What is Data wareHouse

1. It is a database that is designed for querying and analysis rather than for transaction Processing.

2. It separates analysis workload from transaction system.

3. This helps in:

i. Maintaining historical records

ii. Analyzing the data to gain a better understanding of the business and to improve the Business.

4. Data warehouse is a subject-oriented, integrated, time varying, non-volatile collection of data in support of the management's decision-making process.

i. **Subject** Oriented: This is used to analyze particular subject area.

ii. **Integrated**: This shows that integrates data from different sources.

iii. **Time variant**: Historical data is usually maintained in a Data warehouse, i.e. retrieval can be for any period. In transactional system only the most recent/current data is maintained. But in the Data warehouse recent/current and the previous/historical data is maintained.

iv. **Non**-**Volatile**: Once the data is placed in the data warehouse, it cannot be changed, which means we will never be able to change the data.

What is ETL?

1. ETL stands for Extract-Transform-Load.

- Extract is the process of reading data from a source database/ transactional system.
- Transform is the process of converting the extracted data to required from.
- Load is the process of writing the data into the target database/ analytical system.

2. It is a process which defines how data is loaded from the source system to the target system (data warehouse).

Data Ware House Architecture

There are four layers in DWH architecture:

 Data Source Layer

 Data Staging Area

 Data Storage Layer

 Reporting Layer

Source Layer --First layer is the Data Source layer, which refers to various data stores in multiple formats like relational database, Flat Files, Excel files, Xml Files etc.

-- These data stores business data like Sales, Customer, Finance, Product etc.

Staging Layer--After that the next step is Extract, where the required data from data source layer is extracted and put into the data staging area.

-- Data Staging area is intermediate layer between Data Source Layer and Data Storage Layer used for processing data during the ETL process.

-- Basically needs staging area to hold the data and to perform data transformations, before loading the data into warehouse.

--Actual transformation transactional data into analytical data is done in data staging area.

Storage Layer--And finally, we have the Data Storage layer i.e. data warehouse, the place where the successfully cleaned, integrated, transformed and ordered data is stored in a multi-dimensional environment. Now, the data is available for analysis and query purposes.

Reporting Layer-In reporting layer, data in data storage layer is used to create various type of management reports from where user can take business decisions for planning, designing, forecasting etc.

-- Meta data is nothing but the data about data.

--**Meta data repository** is used to store meta data of data which is actually present in data warehouse i.e. Data storage layer

--**Data mart** can be defined as the subset of data warehouse.

A data mart is focused on a single functional area e.g. product, customers, employees, sales etc.

It is a subject-oriented database and is also known as High Performance Query Structures (HPQS).

OLTP (Online Transaction Processing System):

1. OLTP is nothing but a database which actually stores the daily transactions which are created from one and more applications.
2. Data in OLTP is called as the current data.
3. Mostly normalized data is used in OLTP system.

OLAP (Online Analytical Processing System) :

1. OLAP is use to store analytical data
2. It deals with analyzing the data for decision making and planning, designing etc.

3. Data in OLAP is called as the Historical data.

4. Mostly DE normalized data is used in OLAP system.

Normalization

--Normalization is the process of efficiently organizing the data in the database.

-- Normalization is used to minimize the redundancy.

--Normalization divides the larger table into the smaller table and links them using relationship.

Data Models

Data model tells how the logical structure of a database is modeled/designed.

Data models define how data is connected to each other and how it will be processed and stored inside the system.

---Types of Data Models:

i. Conceptual Data Model

ii. Logical Data Model

iii. Physical Data Model

Conceptual Data Model

-- A conceptual data model is high level design of database.
--Features of conceptual data model include:
1. Displays the important entities and the relationships among them.
2. No attribute is specified.
3. No primary key is specified.

Logical Data Model

-- Logical Data Model defines the data as much as possible, to show how they can be physically implemented in the database.
i. Includes all entities and relationships among them.
ii. All attributes/columns for each entity/table are specified.
iii. The primary key for each entity is specified.
iv. Foreign keys (keys identifying the relationship between different entities) are specified.
v. Constraints are defined. (Unique, Not null, Check, default etc..)

