

# DECATHLON

## Specification Document

Sales Department Data Mart

BI33 Final Project

Date: 02.07.2024

Version 3 - Functional Specification

Alex Chagan



[Github](#)



[LinkedIn](#)

## Table of Contents

### 1. General

- 1.1. Project Objectives..... Page 2
- 1.2. Business Objectives..... Page 2
- 1.3. Project Contents..... Page 2
- 1.4. GANTT..... Page 2

### 2. Technical Specifications

- 2.1. Prerequisites..... Page 3
- 2.2. Source To Target..... Page 3
- 2.3. ERD..... Page 3
- 2.4. HLD..... Page 3
- 2.5. Analysis Measures and Reports..... Page 4

### 3. Functional Specification

- 3.1. ETL Overview..... Page 5
- 3.2. Solutions & Packages..... Page 5
- 3.3. General Notes..... Page 5
- 3.4. Mirror Stage..... Page 6
- 3.5. Staging Stage..... Page 8
- 3.6. Datamart Stage..... Page 10
- 3.7. History Table Products..... Page 11

### 4. Deployment and Job Scheduling

- 4.1. Jobs Overview..... Page 12
- 4.2. Jobs Definition..... Page 12
- 4.3. Jobs Scheduling..... Page 13
- 4.4. Development Environment..... Page 13

### 5. Visualization

- 5.1. Power BI Overview..... Page 14
- 5.2. Key Measures..... Page 15
- 5.3. Dashboard..... Page 16
- 5.4. Customers Analysis..... Page 18
- 5.5. Employees Analysis..... Page 19
- 5.6. App creation..... Page 20

## 1.1. Project Objective

This project aims to develop a data mart for the Sales Department of Decathlon, a leading sports retail company. The data mart will serve as a centralized repository for sales-related data, enabling efficient and effective data analysis and reporting for decision-making purposes.

## 1.2. Business Objectives

- In-depth analysis of sales data to identify trends, patterns, and opportunities for revenue growth.
- To optimize sales strategies, and provide insights into top-performing products, sales regions, and customer segments.
- Enable a comprehensive view of customer behavior, preferences, and purchasing patterns.
- Provide detailed analytics on product performance, including sales volume, revenue, and profitability.
- Identify top-performing sales representatives and provide insights into their strategies and techniques.

## 1.3. Project Contents

**Fact Sales** - captures detailed information about sales transactions, including the date, employee, customer, product, store, quantity, and revenue. It serves as the central table in the data mart.

**Dim Employees** - contains information about the employees in Decathlon's Sales Department. It includes employee ID, name, department, role, hire date, and location.

**Dim Customers** - holds customer-related information, including customer ID, name, address, and geographical details. It allows for customer segmentation and analysis to understand customer behavior.

**Dim Products** - provides details about the products sold by Decathlon. It includes product ID, name, category, subcategory, brand, and price attributes.

**Dim Stores** - contains information about the stores operated by Decathlon. It includes store ID, name, and location.

**Dim Products History** - functions as a repository of alterations to product-related attributes over time, preserving a chronological record of modifications such as updates, inserts, and deletes.

## 1.4. GANTT - [Link](#)

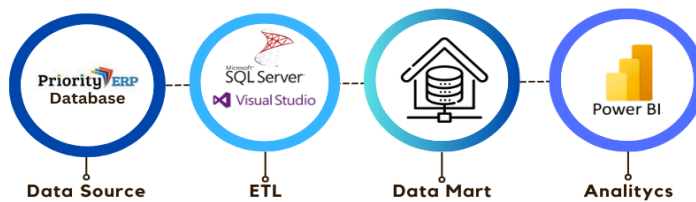
## 2.1. Prerequisites

System / Process	Description
SQL Server OLTP	Operational DataBase: PriorityERP
SSIS Path	C:\Users\alexc\source\repos\Decathlon\DecSalesDM\DecSalesDM.sln With all the Solution files for each DIM/Fact Table
SQL Server OLAP	Analytical Database (Data Mart): DecSalesDM
PowerBI	pbix file in GitHub: <a href="#">Link</a>

## 2.2. Source To Target - [Link](#)

## 2.3. ERD - [Link](#)

## 2.4. HLD:



Data collection and exploration from the ERP system will be performed in SQL Server. The data will undergo an ETL process for organization and arrangement into a Data Warehouse using Visual Studio's SSIS. Finally, the measures in reports and visuals will be presented in Power BI.

## **2.5 Analysis Measures and Reports**

The following reports are based on the last 2 years to generate the most accurate and relevant analysis.

### **Customer Analysis:**

- Amount of active customers by Country/Top 5 Regions/Top 5 Cities.
- Number of orders by Country/Top 5 Regions/Top 5 Cities.
- Total Sales and Quantity sold by Country.
- Total Sales and Quantity sold each month for the last year.
- Number of customers who made purchases for each Store.

### **Employee Performance Analysis:**

- Top 5 Employees by orders and total sales
- Total Sales Comparison with the previous year
- Quantity sold distribution in percentage by Product Category
- Top 5 Stores by Total Sales
- Total Sales Comparison between Physical Stores and Internet Stores

### **Visualization Dashboard:**

A visual representation of key business data and metrics, providing a consolidated and interactive view of critical information within an organization. an introduction to the underlying reports and semantic models. It contains key KPIs and graphs that represent the most important aspects of the report.

### 3.1 ETL Process Overview (SSIS)

The ETL process is designed to extract relevant data from diverse sources, transform it into a standardized format, and load it into the relevant tables. This streamlined approach ensures the availability of accurate and consistent data for analytical purposes.

- **Extraction:** Retrieve data from the source database, including sales transactions, store details, customer information, employee records, and product data.
- **Transformation:** Standardize, cleanse, and enrich the extracted data to ensure consistency and accuracy.
- **Loading:** Populate the FACT and DIM tables with transformed data, ensuring proper relationships to the data model.

### 3.2 Solutions and SSIS Packages

The ETL processes in SSIS consist of several Solution projects that deploy each stage in the workflow. Each Solution project contains the relevant packages for the deployment purposes of the given stage.

