DALHOUSIE UNIVERSITY

CSCI 5408: Data Management and Warehousing Analytics

Group 30

Case Study Report

Instructor: Dr. Saurabh Dey

Date of Submission: 3rd April 2020

Submitted By:

Nupur Bhatt (B00842470)

Pallavi Desai(B00837405)

Sneha Jayavardhini Doguparthi (B00846995)

Purpose:

The purpose of this case study was to take the sales dataset given to us and present it in a form that is understandable by the end user, to make important business-oriented decisions.

The following steps were performed in the case study:

1. Gathering and transformation of a sample sales data
2. Building a data warehouse
3. Creating reports using Cognos Analytics BI

Gathering and transformation of a sample sales data

Summary of the data:

For this case study, we downloaded a sample sales dataset from Kaggle.[7] It contains 25 columns that are attributes describing the details of product sales and 2823 sales records.

We concluded the following about the granularity of the sales dataset: Every record in the database describes the sale of a product specific to a particular order.

Although, not much information was provided about the metadata of the dataset, based on observations by filtering and sorting the data, we deduced the following about the attributes describing the sales:

1. **ORDERNUMBER** – order number that is different for every order in the database. Contains no missing values.
2. **QUANTITYORDERED** – the quantity of products ordered inside a particular order. Contains no missing values.
3. **PRICEEACH** – describes the price of a product inside an order. It is different for every order. Contains no missing values.
4. **ORDERLINENUMBER** – sequential number for each product in a particular order.
5. **SALES** – sale price for each product in a particular order. It is the result of QUANTITYORDERED*PRICEEACH. It is different for every order.
6. **ORDERDATE** – It is the date of order purchase.
7. **STATUS** – Status of the order. Can be ‘Cancelled’, ‘In Process’, ‘Shipped’, ‘Disputed’, ‘On Hold’, ‘Resolved’
8. **QTR_ID** – quarter in which order was placed. Derived from ORDERDATE.
9. **MONTH_ID** – month in which order was placed. Derived from ORDERDATE.
10. **YEAR_ID** – year in which order was placed. Derived from ORDERDATE.
11. **PRODUCTLINE** – The category of product in an order. Contain 7 unique values.
12. **MSRP** – manufacturer suggested retail price. Different for every product.
13. **PRODUCTCODE** – unique code for every product. Contain 109 unique values.
14. **CUSTOMERNAME** – name of the customer who placed an order. Contains 92 unique values.

15. **PHONE** – contact detail of the customer. Contains 91 unique values.
16. **ADDRESSLINE1** – street address. Contains 92 unique values.
17. **ADDRESSLINE2** – apartment number, suite or space number. Contains 2521 missing values and 9 unique values.
18. **CITY** – city of the customer's headquarters. Contains 73 unique values.
19. **STATE** – state of the customer's headquarters. Contains 1486 missing values and 16 unique values.
20. **POSTAL CODE** – postal code of the customer's headquarters. Contains 76 missing values and 73 unique values.
21. **COUNTRY** – country of the customer's headquarters. Contains no missing values and 19 unique values.
22. **TERRITORY** – territory the customer's headquarters belong to. Contains 4 unique values.
23. **CONTACTLASTNAME** – last name of the sales representative for a customer. Contains 77 unique values.
24. **CONTACTFIRSTNAME** – first name of the sales representative for a customer. Contains 72 unique values.
25. **DEALSIZE** – based on SALES for a product inside an order. Based on estimation, 'Small' deal size indicates sales between the smallest sale in the database and 3000. 'Medium' deal size indicates sales between 3000 and 7000. 'Large' deal size indicates sales between 7000 and largest sale in the database.

Cleaning and transformation of the data:

For performing cleaning and transformation, we used Python's pandas library to store the table in a data frame so that cleaning would be easier and straightforward that way.
We started by

1. Removed inconsistencies
    a. Merged columns AddressLine1 and AddressLine2, renamed it to ADDRESS
    b. Removed leading and trailing whitespaces in Address
2. Removed unnecessary data
    a. Extracted month, year and day from the order date timestamp
3. Handled NULL values
    a. Replacing them with 'NA'

Building a Data Warehouse

Data warehouse

Data warehouse (DW) is a core component of business intelligence used for data analysis and reporting also known as enterprise data warehouse (EDW). The data from various resources are integrated to DW's central repositories and generates analytical reports using current data and historical data. The reports generated will be used for crucial decision making in an organization. The data may pass through an operational data store and requires additional

cleaning before it is used for reporting to ensure data quality before it is used for reporting in DW [1].

