Unit 9: Data Warehousing **LH 7**

9.1 Operational Data sources

9.2 ETL (Extract, Transform, Load)

9.3 Data Warehouse Processes, Managers and their functions

9.4 Data Warehouses and Data Warehouses Design

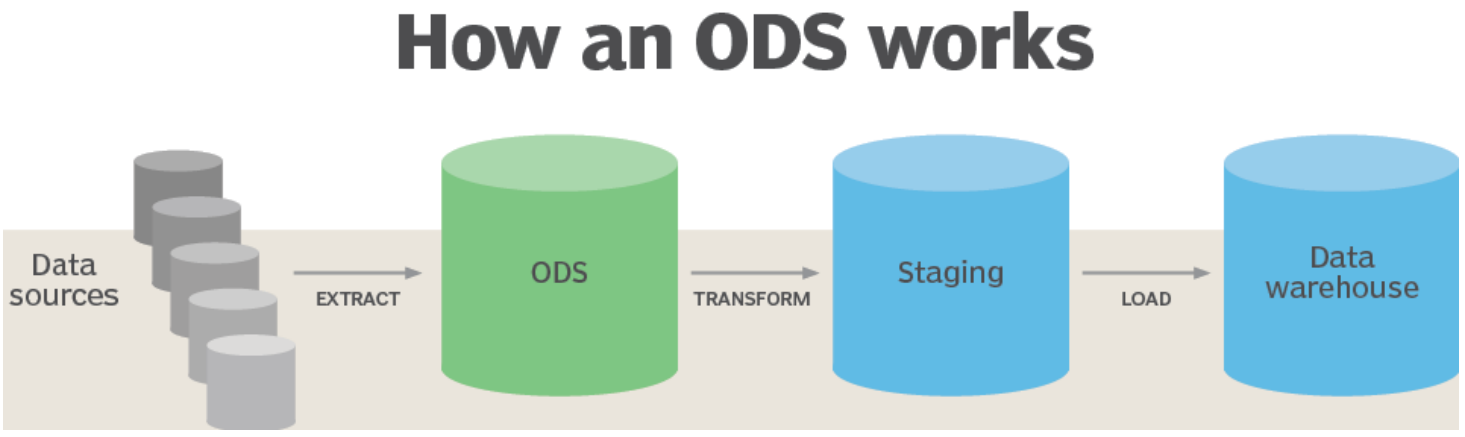
9.5 Guidelines for Data Warehouse Implementation

Operational Data Store

- An operational data store (ODS) is a type of database that integrate data from multiple sources for lightweight data processing activities such as operational reporting and real-time analysis.

- It can be used for integrate data from multiple sources so that business operations, analysis and reporting can be carried out while business operations are occurring.

- This is where most of the data used in current operations is housed before it's transferred to the data warehouse for longer-term storage or archiving.



Source data coming into the data warehouses may be grouped into four broad categories:

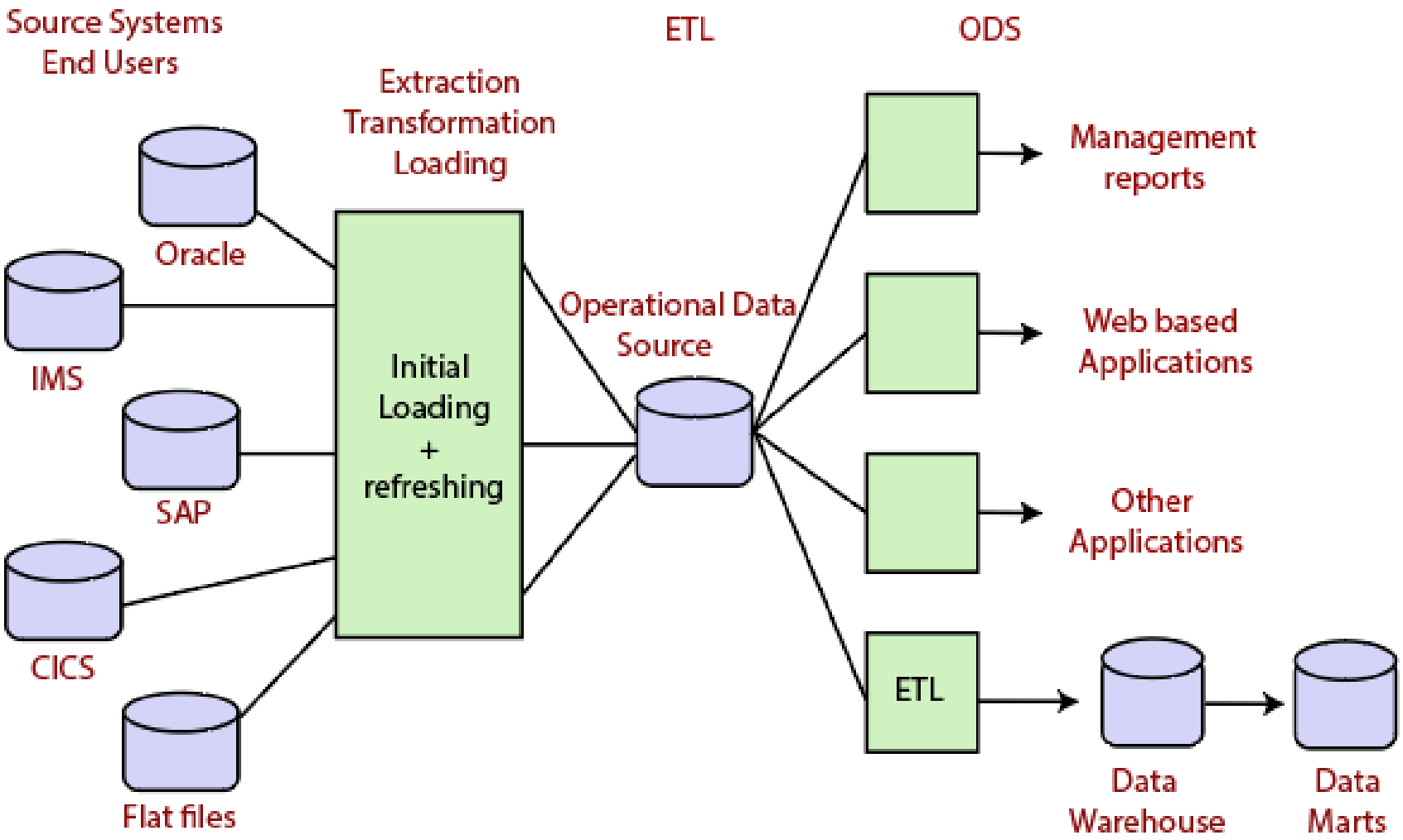
Production Data: This type of data comes from the different operating systems of the enterprise. Based on the data requirements in the data warehouse, we choose segments of the data from the various operational modes.

