AudioCodes's Sales Data Mart

BI System Specifications Document

Written by: Gil Emanuel

Content

1. **General**

1.1. Project Objective:

This project's objective is the creation of full BI solution for AudioCodes's sales department, to support AudioCodes's growth in products sales. The project was designed according to AudioCodes's sales department KPIs and is aimed at increasing the company's overall ROI.

AudioCodes is a leading provider of communication software, products and productivity solutions for the digital workplace. AudioCodes enables enterprises and service providers to built and operate all-IP voice networks for delivering unified communications, contact centers, and hosted business services, whether in the cloud or on premises. AudioCodes's products are used by large multinational enterprises and leading tier one operators worldwide. AudioCodes's underlying high-definition voice technology and products provide enhanced quality and an improved end user communications experience.
This project will focus only on sales performed by AudioCodes.

The Data Mart creation will be done using information derived from the PriorityERP database (AudioCodes's operational database). The solution will include summarized data tables, focusing on products sales data, as well as data regarding AudioCodes's customers, salespersons, products, and stores. In addition, the BI solution will include costumed reports containing sales analysis, customer analysis, and executive dashboard. These reports will be tailored for the sales departments' needs and will contribute to AudioCodes's products sales growth.

1.2. Project contents:
The project includes the building of a Data Mart which contains sales data. The data will be transferred through an ETL process from the PriorityERP operational database to the Data Mart – AudioCodes_DEV.
ERD model of the AudioCodes_DEV database: ERD Link.

1.2.1. The data mart will include 1 fact table and 4 dimension tables, and 1 history table:

- Fact_Sales – Data regrading all sales, including the ID of the order, products bought, quantities, and prices. Data loading process for this table will be incremental.
- Dim_Customers – Data regarding the company's customers.
- Dim_Stores – Data regarding the company's Stores that sale the company's products.
- Dim_Employees – Data regarding the store's employees
- Dim_Products – Data regarding the company's products.
- Dim_StoresrsHistory - Historic data regarding the company's stores.

Source To Target Link.

The tables will be updated daily at 04:00:00 using an automated process configured in the SQL Server Management Studio.

1.2.2. The reports will include data visualizations that will support the project's objective in the following ways

- Sales Analysis:
  The sales report will include data about sales (revenue, number of orders, and number of units) by date, country, product, store (online vs. physical), and salespersons which will help the department to assess the performance of all the parts needed for sales growth.

The reports will help to identify sale trends like seasonality and trending product categories, analyze products orders and revenue, spot top performing salespersons, and analyze the differences in behavior between the online store and physical stores. All of these will support data driven strategic decision making which can lead to growth in sales and revenue.

- Customer analysis:
  The customers analysis report will include data regarding AudioCodes's customers by date, country, store, product, and category. This report is aimed to help AudioCodes's customer department to better understand their customers' behavior, like what (products), where (countries and stores), and when do they shop. This is vital to retain current customers and reach new ones.
- Executive Dashboard:
  The dashboard will include key visuals from the two reports. The dashboard will allow a wider perspective on the data and will integrate measures both from sales and customer analysis.

2. **Gantt**

Gantt Link.

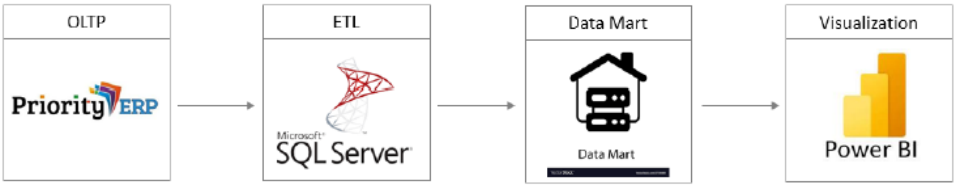
3. **Technical Specification**

3.1. Prerequisites

SQL Server	ERP system in the operational DB (PriorityERP) – tables, data(SQL files)
SSIS	ETL processes using SSIS in Visual Studio
Data Refresh Processes	Definition od JOBS in SSMS
Power BI	Creating reports and dashboards using Power BI

3.2. Solution Architecture

3.2.1. High Level Design:



The ETL process, which includes arranging the data into a Data Mart will be performed in SQL Server using SSIS. After the Data Mart creation, reports will be created using Power BI.

3.2.2. Power BI Reports:

3.2.2.1. The reports for the sales department will consist of:

- Total Revenue
- YTD Revenue/Orders/Units
- Total Orders
- Total Units
- Total Sales per country / region (hierarchy)
- Currently vs. Year over Year Revenue By Year/Quarter/Month
- Total Orders/Units/Revenue per Store (you can choose Orders, Revenue or units)

3.2.2.2. The report for the customer department will consist of:

- Total numbers of customers
- Number of new customers

- YTD customers
- Online Customer Orders by Country/Region/City/Address
- New Customers vs. All Orders Per Year/Quarter/Month
- Online vs. Frontal Customers
- Frontal vs. Online Customers Per Sub Category/Category/Product

3.2.2.3. The executive dashboard will consist of:

- Total orders
- Total units sold
- Total revenue
- Online vs. Frontal Customers
- Top 3 Agents by Orders
- Top 5 Categories By Average Quantity, Max Unit Price And Total Orders
- Number of different customers per year/quarter/month
- The Store With The Highest Transaction
- The Employee With The Highest Transaction

4. Functional Specification

4.1. Creation of final Source To Target and ERD models.

4.1.1. Source To Target

Sourct To Target Link

4.1.2. ERD model of the AudioCodes database

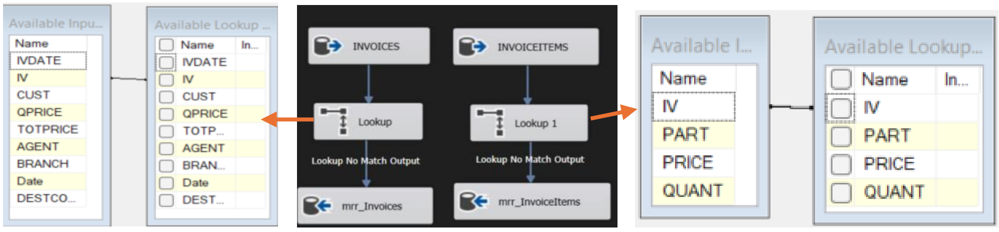
ERD Link

4.2. ETL processes

The ETL process was done in SSIS using 15 packages.

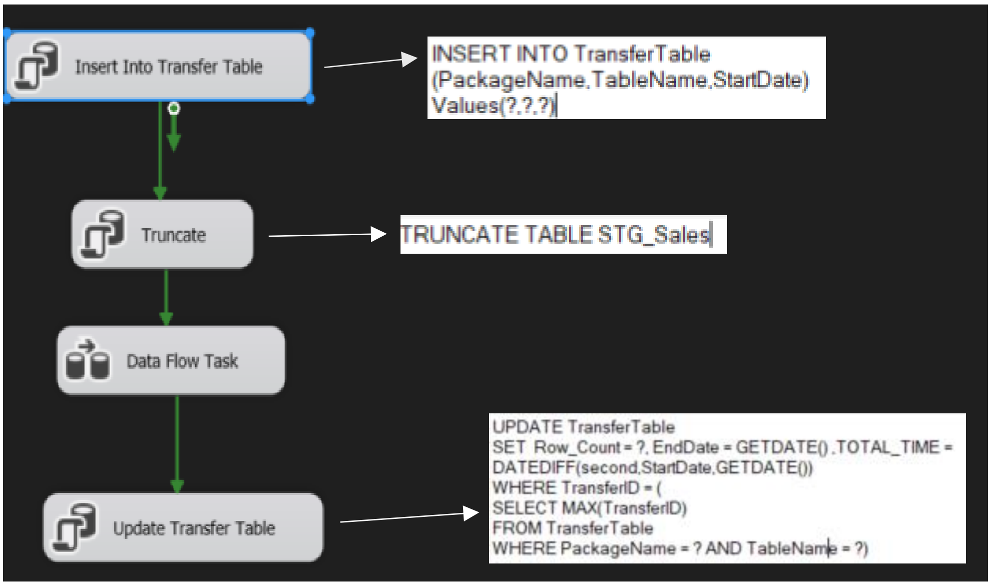
All the STG packages and the DWH packages include 2 reoccurring Execute SQL tasks (Insert To Transfer Table and Update Transfer Table), and Row Count transformations which oversee updating the Transfer Table. These will be explained later in the Transfer Table section.

- **Fact_Sales Table:**
  - Mrr_Orders Package:
    In the data flow, data is incrementally loaded using lookup transformations, meaning only new transactions that cannot be found in the FactSales table are loaded:



  - STG_Sales Package:

The STG table is truncated, and the mirror tables are joined and loaded using a data flow task. In Addition, there is an insertion the load to a Transfer table:

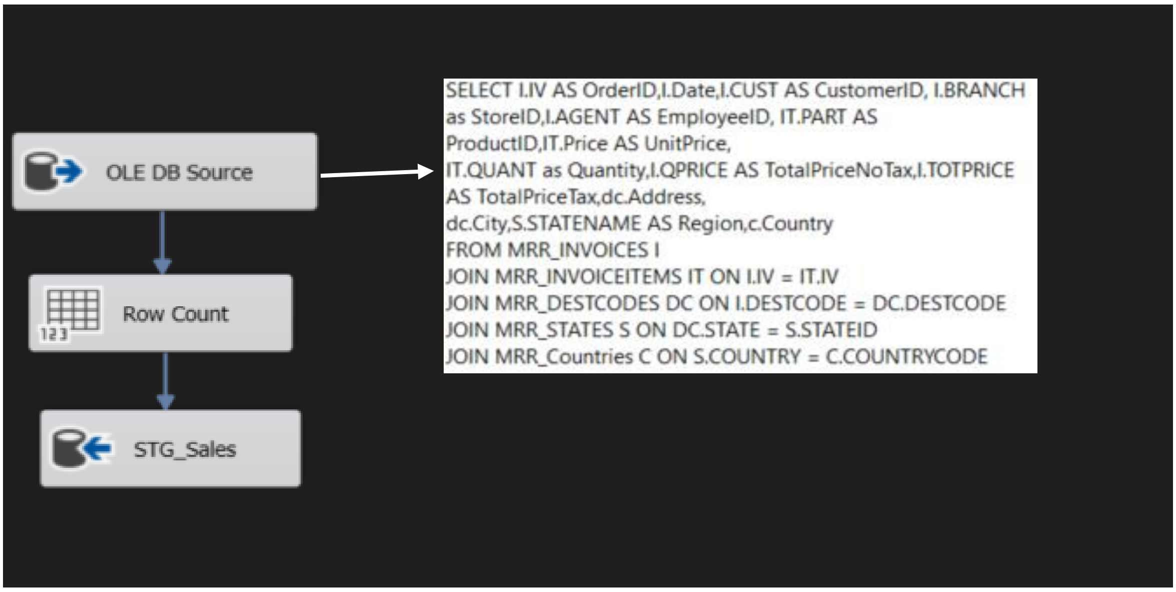


```
INSERT INTO TransferTable
(PackageName,TableName,StartDate)
Values(?,?,?)
```

```
TRUNCATE TABLE STG_Sales
```

```
UPDATE TransferTable
SET  Row_Count = ?, EndDate = GETDATE() ,TOTAL_TIME =
DATEDIFF(second,StartDate,GETDATE())
WHERE TransferID = (
SELECT MAX(TransferID)
FROM TransferTable
WHERE PackageName = ? AND TableName = ?)
```

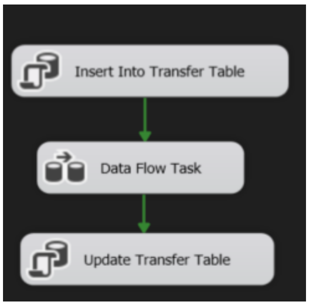
In the data flow, the mirror tables (Mrr_Invoices and Mrr_InvoiceItems) are joined, and the data is loaded to Stg_Sales table.



