Cisco Specification Document

Date: 08.02.2024

Version: 3.0

Idan Gadasin

CONTENT

1 DOCUMENT MANAGEMENT

1.1 Project Scope & Objective

The project aims to create a comprehensive Business Intelligence (BI) solution for Cisco, utilizing the PriorityERP Database. It involves condensing key data tables, with a focus on sales, customer details, employee records, etc. The goal is to provide Cisco's leadership with visually represented data through dashboards and reports, facilitating efficient decision-making. The analysis of customer data aims to offer insights into preferences, purchasing behaviors, and product performance, helping Cisco shape targeted marketing campaigns and optimize production strategies.

1.2 Project Content

In this project, we will build a Data Mart that will include information about many aspects of a real-world business order detailing from end-to-end perspective.

1. Main summary tables that will be built for the company's demands:

    - **FactOrders** – Contains information about all the orders, including dates, pricing and quantity per customer from the transactions of the transactional database.
    - **DimEmployees** – Information about the company's employees**.**
    - **DimCustomers** – Information about the company's customers.
    - **DimProducts** - Information about the products that are sold by the company.
    - **DimProductsHistory -** Information about the products and the changes that occurred to the products.
    - **DimStores -** Information about the stores that took place during the transaction.

    Source to Target

2. The project will contain measures that will lead into action to the achievement of the project's goal:

    ✓ **Sales Department Analysis:**

    The Sales Department will concentrate on data related to sales activities, including the order timestamp, customer details, and the initiating agent. The department's primary focus is to distinct trends in sales, considering factors such as product category and store type. A comparative analysis with the previous year's results will be conducted to provide a comprehensive view of goal achievement. The department routinely collects information with each new order entry into the system, assisting users in monitoring ongoing transactions. Employee performance will be evaluated based on the top 5 most beneficial agents based on total sales, year-over-year sales variation, and the popularity of the brands they promote. Additionally, the top 10 best-selling brands will be highlighted, showcasing total sales categorized by stores and locations. To derive future business insights and implement strategic actions for business growth, a detailed analysis of sales data by store type will be performed.

✓ **Customer Department Analysis:**

The company's customers will have access to a range of data related to their purchases and overall interactions with the brand. The available information includes total sales, sales categorized by country and store, top-performing brands, quantities of products sold by country and store, order counts per country, customer counts by country. By leveraging this information, customers can make well-informed choices for their future purchases and deepen their understanding of the diverse array of products offered by the company.

✓ **Executive Dashboard Analysis:**

The executive department will concentrate on overall data which related to the highest scope of the organization, including merged data between products, employees, sales and more measurable information as: amount of products that were sold, most profitable products to the company, comparing between income revenues between store types, most profitable countries by sales and compare the revenue of the company in comparison with the previous year.

2 Strategic Work Plan: Task Allocation & Gantt Scheduling

Gantt Scheduling

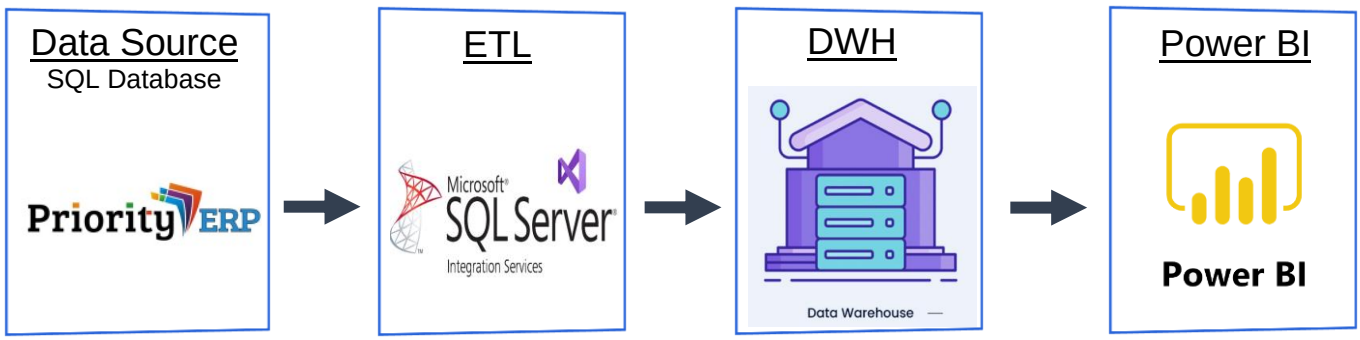
3 Technical Specifications

3.1 Prerequisites – the involved systems during the process

System / Process	Explanation
SQL Server	Operational DB – tables – data (SQL Files)
SSIS	ETL processes using SSIS in Visual Studio All the files are located at: C:\Users\nulla\Desktop\Career + Education\Career\Experis\Bootcamp\ פרויקט גמר\e2e prod cisco sales project
Data Refreshing	Refreshing processes through an attribute of Employees in SSMS
Power BI	Creating reports & dashboards for Power BI usage

3.2 Solution Architecture

HLD:



Data collection and exploration from the ERP system will be performed in SQL Server. The data will flow through ETL process for organization and arrangement into a Data Mart using SSIS. Finally, the presentation of measures through reports and visuals will be presented in Power BI.

The report for the Customer department will include:

- Behavioral purchasing preferences among customers (online vs physical purchasing).
- Average price per order.
- Distribution of the ratio of purchases per country vs total purchases .
- Number of products per country.
- Amount of customers.

