April 03, 2020

CSCI 5408 – Data Management, Warehousing, and Analytics

Case Study

DALHOUSIE UNIVERSITY

Faculty of Computer Science

Submitted By:
Group 29
Shah Moni B00830791
Itekela Satya kumar B00839907
Parmar Parth B00853913

TABLE OF CONTENTS

LIST OF TABLES AND FIGURES

1. INTRODUCTION

The importance of data processing includes increased productivity and profits, better decisions, more accurate and reliable. Further cost reduction, ease in storage, distributing, and report making, followed by better analysis and presentation, are other advantages.[1] A dataset is essential for several reasons, improving future performance, understanding your customers better. In other words, a quality data set leads to accurate sales insights.[2] Data is important only if it can be interpreted and analyzed. Every business has lots of data and information within its data warehouses and systems and software solutions but, without a way to understand the data, it is useless.[1]

A **sales dataset report** shows the trends that occur in a company's sales volume over time. In its most basic form, a sales report shows whether sales are increasing or declining. At any time during the fiscal year, sales managers may analyze the trends in the report to determine the best course of action.[3] In this case study, we have analyzed the dataset, extracted the dataset, cleaned and transformed the dataset, created a fact and dimension tables, created ER Diagram, loaded the data into a database, connected the database with Cognos BI and then performed Visualization in the Cognos BI tool. **Cognos** is IBM's **business intelligence** (**BI**) and performance management software suite. The software enables business users without technical knowledge to extract corporate data, analyze it, and assemble reports.[4]

2. DATA GATHERING AND TRANSFORMING

2.1 DATASET OVERVIEW

The “Sample Sales Data” dataset conveys information about the sales data of a company and is inspired for retails analytics. This dataset was created by María Carina Roldán and is licensed under the Creative Commons Attribution-Non-Commercial-Share Alike 3.0 Unported License. The dataset contains the following attributes:

ORDERNUMBER – It represents the order number of an order in the sales. This is a number attribute.

QUANTITYORDERED – It represents the total number of quantities ordered for a particular product in an order. This is a number attribute.

PRICEEACH – It represents the price of each product. This is a decimal attribute.

ORDERLINENUMBER – It represents an individual order line for the order. This is a number attribute.

ORDERDATE – It represents the order date of an order. This is a date attribute.

SALES – It represents the sales i.e., PRICEEACH*QUANTITYORDERED of a product in an order. This is a float attribute.

STATUS – It represents the order status whether it is shipped, delivered or pending. This is a string field.

QTR_ID – It represents the quarter in which the order was placed. This is a integer attribute.

MONTH_ID – It represents the month in which the order was placed. This is an integer attribute.

YEAR_ID – It represents the year in which the order was placed. This is an integer attribute.

PRODUCTLINE – It represents the product type. This is a string field.

MSRP – It represents the maximum selling price for a product. This is an integer attribute.

PRODUCTCODE – It represents a unique code assigned to every product. This is a string field.

CUSTOMERNAME – It represents the name of the customer who placed the order. This is a string field.

PHONE – It represents the phone number of customer who placed the order. This is a string field.

ADDRESSLINE1– It represents the addressline1 of the customer who placed the order. This is a string field.

ADDRESSLINE2 – It represents the addressline2 of the customer who placed the order. This is a string field.

CITY– It represents the city in which the order was placed. This is a string field.

STATE – It represents the state in which the order was placed. This is a string field.

COUNTRY – It represents the country in which the order was placed. This is a string field.

POSTALCODE – It represents the postal code of the customer address. This is a string field.

TERRITORY – It represents the territory of the country in which order was placed. This is a string field.

CONTACTLASTNAME – It represents the last name of the contact person for the customer. This is a string field.

CONTACTFIRSTNAME – It represents the first name of the contact person for the customer. This is a string field.

DEALSIZE – It represents the deal size bases on the order quantity volume. This is a string field.

2.2 Data Extraction and Transformation

Data cleaning is the initial and vital step in preprocessing to extract cleaned data for further processing. It is essential to apply data extraction before data cleaning on raw log data in the analysis. The main objective to clean data is to handle the missing data through the identification

of the noises and hence to remove outliers and resolve inconsistencies.[5] After analysis of the entire dataset, we have shortlisted many columns that require cleaning.