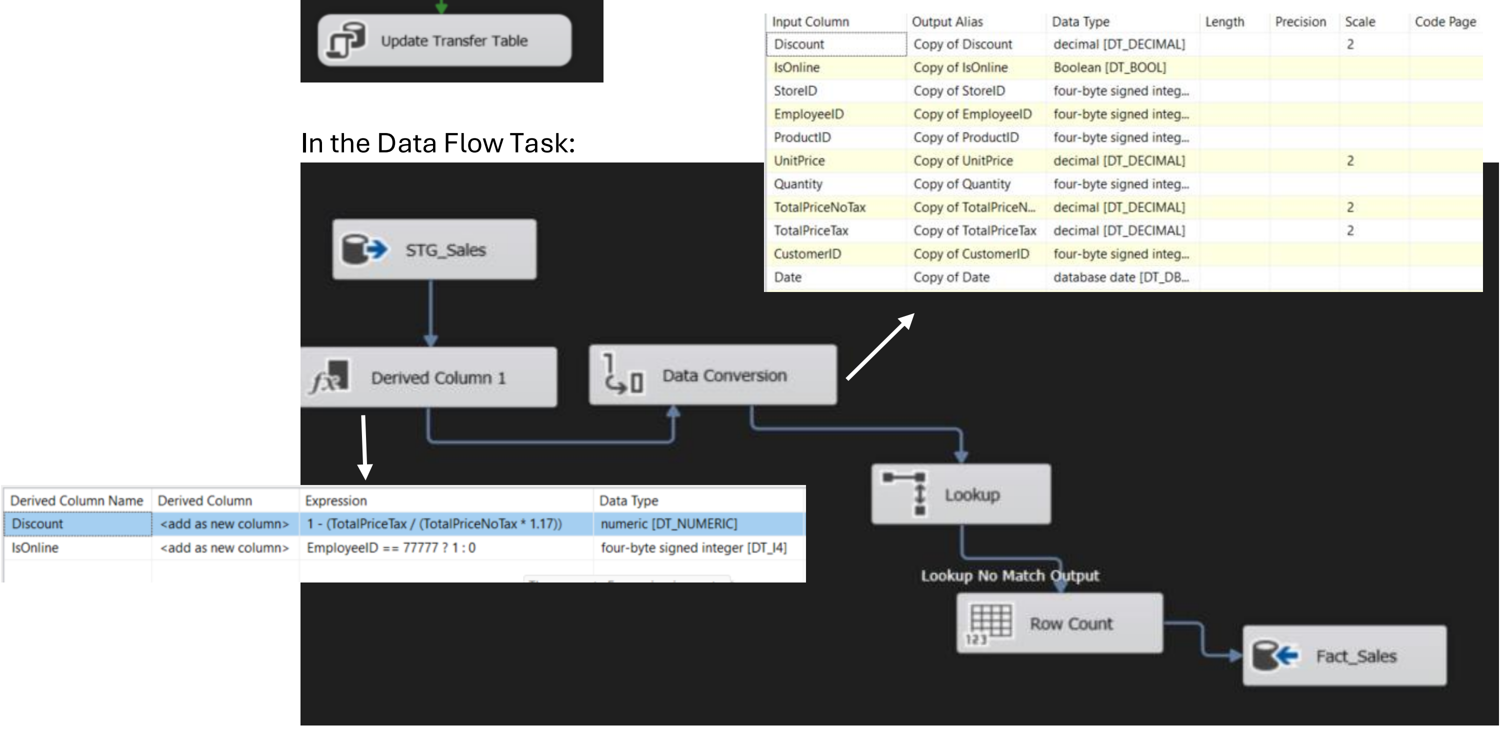
```
SELECT I.IV AS OrderID,I.Date,I.CUST AS CustomerID, I.BRANCH
as StoreID,I.AGENT AS EmployeeID, IT.PART AS
ProductID,IT.Price AS UnitPrice,
IT.QUANT as Quantity,I.QPRICE AS TotalPriceNoTax,I.TOTPRICE
AS TotalPriceTax,dc.Address,
dc.City,S.STATENAME AS Region,c.Country
FROM MRR_INVOICES I
JOIN MRR_INVOICEITEMS IT ON I.IV = IT.IV
JOIN MRR_DESTCODES DC ON I.DESTCODE = DC.DESTCODE
JOIN MRR_STATES S ON DC.STATE = S.STATEID
JOIN MRR_Countries C ON S.COUNTRY = C.COUNTRYCODE
```

- <u>DWH_Fact_Sales Package:</u>

Data is loaded from Stg_Sales to Fact_Sales, and a Discount and IsOnline columns are added.



In the Data Flow Task:



Input Column	Output Alias	Data Type	Length	Precision	Scale	Code Page
Discount	Copy of Discount	decimal [DT_DECIMAL]			2	
IsOnline	Copy of IsOnline	Boolean [DT_BOOL]				
StoreID	Copy of StoreID	four-byte signed integ...				
EmployeeID	Copy of EmployeeID	four-byte signed integ...				
ProductID	Copy of ProductID	four-byte signed integ...				
UnitPrice	Copy of UnitPrice	decimal [DT_DECIMAL]			2	
Quantity	Copy of Quantity	four-byte signed integ...				
TotalPriceNoTax	Copy of TotalPriceN...	decimal [DT_DECIMAL]			2	
TotalPriceTax	Copy of TotalPriceTax	decimal [DT_DECIMAL]			2	
CustomerID	Copy of CustomerID	four-byte signed integ...				
Date	Copy of Date	database date [DT_DB...				

Derived Column Name	Derived Column	Expression	Data Type
Discount	<add as new column>	1 - (TotalPriceTax / (TotalPriceNoTax * 1.17))	numeric [DT_NUMERIC]
IsOnline	<add as new column>	EmployeeID == 77777 ? 1 : 0	four-byte signed integer [DT_I4]

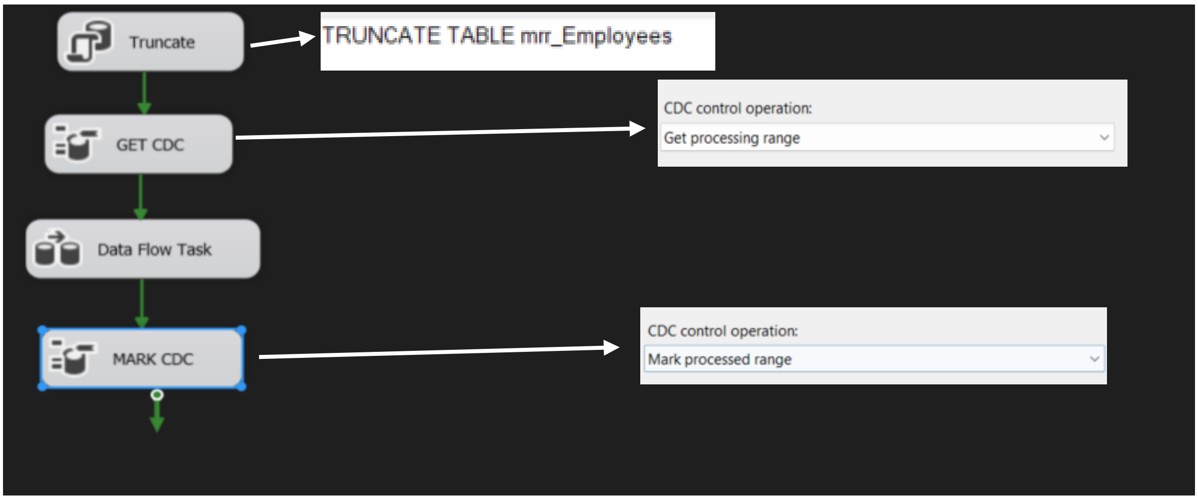
- **Dim_Employees Table:**

  This table is loaded using CDC (Change Data Capture).

  Below are the packages that carry out the loading process:

  - <u>CDC_Employees Package:</u>

    In this package there are Truncate of the mrr table that I create here, and start of the CDC processing

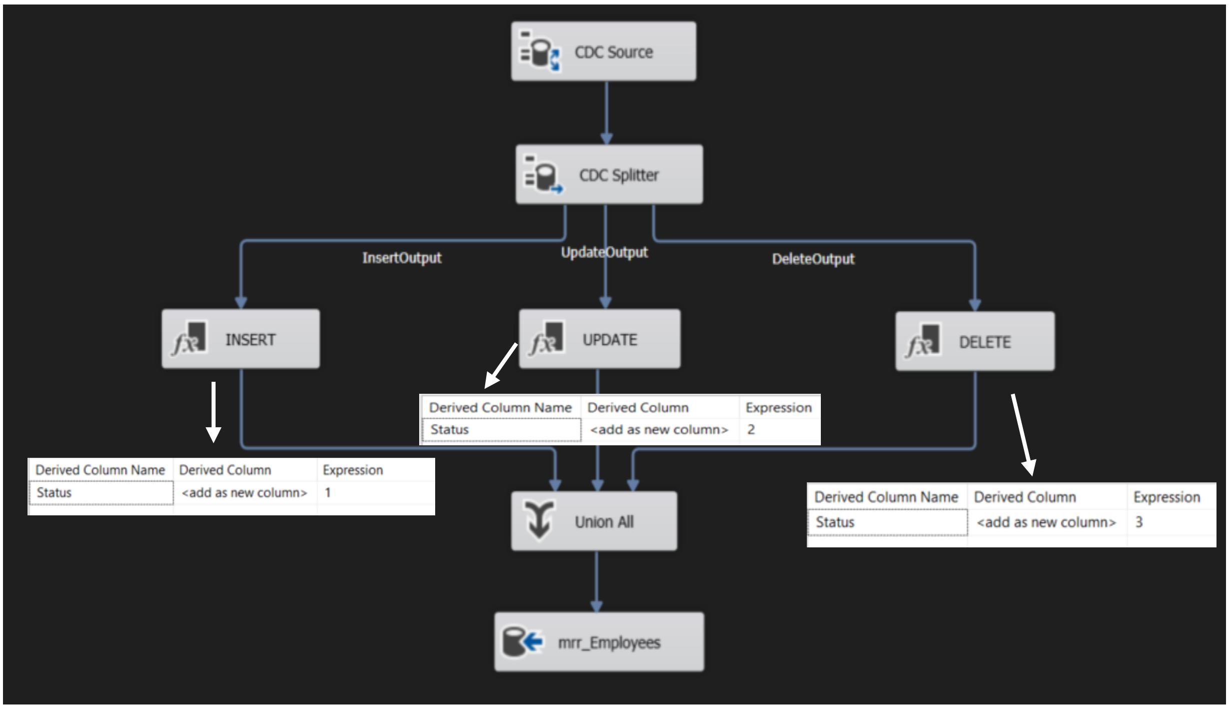


```
TRUNCATE TABLE mrr_Employees
```

CDC control operation:
Get processing range

CDC control operation:
Mark processed range

In the Data Flow Task, I used in CDC Splitter for split the data by status (1- insert, 2- update, 3- delete) and then union all the data for insertion into the mrr_Employees:



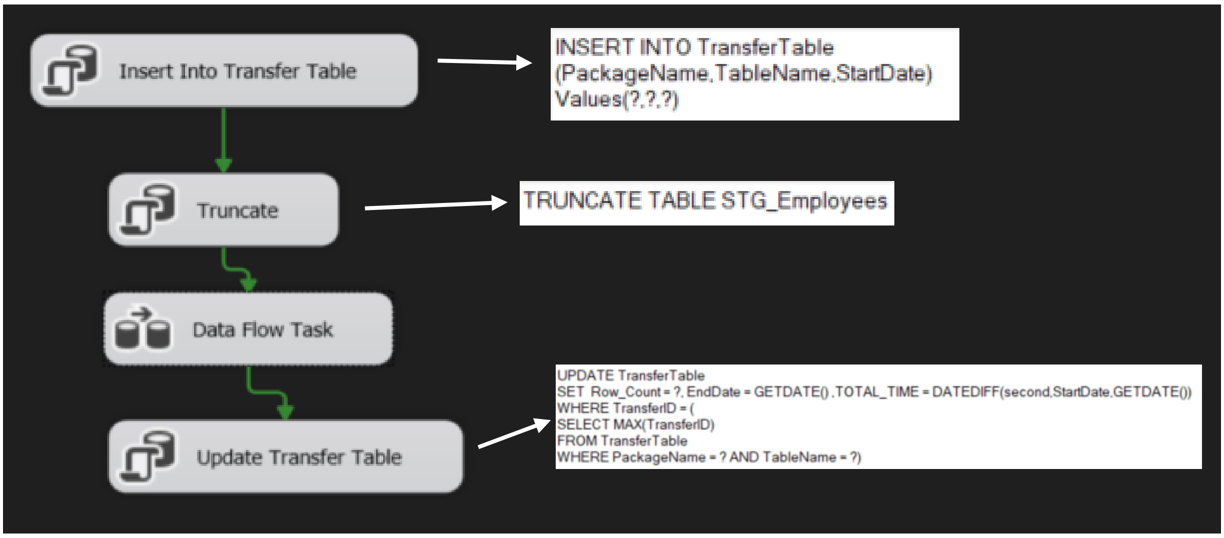
Derived Column Name	Derived Column	Expression
Status	<add as new column>	1

Derived Column Name	Derived Column	Expression
Status	<add as new column>	2

Derived Column Name	Derived Column	Expression
Status	<add as new column>	3

- ○ STG_Employees Package:

In this package I inserted to the Transfer Table the data of the process and truncated the table that this package is build.