- **Internal Data:** In each organization, the client keeps their "private" spreadsheets, reports, customer profiles, and sometimes even department databases. This is the internal data, part of which could be useful in a data warehouse.

- **Archived Data:** Operational systems are mainly intended to run the current business. In every operational system, we periodically take the old data and store it in achieved files.

- **External Data:** Most executives depend on information from external sources for a large percentage of the information they use. They use statistics associating to their industry produced by the external department

Operational Data Store:



Operational Data Store Structure

- Operational Data Store acts as a source for the data warehouse and data mart.
- There are some data Marts which are completely depend on data warehouse and some depend on ODS (Operational Data Store).

ODS Properties

- Integrated
  - Data Are Available from different sources
- Persisted (To an Extent)
  - It persist data in certain extend only
- Business Rules (ETL)
  - Data in ODS is derive from ETL process. i.e data are not but it are align to certain business rule.
- Primary Current Valued (Active Data)
  - ODS will have only recent information. For example Data warehouse JMC college will contain information of all current, enrolled student, faculty while ODS will contain only current information.
- Partial Scope:
  - Since it store only current information it will not have full inforamtion
- Get Updated Daily

Data WareHouse

- DWH contains historical data (Time Slicing – You can query in past and retrieve details).
- DWH contains Enterprise Wide Data.
- It is Read Only.
- All information are store from supermarket

Data Mart:

- All Department (Sales, Inverntory, Marketing) are store in different form.

Data Lake (Load First and Think):

- Huge amount of data collected from different sources.
- Data Lake will not have any define format

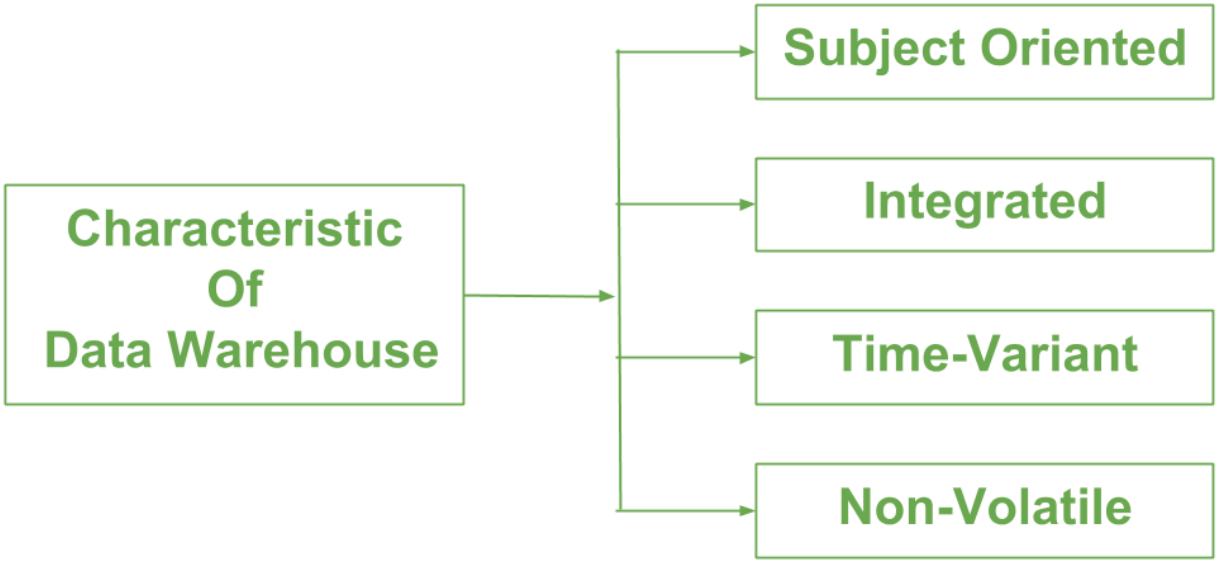
	Data Lake	Data Warehouse	Data Mart
Usages	Predictive and Advance Analytics	Multi Purpose Enable and Performance Analytics	Specific Reporting and Analytics
Duration	Weeks – Month	Days	Hourly- Minutes
Implementation Cost	High	Medium	Low
Data Size	Huge	Medium	Small

Data Warehouse

- A data warehouse is a subject-oriented, integrated, time-variant, and non-volatile collection of data to support management's decision-making, according to William H. Inmon.

- Data Warehouse is a large collection of business data used to help an organization.

- It is not used for daily operations and transaction processing but used for decision



Subject-oriented:

Focus is on Subject Areas rather than Applications

Organized around major subjects, such as customer, product, sales.

Provide a simple and concise view around particular subject issues by excluding data that are not useful in the decision support process.

Provide information on specific topic rather than information about an organization ongoing operations.

Integrated (Filter data and Make common foramt)

- When data resides in many separate applications in the operational environment, the encoding of data is often inconsistent. For instance, in one application, gender might be coded as “m” and “f” in another by 0 and 1. When data are moved from the operational environment into the data warehouse, they assume a consistent coding convention e.g. gender data is transformed to “m” and “f”.

- Integration tasks handles naming conventions as well as physical attributes of data.

Time-variant: (Consider Past Data)

The time horizon for the data warehouse is significantly longer than that of operational systems.

- Operational database: current value data. (60 to 90 days)

- The data warehouse contains a place for storing data that are five to 10 years old, or older, to be used for comparisons, trends, and forecasting. These data are not updated.

