**ASSIGNMENT-9.3**

Explain the below concepts with an example in brief.

1. **Nosql Databases**

A NoSQL database is exactly the type of database that can handle the sort of unstructured, messy and unpredictable data that our system of engagement requires.

NoSQL is an approach to database design that can accommodate a wide variety of data models, including key-value, document, columnar and graph formats. NoSQL, which stand for "not only SQL," is an alternative to traditional relational databases in which data is placed in tables and data schema is carefully designed before the database is built. NoSQL databases are especially useful for working with large sets of distributed data.

Some examples of NoSQL databases are MongoDb, Accumulo, Cassandra.

1. **Types of NOSQL database**
2. Key-Value Store – It has a Big Hash Table of keys & values {Example- Riak, Amazon S3 (Dynamo)}
3. Document-based Store- It stores documents made up of tagged elements. {Example- CouchDB}
4. Column-based Store- Each storage block contains data from only one column, {Example- HBase, Cassandra}
5. Graph-based-A network database that uses edges and nodes to represent and store data. {Example- Neo4J}
6. Key Value Store NoSQL Database

Key-value stores, or key-value databases, implement a simple data model that pairs a unique key with an associated value. Because this model is simple, it can lead to the development of key-value databases, which are extremely performant and highly scalable for session management and caching in web applications. Implementations differ in the way they are oriented to work with RAM, solid-state drives or disk drives. Examples include Aerospike, Berkeley DB, MemchacheDB, Redis and Riak.

1. Document Store NoSQL Database

Document databases, also called document stores, store semi-structured data and descriptions of that data in document format. They allow developers to create and update programs without needing to reference master schema. Use of document databases has increased along with use of JavaScript and the JavaScript Object Notation (JSON), a data interchange format that has gained wide currency among web application developers, although XML and other data formats can be used as well. Document databases are used for content management and mobile application data handling. Couchbase Server, CouchDB, DocumentDB, MarkLogic and MongoDB are examples of document databases.

1. Column Store NoSQL Database

In column-oriented NoSQL database, 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 the definition of the schema. Read and write is done using columns rather than rows.

In comparison, most relational DBMS store data in rows, the benefit of storing data in columns, is fast search/ access and data aggregation. Relational databases store a single row as a continuous disk entry. Different rows are stored in different places on disk while Columnar databases store all the cells corresponding to a column as a continuous disk entry thus makes the search/access faster.

For example: To query the titles from a bunch of a million articles will be a painstaking task while using relational databases as it will go over each location to get item titles. On the other hand, with just one disk access, title of all the items can be obtained.

**Data Model**

ColumnFamily: ColumnFamily is a single structure that can group Columns and SuperColumns with ease.

Key: the permanent name of the record. Keys have different numbers of columns, so the database can scale in an irregular way.

Keyspace: This defines the outermost level of an organization, typically the name of the application. For example, ‘3PillarDataBase’ (database name).

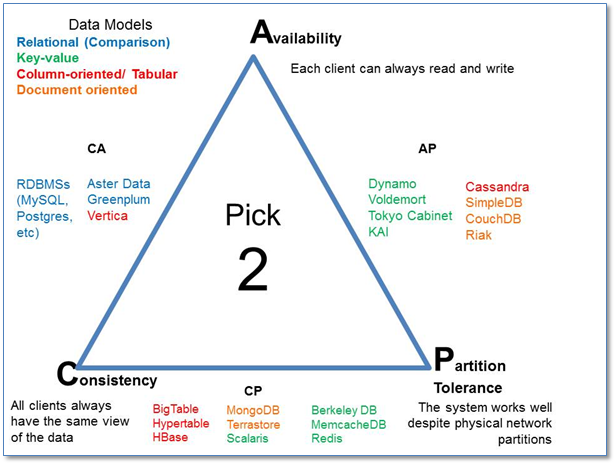
Column: It has an ordered list of elements aka tuple with a name and a value defined.

1. Graph Base NoSQL Database

Graph data stores organize data as nodes, which are like records in a relational database, and edges, which represent connections between nodes. Because the graph system stores the relationship between nodes, it can support richer representations of data relationships. Also, unlike relational models reliant on strict schemas, the graph data model can evolve over time and use. Graph databases are applied in systems that must map relationships, such as reservation systems or customer relationship management. Examples of graph databases include AllegroGraph, IBM Graph, Neo4j and Titan.

1. CAP Theorem

CAP theorem is designed for distributed file systems(collection of interconnected nodes).CAP Theorem also known as Brewer’s theorem and used to distributed consistency. it contains follwing three technical terms for distributed systems.



C – Consistency

A – Availability

P – Partition Tolerance

Consistency:

When you read data it will give same data how many times read and server send response each and every request but systems always consistent when read data.(all node having same data)

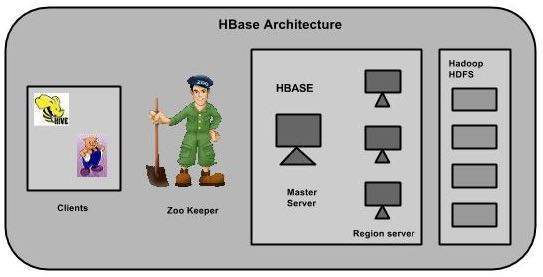
Availability:

It means all requests give response and no error accured in this systems.

Partition Tolerance:

All functions run all time when more nodes not responsive and commnication break between two nodes

1. HBASE Architecture



HBase has three major components: the client library, a master server, and region servers. Region servers can be added or removed as per requirement.

MasterServer

The master server -

* Assigns regions to the region servers and takes the help of Apache ZooKeeper for this task.
* Handles load balancing of the regions across region servers. It unloads the busy servers and shifts the regions to less occupied servers.
* Maintains the state of the cluster by negotiating the load balancing.
* Is responsible for schema changes and other metadata operations such as creation of tables and column families.

Regions

Regions are nothing but tables that are split up and spread across the region servers.

Region server

The region servers have regions that -

* Communicate with the client and handle data-related operations.
* Handle read and write requests for all the regions under it.
* Decide the size of the region by following the region size thresholds.

When we take a deeper look into the region server, it contain regions and stores as shown below:

The store contains memory store and HFiles. Memstore is just like a cache memory. 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.

Zookeeper

* Zookeeper is an open-source project that provides services like maintaining configuration information, naming, providing distributed synchronization, etc.
* Zookeeper has ephemeral nodes representing different region servers. Master servers use these nodes to discover available servers.
* In addition to availability, the nodes are also used to track server failures or network partitions.
* Clients communicate with region servers via zookeeper.
* In pseudo and standalone modes, HBase itself will take care of zookeeper.

1. HBASE vs RDBMS

|  |  |
| --- | --- |
| ***RDBMS*** | ***HBase*** |
| Row-oriented databases | Distributed, column-oriented data storage system |
| fixed-schema | Flexible Schema,add columns on the fly |
| Not optimised for sparse tables | Good with sparse tables |
| Uses SQL (Structuredquery Langauge ) to query the data. | Uses Java client API and Jruby |
| Hard to shard and scale | Horizontal scalability just add hardware |
| Optimized for joins | Joins using MR |