NAME: ASENATH A

DATA WAREHOUSE

A Data warehouse is mainly used to handle big data. It contains a large amount of historical data. A Data warehouse is a system that stores large amounts of data from different places in one central location. It helps businesses analyze their data to make better decisions using the insights that are found. It helps organizations understand the current trends. We can use a Data warehouse to look at lots of data as a single thing rather than handling everyday tasks. Business Intelligence is used in Data warehouses. Overally, it gets data from many different places and combines it into a single data repository.

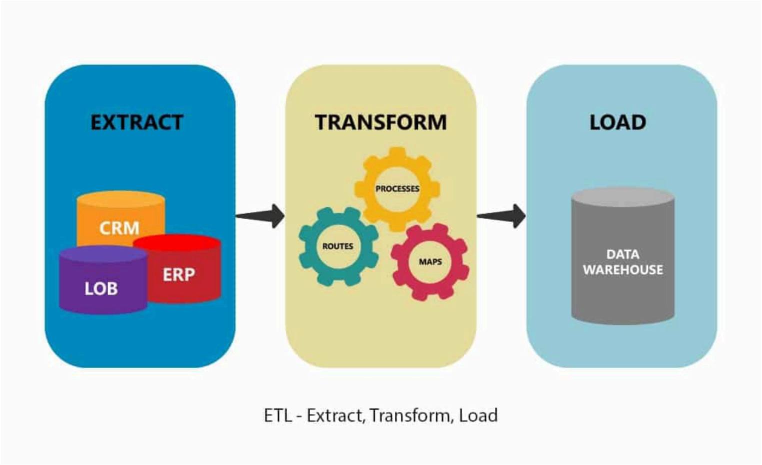
ETL (Extract, Transform, Load)

ETL is the process of moving data into a data warehouse. First of all data is extracted from different sources . Then it is cleaned and transformed for the necessities. At last the data is placed into the data warehouse where it can be used for further analysis.

Extract: Here we will gather data from various sources, like databases, files, or cloud services. This data might be in different formats or stored in different places. We will extract it as a single content.

Transform: Here the data is cleaned and converted into a format that is suitable for analysis. Here we will remove duplicates, noisy data so as to fit with the rest.

Load: The final step is to load the transformed data into the data warehouse. After loading the data, it will be ready for analysis and other insight extractions.

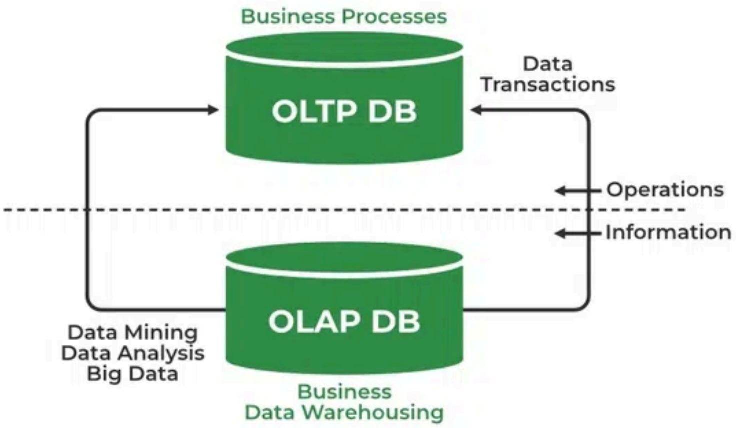


OLAP (Online Analytical Processing)

OLAP is used to maintain the transactions in a database. We can use it to analyze the data quickly. Aggregated data from OLTP is used here to understand the data in a deeper way. It applies complex queries for business intelligence and analysis. It takes more time for responses since it has a large amount of data.

OLTP (Online Transaction Processing)

OLTP handles tasks like adding, updating, or deleting data in a database. It maintains transactions in a database. Banking and credit card related activities are handled through OLTP. The processing time is fast here compared to the OLAP. It is read, written and updated frequently. It uses traditional DBMS to have large volumes of data.



Explain how Acid Properties are implemented In Distributed Database Transactions.

ACID properties are used to ensure data integrity and reliability in traditional database systems. ACID properties are a set of four key principles.

Atomicity: Transaction is treated here as a single unit. Either all operations take place completely or none takes place. For example, If an online payment transaction fails, the system automatically reverses any changes to avoid partial transactions.

Consistency: It maintains integrity before and after the transaction by applying certain rules. For example, after some transaction between accounts, the total balance of all accounts remains accurate and unchanged.

Isolation: Multiple transactions take place individually here without any interference. For example, While one user is updating their profile, another user's changes to their profile are not visible until their transaction is complete.

Durability: If a transaction gets committed it will remain the same without any further changes. It maintains reliability. For example, After a successful order placement, the details remain recorded in the database even if the system crashes immediately.

Find the problems with ACID, and potential solutions without going to NoSQL.

1. **Atomicity**:
   - **Problem**: Ensuring atomicity across distributed systems is a complex process
   - **Solution**: Using distributed transaction protocols like the Two-Phase Commit to manage and ensure atomicity.
2. **Consistency**:
   - **Problem**: Maintaining consistency across multiple nodes can be challenging.
   - **Solution**: Implementing strong consistency models to ensure that all nodes reflect the same state.
3. **Isolation**:
   - **Problem**: High isolation levels can lead to performance bottlenecks due to locking.
   - **Solution**: Using techniques to reduce locking while still providing a high level of isolation.
4. **Durability**:
   - **Problem**: Ensuring durability in distributed systems can be difficult if failures occur leading to data loss or inconsistency.
   - **Solution**: Use strategies to ensure that committed data is reliably stored across multiple nodes.