Week 7: Dimensional Modeling / DW Design

1. Summary

A set of information collectively is called data. Data is important for any organization. Data warehouse is the repository of data where all of the information is stored. Data warehouse is critical for an organization as it helps in maintaining the single version of truth for the organization. The main aim of data warehouse is to make information readily available to its business users.

The Kimball approach defines data warehouse as a centralized data repository. This approach uses Extract-Transformation-Load (ETL) in order to load to the data warehouse. One of the main characteristics of the Kimball approach begins with identifying the important business processes by way of collecting business requirements. When these are answered the data warehouse can be built. Data keeps on changing as per business requirements or in actual changes in data for this update made to the data warehouse is very flexible with the Kimball approach. The data for data warehouse generates normally from operations or transaction systems. Data first arrives in staging area where the transformations are applied before loading into the Datawarehouse. "The model proposed by Kimball for data warehousing—the dimensional model—is not normalized. The fundamental concept of dimensional modeling is the star schema." (Rangarajan, 2016). In the star schema there are multiple dimensions and foreign keys that add up to the subject area. The fact table has many dimensions as a cartesian product is created using various dimension values and the fact table is queried for efficient and better performance. The advantage of star schema is that its easy to understand by the users and highly used for reporting. Many Business intelligence (BI) tools can be integrated with star schema. Additionally, star schema has a great performance and is effective in database operation. "Kimball's architecture, it is known as the dimensional data warehouse that follows the bottom-up approach. In Kimball's philosophy, it

first starts with mission-critical data marts that serve analytic needs of departments. Then it is integrating these data marts for data consistency through a so-called information bus." ("Kimball vs. Inmon Data," n.d.)

Kimball approach is ideal when the requirements for reporting are specific to team or departments as this approach to data warehouse is dimensional in which data marts are first created in order to provide specific reporting and analytical needs for the business units. These data marts are later integrated to create a data warehouse between all the data marts. Thus, in this approach specific data marts are logically combined to for data warehouse. This makes the Kimball approach less time consuming. This approach is beneficial when there is little time for the data warehouse to be built up and running. It requires less number of resources too. This approach may not work best if the requirements keep changing very often. The organization as a whole needs to decide the long-term goals and objectives to best utilize the Kimball approach. The company should also consider the size of the company, its future growth, cost and time frame.

2. Star schema

Star Schema to organize Sales Information for Marketing Department

This Model is implemented based on STAR Schema for the dimension modeling, which involves FACT and Dimensions.

DIMENSION TABLES

Marketing analysis can be done over different dimensions, There are four dimensions identified to meet the need of Marketing Analysis, hence there are four dimension tables in this model.

CUSTOMER_DIM is the CUSTOMER Dimension Information.

```
-- Table: CUSTOMER_DIM
CREATE TABLE customer_dim
  (
     customerid    VARCHAR(25) NOT NULL,
     lastname      VARCHAR(50) NOT NULL,
     firstname     VARCHAR(50) NOT NULL,
     streetaddress VARCHAR(255) NOT NULL,
     city          VARCHAR(25) NOT NULL,
     state         VARCHAR(25) NOT NULL,
     zipcode       VARCHAR(10) NOT NULL,
     CONSTRAINT customer_dim_pk PRIMARY KEY (customerid)
  );
```

SALESREPRESENTATIVE_DIM is the Sales Representative Dimension. (It has region identified as composite key of the dimension)

```
-- Table: SALESREPRESENTATIVE_DIM
CREATE TABLE salesrepresentative_dim
  (
     salesrepid        INTEGER NOT NULL,
     lastname          VARCHAR(50) NOT NULL,
     firstname         VARCHAR(50) NOT NULL,
     streetaddress     VARCHAR(255) NOT NULL,
     city              VARCHAR(25) NOT NULL,
     state             VARCHAR(25) NOT NULL,
     zipcode           VARCHAR(10) NOT NULL,
     salesrepregion    VARCHAR(50) NOT NULL,
     regiondescription VARCHAR(255) NOT NULL,
     totalcommission   FLOAT NOT NULL,
     commissionrate    FLOAT NOT NULL,
     CONSTRAINT salesrepresentative_dim_pk PRIMARY KEY (salesrepid,
     salesrepregion)
  );
```

PART_DIM is the Product Part Dimension Information.

```
-- Table: PART_DIM
CREATE TABLE part_dim
  (
     partid          VARCHAR(25) NOT NULL,
     partdescription VARCHAR(255) NOT NULL,
     pclass          VARCHAR(255) NOT NULL,
     warehousenumber VARCHAR(25) NOT NULL,
     CONSTRAINT part_dim_pk PRIMARY KEY (partid)
  );
```

DATE_DIM is Date and Time Dimension Information.

```
-- Table: DATE_DIM
CREATE TABLE date_dim
  (
     odate         TIMESTAMP(0) NOT NULL,
     datepart      DATE NOT NULL,
     year_num      INTEGER NOT NULL,
     month_num     INTEGER NOT NULL,
     day_of_month  INTEGER NOT NULL,
     hour_of_day   INTEGER NOT NULL,
     dayofweek     VARCHAR(16) NOT NULL,
     weekofyear    INTEGER NOT NULL,
     qtrofyear     INTEGER NOT NULL,
     CONSTRAINT date_dim_pk PRIMARY KEY (odate)
  );
```

FACT TABLE

Each sale line record is considered FACT and stored in the **SALES_ORDER_FACT** table. Each record includes foreign key from each dimension tables.