Firstly, the values provided in the SALES column in the dataset are inaccurate. Its value is not equal to PRICEEACH*QUANTITYORDERED in some orders. We had considered cleaning this column first, but then we thought that there might be shipping or delivery charges added in this amount. So, if we clean this column, the data integrity would get hindered, and hence, we kept the values of the column as it is.

The next column is ORDERDATE, where the date format needs to be changed. With the help of strptime and strftime functions in python, we formatted the date in MM-DD-YYYY format. Columns ADDRESSLINE1 had special characters and Latin characters in it. So, to overcome this, we decided to clean the column using regex to remove special characters in python. Also, in column ADDRESSLINE2, there were only 300 values in the entire dataset, whereas rest all the values were NULL. As a lot of space was unused, which was not needed. To that, we merged ADDRESSLINE1 and ADDRESSLINE2 and created a new column ADDRESSLINE. Cleaning the ADDRESSLINE1 and then merging the two columns helped in making the dataset more efficient.

The PHONENUMBER column in the dataset had a variety of characters like (-, .). But as every country has its format to represent a PHONENUMBER, so we didn't clean this column as the data would have been changed. The same was with the column named CONTACTLASTNAME. There was an issue with the TERRITORY column in which Japan country had its territory as Japan. The territory of Japan is APAC (Asia – PACific countries). We thought that by changing the territory of Japan, the data would be more efficient as well as the integrity would be maintained. So, we changed the territory of Japan to APAC from Japan. We changed the values in this column by substitute method and using the regex library in python.

3. DATA WAREHOUSE CONSTRUCTION

3.1 DATA WAREHOUSE OVERVIEW

Data and analytics have become an indispensable part of any business to stay competitive in the current market. The organization uses reports, dashboards, and analytics tools to understand the various performance metrics, extract insights from their data, and make informed decisions to grow their business. These functionalities are made available by data warehouse, which manages data for efficient storage and retrieval. [6]

The Data warehouse is a central storage facility with information gathered from many different sources, which can be further analyzed to support decision making and to identify various trends. It contains heterogeneous data from different sources such as transactional systems, relational databases, market feeds, and multiple operational databases. To make future predictions based on past trends require historical data, which is not present in operational databases as they only store current data. To support multidimensional data models and on-line analytical processing (OLAP) operations require special data organization, access methods, implementation methods, which are

not available in commercial database management systems (DBMSs) targeted for on-line transactional processing (OLTP). [7] Therefore, data warehouses are implemented separately from operational databases.

3.2 Database and Data Warehouse

The following table iterates through the differences between the data warehouse and database:

Table 1 - Difference between Data Warehouse and Database

Data Warehouse	Database
Supports On-line analytical processing (OLAP)	Supports on-line transactional processing (OLTP) applications
Targeted for decision support and to identify various trends	The OLTP applications automate tasks such as order entry, banking transactions
Stores large quantities of historical data	Stores current transactions
Enables fast and complex queries across all the data using OLAP	Enables fast access to specific transactions for ongoing business process (I.e. OLTP)
Support a limited number of concurrent users compared to operational systems	Supports thousands of concurrent users (Number may vary based on the system users)

3.3 Building Data Warehouse

Dimensional modelling is the design concept used by many data warehouse designers to build data warehouse systems. It uses the concept of facts and dimensions where the fact is an aggregated numerical value, and dimension is a descriptor defining the fact. [8] A data warehouse uses either star, snowflake, or fact constellation schema to represent the entire database logically. Irrespective of schema, there are mainly two types of tables, a fact table, and a dimension table. The fact table consists of aggregated values such as metrics, measurements, or facts for a given business process, and their associated dimensions. The dimension table consists of attributes that describe the objects in a fact table.

The following are the steps to design a schema for the given sales dataset [4]:

1. **The Initial Star Schema**
   We initially identified various dimension tables and a fact table from the cleaned and transformed dataset. ***Customer***, ***location***, ***product***, and ***time*** are considered as dimension tables, whereas ***sales*** was defined as a fact table. The initial star schema was designed using the identified tables:

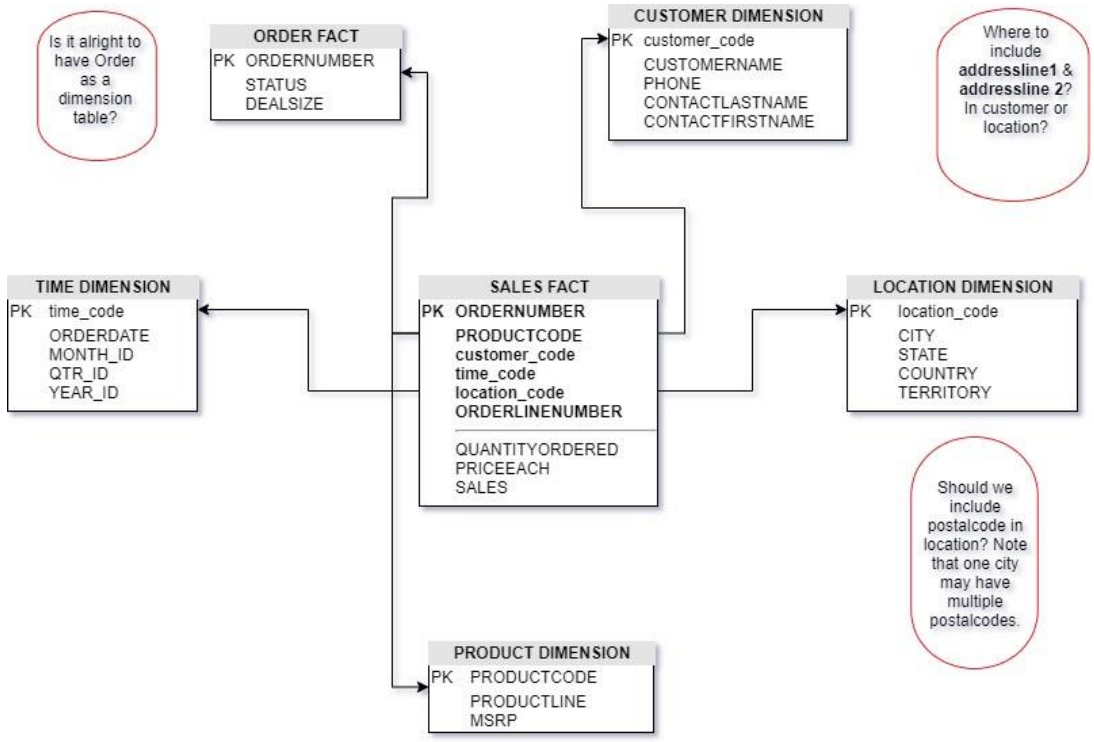

Figure 1 - Initial Star Schema for Sales data

2. Problems with the Initial Schema

The initial schema had a few problems as below:

- Whether to consider the ***order*** as a fact or a separate dimension table
- Where to keep the *address* and *postal code* attributes (I.e., ***customer*** or ***location*** table)?
- Repetitive data in the ***location*** and ***time*** dimension tables

3. Snowflake Schema – A Better Option

We extended the star schema to a snowflake schema considering the reality that normalizing the tables from previous interpretation can offer a reduction in the disk space and fewer risks of data corruption. The snowflake schema query is more complex as the dimension tables are normalized, and therefore, the query executes more slowly compared to a star schema query. However, the difference will be a fraction of milliseconds for the given dataset, as it has only 2823 records. On the other hand, it seems more logically organized than the star schema. Moreover, many OLAP database tools are specifically designed to work with snowflake schemas. [10] The following figure shows the snowflake schema where the ***location*** and ***time*** dimension tables are further normalized into sub-dimension tables:

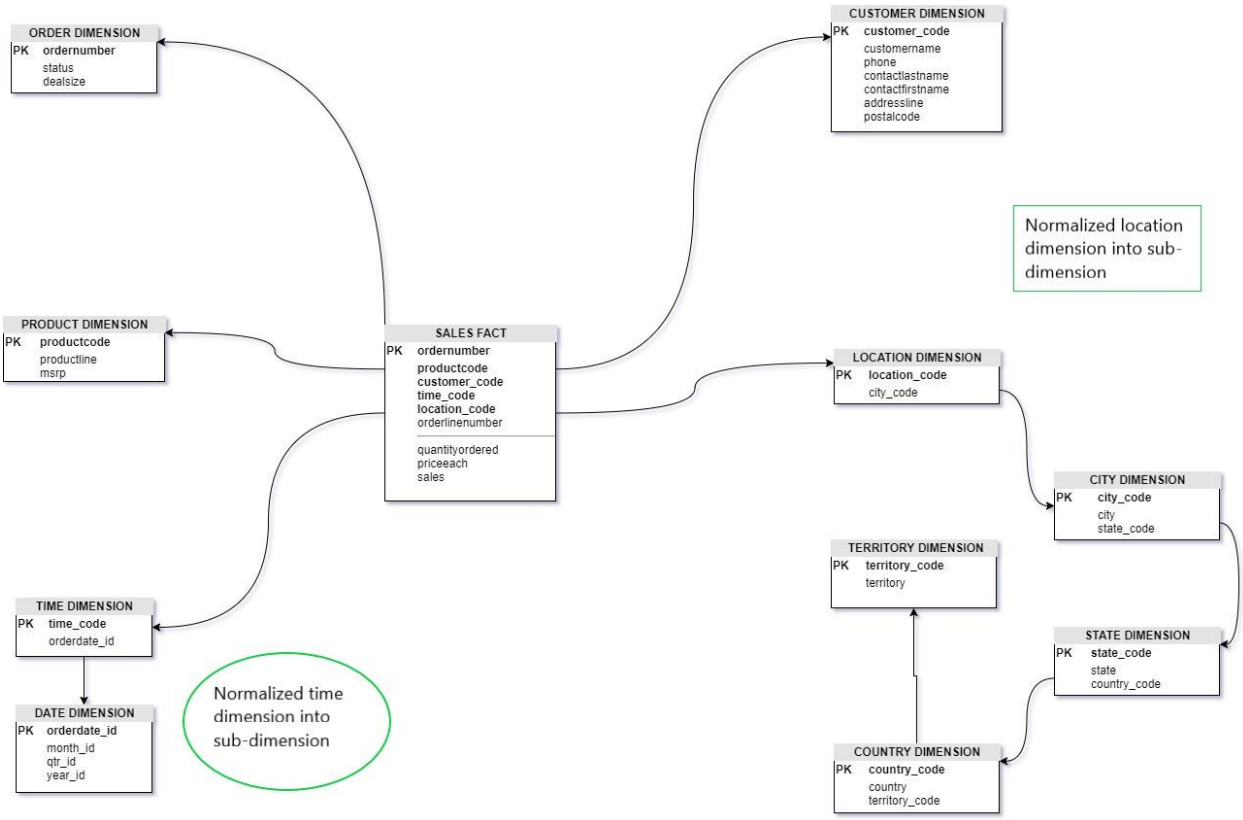

Figure 2 - Snowflake Schema for Sales data

4. **Addressed Problems in New Schema**
   The schema in figure 2 also addressed the problems from point number 2. ***Order*** is a separate dimension table as its attributes (*status* and *dealsize*) are descriptors and not an aggregated value (I.e., measurements or metrics). The ***customer*** dimension includes the *address*, and *postal code* attributes as they are related to a customer. Also, the data redundancy is removed from the ***location*** and ***time*** dimension tables.
5. **Problems with Seconds iteration**
   Although the *addressline* and *postal code* are related to a customer, one needs to use the ***customer*** dimension to retrieve sales details based on the postal code. Hence, it is better to remove those attributes from the ***customer*** dimension. Additionally, the *orderlinenumber* in the fact table is not a measurement for sales, and therefore, it is moved to the ***order*** dimension. Lastly, ***order*** dimension contains the mapping between ordernumber and its related attributes (I.e., *orderlinenumber* and *dealsize*); hence, we have added an attribute *order_code* to identify a record uniquely. Below is the final snowflake schema for sales dataset [9]:

