**Data Warehousing Project Report**

**Executive Summary:**

Best Data Warehousing Design Consulting also known as Group One, was created as a global IT Consulting firm in Spring 2020. Based on various expertise of its founders, Group One has an excellent reputation of continuously designing and implementing excellent data warehousing solutions for various types of businesses. The focus of Group One is to design and implement an efficient and optimum enterprise data warehousing solution for its newest customer named “YL Electronic”.

As a brief Overview, YL Electronic is an established business that sells electronics via several channels. Due to its expected growth in sales, the company needs a data warehousing solution that will facilitate decision support views and streamline its operational database. Our goal is to satisfy YL Electronics' business needs, by providing an optimum data warehousing solution that will efficiently handle its complex database. This user-friendly data warehousing solution will store data, decision support views and ensure data integrity and security just to cite a few.

The content of this proposal includes a well elaborated data understanding, data modeling, and ETL process using Pentaho Data Integration tool.

As a conclusion, based on the YL Electronics' goal, objectives, structural hierarchy, reporting structure, and sales history transitions, we successfully designed and implemented an excellent data warehousing solution as shown in the steps below. Also, the final deliverable includes Data Warehousing Term Project, Appendix 1: ETL Process Using Pentaho Data Integration, Appendix 2: Estimate Storage Requirements for Dimensions and Fact table, Appendix 3: Decision Support Views, Summary Tables, and Data Understanding

# Table of Contents.

1. **Organizational Context**

**A) Describe the organization’s goals, objectives, structural hierarchy, reporting structure, decision-making processes based on your background reading on sales history transitions.**

The main goal of YL Electronics' Data Warehouse initiative is to reduce costs and yield more profits. To achieve this business goal, its overall objectives are to improve sales, increase revenues, and improve efficiency in their operational database. In addition, for an effective and efficient reporting structure, we have identified relevant decision support users and created well-crafted decision support views that allow them to generate great reports. Concerning the hierarchical structure, this company is a multinational organization that operates at the continental, country, regional, and sub regional level.

**B) Identify & classify all relevant Decision Support (DS) type user**s

After meticulous analysis and interactions with our client, the following list of DS users will be sufficient at the time the Data Warehouse is deployed. This does not preclude the need for other, or revised, users at a future date.

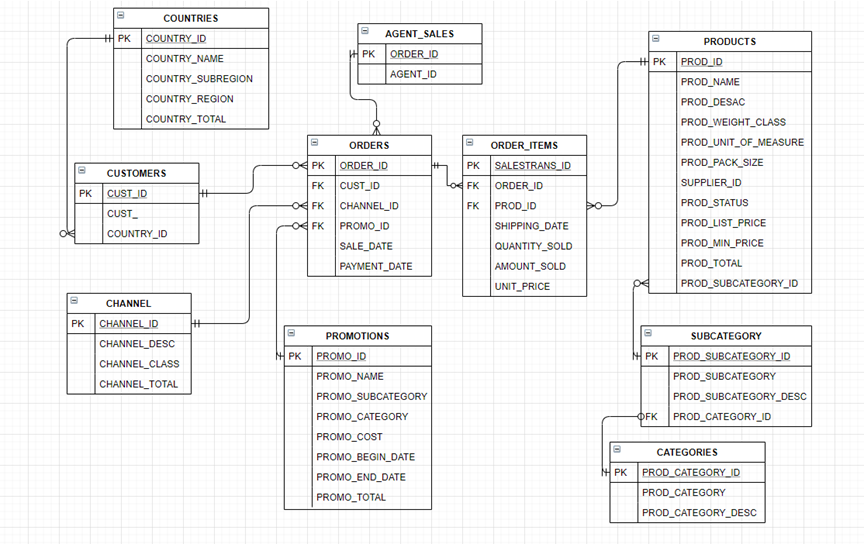
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **LABEL** | **NAME** | **USER TYPE** | **IMPORTANCE** | **SECURITY** | **QUERYING** | |
|  |  |  |  |  | **EXECUTE** | **CREATE** |
| Sales Manager | user01 | [X] INTERNAL  [ ] EXTERNAL  [ ] CUSTOMER  [ ] PUBLIC  [ ] REGULATOR  [ ]……. | [X] HIGHEST  [ ] VERY HIGH  [ ] MEDIUM  [ ] LOW  [ ] NONE | [X] HIGH  [ ] MEDIUM  [ ] LOW  [ ] NONE | [X] YES  [ ] NO | [ ] YES  [X] NO |
| Customer Success Manager | user02 | [X] INTERNAL  [ ] EXTERNAL  [ ] CUSTOMER  [ ] PUBLIC  [ ] REGULATOR  [ ]……. | [ ] HIGHEST  [X ] VERY HIGH  [ ] MEDIUM  [ ] LOW  [ ] NONE | [ ] HIGH  [X] MEDIUM  [ ] LOW  [ ] NONE | [ ] YES  [X] NO | [ ] YES  [X] NO |
| Marketing Manager | user03 | [X] INTERNAL  [ ] EXTERNAL  [ ] CUSTOMER  [ ] PUBLIC  [ ] REGULATOR  [ ]……. | [X] HIGHEST  [ ] VERY HIGH  [ ] MEDIUM  [ ] LOW  [ ] NONE | [X] HIGH  [ ] MEDIUM  [ ] LOW  [ ] NONE | [X] YES  [ ] NO | [X] YES  [ ] NO |
| Financial Manager | user04 | [X] INTERNAL  [ ] EXTERNAL  [ ] CUSTOMER  [ ] PUBLIC  [ ] REGULATOR  [ ]……. | [X] HIGHEST  [ ] VERY HIGH  [ ] MEDIUM  [ ] LOW  [ ] NONE | [X] HIGH  [ ] MEDIUM  [ ] LOW  [ ] NONE | [X] YES  [ ] NO | [ ]YES  [ ]NO |
| Product Manager | user05 | [X] INTERNAL  [ ] EXTERNAL  [ ] CUSTOMER  [ ] PUBLIC  [ ] REGULATOR  [ ]……. | [ ] HIGHEST  [X] VERY HIGH  [ ] MEDIUM  [ ] LOW  [ ] NONE | [ ] HIGH  [X] MEDIUM  [ ] LOW  [ ] NONE | [ ] YES  [X] NO | [ ] YES  [X] NO |

**C) Identify all relevant Decision Support Information Needs (i.e. Decision Support View)**  
Through discussions with the client about the goals of this DW project and subsequent analysis of their environment, the following Decision Support Views were developed for the list of DS users in Section 1.B above. Also, refer to Appendix 3 for views implementations

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **LABEL** | **DESCRIPTION OF DECISION SUPPORT VIEW[1]** | **MEASURES[2]** | **AGGREGATE** | **DIMENSIONS[3]** | **SECURITY** | **Length of Historical Period** |
| Sales Manager | Sales per agent  Agent Name, Sale Transaction,Sale Amount | number of transaction    Sales Amount | [ X ]SUM  [ X ]AVG  [ X ]COUNT  [ ]MAX  [ ]MIN  [ X ]RANK  [ ]% TILE | Date  Agent | [ ]HIGH  [x ] MEDIUM  [ ]LOW  [ ]NONE | [ ] Three Months  [ ] Six Months  [ ] 1 Year  [ ] 3 Years |
|  | Sales per product  Product Name, Sale Transaction,Sale Amount | number of transaction    Sales Amount | [ X ]SUM  [ ]AVG  [ X ]COUNT  [ ]MAX  [ ]MIN  [ X ]RANK  [ ]% TILE | Product  Date | [ ]HIGH  [ ] MEDIUM  [ X ]LOW  [ ]NONE | [ ] Three Months  [ ] Six Months  [ X ] 1 Year  [ ] 3 Years |
|  | Orders per customer  Customer Name, Sale Transaction,Sale Amount | number of transaction    quantity\_sold    amount\_sold | [ X ]SUM  [ ]AVG  [ X ]COUNT  [ ]MAX  [ ]MIN  [ X ]RANK  [ ]% TILE | Customer | [ ]HIGH  [x ] MEDIUM  [ ]LOW  [ ]NONE | [ ] Three Months  [ ] Six Months  [ X ] 1 Year  [ ] 3 Years |
|  | Sales per customer  Customer Name, Sale Transaction,Sale Amount | number of order | [ X ]SUM  [ ]AVG  [ X ]COUNT  [ ]MAX  [ ]MIN  [ X ]RANK  [ ]% TILE | Customer  Date | [ ]HIGH  [x ] MEDIUM  [ ]LOW  [ ]NONE | [ ] Three Months  [ ] Six Months  [ X ] 1 Year  [ ] 3 Years |
|  | Sales per location  Country, State, City, region, Sale Transaction,Sale Amount | number of transaction    amount\_sold | [ X ]SUM  [ ]AVG  [ X ]COUNT  [ ]MAX  [ ]MIN  [ X ]RANK  [ ]% TILE | Location  Date | [ ]HIGH  [ ] MEDIUM  [ X ]LOW  [ ]NONE | [ ] Three Months  [ ] Six Months  [ X ] 1 Year  [ ] 3 Years |
|  | Sales per channel  Channel Name, Sale Transaction,Sale Amount | number of transaction    amount\_sold | [ X ]SUM  [ ]AVG  [ X ]COUNT  [ ]MAX  [ ]MIN  [ X ]RANK  [ ]% TILE | Channel  Date  Customer | [ ]HIGH  [ ] MEDIUM  [ X ]LOW  [ ]NONE | [ ] Three Months  [ ] Six Months  [ X ] 1 Year  [ ] 3 Years |
|  | Sales per category/sub category  Category, Sub Category, Sale Transaction,Sale Amount | number of transaction    amount\_sold | [ X ]SUM  [ ]AVG  [ X ]COUNT  [ ]MAX  [ ]MIN  [ X ]RANK  [ ]% TILE | Category  Date | [ ]HIGH  [ ] MEDIUM  [ X ]LOW  [ ]NONE | [ ] Three Months  [ ] Six Months  [ X ] 1 Year  [ ] 3 Years |
| Customer Success Manager | Customer Retention | Sales Pattern (last purchase date, frequency of purchases)  Orders not turned into sales | [ X ]SUM  [ ]AVG  [ X ]COUNT  [ ]MAX  [ ]MIN  [ ]RANK  [ ]% TILE | Customers  Orders/Order Items  Dates | [ ]HIGH  [ X ]MEDIUM  [ ]LOW  [ ]NONE | [ ] Three Months  [ ] Six Months  [ X ] 1 Year  [ X ] 3 Years |
|  |  |  |  |  |  |  |
| Customer Success Manager | Shipping Turnaround Time | Orders  Shipping Date | [ ]SUM  [ ]AVG  [ ]COUNT  [ X ]MAX  [ X ]MIN  [ ]RANK  [ ]% TILE | Customers  Orders/Order Items  Dates | [ ]HIGH  [ X ]MEDIUM  [ ]LOW  [ ]NONE | [ X ] Three Months  [ X ] Six Months  [ X ] 1 Year  [ ] 3 Years |
| Marketing Manager | Post Promotion Analysis | Total Sales during promotion  Cost of products during promotion | [ X ]SUM  [ X ]AVG  [ X ]COUNT  [ X ]MAX  [ X ]MIN  [ X ]RANK  [ ]% TILE | Orders/ Order Items  Promotions  Products  Dates | [ ]HIGH  [ X ]MEDIUM  [ ]LOW  [ ]NONE | [ X ] Three Months  [ ] Six Months  [ ] 1 Year  [ ] 3 Years |
| Marketing Manager | Personalized Offers | Sales Pattern (last purchase date, frequency of purchases) | [ X ]SUM  [ ]AVG  [ X ]COUNT  [ ]MAX  [ ]MIN  [ ]RANK  [ ]% TILE | Customers  Orders/Order Items  Dates | [ ]HIGH  [ X ]MEDIUM  [ ]LOW  [ ]NONE | [ ] Three Months  [ ] Six Months  [ ] 1 Year  [ X ] 3 Years |
| Marketing Manager | Yearly Promotional Analysis | Total Sales by Channel  Total Sales by promotion | [ X ]SUM  [ X ]AVG  [ X ]COUNT  [ X ]MAX  [ X ]MIN  [ X ]RANK  [ ]% TILE | Orders/ Order Items  Promotions  Channels | [ ]HIGH  [ X ]MEDIUM  [ ]LOW  [ ]NONE | [ ] Three Months  [ ] Six Months  [ X ] 1 Year  [ ] 3 Years |
| Financial Manager | Monthly, Quarterly, Semiannual, Annual sales | Total Sales by period  Total Sales by period by region/subregion | [ X ]SUM  [ X ]AVG  [ X ]COUNT  [ X ]MAX  [ X ]MIN  [ X ]RANK  [ ]% TILE | Orders/ Order Items  Promotions  Channels | [ ]HIGH  [ X ]MEDIUM  [ ]LOW  [ ]NONE | [ X ] One Month  [ X ] Three Months  [ X ] Six Months  [ X ] 1 Year  [ ] 3 Years |

1. **Description of Operational Database**
2. **Develop ERD for operational database**

The current operational database of YL Electronics consists of 13 tables: agent sales, categories, channels, countries, internal customers, external customers, order items from 2012 and 2013, order items from 2014, orders from 2012 and 2013, orders from 2014, products, promotions, and subcategories. After thorough analysis of the current tables and further understanding of the data, we can visualize the current entity-relationship diagram (ERD) to be as followed:



Please note that both the orders and order items have two tables each, one for 2012-2013 data, the second for 2014 data. Additionally, customers in the current client’s database can be found in two separate tables, an internal as well as an external table. For the purpose of the diagram above, each of the three tables noted are shown only once.

