ASSIGNMENT - DATA WAREHOUSING

An Introduction to Data Warehousing

A **Data Warehouse (DW)** is a system used to store large amounts of historical data for reporting and analysis. Unlike transactional systems that deal with day-to-day operations, a data warehouse helps businesses make decisions by analyzing past data.

Key Features:

- Subject-oriented (organized around key subjects like sales, customers)
- Integrated (data from different sources is combined)
- Time-variant (stores historical data)
- Non-volatile (data is not frequently changed)

Purpose of Data Warehouse

The main purpose of a data warehouse is to:

- Help in **decision-making** by providing insights.
- Store **consolidated data** from multiple sources (like CRM, ERP, etc.).
- Enable **data analysis** over time (trend analysis, forecasting).
- Support **business intelligence (BI)** tools like dashboards and reports.
- Improve **data quality and consistency** by cleansing and integrating data.

Data Warehouse Architecture

There are typically **three levels** in a data warehouse architecture:

1. **Data Source Layer:**
   - Contains data from multiple operational databases and external sources.
   - Data is extracted through ETL (Extract, Transform, Load) processes.
2. **Data Staging/Integration Layer:**
   - Data is cleaned, transformed, and loaded into the warehouse.
   - ETL tools like Informatica, Talend, or SQL scripts are used.
3. **Data Storage Layer (Warehouse):**
   - The central database that stores the processed data.
   - Optimized for querying and reporting.
4. **Presentation Layer:**

- o Users access the data via reporting tools, dashboards, OLAP cubes, etc.

- **Metadata Layer**: Stores information about data (source, meaning, format).
- **Data Marts**: Smaller, subject-specific warehouses for departmental use.

Operational Data Store (ODS)

An **Operational Data Store** is a storage area used to hold current (real-time) operational data temporarily before it is moved into the data warehouse.

- Acts as a **bridge** between operational systems and the data warehouse.
- Provides **up-to-date information** for short-term decision-making.
- Unlike data warehouses, ODS is updated frequently.

Example: Showing live transaction data before it's summarized into a warehouse.

OLTP Vs Data Warehouse Applications

Feature	OLTP (Online Transaction Processing)	Data Warehouse (OLAP - Online Analytical Processing)
Purpose	Manage daily operations	Analyze historical data
Data Type	Real-time, current data	Historical, integrated data
Operations	Insert, Update, Delete	Select (Read/Query)
Speed	Fast transactions	Optimized for complex queries
Users	Clerks, Admins	Analysts, Managers
Database Design	Normalized (3NF)	Denormalized (Star/Snowflake schema)

Data Marts

A **Data Mart** is a smaller version of a data warehouse that is focused on a specific business area such as Sales, Finance, or Marketing.

Types:

- **Dependent**: Created from the central data warehouse.

- **Independent**: Built directly from operational systems.

Benefits:

- Faster access for a specific department.
- Easier to maintain and cost-effective.

Data Marts Vs Data Warehouses

Feature	Data Mart	Data Warehouse
Scope	Department-specific	Organization-wide
Size	Smaller	Large-scale
Source	From DW or operational systems	From multiple, integrated sources
Development Time	Shorter	Longer
Complexity	Low	High

Data Warehouse Life Cycle

The data warehouse development process typically includes the following steps:

1. **Planning**:
   - Define goals, data sources, and expected output.
2. **Requirements Gathering**:
   - Interact with stakeholders to understand data needs.
3. **Design**:
   - Design ETL, schema (Star/Snowflake), data flows.
4. **Data Extraction & Cleaning**:
   - Pull data from sources, clean and validate it.
5. **Loading & Storage**:
   - Load clean data into warehouse/storage systems.
6. **Implementation**:
   - Set up warehouse, dashboards, and reports.
7. **Testing & Deployment**:
   - Validate accuracy, performance, and security.

8. **Maintenance & Updates**:
   o Regularly update data and optimize performance.