Explore Data Warehouses

Waliu Ayuba

Question 1

Data warehouses are technical systems developed for analytics and reporting, distinct from functional databases that support day-to-day transaction processing. They are used in relational databases, mainly focusing on fact tables, star schemas, and comparisons with NoSQL databases.

Fact Tables and Star Schemas in Relational Databases

- Fact Tables: Central to a data warehouse, fact tables store quantitative data for analysis and reporting. These tables contain metrics or facts about a business process (e.g., sales amount, number of items sold). Keys in the fact table usually correspond to foreign keys that relate to dimension tables.
- Star Schema: It's a database schema that manages data into one or additional fact tables referencing any number of dimension tables.The schema resembles a star, with a fact table at the center and dimension tables casting out. Dimension Tables contain descriptive attributes (or dimensions) associated with fact data. Examples such as customer details, product information, and time data.
- Simplicity and Performance: The star schema streamlines data modeling and querying. It's also efficient for large datasets typically encountered in data warehouses.

Transactional Databases for OLAP (Online Analytical Processing) typically requires complex queries, multi-dimensional data analysis, and aggregations.

Transactional databases (OLTP systems) are designed for fast, dependable transaction processing but not essentially for complex analytical queries. Therefore, while technically possible, using a transactional OLAP database is usually inadequate or practical.

Relational Database with Star Schema vs. NoSQL for Data Warehousing

- Performance and Complexity: Relational databases with star schemas are optimized for specific queries typical in data warehousing. They offer exemplary implementation for complex queries that apply joins and aggregations.
- ACID Compliance: Relational databases provide ACID (Atomicity, Consistency, Isolation, Durability) properties, securing reliable transaction processing.
- Familiarity and Tools: Many organizations know relational databases, have mechanisms and have expertise in SQL. Transitioning to a NoSQL solution might involve a steep learning curve and changes in infrastructure.
- NoSQL Databases: NoSQL databases are highly scalable and engineered to handle extensive amounts of data. They are perfect for unstructured data or where data schemas are not specified or known in advance.

However, their strengths provide a further level of support for complex queries and transactional integrity than relational systems.

In recap, while NoSQL databases offer scalability and flexibility, especially for unstructured data, relational databases remain a vital choice for data warehousing due to their robust querying abilities, ACID compliance, and the widespread familiarity of SQL among database professionals. The choice often relies on an organization's specific requirements and existing infrastructure.

Question 2

Data warehouses, data marts, and data lakes are three distinct types of data storage systems used for handling large volumes of data differently.

Data Warehouse:

- Definition: The data warehouse is a centralized repository for structured data. It is preprocessed for analytics and business intelligence. The data is highly classified, and the schema is often created before the data is stored.
- Example of Use: An organization's strength is using a data warehouse to converge data from departments such as sales, finance, and processes for company-wide analysis and reporting.
- Characteristics: Centralized, multiple subject areas, organization-wide use, many data sources, scheduled top-down, complete detailed data.

Data Mart:

- Definition: The data mart is a subset of a data warehouse tailored to the needs of a typical business unit or department. It keeps a smaller amount of data focused on a particular subject.
- Example of Use: A company's marketing department may have a data mart to store and analyze consumer conduct and campaign performance data.
- Characteristics: Decentralized, specific topic area focus, used by a single community or department, fewer data sources, developed bottom-up, may contain summarized data.

Data Lake:

- Definition: A data lake is a vast pool of raw data, including unstructured and semi-structured data. It holds data in its natural format and only processes once needed.
- Example of Use: A company might employ a data lake to store all types of raw data, such as logs, social media content, sensor data, etc., which can be later processed for diverse analytical purposes like machine learning.
- Characteristics: All data types, schema-on-read, flexible, and low-cost storage, can contain uncurated raw data used by different users, including data scientists and engineers.

When to use each?

- Data Lakes: These systems handle all data types, especially extensive unstructured or semi-structured data, for better flexibility and lower costs.
- Data Warehouses are for structured relational data, precisely when fast query implementation and data quality are required.
- Data Marts are for department-specific applications that focus on a specific business need or data set. They are usually formed from data already kept in a data warehouse.