Hierarchy of Solution and Packages:

DecSalesDM\_MRR\_ETL

- MRR\_DIMS - Creation of mirror tables for dimension processes
- MRR\_FACTS - Creation of mirror tables for fact purposes

DecSalesDM\_Stores\_ETL

- STG\_Stores
- DIM\_Stores

DecSalesDM\_Customers\_ETL

- STG\_Customers
- DIM\_Customers

DecSalesDM\_Employees\_ETL

- STG\_Employees
- DIM\_Employees

DecSalesDM\_Products\_ETL

- STG\_Products
- DIM\_Products

## 3.3 General Notes

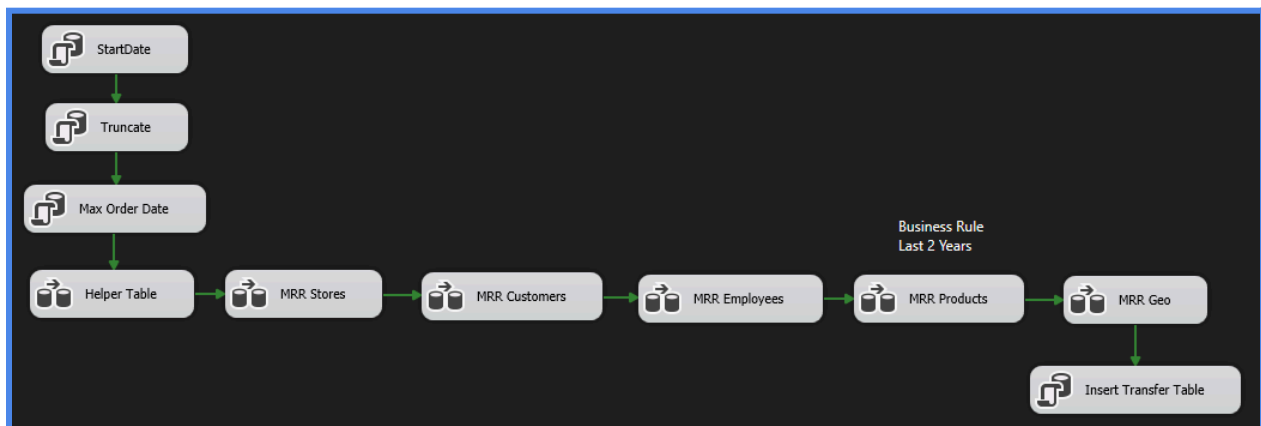
- For each package, we record the date and time of the execution's beginning and end, inserting them into the transfer table for better documentation.
- We also insert the number of rows transferred in each step into the transfer table.
- For the mirroring and staging packages, we perform a truncation on the relevant tables.

## 3.4. Mirror Stage

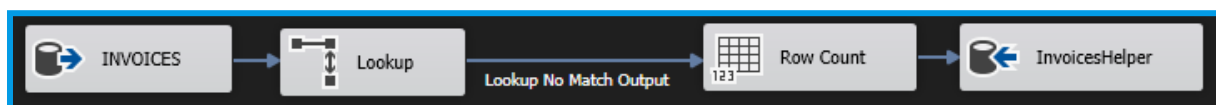
The primary objective of the mirror stage is to replicate and temporarily store raw data exactly as it exists in the source systems. This ensures the integrity of the original data.

### Dimension-related Tables Mirroring:

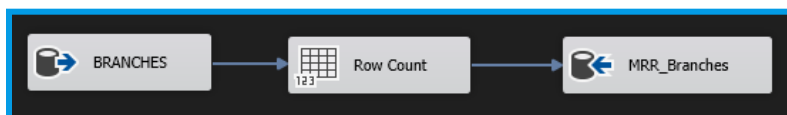
#### Control Flow:



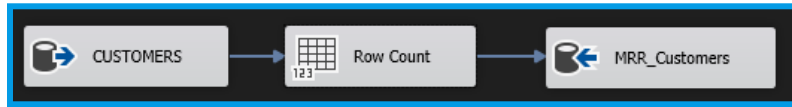
#### Data flows:



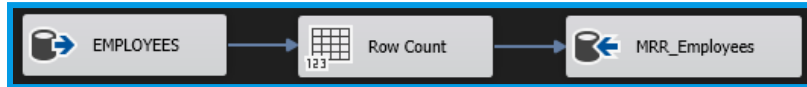
Helper table that saves data from the invoices table of the source database. We only take new data that isn't already in the helper table. In addition, we don't truncate this table because we need all the data to create and update some of the dimension staging tables.



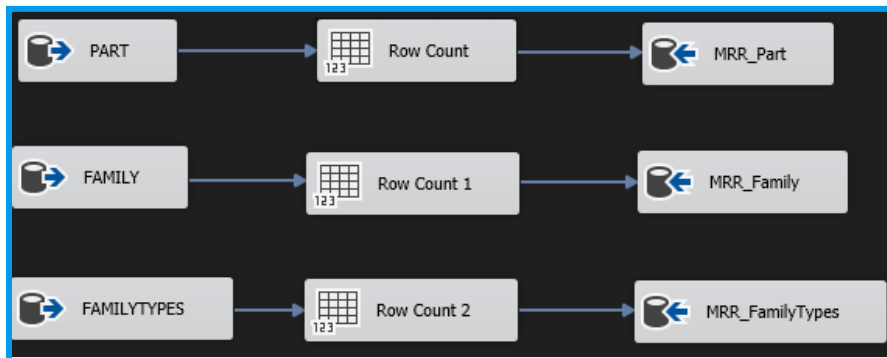
The Stores dimension-related tables from the source database



The Customers' dimension-related tables from the source database

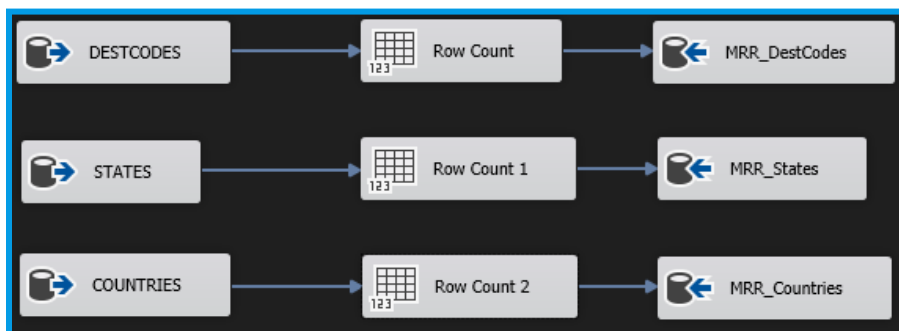


The Employees' dimension-related tables from the source database



The Products' dimension-related tables from the source database

- **MRR\_Part** is affected by a business rule in the form of an SQL query. Only products that were sold in the last 2 years are inserted into the mirror table.