Non-volatile:

- Data is not updated or changed in any way once they enter the data warehouse, but are only loaded and accessed.

Data Warehouse Features

- It is separate from the Operational Database.
- Integrates data from heterogeneous systems.
- Stores HUGE amount of data, more historical than current data.
- Does not require data to be highly accurate.
- Queries are generally complex.
- The goal is to execute statistical queries and provide results that can influence decision- making in favor of the Enterprise.
- These systems are thus called Online Analytical Processing Systems (OLAP).

Architecture of a Data Warehouse System

A typical data warehouse system has three main phases:

- **Data acquisition**
  - Relevant data collection
  - Recovering: transformation into the data warehouse model from existing models
  - Loading: cleaning and loading in the DW
- **Storage**
- **Data extraction**
  - Tool examples: Query report, SQL, multidimensional analysis (OLAP tools), data mining

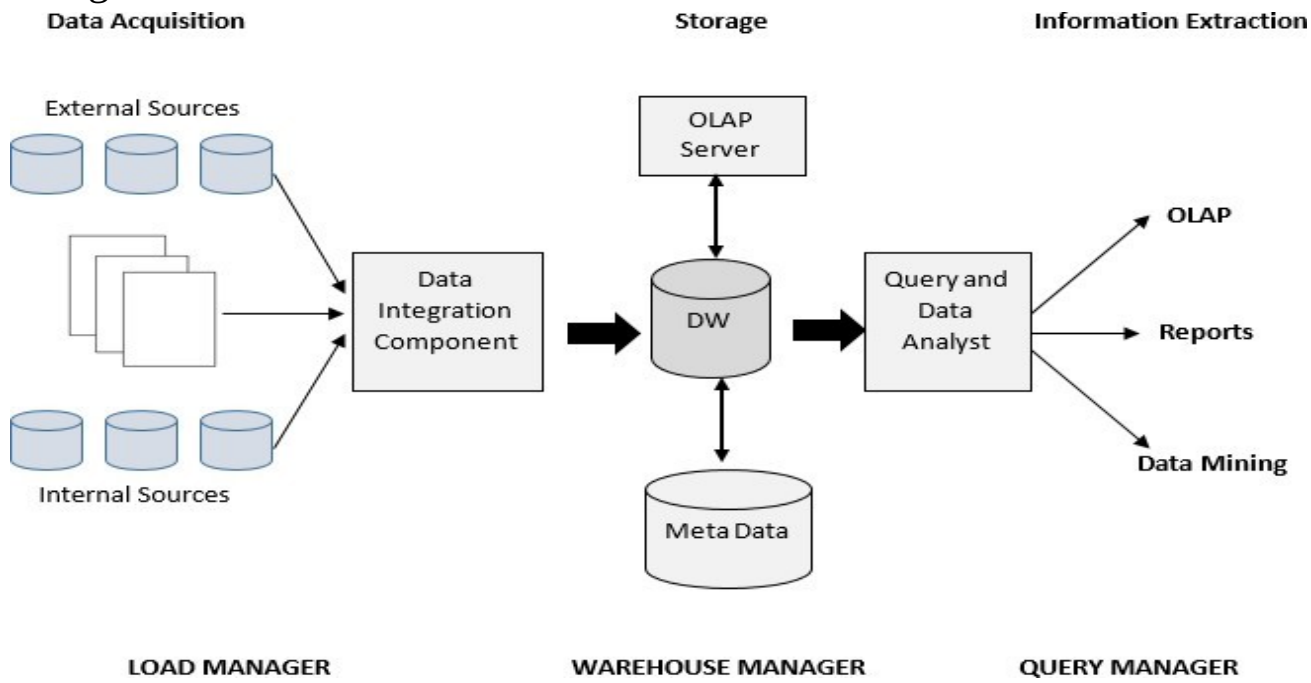
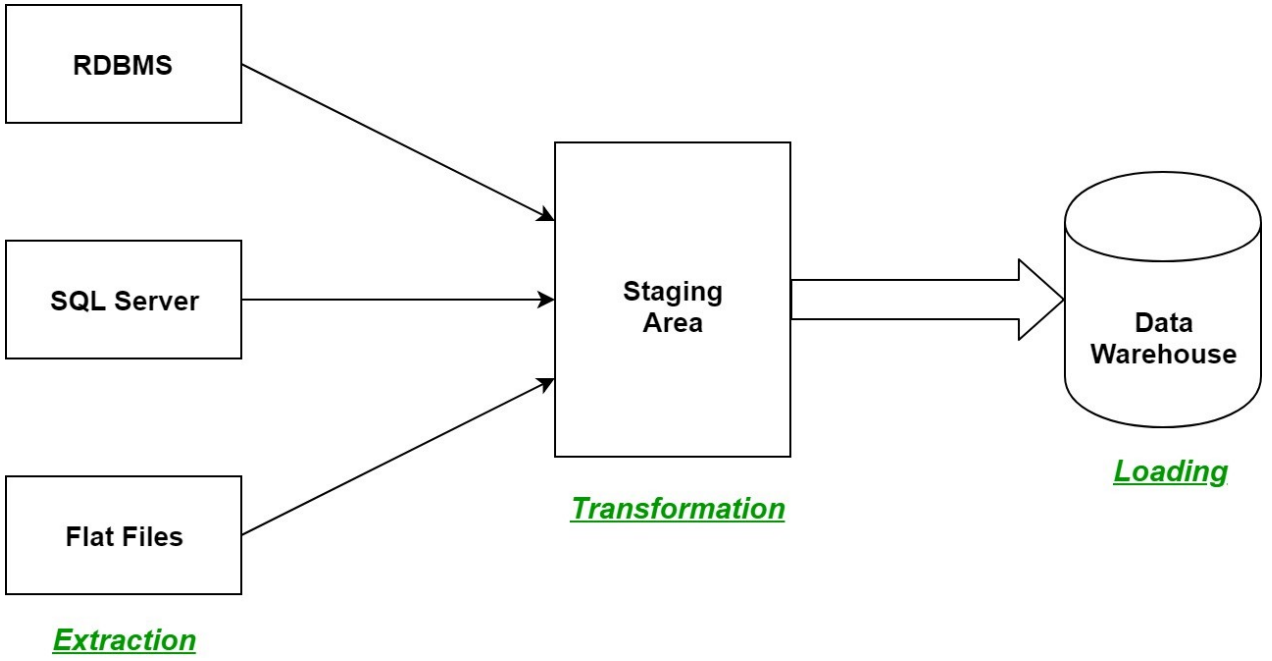