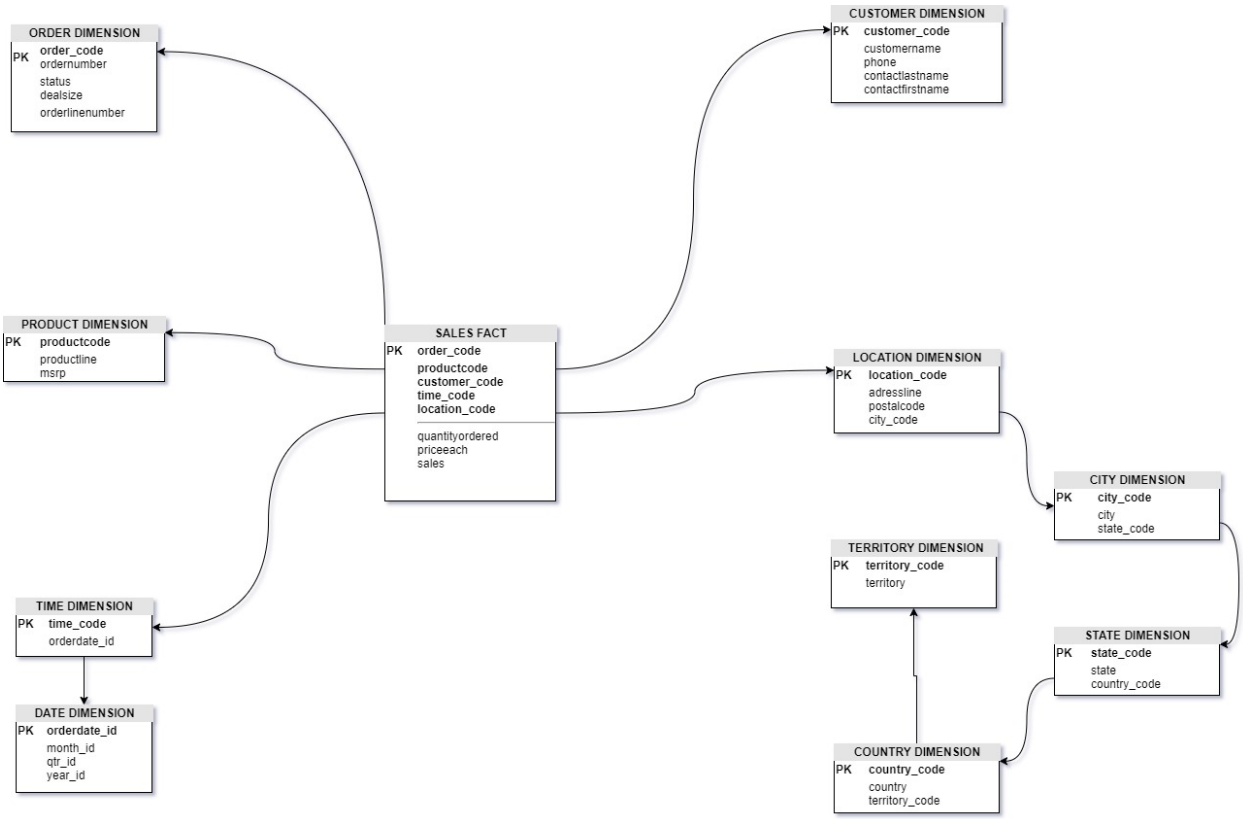

Figure 3 - Final Snowflake Schema for Sales data

4. ANALYTICS

4.1. COGNOS BI

We used the IBM Cognos Business Intelligence tool[11] for generating reports and for analysis of sales data. We created a database using AWS Amazon RDS[12] and created a schema using MySQL workbench for the sales data that was extracted after the cleaning process. Further, we have established a connection of the Amazon RDS from Cognos BI using the connection endpoint and retrieved the tables and their data.

Using the relationships established in AWS, the following figure 4 represents the snowflake schema for sales data that is generated in Cognos BI.

