nvarchar(50)</td><td>Quantity of product</td></tr>
<tr><td>Amount</td><td>float</td><td>Product Amount</td></tr>
<tr><td>Owner_Name</td><td>nvarchar(50)</td><td>Owner Name</td></tr>
</table>
</td></tr>
</table>

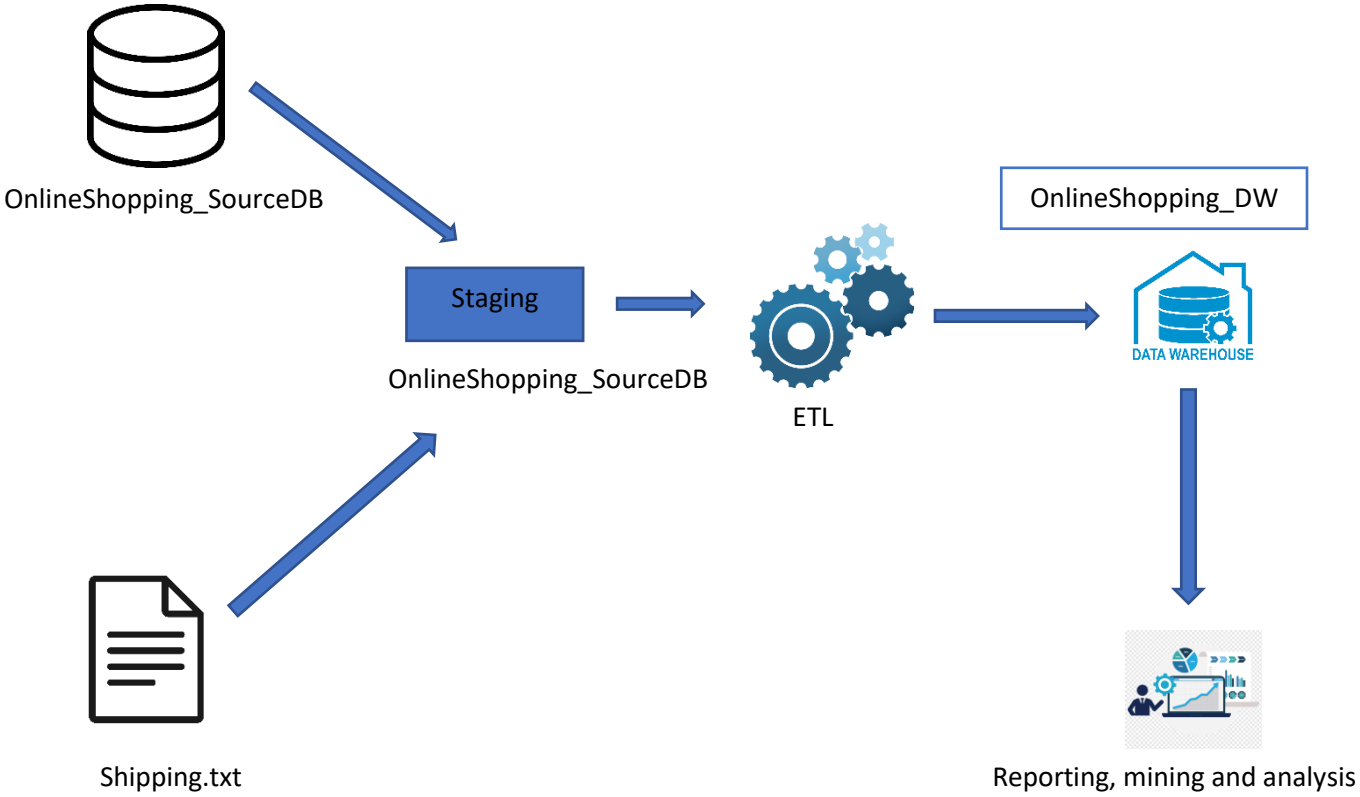
Market

Column	Data type	Description
Ord_id	int	Unique id for Order
Prod_id	int	Unique id for a Product
Cust_id	int	Unique id for an Customer
Sales	float	Sales Values
Discount	float	Discount amounts
Order_Quantity	int	Quantity of item bought
Shipping_Cost	float	Cost of Shipping
Profit	float	Profit Amounts
Product_Base_Margin	int	Base Margin of Product

Order

Column	Data type	Description
Ord_id	int	Unique id for order
Order_id	int	Order key
Order_Date	date	Date of the Order
Order_Priority	int	Booked order

3.SOLUTION ARCHITECTURE



As the figure 2 shows for the ETL processing, initially **Online_Shopping_SourceDB**: Source Database, **Shipping.txt**: Text file, used for the data extraction to the Staging Destination.

4.DATA WAREHOUSE DESIGN & DEVELOPMENT

i. Design

The Online Shopping (warehouse) is designed according to the given below snowflake schema with one fact table (dbo.FactMarket) and five dimension tables including Date dimension.

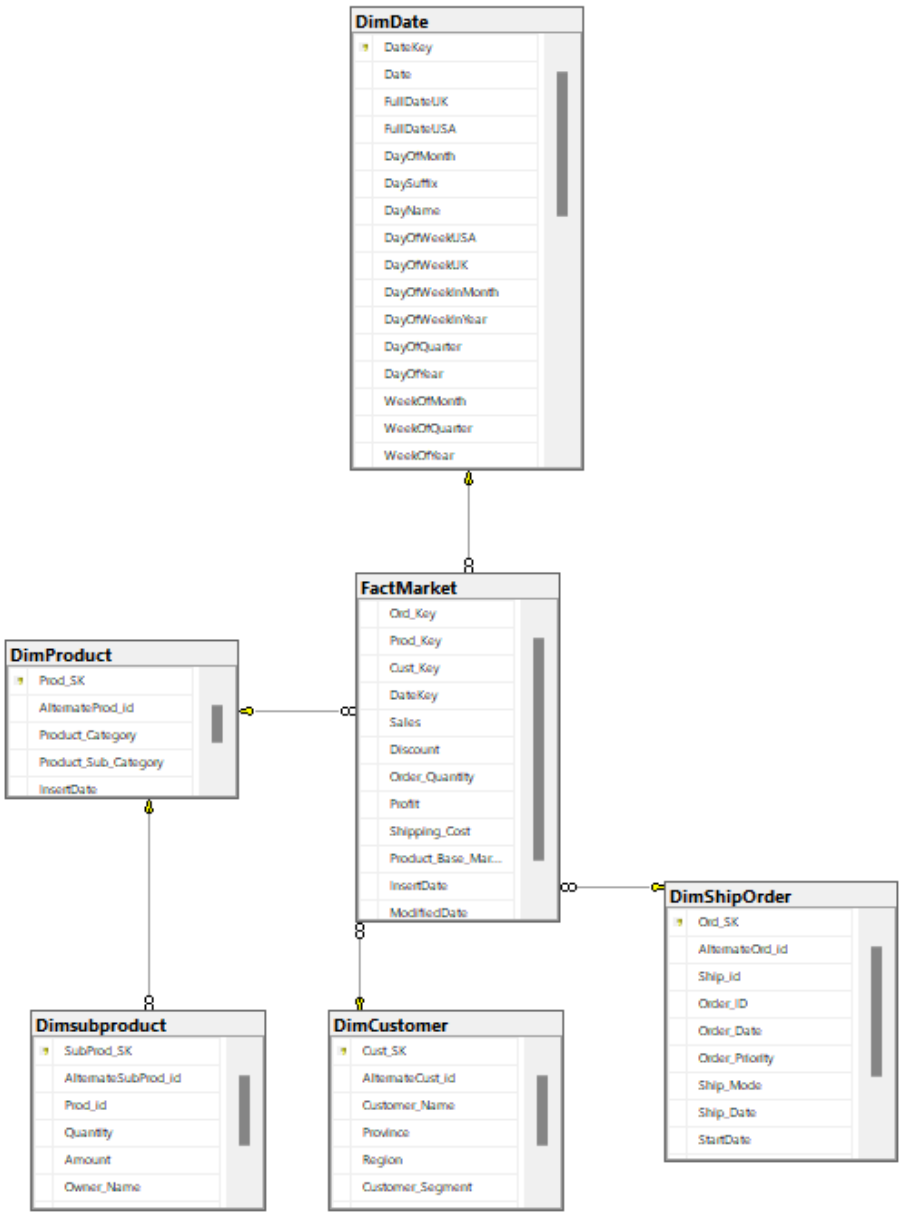


Figure 3: Snowflake schema

- Hierarchies

  - DimSubProducct is applied as a hierarchical dimension of DimProduct table.

ii. Assumptions

- dbo.DimDate is added to the Data Warehouse for better performance.
- dbo.Shipping is used in creating the fact table because it have links to all other dimension tables.