The report for the Employees department will include:

- Top 5 most beneficial agents based on total sales.
- Average sales price per employee \ all employees.
- Sales difference rate with previous year \ month.
- Employees rank based on his selling results.

The report for the Management department will include:

- Top 10 bestselling products on total income.
- Number of products that were sold.
- Income revenue of regular store compared to online store.
- Income revenue in comparison to the previous year \ month via drill down.
- KPI Reports that include trends like customers change, order amount change, etc.

4 Functional Specification

4.1 ERD model

ERD Link

4.2 Detailed description of the ETL Process

- In SSIS, I have 13 packages for the required tables.

1. FactOrders Table:

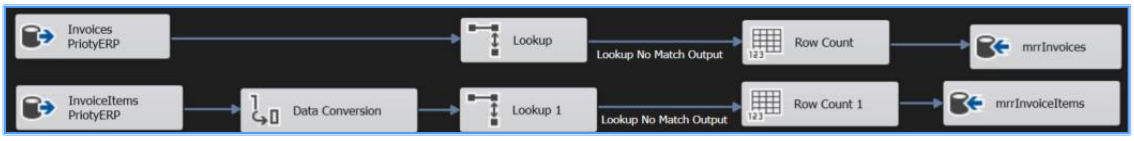
- **FactOrders**:
  The source is StgOrders table, after loading the source, the process converts the data of some columns to match the types of the loaded data and adds a variable that counts the amount of rows. The information inside the table is transferred to 'FactOrders'

  After that the records of: Package name, tablename, StartDate, EndDate, the amount of transferred rows are inserted inside the table 'Transfertable'

- **StgOrders**:
  Creating a table 'stgOrders' that includes a united results of the SQL query that joins between the different tables into our desired result. After th at we count the amount of the transferred rows. After that the data will be loaded inside 'stgOrders'

  Every time the package runs, all the previous records from stgOrders will be deleted, ONLY the new records will be inserted inside 'StgOrders' from the mrr tables. After that the records of: Package name, tablename, StartDate, EndDate, the amount of transferred rows are inserted inside the table 'Transfertable'

- **mrrOrders**:
  Transferring the Tables 'INVOICES', 'INVOICEITEMS' into mrr tables.
  The load action will load ONLY new records that did not appear in the 'FactOrders' table before to avoid a case that we load duplicated data. All the new records will be counted.



  Every time the package runs, all the previous records from mrrINVOICES,mrrINVOICEITEMS will be deleted, ONLY the new records will be inserted inside mrrINVOICES,mrrINVOICEITEMS. After that the records of: Package name, tablename,Start_Date, End_Date, the amount of transferred rows are inserted inside the table 'Transfertable'

2. DimEmployees Table:

DimEmployees:
Loading all the data from StgEmployees, the data will be loaded according to the Upsert Merge Command that will load & handle the data in few scenarios: 1. Load New Data, 2. Update Existing data that had changes in the Online Transaction processing Tables (EMPLOYEES). 3. Data that has been deleted from the Online Transaction processing.

- **StgEmployees**:
Creating a table 'stgEmployees' that includes a united results of the SQL query that joins between the different tables into our desired result. Next, we convert column datatypes to match those of the destination and lastly: count the amount of the transferred rows. After that the data will be loaded inside 'stgEmployees'

Every time the package runs, all the previous records from stgEmployees will be deleted, ONLY the new records will be inserted inside ' stgEmployees ' from the mrr tables. After that the records of: Package name, tablename, StartDate,EndDate, the amount of transferred rows are inserted inside the table 'Transfertable'

3. DimCustomers Table:

- **DimCustomers:**
Loading all the data from StgCustomers, , the data will be loaded according to the Upsert Merge Command that will load & handle the data in few scenarios: 1. Load New Data, 2. Update Existing data that had changes in the Online Transaction processing Tables (CUSTOMERS). 3. Data that has been deleted from the Online Transaction processing. After Loading the 'dimCustomers' table, the records of: Package name, tablename, Start_Date,EndDate, the amount of transferred rows are inserted inside the table 'Transfertable'

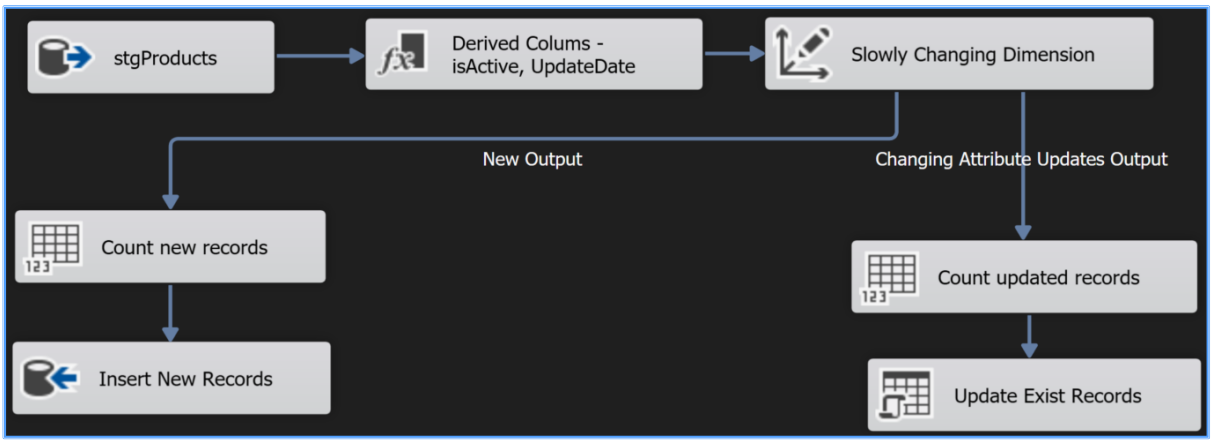
- **StgCustomers:**
Creating a table 'stgCustomers' that includes a united results of the SQL query that joins between the different tables into our desired result. Next, we convert column datatypes to match those of the destination and lastly: count the amount of the transferred rows. After that the data will be loaded inside 'stgCustomers'