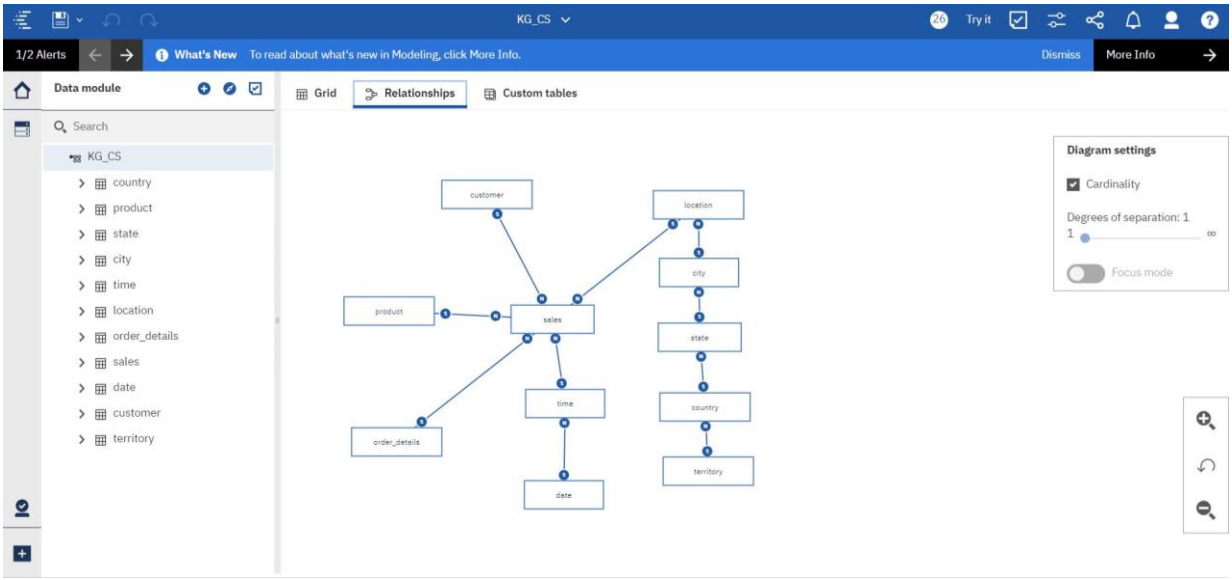

Figure 4 - Snowflake Schema in Cognos BI

4.2. REPORTS

We have generated multiple reports in Cognos BI tool. The following are the multiple scenarios that we have generated reports.

4.2.1. HIGHEST SELLING PRODUCT

We visualized the data among the multiple products and analyzed the record with the sales and the quantity ordered as shown in the below figure 5.

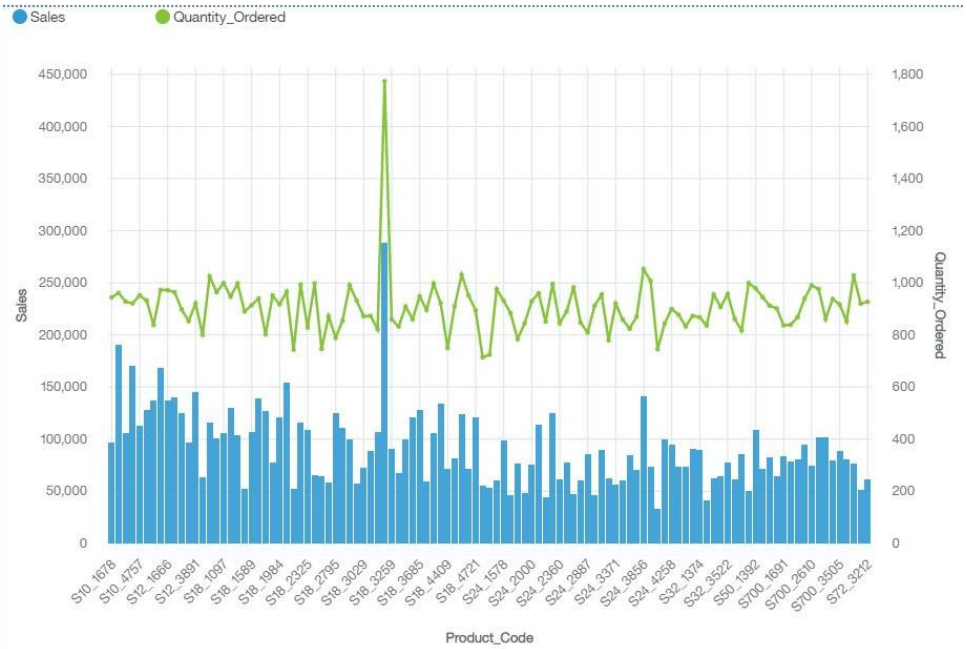

Figure 5 - Highest Selling Product

4.2.2. SALES BY CITY

We visualized the data among the cities and analyzed the record with the sales and the quantity ordered as shown in the below figure 6.