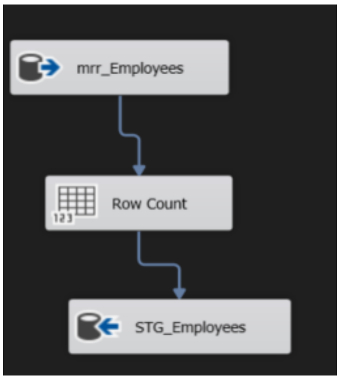


```
INSERT INTO TransferTable
(PackageName,TableName,StartDate)
Values(?,?,?)
```

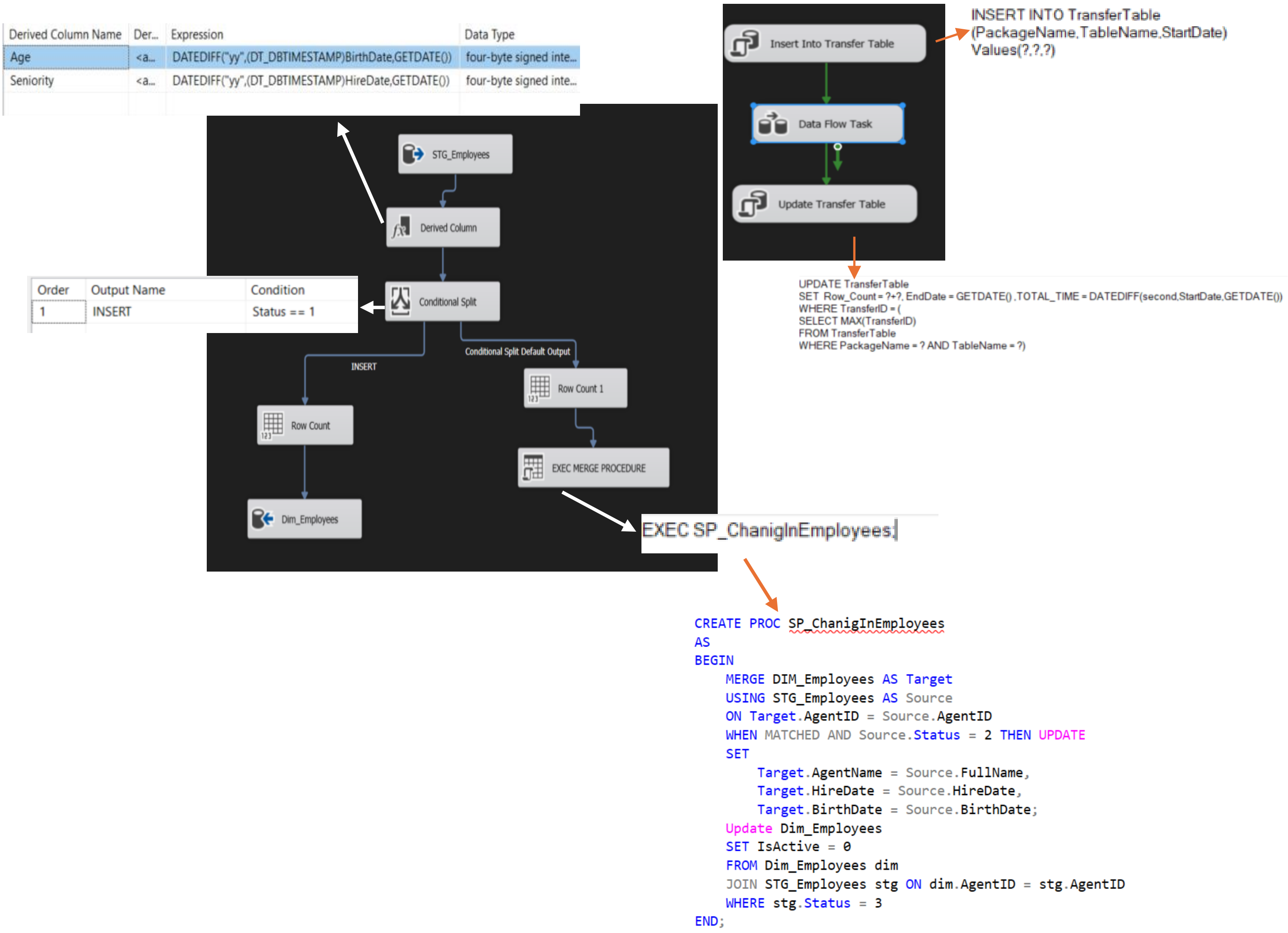
```
TRUNCATE TABLE STG_Employees
```

```
UPDATE TransferTable
SET  Row_Count = ?, EndDate = GETDATE() ,TOTAL_TIME = DATEDIFF(second,StartDate,GETDATE())
WHERE TransferID = (
SELECT MAX(TransferID)
FROM TransferTable
WHERE PackageName = ? AND TableName = ?)
```

And the Data Flow Task:



- <u>DWH_Employees Package:</u>
  The Control Flow Task include the insertion to Transfer Table the processing data
  And the Data Flow Task includes the creation of the columns Age and Seniority columns.
  After that I split the data by status:
  Status = 1 – Insert to table
  Status = 2– Go to procedure and update the data
  Status = 3 – Go to procedure and set IsActive = 0.



Derived Column Name	Der...	Expression	Data Type
Age	<a...	DATEDIFF("yy",(DT_DBTIMESTAMP)BirthDate,GETDATE())	four-byte signed inte...
Seniority	<a...	DATEDIFF("yy",(DT_DBTIMESTAMP)HireDate,GETDATE())	four-byte signed inte...

Order	Output Name	Condition
1	INSERT	Status == 1

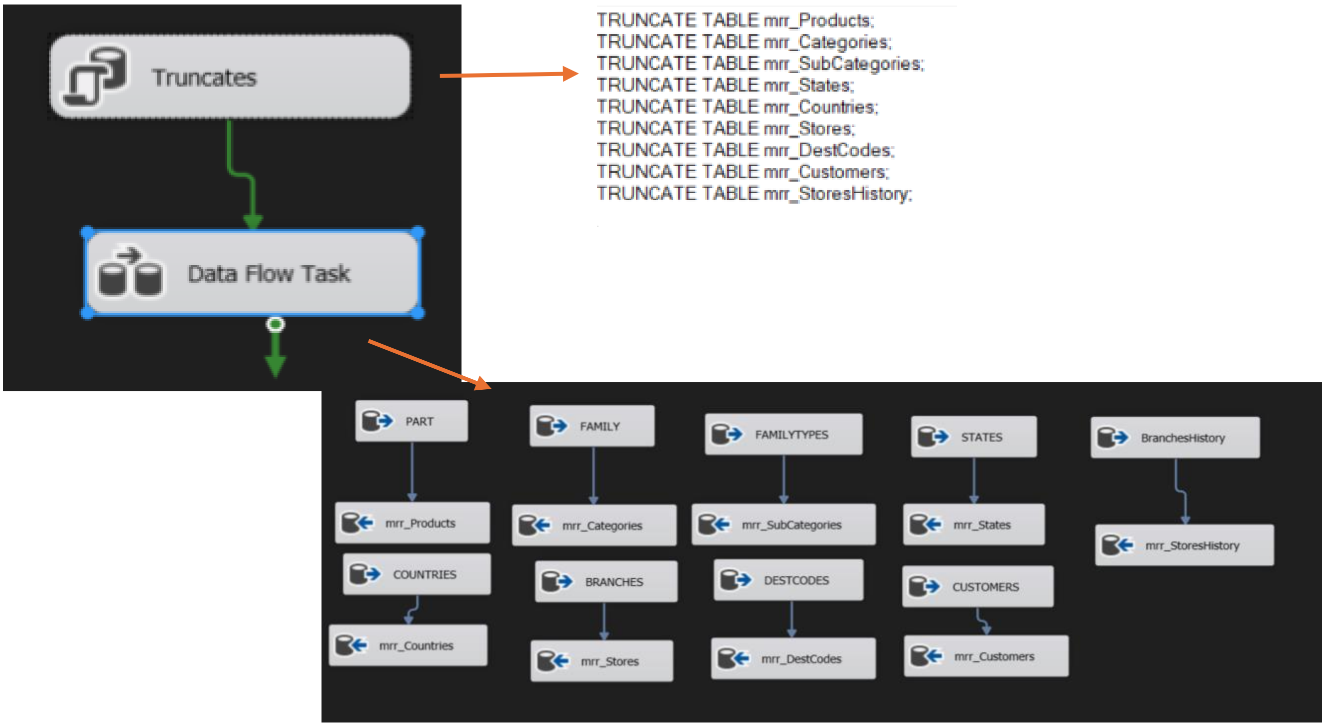
```
INSERT INTO TransferTable
(PackageName,TableName,StartDate)
Values(?,?,?)
```

```
UPDATE TransferTable
SET  Row_Count = ?+?, EndDate = GETDATE() ,TOTAL_TIME = DATEDIFF(second,StartDate,GETDATE())
WHERE TransferID = (
SELECT MAX(TransferID)
FROM TransferTable
WHERE PackageName = ? AND TableName = ?)
```

```
EXEC SP_ChanigInEmployees
```

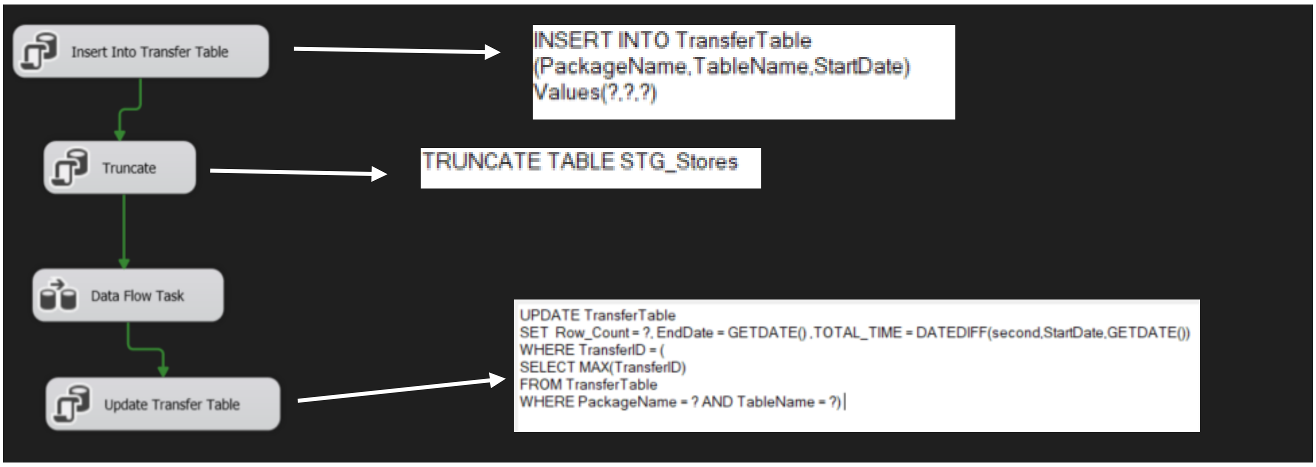
```
CREATE PROC SP_ChanigInEmployees
AS
BEGIN
    MERGE DIM_Employees AS Target
    USING STG_Employees AS Source
    ON Target.AgentID = Source.AgentID
    WHEN MATCHED AND Source.Status = 2 THEN UPDATE
    SET
        Target.AgentName = Source.FullName,
        Target.HireDate = Source.HireDate,
        Target.BirthDate = Source.BirthDate;
    Update Dim_Employees
    SET IsActive = 0
    FROM Dim_Employees dim
    JOIN STG_Employees stg ON dim.AgentID = stg.AgentID
    WHERE stg.Status = 3
END;
```

- **Other the mrr Tables Package**
  This package is responsible for loading data from PriorityERP tables to all mirror tables relevant for the dim tables (9 tables in total). All mirror tables (except mrr_Employees, mrr_Invoices,mrr_InvoiceItems) are.



```
TRUNCATE TABLE mrr_Products;
TRUNCATE TABLE mrr_Categories;
TRUNCATE TABLE mrr_SubCategories;
TRUNCATE TABLE mrr_States;
TRUNCATE TABLE mrr_Countries;
TRUNCATE TABLE mrr_Stores;
TRUNCATE TABLE mrr_DestCodes;
TRUNCATE TABLE mrr_Customers;
TRUNCATE TABLE mrr_StoresHistory;
```

- **Dim_Stores Table**
  - STG_Stores Package:
  - The STG table is truncated, and the mirror tables are joined and loaded using a data flow task. In Addition, there is an insertion the load to a Transfer table:

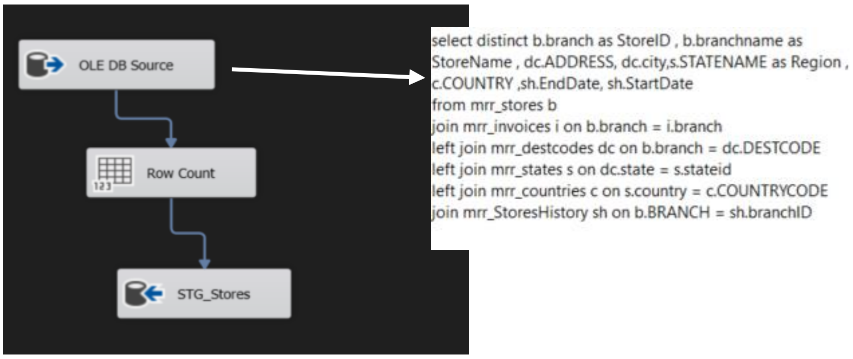


```
INSERT INTO TransferTable
(PackageName,TableName,StartDate)
Values(?,?,?)
```

```
TRUNCATE TABLE STG_Stores
```

```
UPDATE TransferTable
SET  Row_Count = ?, EndDate = GETDATE() ,TOTAL_TIME = DATEDIFF(second,StartDate,GETDATE())
WHERE TransferID = (
SELECT MAX(TransferID)
FROM TransferTable
WHERE PackageName = ? AND TableName = ?)
```

And the Date Flow Task I used the table: mrr_Stores, mrr_Invoices, mrr_DestCodes, mrr_States, mrr_Countries and mrr_StoresHistory.