The geography-related tables from the source database

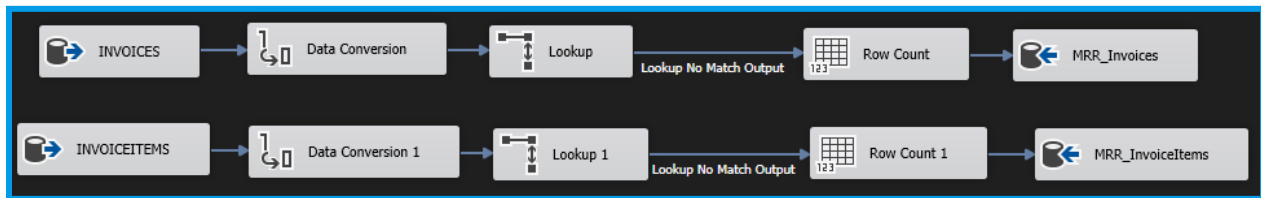


## Fact-related tables mirroring

### Control Flow:



### Data Flow:



## The Fact/Sales related tables from the source database

- We use lookup only to extract data that isn't already in the Fact Sales table.

### 3.5. Staging Stage

After completing the mirror phase, the ETL process transitions into the staging phase. The data is refined, standardized, and prepared for the subsequent transformations in the ETL process. Staging acts as a critical bridge, ensuring that the data is consistent and reliable before being loaded into the datamart for further analysis and reporting.

For each Dimension/Fact Staging table, we use a JOIN SQL query to retrieve the relevant columns from the correlating mirror tables.

### Data Task pipelines



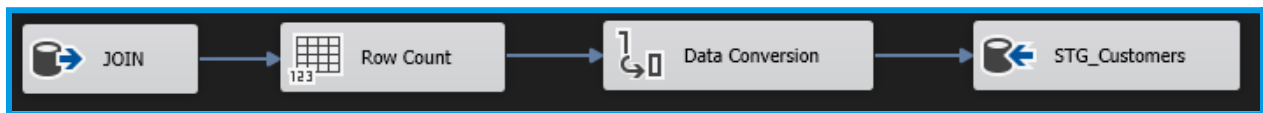
## Data Flows:

- **Stores:**



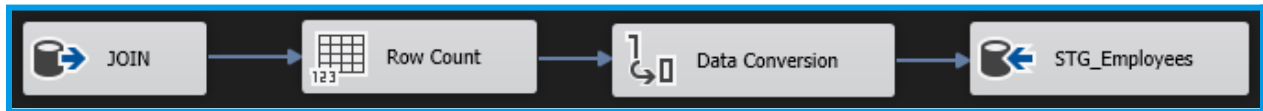
Joined Tables: MRR\_Branches, MRR\_States, MRR\_Countries

- **Customers:**



Joined Tables: MRR\_Customers, MRR\_DestCodes, MRR\_States, MRR\_Countries, InvoicesHelper

- **Employees:**



Joined Tables: MRR\_Employees, InvoicesHelper

- **Products:**



Joined Tables: MRR\_Part, MRR\_FamilyTypes, MRR\_Family

- **Sales:**



Joined Tables: MRR\_Invoices, MRR\_InvoiceItems

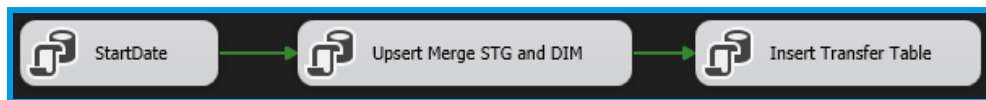
## 3.6. DataMart Stage

The Data Mart is a specialized subset of the overall data warehouse, tailored to meet the specific analytical needs of the sales department. In this stage, data is further transformed, aggregated, and organized into dimensional models that align with the business requirements of the sales team.

### Upsert (Merge) Approach

When applied to the transition from the staging table to the dimension table, this method involves comparing the data in the staging table with the existing records in the dimension table. Rows that match based on certain criteria (e.g., primary keys or unique identifiers) are updated with the latest information from the staging table, ensuring that the dimension table reflects the most recent and accurate data.

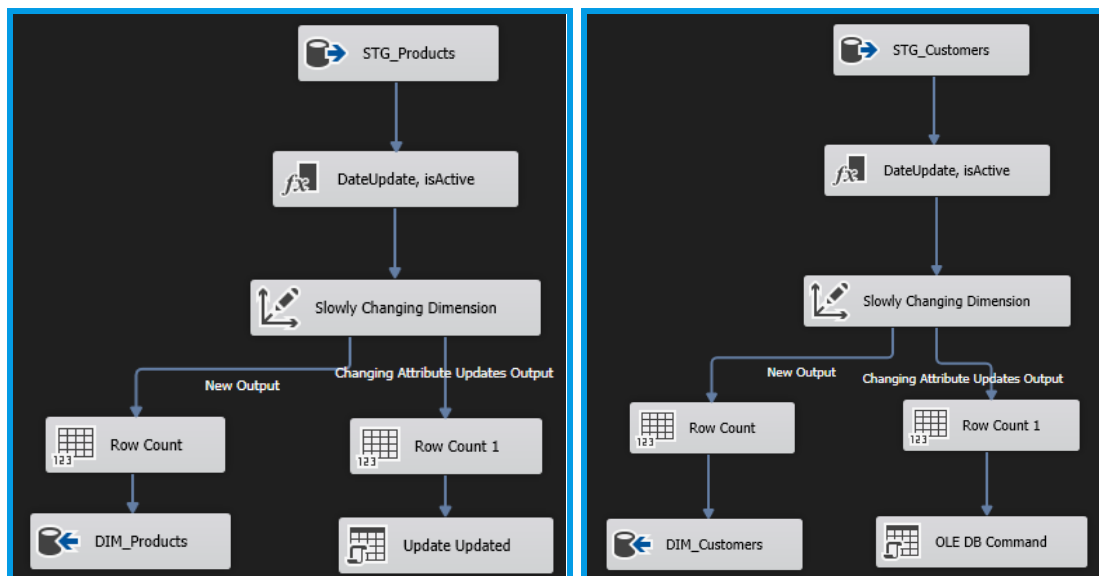
### Stores Dimension & Employees Dimension



### Slowly Changing Dimension Approach

The Slowly Changing Dimension approach ensures that changes in dimension attributes are handled appropriately, aligning the data mart with business requirements and enabling users to analyze data trends over time. This meticulous management of dimension changes enhances the data mart's ability to provide accurate and comprehensive insights for decision-making within the specific context of the sales department.