iii. Slowly changing dimensions

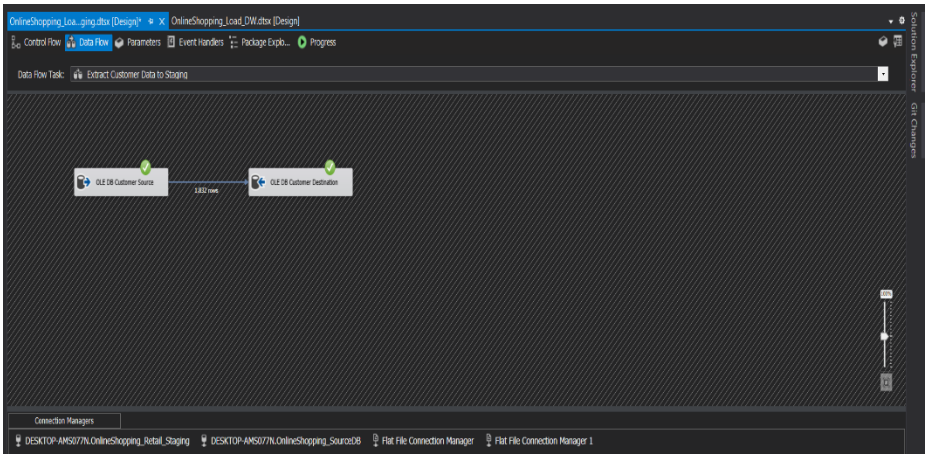
- ShipOrder Details were considered as a slowly changing dimension
  - Here first I merged two tables Shipping table and Order table after merging two tables added Slowly changing Dimensions

Dimension table	Attributes
DimShipOrder	Order_Date (Historical attribute) Order_ID (Fixed attribute) Order_Priority (Changing Attribute) Ship_Date (Historical Attribute) Ship_id (Fixed Attribute) Ship_Mode (Changing Attribute)

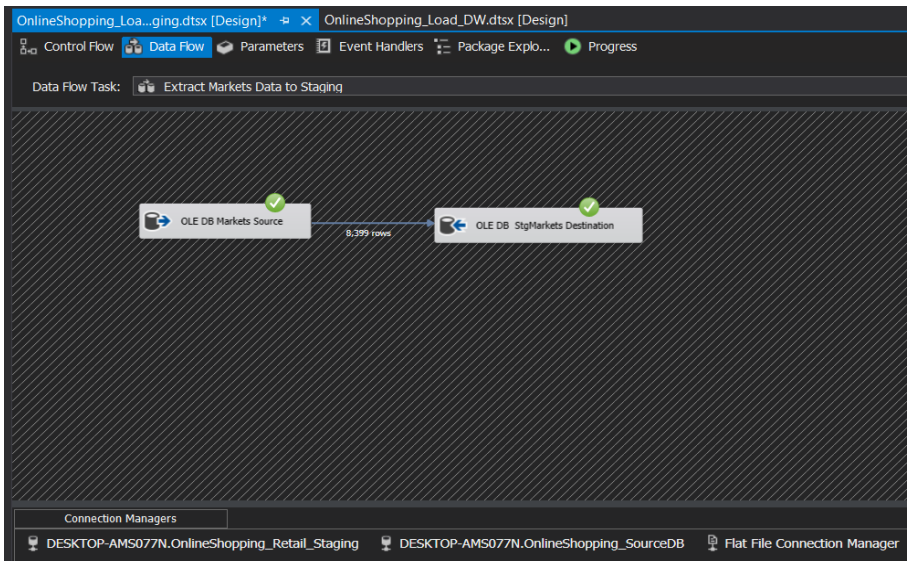
5.ETL DEVELOPMENT

i. Data Extraction & Load into Staging tables

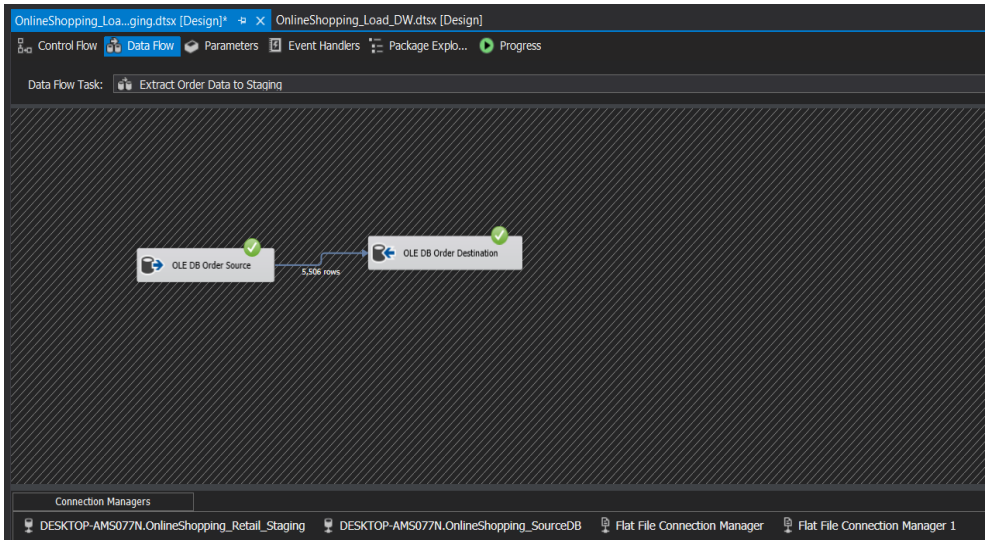
- Data Extraction is done by using the provided data sources mentioned above in Visual Studio 2019 (Data Tool) development environment. The text file and the source database were used here.
- Initially, **OLE DB SOURCE** (for source database) or **FLAT FILE SOURCE** (for flat files txt) is used to extract data for the Staging criteria. In this step developer can select the columns what would be included in the Staging from available data columns. As the next step of Staging, **OLE DB DESTINATION** has applied here to storing data in the Staging tables of **OnlineShopping_Retail_Staging**.



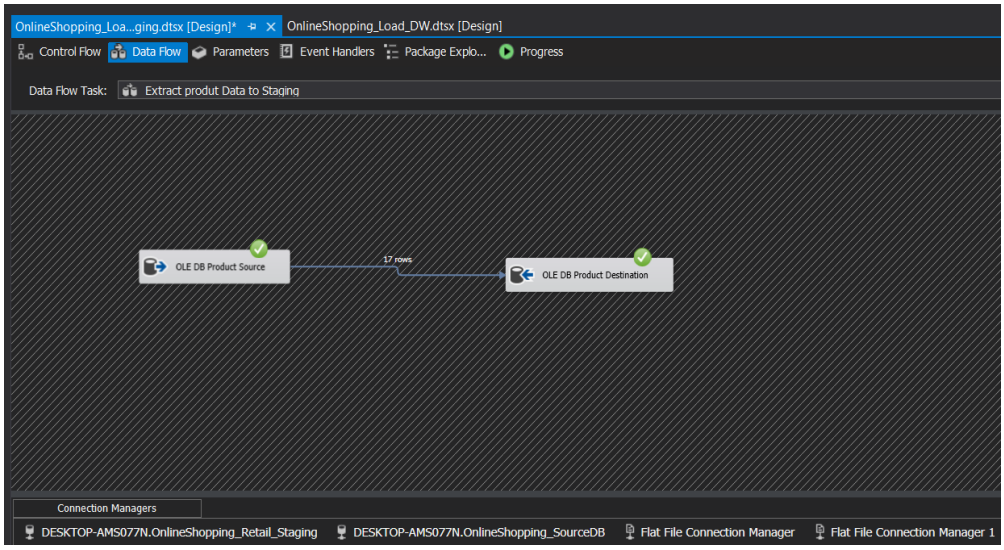
Customer Data is extracted from the Customer table in the source database and inserted to the Customer Staging Table



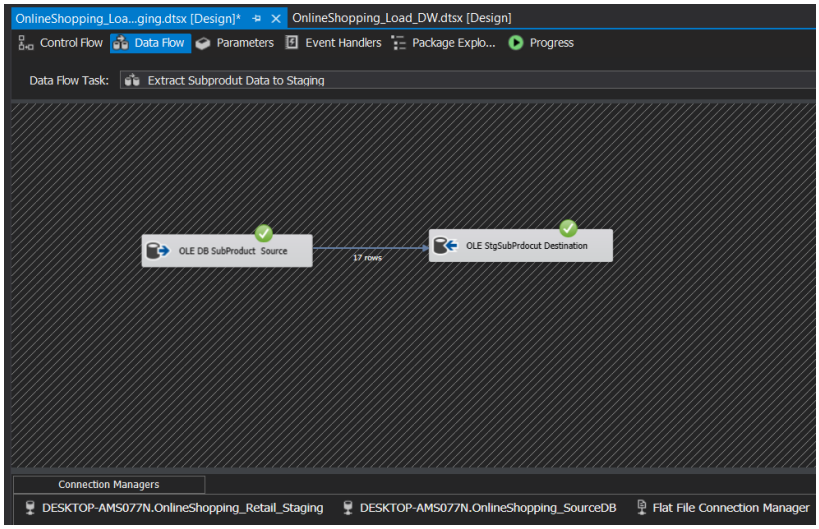
Market Data is extracted from the Market table in the source database and inserted to the Market Staging Table



