

Bigdata Assignment 3.9

Explain the below concepts with an example in brief.

- **Nosql Databases** - NOSQL stands for Not Only SQL Database. They are nonrelational databases optimized for scalable performance and schemaless data models. It provides a DBMS system that does not use the conventional tabular relations used in RDBMS instead they use a variety of data models, including columnar, document, graph, and in-memory key-value stores. These databases are a great fit for many big data, mobile, and web applications that require greater scale and higher responsiveness than traditional relational databases. Due to simpler data structures and horizontal scaling, NoSQL databases typically respond faster and are easier to scale than relational databases.
- **Types of Nosql Databases** -
There are of 4 types :-
 - a) **Key-value store NoSQL database** - Key-value stores are the simplest NoSQL data stores to use. The client can either get the value for the key, assign a value for a key or delete a key from the data store. The key-value database uses a hash table to store unique keys and pointers (in some databases it's also called the inverted index) with respect to each data value it stores. Key-value databases give great performance and can be very easily scaled as per business needs. Examples are MUMPS , Oracle No SQL , Zookeeper, etc.
 - b) **Document store NoSQL database** – It is similar to key-value databases in that there's a key and a value. Data is stored as a value. Its associated key is the unique identifier for that value. The difference is that, in a document database, the value contains structured or semi-structured data. This structured/semi-structured value is referred to as a document and can be in XML, JSON or BSON format. Examples are RangoDB , MongoDB , etc

c) **Column store NoSQL database** - Here, data is stored in cells grouped in columns of data rather than as rows of data. Columns are logically grouped into column families. Column families can contain a virtually unlimited number of columns that can be created at runtime or while defining the schema. Read and write is done using columns rather than rows. Examples are Cassandra , Hbase , etc.

d) **Graph base NoSQL database** – These databases are basically built upon the Entity – Attribute – Value model. Entities are also known as nodes, which have properties. It is a very flexible way to describe how data relates to other data. Nodes store data about each entity in the database, relationships describe a relationship between nodes, and a property is simply the node on the opposite end of the relationship. Examples are Apache Giraph , OrientDB , etc.

- **CAP Theorem** -

CAP theorem is basically a map of different database solutions depicting its CAP properties that show us the tradeoff of choosing a particular database system.

CAP stands for Consistency, Availability and Partition. As per CAP theorem we cannot have a database system that is Consistent, has high availability and partition tolerant all at once. There has to be a tradeoff. It is denoted by a triangle.

Availability - A request made to the data store shall always eventually complete, no matter what pattern of failures have occurred.

Consistency - The results of 'earlier' writes are consistent and are always seen by 'later' reads.

Partition tolerance – In case of network issues, it might face refusal of the network to deliver any subset of the messages sent between nodes. Hence it should be tolerant.

- **HBase Architecture -**

HBase architecture comprises of Regions and Stores.

Master Server : It assigns regions to region servers with the help of zookeeper. It also facilitates load balancing by distributing loads among different region servers. It is responsible for schema and metastore operations.

Region Server : Handles the data related operations, takes requests from clients. It is responsible only for the regions under itself. This is the workhorse of HBase. Region Server runs on HDFS DataNode and consists of the following components :-

1. **Block Cache** – This is the read cache. Most frequently read data is stored in the read cache and whenever the block cache is full, recently used data is evicted.
2. **MemStore**- This is the write cache and stores new data that is not yet written to the disk. Every column family in a region has a MemStore.
3. **Write Ahead Log (WAL)** is a file that stores new data that is not persisted to permanent storage.
4. **HFile** is the actual storage file that stores the rows as sorted key values on a disk.

Zookeeper : This is a helper solution to the Master Server that helps in the synchronizations, maintaining configurations etc. It tracks server failures and provides ephemeral nodes, which represent different region servers.

- **HBase vs RDBMS**

HBASE	RDBMS
<ul style="list-style-type: none">• Flexible Schema	<ul style="list-style-type: none">• Fixed Schema
<ul style="list-style-type: none">• Column oriented Schema	<ul style="list-style-type: none">• Row oriented Schema
<ul style="list-style-type: none">• Uses Java API client and Jruby.	<ul style="list-style-type: none">• Uses SQL

<ul style="list-style-type: none">• De-normalized data	<ul style="list-style-type: none">• Normalized data
<ul style="list-style-type: none">• Good for semi-structured data as well as unstructured data.	<ul style="list-style-type: none">• Good for structured data