Member-only story

Google Cloud Data Engineer Interview Questions & Answers

Vishal Bulbule

8 min read · Just now

Conceptual Differences Between a Database, Data Warehouse, Data Lake, and Data Mart

1. **Database:** A database is an organized system for storing, managing, and retrieving structured data. It is optimized for handling daily transactional processes and supports real-time data operations.

2. **Data Warehouse:** A data warehouse is a centralized repository designed to consolidate and store structured data from various sources. It is optimized for complex querying, reporting, and analysis, and typically handles historical data.

3. **Data Lake:** A data lake is a vast storage repository that holds large volumes of raw and diverse data, including structured, semi-structured,

and unstructured data. It provides flexibility for storing data in its native format and supports advanced analytics and machine learning.

4. **Data Mart:** A data mart is a specialized subset of a data warehouse, focused on a specific business area or department. It provides tailored access to relevant data for particular business needs and analytical tasks.

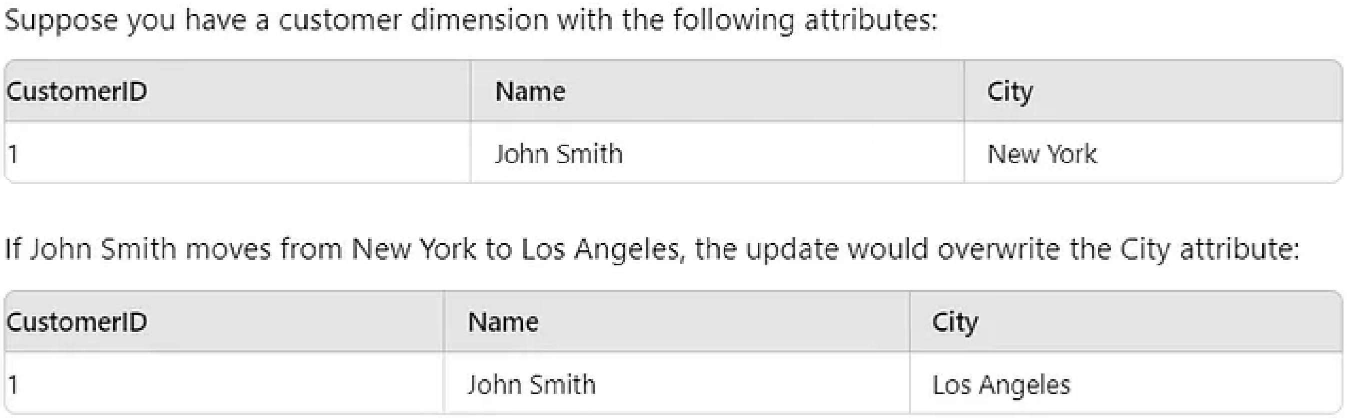
What are Fact Tables and Dimension Tables?

- **Fact Tables:** Store quantitative data (measures) for analysis, such as sales amounts, and contain foreign keys that reference dimension tables.
- **Dimension Tables:** Store descriptive attributes (dimensions) related to the facts, such as time, product, and location.

What is a Slowly Changing Dimension (SCD)?

A Slowly Changing Dimension (SCD) is a dimension in a data warehouse that changes slowly over time, rather than changing on a regular schedule or in real-time. There are different types of SCDs:

- **SCD Type 1:** Overwrites existing data, no history tracking.