1. **Classify tables as being Transaction, Component, Classification (optional)**

Immediately, we can see that the order items table and the orders table share a special relationship known as the master and detail. The master table (orders) has additional detailed information on the detail table (order items). While the detail table is clearly linked to the master table with the ORDER\_ID field as a foreign key, the master table does not have a key linking it to the details table. As this is the case, the orders and order items will now be linked and seen as one. Aside from the master detail relationship, the order/order items tables are known as a weak entity since it is on the many side to all the relationships it is involved with. Additionally, there is measurement data in the details table for QUANTITY\_SOLD, AMOUNT\_SOLD, and UNIT\_PRICE. For these reasons, the order/order items table is identified as the transaction entity.

The entities directly related to the transaction entity and answer the who, what, where, how, and why are the component entities. These can also be identified as being on the one side of the one to many relationships for the transaction table. With the orders/order items as the event, the customer table and the agent sales table answer the who involved. Products, of course is the what, while the channels and the promotions table answer how. While there is no date table, a date component will also be identified as there are dates found in the transaction table and it will answer the when question.

From the components, we can identify two classification entities, countries for customer and subcategories for products. Countries and subcategories are embedded hierarchies and share a one-to-many relationship with the component.

1. **Identify all hierarchies**

Location is a hierarchy that can be broken down to customer address > city > state/province > country > country region > country subregion. Category is also a hierarchy that can be defined as product > category > subcategory.

1. **Identify relevant domains for low cardinality non-numeric attributes & numeric attributes**

There are many columns throughout the ERD with low cardinality. On the channel table, CHANNEL\_CLASS has only three possible values: direct, indirect, and others. COUNTRY\_REGION from the country table has five values: Americas, Europe, Africa, Asia, and Oceania. For the customer tables, there are four columns with low cardinality: gender, marital status, education, and race. CUST\_GENDER has F for female and M for male. CUST\_EDUCATION has values of 1, 2, 3, and 4. RACE only has values 1 and 2 as well as CUST\_MARITAL\_STATUS with single and married. Lastly, PROD\_WEIGHT\_CLASS also has low cardinality with only known values of either 1 or 4.

The columns for CHANNEL\_TOTAL, COUNTRY\_TOTAL, CUST\_TOTAL, PROD\_UNIT\_OF\_MEASURE, PROD\_PACK\_SIZE, SUPPLIER\_ID, PROD\_STATUS, PROD\_TOTAL, and PROMO\_TOTAL all have one value. While these can be recognized as low cardinality, the value in these fields may not add value to the business. Therefore, these fields may be removed later in the process.

1. **Requirements Analysis & Formulation**
2. **Complete all tables in document entitled “Forms for Recording Requirements for DW Project**” Please refer to appendix 3 for Decision Support Views
3. **For each DS Information Need (i.e. Decision Support View), do analysis (e.g. data analysis) to determine whether it can be met by the proposed available operational data or whether it is unanswerable. Provide justification.**  
   **Sales Manager:** The Sales Manager is one of the most crucial roles that would need data from most if not all the data in system. There are multiple views the Sales Manager requires in order to make business decisions. For example, monthly sales, quarterly sales, and yearly sales are important to determine the profitability of the company as well as agent information to determine performance. To grab this information, the Sales manager would have to view two separate tables, orders and order items for most information. And then to get the agent information, the Sales Manager would need to reference a different table for this information.

**Customer Success Manager:** The Customer Success Manager (sometimes known as “account support”) has a direct need for information to support her role in Customer Retention. Her information needs differ from other similar views, such as when the Marketing Manager evaluates the success of a marketing campaign or analyzing the yearly marketing efficacy. Customer Retention, the primary responsibility of the Customer Success Manager, requires a much more customer-specific view, wherein a customer’s pattern of purchasing, would be necessary. This information can be obtained from the customer’s purchase history – to include (but not be limited to) last purchase made and frequency of purchases. Orders that do not become sales are of high importance to the Customer Success Manager, as a high number of them can indicate problems with the customer’s ordering or a general dissatisfaction with YL Electronic’s offerings.

In further support of the Customer Success Manager’s customer retention efforts, a view representing the time between orders and shipping provides valuable insight into a potential area of customer dissatisfaction.

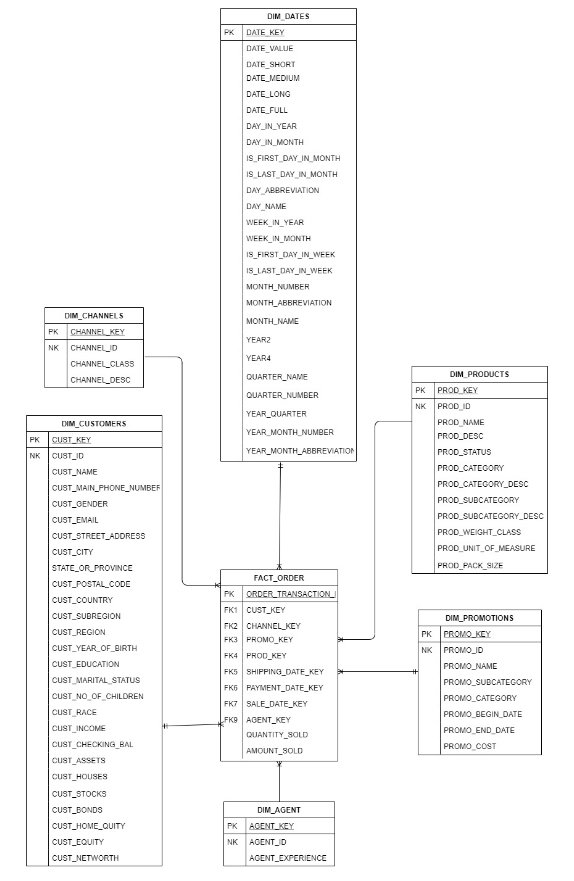
**Financial Manager:** The Financial Manager has an important, although distinct from sales, role in the business. The financial manager (sometimes referred to as “CFO” if the company uses C-level titles, is distinct from Sales. As such the Financial Manager has different reporting needs, and a higher sensitivity of those reporting needs. The Financial Manager’s security needs are more stringent than are most users’. The Financial Manager will need to view sales information, both by region and sub-region and company-wide, on a monthly, quarterly and annual reporting period. The Financial Manager must be privy to not only the revenue derived from sales, but also the discounts and cost to YL Electronics. The Financial Manager has a strong say in decisions affecting any future directions the company takes.

**Marketing Manager:** The Marketing Manager plays an integral role in sales without being directly involved with the day to day sales function. The marketing manager must plan and create promotions in order to bring in more sales and attract more customers. To complete this, the marketing manager will need to have access to the details of past promotional events and the sales brought in by the event. When viewing all the promotions, let’s say, for the last fiscal year, she can easily determine the total cost of the promotional event and the total sales from the event for a quick profit/loss analysis. In the current operational schema, the marketing manager will need to look at the orders items table to reference the order ID and then from there look at the promotion table to see what the promotion ID is. It’s a long and somewhat difficult process if the marketing manager is not familiar with the system.

Another responsibility the Marketing Manager has would be the review at year-end of the success of Marketing’s efforts overall. As this measure is not tied to a specific single Promotion, the scope is different from the Marketing Manager’s other views. Access to channel-specific order totals would be necessary, while the scope would be limited by year rather than promotion.

The Marketing Manager is not necessary interested in the dollar amount of sales, rather what is being sold and the rate it is sold at. Promotional events offer large discounts to customers and usually results in more products sold. Product analysis to determine which products are sold the most and the average discount of a product can help the Product Manager make decisions like which products to buy more of, what product categories are most popular, which product is mostly sold when the discount rate is high, etc. Currently, the Product Manager has six different places to source this information. And even then, there are multiple formulas and aggregations needed in order to produce the numbers needed.

1. **Dimensional Model**
2. **Provide a description of your model**



1. **Provide justification for your model**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Transaction Entity | Component Entity | Classification Entity | Hierarchy | Measures | Date Dimension Columns |
| ORDERS/ ORDER ITEMS | CUSTOMERS | COUNTRIES | Client Address– City-State/Province - Country | QUANTITY\_SOLD  AMOUNT\_SOLD | SHIPPING DATE  PAYMENT DATE  SALE DATE |
| CHANNELS |  |
| PROMOTIONS |  |
| AGENT |  |
| PRODUCTS | SUBCATEGORIES  CATEGORIES |

We decide to use star schema to data modeling our data warehouse. We have embedded country information into customer table and subcategories and categories into products table. Because the order table only has the associated ID, we decide to embed the order table into the order\_item table, and it becomes our fact table.

1. **Physical Database Design (Part 1)**
2. **Provide a description of all tables**

DIM\_PROMOTIONS: Promotion information include promotion name, promotion start date, and end date.

DIM\_PRODUCTS: Product information include product name, product description, product’s category and subcategory, product’s list price and minimum price

DIM\_CUSTOMERS: All customer information includes customer name, customer’s contact information, customer’s address along with customer’s personal information. DIM\_CUSTOMERS is a type 2-dimension table.

DIM\_AGENTS: Sales agent information includes agent experience level and education level.

DIM\_DATE: Date information

DIM\_CHANNEL: Channel information include channel class and description

FACT\_ORDER: The fact table that embeds order table and order item in the transaction database. Provides the sold amount and sold number along with all key associated with dimension tables.

1. **Estimate Storage Requirements for Dimension & Fact tables. Provide a justification.**

Refer to appendix 2; Storage Estimation, for detailed calculations and justification

In addition, below is the summary table for the estimation of storage requirements.

Resulted Storage Requirements Table:

|  |  |
| --- | --- |
| Table | Estimated Heap Size in Bytes |
| DIM\_PROMOTIONS | 278528 |
| DIM\_CUSTOMERS | 27557888 |
| DIM\_DATE | 1720320 |
| DIM\_CHANNELS | 528 |
| DIM\_AGENTS | 40960 |
| DIM\_PRODUCTS | 7946.24 |
| FACT\_ORDERS | 61079552 |

1. **Provide SQL Scripts for creating Dimension & Fact tables**.

