Hevo Data

Data Engineering & Analytics | Online Assessment

**Ques 1**. What is data warehouse schema? Explain different types of schema.

**Ans 1**. To understand the word data warehouse schema, first of all we need to brief a little about what a data warehouse is?

**Data Warehouse** is a centralized data storage facility for data coming from multiple sources in huge quantities, which is further used for analyses and deriving insights for different operations.

Now as we are clear with the word data warehouse, let’s move to what a data warehouse schema really means?

From our traditional databases we all are aware of the term “**Schema**” i.e. a blueprint/structure/logical description of the entire database. As schema plays an important role in defining the properties/attributes of out data-items. So is it’s important for a data warehouse to have the same.

Data warehouse schema defines the names, descriptions, data-types of all the records in the data warehouse. But the main difference comes in terms of it’s architecture and other finer aspects. Let’s discuss them in detail.

1. **Star Schema**: This is one of the most used schemas in all. Here the structure of schemas of tables is more arranged like star topology of networks.

There is one schema table in the center, more like the head table from which other schemas are connected.

The main point to note down here is that, each dimension (vertex of star) is represented as 1-dimensional(1-D) table.

Certain other points to mention are: It is a top-down approach; Generally time taken for execution is less in this case; But due to individual tables for each dimension, there is high data redundancy.

2. **Snowflake Schema**: This is also an alternative to star schema. The main difference in star and snowflake comes in the normalization part.

Here data is normalized in the schema. It is multi-dimensional approach as opposed to single-dimensional star approach. It’s much complex due to multiple dependencies of schemas.

The main advantage comes that it offers much lower data redundancy.

But due to complex structure, takes much more time in execution of queries.

3. **Fact Constellation Schema**: This is another approach of schemas for warehouses.

Here schemas consist of fact tables as opposed to schematic description in case of star and snowflake schemas. It’s also known as galaxy schema.

**Ques 2**. What is the difference between OLTP and OLAP? Explain their application with the help of one example each.

**Ans 2**. OLTP stands for Online Transactional Processing.

OLAP stands for Online Analytical Processing.

Let’s discuss their difference in detail with appropriate example.

|  |  |
| --- | --- |
| OLTP | OLAP |
| Deals with online transaction and databases. | Major purpose for data retrieval and analysis. |
| Simple queries. | Complex Queries. |
| Normalization is of prior concern. | Normalization is not a concern. |
| Transaction are the source of data here. | Multiple OLTP generally fetch data to OLAP for analysis. |
| Simple queries -> less execution time. | Complex Queries -> More Execution time. |

These are the main differences b/w OLTP and OLAP. Apart from this there are several other parameters too which differ in both the cases.

Example: OLTP: When you go for any shopping and you pay with your debit or credit card, your account balance in updated as per the transaction made by you. Here simply a query is mentioned inside the database of your debit/credit card operating bank.

OLAP: When a sales manager queries data from multiple sources like, historical data of customers, live streaming data from market, production data of company, etc. for better analysis of overall sales and predicting for future references.

**Ques 3**. Explain Fact Table and Dimension Table.

**Ans 3**. Fact tables from its name highlights, it deals with all the facts and figures of the data.

Dimension Table from its name highlights it deals with dimension of the fact table.

Let’s see both in detail.

**Fact table**: As fact table shows all the facts and figures of dataset, it is located at the center of star schema or snowflake schema. They contain the labels from the reports. Generally, it does not contain hierarchy.

**Dimension Table**: As fact tales make the heart of star or snowflake schema, dimension tables form the edges for the same. As Fact tables contains all the data related to labels, fine details are stored in dimension table. It contains hierarchy.

**Ques 4**. Compare AWS Redshift and Google BigQuery cloud data warehouses. Which of these is best suited for what purpose?

**Ans 4**. AWS Redshift is a warehouse facility provided by Amazon while BigQuery is obvious from its name is a product of Google.

There are various factors on which both of them can be compared, let’s look some important one in details:

**Price**: Redshift charges on the basis of nodes/ computers used. Eg. $x for 2 nodes, $2x for cluster having 3 nodes and one head node, etc.