the new records will be inserted inside ' stgCustomers ' from the mrr tables. After that the records of: Package name, tablename, StartDate,EndDate, the amount of transferred rows are inserted inside the table 'Transfertable'

4. DimProducts Table:

- **DimProducts:**
  Loading all the data from StgProducts step by step, Only records that had changes (New Records, Updated Records or Deleted records) will be loaded inside 'DimProducts' table, after that the process will create the updated dimProducts.
  In the first step, we deal with records that have been deleted from the source(PART) and appear in the target (dimProducts).

After that, the flow creates 2 derived columns that adjust the updated date and the isActive status of the product, Next - all the new data or the updated data inside the Target (dimProducts)



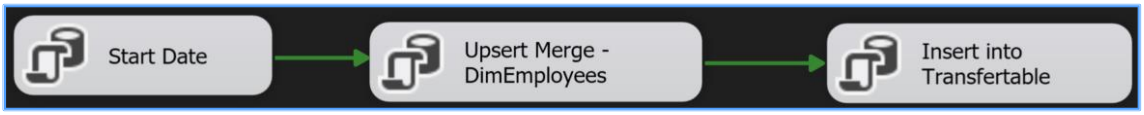
After Loading the 'dimProducts' table, the records of: Package name, tablename, Date, the amount of transferred rows are inserted inside the table 'Transfertable'

- **StgProducts:**
  Creating a table 'stgProducts' that includes a united results of the SQL query that joins between the different tables into our desired result. Next, we convert column datatypes to match those of the destination and lastly: count the amount of the transferred rows. After that the data will be loaded inside 'stgProducts'

Every time the package runs, all the previous records from stgProducts will be deleted, ONLY the new records will be inserted inside ' stgProducts ' from the mrr tables. After that the records of: Package name, tablename, StartDate,EndDate, the amount of transferred rows are inserted inside the table 'Transfertable'. The package adjusts some product details.

5. DimStores Table:

- **DimStores:**
  Loading all the data from mrrStores , the data will be loaded according to the Upsert Merge Command that will load & handle the data in few scenarios: 1. Load New Data, 2. Update Existing data that had changes in the Online Transaction processing Tables (BRANCHES). 3. Data that has been deleted from the Online Transaction processing. After Loading the 'dimStores' table, the records of: Package name, tablename, StartDate,EndDate, the amount of transferred rows are inserted inside the table 'Transfertable'



- **StgStores:**
  Creating a table 'stgStores' that includes a united results of the SQL query that joins between the different tables into our desired result. Next, we convert column datatypes to match those of the destination and lastly: count the amount of the transferred rows. After that the data will be loaded inside 'stgStores'

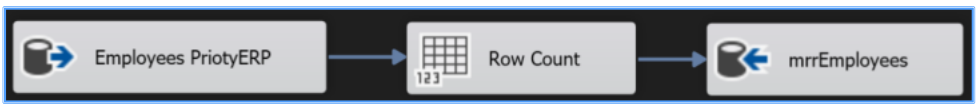
After Loading the 'stgStores' table, the records of: Package name, tablename, StartDate,EndDate, the amount of transferred rows are inserted inside the table 'Transfertable'

6. MrrDimTables:

Note: *All the mrr tables were created in a single package, they all share the same control flow, this is why it is displayed here and not in the dim tables explanation.*
All the tables are created via data flow that creates each mrr table and counts how many rows were transferred in the transaction.

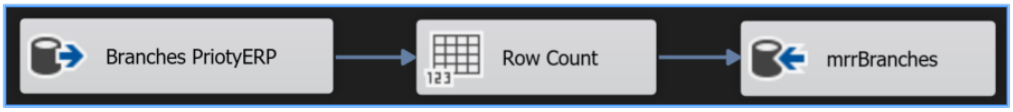
mrrEmployees:

Loading the source tables from the Online Transaction Processing tables that related to employees into new mrr tables in our database, after loading each mrr table, the amount of the transferred rows will be counted to check how many rows transferred and recognize what happened during the process.