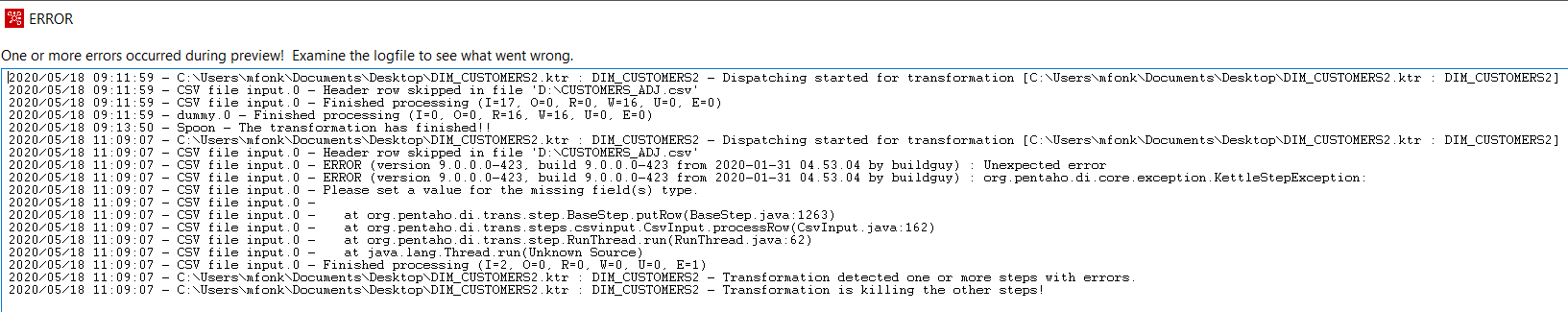
|  |  |
| --- | --- |
| Table | SQL Scripts |
| Schema | CREATE SCHEMA GROUP1\_DWH; |
| DIM\_DATE | CREATE TABLE GROUP1\_DWH.DIM\_DATE (  DATE\_KEY INT NOT NULL,  DATE\_VALUE DATE NOT NULL,  DATE\_SHORT CHAR(12) NOT NULL,  DATE\_MEDIUM CHAR(16) NOT NULL,  DATE\_LONG CHAR(24) NOT NULL,  DATE\_FULL CHAR(32) NOT NULL,  DAY\_IN\_YEAR SMALLINT NOT NULL,  DAY\_IN\_MONTH SMALLINT NOT NULL,  IS\_FIRST\_DAY\_IN\_MONTH CHAR(10) NOT NULL,  IS\_LAST\_DAY\_IN\_MONTH CHAR(10) NOT NULL,  DAY\_ABBREVIATION CHAR(3) NOT NULL,  DAY\_NAME CHAR(12) NOT NULL,  WEEK\_IN\_YEAR SMALLINT NOT NULL,  WEEK\_IN\_MONTH SMALLINT NOT NULL,  IS\_FIRST\_DAY\_IN\_WEEK CHAR(10) NOT NULL,  IS\_LAST\_DAY\_IN\_WEEK CHAR(10) NOT NULL,  MONTH\_NUMBER SMALLINT NOT NULL,  MONTH\_ABBREVIATION CHAR(3) NOT NULL,  MONTH\_NAME CHAR(12) NOT NULL,  YEAR2 CHAR(2) NOT NULL,  YEAR4 SMALLINT NOT NULL,  QUARTER\_NAME CHAR(2) NOT NULL,  QUARTER\_NUMBER SMALLINT NOT NULL,  YEAR\_QUARTER CHAR(7) NOT NULL,  YEAR\_MONTH\_NUMBER CHAR(7) NOT NULL,  YEAR\_MONTH\_ABBREVIATION CHAR(8) NOT NULL  ); |
| DIM\_CHANNELS | CREATE TABLE GROUP1\_DWH.DIM\_CHANNELS (  CHANNEL\_KEY INT NOT NULL,  CHANNEL\_ID INT NOT NULL,  CHANNEL\_CLASS VARCHAR(20),  CHANNEL\_DESC VARCHAR(20) NOT NULL  ); |
| DIM\_PROMOTIONS | CREATE TABLE GROUP1\_DWH.DIM\_PROMOTIONS (  PROMO\_KEY INT NOT NULL,  PROMO\_ID INT NOT NULL,  PROMO\_NAME VARCHAR(30) NOT NULL,  PROMO\_SUBCATEGORY VARCHAR(30) NOT NULL,  PROMO\_CATEGORY VARCHAR(30) NOT NULL,  PROMO\_COST DECIMAL(8, 2) NOT NULL,  PROMO\_BEGIN\_DATE DATETIME NOT NULL,  PROMO\_END\_DATE DATETIME NOT NULL  ); |
| DIM\_PRODUCTS | CREATE TABLE GROUP1\_DWH.DIM\_PRODUCTS (  PROD\_KEY INT NOT NULL,  PROD\_ID INT NOT NULL,  PROD\_NAME VARCHAR(50) NOT NULL,  PROD\_DESC VARCHAR(4000) NOT NULL,  PROD\_STATUS VARCHAR(20) NOT NULL,  PROD\_SUBCATEGORY VARCHAR(50) NOT NULL,  PROD\_SUBCATEGORY\_DESC VARCHAR(2000) NOT NULL,  PROD\_CATEGORY VARCHAR(50) NOT NULL,  PROD\_CATEGORY\_DESC VARCHAR(2000) NOT NULL,  PROD\_WEIGHT\_CLASS SMALLINT NOT NULL,  PROD\_UNIT\_OF\_MEASURE CHAR(1) NOT NULL,  PROD\_PACK\_SIZE CHAR(1) NOT NULL  ); |
| DIM\_AGENT | CREATE TABLE GROUP1\_DWH.DIM\_AGENT (  AGENT\_KEY INT NOT NULL,  AGENT\_ID INT NOT NULL,  AGENT\_EXPERIENCE VARCHAR(20) NOT NULL  ); |
| DIM\_CUSTOMERS | CREATE TABLE GROUP1\_DWH.DIM\_CUSTOMERS (  CUST\_KEY INT NOT NULL,  CUST\_ID INT NOT NULL,  CUST\_NAME VARCHAR(60) NOT NULL,  CUST\_GENDER CHAR(1),  CUST\_MAIN\_PHONE\_NUMBER VARCHAR(25) NOT NULL,  CUST\_EMAIL VARCHAR(30) NOT NULL,  CUST\_STREET\_ADDRESS VARCHAR(40) NOT NULL,  CUST\_CITY VARCHAR(30) NOT NULL,  CUST\_STATE\_PROVINCE VARCHAR(40) NOT NULL,  CUST\_POSTAL\_COD INT NOT NULL,  CUST\_COUNTRY\_NAME VARCHAR(50) NOT NULL,  CUST\_SUB\_REGION VARCHAR(30) NOT NULL,  CUST\_REGIO VARCHAR(20) NOT NULL,  CUST\_YEAR\_OF\_BIRTH SMALLINT,  CUST\_EDUCATION INT,  CUST\_MARITAL\_STATUS VARCHAR(20),  CUST\_NO\_OF\_CHILDREN INT,  CUST\_CHECKING\_BAL INT,  CUST\_RACE INT,  CUST\_INCOME FLOAT,  CUST\_ASSET INT,  CUST\_HOUSES INT,  CUST\_STOCKS INT,  CUST\_BONDS INT,  CUST\_HOME\_EQUITY INT,  CUST\_EQUITY INT,  CUST\_NETWORTH INT,  CUST\_VERSION SMALLINT,  CUST\_DATE\_FROM DATETIME,  CUST\_DATE\_THROUGH DATETIME  ); |
| FACT\_ORDERS | CREATE TABLE GROUP1\_DWH.FACT\_ORDERS(  CUST\_KEY INT NOT NULL,  CHANNEL\_KEY INT NOT NULL,  PROMO\_KEY INT NOT NULL,  PROD\_KEY INT NOT NULL,  AGENT\_KEY INT NOT NULL,  SHIPPING\_DATE\_KEY INT NOT NULL,  PAYMENT\_DATE\_KEY INT NOT NULL,  SALE\_DATE\_KEY INT NOT NULL,  QUANTITY\_SOLD INT NOT NULL,  AMOUNT\_SOLD DECIMAL(8,2) NOT NULL,  UNIT\_PRICE DECIMAL(8,2) NOT NULL  ); |
| OTHER USEFUL STATEMENTS | DROP TABLE GROUP1\_DWH.FACT\_ORDERS;  DROP TABLE GROUP1\_DWH.DIM\_AGENT;  DROP TABLE GROUP1\_DWH.DIM\_PRODUCTS;  DROP TABLE GROUP1\_DWH.DIM\_PROMOTIONS;  DROP TABLE GROUP1\_DWH.DIM\_CUSTOMERS;  DROP TABLE GROUP1\_DWH.DIM\_CHANNELS;  DROP TABLE GROUP1\_DWH.DIM\_DATE; |

1. **Extract, Transformation, Load (ETL) Processes**
2. **Describe processes, including sequence of execution of the constituent activities**

We started the ETL process by loading randomly our type 1 dimension and DIM\_DATE. Then, we successfully processed our type2 dimension DIM\_CUSTOMERS. Lastly, we transformed and loaded our FACT\_ORDERS table. This is because in order to successfully load fact\_orders, we had to get different keys from dimensional tables.

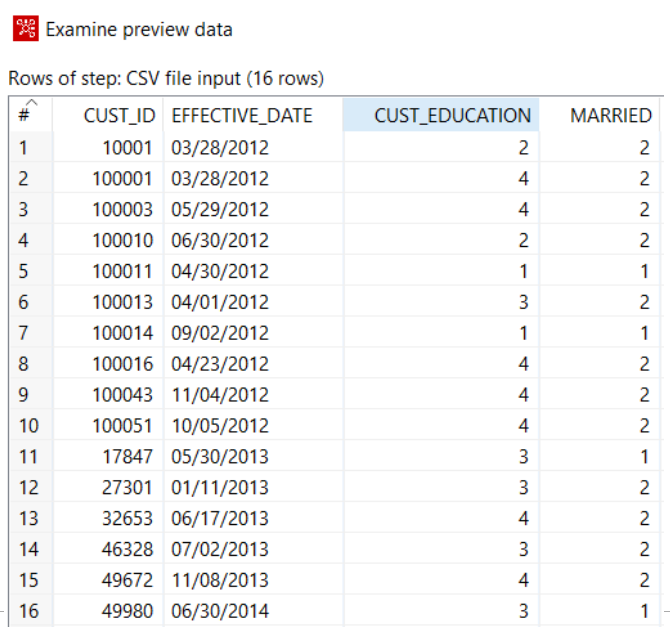
1. **Description of data that had to be cleaned & actions that were taken**

During the ETL process, most of our data was clean except for the following additional data set: CUSTOMERS\_ADJ.CSV and AGENT TABLE. Major cleansings that we carried out were on the date formatting. Initially, when we attempted to read the csv files, the error shown below appeared:



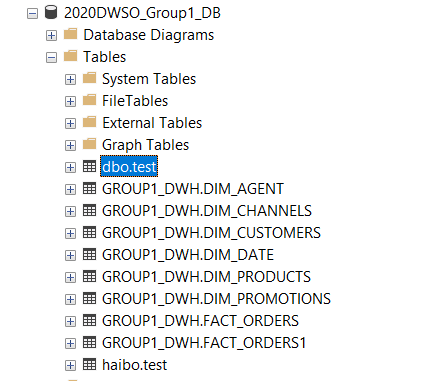
Upon cleansing our data set and reloading, we were successfully able to read the file as shown

below:



1. ETL process using Pentaho Data Integration tool:

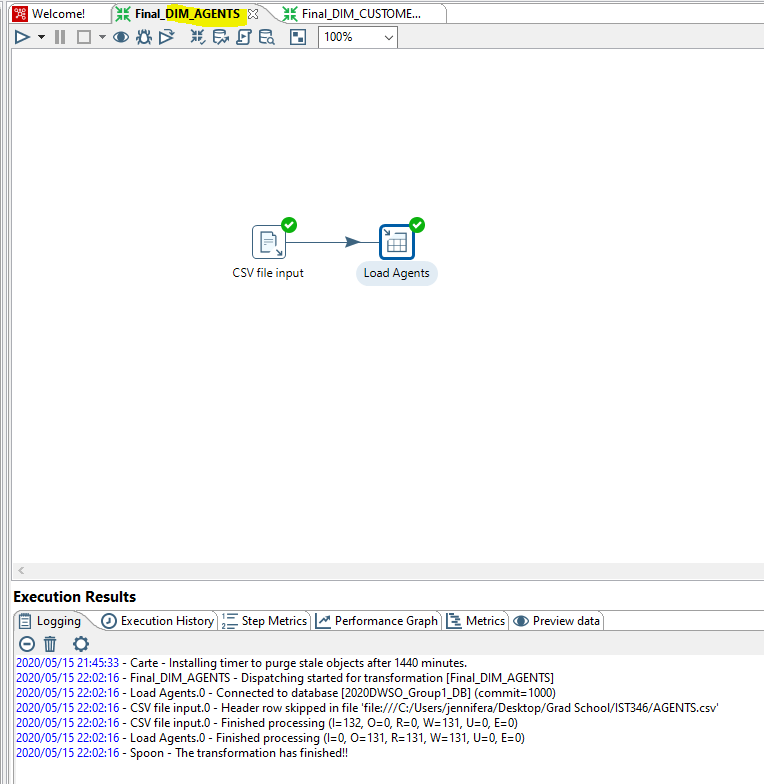
* store the resulting data in Temporary Tables in our own schema (i.e. a staging area) :