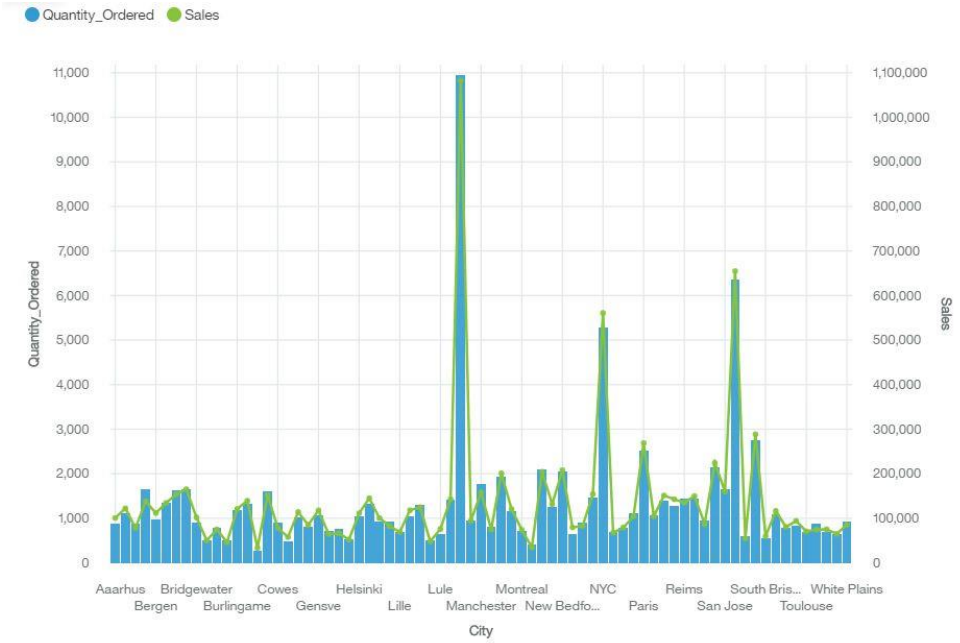

Figure 6 - Sales By City

4.2.3. SALES BY COUNTRY

We visualized the data among the cities and analyzed the record with the sales as shown in the below figure 7.

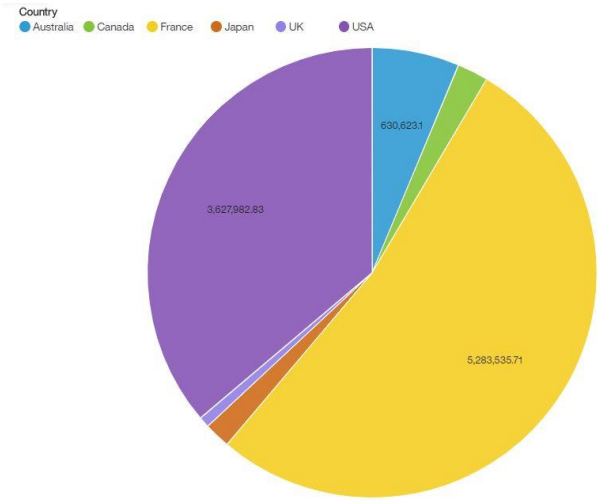

Figure 7 - Sales By Country

4.2.4. SALES BY TERRITORY

We visualized the data among the territories and analyzed the record with the sales as shown in the below figure 8.

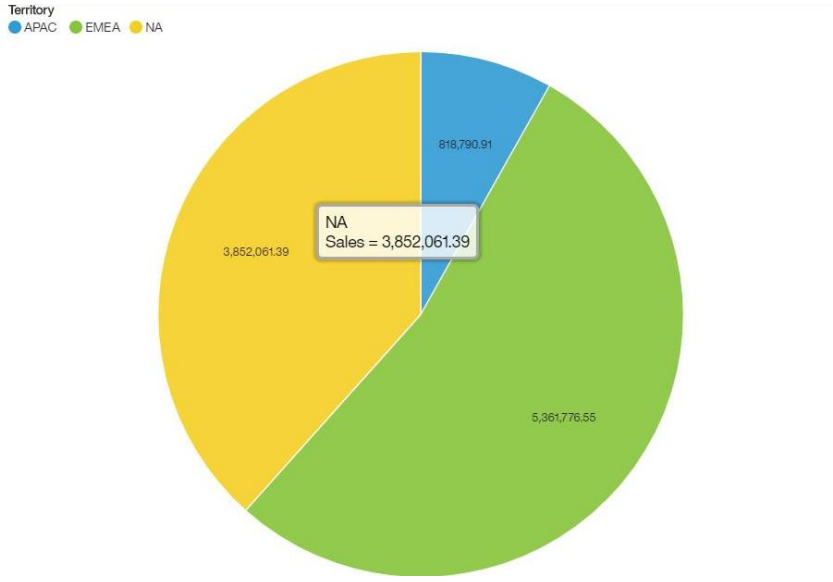

Figure 8 - Sales By Territory

4.2.4. SALES BY YEAR

We visualized the data among the multiple years and analyzed the record with the sales as shown in the below figure 9.