Data Warehouse vs Database [2]

Data warehouse and databases are both relational data systems. DW is used for storing large amount of historical data as well as current data and allows complex queries processing using Online Analytical Processing (OLAP). Databases store current transaction and allow fast processing and accessing of business transactions known as Online Transaction Processing (OLTP).

Optimization

A database is optimized to maximize the speed by which data access and analysis took place. Also, optimization is necessary for the efficiency with which data is updated using OLTP. Databases performing transactions using OLTP has a detailed data from a current source and hence performing analytical queries on table joins is difficult and adds complexity.

Data warehouse uses OLAP to perform complex queries on large aggregated datasets. As a part of business intelligence, OLAP allows the user to understand the corporate trends and identify potential issues.

Data Structure

Databases uses normalized data which takes minimum disk space but maximizes response time. Data Warehouse uses denormalized data which has fewer tables and data redundancies. Denormalization offers better performance when used for analytical purposes.

Analysis

Databases are normally used for transactional process but, analytical queries can be performed on databases. Due to the normalized structure of databases, it becomes difficult to perform analytical queries on databases. It requires experts who can perform complex analytical queries on databases.

Data Warehouse has a denormalized structure which eases the task of analysis and reporting. Data analytics can be performed dynamically in data warehouse since it takes the data which changes over the time.

Concurrent users

Many users can simultaneously access databases without affecting its performance. Data warehouses allows only limited users to access the data compared to operational systems.

Online Analytical Processing (OLAP)

OLAP allows the user to analyse data from different databases at the same time. OLAP can generate pre-aggregated and pre-calculated data which is necessary for analysis and making performance faster [3].

Basic analytical operations of OLAP:

Roll-up

Roll-up process allows aggregation of data either by reducing dimensions or by climbing up concept hierarchy.

For e.g. If region dimension has data for New York city and Los Angeles city which can be roll-up to USA country. Also, sales corresponding to NYC is 400 and LA is 600 which will be aggregated to 1000 for USA.

Drill-down

Drill-down process allows the fragmentation of data into smaller parts either by increasing the dimensions or by moving down the concept hierarchy. It is opposite to the roll-up process.

For e.g. Quarter dimension can be drilled down to months dimension.

Slice

In this process, one dimension is selected and sliced to form a new sub-cube.

For e.g. If OLAP cube has 3 dimensions, City, Quarters, Products then using Quarters dimension a new cube can be formed for city and products dimension for a specific quarter.

Dice

It is like a slice operation with the only difference that sub cube can be formed using 2 or more dimensions.

Pivot

In this operation, the data axes are rotated to give a substitute presentation of data.

Dimensional Modelling:

Choosing Facts and Dimensions:

A fact table contains measurements or metrics about the sales and is de-normalized.[9] In the sample sales dataset, the facts describe the sale of a particular product in a particular order. A dimension table is independent and connected to the fact table via a foreign key and can stand on its own. [9] Meaning, in this case study, a dimension, be it product, time, customer or region, would exist without customers issuing any orders. Figure 1 displays the star schema we chose for this case study.