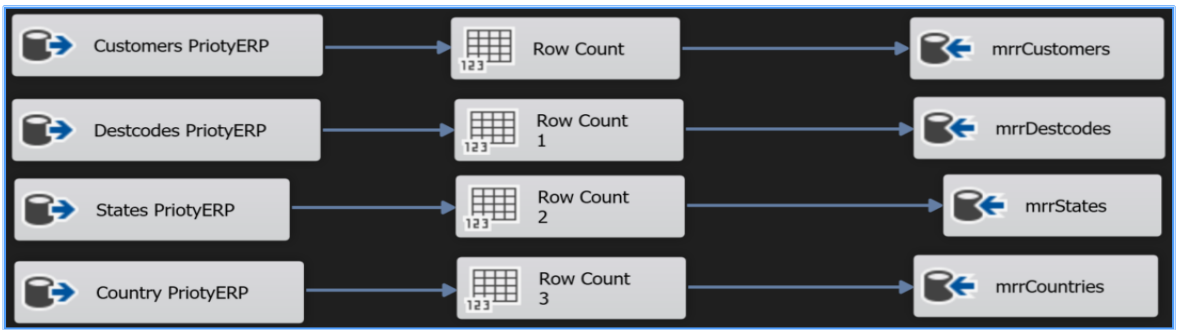
mrrStores:

Loading the source tables from the Online Transaction Processing tables that related to stores into new mrr tables in our database, after loading each mrr table, the amount of the transferred rows will be counted to check how many rows transferred and recognize what happened during the process.



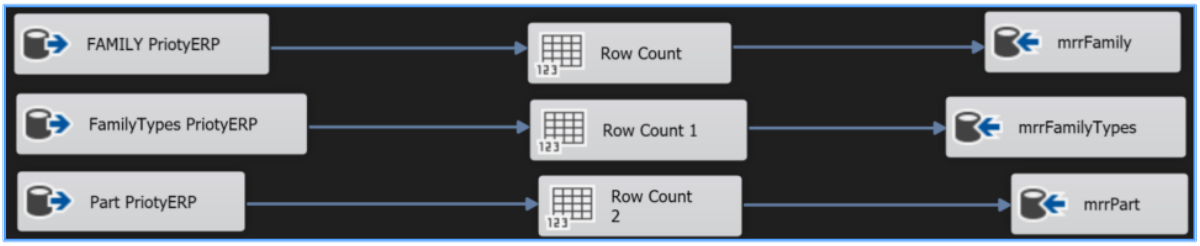
mrrCustomers:

Loading the source tables from the Online Transaction Processing tables that related to customers into new mrr tables in our database, after loading each mrr table, the amount of the transferred rows will be counted to check how many rows transferred and recognize what happened during the process.



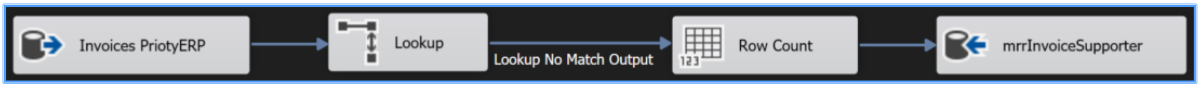
mrrProducts:

Loading the source tables from the Online Transaction Processing tables that related to products into new mrr tables in our database. ONLY relevant products will be loaded into mrrProducts table based on business rule (Active product is a product that has been purchases in the last 2 years). New data is data that has been modified \ updated after the last date that appears on dimProducts
After loading each mrr table, the amount of the transferred rows will be counted to check how many rows transferred and recognize what happened during the process.



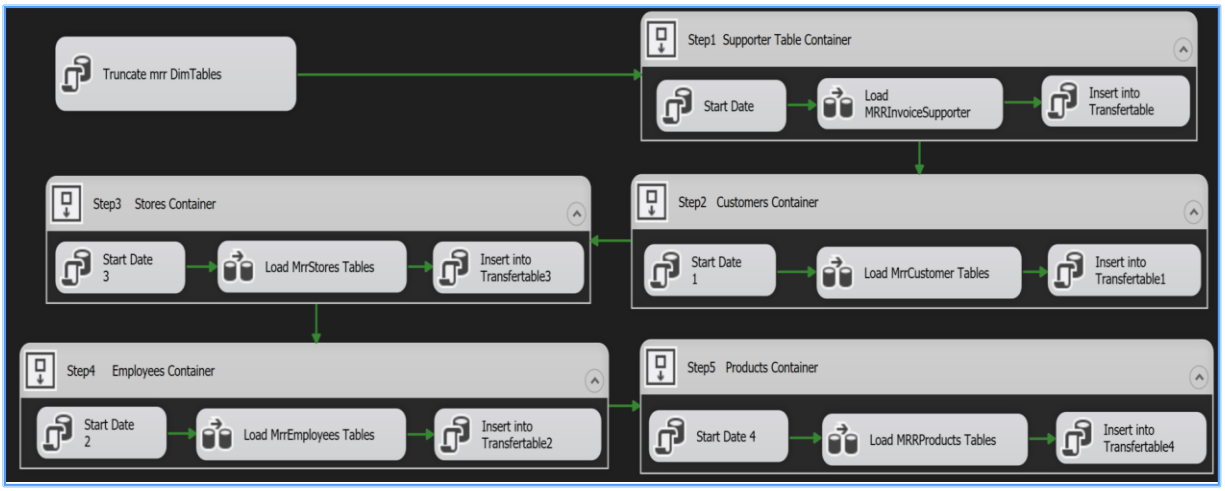
mrrInvoiceSupporter:

A table that will help to handle future table merging that uses the INVOICES table in addition to the original table. Loading the source tables from the Online Transaction Processing tables that related to products into new mrr tables in our database. ONLY new data will be loaded into mrrInvoiceSupporter table.
After loading each mrr table, the amount of the transferred rows will be counted to check how many rows transferred and recognize what happened during the process.