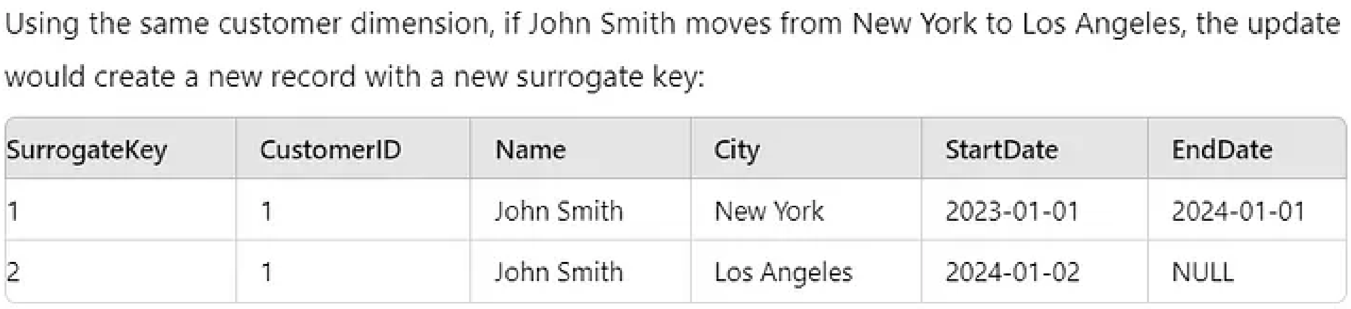
Suppose you have a customer dimension with the following attributes:

CustomerID	Name	City
1	John Smith	New York

If John Smith moves from New York to Los Angeles, the update would overwrite the City attribute:

CustomerID	Name	City
1	John Smith	Los Angeles

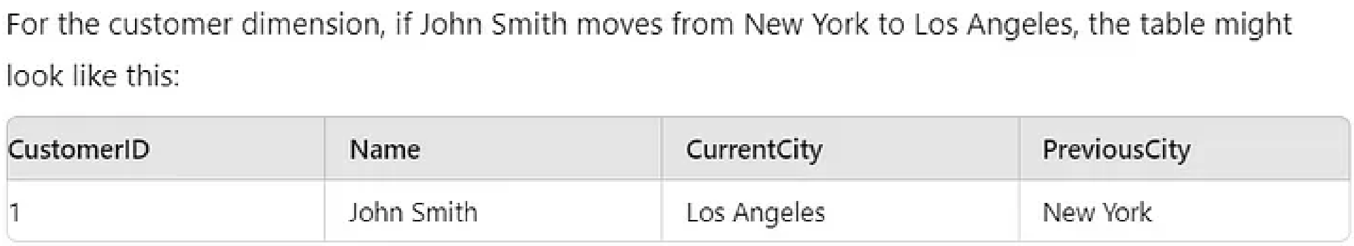
- **SCD Type 2:** Adds new records for changes, keeps full history with separate surrogate keys.



Using the same customer dimension, if John Smith moves from New York to Los Angeles, the update would create a new record with a new surrogate key:

SurrogateKey	CustomerID	Name	City	StartDate	EndDate
1	1	John Smith	New York	2023-01-01	2024-01-01
2	1	John Smith	Los Angeles	2024-01-02	NULL

- **SCD Type 3:** Adds new columns to track limited history (typically one previous value).



For the customer dimension, if John Smith moves from New York to Los Angeles, the table might look like this:

CustomerID	Name	CurrentCity	PreviousCity
1	John Smith	Los Angeles	New York

Explain Difference between OLTP vs. OLAP

OLTP (Online Transaction Processing):

OLTP systems are designed to manage transactional data. They handle a large number of short online transactions such as insert, update, and delete operations.

- **Purpose:** To support day-to-day operations and transactional applications.
- **Examples:** Banking systems, order entry systems, retail sales systems.

OLAP (Online Analytical Processing):

OLAP systems are designed to manage and analyze large volumes of data. They support complex queries, multidimensional analysis, and business intelligence activities.

- **Purpose:** To enable strategic analysis and decision-making through data warehousing and business intelligence tools.
- **Examples:** Data warehouses, financial reporting systems, market research analysis.

Explain the difference Between ETL and ELT

ETL (Extract, Transform, Load) and ELT (Extract, Load, Transform) are two data integration processes that differ primarily in the sequence of their steps and the way data transformation is handled. Below, we'll explore the key differences between these two approaches:

ETL (Extract, Transform, Load)

- **Extract:** Data is extracted from various sources. **Transform:** Data is transformed and cleaned before being loaded into the target system. **Load:** Transformed data is then loaded into the target data warehouse or data storage system.