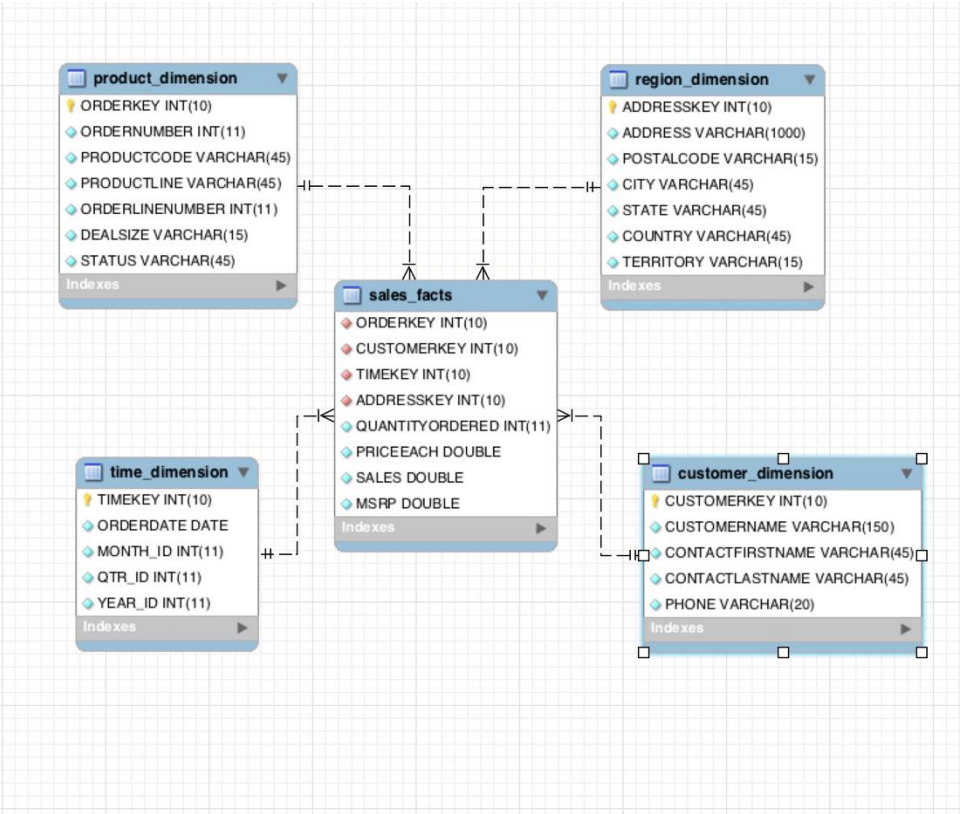

Figure 1: Star schema with fact and dimension tables

We chose the following attributes from the sample sales dataset to form the fact table:

Facts: QUANTITYORDERED, PRICEEACH, SALES, MSRP

ORDERKEY, CUSTOMERKEY, TIMEKEY and ADDRESSKEY are foreign keys in the facts table.

We chose the following dimensions to the facts:

1. Product Dimension
    a. Attributes: ORDERKEY, ORDERNUMBER, PRODUCTCODE, PRODUCTLINE, ORDERLINENUMBER, DEALSIZE, STATUS
2. Time Dimension
    a. Attributes: TIMEKEY, ORDERDATE, MONTH_ID, QTR_ID, YEAR_ID

3. Customer Dimension
    a. Attributes: CUSTOMERKEY, CUSTOMERNAME, CONTACTFIRSTNAME, CONTACTLASTNAME, PHONE
4. Region Dimension
    a. Attributes: ADDRESSKEY, ADDRESS, POSTALCODE, CITY, STATE, COUNTRY, TERRITORY

ORDERKEY, TIMEKEY, CUSTOMERKEY and ADDRESSKEY are surrogate keys in their respective dimension tables.

Choosing Data Warehouse Schema:

We chose the star schema for this case study. We weighed the pros and cons of both a snowflake schema and a star schema. With the snowflake schema, we would require less storage space as data would not be redundant and hence, data consistency would be maintained. But with the star schema, we would get faster query performance, a simple database design and less complex queries, since lesser number of join operations would be performed.[8] The pros of choosing a star schema for this particular case study outweighed the pros of choosing a snowflake schema, so we chose the former.

Creating reports using Cognos Analytics BI

Business Intelligence

BI (Business Intelligence) is a set of processes, architectures, and technologies that convert raw data into meaningful information that drives profitable business actions. It is a suite of software and services to transform data into actionable intelligence and knowledge.[4] BI has a direct impact on organization's strategic, tactical and operational business decisions. It supports fact-based decision making using historical data rather than assumptions and gut feeling.

IBM Cognos Analytics

BI tools like Cognos Analytics perform data analysis and create reports, summaries, dashboards, maps, graphs, and charts to provide users with detailed intelligence about the nature of the business. As reporting is a foundational part of business intelligence which focuses on visualizing data in different types of visualizations such as tables, graphs, and charts. Visualizations within the context of reporting are a graphical representation of data, the goal of which is to accurately present information in a form that is digestible to end users.

We have used IBM Cognos Analytics for creating reports and dashboards. Cognos Analytics integrates reporting, modelling, analysis, dashboards, stories, and event management so that we can understand the organization data and make effective business decisions.[5]

For our case study we have used the IBM Cognos Analytics trial version, we have created an IBM account and explored the guided demo to get a brief overview of Cognos Analytics. [6]

Following are the steps performed to develop a dashboard in Cognos Analytics:

1. In order to upload the files, after creating the star schema, the .csv files are uploaded to ‘My content’ using the upload files options. ‘Operation completed’ status bar is shown on the top to confirm a successful upload. In addition to .csv files, we can also upload .xlsx files.
2. Data modules are the source objects that contain data. A data module must be created for the development of graphs. After clicking on ‘New data module’, the files from My content are uploaded.
3. In order to relate the .csv files and use the star schema to perform the reporting task the files must be joined.