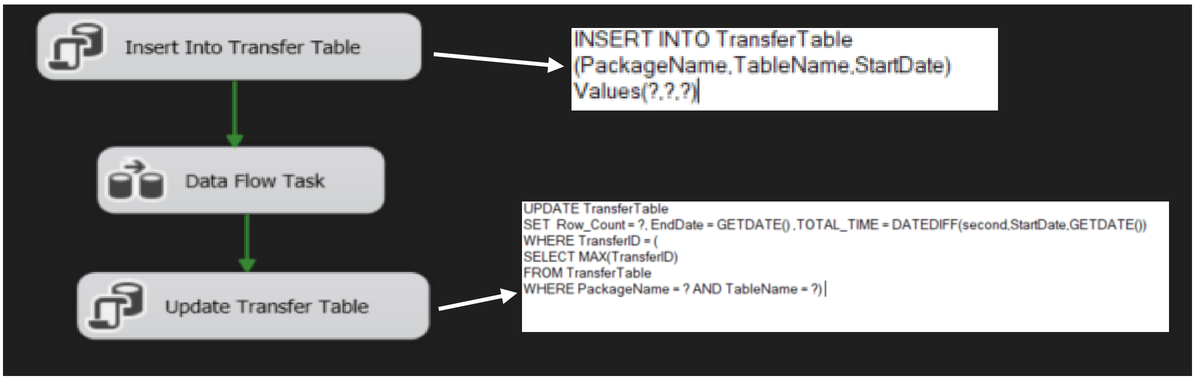


```
select distinct b.branch as StoreID , b.branchname as
StoreName , dc.ADDRESS, dc.city,s.STATENAME as Region ,
c.COUNTRY ,sh.EndDate, sh.StartDate
from mrr_stores b
join mrr_invoices i on b.branch = i.branch
left join mrr_destcodes dc on b.branch = dc.DESTCODE
left join mrr_states s on dc.state = s.stateid
left join mrr_countries c on s.country = c.COUNTRYCODE
join mrr_StoresHistory sh on b.BRANCH = sh.branchID
```

- <u>DWH_Stores Package</u>

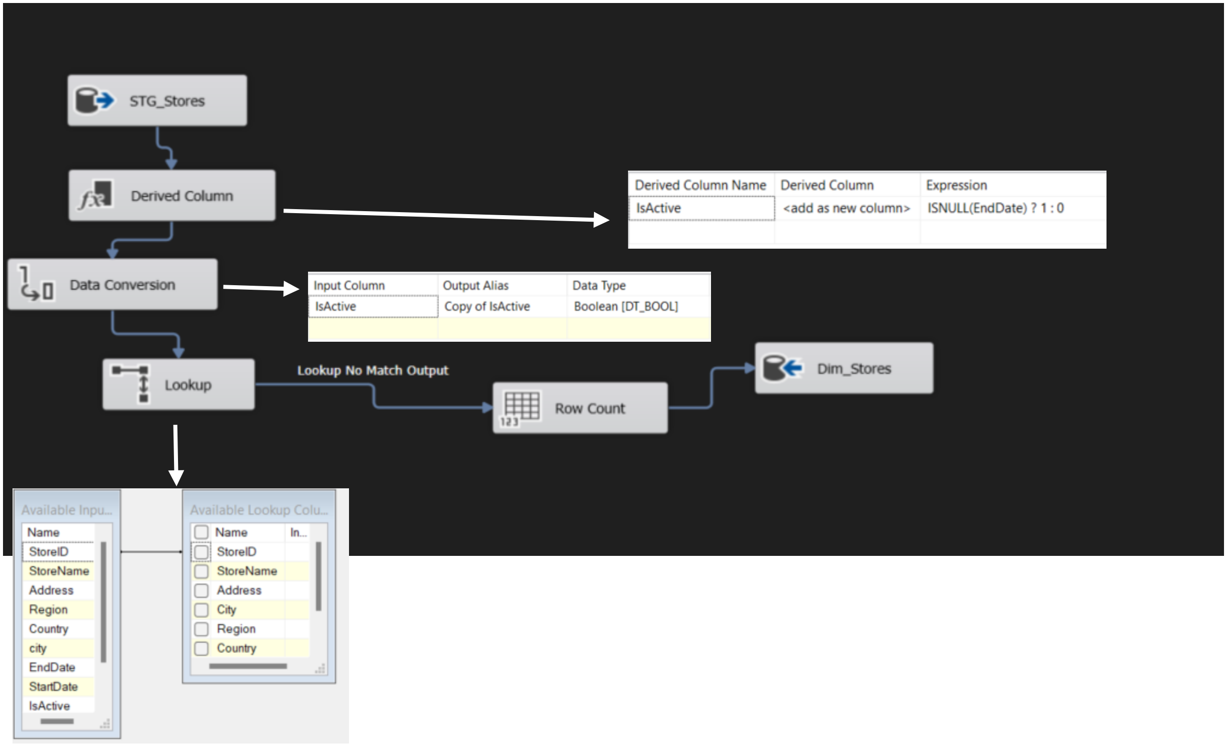
  Load the data from STG_Stores to the Dim_Stores table using an incrementally loaded using lookup and creation the IsActive column.

  In addition, insertion data into the Transfer Table.



```
INSERT INTO TransferTable
(PackageName,TableName,StartDate)
Values(?,?,?)
```

```
UPDATE TransferTable
SET  Row_Count = ?, EndDate = GETDATE() ,TOTAL_TIME = DATEDIFF(second,StartDate,GETDATE())
WHERE TransferID = (
SELECT MAX(TransferID)
FROM TransferTable
WHERE PackageName = ? AND TableName = ?)
```



Derived Column Name	Derived Column	Expression
IsActive	<add as new column>	ISNULL(EndDate) ? 1 : 0

Input Column	Output Alias	Data Type
IsActive	Copy of IsActive	Boolean [DT_BOOL]

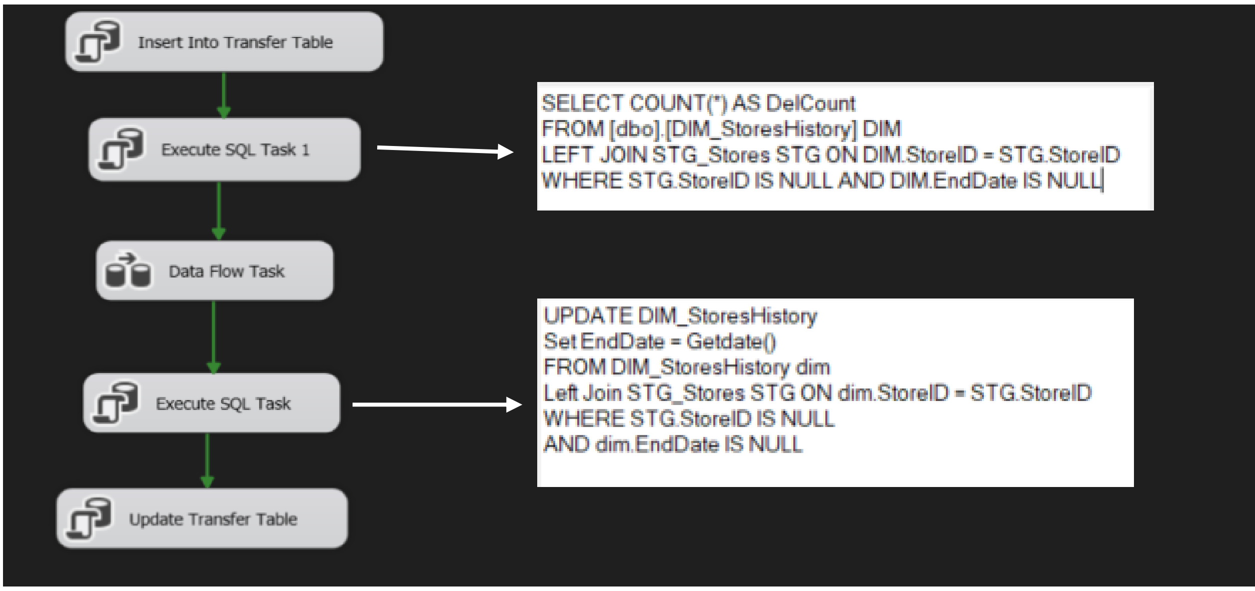
Available Inpu...

Name
StoreID
StoreName
Address
Region
Country
city
EndDate
StartDate
IsActive

Available Lookup Colu...

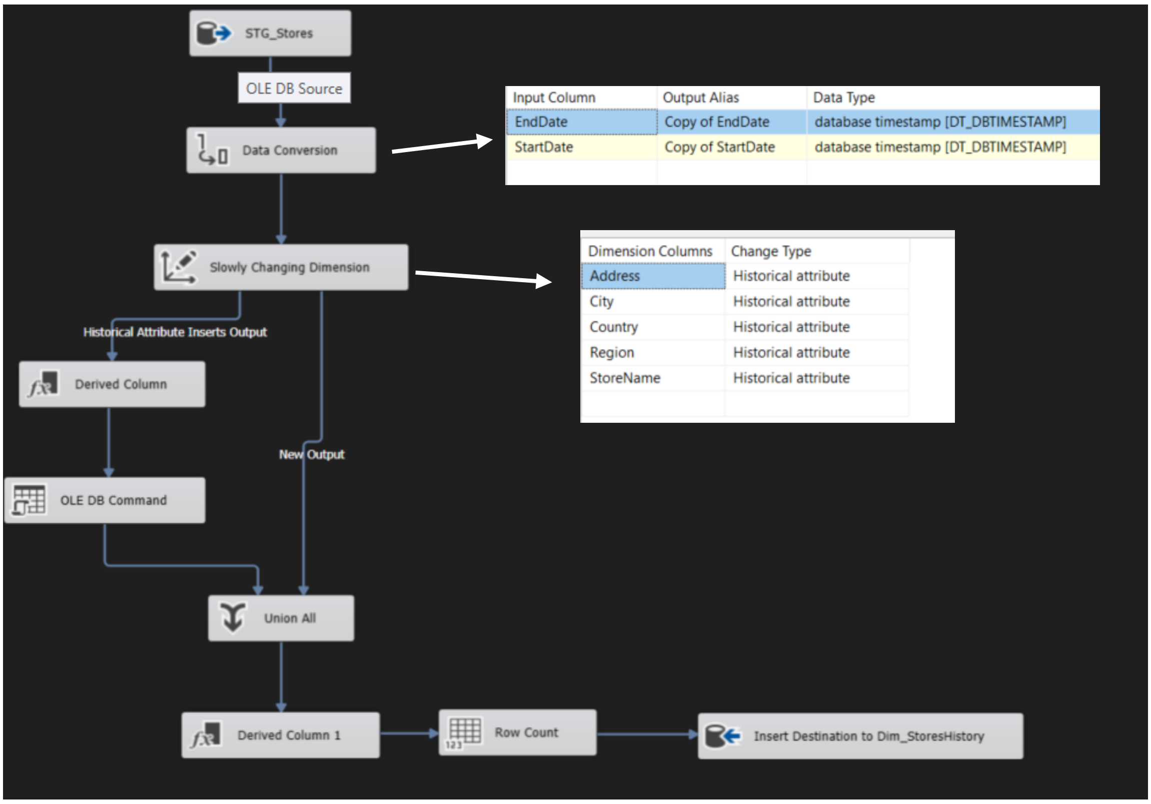
□	Name	In...
□	StoreID	
□	StoreName	
□	Address	
□	City	
□	Region	
□	Country	

- **Dim_StoresHistory Table**
  For this table SCD (Slowly Changing Dimension) with type 4 (historical atterbute) is used in the DWH_StoresHistory Package.
  First, In the Control Flow Task there is an insertion data into the Transfer Table and then there is a query that count for us how many stores we need to update in this table and update it after that.



```
SELECT COUNT(*) AS DelCount
FROM [dbo].[DIM_StoresHistory] DIM
LEFT JOIN STG_Stores STG ON DIM.StoreID = STG.StoreID
WHERE STG.StoreID IS NULL AND DIM.EndDate IS NULL
```

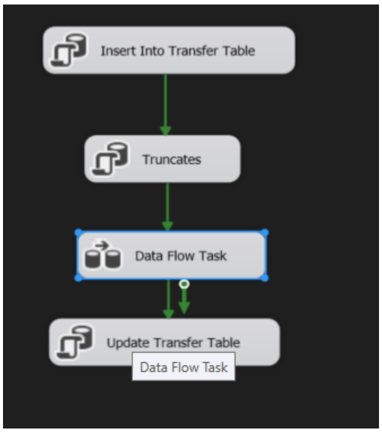
```
UPDATE DIM_StoresHistory
Set EndDate = Getdate()
FROM DIM_StoresHistory dim
Left Join STG_Stores STG ON dim.StoreID = STG.StoreID
WHERE STG.StoreID IS NULL
AND dim.EndDate IS NULL
```



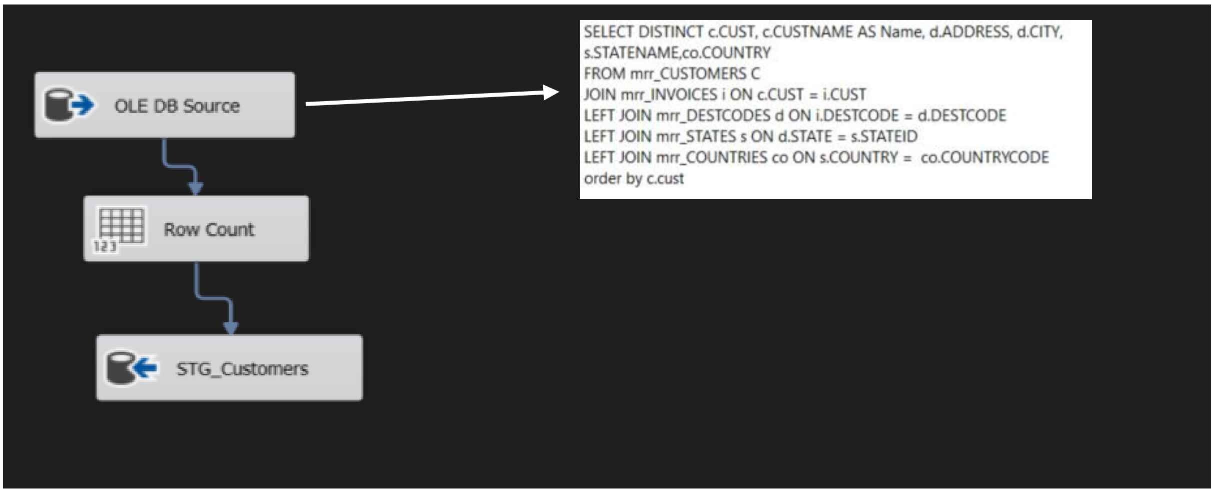
Input Column	Output Alias	Data Type
EndDate	Copy of EndDate	database timestamp [DT_DBTIMESTAMP]
StartDate	Copy of StartDate	database timestamp [DT_DBTIMESTAMP]

Dimension Columns	Change Type
Address	Historical attribute
City	Historical attribute
Country	Historical attribute
Region	Historical attribute
StoreName	Historical attribute

- **Dim_Customers Table**
  - STG_Customers Packages:



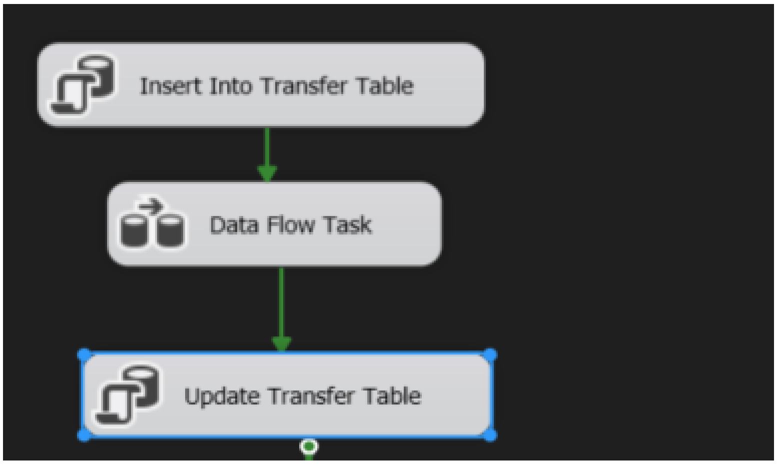
In the Data Flow Task, I used the tables: mrr_Customers, mrr_Invoices, mrr_DestCodes, mrr_States and mrr_Countries.