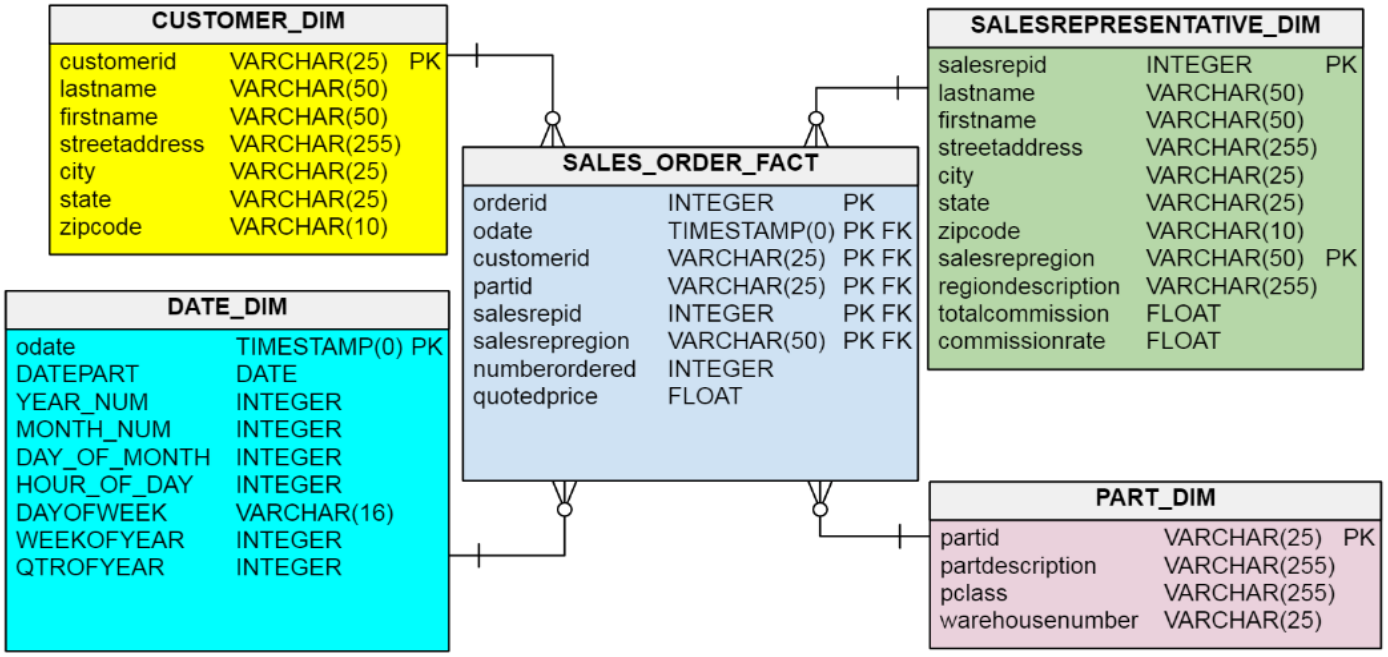
SALES_ORDER_FACT

```
-- Table: SALES_ORDER_FACT
CREATE TABLE sales_order_fact
(
   orderid         INTEGER NOT NULL,
   odate           TIMESTAMP(0) NOT NULL,
   customerid      VARCHAR(25) NOT NULL,
   partid          VARCHAR(25) NOT NULL,
   salesrepid      INTEGER NOT NULL,
   salesrepregion  VARCHAR(50) NOT NULL,
```

```
    numberordered   INTEGER NOT NULL,
    quotedprice     FLOAT NOT NULL,
CONSTRAINT sales_order_fact_pk PRIMARY KEY (orderid, odate, customerid,
partid, salesrepid, salesrepregion),
CONSTRAINT fk_cust_dim FOREIGN KEY (customerid) REFERENCES customer_dim(custo
merid),
CONSTRAINT fk_sr_dim FOREIGN KEY (salesrepid, salesrepregion) REFERENCES
salesrepresentative_dim (salesrepid, salesrepregion),
CONSTRAINT fk_part_dim FOREIGN KEY (partid) REFERENCES part_dim (partid),
CONSTRAINT fk_date_dim FOREIGN KEY (odate) REFERENCES date_dim (odate)
```

ER Diagram

This ER diagram shows STAR schema arrangement for Fact and Dimension tables



References

Rangarajan, S. (2016, September 1). Data Warehouse Design – Inmon versus Kimball. Retrieved from http://tdan.com/data-warehouse-design-inmon-versus-kimball/20300.

Kimball, R., & Ross, M. (2013). The data warehouse toolkit: The definite guide to dimensional modeling. Indianapolis, IN: Wiley & Sons.

Poolet, M. A. (2007). Discover the star schema. SQL Server Magazine, 9(7), 29–31.

Corral, K., Schuff, D., & St. Louis, R. D. (2006). The impact of alternative diagrams on the accuracy of recall: A comparison of star-schema diagrams and entity-relationship diagrams. Decision Support Systems, 42(1), 450–468.

(n.d.) Kimball vs. Inmon Data Warehouse Architectures. Retrieved from https://www.zentut.com/data-warehouse/kimball-and-inmon-data-warehouse-architectures/