**Session 31**

**Assignment 1**

**Problem Statement:**

**Part 1:**

|  |  |
| --- | --- |
| HDFS | HBASE |
| * HDFS is a Distributed file system abstracted on top of local file system by Hadoop, suitable for storing huge files, it does not provide facility of tabular form of storage as such. * HDFS does not support fast individual record lookups. * Large data files are stored in a distributed manner on cheap commodity hardware in a secure and cost-effective way. * The HDFS files are write once and read multiple times. HDFS does not support the option of random write or read. * In HDFS data is stored reliably. Files are broken into blocks and distributed across nodes in a cluster. * It is highly fault tolerant. * HDFS provides faster file read and write mechanism, as data is stored in different nodes in a cluster. But does not support random read or write. | * HBASE is a Column oriented distributed (on top of Hadoop) data store which runs on top of HDFS for providing structural data models and stores data in table row column. * It provides faster data lookup across the tables. * It gives ability to do random read/writes on the given data. * HBase is specifically designed for working with sparse data sets and provides low latency access to a single row of data among billions of data * Data is indexed by the row key and it has a very flexible data model and data is stored in a hashed table and access is available in a random manner * An HBase system comprises a set of tables. Each table must have an element defined as a Primary Key, and all access attempts to HBase tables must use this Primary Key. |

Differences between HBASE and HDFS.

**Part 2:**

List and explain the main components of HBASE.

HBase is composed of three types of servers in a master slave model architecture.

* **Region servers** serve data for reads and writes.
* **HBase Master** server handles the Region assignment, DDL (create, delete tables) operations
* **Zookeeper** maintains a live cluster state.

**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.

The store contains memory store and HFiles. Memory store is just like a cache memory where everything is stored when anything is entered into the HBase initially. Later, that data is transferred and saved in HFiles as blocks and the memory store is flushed.

**Master Server:**

* Assigns regions to the region servers and takes the help of Apache Zoo Keeper 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.

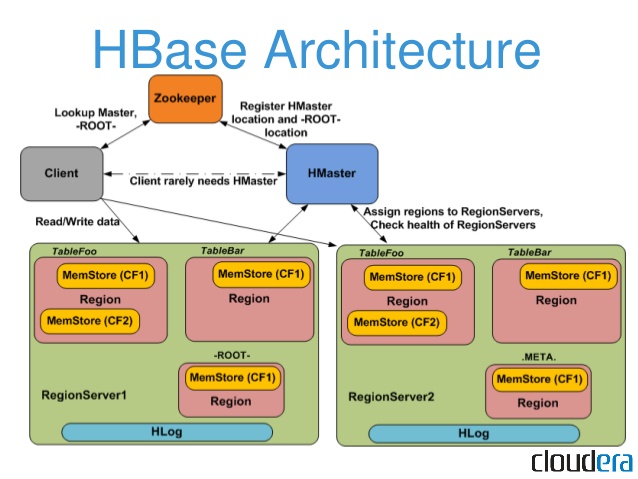
## Zookeeper:

## Various services that Zookeeper provides include

* 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.
* Client communication with region servers is established via zookeeper.
* In pseudo and standalone modes, HBase itself will take care of zookeeper.
* HBase uses Zoo Keeper as a distributed coordination service to maintain server state in the cluster.
* Zookeeper maintains which servers are alive and available, and provides server failure notification.
* Zookeeper uses consensus to guarantee common shared state. Note that there should be three or five machines for consensus.

**HBase Architecture Diagram**

This diagram is taken from the Cloudera site where the theory is been understood and explained above.



**Part 3:**

Does HBase support SQL?

HBase does not support a structured query language like SQL because HBase isn't a relational data store at all.

HBase non-relational (NoSQL) database that runs on top of HDFS.

HBaseapplications are written in Java much like a typical Map Reduce application.

HBase is much like a traditional database.