BigQuery cost model is based on your computation time. More you process, more is your cost. Eg. $x for 1GB queries processed.

**Performance**: When it comes to performance, it’s not an easy task to compare as both work differently for different data types, sources.

Redshift majorly works for streaming data due to which it is more efficient in performance wise for continuous data.

BigQuery on other hand handles on demand queries related to computation related to some specific analysis, operation, etc. So it works more efficiently for batch operation for specific purposes.

**Manageability**: Due to large array of features under both, it’s highly complex to compare the manageability index b/w both.

Redshift lays many features missing in BigQuery like transaction roll back, etc.

While BigQuery on other hand is much simpler to use and manage due to integrated storage and compute facility under one roof.

There are various other subtle aspects regarding security, end-user compatibility which adds value to both in terms of user demand and facility provided.

Redshift is best suited for scenarios that require continuous computation. Eg: Analyzing time-series related to sale of a certain product, price of a share, demand of a vaccine, etc.

BigQuery is best suited for scenarios with lots of queries being processed occasionally, with high idle time. Eg: Processing, analyzing data and feeding to ML model for discovering pattern, once in a day or week.

**Ques 5**. What are the benefits of using a data warehouse over a database?

**Ans 5**. Though both database and data warehouse have slightly different needs, uses, requirements but to consider, let’s see how a data warehouse is beneficial over database.

• Data warehouse provides data input of different kinds from multiple sources, which is not available in databases.

• Data warehouse is made to handle large amounts of data efficiently.

• Using data warehouse not only provides storage but also gives an option of analyzing data which is missing in traditional databases.

• It is easier to scale up and down in warehouse as compared to databases

• Modern data warehouse like, BigQuery provides high level of security.

• The main advantage is that warehouse is an integrated system consisting of ETL (Extract, Transform, Load) due to which it is very efficient to analyze data and drive revenue for any organization.

In short, data warehouse improves business decision making, which in turn gives any business a key competitive advantage.

**Ques 6**. Explain the components of a Data Analytics Stack. What is its significance?

**Ans 6**. Components of Data Analytics Stack are:

Data Sources -> Data Storage -> Insights

The whole Stack revolves around the above mentioned flow chart.

• Data Pipeline: The crucial part of any data analytics stack, data pipeline. This is where the main operations i.e. ETL (Extract, Transform, Load) takes place.

Let’s discuss them in brief:

Extract: Fetching data from multiple sources like, historical data from database, live streaming data, application data, IoT device data, etc. to feed into the storage house.

Transform: Data wrangling, data transformation, data segmentation, etc. takes place in this part. It includes sorting, filtering and formatting data for future use.

Load: This is related to physical transportation of data into a warehouse.

• Data Warehouse: The main house for all the operations like storing data, gaining insights, etc, takes place here only. There are various service providers in the market which provide on-click, on-demand warehouse facilities.

• Data Visualization: Gaining insights, producing results for iterations, handling production rate, etc. can only be made possible after in depth visual analyses of data.

Apart from these components, one more important aspect of any data analytics stack is Team. This is the crucial part while building any data driven team because the above mentioned components can only produce results inly if the team holds the potential.

Ques 7. What is the difference between SQL and NoSQL databases? Explain with examples.

Ans 7. Difference between SQL and NoSQL

|  |  |
| --- | --- |
| SQL | NoSQL |
| It is relational in nature. | It is non-relational in nature. |
| Better for multi-row transactions i.e. tabular data. | Better for unstructured data like document of JSON. |
| Vertical scalability. | Horizontal scalability. |
| Use predefined schemas. | Use dynamic schemas. |
| Suited for complex queries. | Not so good for complex queries. |
| Follow ACID properties. | Follow CAP. |

These were some of the differences between SQL and NoSQL databases.

Their example can include:

SQL databases: MySQL, Oracle, Postgre, etc.

NoSQL databases: MongoDB, BigTable, Redis, etc.