<u>The flow of the mrr tables:</u>

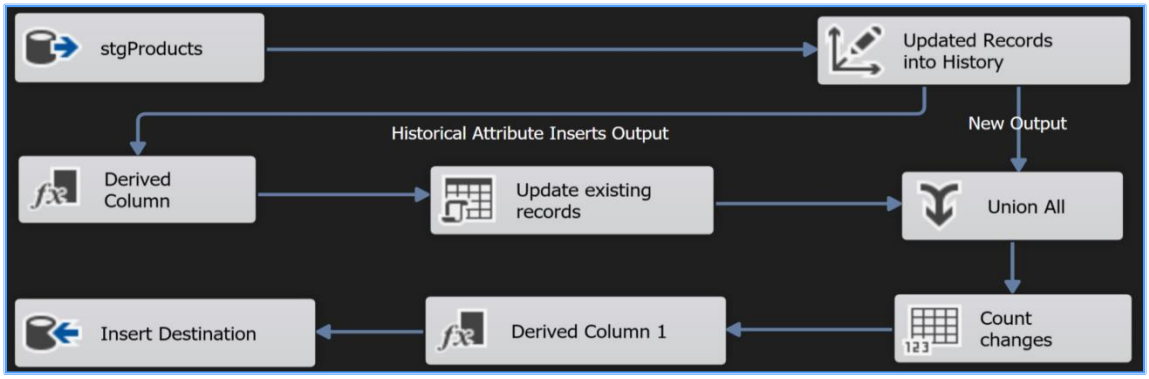
Before every load of the mrr tables, we truncate them, load each table – starting with mrrInvoiceSupporter table, insert into transfertable the data about the transaction, moving to mrrCustomers, insert into transfertable the data about the transaction, loading mrrStores tables, moving to mrrEmployees, insert into transfertable the data about the transaction, moving to mrrProducts, insert into transfertable the data about the transaction.



7. DimProductsHistory:

- **DimProductsHistory:**

  The table contains all the data that is inside DimProducts and in addition the change that happened per product (Update, Deletion) or insert new record (means no previous history for the specific product). The information about each record will be given from the status of the record in the 'StgProducts' table and it will be compared to the history table that will make the required changes.



The table will be loaded in data according to the given description before.

After that the package will handle records that have been deleted from the source and exist inside the DimProducts

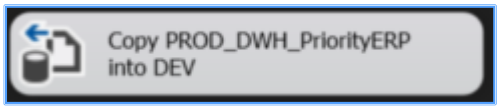
8. Transfertable Table:

- **Transfertable:**

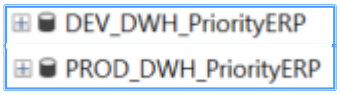
The table itself updates itself during the Control Flow of the previous screen shots that were displayed in the file.

The packages that included an 'EXECUTE SQL COMMAND' of insert are:
MRR for Dim Tables, MRR for Fact, All the MRR, Stg, Dim tables are being inserted into TransferTable transaction after transaction automatically.

The next step is to copy the database into development environment
Via SSIS Task that copies the database into a new one that we can enter a "sandbox" environment which we can try to manipulate the data in many various methods:



The environments that the jobs will be applied to:



There are 4 important stages in the job task: 1. Mrr ETL, 2. Staging ETL, 3. Dim ETL, 4. DWH ETL

The next step is to deploy the Project

The deployed path : /SSISDB/Cisco DWH Project/[PackageName].
After that I create a job that called 'Full ETL Process Job' which will load every job all the jobs will be done.
The jobs are separated into small parts because if an error will occur, we want to detect it in the specific job and be more efficient instead of loading massive amount of data each time.

There are 4 important stages in the job task: 1. Mrr ETL, 2. Staging ETL, 3. Dim ETL, 4. DWH ETL

This is the chronological order of loading the tables:
Mrr Tables ⟶ Employees Tables ⟶ Products Tables ⟶ Store Tables ⟶ Customers Tables ⟶ DWH Orders Tables

1. MRR ETL:

Step	Name	Type	On Su...	On Fail...
1	Load MRR Orders	SQL S...	Go to t...	Quit th...
2	Load MRR Dim Tables	SQL S...	Go to t...	Quit th...

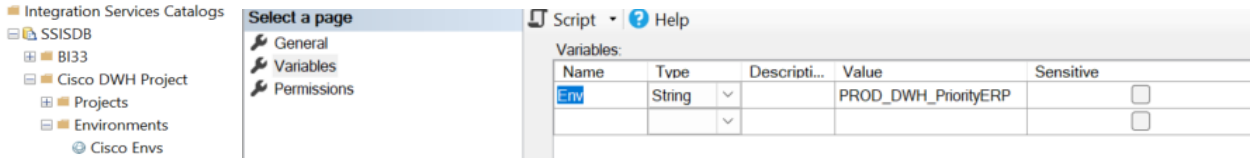
2 + 3. Staging ETL + Dim ETL:

3	Load Stg Employees	SQL S...	Go to t...	Quit th...
4	Load dimEmployees	SQL S...	Go to t...	Quit th...
5	Load stgProducts	SQL S...	Go to t...	Quit th...
6	Load dimProducts	SQL S...	Go to t...	Quit th...
7	Load dimProductHistory	SQL S...	Go to t...	Quit th...
8	Load stgStores	SQL S...	Go to t...	Quit th...
9	Load dimStores	SQL S...	Go to t...	Quit th...
10	Load stgCustomers	SQL S...	Go to t...	Quit th...
11	Load dimCustomers	SQL S...	Go to t...	Quit th...