ELT (Extract, Load, Transform)

- **Extract:** Data is extracted from various sources. **Load:** Raw data is immediately loaded into the target data storage system. **Transform:** Data transformation is performed within the target system after loading.

What are the key differences between SQL and NoSQL databases?

- **SQL Databases:** Use structured query language (SQL) for defining and manipulating data. They are relational databases that store data in tables with predefined schemas. Examples include MySQL, PostgreSQL, and Oracle.
- **NoSQL Databases:** Designed for flexible schema and can store unstructured or semi-structured data. They include document stores, key-value stores, wide-column stores, and graph databases. Examples include MongoDB, Cassandra, and Redis.

Can you explain the concept of data partitioning and why it is used?

Data partitioning is the process of dividing a database or data warehouse into smaller, more manageable pieces, or partitions. It is used to improve performance, manageability, and scalability. By partitioning data, queries can be executed more efficiently, as they can target specific partitions rather than scanning the entire dataset.

Explain Batch vs. Stream Processing

Batch processing and stream processing are two distinct methods for handling data workflows. Batch processing involves collecting data over a period and processing it in large chunks at scheduled intervals, which is efficient and simpler to manage but not suitable for real-time applications due to its higher latency. In contrast, stream processing handles data continuously in real-time as it arrives, making it ideal for applications like real-time monitoring and live analytics.

What are different Types of Data Models

When designing data warehouses, specific data modeling techniques are employed to optimize query performance and ensure data integrity. Among the most common are star schema, snowflake schema, and dimensional modeling. Each of these models is designed to support efficient querying and reporting.

1. Star Schema

A star schema is a type of database schema that is designed to optimize query performance in a data warehouse. It consists of a central fact table surrounded by dimension tables, forming a star-like structure.

2. Snowflake Schema

A snowflake schema is a more normalized form of the star schema, where dimension tables are further divided into related sub-dimension tables, resembling a snowflake shape.

3. Dimensional Modeling

Dimensional modeling is a design technique used for data warehouses that organizes data into dimensions and facts, facilitating easy querying and reporting. It encompasses both star and snowflake schemas.

Explain the concept of data lineage.

Data lineage refers to the tracking and visualization of data as it flows from its source to its destination. It helps in understanding the data’s origin, transformations, and journey through various processes, ensuring transparency, traceability, and data quality.

Star Schema vs Snowflake Schema - Difference Between Them

Key Difference Between Star Schema and Snowflake Schema - Comparing Snowflake vs Star schema, a Snowflake Schema is an…

www.guru99.com

```
#Google Cloud Specific Questions
```

How would you optimize BigQuery performance for large datasets?

Optimizing BigQuery performance involves:

- **Partitioning Tables:** Use date or range partitioning to limit the amount of data scanned.
- **Clustering Tables:** Organize data to improve query performance by reducing the amount of data scanned.
- **Query Optimization:** Use appropriate SQL functions and avoid using `SELECT *`.
- **Materialized Views:** Use materialized views to precompute and store query results.
- **Storage Optimization:** Use compressed and efficient data formats like Avro or Parquet.

- **Use Proper Data Types:** Choose the most appropriate and efficient data types for your columns.
- **Use JOINs and Subqueries Wisely:** Optimize JOINs and subqueries to avoid performance bottlenecks.

How do you secure data in GCP?

Securing data in GCP involves several practices:

- **Encryption:** Data is encrypted at rest and in transit.
- **IAM:** Use Identity and Access Management (IAM) to control access to resources.
- **VPC:** Set up Virtual Private Cloud (VPC) for network isolation and security.
- **Auditing:** Enable logging and auditing with Cloud Audit Logs.
- **DLP:** Use Cloud Data Loss Prevention (DLP) to detect and protect sensitive data.

How does BigQuery handle schema changes?

- BigQuery supports schema changes such as adding new columns and modifying column descriptions. You can add new columns without affecting existing data:
- Adding Columns: Use the `**ALTER TABLE**` statement.
- Deleting Columns: **Not directly supported**, but you can create a new table with the desired schema and copy the data over.

- Schema Auto-Detection: When loading new data, BigQuery can automatically detect and adjust the schema based on the incoming data.