Figure 3. Data Warehouse Architecture

- ETL(Extract, Transform and Load)

ETL Process

- ETL stands for Extract, Transform, Load and it is a process used in data warehousing to extract data from various sources, transform it into a format suitable for loading into a data warehouse, and then load it into the warehouse.
- The process of ETL can be broken down into the following three stages:
    1. Data Extraction
    2. Data Transformation
    3. Data Loading



i) Data Extraction

- The first step of the ETL process is extraction.
- In this step, data from various source systems is extracted which can bein various formats like relational databases, No SQL, XML, and flat filesinto the staging area.
- It is important to extract the data from various source systems and store it into the staging area first and not directly into the data warehouse because the extracted data is in various formats and can be corrupted also.
- Hence loading it directly into the data warehouse may damage it and
- rollback will be much more difficult.
- Therefore, this is one of the most important steps of ETL process.

ii) Data Transformation

- The second step of the ETL process is transformation.
- In this step, a set of rules or functions are applied on the extracted data to convert it into a single standard format.

It may involve following processes/tasks:

1. Filtering – loading only certain attributes into the data warehouse.
2. Cleaning – filling up the NULL values with some default values, mapping U.S.A, United States, and America into USA, etc.
3. Joining – joining multiple attributes into one.
4. Splitting – splitting a single attribute into multiple attributes.
5. Sorting – sorting tuples on the basis of some attribute (generally key-attribute).

iii) Data Loading

- The third and final step of the ETL process is loading.
- In this step, the transformed data is finally loaded into the data warehouse.
- Sometimes the data is updated by loading into the data warehouse very frequently and sometimes it is done after longer but regular intervals.
- The rate and period of loading solely depends on the requirements and varies from system to system.

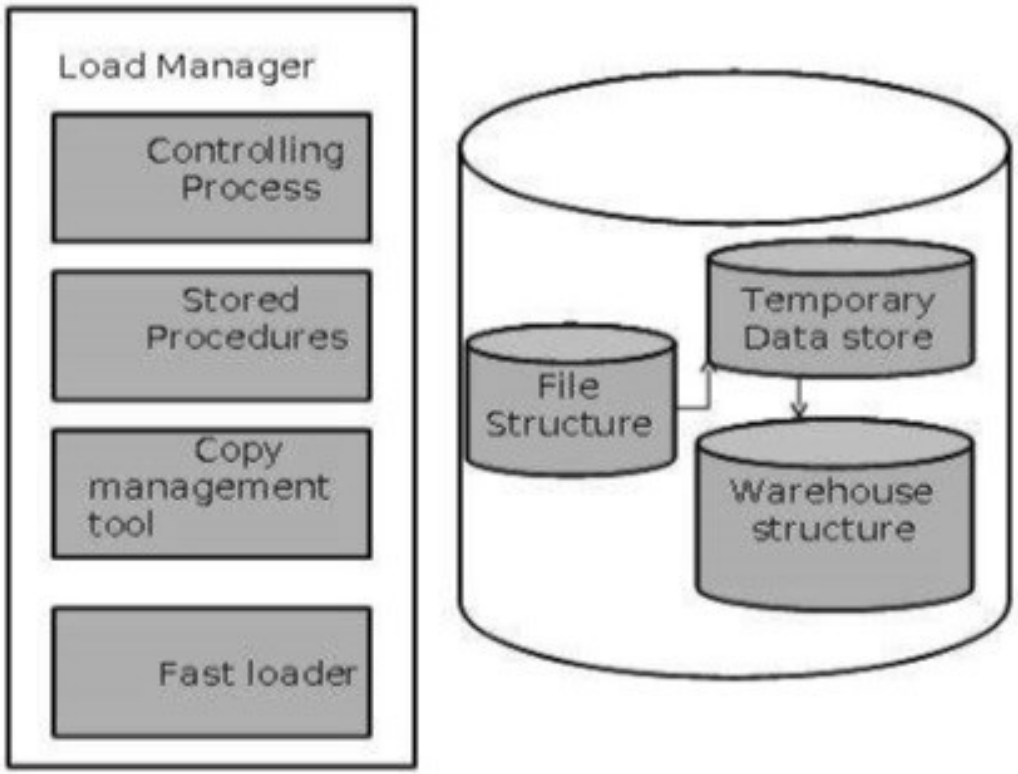
Data Warehouse Process Managers and Their functions

DW Process Managers

- Process managers are responsible for maintaining the flow of data both into and out of the data warehouse.
- There are three different types of process managers −
  - i. Load manager
  - ii. Warehouse manager
  - iii. Query manager

