SESSION 9 ASSIGNMENT 3

**Problem. Explain the below concepts with an example in brief.**

Nosql Databases

A NoSQL database provides a mechanism for storage and retrieval of data that is modeled in means other than the tabular relations used in relational databases.The advantages of this type of DB are simplicity of design, simpler "horizontal" scaling to clusters of machines (which is a problem for relational databases),and finer control over availability.

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

Types of Nosql Databases

There have been various approaches to classify NoSQL databases, each with different categories and subcategories, some of which overlap. What follows is a basic classification by data model, with examples:

* + - **Column**: Accumulo, Cassandra, Druid, HBase, Vertica.
    - **Document**: The central concept of a document store is the notion of a "document". One of the other defining characteristics of a document-oriented database is that in addition to the key lookup performed by a key-value store, the database offers an API or query language that retrieves documents based on their contents.

For eg.Apache CouchDB, ArangoDB, BaseX,Cosmos DB, HyperDex, IBM Domino, MarkLogic, MongoDB

* + - **Key**-**value**: Key-value (KV) stores use the associative array (also known as a map or dictionary) as their fundamental data model. In this model, data is represented as a collection of key-value pairs, such that each possible key appears at most once in the collection.

For eg. Aerospike, Apache Ignite, ArangoDB, Couchbase, Dynamo, FairCom c-treeACE, FoundationDB, HyperDex, InfinityDB, MemcacheDB

* + - **Graph**: This kind of database is designed for data whose relations are well represented as a graph consisting of elements interconnected with a finite number of relations between them.

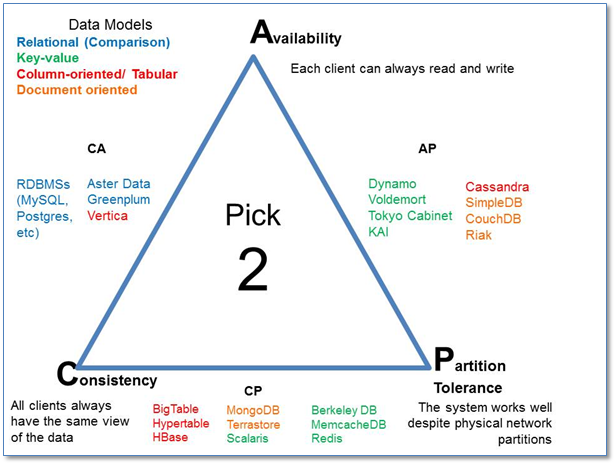
For eg. AllegroGraph, ArangoDB, InfiniteGraph, Apache Giraph, MarkLogic, Neo4J, OrientDB, Virtuoso.

* + - **Multi-model**: Apache Ignite, ArangoDB, Couchbase, FoundationDB, InfinityDB, MarkLogic, OrientDB

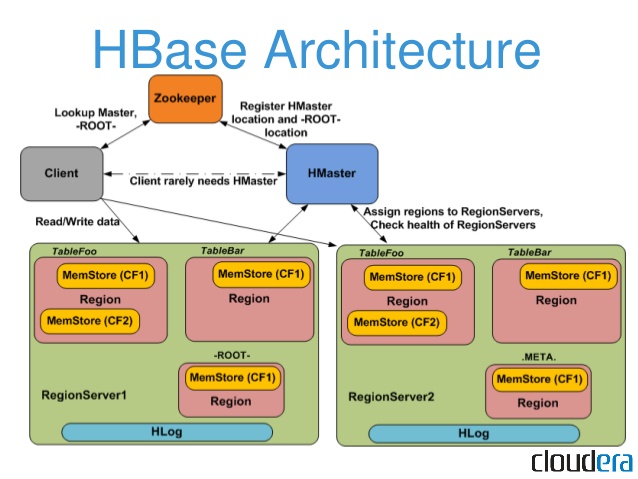
CAP Theorem

The CAP theorem states that it is impossible for a distributed data store to simultaneously provide more than two out of the following three guarantees:

|  |  |  |
| --- | --- | --- |
| ***Consistency*** | [***Availability***](https://en.wikipedia.org/wiki/Availability) | [***Partition tolerance***](https://en.wikipedia.org/wiki/Network_partitioning) |
| The data in the database remains consistent after the execution of an operation.  For example after an update operation, all clients see the  same data. | This means that the system is always on (service guarantee availability),no downtime. | This means that the system continues to function even if the communication among the servers is unreliable, i.e. the servers may be partitioned into multiple groups that cannot communicate with one another. |



HBase Architecture



Components of Apache HBase Architecture

**HBase architecture has 3 important components- HMaster, Region Server and ZooKeeper.**

* **HMaster**

HBase HMaster is a lightweight process that assigns regions to region servers in the Hadoop cluster for load balancing. Responsibilities of HMaster –

* + - Manages and Monitors the Hadoop Cluster
    - Performs Administration (Interface for creating, updating and deleting tables.)
    - Controlling the failover
    - DDL operations are handled by the HMaster
    - Whenever a client wants to change the schema and change any of the metadata operations, HMaster is responsible for all these operations.
* **Region** **Server**

These are the worker nodes which handle read, write, update, and delete requests from clients. Region Server process, runs on every node in the hadoop cluster. Region Server runs on HDFS DataNode and consists of the following components –

* + - **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.
    - **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.
    - **Write** **Ahead** **Log** (**WAL**) is a file that stores new data that is not persisted to permanent storage.
    - **HFile** is the actual storage file that stores the rows as sorted key values on a disk.
* **Zookeeper**

HBase uses ZooKeeper as a distributed coordination service for region assignments and to recover any region server crashes by loading them onto other region servers that are functioning. ZooKeeper is a centralized monitoring server that maintains configuration information and provides distributed synchronization. Whenever a client wants to communicate with regions, they have to approach Zookeeper first. HMaster and Region servers are registered with ZooKeeper service, client needs to access ZooKeeper quorum in order to connect with region servers and HMaster. In case of node failure within an HBase cluster, ZKquoram will trigger error messages and start repairing failed nodes.

ZooKeeper service keeps track of all the region servers that are there in an HBase cluster- tracking information about how many region servers are there and which region servers are holding which DataNode. HMaster contacts ZooKeeper to get the details of region servers. Various services that Zookeeper provides include –

* + - Establishing client communication with region servers.
    - Tracking server failure and network partitions.
    - Maintain Configuration Information
    - Provides ephemeral nodes, which represent different region servers.

HBase vs RDBMS

|  |  |
| --- | --- |
| ***RDBMS*** | ***HBase*** |
| Row-oriented databases | Distributed, column-oriented data storage system |
| Have fixed-schema | Do not have fixed-schema |
| Tables guarantee acid properties | Guarantee consistency and partition tolerance |
| Uses SQL (Structuredquery Langauge ) to query the data. | Uses Java client API and Jruby |
| Good for structured data. | Good for semi-structured data as well as structured data. |