How would you handle accidental data deletion in BigQuery?

Accidental data deletion can be a critical issue. In BigQuery, I would leverage the following features to mitigate this risk:

- **Time Travel:** This feature allows you to query and restore data to a specific point in time within a designated window.By enabling time travel, I can recover accidentally deleted data if the deletion occurred within the specified time frame.
- **Fail-Safe:** Although not directly accessible, BigQuery retains deleted data for an additional 7 days after the time travel window expires. In case of a catastrophic data loss, I can contact Google Cloud Customer Care to initiate a recovery process.

Apart from this we can setup data retenton policies, Backup strategies for regular backup/Snapshots.

Can you explain the concept of a streaming buffer in BigQuery? How does it impact data ingestion and query performance?

The **streaming buffer** in BigQuery is a temporary storage area that allows for real-time data ingestion. When data is streamed into a BigQuery table using the streaming insert API, it first goes into the streaming buffer.The data remains in the streaming buffer for a short period (usually up to 90 minutes) before it is moved to the permanent table storage. You might not not perform DELETE/UPDATE operation on streaming buffer data. (Now some additional feature to work on Streaming buffer data as well)

Life of a BigQuery streaming insert | Google Cloud Blog

Editor's Note (August, 2018): When this article was originally written, DML statements such as UPDATE and DELETE would…

cloud.google.com

What is the role of Google Cloud Data Catalog, and how does it integrate with other services?

Google Cloud Data Catalog is a fully managed metadata management service that helps you discover, manage, and understand data assets. It integrates with other services by:

- **Metadata Management:** Automatically cataloging metadata from BigQuery, Cloud Storage, and other data sources.
- **Search and Discovery:** Allowing users to search for data assets and understand their lineage.
- **Policy Management:** Providing data governance and access control features.

How to choose right data processing Tool among Dataflow vs Cloud Composer(Airflow) vs Dataproc vs data fusion in Google Cloud

Choosing correct tool depend on requirements as each of them have seperate pros & cons.

Google Cloud Dataflow is ideal for real-time and batch data processing, such as analyzing live user activity streams on an e-commerce site.

Cloud Composer (Airflow) is best for orchestrating workflows, like automating a daily ETL pipeline that extracts data from an API, transforms it, and loads it into BigQuery.

Dataproc excels at running large-scale data processing tasks with Hadoop or Spark, such as analyzing terabytes of log data.

Data Fusion is suited for building and managing ETL pipelines with a visual

Scenario-Based Questions

How do you design a data pipeline in Google Cloud giving some usecase?

Designing a data pipeline in Google Cloud typically involves:

- **Data Ingestion:** Using services like Cloud Pub/Sub or Dataflow for real-time or batch data ingestion. GCS also used for data ingestion or staging on cloud in form of files.
- **Data Storage:** Storing data in Cloud Storage (for raw data) or BigQuery (for structured data).
- **Data Processing:** Using Dataflow for ETL processes or Dataproc for big data processing.
- **Data Analysis:** Querying and analyzing data using BigQuery.
- **Data Visualization:** Creating dashboards and reports using Looker or Data Studio.

Describe a scenario where you had to troubleshoot a data pipeline issue. How did you approach the problem?

(Provide a specific example from your experience, explaining the issue, the troubleshooting steps you took, tools you used, and the resolution. Highlight your problem-solving skills and attention to detail.)

How would you design a data solution for a company with high-volume real-time data processing needs?

For high-volume real-time data processing:

- **Data Ingestion:** Use Cloud Pub/Sub for ingesting streaming data.
- **Data Processing:** Use Cloud Dataflow or Apache Beam to process and transform data in real-time.
- **Data Storage:** Store processed data in BigQuery for analytics or Cloud Storage for raw data.
- **Data Visualization:** Use Looker or Data Studio to create real-time dashboards and reports.

Data Engineer

Interview Question Answer

Written by Vishal Bulbule

Edit profile

462 Followers · 44 Following

Google Cloud Architect || Believe in Learn , work and share knowledge !
https://www.youtube.com/@techtrapture