Assignment 9.1

Problem Statement 1 :-

What is NoSQL data base?

Solution:-

The "NoSQL" moniker refers to a class of data stores that are non-relational in nature.NoRDBMS might be the more accurate term - but "NoSQL" was catchier as traditional relational databases (SQL Server, Oracle, MySql, Postgres) share in common a means of querying their data using a structured query language (SQL) to pull relational information together in a unified result.

There are a variety of NoSQL data stores available, the most popular ones currently include Cassandra, MongoDB, Redis, Membase, and CouchDb among others.

Each of these persist data slightly differently - some are key-value pair data stores (Redis, Membase), others are columnar data stores (Cassandra), and others are document data stores (MongoDB).

Problem Statement 2 :-

How does data get stored in NoSQl database?

Solution:-

There are various NoSQL Databases. Each one uses a different method to store data. Some might use column store, some document, some graph etc.., Each database has its own unique characteristics.

Imagine you have an ordering system, storing customer details and product information in a MSSQL database and a NoSQL Event-Storing solution. Since there is no schema limitations you can easily archive your events including all

the relevant data as their properties and have them serialized for you automatically.

Let's assume we are storing an "Order Processed" event. Now, each time you create/store a new instance of this event, all the product and customer data will be serialized "as is" at the current point of time in your SQL database and stored right into your NoSQL database along with other events. A clear benefit from this would be the fact that if a customer decides to change his/her invoicing address or any other data for that matter, this event's details would not change as it holds serialized data with no relation to the outside world or other entities. This fact makes it perfect for reporting purposes since the data is always accurate to single point in time when it was created!

Problem Statement 3 :-

What is a column family in HBase?

Solution:-

Columns in HBase are grouped into column families. All column members of a column family have the same prefix. For example, the columns "courses:history" and "courses:math" are both members of the "courses" column family.

The colon character (:) delimits the column family from the. The column family prefix must be composed of printable characters.

Column families must be declared up front at schema definition time

Physically, all column family members are stored together on the File System. Because tunings and storage specifications are done at the column family level, it is advised that all column family members have the same general access pattern and size characteristics.

Problem Statement 4 :-

How many maximum number of columns can be added to HBase table?

Solution:-

Generally, column families remain fixed throughout the lifetime of an HBase table but new column families can be added by using administrative commands. The official recommendation for the number of column families per table is three or less.

Problem Statement 5 :-

Why columns are not defined at the time of table creation in HBase?

Solution:-

Column families are part of the schema of the table. We can add them at runtime with an online schema change. But we wouldn't add them dynamically the way that we can dynamically create new "columns" in an HBase table.

The reason column families are part of the schema and would require a schema change is that they profoundly impact the way the data is stored, both on disk and in memory. Each column family has its own set of HFiles, and its own set of data structures in memory of the RegionServer. It would be pretty expensive to dynamically create or start using new column families.

Column families are only needed when we need to configure differently various parts of a table (for instance you want some columns to have a TTL and others to not expire), or when we want to control the locality of accesses (things accessed together should better be in the same column family if we want good performance, as the cost of operations grows linearly with the number of column families). So, again, because of those specialized reasons, it doesn't make sense to dynamically add new column families at runtime the way we would add regular "columns" within a family.

Problem Statement 6 :-

How does data get managed in HBase?

Solution:-

HBase is a column-oriented database that's an open-source implementation of Google's Big Table storage architecture. It can manage structured and semi-structured data and has some built-in features such as scalability, versioning, compression and garbage collection.

Since its uses write-ahead logging and distributed configuration, it can provide fault-tolerance and quick recovery from individual server failures. HBase built on top of Hadoop / HDFS and the data stored in HBase can be manipulated using Hadoop's MapReduce capabilities.

Let's now take a look at how HBase (a column-oriented database) is different from some other data structures and concepts that we are familiar with Row-Oriented vs. Column-Oriented data stores. As shown below, in a row-oriented data store, a row is a unit of data that is read or written together. In a column-oriented data store, the data in a column is stored together and hence quickly retrieved.

Row ID	Customer	Product	Amount
101	John White	Chairs	$400.00
102	Jane Brown	Lamps	$500.00
103	Bill Green	Lamps	$150.00
104	Jack Black	Desk	$700.00
105	Jane Brown	Desk	$650.00
106	Bill Green	Desk	$900.00

Problem Statement 7 :-

What happens internally when new data gets inserted into HBase table?

Solution:-

When new data entered using the put command ,It stored initially into a MemStore ,Actually anything that is entered into the HBase is stored here initially. Later, the data is transferred and saved in Hfiles as blocks and the memstore is flushed.

Submitted By:-

Hardik kaushik