4. DWH ETL:

12	Load stgOrders	SQL S...	Go to t...	Quit th...
13	Load DWH Orders	SQL S...	Quit th...	Quit th...

After planning the job stages I tested my jobs on the DEV_DWH_Priority environment to check the data quality and conduct testing. Once the data integrity works well and all the packages works via SQL Agent we switch the Environment from Development to Production.



At this stage the jobs are estimated to work on the Production environment.
In the last stage we automate the jobs to work by a schedule:

All the Extract-Transform-Load process will happen every day at 02:00 AM.

ID	Name	Enabled	Description	Jobs in Schedule
20	Daily ETL Loading	Yes	Occurs every day at 2:00:00 AM. Schedule will...	View

5 Power BI Reports & Visualization

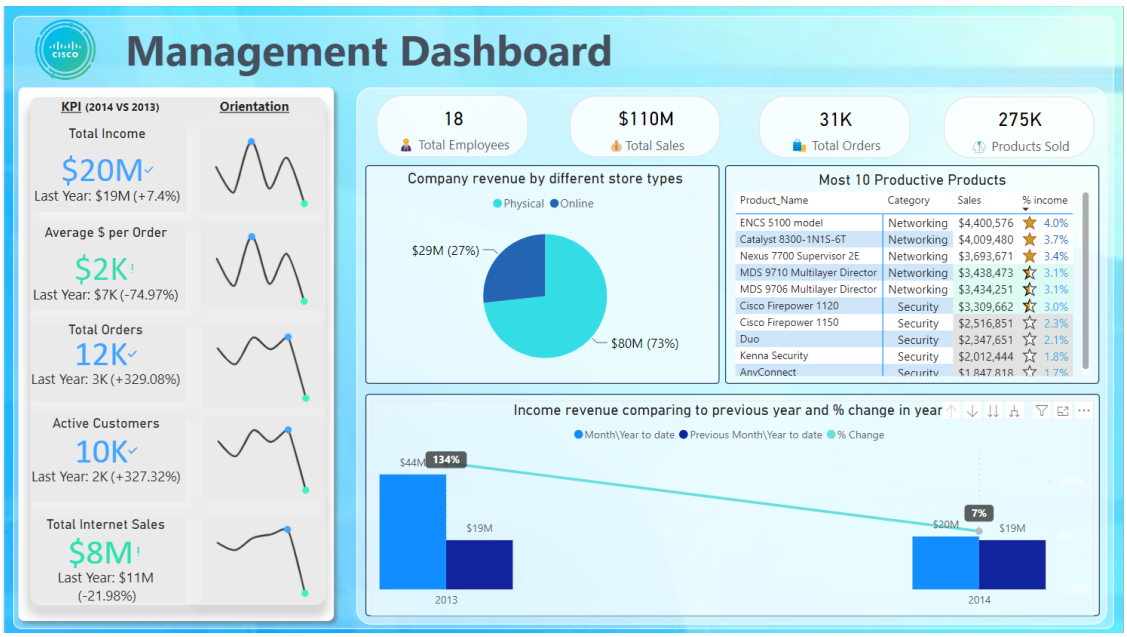
Management Dashboard:

This dashboard provides a summarized overview about Cisco's international sales department, including various information as:

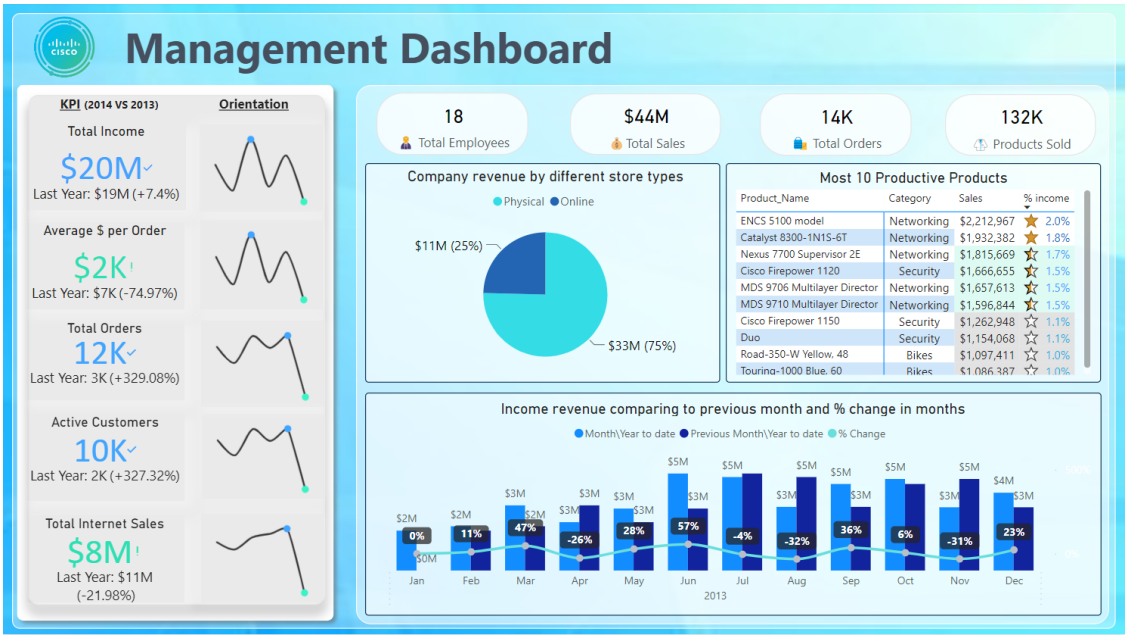
- Summarized information as: total employees , total sales, total orders and amount of products sold.
- KPI measures that may help with the analysis and identification of new trends in many aspects of the organization.
- Most 10 productive products based on the total sales per product.
- Display revenue of the company divided to store types to identify a behavioural pattern.
- Income revenue comparison in a drill-down from the last year to the previous year.
  The graph is built in a hierarchy to explore the revenue comparison into months.