4. Since we have a star schema, all the tables are connected to the central fact table. The relationship between dimensions and facts is one to many relationships. Now, the join is made by selecting the primary key from dimension table and foreign key from facts table. 'Match selected columns' is clicked in order to match and view the data which is related.
5. Now the joins between all the facts and dimensions are completed as show in the Figure 2.
6. In order to create a dashboard, New-> Dashboard->Select the template
7. We must select a source from 'My Content' or 'Team Content', the new data module which is created is selected.
8. Following are some of the reports which are created using the above steps:

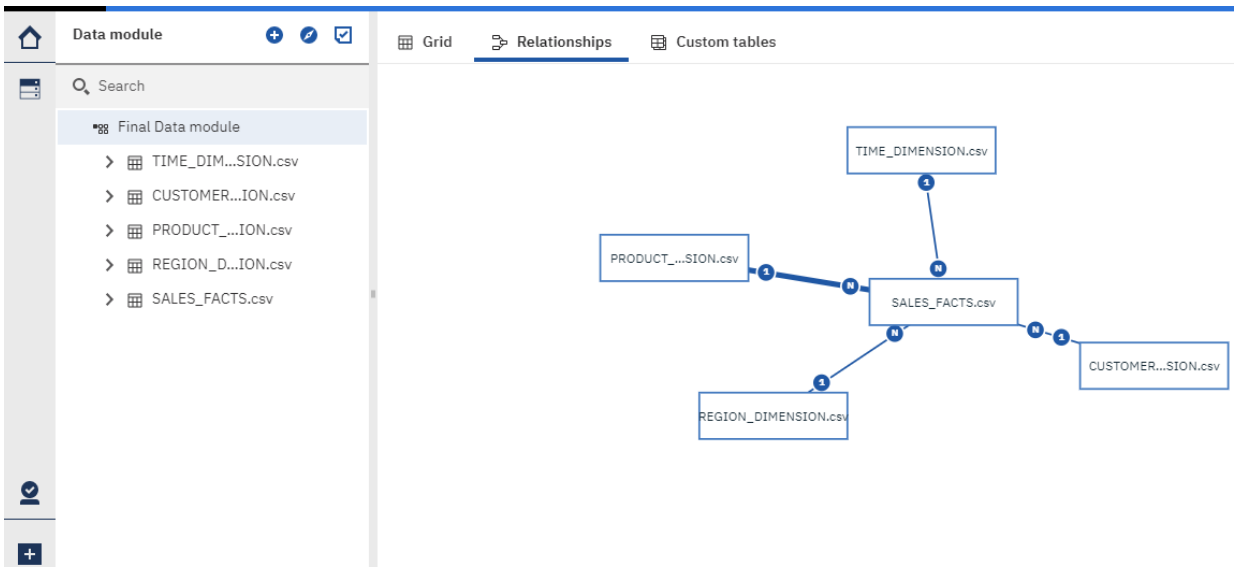

Figure 2: Data module in Cognos Analytics

Figure 3 Sales for each product line

Figure 3 shows the sales for each product line in different months. The user can select a year and see the sales for each product line in all the months. The deal size can also be changed and the change in the line chart can be observed. The pi-chart majorly shows the product line with highest sales and lowest sales.

Figure 4 Sales of top 5 products

Figure 4 shows the sales of top 5 products in year 2004, the packed bubble chart show that S18_3232 product has the highest sales, and the tabular graph shows the top 5 products along with their product lines. User can change the year with help of Data player on the top.