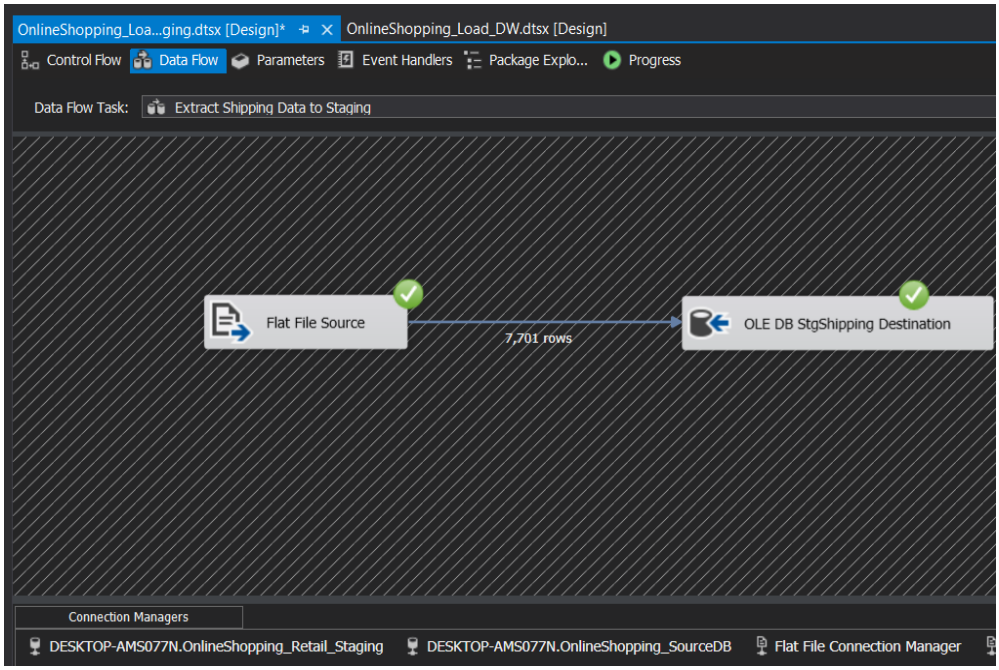
Order Data is extracted from the Order table in the source database and inserted to the Order Staging Table



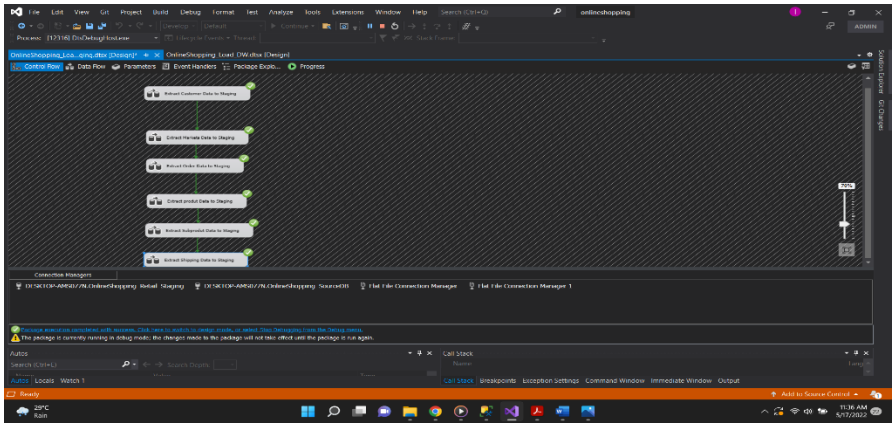
Extracted from the Product table in the source database and inserted to the Product Staging Table



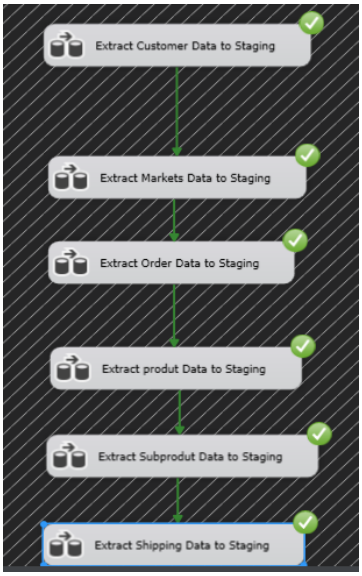
Extracted from the SubProduct table in the source database and inserted to the SubProduct Staging Table

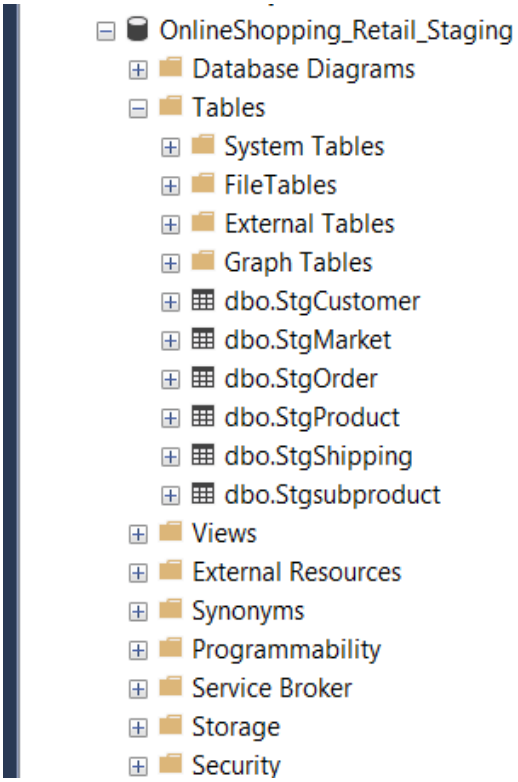


Shipping Data is Extracted from the Shipping table in the text file and inserted to the Shipping Staging Table

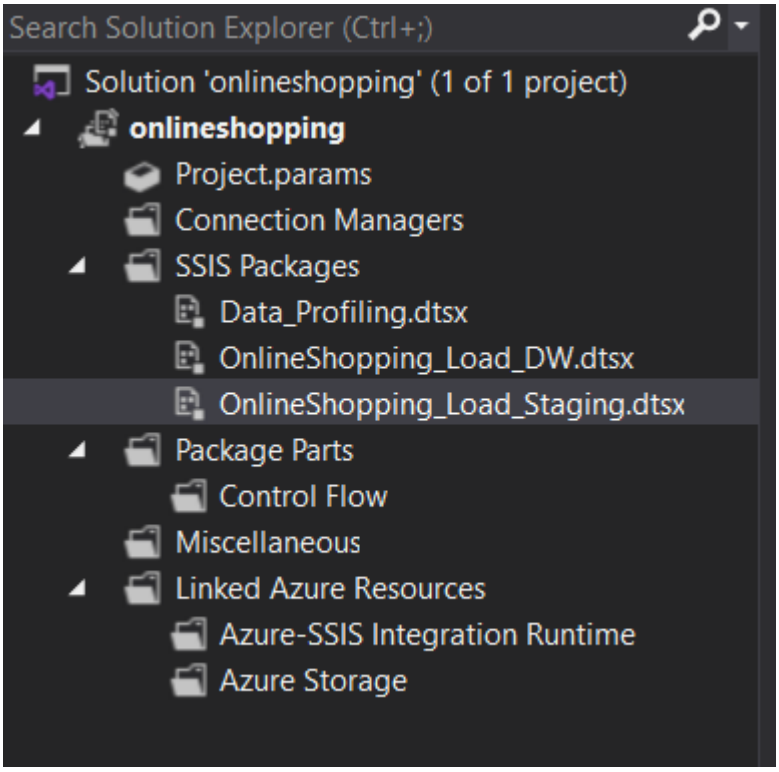


The Control Flow of 'Extract Data and Load into Staging ' Steps can be illustrate as the given figure.





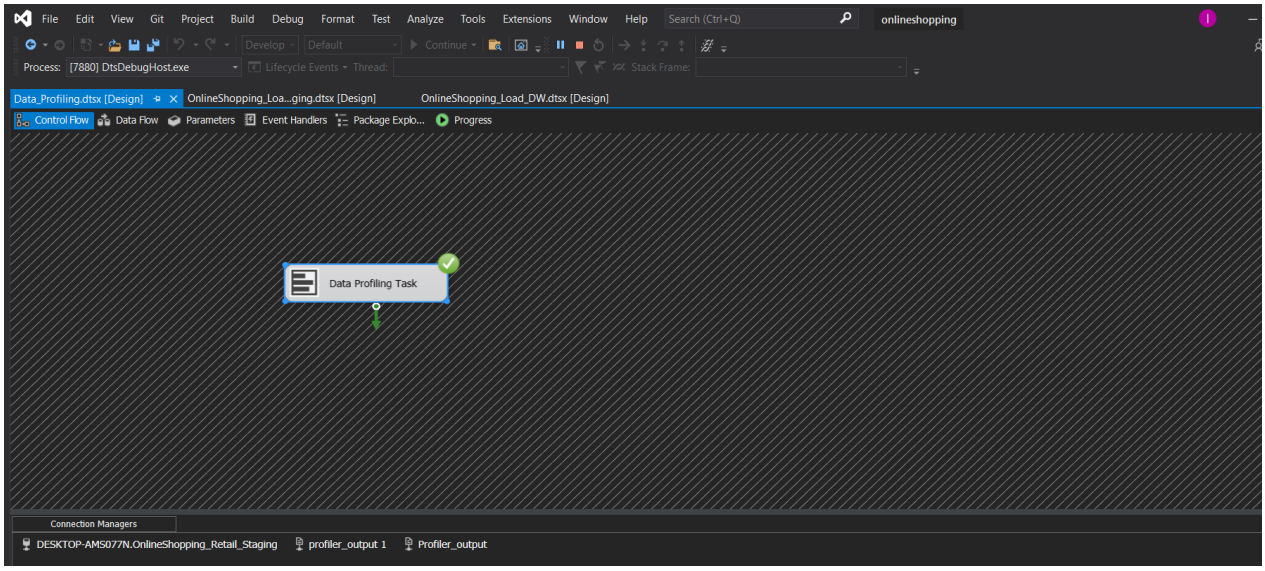
Staging Tables Created and Values inserted