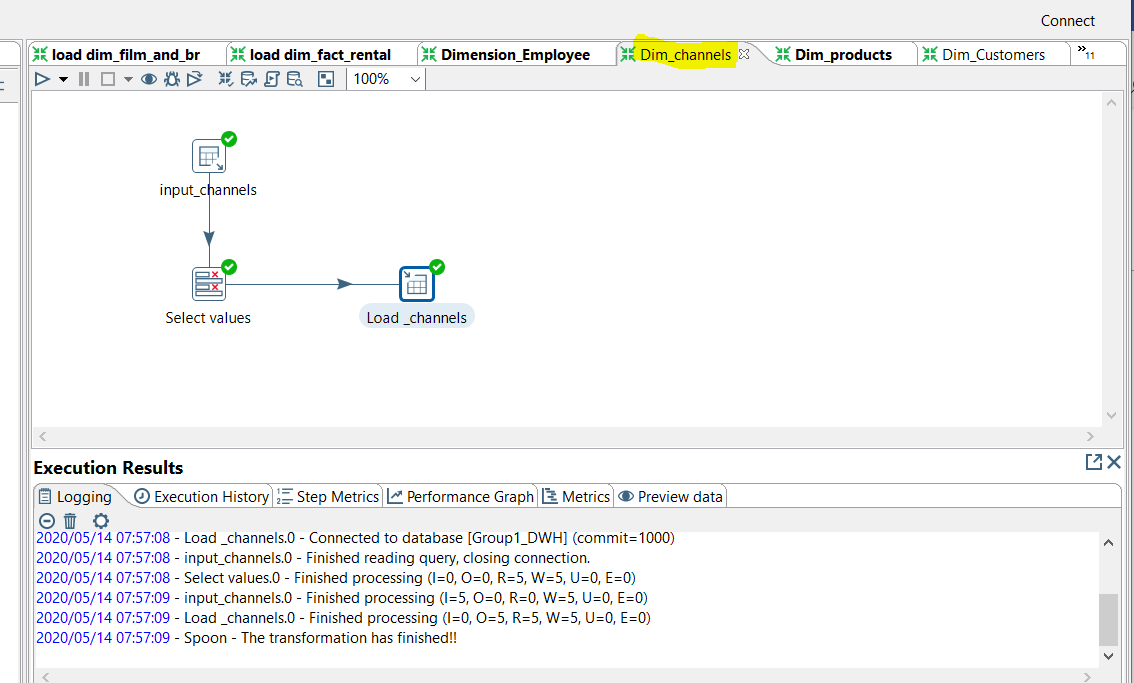


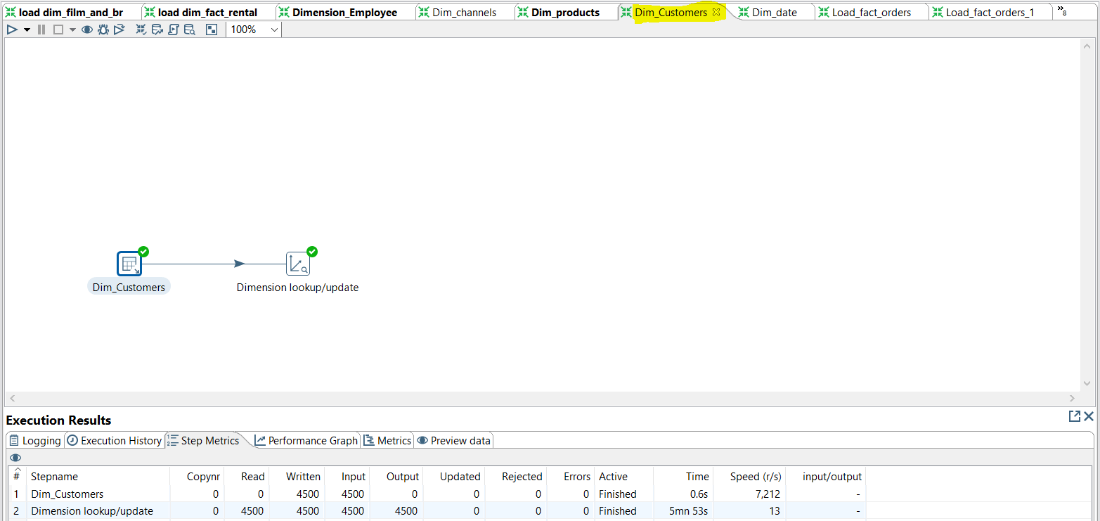
* Provide a screenshot for each transformation and describe the transformation logic.

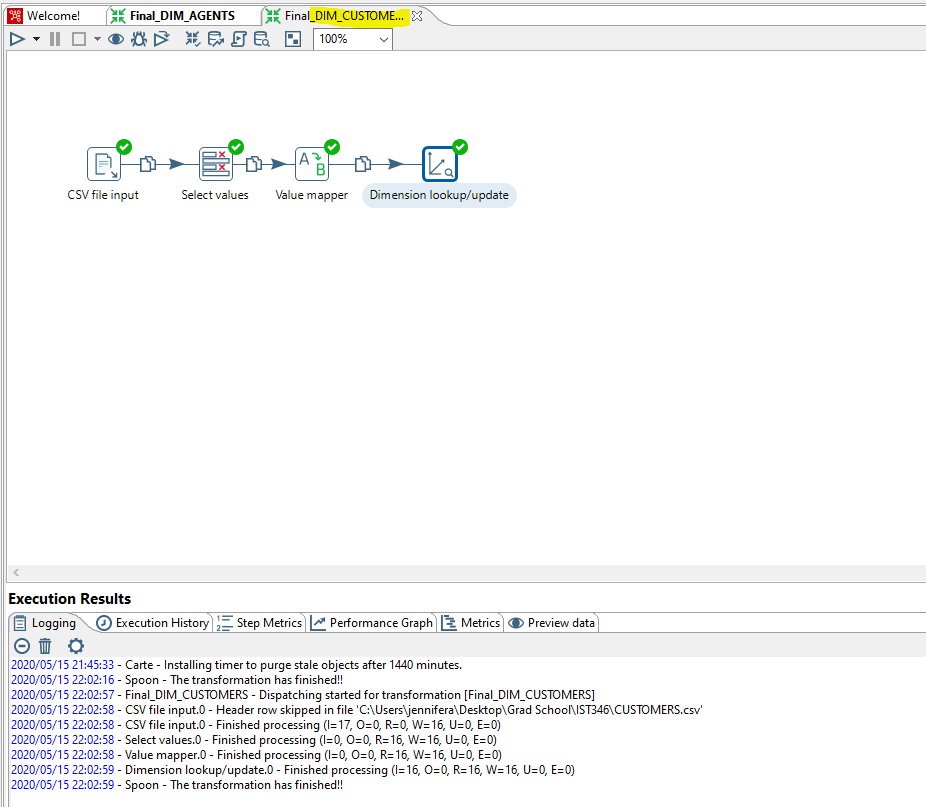
Please, Refer to Appendix 1, for each transformation and description of the transformation logic