Physical Data model

Actual implementation of logical model into Database is called Physical Data Model

Dimensional Model

Q. What is Fact (Measures) ?

--- It is counted or measured event.

Q. What is Dimension?

---It contains referential information about fact.

Q. What is Fact Table ?

-- Fact table consist of measurements or facts of a business process.

--It is central table in dimension model surrounded by dimension tables.

-- A fact table typically has two types of columns:
i. Those that contain facts.
ii. Those that are a foreign key to dimension tables.

Q. What is Dimension Table?

--Dimension tables are used to describe dimensions.

******Type of Dimension model**

1) Star Schema
2) Snowflake Schema
3) Galaxy or fact Constellation schema

1) Star Schema

1) It is simplest form of dimensional model
2) In Star schema design, central table is called fact table and
Radially connect other tables are called as dimension tables.
3) It is known as star schema because the entity-relationship
Diagram of this schemas look like a star.
4) Dimension tables in star schema are in De-Normalized form.
5) Star Schema is good for data marts with simple relationships.

2) Snowflake Schema

1) The process of normalizing dimension tables is called snow flaking.
2) In Snowflake schema, Dimension Tables are in Normalized form.
3) Snowflake schema is a extension of star schema.
4) It's ER diagram look like a snowflake shape that's why is called as
snowflake schema.

3) Galaxy Schema

1) Galaxy Schema contains two and more fact tables that share
Same dimension tables between them.
2) It is also called Fact Constellation Schema.
3) The schema is viewed as a collection of stars hence the name
Galaxy Schema.

****** Type Of Facts**

1. Additive Fact
2. Non-Additive Fact

3. Semi-Additive Fact

Addictive Fact

-- Additive facts are facts that can be summed up through all of the dimensions in the fact table.

Non-Addictive Fact

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

Semi-Addictive Fact

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

Types of Dimensions

1. Slowly Changing Dimensions

2. Conformed Dimensions

3. Degenerated Dimensions

4. Junk Dimensions

Slowly Changing Dimensions

Slowly Changing Dimensions :

--Dimensions that changes slowly over a period of time, rather than changing on

regular schedule.

-- A Slowly Changing Dimension (SCD) is a dimension that stores and manages both current and historical data over time in a data warehouse

-- It is considered and implemented as one of the most critical ETL tasks in tracking

the history of dimension records.

There are many approaches how to deal with SCD. The most popular are:

Type 0 - The passive method

Type 1 - Overwriting the old value

Type 2 - Creating a new additional record

Type 3 - Adding a new column

Type 4 - Using historical table

Type 6 - Combine approaches of types 1,2,3 (1+2+3=6)

Type 0 - The passive method.

In type 0, no special action is performed upon dimensional changes.

Dimension data that remains same as it was first time inserted.

Type 1 - Overwriting the old value.

In type 1, old value is simply overwritten by new value.

Only new value is maintained.

History of dimension changes is not kept in the database.

This type is easy to maintain and is often use for data which changes are caused

by processing corrections. (e.g. removal special characters, correcting spelling Errors).

Type 2 - New row is created for new data.

Old value and new value is present is same table.

New row is created for new value in dimension table.

In this method all history of dimension changes is kept in the database.

Type 3 - Adding a new column.

In type 3, old and new value is kept in same table and same row.

The new value is loaded into 'new' column and the old one into 'previous'

Column.

History is limited to the number of columns which are created for storing

Historical data.

This is the least commonly needed technique.

Type 4 -Using historical table.

In Type 4, separate table are there for old value and new value

Separate historical table is used to track all historical changes for each of the

dimension.

The 'main' table keeps only the New data (current data) .

e.g. customer and customer history tables.

Type 6 - Combine approaches of types 1,2,3 (1+2+3=6).

In this type we have additional columns in dimension table such as

Current_Address, Current_Year : for keeping current value of the attribute.

Previous_Address, Previous_Year : for keeping historical value of the attribute.

Current_Flag : for keeping information about the most recent record.