The key differences between data warehouses, data marts, and data lakes lie in their objectives, data sources, scope, and utilization:

Data Warehouse:

- Objectives: It's centralized storage for structured data from various sources organized for comprehensive analytics across numerous business units.
- Users & Scope: A data warehouse is utilized organization-wide, integrates multiple subject areas, and has large amounts of data, ranging from gigabytes to petabytes.
- Data Detail & Quality: Has complete, detailed, and highly curated data due to preprocessing and curation.
- Performance: Optimized for fast query implementation for structured data.
- Examples: Frequently used for batch reporting, business intelligence (BI), and visualizations R DASCA.

Data Mart:

- Purpose: Focused on specific subject areas or departments, it's a decentralized storage, often a subset of a data warehouse.
- Users & Scope: This is typically smaller and tailored to business units like sales, marketing, or finance.
- Data Detail & Quality: This may harbor summarized data, which varies depending on the source and preprocessing.
- Performance: Optimized for query and reporting performance through data volume decrease.
- Examples: Ideal for domain-specific analytics and decision-making within specific company departmentsR DASCA.

Data Lake:

- The purpose is to create a focused repository for storing any data type, including structured, semi-structured, and unstructured data.
- Users & Scope: It can accommodate a broad range of user requirements and scale from small to large volumes of data.
- Data Detail & Quality: Contains raw, unprocessed data with flexibility in preprocessing options. The data quality depends on the curation steps.
- Performance: Optimized for cost and storage volume rather than speed.
- Examples: It is suitable for big data applications, AI, machine learning, and organizations that must store vast amounts of varied data for future analysis R CloudZero

In practice, most large organizations use a variety of these storage systems. Data lakes repeatedly serve as primary repositories, with data being loaded into warehouses and marts for typical use cases. The selection between these systems relies on factors like data types, volume, cost, and the exact analytical needs of the organization.

Question 3

Fact Table Design: "BirdStrikeFacts"

- Purpose: The "BirdStrikeFacts" table supports analyses of bird strike incidents over time and across different geographical locations. It can help to identify patterns and trends, such as peak bird strike periods and high-risk locations.

Structure:

Primary Key: A composite key consisting of FactID (a unique identifier for each record).

Foreign Keys:

- TimeID: References a 'Time' dimension table (detailing periods like week, month, and year).
- LocationID: References a 'Location' dimension table (detailing location attributes like region and airport).

Measures:

- TotalStrikes: The total number of bird strikes.
- AverageStrikes: The average number of bird strikes (calculated for specified periods or locations).

Descriptive Attributes:

- (Optional) Attributes like Severity, ImpactType, or BirdSpecies could be included if these aspects are crucial for the analysis.

Example Entity-Relationship Diagram (ERD):

- BirdStrikeFacts Table: Central fact table.
- Time Dimension Table: Contains attributes like TimeID, Day, Week, Month, and Year.
- Location Dimension Table: Contains attributes like LocationID, AirportName, and Region.

SQL Code for Creating the Fact Table:

```
"CREATE TABLE BirdStrikeFacts (
    FactID INT AUTO_INCREMENT PRIMARY KEY,
    TimeID INT,
    LocationID INT,
    TotalStrikes INT,
    AverageStrikes DECIMAL(10, 2),
    FOREIGN KEY (TimeID) REFERENCES Time(TimeID),
    FOREIGN KEY (LocationID) REFERENCES Location(LocationID)
);"
```

Design Rationale:

- Time Dimension: This feature enables the analysis of trends across varying periods. Location Dimension: This tool allows for geographic analysis of bird strikes, essential for identifying high-risk areas.
- Measures: To better understand bird strike incidents, provide critical metrics such as total and average strikes to analyze the extent and distribution of incidents.

Use Case: - Seasonal Analysis: It is suggested to identify airports or regions that are at a high risk of wildlife-related accidents and take necessary measures to manage the wildlife in those areas. Additionally, pilots should be informed and advised of the potential risks in these locations.

- Regional Safety Planning: Identify airports or regions that are at high risk of wildlife-related incidents, and implement targeted wildlife management strategies and pilot advisories in those areas.