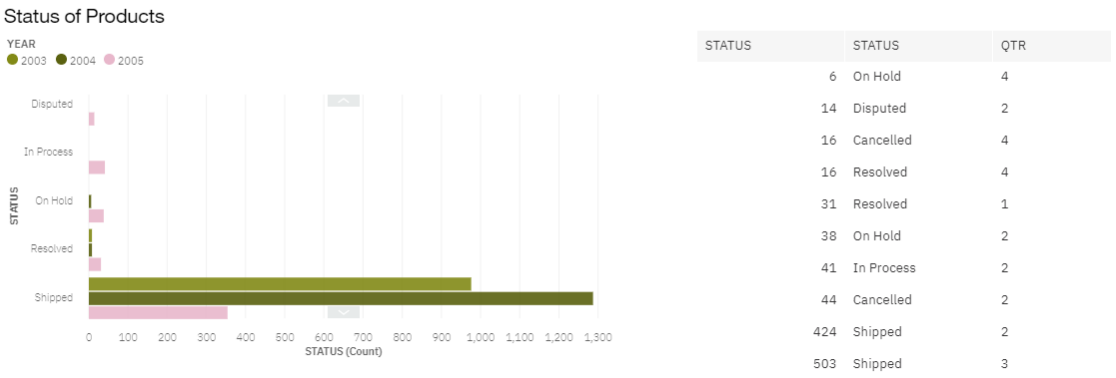

Figure 5 Status of Products

It is important for a user to know the status of products. Figure 5 show the status of products in different quarters of 3 years. In 2004, quarter 2, 503 products were shipped which was highest.

Figure 6 gives the visualization of sales data by region. The first graph word cloud shows the total sale happened in each country and USA has the maximum sale. The pie chart shows the total sale by product line. It gives the visualization of country specific sale of products for a particular year. Also, this report helps to visualize the total sale in every country for a specific product line.

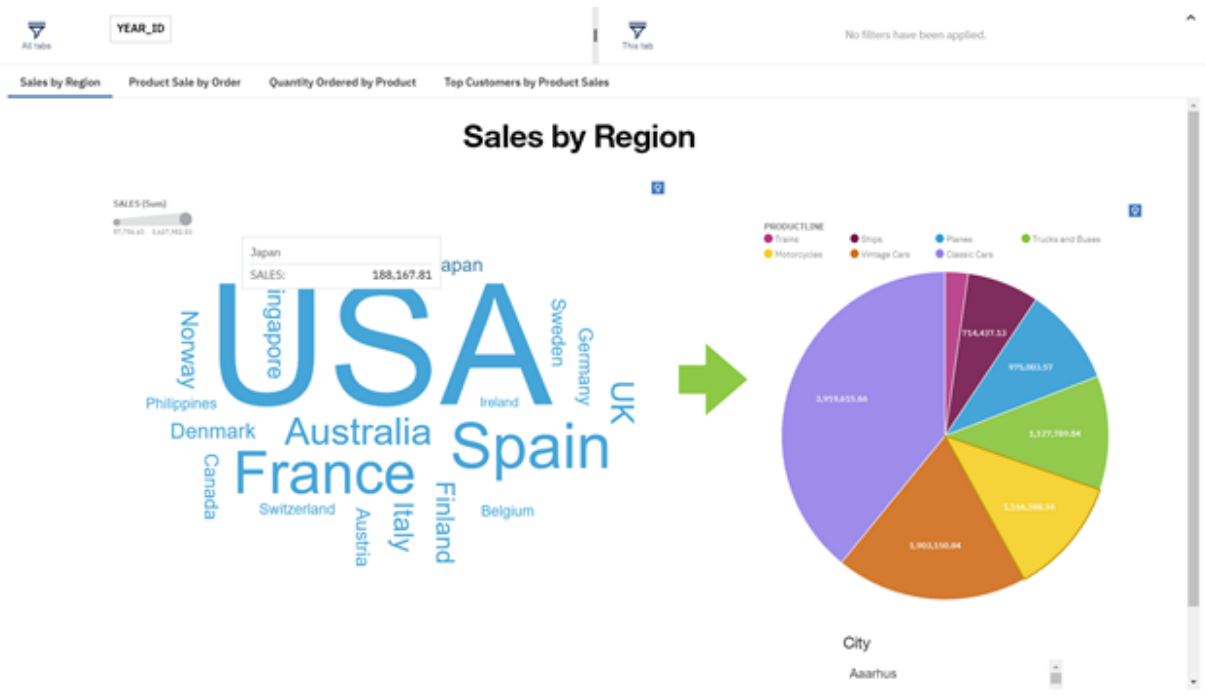

Figure 6 Product sales by region

It is necessary to understand the product sale by each customer. Hence, figure 7 gives the visualization of top 5 customers having maximum sale by each product line. The bar chart shows the total sale of each product line by deal size and the point graph shows top 5 customers having maximum total sale. Based on the selection of a product line, top 5 customers with maximum sale for that product line will be displayed.