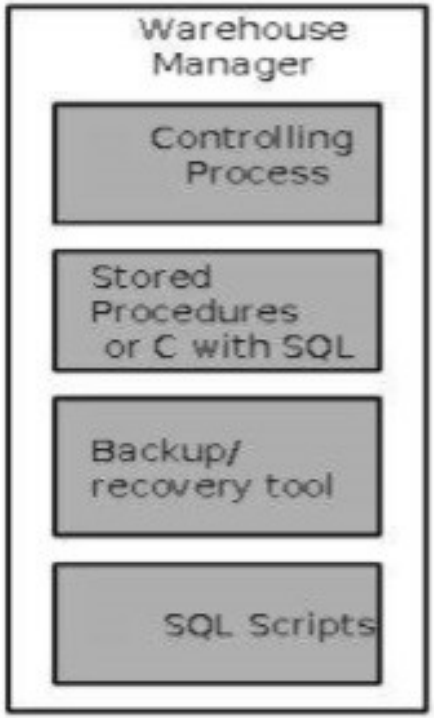
i) Load Manager

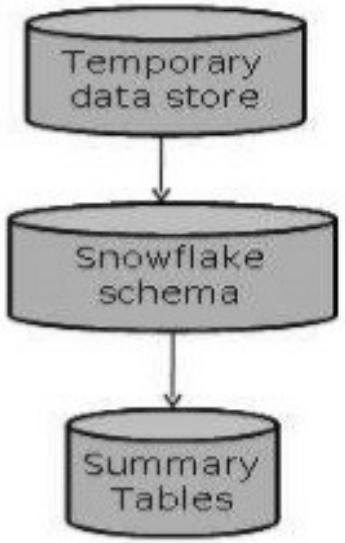
- Load manager performs the operations required to extract and load the data into the database.
- The size and complexity of a load manager varies between specific solutions from one data warehouse to another.
- The load manager does performs the following functions −
  - Extract data from the source system.
  - Fast load the extracted data into temporary data store.
  - Perform simple transformations into structure similar to the one in the data warehouse.



Warehouse manager:

- The warehouse manager is responsible for the warehouse management process.
- It consists of a third-party system software, C programs, and shell scripts.
- Functions of Warehouse Manager
  - Analyzes the data to perform consistency and referential integrity check
  - Generates normalizations.
  - Transforms and merges the source data of the temporary store into the published data warehouse.
  - Backs up the data in the data warehouse



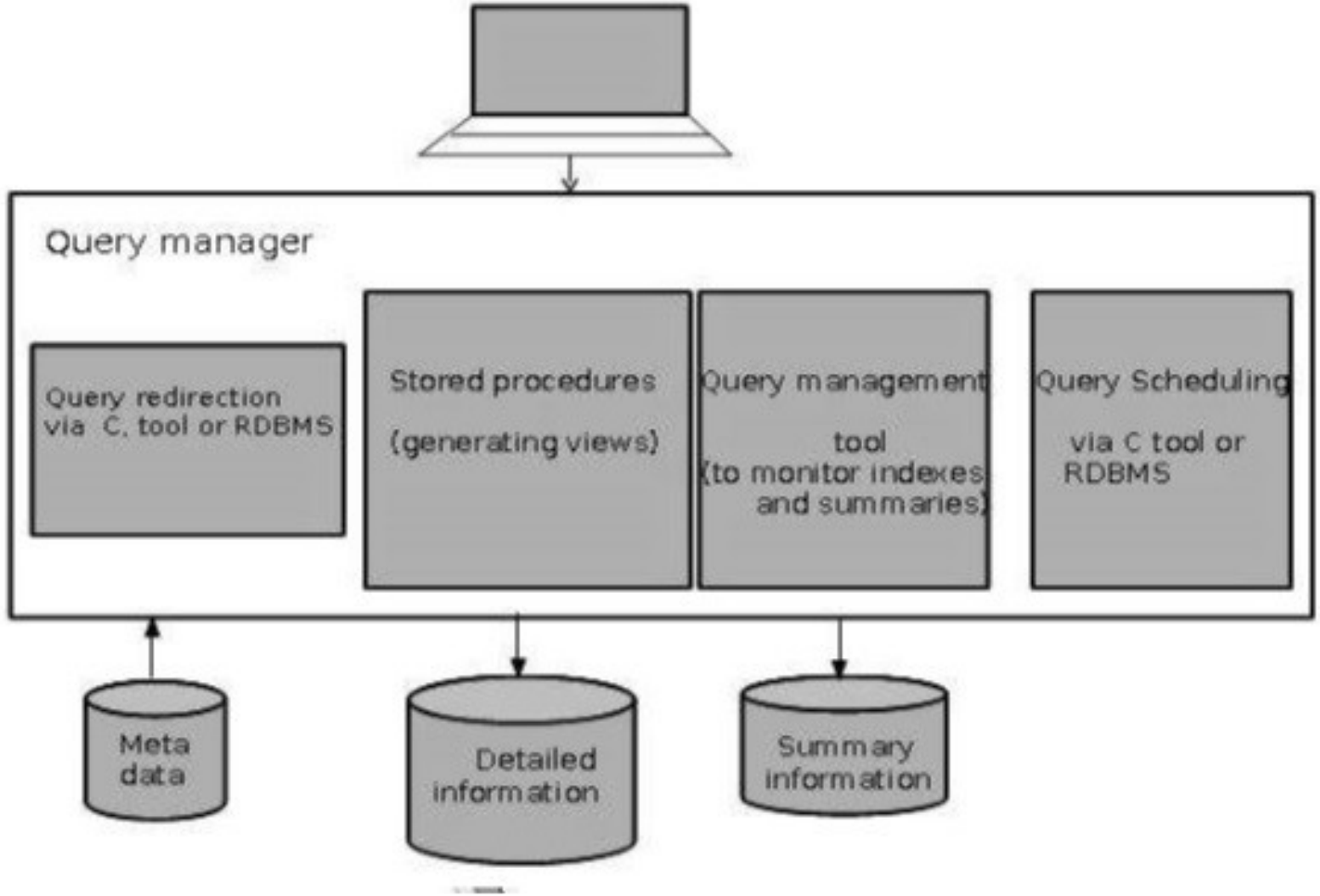


iii) Query Manager

- The query manager is responsible for directing the queries to suitable tables.
- By directing the queries to appropriate tables, it speeds up the query request and response process.
- In addition, the query manager is responsible for scheduling the execution of the queries posted by the user.

Functions of Query Manager

- It presents the data to the user in a form they understand.
- It schedules the execution of the queries posted by the end-user.
- It stores query profiles to allow the warehouse manager to determine which indexes and aggregations are appropriate.