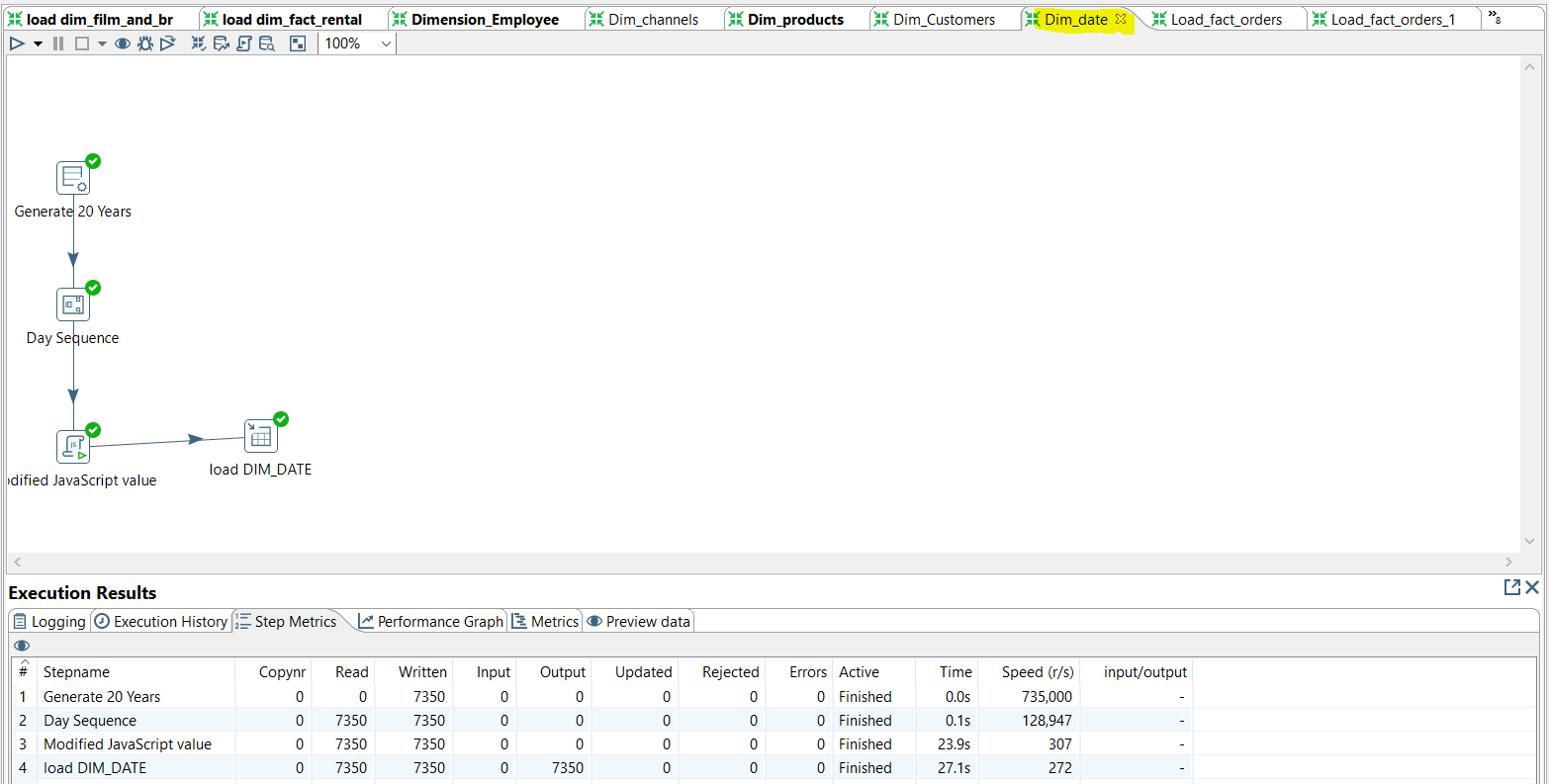
* Provide a screenshot for the ETL job

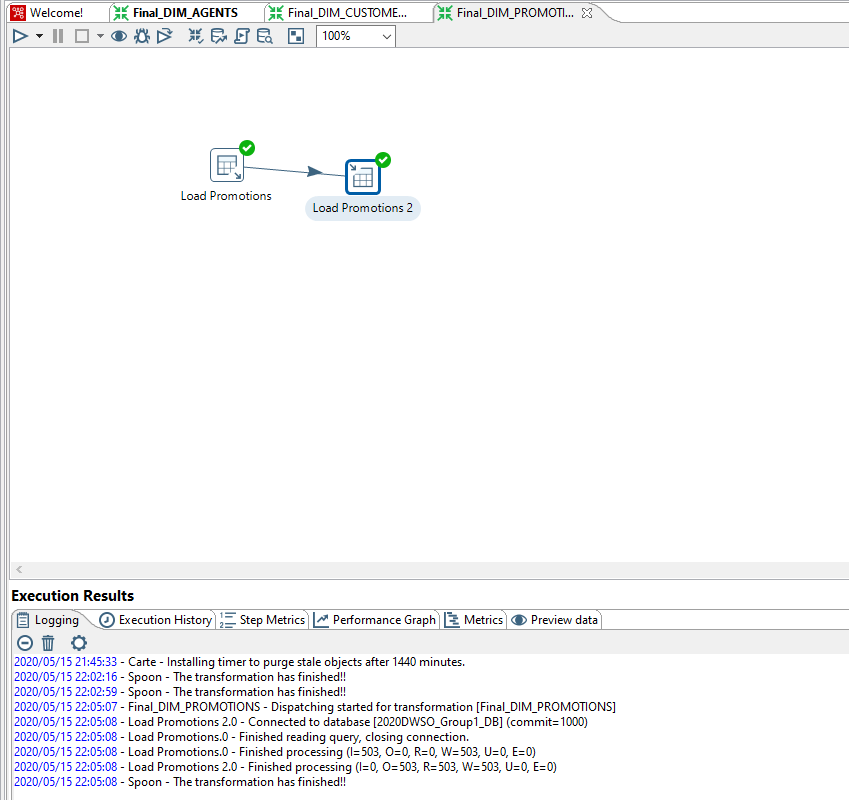


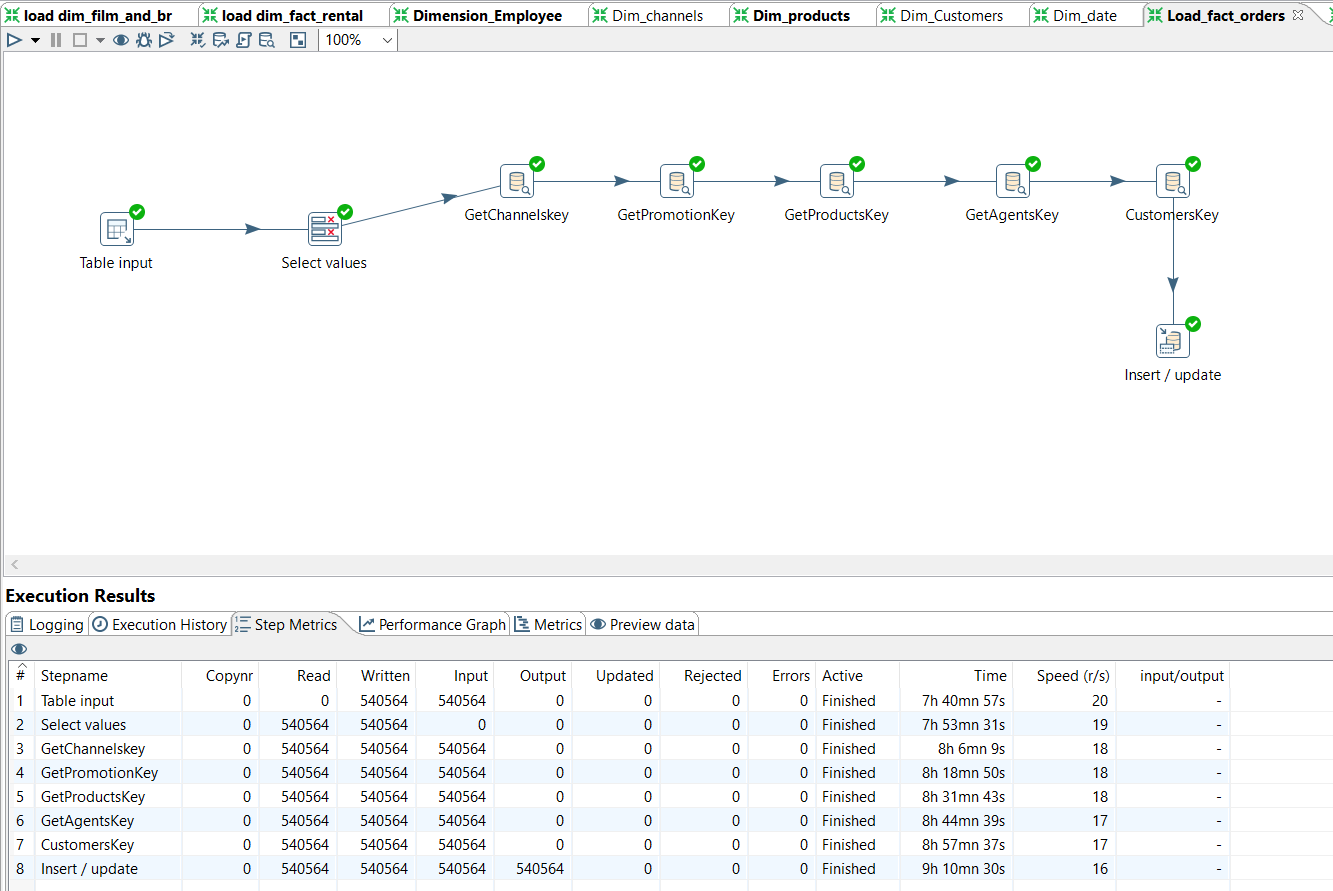






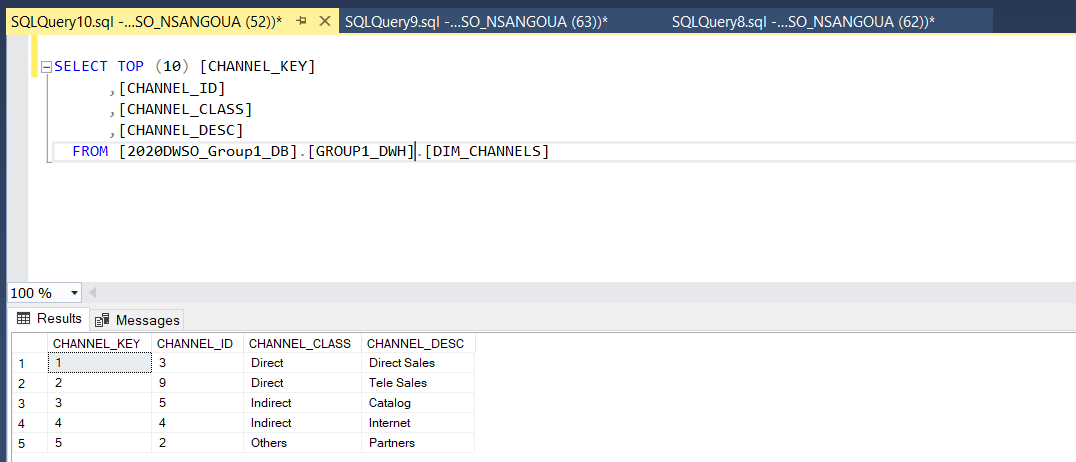


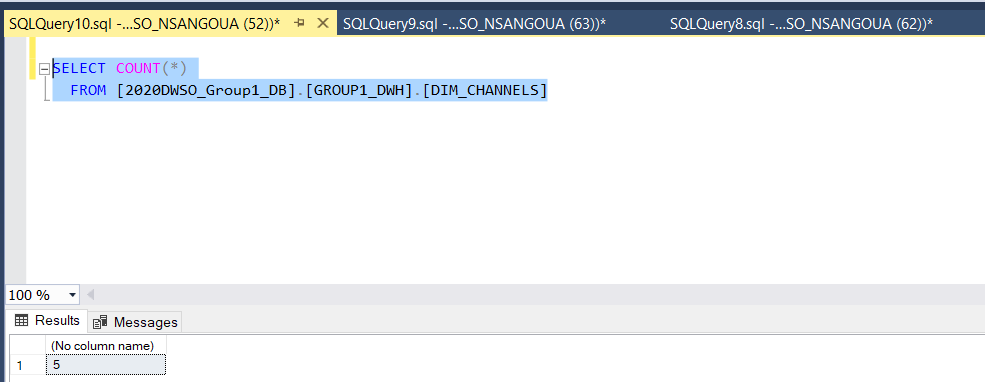




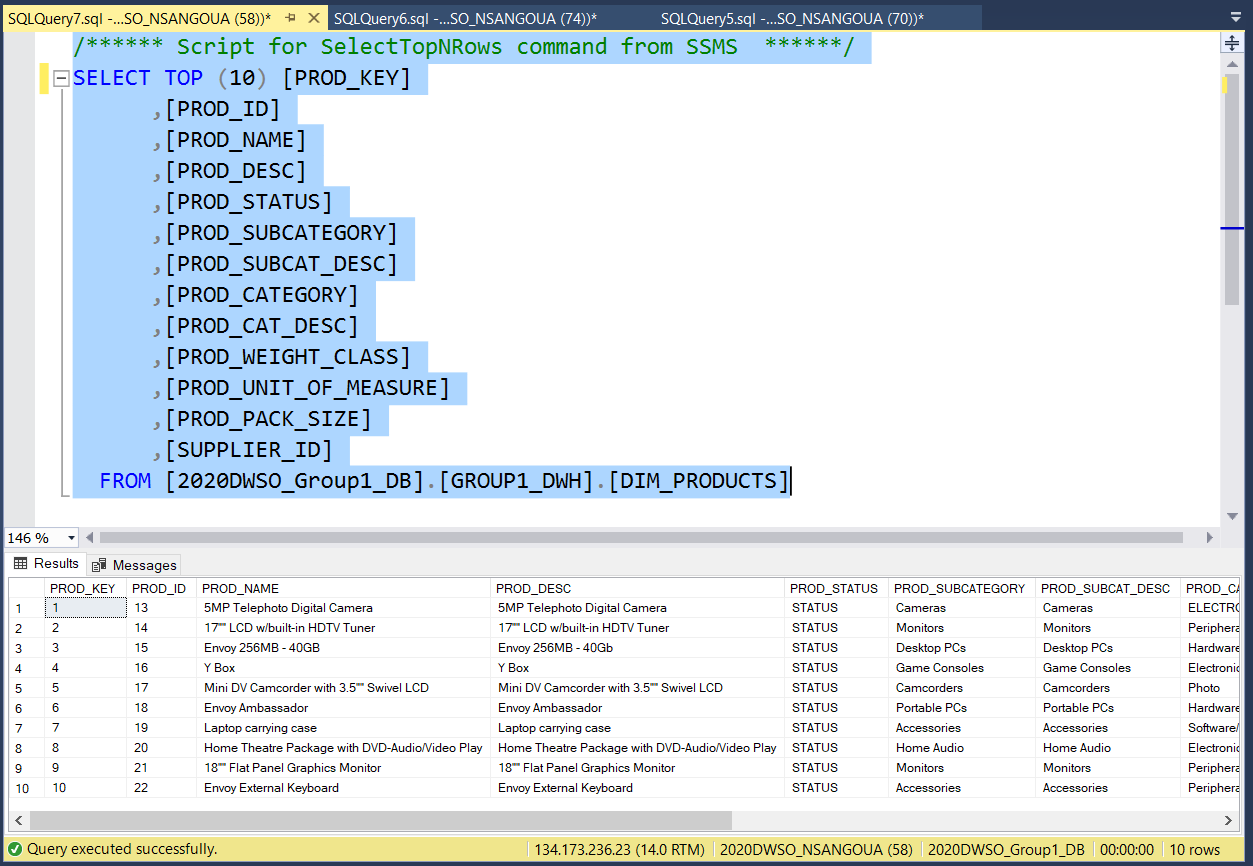
* Provide screenshots for initial load of all tables (e.g. top 10 rows of each table and number of rows in each table)