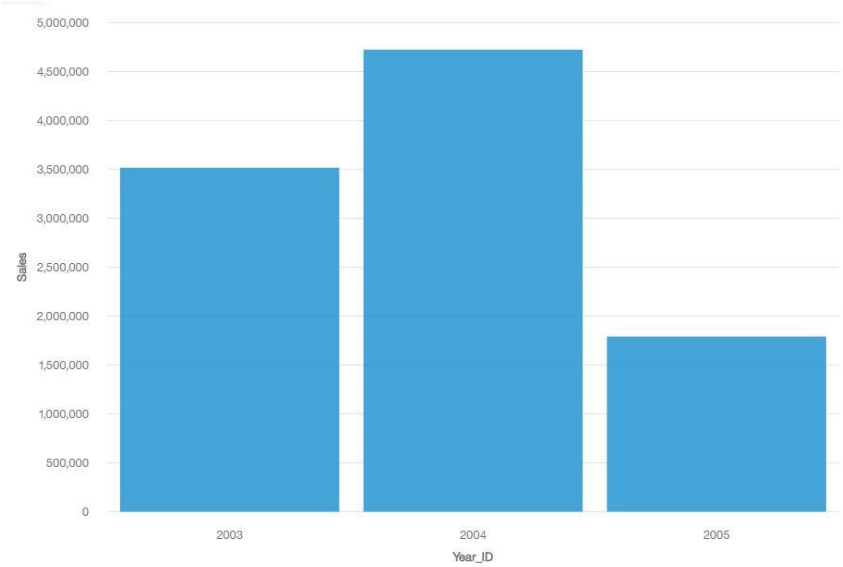

Figure 9 - Sales By Year

REFERENCES

[1] R. Vladimiro, "What is the importance of data?", *Quora*, 2020. [Online]. Available: https://www.quora.com/What-is-the-importance-of-data. [Accessed: 01- Apr- 2020].

[2] "Why Is Data Important for Your Business? | Grow.com", *Grow.com*, 2020. [Online]. Available: https://www.grow.com/blog/data-important-business. [Accessed: 01- Apr- 2020].

[3] "What Is a Sales Analysis Report?", *Smallbusiness.chron.com*, 2020. [Online]. Available: https://smallbusiness.chron.com/sales-analysis-report-58460.html. [Accessed: 01- Apr- 2020].

[4] "What is Cognos? - Definition from WhatIs.com", *SearchCIO*, 2020. [Online]. Available: https://searchcio.techtarget.com/definition/Cognos. [Accessed: 01- Apr- 2020].

[5] researchgate. 2020. *Analysis Of Data Extraction And Data Cleaning In Web Usage Mining*. [online] Available: https://www.researchgate.net/publication/275954899_Analysis_of_Data_Extraction_and_Data_Cleaning_in_Web_Usage_Mining [Accessed 01-Apr- 2020].

[6] D. Meyer and C. Cannon, "Building a better data warehouse," *Amazon*, 1998. [Online]. Available: https://aws.amazon.com/data-warehouse/. [Accessed: 22-Mar-2020].

[7] S. Chaudhuri, Surajit Chaudhuri Microsoft Research, Microsoft Research, and Umeshwar Dayal Hewlett-Packard Labs, "An overview of data warehousing and OLAP technology," *ACM SIGMOD Record*, 01-Mar-1997. [Online]. Available: https://dl.acm.org/doi/10.1145/248603.248616. [Accessed: 24-Mar-2020].

[8] M.K. Sandhu, A. Kaur, R. Kaur, "Data warehouse schemas", *Int. J. of Innovative Research in Advance Engineering (IJIIRAE)*, vol. 2, pp. 47-51, 2015.

[9] G. Segura, "Sample Sales Data," *Kaggle*, 24-Nov-2016. [Online]. Available: https://www.kaggle.com/kyanyoga/sample-sales-data. [Accessed: 20-Mar-2020].

[10] M. Smallcombe, "Snowflake Schemas vs. Star Schemas: What Are They and How Are They Different?," *Xplenty*, 10-Mar-2020. [Online]. Available: https://www.xplenty.com/blog/snowflake-schemas-vs-star-schemas-what-are-they-and-how-are-they-different/. [Accessed: 26-Mar-2020].

[11] "Start your analytics journey anywhere," *Business Intelligence | Cognos Analytics, Watson Analytics*. [Online]. Available: https://www.ibm.com/analytics/business-intelligence. [Accessed: 02-Apr-2020].

[12] "RDS: understanding animal research in medicine," *Amazon*, 2007. [Online]. Available: https://aws.amazon.com/rds/. [Accessed: 02-Apr-2020].