Query manager
Query redirection via C, tool or RDBMS
Stored procedures (generating views)
Query management tool (to monitor indexes and summaries)
Query Scheduling via C tool or RDBMS
Meta data
Detailed information
Summary information

Data Warehouses and Data Warehouses Design

DW Design

- Data warehouse design is the process of creating a structure for organizing and storing data in a way that enables efficient querying, analysis, and reporting. The goal of data warehouse design is to provide a centralized repository of data that can be used to support business intelligence (BI) and decision-making processes.
- There are several key steps involved in designing a data warehouse:
  - Identify the business requirements
  - Choose a data model
  - Design the schema
  - Define the data sources
  - Create the ETL process
  - Populate the data warehouse
  - Test and validate

- **Top-down approach** : This approach was developed by Bill Inmon. It involves designing the data warehouse as a centralized repository of data that is integrated from various sources. The focus is on creating a single version of the truth, with a detailed data model and strict data governance.

- **Bottom-up approach**: This approach was developed by Ralph Kimball. It involves designing the data warehouse as a set of dimensional models, with a focus on providing quick and easy access to data for business users. The data warehouse is built incrementally, with each iteration adding more data and functionality.

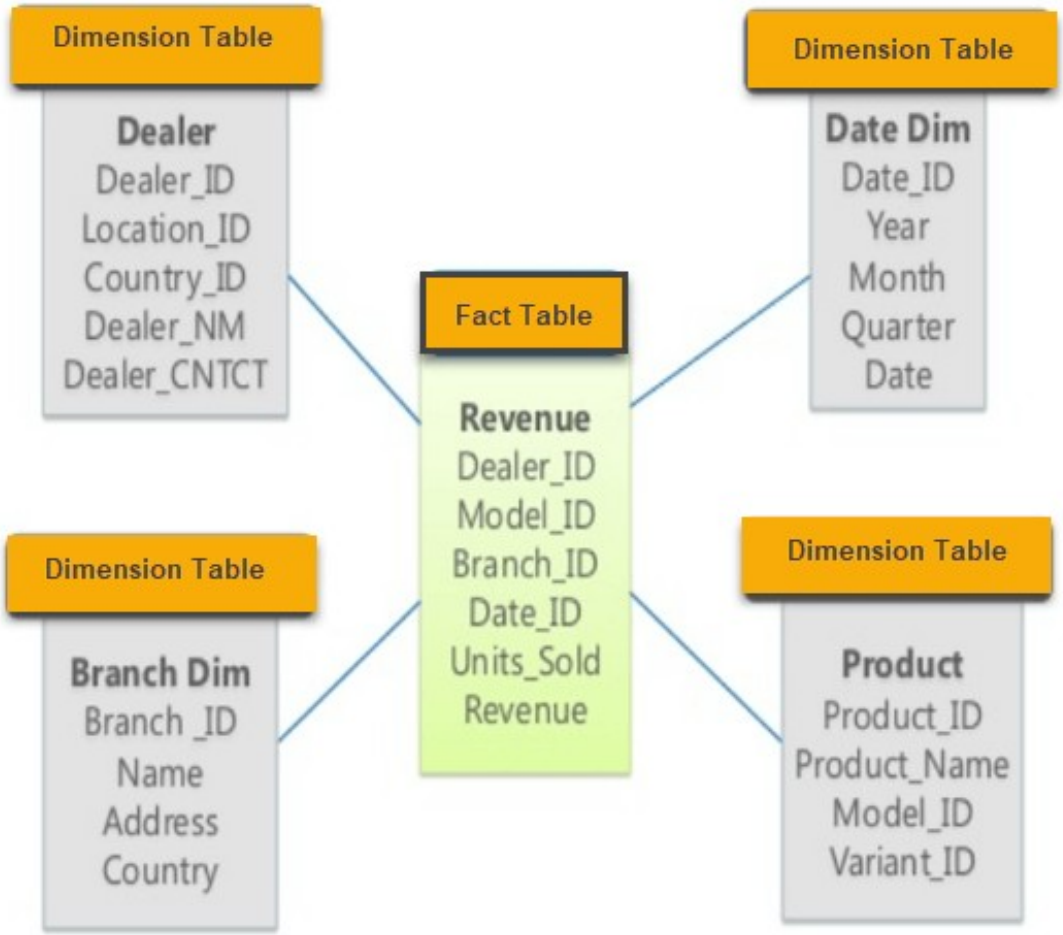
- **Hybrid approach**: This approach combines elements of both the Inmon and Kimball approaches. It involves designing the data warehouse as a centralized repository of data, but using dimensional modeling techniques to provide quick and easy access to data for business users.

Conceptual Modeling of Data Warehouse

- Modeling data warehouses: dimensions & measures
- **Star schema:** A fact table in the middle connected to a set of dimension tables
- **Snowflake schema:** A refinement of star schema where some dimensional hierarchy is normalized into a set of smaller dimension tables, forming a shape similar to snowflake
- **Fact constellations:** Multiple fact tables share dimension tables, viewed as a collection of stars, therefore called galaxy schema or fact constellation

