Dimensional modelling

3NF Modelling

Types of dimensions in data warehouse

Confirmed dimensions

Degenerate dimensions: a column in a fact table does not have any dimention table linked is Call a degenerate dimension. Eg: - invoice number.

Junk dimensions: - attributes which are not having any value to the fact table is called junk dimensions.

Role-playing dimensions: - A dimension table that has a multiple valid relationships with a fact table is called a role playing dimension

Ex:- Date Dimension can be used to fetch order placed date, order shipment date, order closure date.

Slowly Changing Dimensions

SCD Type 1 – history is **not** maintained and data will be over written

SCD Type 2 – **history is maintained** and data will be maintained by adding **a new row**

SCD Type 3 – **history is maintained** and data will be maintained by adding a **new column**

SCD Type 4 – rapidly changing dimensions example salary attributes changes every year

Type of facts

Additive: Additive facts are facts that can be summed up through all of the dimesions in the fact table. A sales fact is a good example for additive fact.

Semi-additive: semi-additive facts are facts that can be summed up for some of the dimensions in the fact table, but not the others.

Eg: - Daily balances fact can be summed up through the customers dimension but not throguh the time dimension.

Non-additive: Non-Additive facts are facts that cannot be summed up for any of the dimensions present in the fact table.

Eg:- Facts which have percentages, ratios calculated.

Factless fact table: In the real world, it is possible to have a fact table that contains no measures or facts. These tables are called “factless fact tables”

A Fact tables that contain aggregated facts are often called summary tables.

Schema Types: 4 types of schemas are available

Star schema: A star schema is an enhancement

Snoke flake schema

Galaxy schema

Fact constellation schema

Extractions methods in data warehouse

Logical extraction

Full extraction

Incremental extraction

Physical extraction

Online extraction

Offline extraction

**Logical and Physical Design of Data Warehouse**

**Logical design:**  
  
Logical design deals with the logical relationships between objects. Entity-relationship (ER) modeling technique can be used for logical design of data warehouse. ER modeling involves identifying the entities (important objects), attributes (properties about objects) and the relationship among them.  
  
An entity is a chunk of information, which maps to a table in database. An attribute is a part of an entity, which maps to a column in database.  
  
A unique identifier can be used to make sure the data is consistent.  
  
**Physical design:**  
  
Physical design deals with the effective way of storing and retrieving the data. In the physical design, the logical design needs to be converted into a description of the physical database structures.  
  
Physical design involves creation of the database objects like tables, columns, indexes, primary keys, foreign keys, views, sequences etc.

**Types of Dimensions in data warehouse**

A dimension table consists of the attributes about the facts. Dimensions store the textual descriptions of the business. With out the dimensions, we cannot measure the facts. The different types of dimension tables are explained in detail below.  
  
**Conformed Dimension:**  
  
Conformed dimensions mean the exact same thing with every possible fact table to which they are joined.   
  
Eg: The date dimension table connected to the sales facts is identical to the date dimension connected to the inventory facts.  
  
**Junk Dimension:**  
  
A junk dimension is a collection of random transactional codes flags and/or text attributes that are unrelated to any particular dimension. The junk dimension is simply a structure that provides a convenient place to store the junk attributes.  
  
Eg: Assume that we have a gender dimension and marital status dimension. In the fact table we need to maintain two keys referring to these dimensions. Instead of that create a junk dimension which has all the combinations of gender and marital status (cross join gender and marital status table and create a junk table). Now we can maintain only one key in the fact table.  
  
**Degenerated Dimension:**  
  
A degenerate dimension is a dimension which is derived from the fact table and doesn't have its own dimension table.  
  
Eg: A transactional code in a fact table.  
  
**Role-playing dimension:**  
  
Dimensions which are often used for multiple purposes within the same database are called role-playing dimensions. For example, a date dimension can be used for “date of sale", as well as "date of delivery", or "date of hire".

**SCD type 4 - Fast growing dimension**

In normal scenarios the dimension tables tend to grow slowly. That is the reason, they are called slowly changing dimensions. Example: Location attribute of a customer changes very rarely. However the salary band of a customer is likely to change every year. These type of attributes causes the customer dimension table to grow rapidly.   
  
SCD type 4 provides a solution to handle the rapid changes in the dimension tables. The concept lies in creating a junk dimension or a small dimension table with all the possible values of the rapid growing attributes of the dimension.   
  
