DSA 2040A US 2025 LAB 2

Lab Manual: EXTENDED LAB MANUAL: OLTP + OLAP Integration

Reflection

1. Why is the OLTP system normalized and the OLAP system denormalized?

OLTP (Online Transaction Processing) Systems: Normalized

Purpose: Designed for high-volume, real-time transactional operations (e.g., inserts, updates, deletes). Think of systems supporting online banking, e-commerce transactions, or airline reservations.

Normalization: Data is organized into multiple tables to reduce data redundancy and improve data integrity. This involves following normalization forms (1NF, 2NF, 3NF, BCNF, etc.) to eliminate duplicate data and ensure that data dependencies are logical.

Benefits of Normalization in OLTP

Reduced Redundancy: Minimizes storage space and prevents data inconsistencies. For example, a customer's address is stored only once, even if they place multiple orders.

Improved Data Integrity: Ensures that data is consistent and accurate across the database. Updates to a single piece of information (e.g., a customer's address) only need to happen in one place.

Faster Writes (DML Operations): Because data is stored in smaller, distinct tables, insert, update, and delete operations are more efficient, as less data needs to be modified.

Concurrency Control: Easier to manage concurrent access and lock specific rows or tables without affecting large parts of the database.

OLAP (Online Analytical Processing) Systems: Denormalized

Purpose: Designed for complex analytical queries, reporting, and business intelligence. These systems support data aggregation, trend analysis, and strategic decision-making. Think of data warehouses or data marts.

Denormalization: Involves intentionally introducing redundancy into the database, often by combining data from multiple normalized tables into fewer, larger tables. This is commonly seen in dimensional modeling (star or snowflake schemas).

Benefits of Denormalization in OLAP

Faster Reads (Query Performance): By reducing the number of table joins required for analytical queries, query execution time is significantly decreased. Data needed for a report is often pre-joined or pre-aggregated.

Simplified Queries: Analytical queries become simpler to write and understand because data is more consolidated.

Optimized for Aggregation: Denormalized structures, especially fact and dimension tables, are inherently optimized for aggregate functions (SUM, AVG, COUNT) critical for analytical reporting.

Improved User Experience: Faster query response times lead to a more fluid experience for business analysts and decision-makers.

2. What challenges would you face if you ran analytical queries directly on the OLTP system?

Running complex analytical queries directly on a production OLTP system is generally a bad practice due to several significant challenges:

Performance Degradation for OLTP Operations

Analytical queries are typically resource-intensive, requiring full table scans, complex joins, and heavy aggregation.

These queries consume significant CPU, I/O, and memory resources, leading to contention with the routine DML (Data Manipulation Language) operations of the OLTP system.

This resource contention can cause transactional queries to slow down dramatically, impacting the responsiveness of critical business applications (e.g., delaying customer transactions, increasing website load times).

Database Locking and Concurrency Issues

Long-running analytical queries can acquire locks on tables or rows for extended periods to ensure data consistency during their execution.

These locks can block or dead-lock concurrent OLTP transactions, leading to transaction timeouts, application errors, and user frustration.

Impact on Data Integrity and Availability

Increased load and contention raise the risk of system instability, potential crashes, or unplanned downtime for the OLTP database.

Any disruption to the OLTP system directly affects core business operations, leading to financial losses and reputational damage.

Inappropriate Data Structure

OLTP databases are optimized for fast writing (normalization), which means analytical queries often require numerous complex joins across many tables. This inherently makes analytical queries inefficient and slow on a normalized schema.

The absence of pre-computed aggregates or summarized data means analytical queries must perform these calculations on the fly, further increasing their resource footprint.

Data Latency vs. Freshness

While OLTP data is "fresh," direct querying means analysts are working on the exact operational data, which might not be desirable for historical analysis or long-term trends if intermediate transactional states are being captured. OLAP systems often contain snapshot data or historical aggregates.

3. How can automation (e.g., scheduled ETL jobs) help in a real-world data pipeline?

Automation, particularly through scheduled ETL (Extract, Transform, Load) jobs, is a cornerstone of efficient and reliable data pipelines, addressing the challenges mentioned above:

Efficient Data Movement and Synchronization

Extract: Automatically pulls data from various source systems (primarily OLTP databases, but also other internal/external data sources) at predefined intervals (e.g., hourly, daily, nightly).

Transform: Applies a series of rules, cleanups, aggregations, and business logic to the extracted data. This prepares the data for analytical consumption, ensuring consistency, quality, and readiness for reporting (e.g., converting data types, handling missing values, calculating KPIs, rolling up granular data).

Load: Inserts the transformed data into the target analytical environment (e.g., data warehouse, data mart, data lake). This ensures that the OLAP system is populated with up-to-date and ready-to-analyze information.

Decoupling OLTP and OLAP Systems

ETL jobs create a clear separation between operational and analytical workloads. Analytical queries run on the dedicated OLAP system, which is optimized for complex reads, without impacting the performance or availability of the production OLTP system.

Ensuring Data Freshness and Timeliness

Scheduled execution guarantees that the analytical data is consistently updated without manual intervention. This provides business users with reliable and timely insights, enabling data-driven decision-making based on the latest available information.

Improved Data Quality and Consistency

Automated transformation steps enforce data quality rules and standardize data formats across different sources, reducing human error and ensuring that reports and dashboards are based on clean, consistent data.

Scalability and Maintainability

As data volumes grow, automated pipelines can be scaled more easily than manual processes. They also provide a structured and auditable way to manage data flows, making it easier to troubleshoot issues and maintain the pipeline over time.

Resource Optimization

ETL jobs can be scheduled during off-peak hours (e.g., overnight) to minimize resource contention, further optimizing the use of infrastructure.