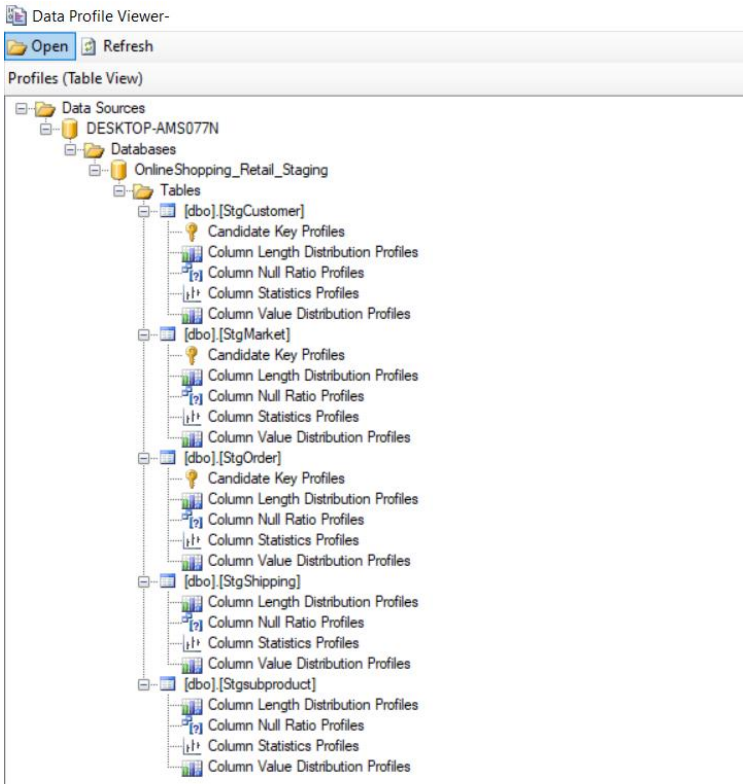


Online Shopping Solution Explorer image

ii. Data Profiling

Data Profiling provides the means of analyzing large amount of data using different kind of processes. In this step, null values, repeated values, and quality of the data is checked. Here I created Data Profiling Task with my all tables in Source DB.



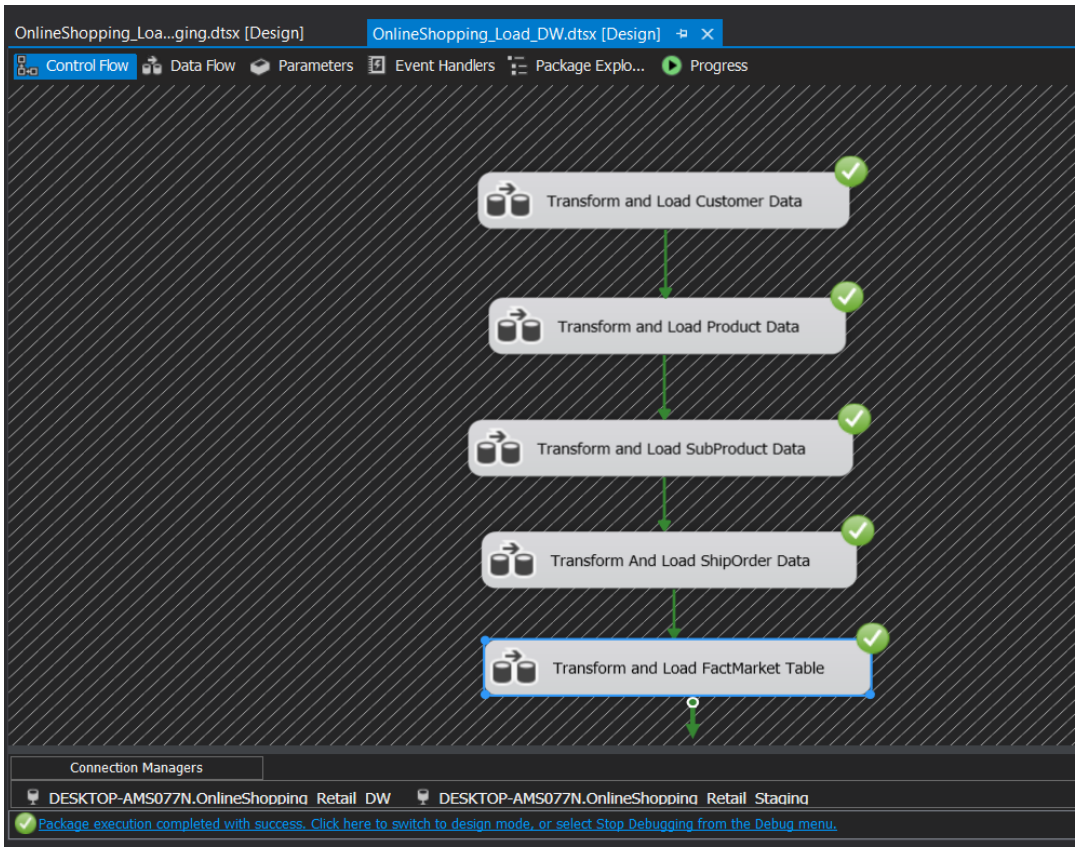


Data Profile Viewer of our Staging tables.it is easy to view all the necessary details of our staging table

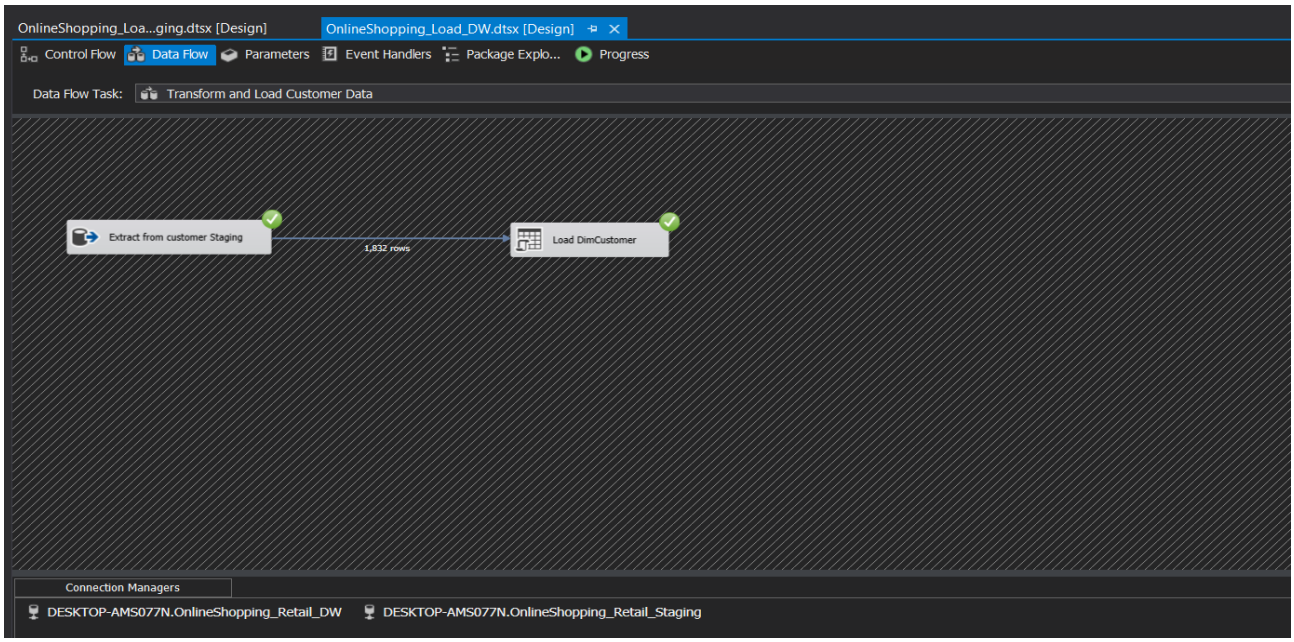
- Every staging table is profiled and saved in a selected location.
- As the figure shows, after the Staging step doing this task shows the things what the developer must consider about the data which are stored in staging table and the developer is able to identify the issues with staging data by data profiling (such as null values).
- The given figure illustrated the complete part of Data Profiling relevant to the Staging.