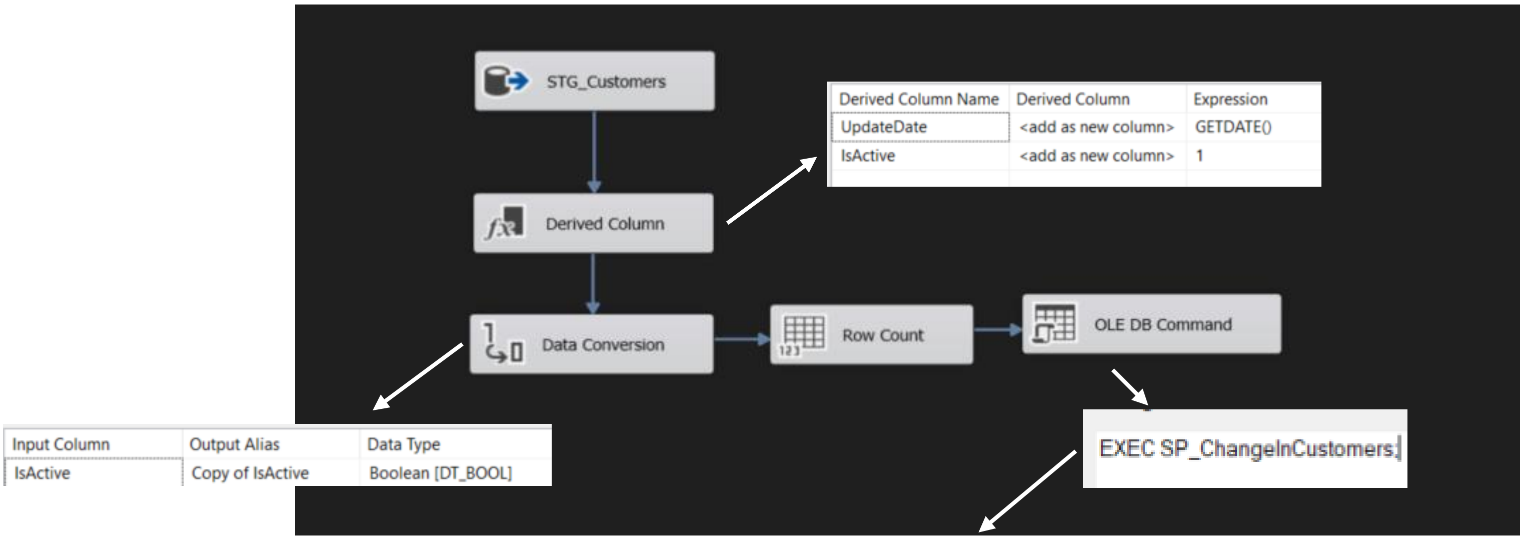


```
SELECT DISTINCT c.CUST, c.CUSTNAME AS Name, d.ADDRESS, d.CITY,
s.STATENAME,co.COUNTRY
FROM mrr_CUSTOMERS C
JOIN mrr_INVOICES i ON c.CUST = i.CUST
LEFT JOIN mrr_DESTCODES d ON i.DESTCODE = d.DESTCODE
LEFT JOIN mrr_STATES s ON d.STATE = s.STATEID
LEFT JOIN mrr_COUNTRIES co ON s.COUNTRY =  co.COUNTRYCODE
order by c.cust
```

  - DWH_Customers Package:

In the Control Flow Task there is an insertion date into Transfer Table, and in the Data Flow Task created the columns UpdateDate and IsActive, and then start a process in Store Procedure to insert or update(Merge Query) the relevant rows.

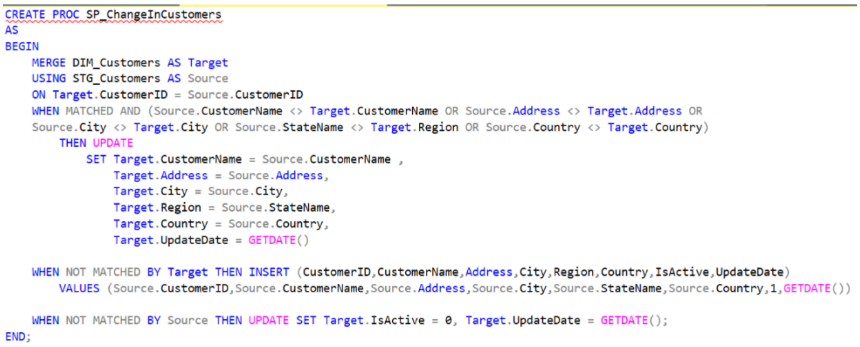




Derived Column Name	Derived Column	Expression
UpdateDate	<add as new column>	GETDATE()
IsActive	<add as new column>	1

Input Column	Output Alias	Data Type
IsActive	Copy of IsActive	Boolean [DT_BOOL]

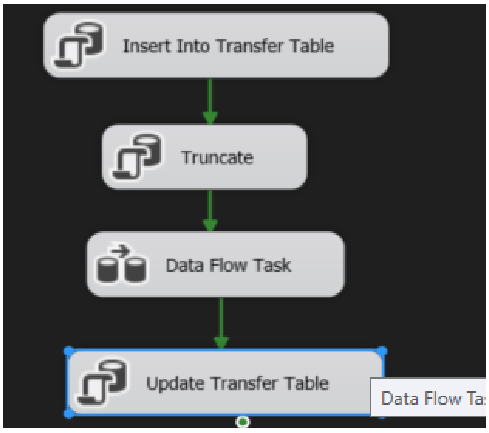
```
EXEC SP_ChangeInCustomers
```



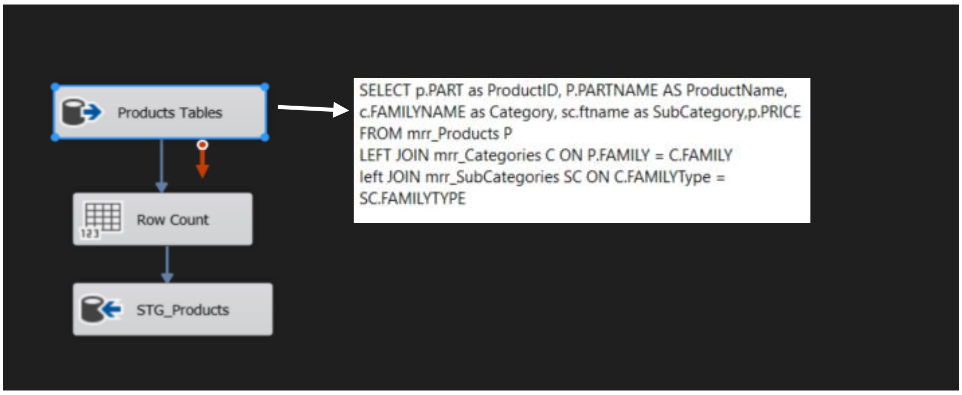
```
CREATE PROC SP_ChangeInCustomers
AS
BEGIN
    MERGE DIM_Customers AS Target
    USING STG_Customers AS Source
    ON Target.CustomerID = Source.CustomerID
    WHEN MATCHED AND (Source.CustomerName <> Target.CustomerName OR Source.Address <> Target.Address OR
    Source.City <> Target.City OR Source.StateName <> Target.Region OR Source.Country <> Target.Country)
        THEN UPDATE
            SET Target.CustomerName = Source.CustomerName ,
                Target.Address = Source.Address,
                Target.City = Source.City,
                Target.Region = Source.StateName,
                Target.Country = Source.Country,
                Target.UpdateDate = GETDATE()

    WHEN NOT MATCHED BY Target THEN INSERT (CustomerID,CustomerName,Address,City,Region,Country,IsActive,UpdateDate)
        VALUES (Source.CustomerID,Source.CustomerName,Source.Address,Source.City,Source.StateName,Source.Country,1,GETDATE())

    WHEN NOT MATCHED BY Source THEN UPDATE SET Target.IsActive = 0, Target.UpdateDate = GETDATE();
END;
```

- **Dim_Products Table**
  - STG_Products Paclage:

    The Control Flow Task is insert date into the Transfer Table and truncate STG_Products Table:

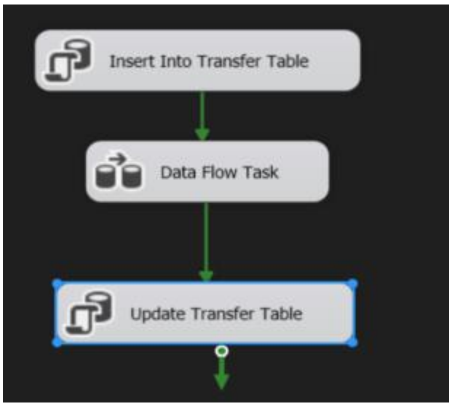


The Data Flow Task create the STG_Products Table and get the data from the tables: mrr_Products, mrr_Categories, mrr_SubCategories

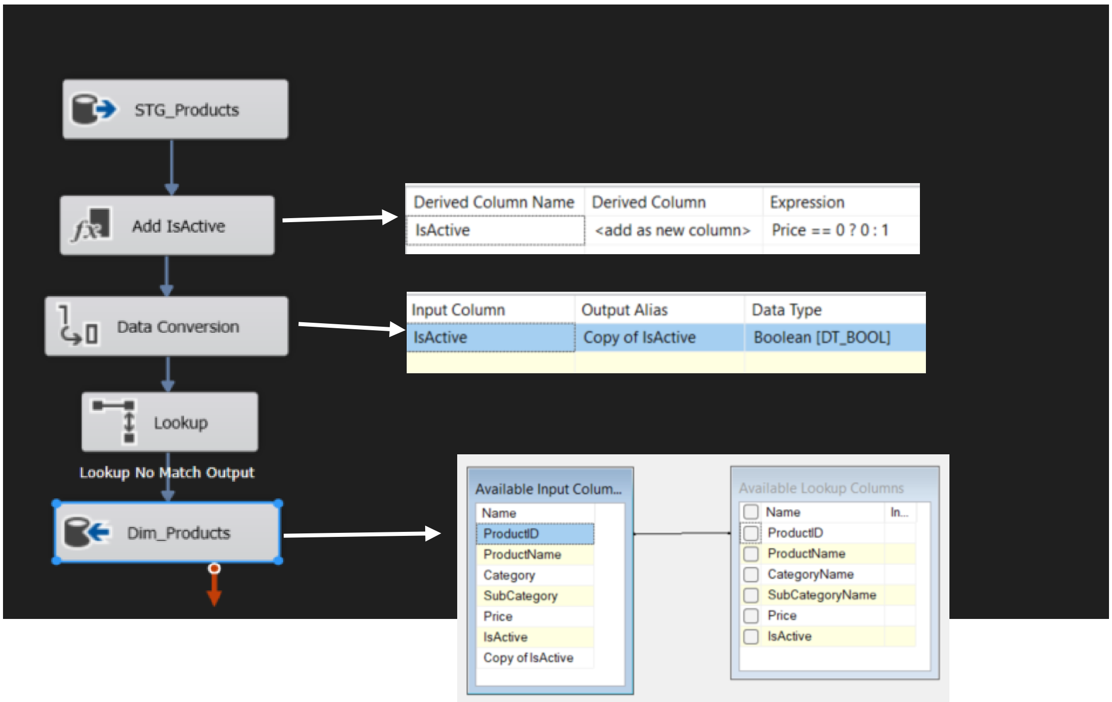


```
SELECT p.PART as ProductID, P.PARTNAME AS ProductName,
c.FAMILYNAME as Category, sc.ftname as SubCategory,p.PRICE
FROM mrr_Products P
LEFT JOIN mrr_Categories C ON P.FAMILY = C.FAMILY
left JOIN mrr_SubCategories SC ON C.FAMILYType =
SC.FAMILYTYPE
```

- DWH_Products Package:
  The Control Flow Task is inserting the processing data into the Transfer Table



The Data Flow Task is creating a new column IsActive and load data into Dim_Products using incrementally loaded using Lookup



Derived Column Name	Derived Column	Expression
IsActive	<add as new column>	Price == 0 ? 0 : 1

Input Column	Output Alias	Data Type
IsActive	Copy of IsActive	Boolean [DT_BOOL]

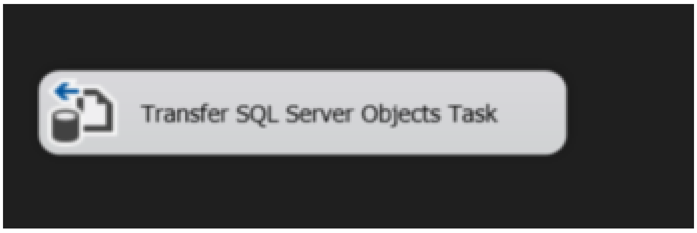
Available Input Colum...

