Tutorial 5
Column-Oriented Databases

SOLUTIONS

Discuss the following questions:

1. What is Column-Oriented DBMS?

   Column-Oriented DBMS is a DBMS that stores data tables as sections of columns of data, rather than as rows of data as in most DBMS implementations.

2. What are the three components of data values in Column-Oriented Databases?

   Data values are indexed by row identifier, column name, and time stamp.

3. Why are time stamps used in Column-Oriented Databases?

   The time stamp orders versions of the column value and it allows applications to determine the latest version of a column value.

4. Identify one similarity between column-oriented databases and document databases.

   Column-oriented and document databases support similar types of querying that allow users to select subsets of data available in a row.
   Column-oriented databases, like document databases, do not require all columns in all rows. In both column-oriented and document databases, columns or fields can be added as needed by developers.

5. Identify one similarity between column-oriented databases and relational databases.

   Both column-oriented databases and relational databases use unique identifiers for rows of data. These are known as row keys in column-oriented databases and as primary keys in relational databases. Both row keys and primary keys are indexed for rapid retrieval.

6. Describe the essential characteristics of a peer-to-peer architecture.

   Peer-to-peer architectures have only one type of node. Any node can assume responsibility for any service or task that must be run in the cluster.

7. Why does Cassandra use a gossip protocol to exchange server status information?

   An “all-servers-to-all-other-servers” protocol can quickly increase the volume of traffic on the network and the amount of time each server has to dedicate to communicating with other servers. The number of messages sent is a function of the number of servers in the cluster. If N is the number of servers, then N × (N–1) is the number of messages needed to

update all servers with information about all other servers. Gossip protocols are more efficient because one server can update another server about itself as well as all the servers it knows about.

8. When would you use a column-oriented database instead of another type of NoSQL database?

   Column-oriented databases are appropriate choices for large-scale database deployments that require high levels of write performance, a large number of servers, or multi–data center availability.
   Column-oriented databases are also appropriate when a large number of servers are required to meet expected workloads.

9. How do columns in column-oriented databases differ from columns in relational databases?

   Columns in column-oriented are dynamic. Columns in a relational database table are not as dynamic as in column-oriented databases. Adding a column in a relational database requires changing its schema definition. Adding a column in a column-oriented database just requires making a reference to it from a client application, for example, inserting a value to a column name.

10. When should columns be grouped together in a column family? When should they be in separate column families?

    Columns that are frequently used together should be grouped in the same column family.

The End