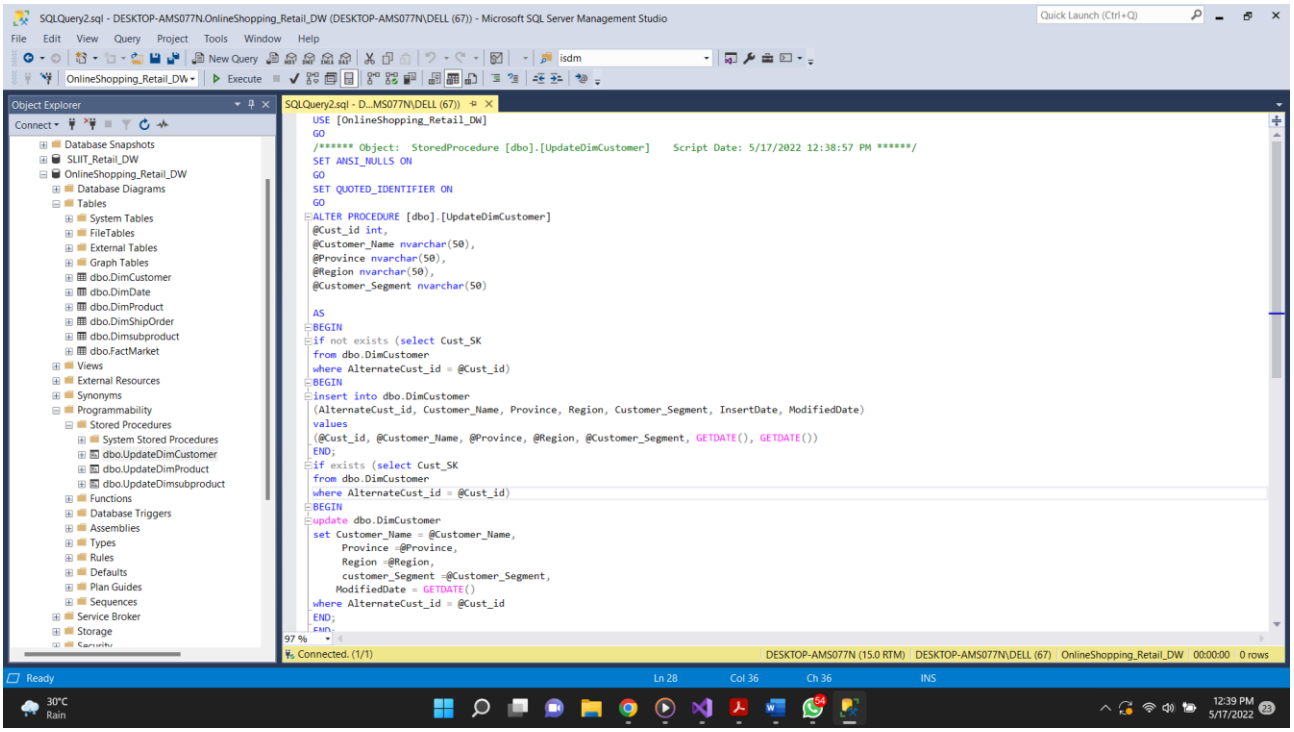
iii. Data Transformation and Loading

- Data Transformation is developed according to the dimensional modeling designed above.

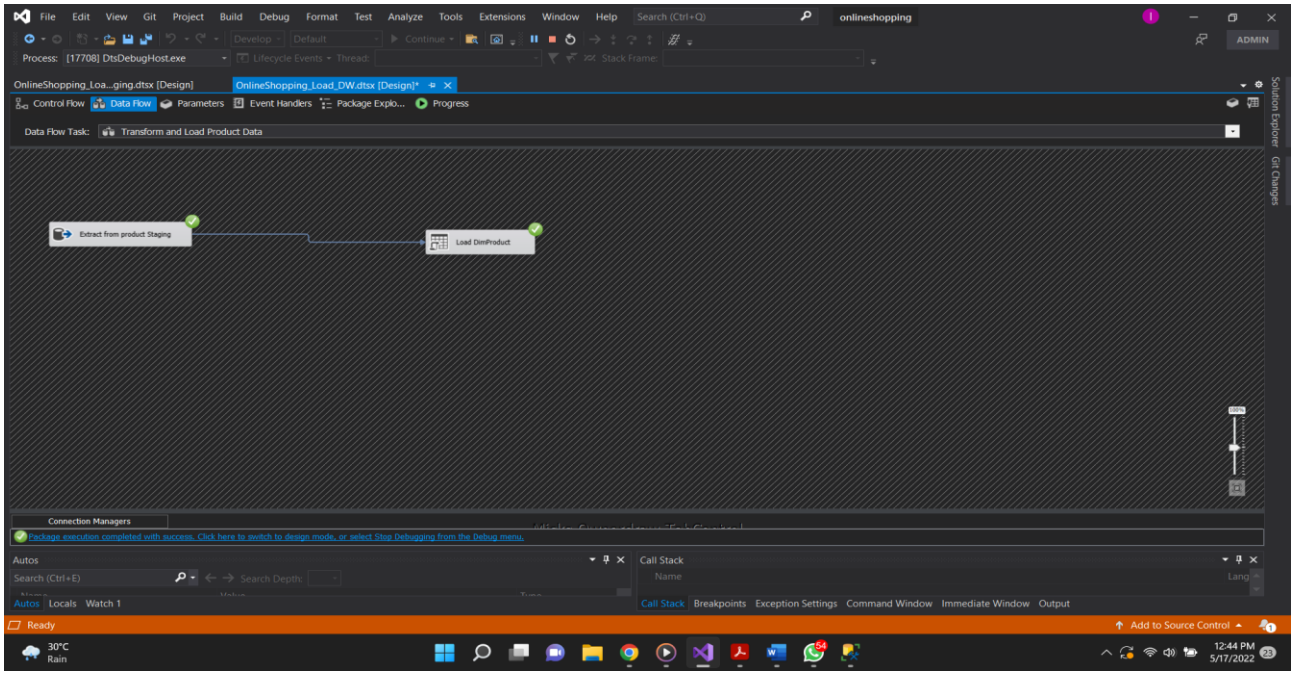


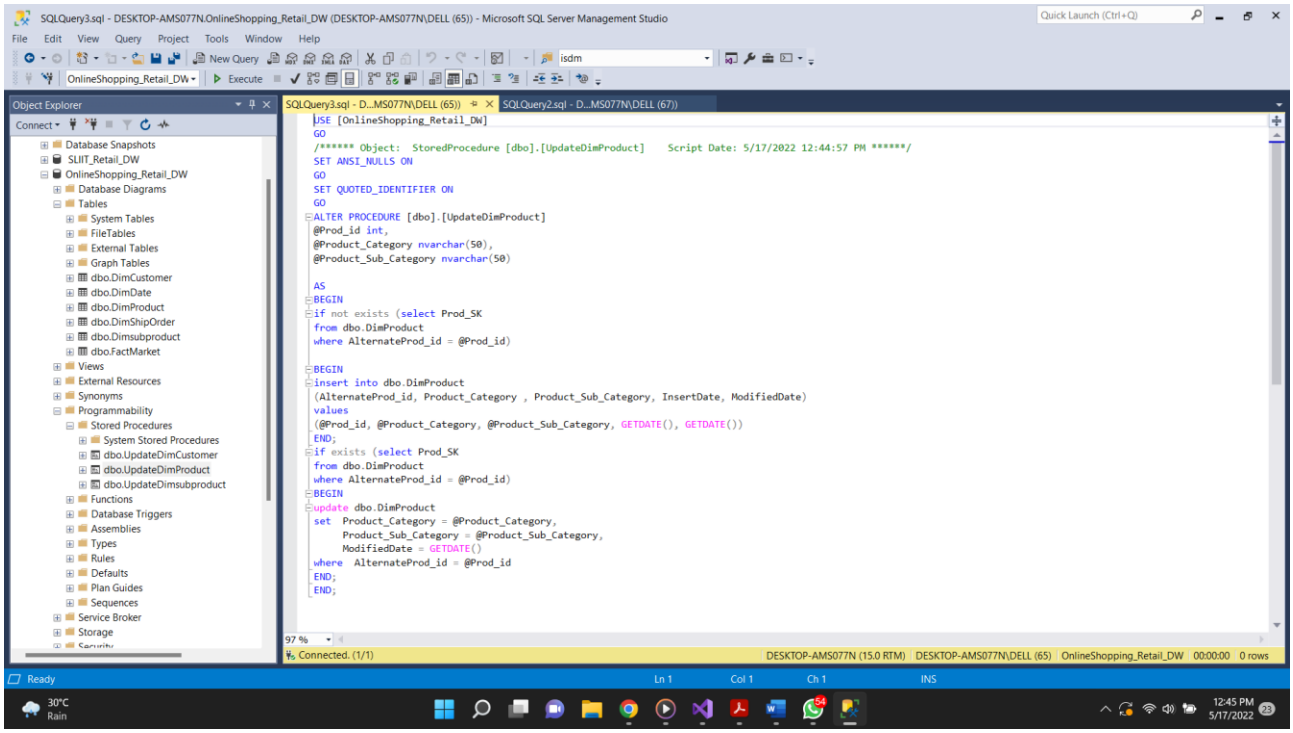
In this step, the Dimension Tables created in OnlineShopping_DW are loaded with the data of relevant staging tables.