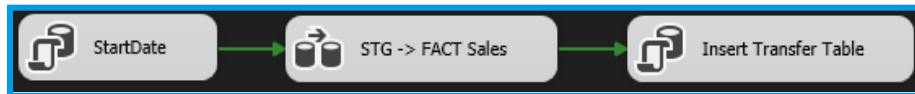
### Products Dimension & Customers Dimension



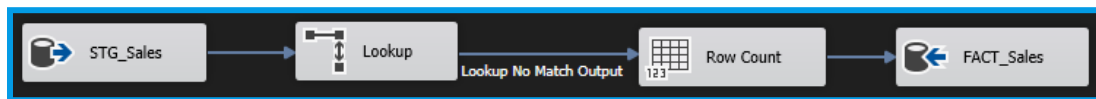
## Fact Sales

The population of the fact table is performed by a simple lookup transformation that filters irrelevant data (e.g. Data that is already recorded in the fact table).

### Control Flow:

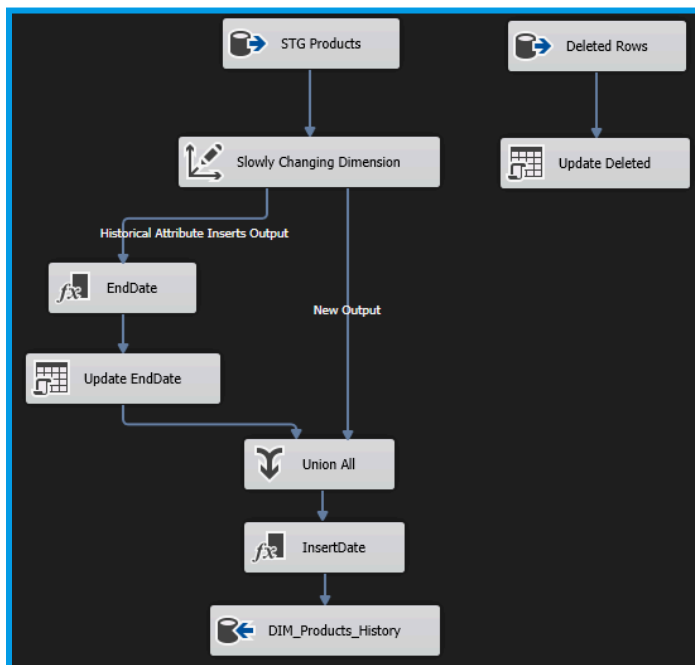


### Data Flow:



## 3.7. History Table for Products Dimension

Utilizing the SCD methodology, the History Table retains historical versions of product attributes, allowing for the tracking of changes such as updates, inserts, and deletions. Each row in the table represents a specific version of a product, complete with effective dates that delineate when the information became valid.



## 4.1. Jobs Overview

SQL Agent jobs consist of one or more steps, each representing a specific action or set of actions. These jobs can include tasks such as data extraction, transformation, loading, backups, and other database maintenance activities. The scheduling capabilities of SQL Agent enable users to plan and automate these tasks, reducing manual intervention and ensuring timely and consistent execution.

These SQL Agent jobs collectively form a robust and automated ETL process, enabling the Sales Department to harness the power of data for informed decision-making and strategic insights.

## 4.2. Jobs Definition

### ETL\_Mirror:

Step	Name
1	MRR_DIMS
2	MRR_FACTS

### ETL\_Sales:

Step	Name
1	STG_Sales
2	FACT_Sales

### ETL\_Stores:

Step	Name
1	STG_Stores
2	DIM_Stores

### ETL\_Complete

Step	Name
1	MRR_DIMS
2	MRR_FACTS
3	STG Stores
4	DIM Stores
5	STG_Employees
6	DIM_Employees
7	STG_Products
8	DIM_Products
9	STG_Customers
10	DIM_Customers
11	STG_Sales
12	FACT_Sales

### ETL\_Customers:

Step	Name
1	STG_Customers
2	DIM_Customers

### ETL\_Employees:

Step	Name
1	STG_Employees
2	DIM_Employees

### ETL\_Products:

Step	Name
1	STG_Products
2	DIM_Products

## 4.3. Jobs Scheduling

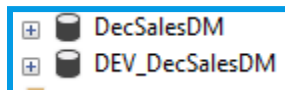
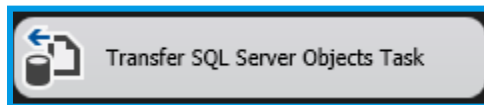
The scheduling capabilities of SQL Agent enable users to plan and automate these tasks, reducing manual intervention and ensuring timely and consistent execution.

ID	Name	Enabl...	Description
1014	Morning Update	Yes	Occurs every day at 8:00:00 AM. Schedule will be used starting on 1/4/2024.

## 4.4. Development Environment

I established a development environment after confirming that the ETL process was functioning end-to-end in the production environment and that all the data was accurate and transferred as planned into the dimension and fact tables. I copied all the tables, along with the production data, to this new environment. This allows additional development work in the new environment without impacting the already deployed and operational production environment.