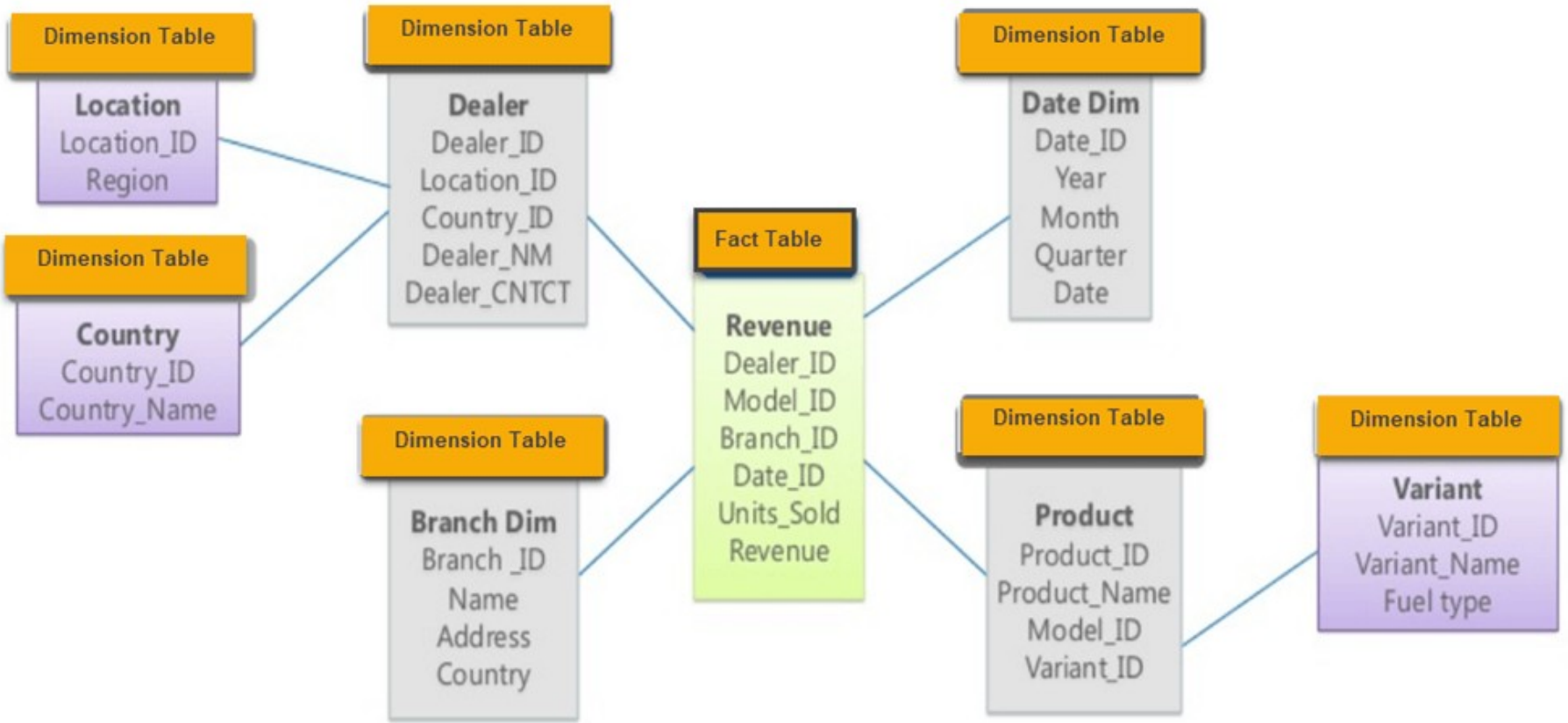
- **Star schema** is the fundamental schema. This schema is widely used to develop or build a data warehouse and dimensional data marts. **It includes one or more fact tables indexing any number of dimensional tables.**

Fact Table and Dimension Table are linked through foreign Key



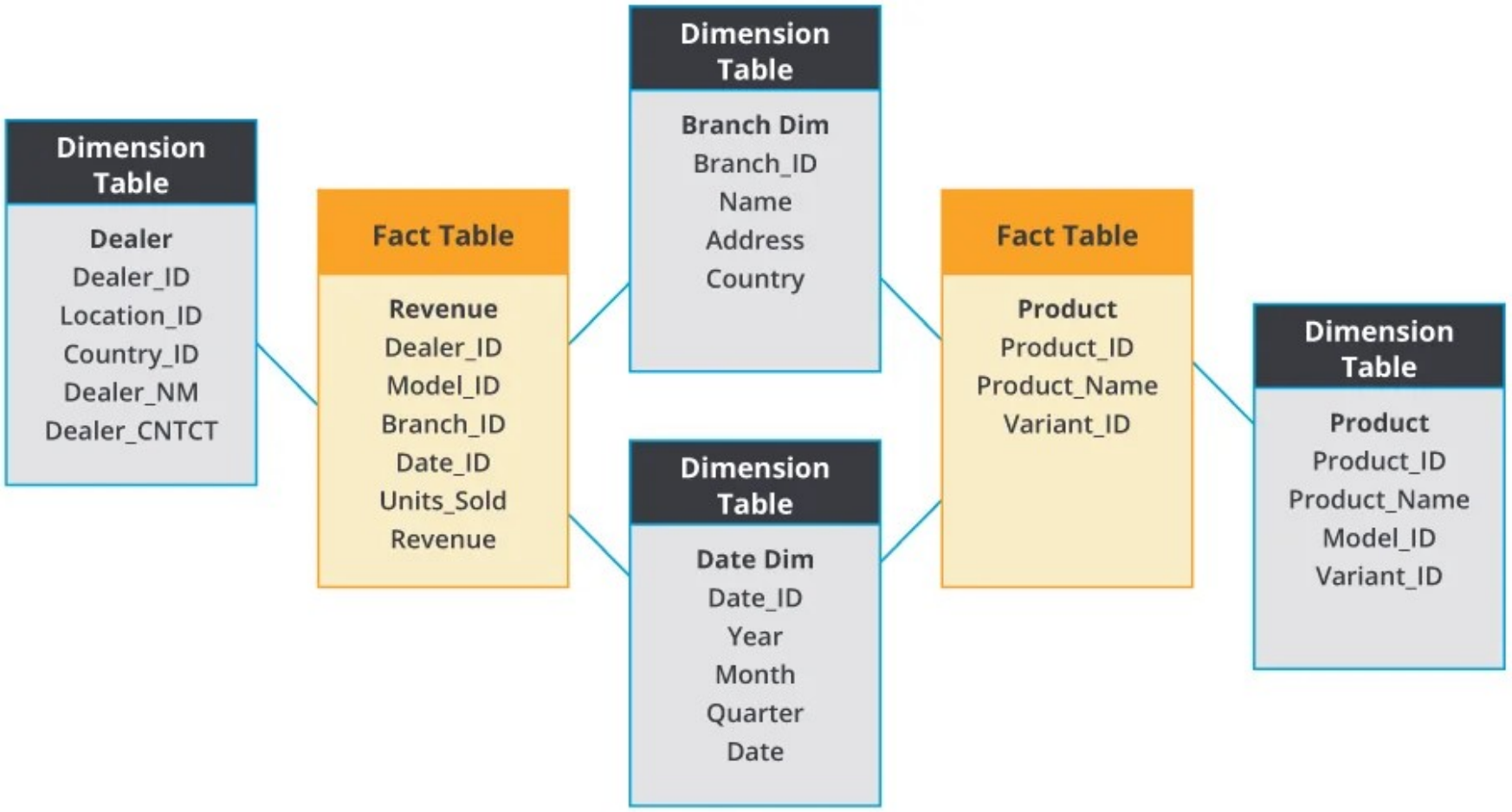
Snowflake schema

- A **snowflake schema** is a multi-dimensional data model that is an extension of a star schema, where dimension tables are broken down into sub dimensions.