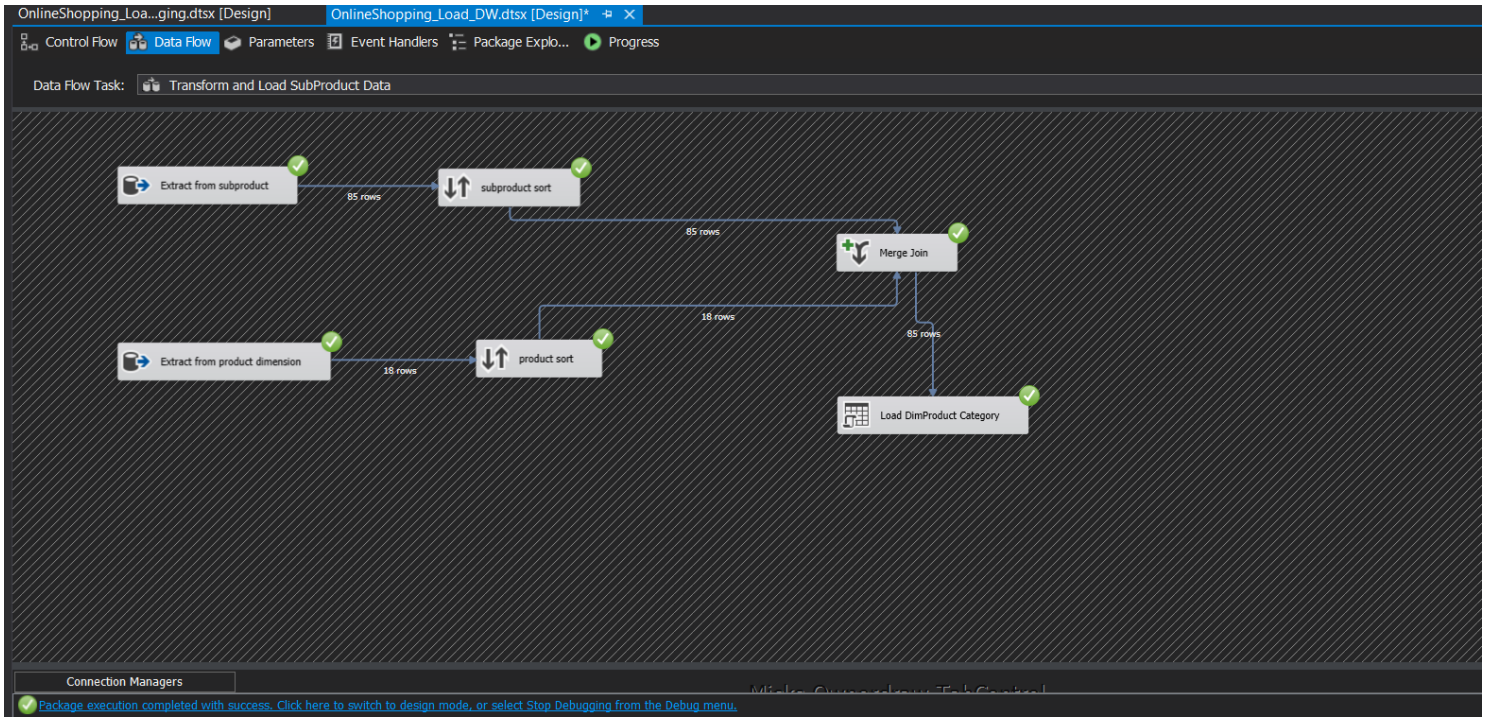


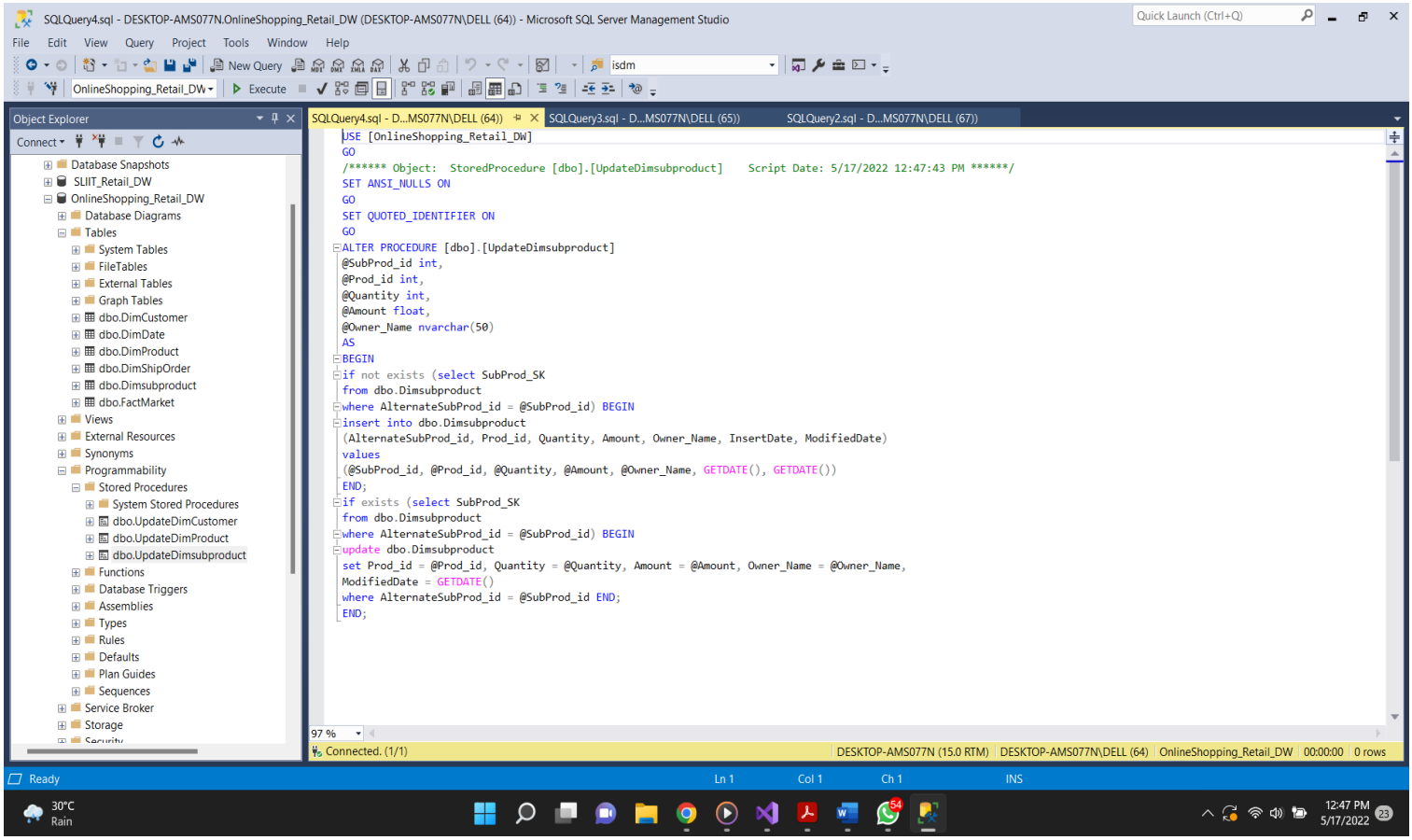
- Customer Data is Loaded to the DimCustomer
- UpdateDimCustomer procedure is used to check whether the data inserted or not





- Product data is loaded to the DimProduct
- UpdateDimProduct procedure is used to check whether the data inserted or not


OnlineShopping_Loa...ging.dtsx [Design]
OnlineShopping_Load_DW.dtsx [Design]*
Control Flow
Data Flow
Parameters
Event Handlers
Package Explo...
Progress
Data Flow Task:
Transform and Load SubProduct Data
Extract from subproduct
85 rows
subproduct sort
85 rows
Merge Join
18 rows
85 rows
Extract from product dimension
18 rows
product sort
Load DimProduct Category
Connection Managers
Package execution completed with success. Click here to switch to design mode, or select Stop Debugging from the Debug menu.


