Day 1 Assignment: Amaan Shaikh

Qs:

Data Warehouses, ETL, OLTP & OLAP, ACID

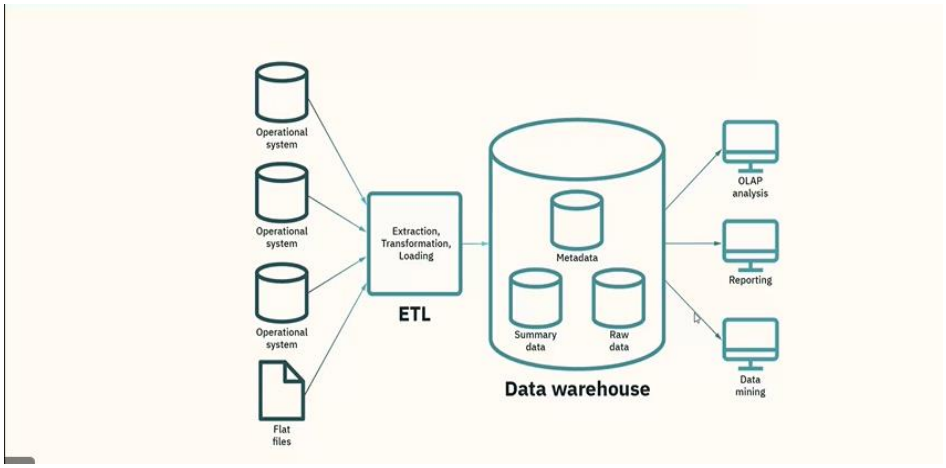
1. Data Warehouse:

A **Data Warehouse** is a centralised repository that stores large volumes of structured and semi-structured data from multiple sources, designed to support query, BI and analysis rather than transaction processing. It supports decision-making by providing a consolidated view of data over time.

Enables an organisation to run powerful analytics on huge volumes of data in ways taht standard systems cannot.

Benefits: Better data optimised for querying and analysis, Faster decision making, acts as central repository for variety of data forms, higher quality data.

Components of a Data Warehouse:



A. Data processing Layer

Data Sources: Systems and databases from which data is extracted, including CRM, ERP, and external data sources.

ETL (Extract, Transform, Load) Process: The process of extracting data from sources, transforming it for consistency, and loading it into the data warehouse.

Data Staging Area: A temporary storage space for data during the ETL process, used for cleansing and validation before loading into the warehouse.

B. Storage Layer

Data Storage (Data Warehouse): The central repository where structured data is stored, optimised for fast querying and analysis.

Metadata: Information about the data, such as definitions and processing rules, that helps manage and understand the data.

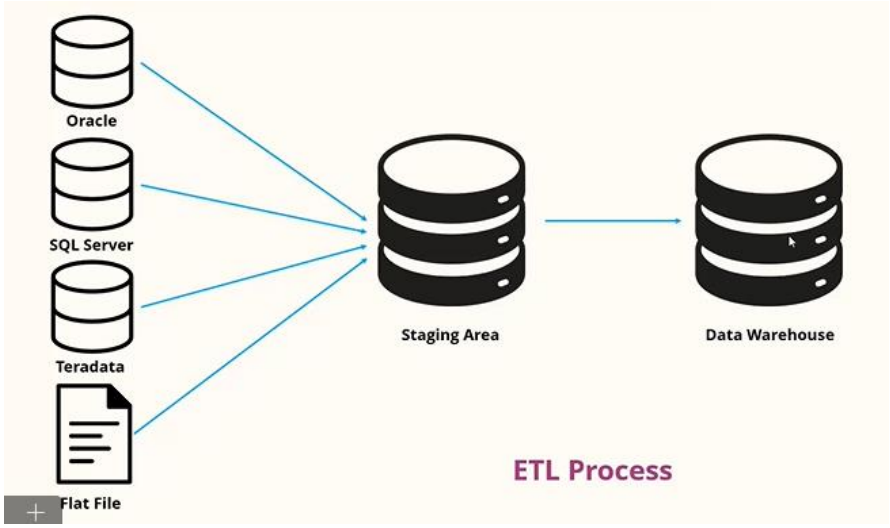
Data Marts: Subsets of the data warehouse tailored for specific business units or functions, like sales or finance.

C. Analysis Layer

OLAP (Online Analytical Processing) Tools: Tools used for complex queries and multidimensional analysis of the data in the warehouse.

BI (Business Intelligence) Tools: Applications for creating reports, dashboards, and visualisations based on warehouse data.

2. ETL:



Extract: Data is extracted from various sources, such as databases, APIs, files, or other systems and converted into a standard format (as required) for standardised processing.

Transform: The extracted data is cleaned, formatted, and transformed to meet the required standards or schema of the target system. This may include data cleansing, filtering, aggregation, and applying business rules.

Load: The transformed data is then loaded into the target database, data warehouse, or another destination for analysis, reporting, or further processing.Incremental or Overwrite loads are done daily, weekly or monthly.

3. OLTP & OLAP:

OLTP (Online Transaction Processing)

OLTP systems are designed to handle a large number of short online transaction queries and updates. They focus on transaction processing, data entry, and retrieval, providing fast query responses and ensuring data integrity.

Characteristics: High transaction volume, fast query processing, and frequent updates.

Examples: Banking Systems, E-Commerce Platforms

OLAP (Online Analytical Processing)

OLAP systems are designed for complex queries and data analysis, providing multi-dimensional views of data for reporting and decision-making. They focus on querying large datasets to analyse trends and patterns.

Characteristics: Complex queries, multidimensional analysis, and large-scale data aggregation, quick response time to queries.

Examples: **BI Tools, Data Warehouses**

4. ACID in Distributed Databases:

Atomicity:

- The **Two-Phase Commit (2PC)** protocol is commonly used. In the first phase, all participating databases agree to commit or abort the transaction. In the second phase, the transaction is either fully committed or rolled back across all databases.

- **Example**: If a money transfer between two bank accounts stored in different databases occurs, either both accounts are updated (withdrawal and deposit), or neither is, ensuring atomicity.

Consistency:

- By ensuring that all participating databases enforce the same validation rules.
- **Example**: A distributed order system checks inventory across multiple warehouses. The transaction ensures the total quantity remains accurate, following business rules, even after splitting the order across different locations.

Isolation:

- **Implementation**: **Concurrency control mechanisms**, like locks or timestamp ordering are used.
- **Example**: Two users placing orders at the same time from different locations shouldn't see each other's intermediate steps, preventing inconsistencies like overbooking of products.

Durability:

- Distributed databases use **replication** and **logging** to ensure that the data changes are permanent and can survive system failures.
- **Example**: After a purchase is completed, the order details are replicated across several data centers, so even if one server fails, the data is not lost.

5. Problems with ACID

a**. Scalability:** Limits scalability and performance in large, distributed environments.

b. **Complexity:** Requires use of complex protocols for implementation.

c. **Resources:** Implementation requires significant computing and storage resources.

d. **Performance Overhead:** Slows down transaction processing.