Name
ProductID
ProductName
Category
SubCategory
Price
IsActive
Copy of IsActive

Available Lookup Columns

	Name	In...
□	ProductID	
□	ProductName	
□	CategoryName	
□	SubCategoryName	
□	Price	
□	IsActive	

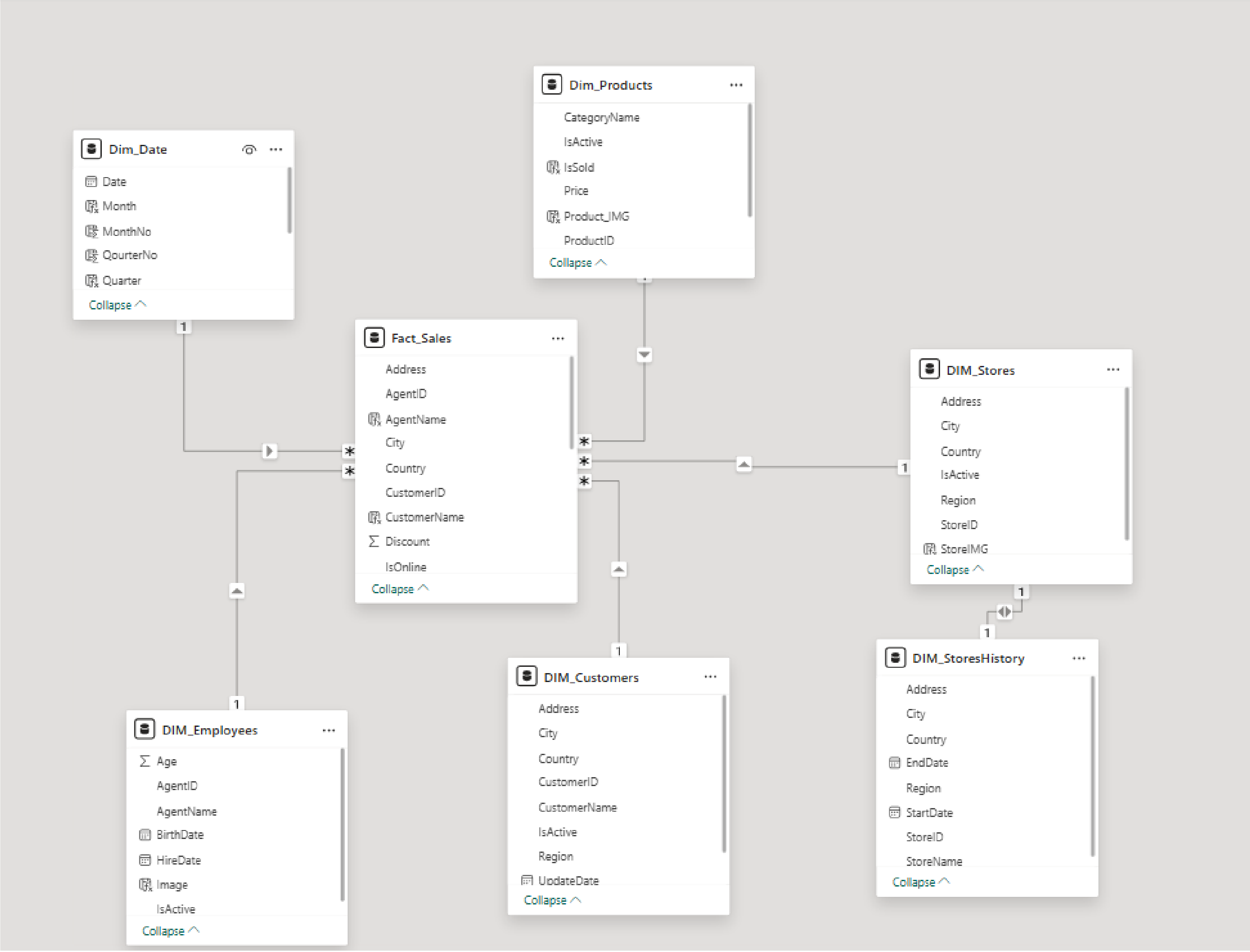
- **MovingToProduction Package:**
  The last package that transfer the database AudioCodes_DEV to AudioCodes_Production database



4.3. Visualization in Power BI

4.3.1. The reports were created using Power BI Desktop and were published to Power BI Service.

The model in the Power BI includes the Fact table and the 4 Dimension tables (not including the product history table). To these tables, a Dim_Date table was added.



4.3.2. To create the visuals, the following measures were created in DAX:

- Total Units = sum(Fact_Sales[Quantity])
- Total Units No Blank = IF(ISBLANK([Total Units]),0,[Total Units])
- Total Units YTD = TOTALYTD([Total Units No Blank],Dim_Date[Date])
- Prev Year Units = CALCULATE([Total Units No Blank],SAMEPERIODLASTYEAR(Dim_Date[Date]))
- YOY Units = IFERROR([Total Units No Blank]/[Prev Year Units] - 1,0)

- Total Revenue = SUMX (DISTINCT (SELECTCOLUMNS (Fact_Sales,"OrderID", Fact_Sales[OrderID],"TotalPriceNoTax", Fact_Sales[TotalPriceNoTax])),[TotalPriceNoTax])
- Total Revenue No Blank = IF(ISBLANK([Total Revenue]),0,[Total Revenue])
- Total Revenue YTD = TOTALYTD([Total Revenue No Blank],Dim_Date[Date])
- Prev Year Revenue = CALCULATE([Total Revenue No Blank],SAMEPERIODLASTYEAR(Dim_Date[Date]))
- YOY Revenue = IFERROR([Total Revenue No Blank]/[Prev Year Revenue] - 1,0)

- Total Orders = CALCULATE(DISTINCTCOUNT(Fact_Sales[OrderID]),Fact_Sales[OrderID])
- Total Orders No Blank = IF(ISBLANK([Total Orders]),0,[Total Orders])
- Total Orders YTD = TOTALYTD([Total Orders No Blank],Dim_Date[Date])
- Prev Year Orders = CALCULATE([Total Orders No Blank],SAMEPERIODLASTYEAR(Dim_Date[Date]))
- YOY Orders = IFERROR([Total Orders No Blank]/[Prev Year Orders] - 1,0)

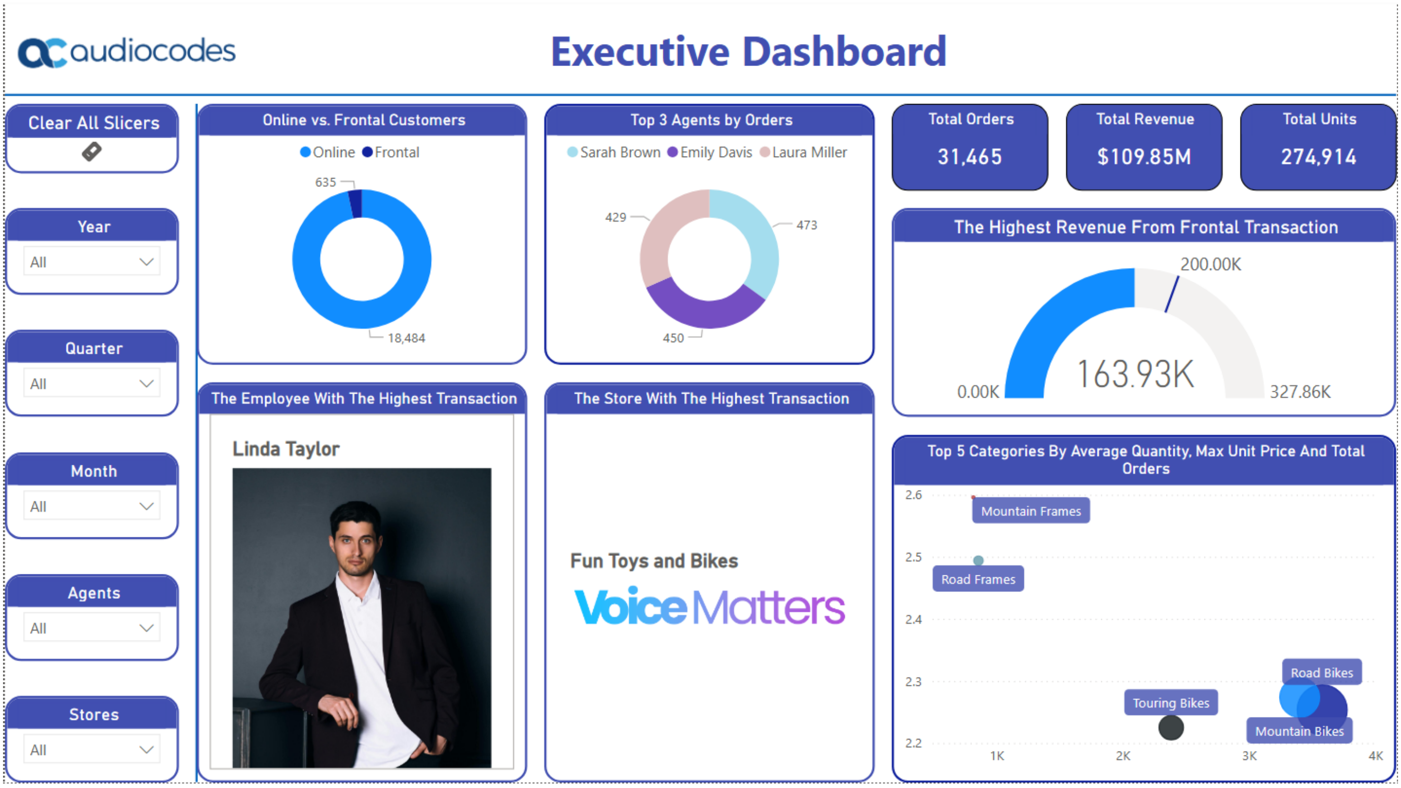
- Amount Of Customers = CALCULATE(DISTINCTCOUNT(DIM_Customers[CustomerID]),DIM_Customers[CustomerID])
- New Customers =
  VAR CurrentCustomer = VALUES(Fact_Sales[CustomerID])
  VAR CurrentDate = MIN(Fact_Sales[OrderDate])
  VAR PastCustomers =
  CALCULATETABLE(VALUES(Fact_Sales[CustomerID]),ALL(Fact_Sales[OrderDate]),Fact_Sales[OrderDate] <CurrentDate)
  VAR newCustomers = EXCEPT(CurrentCustomer,PastCustomers)
  RETURN
  IF(ISBLANK(COUNTROWS(newCustomers)),0,COUNTROWS(newCustomers))
- Total Orders By Customers = CALCULATE(DISTINCTCOUNT(Fact_Sales[OrderID]),DIM_Customers[CustomerID])
- Total Customers YTD = TOTALYTD(IF(ISBLANK([Total Orders By Customers]),0,[Total Orders By Customers]),Dim_Date[Date])
- TotalProductsPerCustomer = CALCULATE(SUM(Fact_Sales[Quantity]),Fact_Sales[ProductID]

- TopAgents = CALCULATE(DISTINCTCOUNT(Fact_Sales[OrderID]),Fact_Sales[AgentID])
- Total Categories Per Orders = CALCULATE(DISTINCTCOUNT(Fact_Sales[OrderID]),Dim_Products[ProductID])
- Average Quantity Per Product = CALCULATE(AVERAGE(Fact_Sales[Quantity]),Fact_Sales[ProductID])
- TopDillStoreIMG = TOPN(1,VALUES(DIM_Stores[StoreIMG]),CALCULATE(MAX(Fact_Sales[TotalPriceNoTax])))
- TopDillStoreName = TOPN(1,VALUES(Fact_Sales[StoreName]),CALCULATE(MAX(Fact_Sales[TotalPriceNoTax])))
- TopDillAgentIMG = TOPN(1,VALUES(DIM_Employees[Image]),CALCULATE(MAX(Fact_Sales[TotalPriceNoTax])))
- TopDillAgentName = TOPN(1,VALUES(DIM_Employees[AgentName]),CALCULATE(MAX(Fact_Sales[TotalPriceNoTax])))

4.3.3. The project includes 3 reports: Executive Dashboard, Sales Analysis, Customer Analysis.

4.3.3.1. Executive Dashboard:

This report shows us employee metrics, transactions and product categories.



audiocodes Executive Dashboard

Clear All Slicers

Year: All

Quarter: All

Month: All

Agents: All

Stores: All

Total Orders 31,465

Total Revenue $109.85M

Total Units 274,914

The Employee With The Highest Transaction: Linda Taylor

The Store With The Highest Transaction: Fun Toys and Bikes

Online vs. Frontal Customers
● Online ● Frontal
635
18,484
Top 3 Agents by Orders
● Sarah Brown ● Emily Davis ● Laura Miller
429
473
450
The Highest Revenue From Frontal Transaction
200.00K
163.93K
0.00K
327.86K
VoiceMatters
Top 5 Categories By Average Quantity, Max Unit Price And Total Orders
2.6
2.5
2.4
2.3
2.2
Mountain Frames
Road Frames
Road Bikes
Touring Bikes
Mountain Bikes
1K
2K
3K
4K