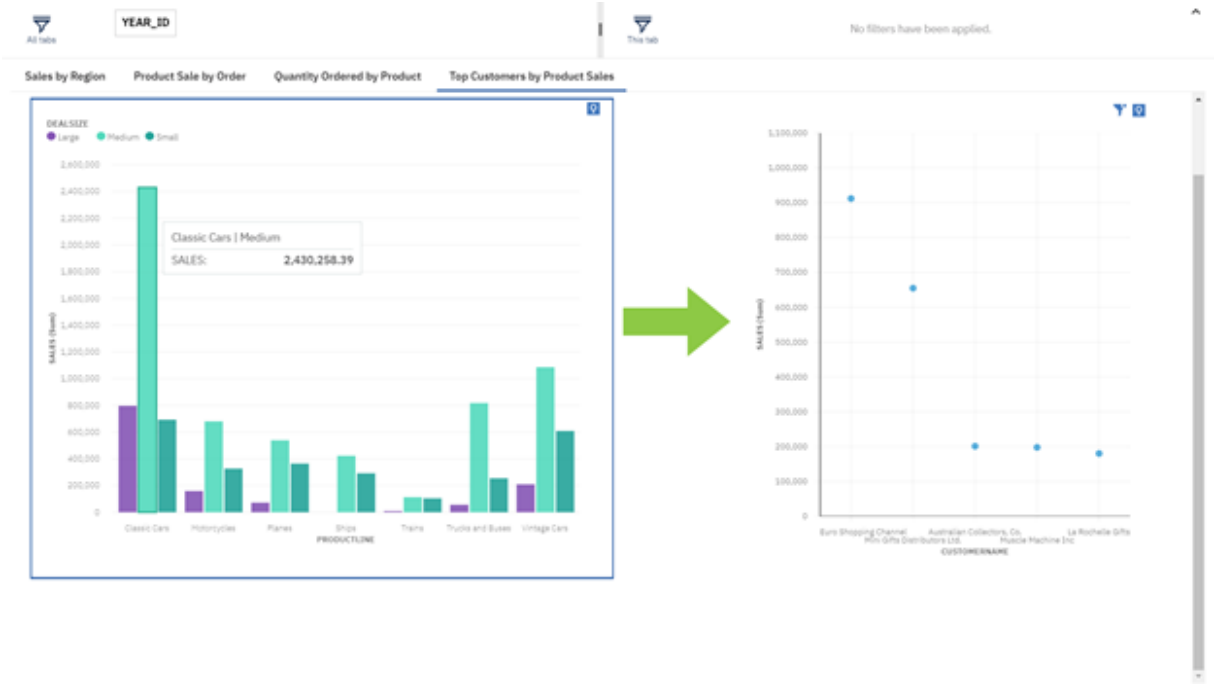

Figure 7 Product sales by customer

References:

[1]"Data warehouse", *En.wikipedia.org*, 2020. [Online]. Available: https://en.wikipedia.org/wiki/Data_warehouse. [Accessed: 02- Apr- 2020].

[2]"The Difference Between a Data Warehouse and a Database", *Panoply*, 2020. [Online]. Available: https://panoply.io/data-warehouse-guide/the-difference-between-a-database-and-a-data-warehouse/. [Accessed: 02- Apr- 2020].

[3]"What is OLAP (Online Analytical Processing): Cube, Operations & Types", *Guru99.com*, 2020. [Online]. Available: https://www.guru99.com/online-analytical-processing.html. [Accessed: 02- Apr- 2020].

[4] "What is Business Intelligence? Definition & Example," *Guru99*. [Online]. Available: https://www.guru99.com/business-intelligence-definition-example.html. [Accessed: 03-Apr-2020].

[5] *IBM Knowledge Center*. [Online]. Available: https://www.ibm.com/support/knowledgecenter/en/SSEP7J_11.0.0/com.ibm.swg.ba.cognos.wig_cr.doc/c_gtstd_ica_overview.html. [Accessed: 03-Apr-2020].

[6] "Business Intelligence Reporting: A Complete Overview," *JReport*. [Online]. Available: https://www.jinfonet.com/resources/bi-defined/bi-reporting/. [Accessed: 03-Apr-2020].

[7] "Sample Sales Data", *Kaggle.com*, 2020. [Online]. Available: https://www.kaggle.com/kyanyoga/sample-sales-data. [Accessed: 03- Apr- 2020].

[8] "Design Tip #105 Snowflakes, Outriggers, and Bridges - Kimball Group", *Kimball Group*, 2020. [Online]. Available: https://www.kimballgroup.com/2008/09/design-tip-105-snowflakes-outriggers-and-bridges/. [Accessed: 03- Apr- 2020].

[9] "Difference Between Fact Table and Dimension Table", *Guru99.com*, 2020. [Online]. Available: https://www.guru99.com/fact-table-vs-dimension-table.html. [Accessed: 03- Apr- 2020].