SQLQuery4.sql - DESKTOP-AMS077N.OnlineShopping_Retail_DW (DESKTOP-AMS077N\DELL (64)) - Microsoft SQL Server Management Studio
Quick Launch (Ctrl+Q)
File Edit View Query Project Tools Window Help
New Query
isdm
OnlineShopping_Retail_DW
Execute
Object Explorer
Connect
Database Snapshots
SLIIT_Retail_DW
OnlineShopping_Retail_DW
Database Diagrams
Tables
System Tables
FileTables
External Tables
Graph Tables
dbo.DimCustomer
dbo.DimDate
dbo.DimProduct
dbo.DimShipOrder
dbo.Dimsubproduct
dbo.FactMarket
Views
External Resources
Synonyms
Programmability
Stored Procedures
System Stored Procedures
dbo.UpdateDimCustomer
dbo.UpdateDimProduct
dbo.UpdateDimsubproduct
Functions
Database Triggers
Assemblies
Types
Rules
Defaults
Plan Guides
Sequences
Service Broker
Storage
SQLQuery4.sql - D...MS077N\DELL (64))
SQLQuery3.sql - D...MS077N\DELL (65))
SQLQuery2.sql - D...MS077N\DELL (67))
USE [OnlineShopping_Retail_DW]
GO
/****** Object: StoredProcedure [dbo].[UpdateDimsubproduct] Script Date: 5/17/2022 12:47:43 PM ******/
SET ANSI_NULLS ON
GO
SET QUOTED_IDENTIFIER ON
GO
ALTER PROCEDURE [dbo].[UpdateDimsubproduct]
@SubProd_id int,
@Prod_id int,
@Quantity int,
@Amount float,
@Owner_Name nvarchar(50)
AS
BEGIN
if not exists (select SubProd_SK
from dbo.Dimsubproduct
where AlternateSubProd_id = @SubProd_id) BEGIN
insert into dbo.Dimsubproduct
(AlternateSubProd_id, Prod_id, Quantity, Amount, Owner_Name, InsertDate, ModifiedDate)
values
(@SubProd_id, @Prod_id, @Quantity, @Amount, @Owner_Name, GETDATE(), GETDATE())
END;
if exists (select SubProd_SK
from dbo.Dimsubproduct
where AlternateSubProd_id = @SubProd_id) BEGIN
update dbo.Dimsubproduct
set Prod_id = @Prod_id, Quantity = @Quantity, Amount = @Amount, Owner_Name = @Owner_Name,
ModifiedDate = GETDATE()
where AlternateSubProd_id = @SubProd_id END;
END;
97 %
Connected. (1/1)
DESKTOP-AMS077N (15.0 RTM)
DESKTOP-AMS077N\DELL (64)
OnlineShopping_Retail_DW
00:00:00
0 rows
Ready
Ln 1
Col 1
Ch 1
INS
30°C
Rain
12:47 PM
5/17/2022

- UpdateDimSubProduct procedure is used to check whether the data inserted or not.
- Sort and Merge transformation tasks are used
- SubProduct data sorted according to the SubProd Id, and Product data extracted from the product dimensional table and sorted according to the Product ID and merged by Merge Join component.
- SubProduct Data is loaded to the DimSubProduct table

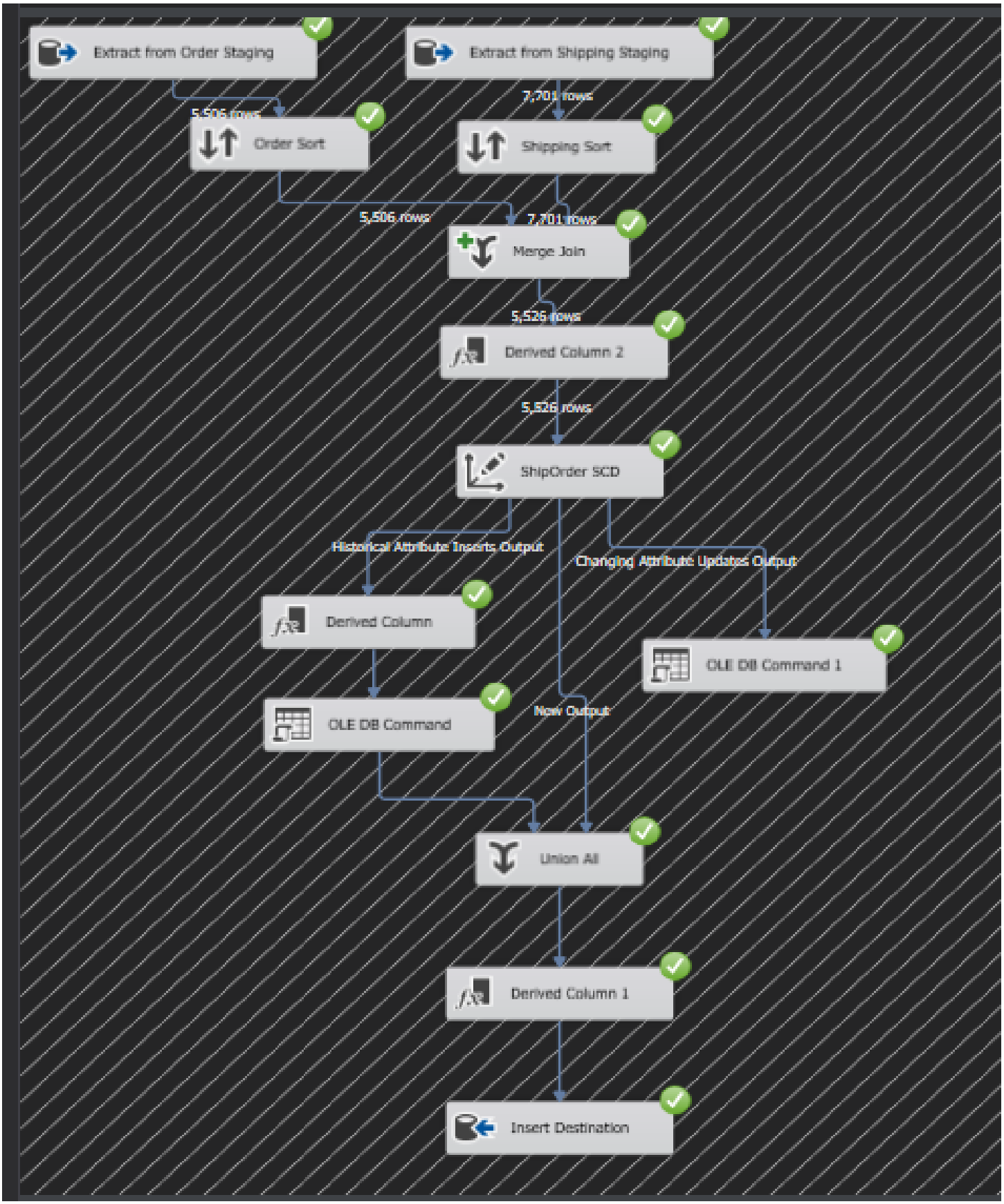
Loading Slowly Changing Dimension

- Here for Slowly Changing Dimension I selected DimShipOrder. For that I firstly merged Order table and Ship table is the slowly changing dimension in this dimensional modeling.
- In Order to load data to Dimension table, the slowly changing dimensions (historical) have two specific columns as StartDate & EndDate to ensure that the data is valid at the moment.
- slowly changing dimension wizard let the developer to select the Dimension table, Business keys of the dimension and what would be the slowly changing attributes.

Steps

- As mentioned earlier under assumptions, Ship Order details were considered as slowly changing details.
  The below mentioned columns were set as changing attributes:
  1. Order Priority (changing)
  2. Ship Mode (changing)

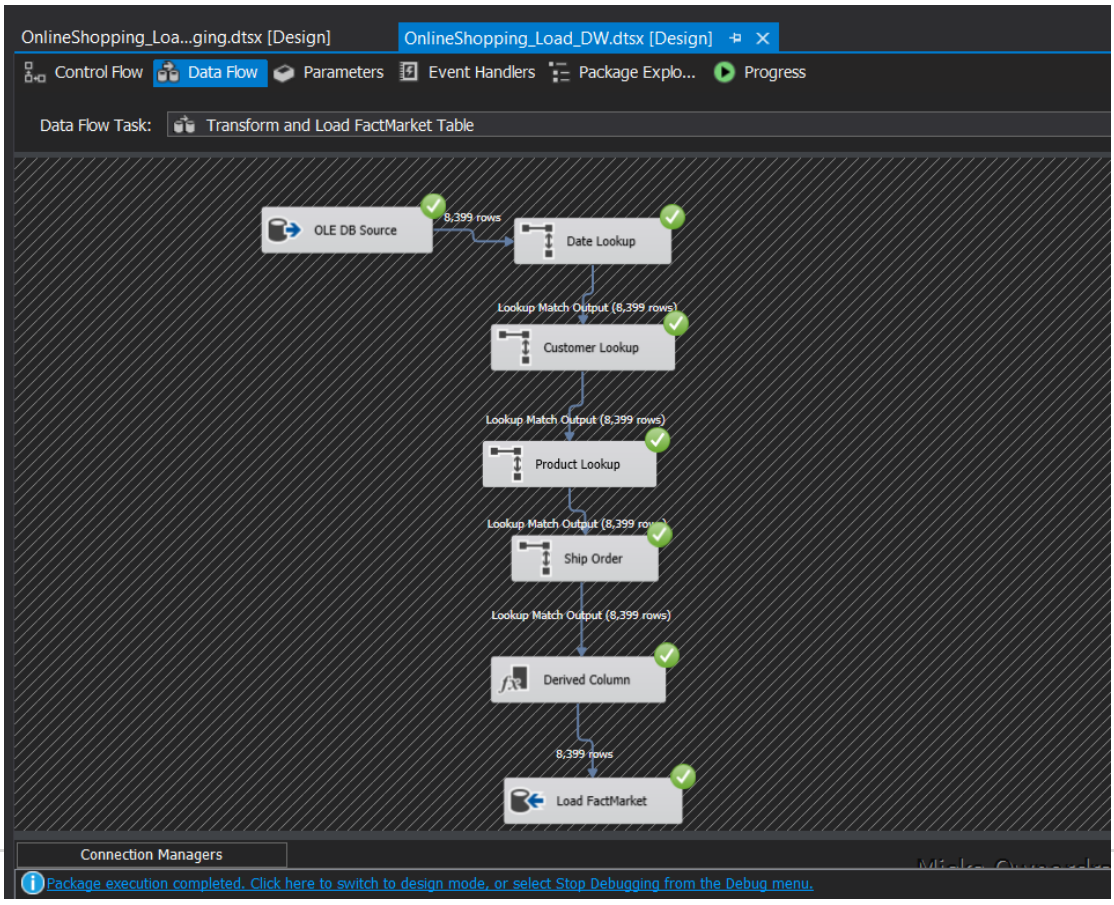
- After extracting data from the ShipOrder staging table, it was sorted according to the Order id and as it was identified as a slowly changing dimension, it was connected as shown above and loaded data to the ShipOrder dimension table.