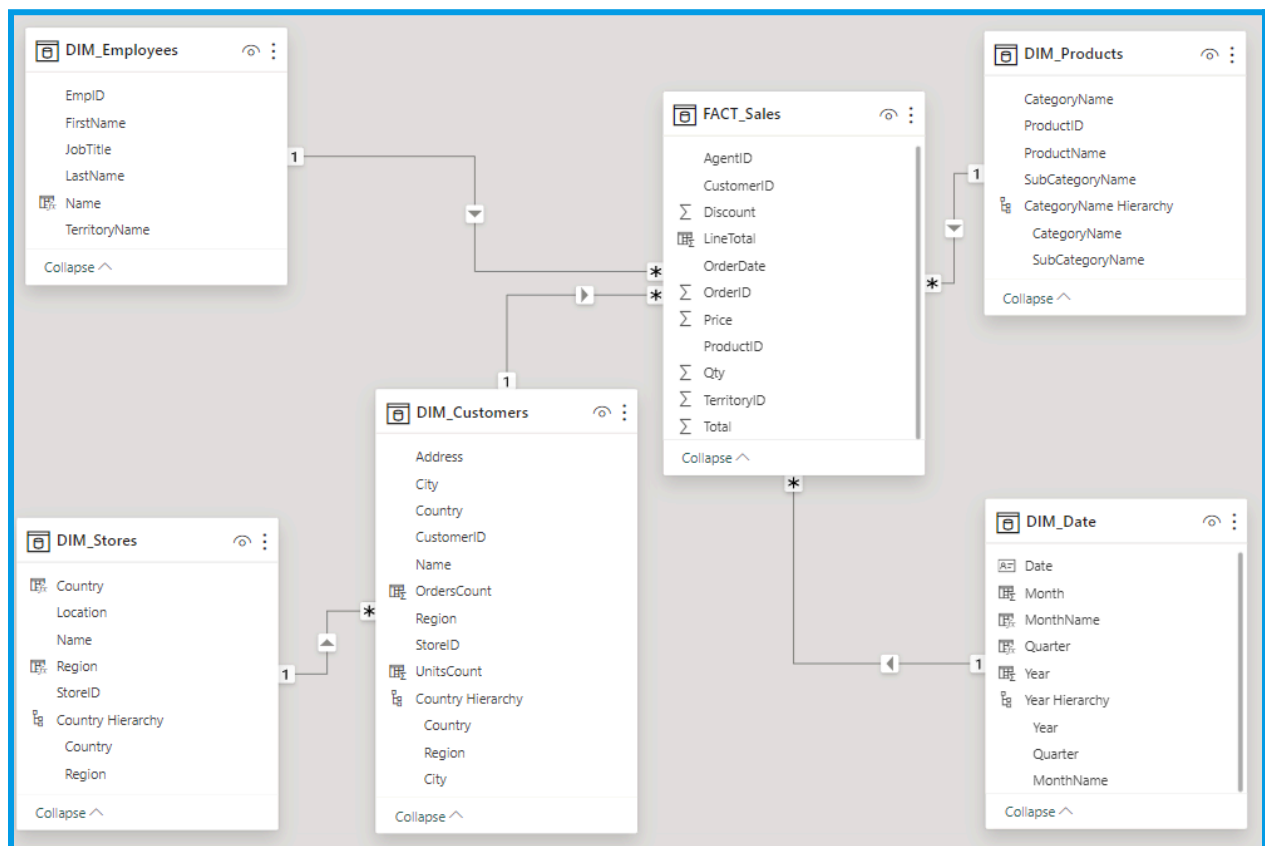
This is done with the following task in SSIS:



## 5.1. PowerBI Overview

For this project, I used Power BI to create visualizations. These visualizations include a dashboard, a customer analysis report, and another for employee analysis. After completing the project, I deployed it to the Power BI Service and organized it into an app. This setup allows easy access and navigation for stakeholders, providing valuable insights to Decathlon's sales team for improving performance and decision-making.

### The PowerBI Model



- **DIM\_Date** was created with the `CALENDERAUTO()` expression. Year, Quarter, and Month columns were derived from the automated date column.

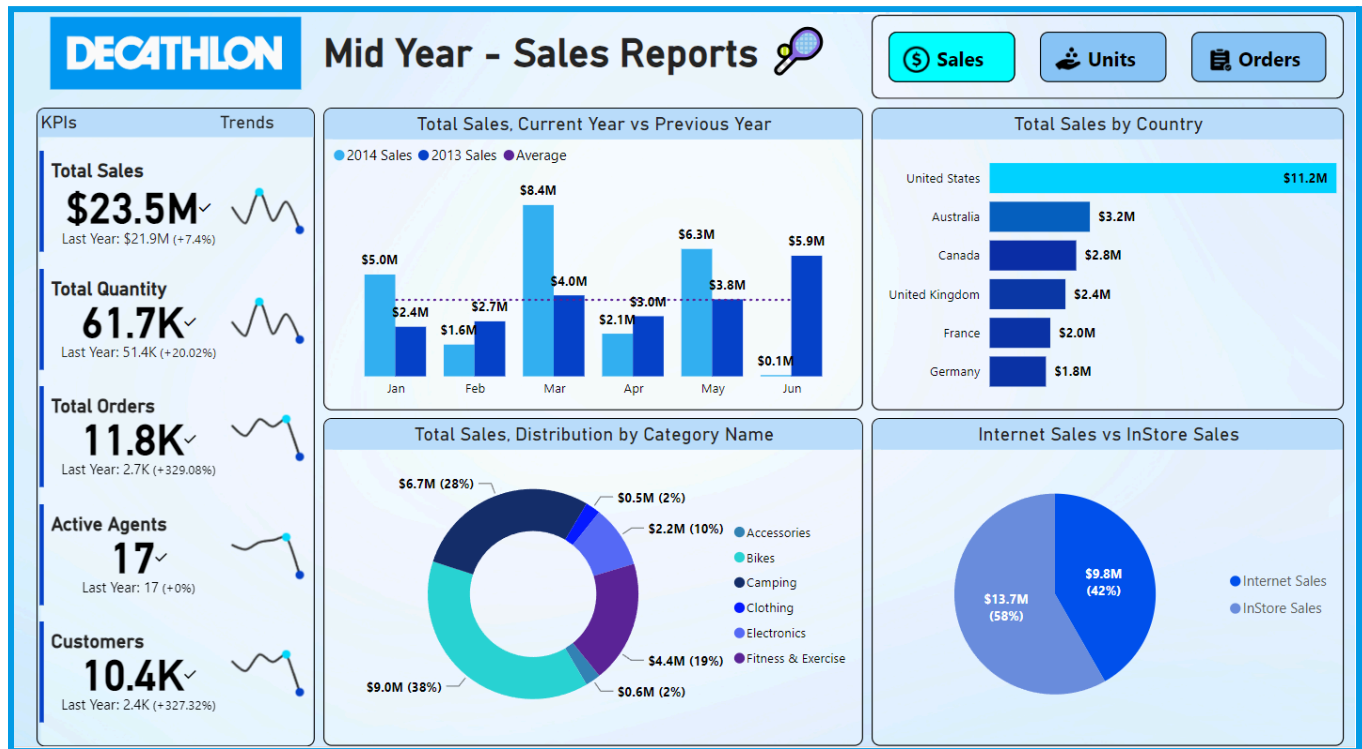
## 5.2. Key Measures

- Total Sales = `CALCULATE(SUM(FACT_Sales[LineTotal]))`
- Total Units = `CALCULATE(SUM(FACT_Sales[Qty]))`
- Total Orders = `DISTINCTCOUNT(Fact_Sales[OrderID])`
- YTD Orders = `TOTALYTD([Total Orders], DIM_Date[Date])`
- YTD Units = `TOTALYTD([Total Units], DIM_Date[Date])`
- YTD Orders = `TOTALYTD([Total Orders], DIM_Date[Date])`
- LY YTD Sales =  
  