**Example**: Take a look at the following dimension attributes of customer

C\_Id

Name

Location

Age\_band

Salary\_band

The age band and salary band are going to change frequently. So create a separate small dimension table with only these attributes. The number of possible values of age band will be around 20 and salary band will be around 30. The total number of rows in the new dimension table are 20x30=60.   
  
The new tables are 

Table name : customer

C\_Id

Name

Location

Table name: customer\_mini

M\_id

Age\_band

Salary\_band

Fact table:

Id

C\_Id

M\_Id

----

The dimension key of the new table should be maintained in the fact table. This way we can handle the rapid changes in the dimension table.

**Extraction Methods in Data Warehouse**

The extraction methods in data warehouse depend on the source system, performance and business requirements. There are two types of extractions, Logical and Physical. We will see in detail about the logical and physical designs.  
  
**Logical extraction**  
  
There are two types of logical extraction methods:  
  
**Full Extraction**: Full extraction is used when the data needs to be extracted and loaded for the first time.  In full extraction, the data from the source is extracted completely. This extraction reflects the current data available in the source system.  
  
**Incremental Extraction**: In incremental extraction, the changes in source data need to be tracked since the last successful extraction. Only these changes in data will be extracted and then loaded. These changes can be detected from the source data which have the last changed timestamp. Also a change table can be created in the source system, which keeps track of the changes in the source data.  
  
One more method to get the incremental changes is to extract the complete source data and then do a difference (minus operation) between the current extraction and last extraction. This approach causes a performance issue.  
  
**Physical extraction**  
  
The data can be extracted physically by two methods:  
  
**Online Extraction**:  In online extraction the data is extracted directly from the source system. The extraction process connects to the source system and extracts the source data.  
  
**Offline Extraction**: The data from the source system is dumped outside of the source system into a flat file. This flat file is used to extract the data. The flat file can be created by a routine process daily.

**Data Warehouse Design Approaches**

Data warehouse design is one of the key technique in building the data warehouse. Choosing a right data warehouse design can save the project time and cost. Basically there are two data warehouse design approaches are popular.  
  
**Bottom-Up Design:**  
  
In the bottom-up design approach, the data marts are created first to provide reporting capability. A data mart addresses a single business area such as sales, Finance etc. These data marts are then integrated to build a complete data warehouse.  The integration of data marts is implemented using data warehouse bus architecture. In the bus architecture, a dimension is shared between facts in two or more data marts. These dimensions are called conformed dimensions. These conformed dimensions are integrated from data marts and then data warehouse is built.  
  
**Advantages of bottom-up design are:**

* This model contains consistent data marts and these data marts can be delivered quickly.
* As the data marts are created first, reports can be generated quickly.
* The data warehouse can be extended easily to accommodate new business units. It is just creating new data marts and then integrating with other data marts.

**Disadvantages of bottom-up design are:**

* The positions of the data warehouse and the data marts are reversed in the bottom-up approach design.

**Top-Down Design:**  
  
In the top-down design approach the, data warehouse is built first. The data marts are then created from the data warehouse.  
  
**Advantages of top-down design are:**

* Provides consistent dimensional views of data across data marts, as all data marts are loaded from the data warehouse.
* This approach is robust against business changes. Creating a new data mart from the data warehouse is very easy.

**Disadvantages of top-down design are:**

* This methodology is inflexible to changing departmental needs during implementation phase.
* It represents a very large project and the cost of implementing the project is significant.

**Data Mining**

Data mining is the process of finding patterns from large data sets and analyzing data from different perspectives. It allows business users to analyze data from different angles and summarize the relationships identified. Data mining can be useful in increasing the revenue and cut costs.  
  
**Example**:  
  
In a supermarket, the persons who bought the tooth brush on Sundays also bought tooth paste. This information can be used in increasing the revenue by providing an offer on tooth brush and tooth paste. There by selling more number of products (tooth paste and tooth brush) on Sundays.  
  
**Data mining process:**  
  
Data mining analyzes relationships and patterns in the stored data based on user queries. Data mining involves four tasks.

* Association: Find the relationship between the variables. For example in retail a store, we can determine which products are bought together frequently and this information can be used to market these products.
* Clustering: Identifying the logical relationship in the data items and grouping them. For example in a retail store, a tooth paste, tooth brush can be logically grouped.
* Classifying: Involves in applying a known pattern to the new data.

**Data mining is identifying the Paterns**

**Data profiling is examining the process of quality**

**Data warehouse**

**Subject oriented**

**Integrated**

**Time Varient**

**Non volatile**