Designing the Star Schema in Data Warehousing

Star schema is the fundamental schema among the data mart schema and it is simplest. This schema is widely used to develop or build a data warehouse and dimensional data marts. It includes one or more fact tables indexing any number of dimensional tables. The star schema is a necessary cause of the snowflake schema. It is also efficient for handling basic queries.
It is said to be a star as its physical model resembles the star shape having a fact table at its center and the dimension tables at its peripheral representing the star's points. In this article, we shall solve one important design problem in Data Warehousing.

Problem statement

Consider an order management operational database that tracks order numbers, dates, the requested ship dates, customers and their shipping and billing addresses, products and their quantity and gross dollar amount, sales representatives that take and process orders, the deals (promotions) and discounts proposed/offered to customers.

You have to design a data warehouse that will be updated from the above operational database and should support decision making by helping to answer analytical questions about the net order dollar amounts per customer, products, promotions or deals, and the performance of their sales representatives or agents. Analysis of requested ship dates is important for analysis as well. It is also important to allow for performing order amount analysis in various currencies: dollars, dirhams, euros.

Draw the star schema(s) showing the main attributes, including primary keys, foreign keys, and facts.

Step 1: Identify the Business process to model in order to identify the fact table. We are talking about Sales here. Fact table will be named as 'Sales'. Facts or measures are:

- Net_amount_per_customer
- Net_amount_per_product
- Net_amount_per_promotion

Step 2: Choose the dimensions for the fact table.
Dimensions are

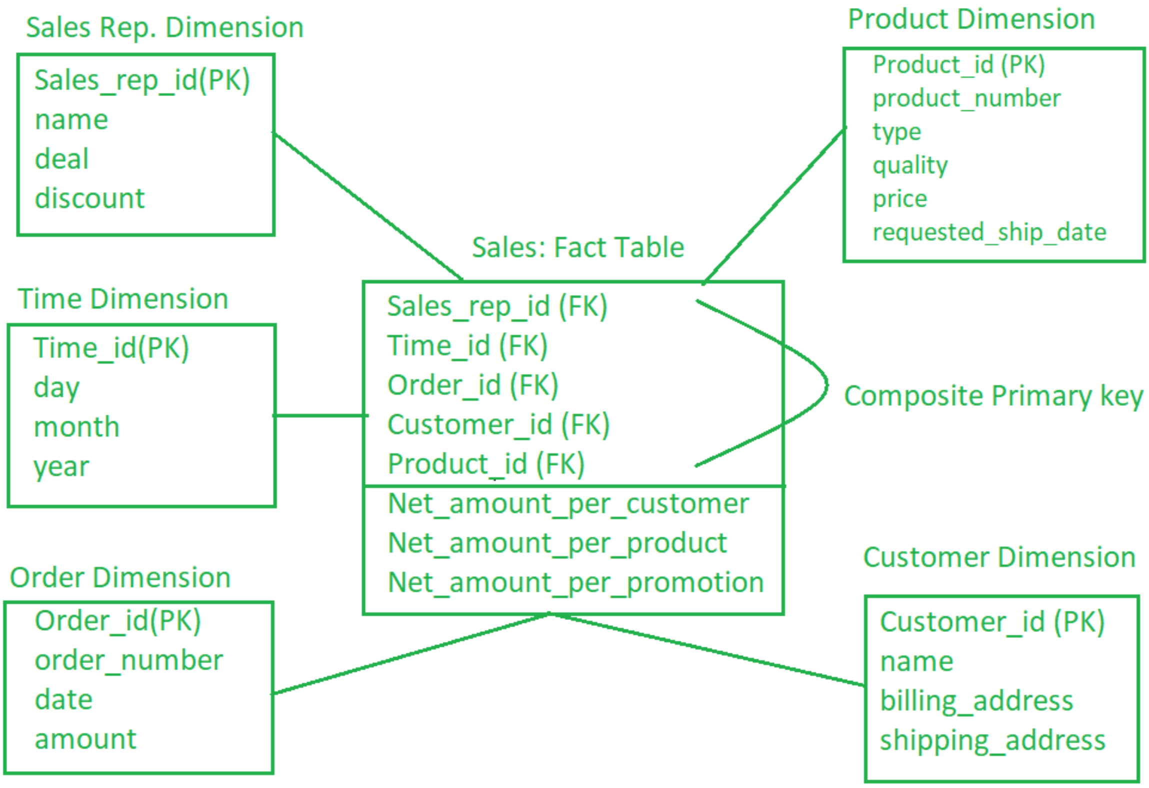
- Sales Representative
- Time
- Customer
- Product
- Order