`CALCULATE(  
  
[Total Sales],FILTER(ALL(FACT_Sales),  
  
FACT_Sales[OrderDate]>=DATE(2013,1,1)&&FACT_Sales[OrderDate]<DATE(2013,6,30)))`
- LY YTD Units =  
  
`CALCULATE(  
  
[Total Units],FILTER(ALL(FACT_Sales),  
  
FACT_Sales[OrderDate]>=DATE(2013,1,1)&&FACT_Sales[OrderDate]<DATE(2013,6,30)))`
- LY YTD Orders =  
  
`CALCULATE(  
  
[Total Orders],FILTER(ALL(FACT_Sales),  
  
FACT_Sales[OrderDate]>=DATE(2013,1,1)&&FACT_Sales[OrderDate]<DATE(2013,6,30)))`
- LY Sales % = `[YTD Sales]/[LY YTD Sales]`
- InStore Sales = `CALCULATE([Total Sales], Dim_Employees[EmpID] <> 77777)`
- Internet Sales = `CALCULATE([Total Sales], Dim_Employees[EmpID] = 77777)`
- AVG Units Per Order = `[Total Units]/[Total Orders]`
- AVG Price Per Order = `[Total Sales]/[Total Orders]`
- PM Total Sales = `CALCULATE([Total Sales],DATEADD(DIM_Date[Date], -1, MONTH))`
- PM Sales Diff = `[Total Sales] - [PM Total Sales]`
- PM Sales Diff % = `IF(AND([Total Sales],[PM Total Sales]), 1-([PM Total Sales])/[Total Sales])`
- Agent Sales Rank = `RANKX(ALL(DIM_Employees[Name]), [Total Sales])`



## 5.3. Dashboard

### Sales Bookmark:

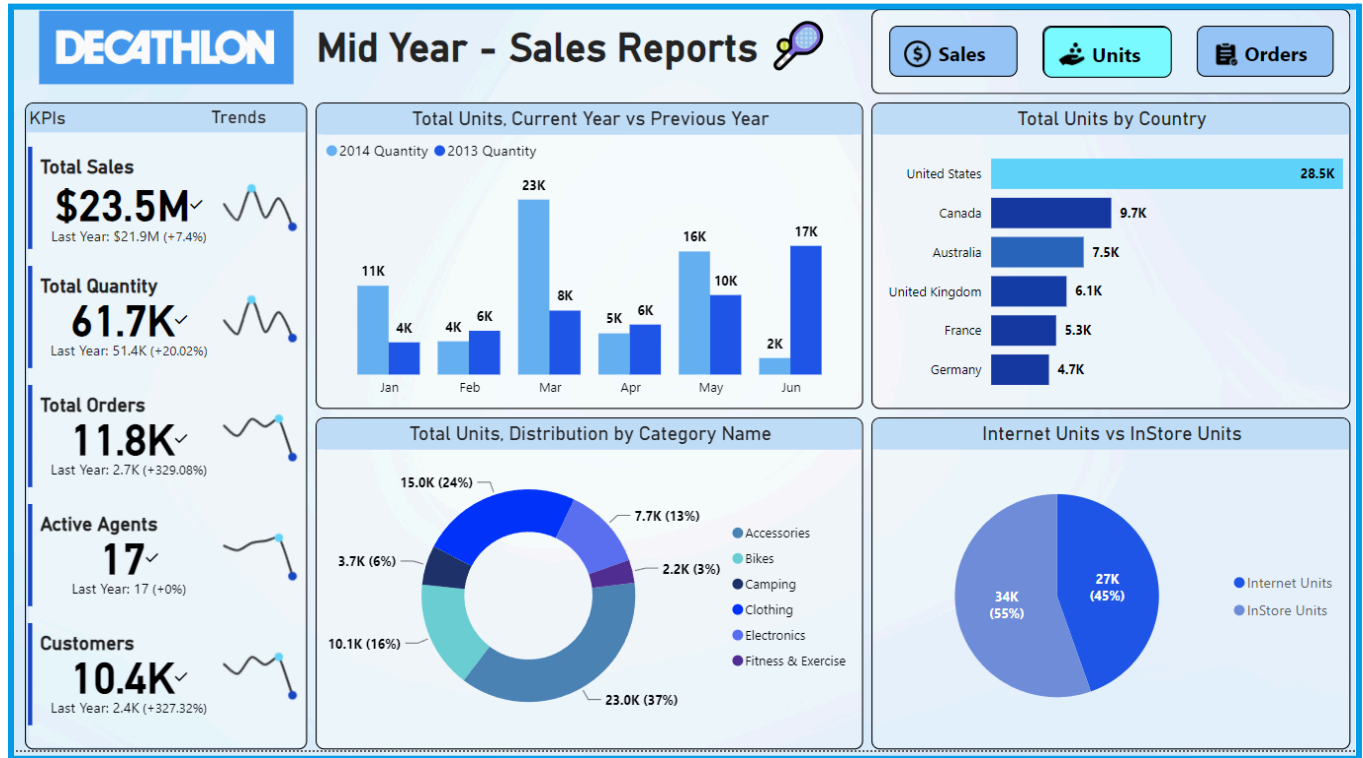


A main dashboard that contains the most important features from both reports.