KPI Cards:

- Total Orders
- Total Revenue
- Total Units Sold

Graphs:

- Online vs. Frontal Customers
- Top 3 Agents By Orders
- Top 5 Categories By Average Quantity, Maximum Unit Price And Total Orders
- The Highest Revenue From Frontal Transaction
- The Employee With The Highest Transaction
- The Store With The Highest Transaction

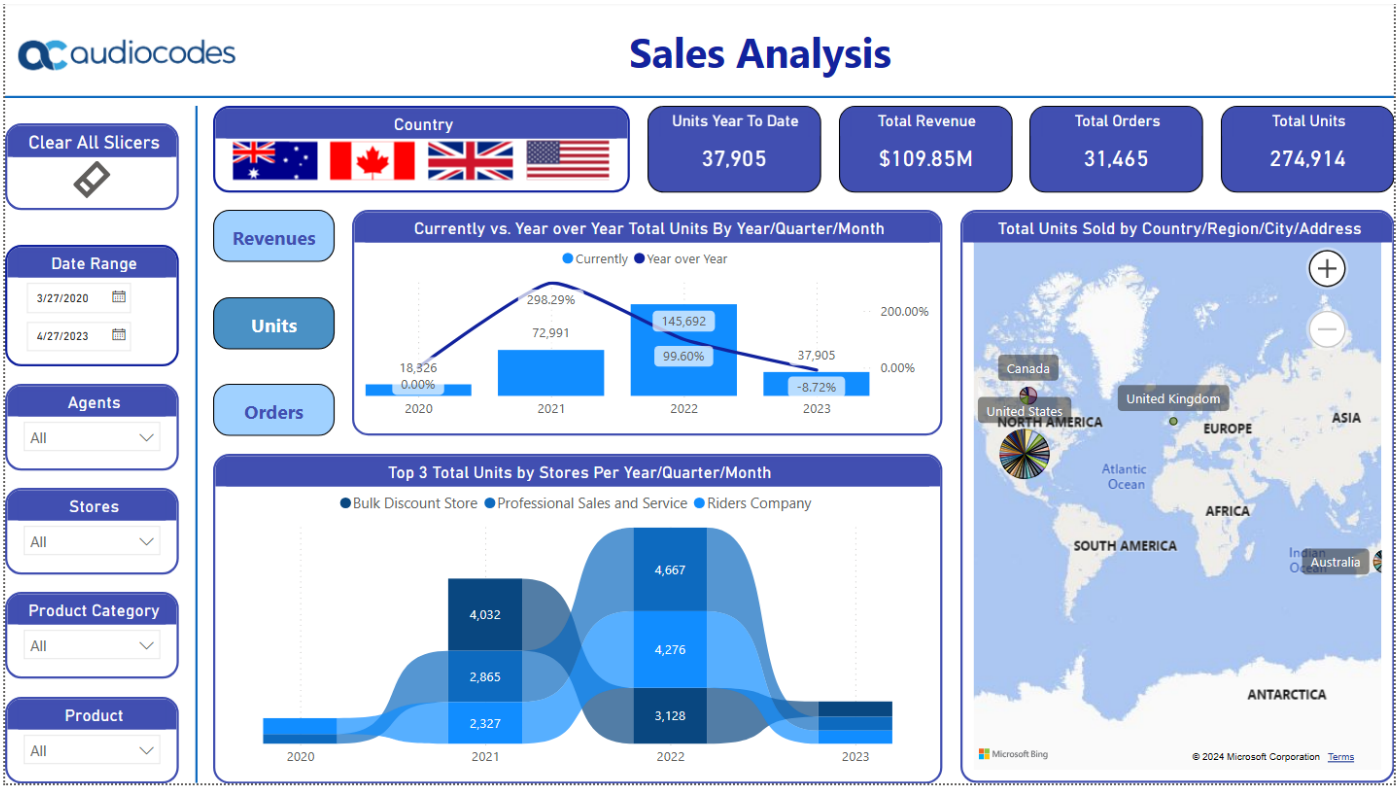
Slicers:

- Year
- Quarter
- Month
- Agents
- Stores

4.3.3.2. Sales Analysis:

This report shows us sales metrics for units, orders, revenue.

It shows us the growth of the sales department from year to year, and shows us exactly where in the world our sales department is most successful.



Sales Analysis

Clear All Slicers

Date Range: 3/27/2020 – 4/27/2023

Agents: All

Stores: All

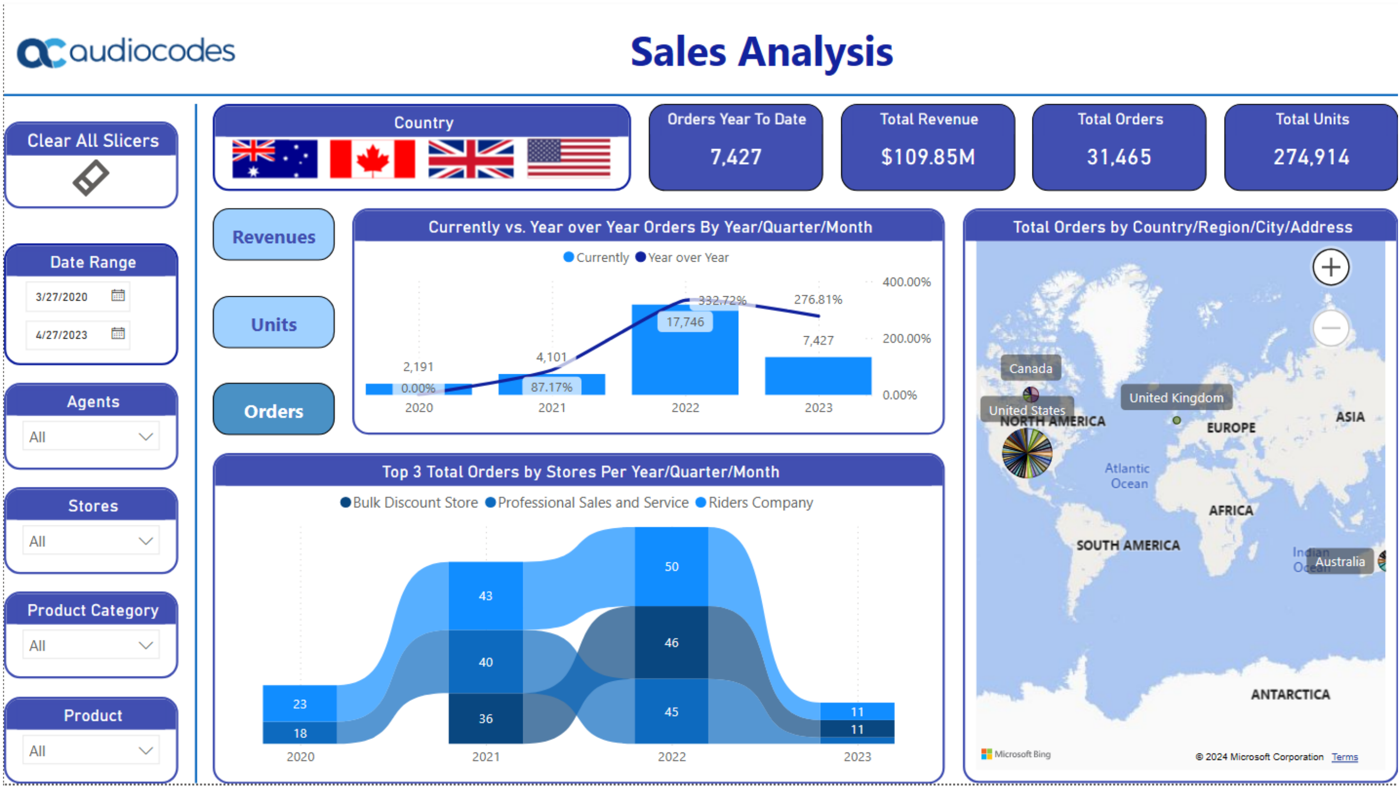
Product Category: All

Product: All

Country

Units Year To Date	Total Revenue	Total Orders	Total Units
37,905	$109.85M	31,465	274,914

audiocodes
Revenues
Units
Orders
Currently vs. Year over Year Total Units By Year/Quarter/Month
● Currently ● Year over Year
18,326
0.00%
298.29%
72,991
145,692
99.60%
37,905
-8.72%
200.00%
0.00%
2020
2021
2022
2023
Top 3 Total Units by Stores Per Year/Quarter/Month
● Bulk Discount Store ● Professional Sales and Service ● Riders Company
4,032
2,865
2,327
4,667
4,276
3,128
2020
2021
2022
2023
Total Units Sold by Country/Region/City/Address
Canada
United States
NORTH AMERICA
United Kingdom
EUROPE
ASIA
Atlantic Ocean
AFRICA
SOUTH AMERICA
Indian Ocean
Australia
ANTARCTICA
Microsoft Bing
© 2024 Microsoft Corporation Terms



Sales Analysis

Clear All Slicers

Date Range: 3/27/2020 – 4/27/2023

Agents: All

Stores: All

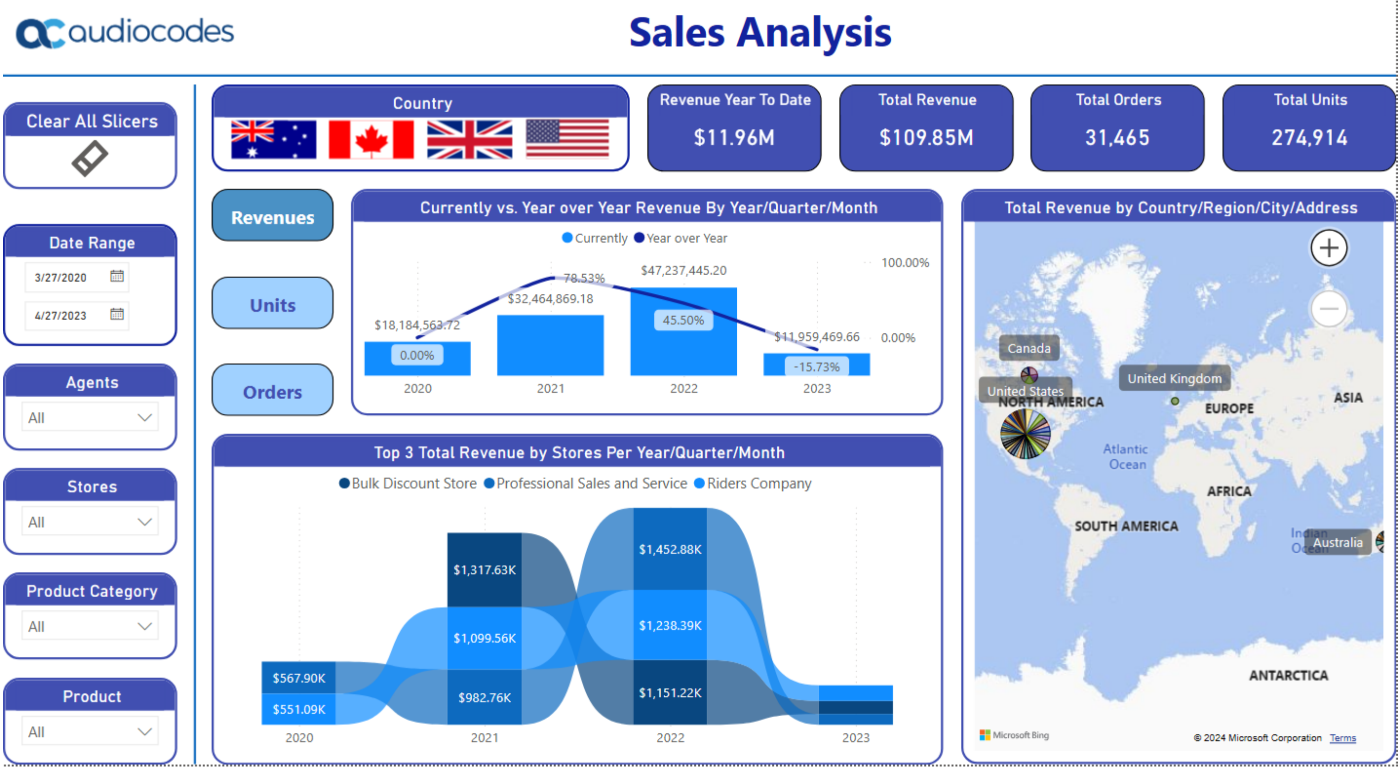
Product Category: All

Product: All

Country

Orders Year To Date	Total Revenue	Total Orders	Total Units
7,427	$109.85M	31,465	274,914

audiocodes
Revenues
Units
Orders
Currently vs. Year over Year Orders By Year/Quarter/Month
● Currently ● Year over Year
2,191
0.00%
4,101
87.17%
332.72%
17,746
276.81%
7,427
400.00%
200.00%
0.00%
2020
2021
2022
2023
Top 3 Total Orders by Stores Per Year/Quarter/Month
● Bulk Discount Store ● Professional Sales and Service ● Riders Company
23
18
43
40
36
50
46
45
11
11
2020
2021
2022
2023
Total Orders by Country/Region/City/Address
Canada
United States
NORTH AMERICA
United Kingdom
EUROPE
ASIA
Atlantic Ocean
AFRICA
SOUTH AMERICA
Indian Ocean
Australia
ANTARCTICA
Microsoft Bing
© 2024 Microsoft Corporation Terms



Sales Analysis

Clear All Slicers

Date Range: 3/27/2020 – 4/27/2023

Agents: All

Stores: All

Product Category: All

Product: All

Country

Revenue Year To Date: $11.96M

Total Revenue: $109.85M

Total Orders: 31,465

Total Units: 274,914

Revenues

Units

Orders

KPI Cards:

- Total Units
- Total Orders
- Total Revenue
- Orders Year To Date
- Units Year To Date
- Revenue Year To Date

Graphs(changing for revenue, orders and units sold):

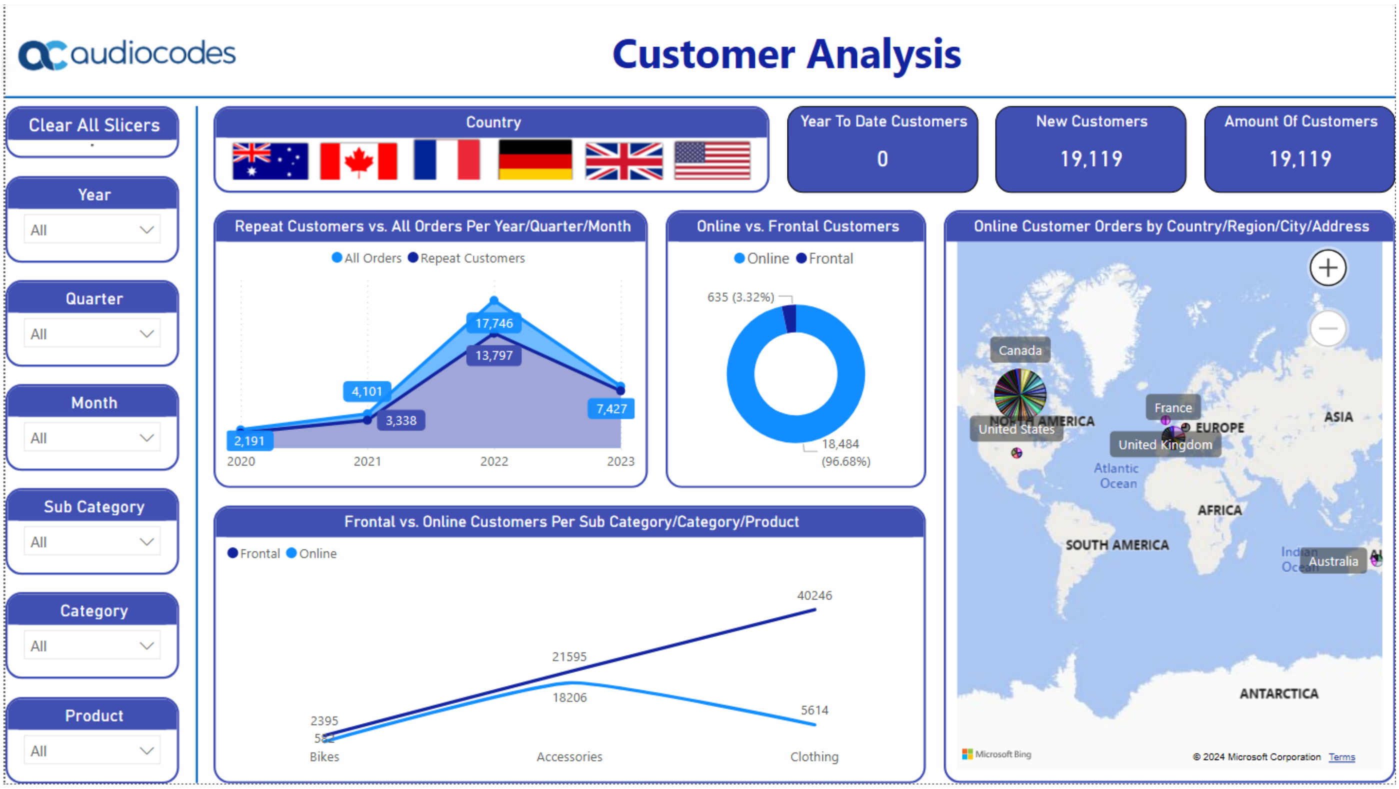
- Currently vs. Year over Year By Year/Quarter/Month(Drill Mode)
- Top 3 by Stores Per Year/Quarter/Month (Drill Mode)
- Total by Country/Region/City/Address (Drill Mode)

Slicers:

- Date Range
- Agents
- Stores
- Product Category
- Product
- Country

4.3.3.3. Customer Analysis:

This report presents for us the level of our accessibility to customers, the level of success among new customers and the levels and purchase online and frontal.



Customer Analysis

Year To Date Customers: 0

New Customers: 19,119

Amount Of Customers: 19,119

Year: All

Quarter: All

Month: All

Sub Category: All

Category: All

Product: All

KPI Cards:

- Amount Of Customers
- New Customers
- Year To Date Customers

Graphs:

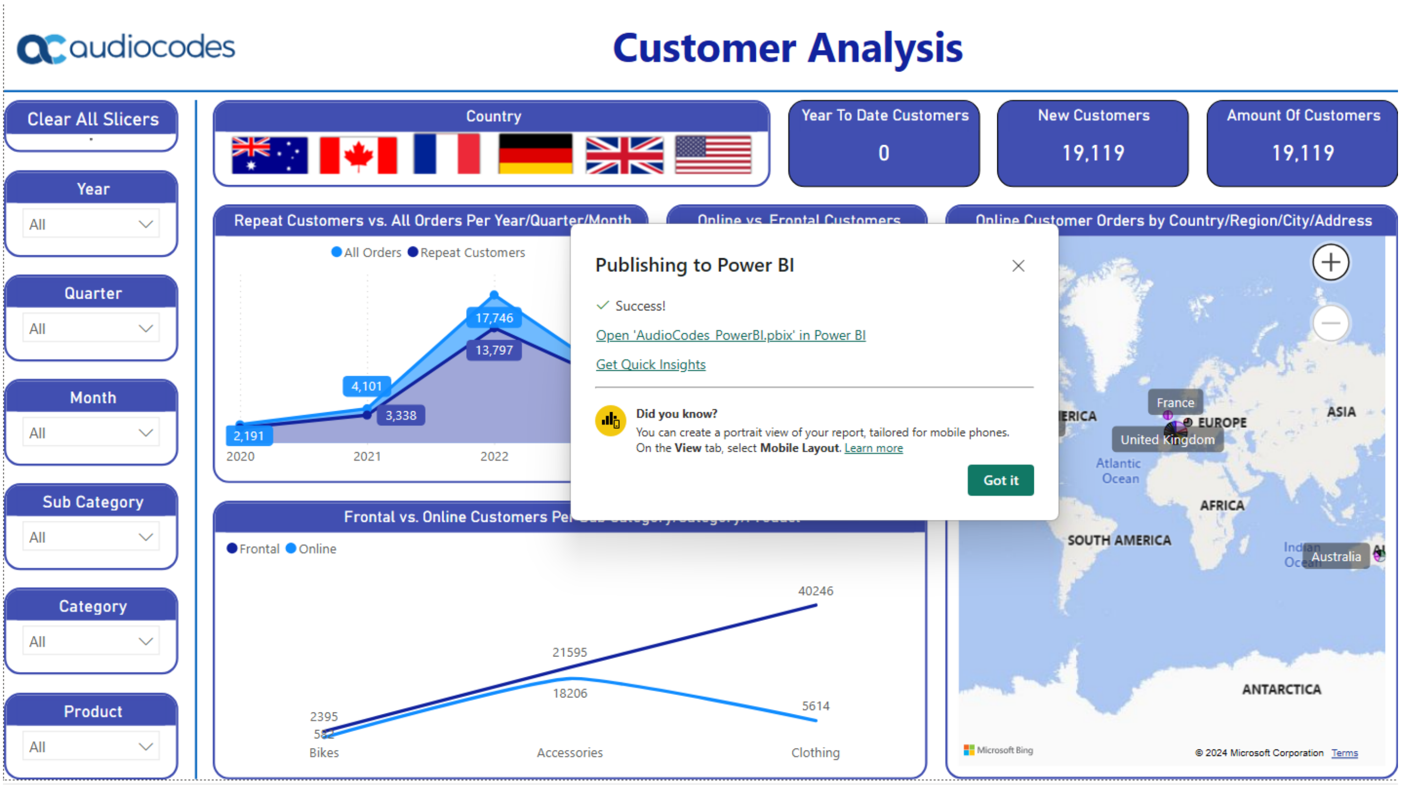
- Repeat Customers vs. All Orders Per Year/Quarter/Month (Drill Mode)
- Online vs. Frontal Customers
- Online Customer Orders by Country/Region/ City/ Address (Drill Mode)
- Frontal vs. Online Customers Per Sub Category/Category/Product (Drill Mode)

Slicers:

- Year
- Quarter
- Month
- Sub Category
- Category
- Product

4.3.4. After creating the reports in Power BI Desktop, they were published to Power BI Service,

And an app was created



Publishing to Power BI

✓ Success!

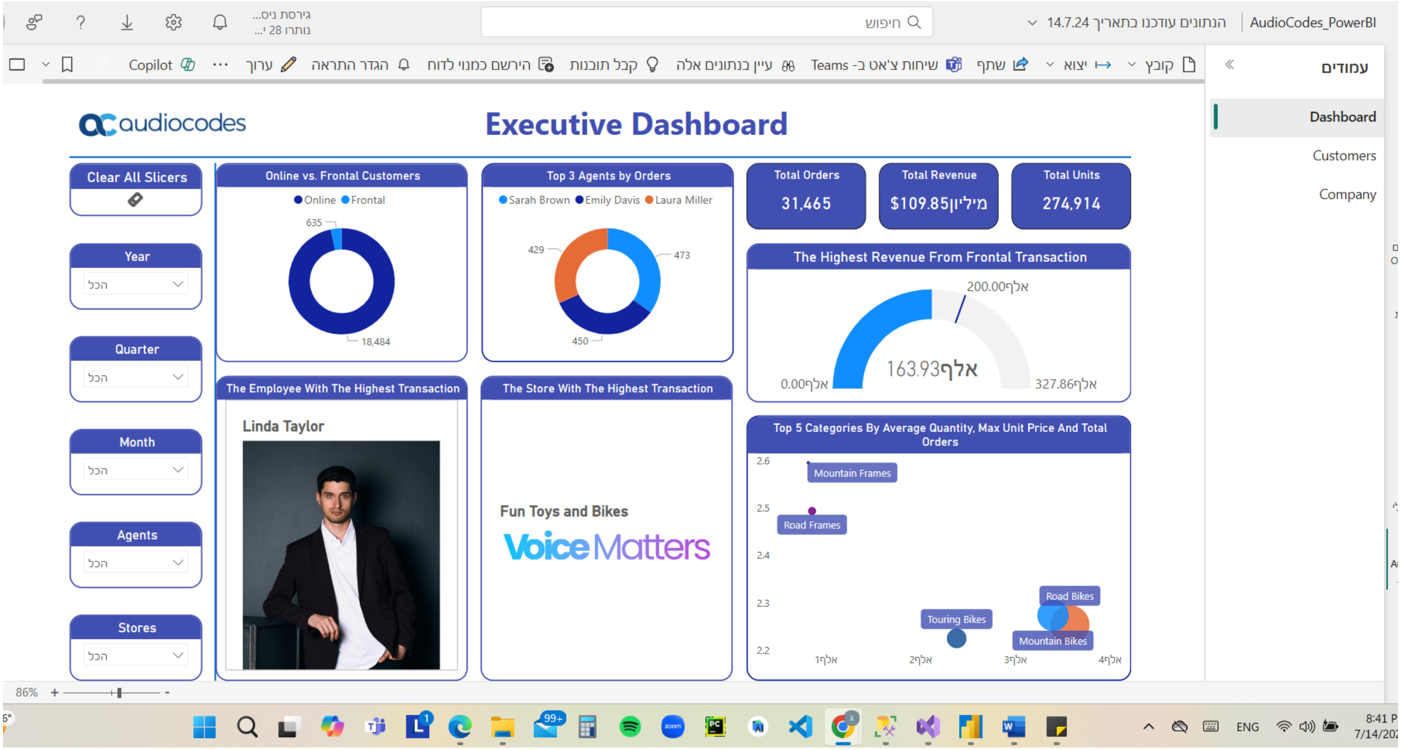
Open 'AudioCodes_PowerBI.pbix' in Power BI

Get Quick Insights

Did you know?

You can create a portrait view of your report, tailored for mobile phones. On the **View** tab, select **Mobile Layout**. Learn more

Got it



AudioCodes_PowerBI
14.7.24 הנתונים עודכנו בתאריך
חיפוש
קובץ
ייצוא
שתף
שיחות צ'אט ב- Teams
עיין בנתונים אלה
קבל תובנות
הירשם כמנוי לדוח
הגדר התראה
ערוך
Copilot
עמודים
Dashboard
Customers
Company
audiocodes
Executive Dashboard
Clear All Slicers
Year
הכל
Quarter
הכל
Month
הכל
Agents
הכל
Stores
הכל
Online vs. Frontal Customers
Online
Frontal
635
18,484
Top 3 Agents by Orders
Sarah Brown
Emily Davis
Laura Miller
429
473
450
Total Orders
31,465
Total Revenue
$109.85מיליון
Total Units
274,914
The Highest Revenue From Frontal Transaction
200.00אלף
163.93אלף
0.00אלף
327.86אלף
The Employee With The Highest Transaction
Linda Taylor
The Store With The Highest Transaction
Fun Toys and Bikes
VoiceMatters
Top 5 Categories By Average Quantity, Max Unit Price And Total Orders
2.6
2.5
2.4
2.3
2.2
Mountain Frames
Road Frames
Touring Bikes
Road Bikes
Mountain Bikes
1אלף
2אלף
3אלף
4אלף
86%
ENG
8:41
7/14/20