Fact Constellations

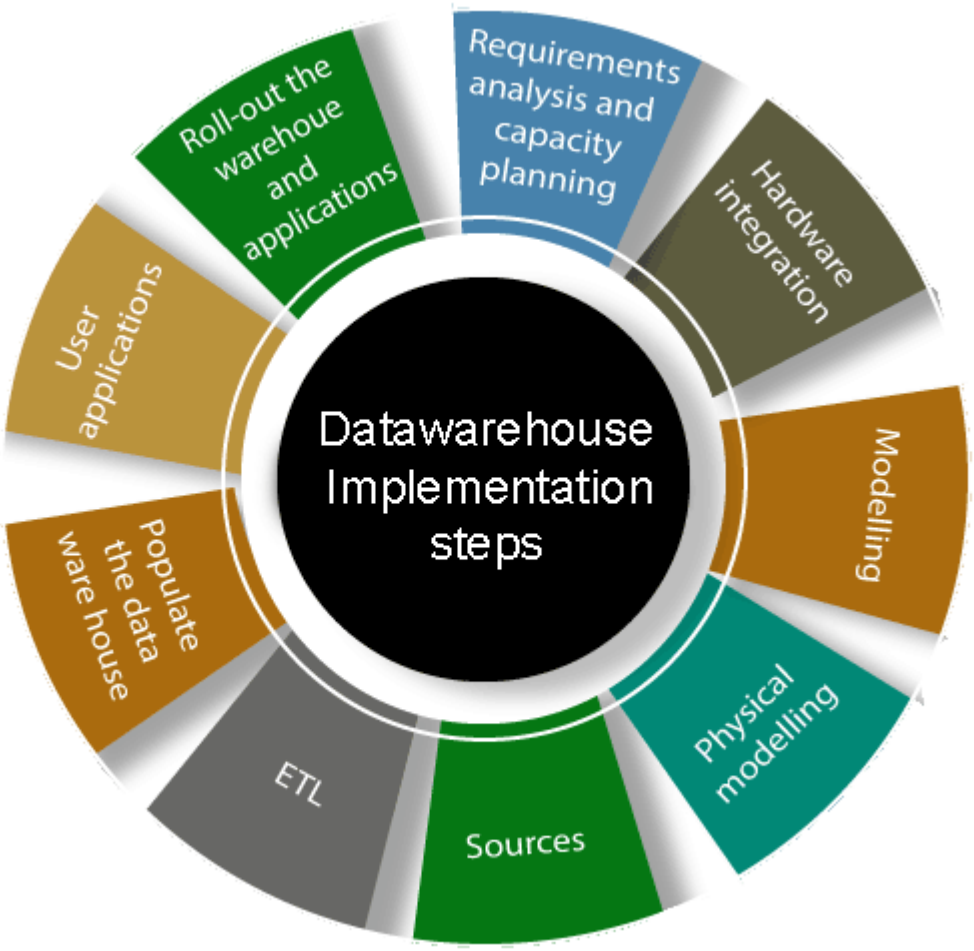
Fact Constellation (Multiple Fact) is a schema for representing multidimensional model. It is a collection of multiple fact tables having some common dimension tables.



Example of Galaxy Schema

Guidelines for Data Warehouse Implementation

Guidelines for Data Warehouse Implementation.



1. **Build incrementally:** Data warehouses must be built incrementally. Generally, it is recommended that a data marts may be created with one particular project in mind, and once it is implemented, several other sections of the enterprise may also want to implement similar systems. An enterprise data warehouses can then be implemented in an iterative manner allowing all data marts to extract information from the data warehouse.

2. **Need a champion:** A data warehouses project must have a champion who is active to carry out considerable researches into expected price and benefit of the project. Data warehousing projects requires inputs from many units in an enterprise and therefore needs to be driven by someone who is needed for interacting with people in the enterprises and can actively persuade colleagues.

3. **Senior management support:** A data warehouses project must be fully supported by senior management. Given the resource-intensive feature of such project and the time they can take to implement, a warehouse project signal for a sustained commitment from senior management.

4. Ensure quality: The only record that has been cleaned and is of a quality that is implicit by the organizations should be loaded in the data warehouses.

5. Corporate strategy: A data warehouse project must be suitable for corporate strategies and business goals. The purpose of the project must be defined before the beginning of the projects.

6. Business plan: The financial costs (hardware, software, and peopleware), expected advantage, and a project plan for a data warehouses project must be clearly outlined and understood by all stakeholders. Without such understanding, rumors about expenditure and benefits can become the only sources of data, subversion the projects.

7. Training: Data warehouses projects must not overlook data warehouses training requirements. For a data warehouses project to be successful, the customers must be trained to use the warehouses and to understand its capabilities.

- **Adaptability:** The project should build in flexibility so that changes may be made to the data warehouses if and when required. Like any system, a data warehouse will require to change, as the needs of an enterprise change.

- **Joint management:** The project must be handled by both IT and business professionals in the enterprise. To ensure that proper communication with the stakeholder and which the project is the target for assisting the enterprise's business, the business professional must be involved in the project along with technical professionals.

Exam questions

1. How data mining differs with data warehousing? [BIM 2021, Group A]
2. Why is data warehouse called nonvolatile? [BIM 2018, Group A]
3. Define data mart. [BIM 2018, Group A]
4. Explain Load manager. [BIM 2018, Group B]
5. What is operational data source? List some guidelines to be considered in data warehouse implementation. [BIM 2022, Group B]
6. Explain ETL process. Distinguish between OLAP and OLTP. [BIM 2021, Group C]
7. Define warehouse manager. Write the functions of Warehouse manager. [BIM 2017, Group C]