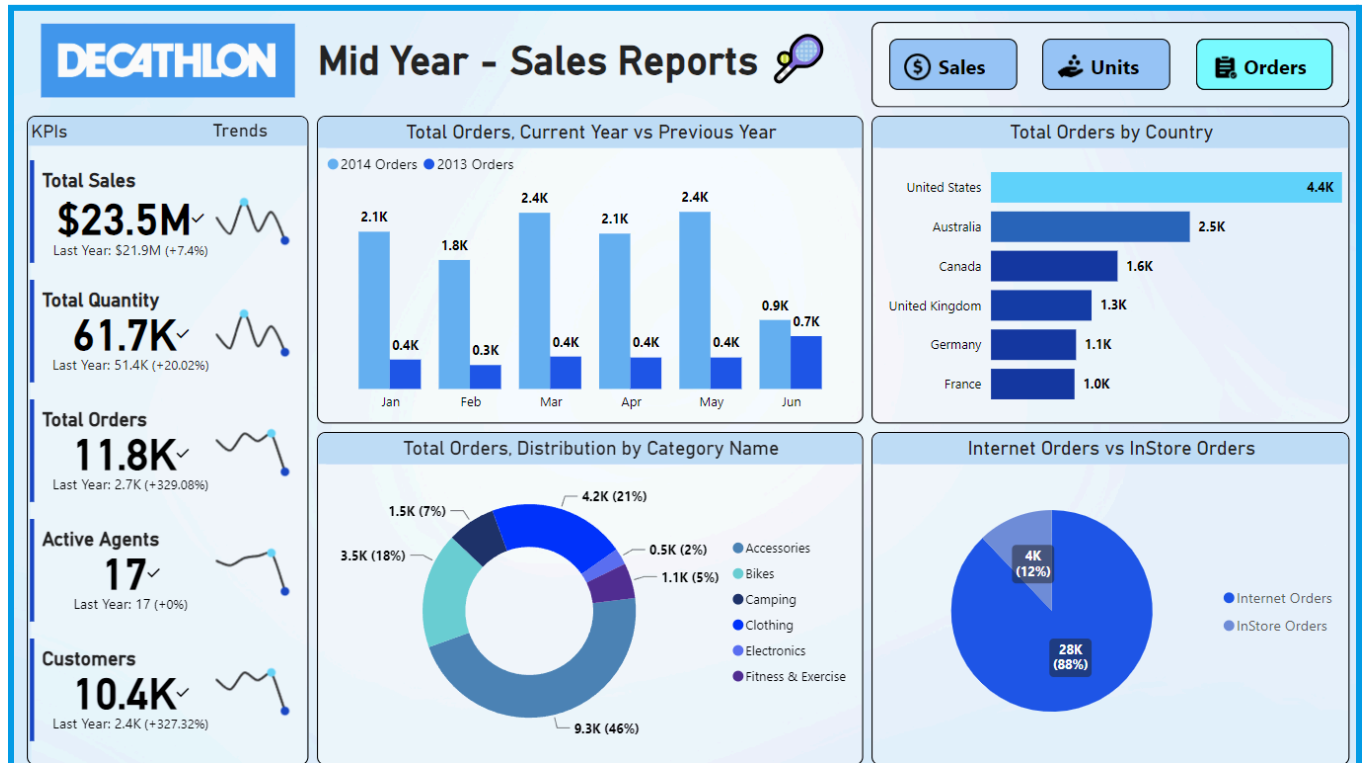
### The Dashboard contains:

- The most important KPIs for the sales department as well as sparklines that show sales trends per month in the current year.
- Graph: Year to Date Sales/Units/Orders Comparison between current year and last year.
- Graph: Total Sales/Units/Orders by Country.
- Graph: Distribution Donut by Category name based on Sales/Units/Orders.
- Graph: Pie that contains a comparison in Sales/Units/Orders between Online Sales and Physical Sales, showing values and percentages.
- Buttons that allow you to switch between Sales/Units/Orders bookmarks.

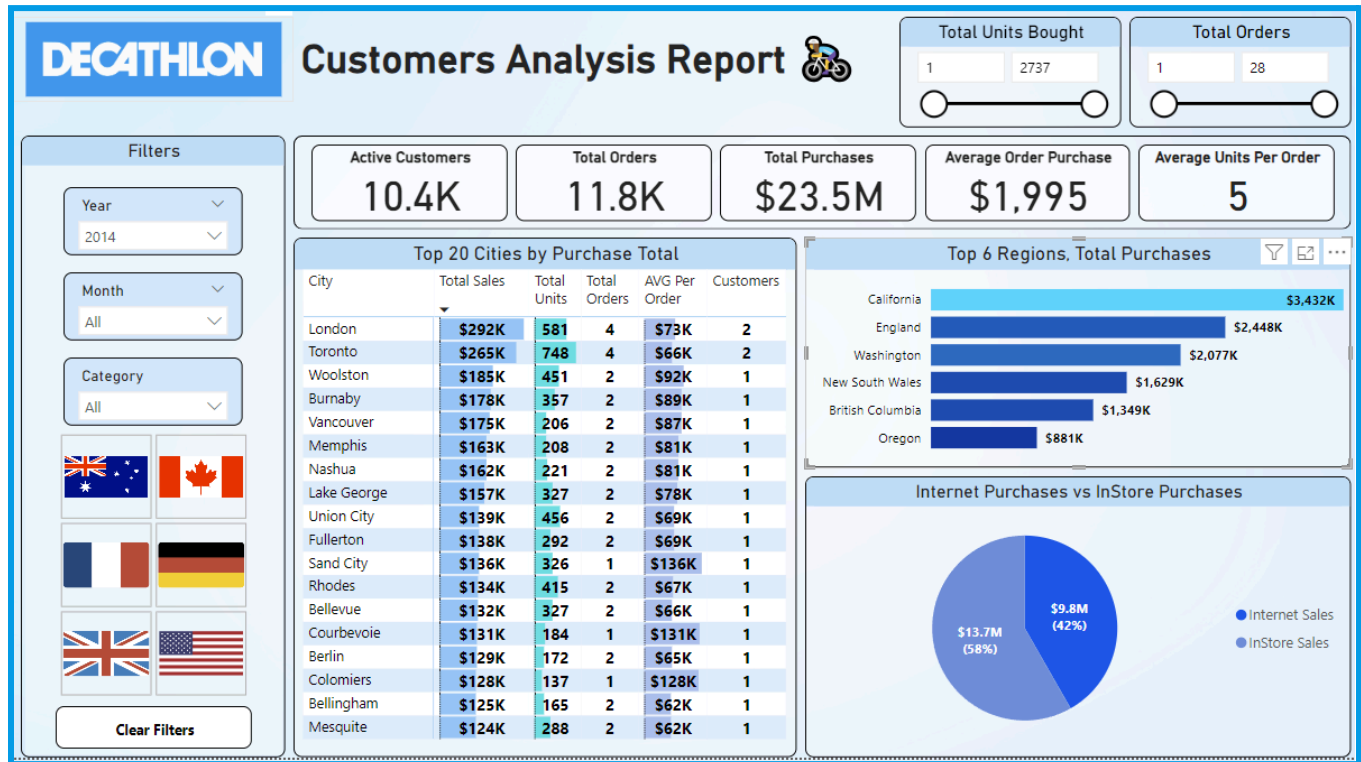
## Units Bookmark:



## Orders Bookmark:



## 5.4. Customers Analysis Report

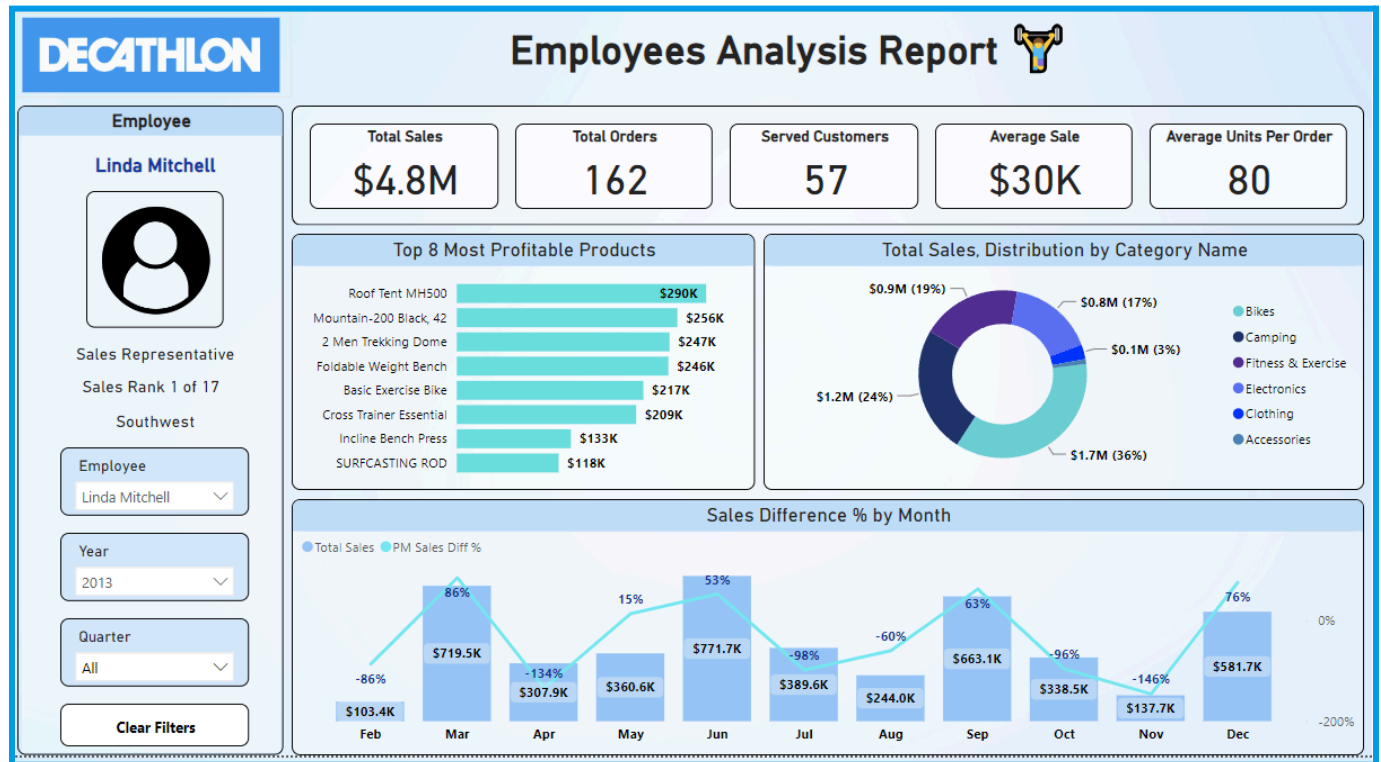


This Power BI report offers comprehensive customer analysis, detailing sales data across countries, regions, and cities while distinguishing between Internet and in-store purchases. Key performance indicators (KPIs) are included as well.

This report contains:

- Filters: Year, Month, Category + SubCategory Hierarchy, Country, Total Units and Total Orders ranges.
- Most relevant KPIs for Customer analysis purposes.
- Graph: A Matrix containing all the relevant data on customer activity per city.
- Graph: Top 6 Regions by total customer purchases.
- Graph: A Pie comparing Internet purchases and Physical purchases by value and percentage.

## 5.5. Employees Analysis Report



This Power BI report offers detailed employee analysis, showcasing individual performance metrics such as name, rank by total sales, and location. It provides insights into the most profitable products sold by each employee, distribution of sales across product categories, and visualizes sales variations between consecutive months. Key performance indicators (KPIs) are included as well.

This report contains:

- Filters: Agent name, Year, Quarter
- Most relevant KPIs for Employee analysis purposes.
- Graph: Top 8 most profitable products per agent.
- Graph: Distribution Donut by Category name based on Sales per agent.
- Graph: Sales difference between every 2 months and sales value for each month. per agent.
- Agent details: Name, Role, Rank based on Sales, Location.

## 5.6. App Creation

- The Project was published in the Power BI Service under the project workspace.
- A Gateway was established as well as a connection to the DataMart.

**Gateway**      **Department**      **Contact information**      **Status**      **Actions**

Personal Gateway      Running on DESKTOP-PGRV4DA

**Cloud connections**

No cloud connections

Apply      Discard

**Data source credentials**

DecSalesDM-DESKTOP-PGRV4DA      [Edit credentials](#)      [Show in lineage view](#)

- An app was created in the workspace containing the Project's reports.



- Data refresh every day at 8:00 AM Jerusalem Timezone

Refresh frequency

Daily ▾

Time zone

(UTC+02:00) Jerusalem ▾

Time

8 ▾ 00 ▾ AM ▾ ×