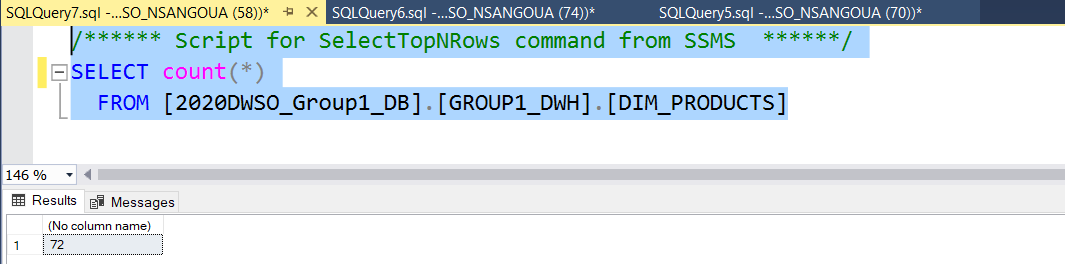
DIM\_CHANNELS



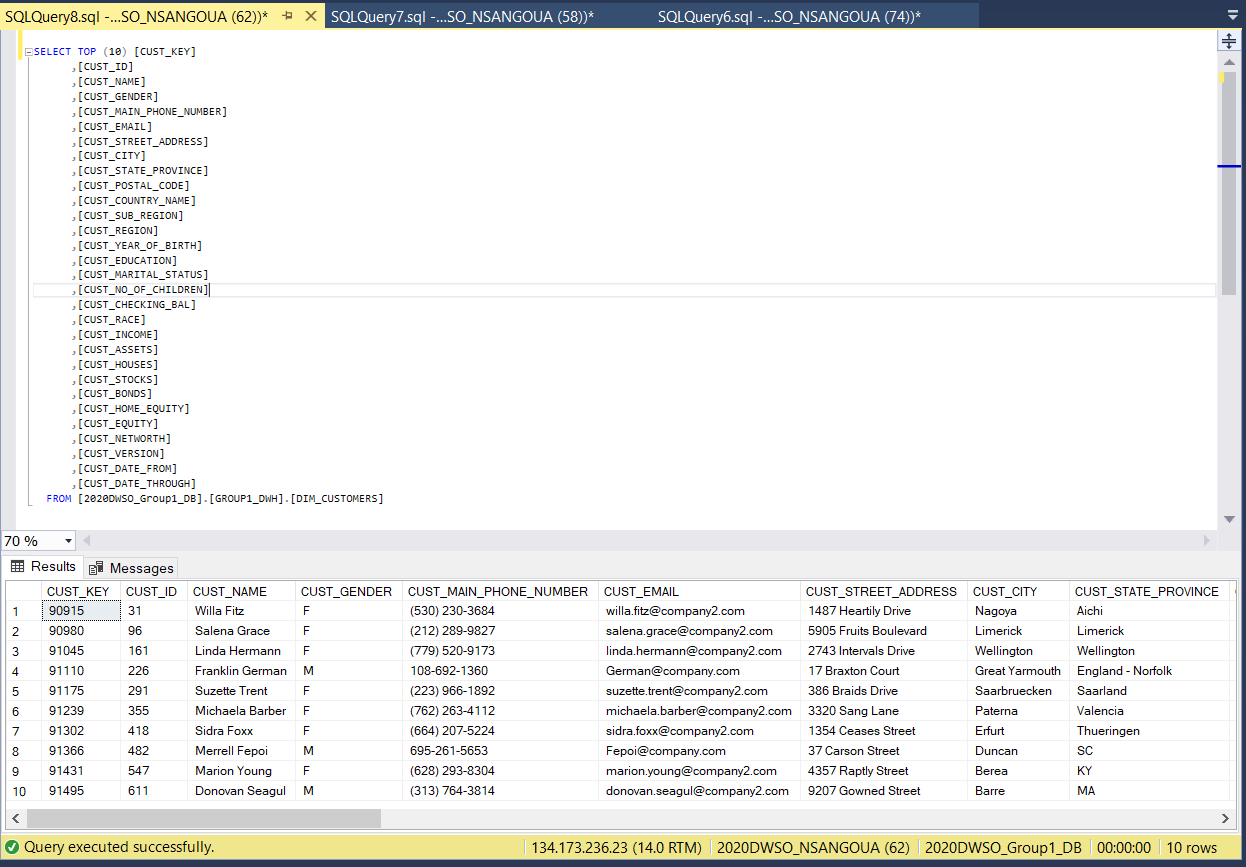


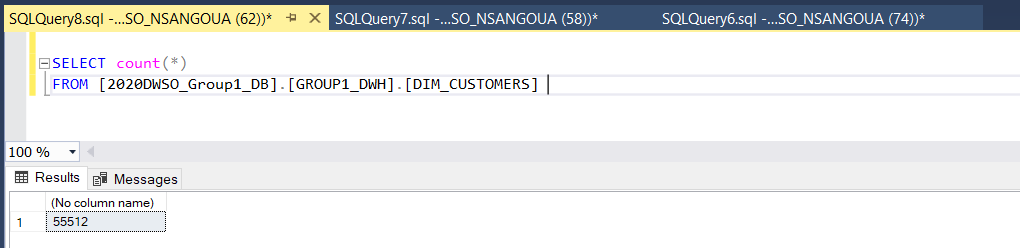
DIM\_PRODUCTS



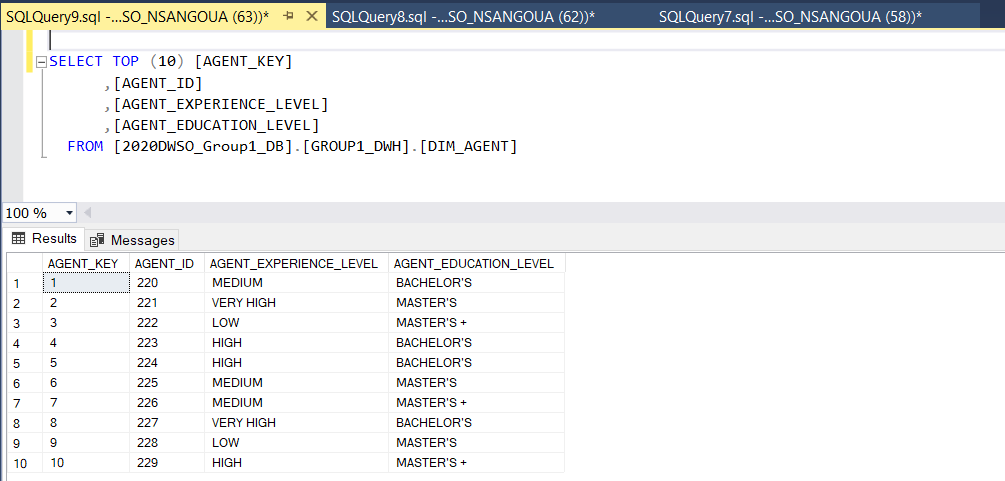


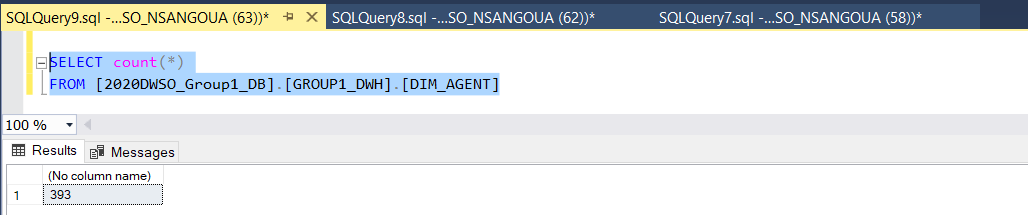
DIM\_CUSTOMERS:



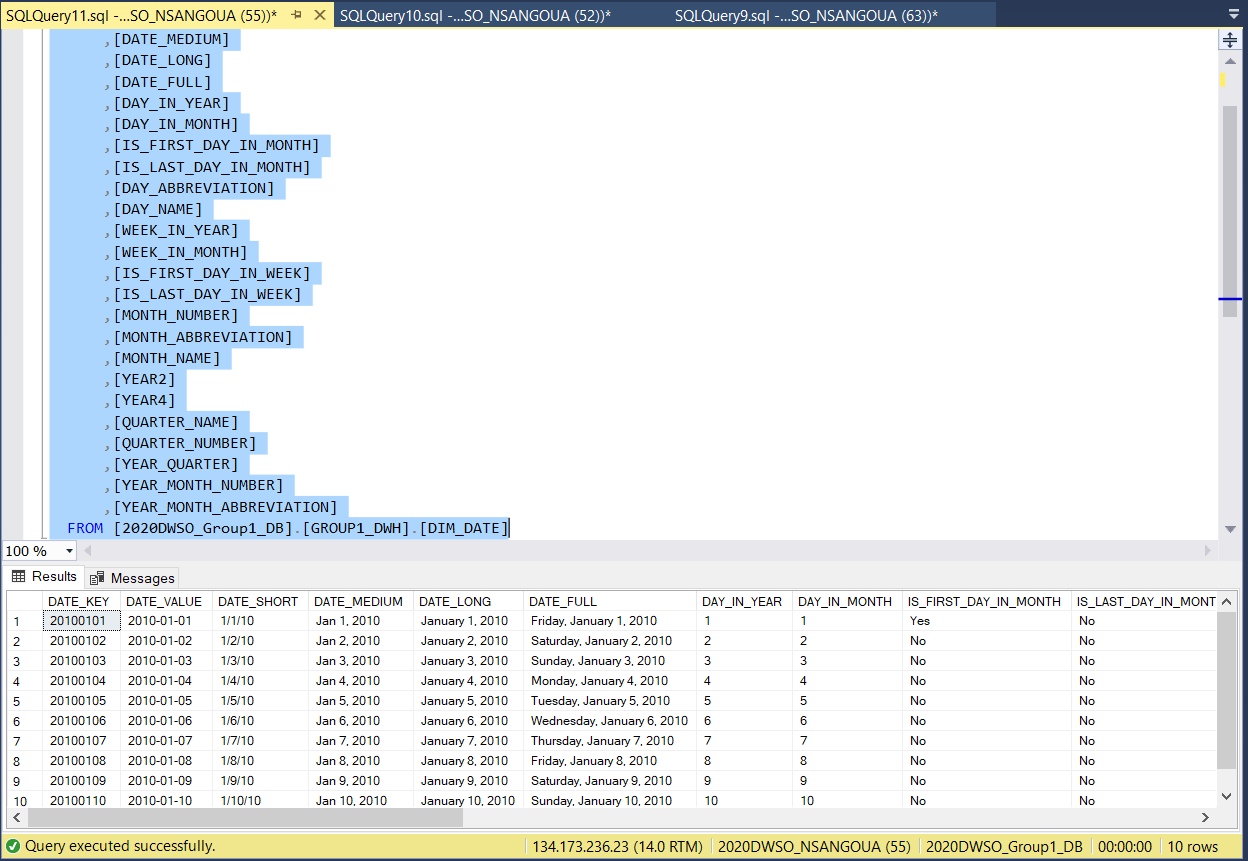


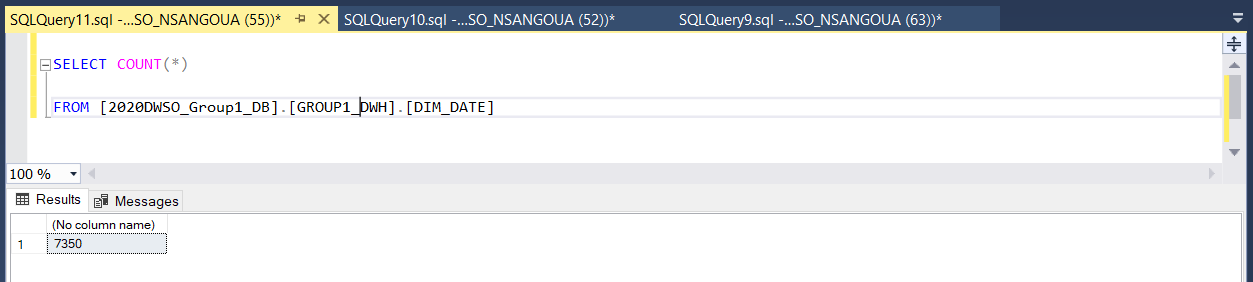
DIM\_AGENTS:



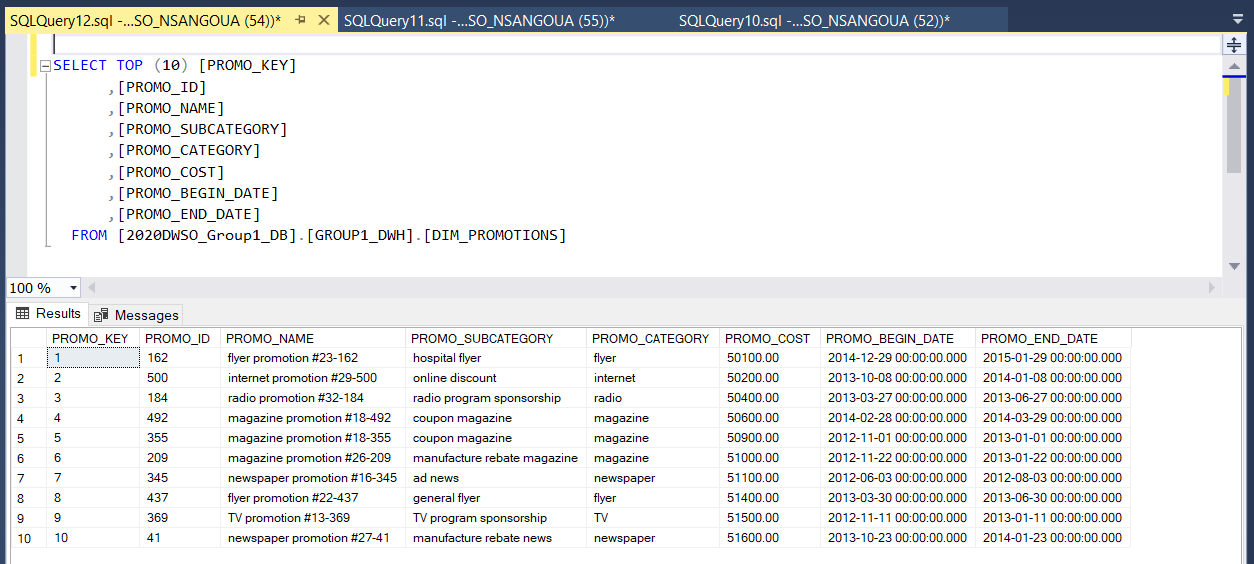


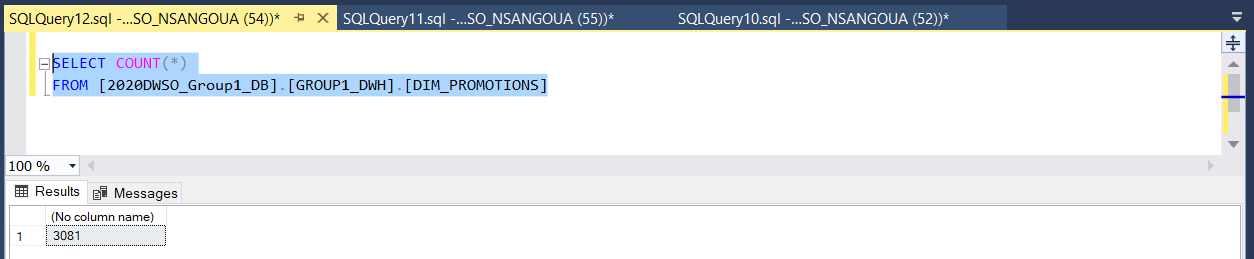
DIM\_DATE



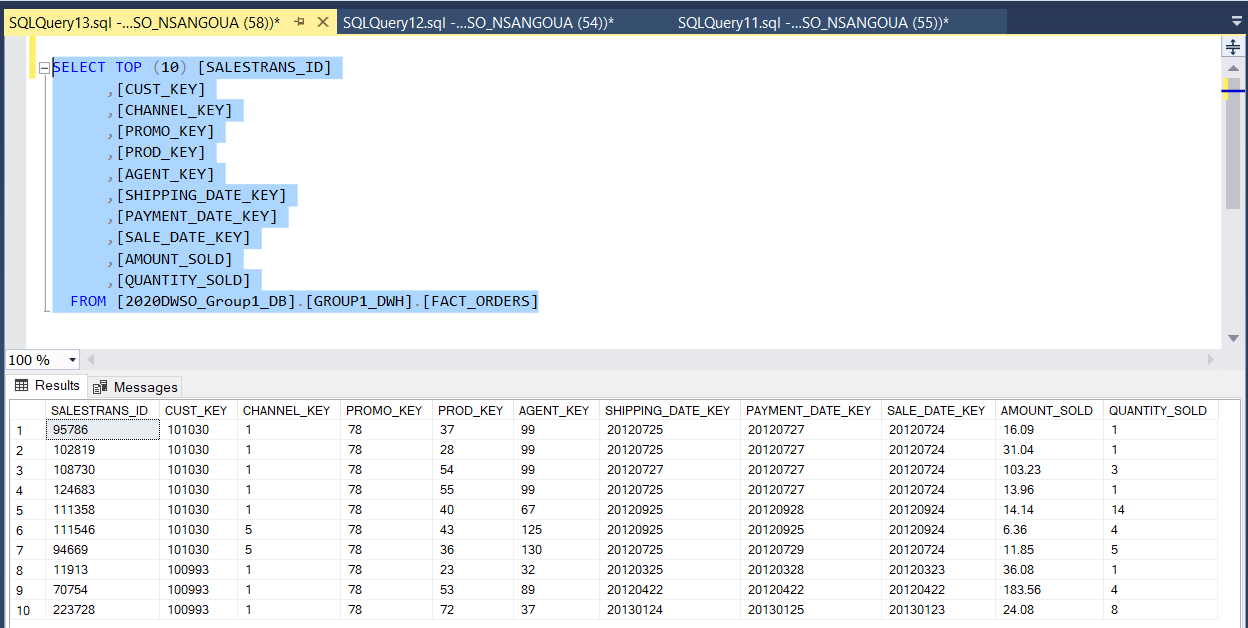


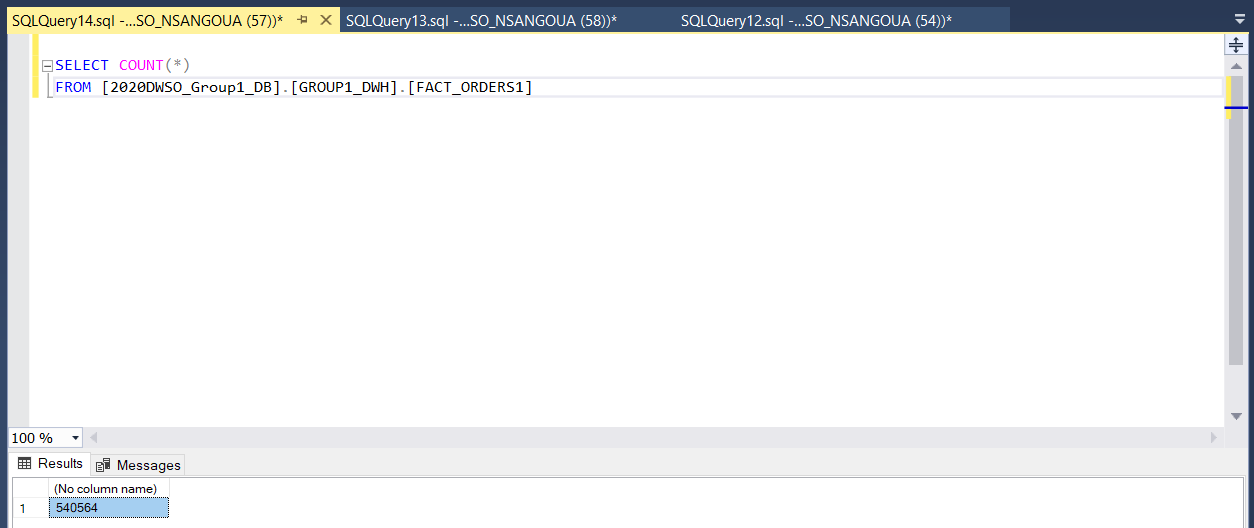
DIM\_PROMOTIONS





FACT\_ORDERS





1. **Physical Database Design (Part 2)**

For a subset of the Decision Support Views (DSVs):

1. **Identify queries that run the longest or use the most resources:**

|  |  |
| --- | --- |
| Transactions | Query statements |
| --- Sales per agent, agent experience, sale Transaction, sale Amount  Client Execution time: | SELECT D.YEAR4, D.YEAR\_MONTH\_NUMBER, A.AGENT\_EXPERIENCE\_LEVEL, SUM(O.AMOUNT\_SOLD), SUM(O.QUANTITY\_SOLD)  FROM GROUP1\_DWH.FACT\_ORDERS1 O JOIN GROUP1\_DWH.DIM\_AGENT A  ON O.AGENT\_KEY = A.AGENT\_KEY  JOIN GROUP1\_DWH.DIM\_DATE D  ON O.PAYMENT\_DATE\_KEY = D.DATE\_KEY  GROUP BY D.YEAR4, D.YEAR\_ The target data warehouse will be using roles-based access control to restrict the data access path. MONTH\_NUMBER, A.AGENT\_EXPERIENCE\_LEVEL  GO |
| --- Sales per product category, product sub category, product name, sale transaction, sale amount | SELECT D.YEAR4, D.YEAR\_MONTH\_NUMBER, P.PROD\_CATEGORY, P.PROD\_SUBCATEGORY, P.PROD\_NAME, SUM(O.AMOUNT\_SOLD), SUM(O.QUANTITY\_SOLD)  FROM GROUP1\_DWH.FACT\_ORDERS1 O JOIN GROUP1\_DWH.DIM\_PRODUCTS P  ON O.PROD\_KEY = P.PROD\_KEY  JOIN GROUP1\_DWH.DIM\_DATE D  ON O.PAYMENT\_DATE\_KEY = D.DATE\_KEY  GROUP BY D.YEAR4, D.YEAR\_MONTH\_NUMBER, P.PROD\_CATEGORY, P.PROD\_SUBCATEGORY, P.PROD\_NAME  GO |
| --- Sales per customer region, customer sub region, customer name, sale transaction, sale amount. | SELECT D.YEAR4, D.YEAR\_MONTH\_NUMBER, C.CUST\_REGION, C.CUST\_SUB\_REGION, C.CUST\_NAME, SUM(O.AMOUNT\_SOLD), SUM(O.QUANTITY\_SOLD)  FROM GROUP1\_DWH.FACT\_ORDERS1 O JOIN GROUP1\_DWH.DIM\_CUSTOMERS C  ON O.CUST\_KEY = C.CUST\_KEY  JOIN GROUP1\_DWH.DIM\_DATE D  ON O.PAYMENT\_DATE\_KEY = D.DATE\_KEY  GROUP BY D.YEAR4, D.YEAR\_MONTH\_NUMBER, C.CUST\_REGION, C.CUST\_SUB\_REGION, C.CUST\_NAME  GO |
| --- Orders per customer region, customer sub region, customer name, sale transaction, sale amount | SELECT D.YEAR4, D.YEAR\_MONTH\_NUMBER, C.CUST\_REGION, C.CUST\_SUB\_REGION, C.CUST\_NAME, SUM(O.AMOUNT\_SOLD), SUM(O.QUANTITY\_SOLD)  FROM GROUP1\_DWH.FACT\_ORDERS1 O JOIN GROUP1\_DWH.DIM\_CUSTOMERS C  ON O.CUST\_KEY = C.CUST\_KEY  JOIN GROUP1\_DWH.DIM\_DATE D  ON O.SALE\_DATE\_KEY = D.DATE\_KEY  GROUP BY D.YEAR4, D.YEAR\_MONTH\_NUMBER, C.CUST\_REGION, C.CUST\_SUB\_REGION, C.CUST\_NAME  GO |
| --- Sales per channel class, channel, sale transaction, sale amount | SELECT D.YEAR4, D.YEAR\_MONTH\_NUMBER, C.CHANNEL\_CLASS, SUM(O.AMOUNT\_SOLD), SUM(O.QUANTITY\_SOLD)  FROM GROUP1\_DWH.FACT\_ORDERS1 O JOIN GROUP1\_DWH.DIM\_CHANNELS C  ON O.CHANNEL\_KEY = C.CHANNEL\_KEY  JOIN GROUP1\_DWH.DIM\_DATE D  ON O.PAYMENT\_DATE\_KEY = D.DATE\_KEY  GROUP BY D.YEAR4, D.YEAR\_MONTH\_NUMBER, C.CHANNEL\_CLASS  GO |
| -- Total sales during the promotion | SELECT P.PROMO\_NAME, SUM(O.AMOUNT\_SOLD), SUM(O.QUANTITY\_SOLD)  FROM GROUP1\_DWH.FACT\_ORDERS1 O JOIN GROUP1\_DWH.DIM\_PROMOTIONS P  ON O.PROMO\_KEY = P.PROMO\_KEY  JOIN GROUP1\_DWH.DIM\_DATE D  ON O.PAYMENT\_DATE\_KEY = D.DATE\_KEY  WHERE D.DATE\_VALUE >= P.PROMO\_BEGIN\_DATE AND D.DATE\_VALUE <= P.PROMO\_END\_DATE  GROUP BY P.PROMO\_NAME  GO |

1. Provide SQL Scripts for generating indexes

CREATE CLUSTERED INDEX AGENT\_ID\_INDEX ON GROUP1\_DWH.DIM\_AGENT(AGENT\_ID)

CREATE CLUSTERED INDEX PROD\_ID\_INDEX ON GROUP1\_DWH.DIM\_PRODUCTS(PROD\_ID)

CREATE CLUSTERED INDEX PROMO\_ID\_INDEX ON GROUP1\_DWH.DIM\_PROMOTIONS(PROMO\_ID)

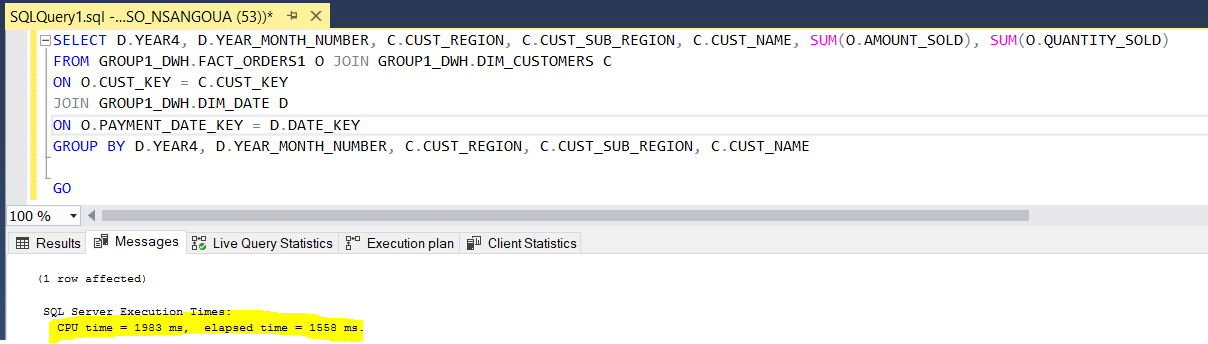
CREATE CLUSTERED INDEX CHANNEL\_ID\_INDEX ON GROUP1\_DWH.DIM\_CHANNELS(CHANNEL\_ID)

CREATE CLUSTERED INDEX CUST\_ID\_INDEX ON GROUP1\_DWH.DIM\_CUSTOMERS(CUST\_ID)

After creating the clustered indexes, the query performance improved as shown in the screen captured of our queries that run the longest:

Initial performance before indexes = CPU time: 2142 ms Elapse time: 1622 ms

Performance after indexes = CPU time: 1983 ms Elapse Time: 1558 ms

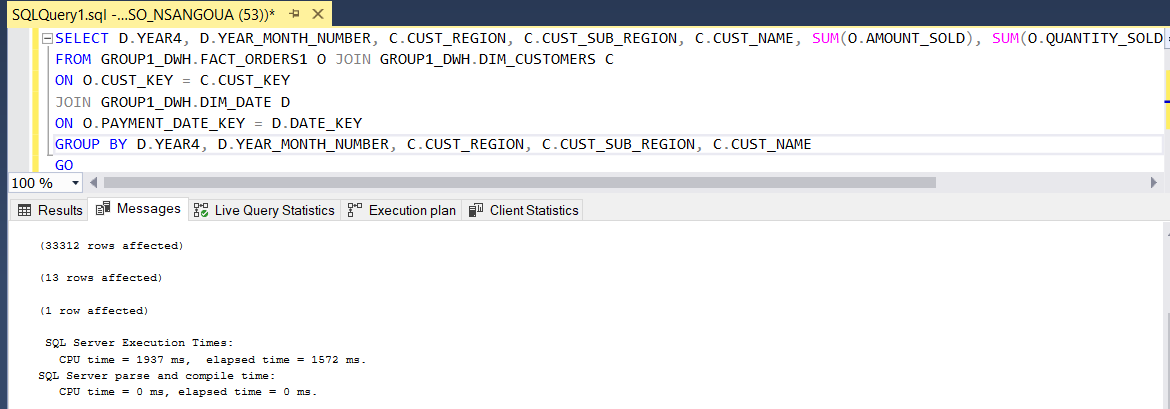


Next, we created a clustered index on Fact\_Orders below which also improved the query performance:

CREATE clustered INDEX Fact\_Orders1 ON GROUP1\_DWH.fact\_orders1(SALESTRANS\_ID, CUST\_KEY, CHANNEL\_KEY, PROMO\_KEY, AGENT\_KEY,