Step 3: Choose the attributes of dimension tables.

Attributes of SalesRepresentative Dimension:

- Sales_rep_id (primary and surrogate key)
- Name
- Deal
- Discount
- **Attributes of Time Dimension:**
    - Time_id (primary and surrogate key)
    - day
    - month
    - year
- **Attributes of Customer Dimension:**
    - Customer_id (primary and surrogate key)
    - name
    - billing_address
    - shipping_address
- **Attributes of Product Dimension:**
    - Product_id (primary key and surrogate key)
    - quality
    - price
    - product_number
    - requested_ship_date
    - type
- **Attributes of Order Dimension:**
    - Order_id (primary key and surrogate key)
    - order_number
    - date
    - amount

Step 4: Draw the star schema.



Star Schema

Now take one example of a surrogate key in our design. and what are the objectives of using this surrogate key?
In our star schema, we have used one surrogate key per dimension table. Surrogate keys used are:

- Sales_rep_id
- Time_id
- Customer_id
- Product_id
- Order_id

The surrogate key is used to uniquely identify the rows in each dimension table. We can't use business keys in the dimension table to uniquely identify the records. Because business keys may change over time or may be reused.

Make necessary assumptions to compute an approximate size in (MB) of your DW over a period of 5 years.

Assume that the initial size of each dimension table is 1 KB.
Since the Fact table is joined with five dimension tables, assume its size = 1 KB = (1^5) KB initially.
Total size initially = 1 KB + 5 KB = 6 KB

Assume that size of dimension tables increases by 2 KB every year.
Size of each DT after 5 years = 10 KB
Size of fact table after 5 years = 10^5 KB = 100 MB

Problem Statement-2:

Consider a franchise of retail stores having the business setup only in India. The analysis requirements of the franchise include getting to know which items are purchased together by each individual consumer. They wish to know the sales figures in terms of sales amount in Rupees as well as quantity of the individual stores and also for the city, state and region in which they are located. They also wish to know how sales differ over different months, quarters and years; how sales figures change with the hour of the day – e.g., how sales of morning hours are different from sales of evening hours, etc.; how buying habits of male consumers are different from that of the female consumers; how buying habits of married consumers are different from that of the unmarried consumers; how buying habits of consumers vary with their native languages (e.g., Kannad, Telugu, Marathi, etc.).

Design a star schema for such a data warehouse clearly identifying the fact table and dimension tables, their primary keys, and foreign keys. Also, mention which columns in the fact table represent dimensions and which ones represent measures or facts.

Step 1: Identify the Business process to model in order to identify the fact table. We are talking about Sales here. Fact table will be named as 'Sales'.Facts or measures are

1. Total_sales_amount, 2. Total_sales_quantity.
Step 2: Choose the dimensions for the fact table.
Dimensions are:

- Location(of stores),
- Date,
- Customer,
- Product,
- Time

Step 3: Choose the attributes of dimension tables.
Attributes of Location dimension:

- Location_id (primary key and surrogate key)
- city
- district
- state
- region (rural or urban)

Attributes of Date dimension:

- Data_id (primary key and surrogate key)
- day
- week
- month
- quarter
- year

Attributes of Customer dimension:

- Customer_id (primary key and surrogate key)
- name
- gender
- marital_status
- language

Attributes of Product dimension:

- Product_id (primary key and surrogate key)
- name
- type
- price

Attributes of Time dimension:

- Time_id (primary key and surrogate key)
- am_pm_indicator

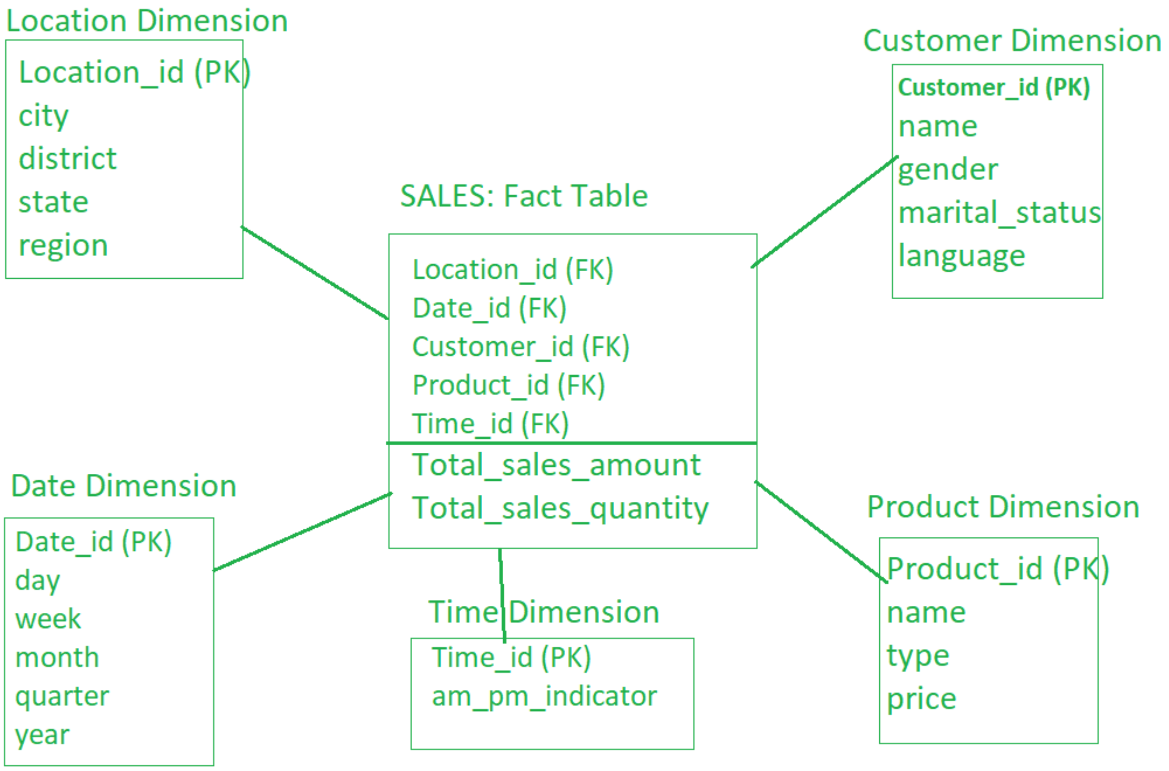
The primary key of the fact table is the composite key, consisting of primary keys of all 5 dimensions.
PK of Sales is {Location_id, Date_id, Customer_id, Product_id, Time_id}

Step 4: Attribute hierarchy in the dimension tables.

```
Location: city -> district -> state

Date: day -> week -> month -> quarter -> year.
```

Step 5: Draw the star schema.



Write one SQL statement that runs on your schema and returns the number of purchases made during the evening hours by the married customers and the unmarried customers in the month of May 2005.

Query:

```
SELECT marital_status, SUM(Total_sales_quantity)

FROM Sales S, Date D, Customer C, Time T

WHERE S.Date_id = D.Date_id AND

S.Customer_id = C.Customer_id AND

S.Time_id = T.TIme_id AND

T.am_pm_indicator = 'PM" AND

D.month = 'May' AND

D.year = 2005

GROUP BY marital_status;
```

It will result in 2 rows, each for married and unmarried customers.