**LECTURE SIX**

**DATA WAREHOUSE SCHEMAS**

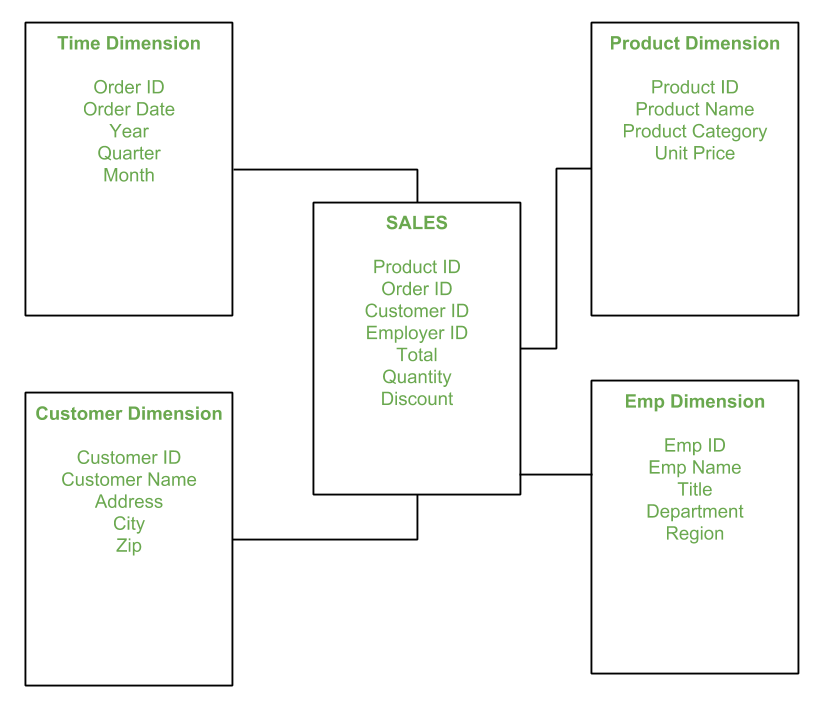
**MULTI DIMENSIONAL DATA MODEL (SCHEMAS EXPLANATION):**

Schema is a logical description of the entire database. It includes the name and description of records of all record types including all associated data-items and aggregates. Much like a database, a data warehouse also requires to maintain a schema. A database uses relational model, while a data warehouse uses Star, Snowflake, and Fact Constellation schema. In this chapter, we will discuss the schemas used in a data warehouse.

**Star Schema**

**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 case of the snowflake schema. It is also efficient for handling basic queries.

It is said to be star as its physical model resembles to the star shape having a fact table at its center and the dimension tables at its peripheral representing the star’s points. Below is an example to demonstrate the Star Schema:



In the above demonstration, SALES is a fact table having attributes i.e. (Product ID, Order ID, Customer ID, Employer ID, Total, Quantity, Discount) which references to the dimension tables. **Employee dimension table** contains the attributes: Emp ID, Emp Name, Title, Department and Region. *Product dimension table* contains the attributes: Product ID, Product Name, Product Category, Unit Price. *Customer dimension table* contains the attributes: Customer ID, Customer Name, Address, City, Zip. *Time dimension table* contains the attributes: Order ID, Order Date, Year, Quarter, Month.

**Model of Star Schema–** In Star Schema, Business process data, that holds the quantitative data about a business is distributed in fact tables, and dimensions which are descriptive characteristics related to fact data. Sales price, sale quantity, distant, speed, weight, and weight measurements are few examples of fact data in star schema. Often, A Star Schema having multiple dimensions is termed as Centipede Schema. It is easy to handle a star schema which have dimensions of few attributes.

**Advantages of Star Schema –**

1. **Simpler Queries:**

Join logic of star schema is quite cinch in compare to other join logic which are needed to fetch data from a transactional schema that is highly normalized.

2. **Simplified Business Reporting Logic:**

In compared to a transactional schema that is highly normalized, the star schema makes simpler common business reporting logic, such as as-of reporting and period- over-period.

3. **Feeding Cubes:**

Star schema is widely used by all OLAP systems to design OLAP cubes efficiently. In fact, major OLAP systems deliver a ROLAP mode of operation which can use a star schema as a source without designing a cube structure.

**Disadvantages of Star Schema –**

1. Data integrity is not enforced well since in a highly de-normalized schema state.

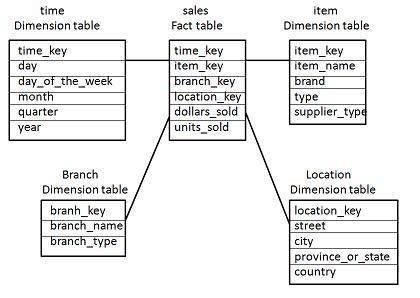
2. Not flexible in terms if analytical needs as a normalized data model.

3. Star schemas don’t reinforce many-to-many relationships within business entities – at least not frequently.

• Each dimension in a star schema is represented with only one-dimension table.

• This dimension table contains the set of attributes.

• The following diagram shows the sales data of a company with respect to the four dimensions, namely time, item, branch, and location.



• There is a fact table at the center. It contains the keys to each of four dimensions.