The dashboard has 2 modes :

1. Income revenue and % change by year:



2. Income revenue and % change by month (after drill down):



Employees Report:

This report provides a summarized overview about Cisco's international agents department, including various information as:

- Summarized information as: total employees , total sales, total orders and amount of products sold.
- Top 5 most productive agents based on the total sales per product.
- Display agent's details like average sales $, his rank as a salesperson, how many he sold from the total sales.
- Income revenue % change comparison in a drill-down from the last year to the previous year. The graph is built in a hierarchy to explore the revenue comparison into months.

The report has 2 modes :

1. Income revenue % change by year:

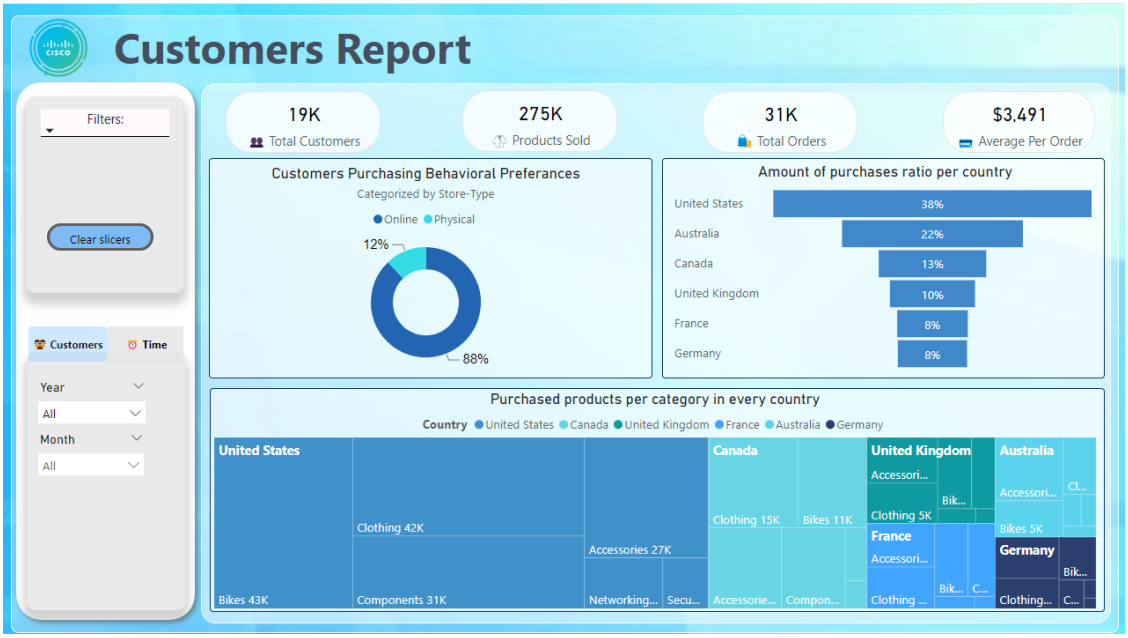
2. Income revenue % change by month(after drill down):

The report has a slicer on the left side that lets the user to filter the report according to his research (via employee details or time-line details). Moreover – once the user selected a filter, it will be displayed on the top of the panel.

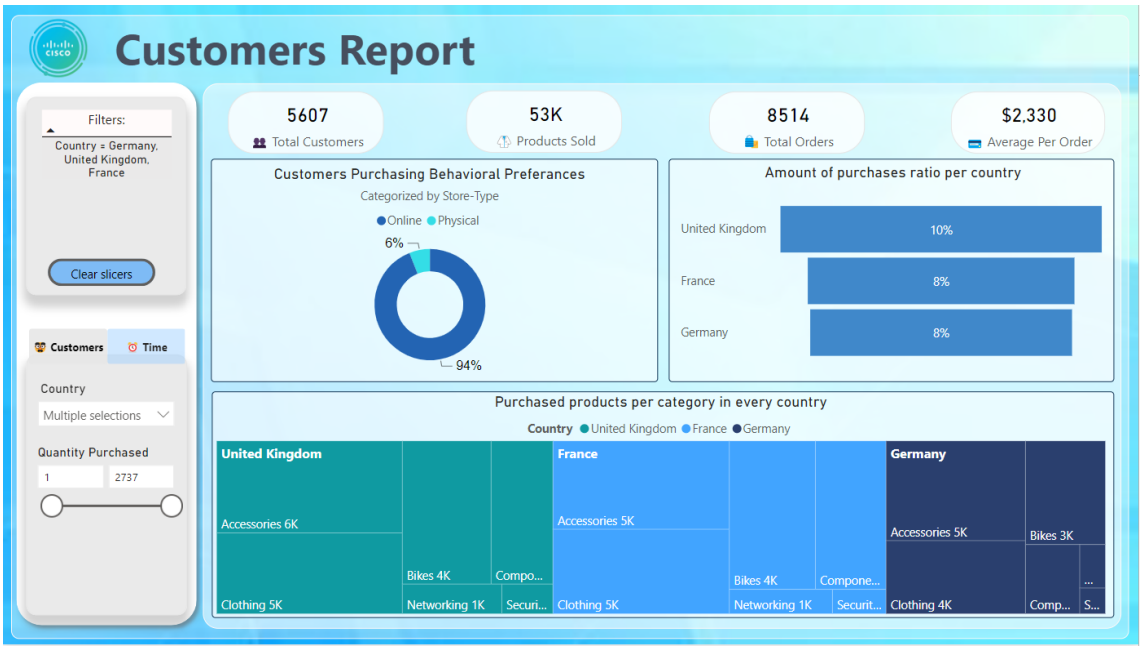
Customers Report:

This report provides a summarized overview about Cisco's international customers service department, including various information as:

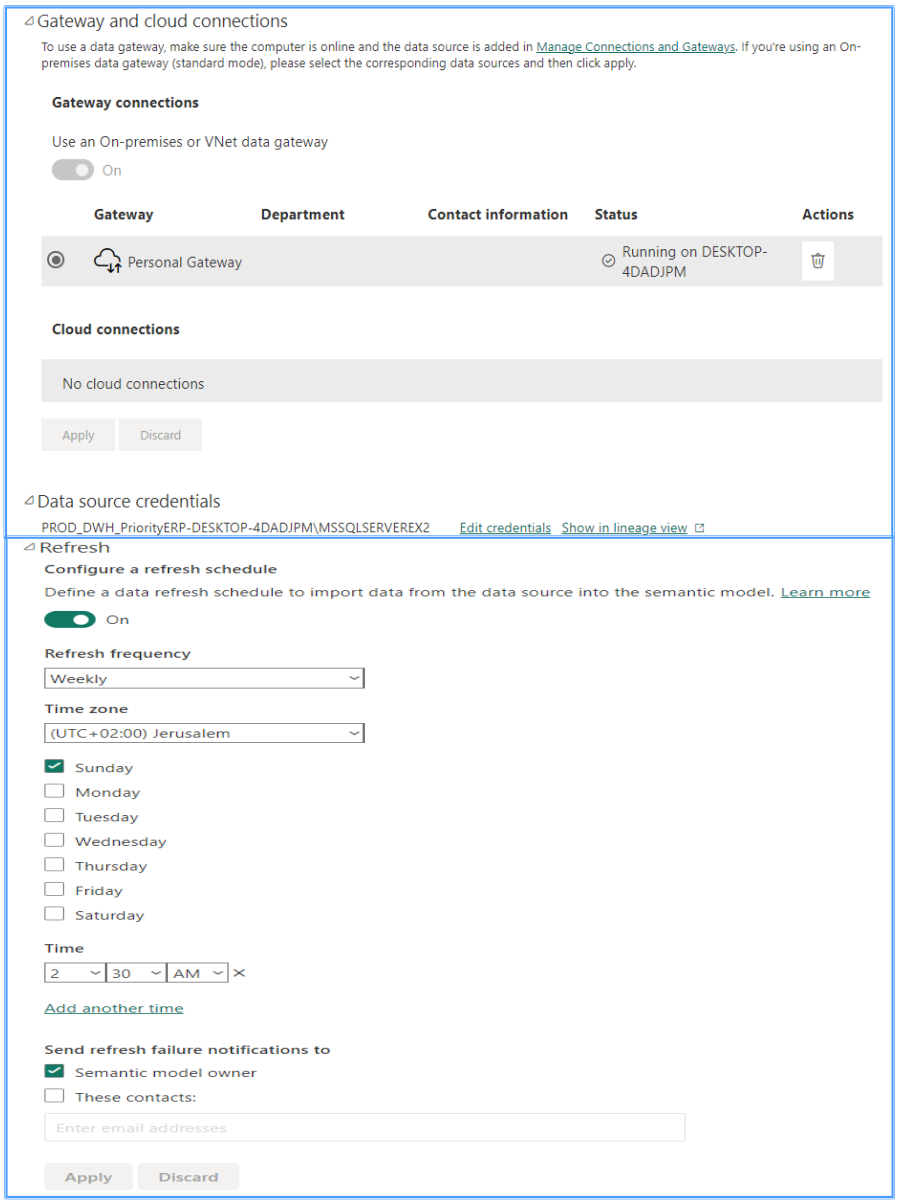
- Summarized information as: total customers , total sales, total orders and amount of products sold.
- Distribution of the customers purchase ratio grouped by country – how much each country benefits to the total income.
- Display the amount of products that were sold by every country divided to categories.
- Analysing behavioural purchasing habits – Online vs Physical purchasing



The report has a slicer on the left side that lets the user to filter the report according to his research (via customers details or time-line details). Moreover – once the user selected a filter, it will be displayed on the top of the panel.



The next step is to publish the pbix file, activate the Gate away data-source and automate the refresh processing of the dashboard:



Since the ETL process scheduled to 02:00 AM on Sunday – the report refresh will be refreshed few minutes later since the jobs take some time and the systems need to be synchronized together.

Lastly, I created an app called 'Cisco Visualization Project':

6 Appendix

1. Link to the app
2. Link to github project