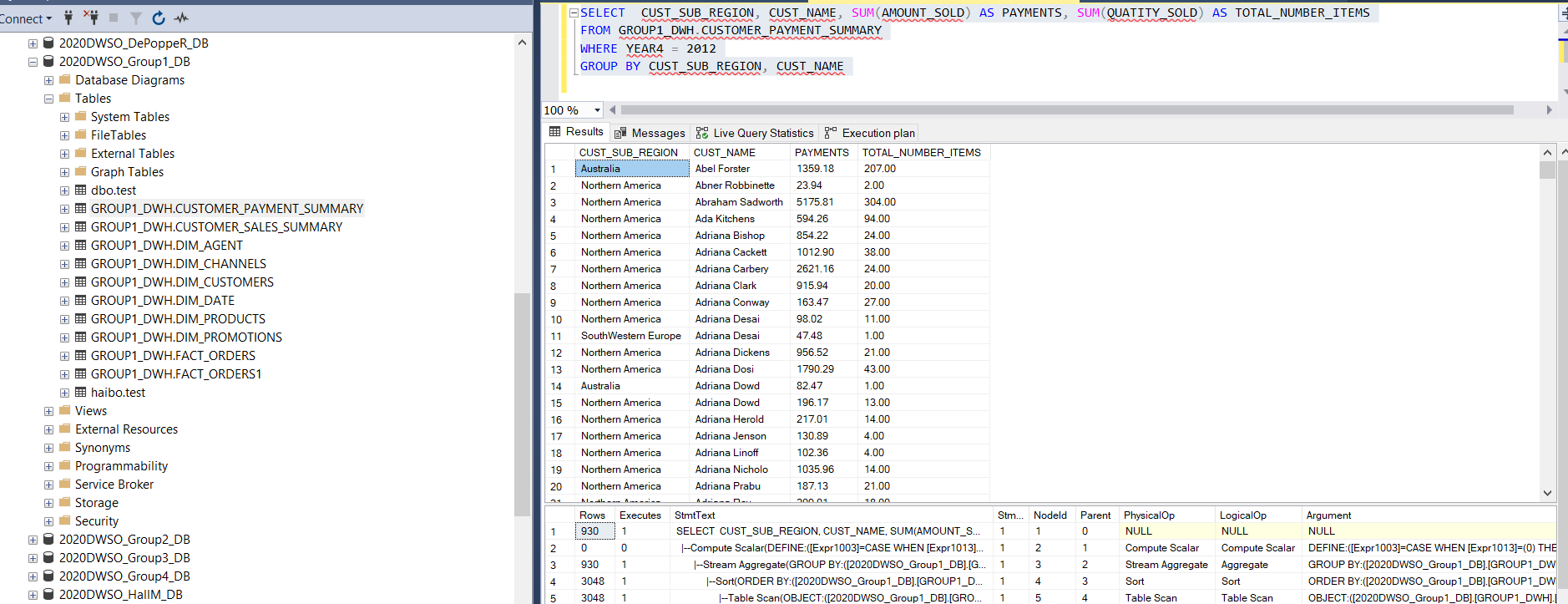
SHIPPING\_DATE\_KEY, PAYMENT\_DATE\_KEY, SALE\_DATE\_KEY, AMOUNT\_SOLD, QUANTITY\_SOLD)

Performance after indexes = CPU time: 1937 ms Elapse Time: 1572 ms

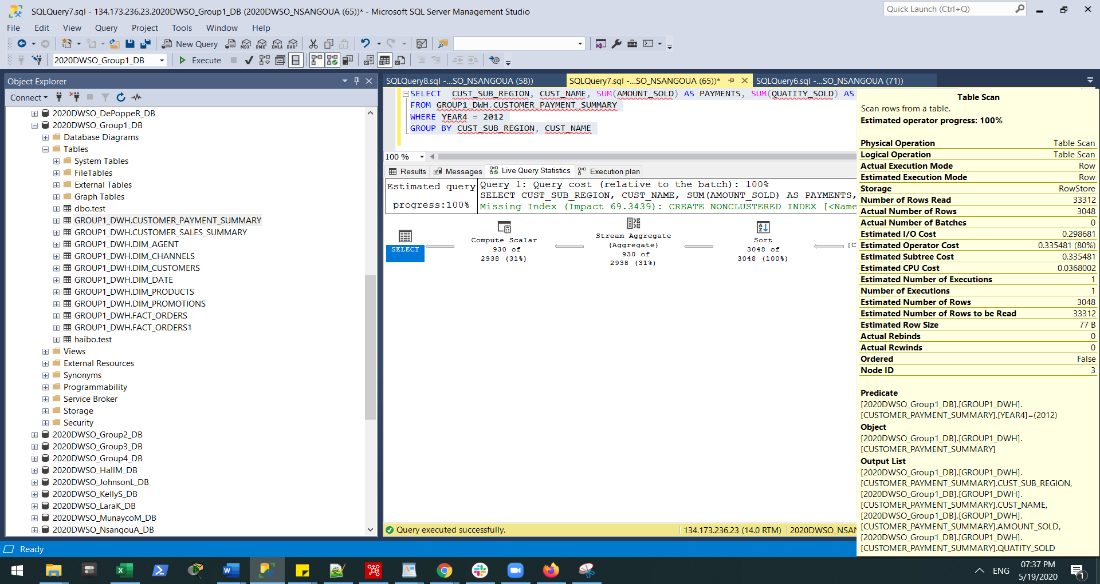


1. Tune the SQL statements, determine where summary tables and/or indexes are needed. Provide justification.

We believe that customers’ payments summary table “CUSTOMER\_PAYMENT\_SUMMARY” is a very interesting summary table that will growth very fast because the company core business is sales. For this reason, we have tuned up this table by adding both clustered and non-clustered indexes to all its columns as shown below:



Before tuning:



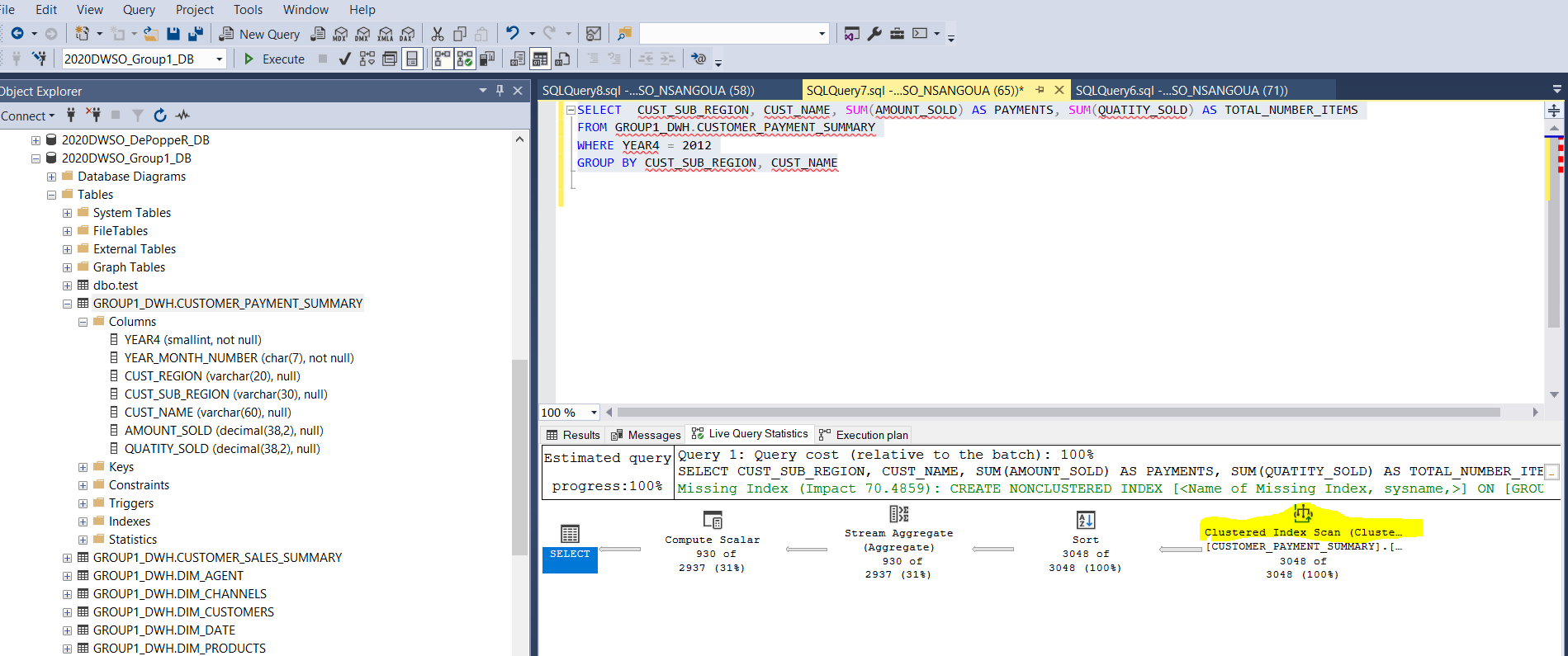
Indexes created for tuning:

CREATE CLUSTERED INDEX CI\_CUST\_NAME ON GROUP1\_DWH.CUSTOMER\_PAYMENT\_SUMMARY ([CUST\_NAME])

CREATE CLUSTERED INDEX CI\_CUST\_REGION ON GROUP1\_DWH.CUSTOMER\_PAYMENT\_SUMMARY ([CUST\_CUST\_REGION])

CREATE NONCLUSTERED INDEX NCI\_CUST\_NAME ON GROUP1\_DWH.CUSTOMER\_PAYMENT\_SUMMARY ([CUST\_NAME])

CREATE NONCLUSTERED INDEX NCI\_CUST\_REGION ON GROUP1\_DWH.CUSTOMER\_PAYMENT\_SUMMARY ([CUST\_REGION])



1. Estimate Storage Requirements for Summary tables. Provide justification.

Appendix 2: Estimate Storage Requirements for Dimensions and Fact table

1. Provide SQL Scripts for creating Summary tables & relevant indexes

Please, refer to Appendix 3: Decision Support Views, Su CUSTOMER\_SALES\_VIEW mmary Tables, and Data Understanding

1. **Query Management**
2. **Describe how user activities (e.g. create workbooks, access workbooks) will be controlled**.

As for the different data mining queries, different strategies will be used, as shown in the table below are:

|  |  |
| --- | --- |
| Query Type | Strategies |
| Prediction Queries | Using prediction query builder to start queries to start queries, let user customize the query by adding WHERE clause and other functions  Also, MDX can be used whenever there is any ad-hoc queries or very complex queries |
| Content Queries | DMV(dynamic management view) can be used to make a content query |
| Drillthrough Queries | Our model is not fully support drillthrough, MDX is using to create case data. |
| Data definition Queries | Initial data objects and all data source must be recorded into workbook. Also export/import tool will be using to restored or backup any existed model. Version management is needed to manage all exported object meta object |

Workbooks will be created on different type of queries, also include security information include role, view mapping, role priority, limitation of different role.

Priority setting from high to low for query from role as: CEO, Sales, Marketing, Product, Finance

1. **Describe how the execution of user queries will be scheduled**.

A query governing service will be placed between end user and our data warehouse to prevent overuse of data warehouse or long query that could exhausts the resource.

Limit must be set for query governing service as follows:

Number of minutes one query can run

Number of rows one query will return

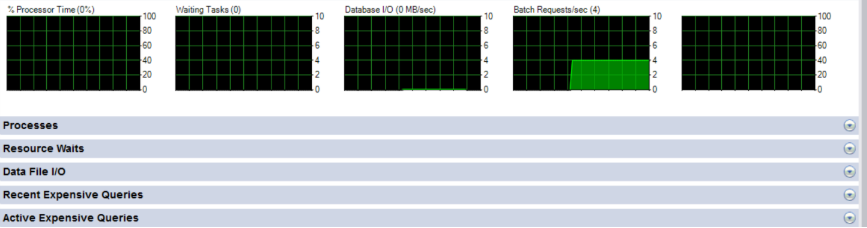
A database estimator will place before a query will be actually run

Only qualified users can run long query.

Only qualified user can schedule a SQL.

1. **Describe how query performance will be monitored**.

There are several solutions that will help to monitor query performance. Some of these solutions are built in or embedded tools that already exist in the platform that we use to manage our data. For instance, “**Activity Monitor** “, which is a built-in tool in SQL Server Management Studio as shown in the screen capture.



As depicted in the image, this tool will help the administrator to see in real time the following: Processor Time(%), Waiting Task(0), Database I/O (0MB/sec) and Batch Requests/sec(4). Based on this report, the database administrator can then figure out where performance tuning and query optimization are needed and quickly implement them. In addition, the size of the data increases you will advise further performance tuning solutions such as SOLARWINDS, which is a sophisticated database performance analyzer.

1. **For a subset of the Decision Support Views:**

* Provide narratives & snapshots that describe how the relevant workbook could be used to provide relevant information

CUSTOMER\_SALES\_VIEW is a view that acquiring total sales information based on customer, same view can be used to query total sales information based on region or sub region as well. Drillthrough can be happened apply to specific region name or sub region name.

For every user’s query, logging format like below will be recorded

Role, Start Timestamp, Running Time, Status, SQL, Application, Is Scheduled, Final Status

1. **Metadata**
2. **Data Extraction, Transformation & Load**

Concerning the ETL process, we have implemented a well-designed set of processes that will help to extract data from other various systems and files, transform, and load it into the data warehouse. This robust ETL pipeline is supported by Pentaho Data Integration tool, internal csv files, and other external tools not listed now that might be needed in the future.

1. **Query Management.**

Our query management approach would be able to satisfy basic functionalities of a query management system. It will help us to monitor and properly document both the “select query” and the “action query”. Furthermore, by select query, we are referring to all the retrieval queries. Whereas, by action query we mean query created for additional operations on the data such as data definition language (DDL) and data manipulation language (DML) queries that will help to sustain the data warehouse in the long run.

1. **Security Access**

Concerning the security access, the target data warehouse will be using roles-based access-control to restrict the data access path. This role-based access is determined based on different security levels depicted in the list of users table. Also, Only the database administrators would have full access to all the tables, views and summary tables.

Appendix 1: ETL Process Using Pentaho Data Integration

Appendix 2: Estimate Storage Requirements for Dimensions and Fact table

Appendix 3: Decision Support Views, Summary Tables, and Data Understanding