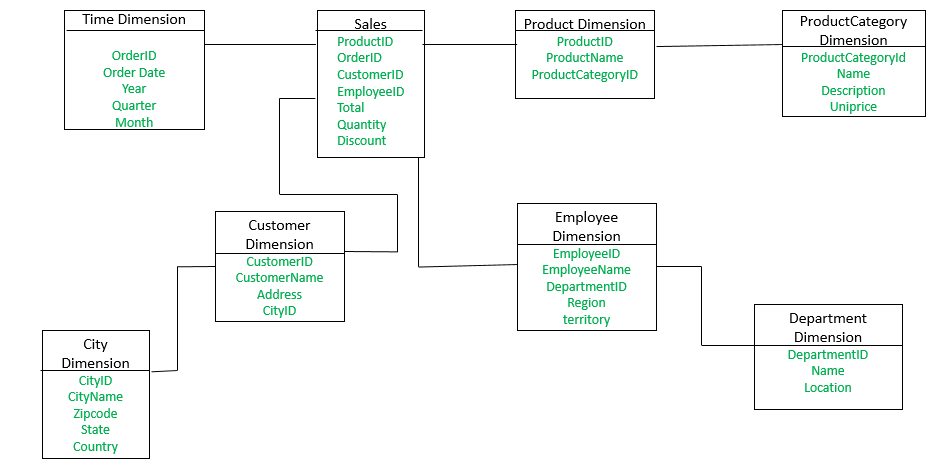
• The fact table also contains the attributes, namely dollars sold and units sold.

**Note** − Each dimension has only one dimension table and each table holds a set of attributes. For example, the location dimension table contains the attribute set {location\_key, street, city, province\_or\_state, country}. This constraint may cause data redundancy. For example, "Vancouver" and "Victoria" both the cities are in the Canadian province of British Columbia. The entries for such cities may cause data redundancy along the attributes province\_or\_state and country.

**Snowflake Schema**

**Introduction:** The snowflake schema is a variant of the star schema. Here, the centralized fact table is connected to multiple dimensions. In the snowflake schema, dimension are present in a normalized from in multiple related tables. The snowflake structure materialized when the dimensions of a star schema are detailed and highly structured, having several levels of relationship, and the child tables have multiple parent table. The snowflake effect affects only the dimension tables and does not affect the fact tables.

**Example:**



The **Employee** dimension table now contains the attributes: EmployeeID, EmployeeName, DepartmentID, Region, Territory. The DepartmentID attribute links with **Employee** table with the **Department** dimension table. The **Department** dimension is used to provide detail about each department, such as Name and Location of the department. The **Customer** dimension table now contains the attributes: CustomerID, CustomerName, Address, CityID. The CityID attributes links the **Customer** dimension table with the **City** dimension table. The **City** dimension table has details about each city such as CityName, Zipcode, State and Country.

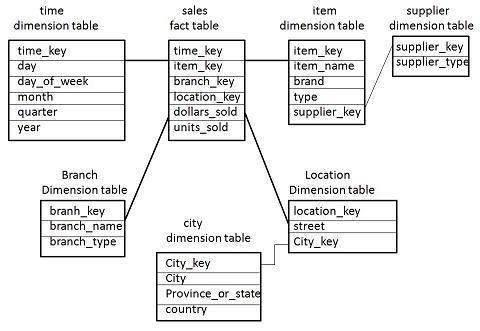
The main difference between star schema and snowflake schema is that the dimension table of the snowflake schema are maintained in normalized form to reduce redundancy. The advantage here is that such table(normalized) are easy to maintain and save storage space. However, it also means that more joins will be needed to execute query. This will adversely impact system performance.

**What is snowflaking?** The snowflake design is the result of further expansion and normalized of the dimension table. In other words, a dimension table is said to be snow flaked if the low-cardinality attribute of the dimensions have been divided into separate normalized tables. These tables are then joined to the original dimension table with referential constrains (foreign key constrain). Generally, snowflaking is not recommended in the dimension table, as it hampers the understandability and performance of the dimension model as more tables would be required to be joined to satisfy the queries.

• Some dimension tables in the Snowflake schema are normalized.

• The normalization splits up the data into additional tables.

• Unlike Star schema, the dimensions table in a snowflake schema are normalized. For example, the item dimension table in star schema is normalized and split into two dimension tables, namely item and supplier table.



• Now the item dimension table contains the attributes item\_key, item\_name, type, brand, and supplier-key.

• The supplier key is linked to the supplier dimension table. The supplier dimension table

contains the attributes supplier\_key and supplier\_type.

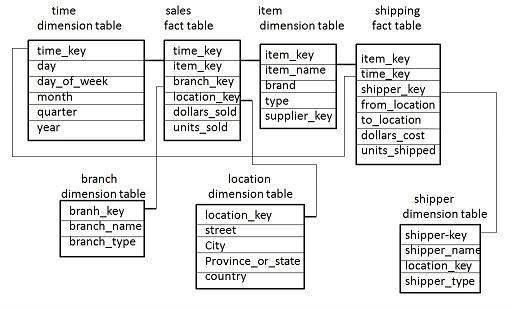
**Note** − Due to normalization in the Snowflake schema, the redundancy is reduced and

therefore, it becomes easy to maintain and the save storage space.

**Fact Constellation Schema**

• A fact constellation has multiple fact tables. It is also known as galaxy schema.

• The following diagram shows two fact tables, namely sales and shipping.



• The sales fact table is same as that in the star schema.

• The shipping fact table has the five dimensions, namely item\_key, time\_key, shipper\_key, from\_location, to\_location.

• The shipping fact table also contains two measures, namely dollars sold and units sold.

• It is also possible to share dimension tables between fact tables. For example, time, item, and location dimension tables are shared between the sales and shipping fact table.

**Schema Definition**

Multidimensional schema is defined using Data Mining Query Language (DMQL). The two primitives, cube definition and dimension definition, can be used for defining the data warehouses and data marts.