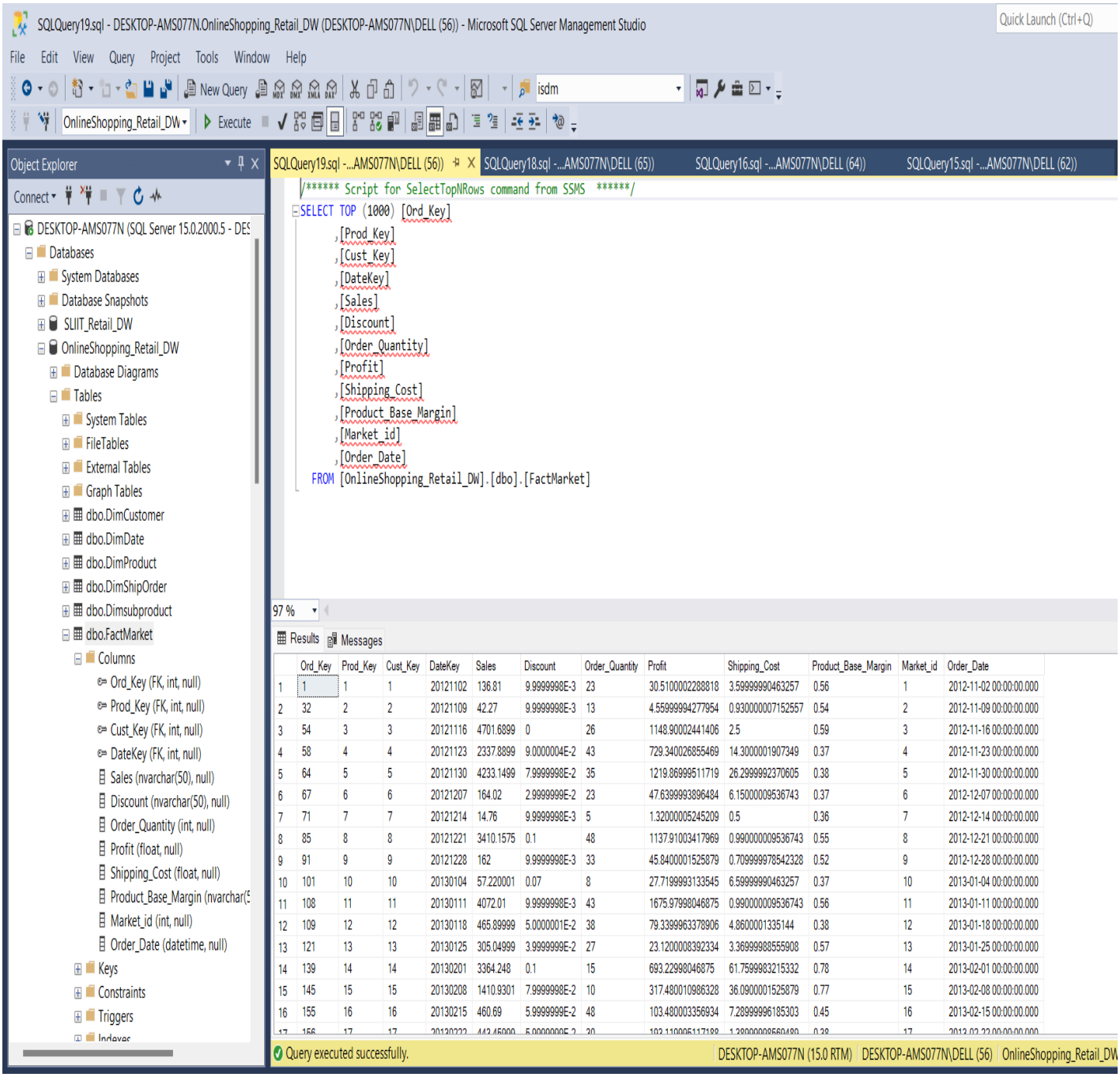

Extract from Order Staging
Extract from Shipping Staging
5,506 rows
7,701 rows
Order Sort
Shipping Sort
5,506 rows
7,701 rows
Merge Join
5,526 rows
Derived Column 2
5,526 rows
ShipOrder SCD
Historical Attribute Inserts Output
Changing Attribute Updates Output
Derived Column
OLE DB Command 1
OLE DB Command
New Output
Union All
Derived Column 1
Insert Destination

Load data to Fact Table

- The final step of Transformation & Loading is load data to fact table. According to the dimensional model, Here I transformed my Market Staging table to FactMarket table in data warehousing source db.
- FactMarket table has one date key which are related to Date Dimension as DateKey.
- After loading to all the dimensions, lastly data was loaded to the FactMarket table. The below steps were followed:

1. Data extracted from the Market staging
2. Join operation is done for the date using look up.
3. Join operation is done for the customer using look up.
4. Join operation is done for the Product look up.
5. Join operation is done for the ShipOrder look up.
6. insert and modified date were derived.
7. Fact details loaded to the FactMarket table.





	Ord_Key	Prod_Key	Cust_Key	DateKey	Sales	Discount	Order_Quantity	Profit	Shipping_Cost	Product_Base_Margin	Market_id	Order_Date
1	1	1	1	20121102	136.81	9.9999998E-3	23	30.5100002288818	3.59999990463257	0.56	1	2012-11-02 00:00:00.000
2	32	2	2	20121109	42.27	9.9999998E-3	13	4.55999994277954	0.930000007152557	0.54	2	2012-11-09 00:00:00.000
3	54	3	3	20121116	4701.6899	0	26	1148.90002441406	2.5	0.59	3	2012-11-16 00:00:00.000
4	58	4	4	20121123	2337.8899	9.0000004E-2	43	729.340026855469	14.3000001907349	0.37	4	2012-11-23 00:00:00.000
5	64	5	5	20121130	4233.1499	7.9999998E-2	35	1219.86999511719	26.2999992370605	0.38	5	2012-11-30 00:00:00.000
6	67	6	6	20121207	164.02	2.9999999E-2	23	47.6399993896484	6.15000009536743	0.37	6	2012-12-07 00:00:00.000
7	71	7	7	20121214	14.76	9.9999998E-3	5	1.32000005245209	0.5	0.36	7	2012-12-14 00:00:00.000
8	85	8	8	20121221	3410.1575	0.1	48	1137.91003417969	0.990000009536743	0.55	8	2012-12-21 00:00:00.000
9	91	9	9	20121228	162	9.9999998E-3	33	45.8400001525879	0.709999978542328	0.52	9	2012-12-28 00:00:00.000
10	101	10	10	20130104	57.220001	0.07	8	27.7199993133545	6.59999990463257	0.37	10	2013-01-04 00:00:00.000
11	108	11	11	20130111	4072.01	9.9999998E-3	43	1675.97998046875	0.990000009536743	0.56	11	2013-01-11 00:00:00.000
12	109	12	12	20130118	465.89999	5.0000001E-2	38	79.3399963378906	4.8600001335144	0.38	12	2013-01-18 00:00:00.000
13	121	13	13	20130125	305.04999	3.9999999E-2	27	23.1200008392334	3.36999988555908	0.57	13	2013-01-25 00:00:00.000
14	139	14	14	20130201	3364.248	0.1	15	693.22998046875	61.7599983215332	0.78	14	2013-02-01 00:00:00.000
15	145	15	15	20130208	1410.9301	7.9999998E-2	10	317.480010986328	36.0900001525879	0.77	15	2013-02-08 00:00:00.000
16	155	16	16	20130215	460.69	5.9999999E-2	48	103.480003356934	7.28999996185303	0.45	16	2013-02-15 00:00:00.000