**Assignments-13.3**

Explain in Brief.

1. HFile

HBase uses HDFS for saving the data on the region servers. HBase writes the data using HFile onto the HDFS.

Data stored in HFile are in the format of key/value pairs.It is a file format for hbase.

When the MemStore accumulates enough data, the entire sorted KeyValue set is written to a new HFile in HDFS.

Features of HFile:

* Row key is primary identifier.
* HFiles store the rows as sorted KeyValues on disk.
* HFile is Unit of Storage used by HBase
* HFile is data file HBase which is stored on HDFS

1. HRegion Server

In HBase, tables are split into regions and are served by the region servers.

Regions are vertically divided by column families into “Stores”. Stores are saved as files in HDFS.

Region servers can be added or removed as per requirement.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.

Region Server runs on an HDFS data node and has the following components -

* WA
* BlockCache
* MemStore
* Hfiles

1. HBase Meta Table

The catalog tables -ROOT- and .META. exist as HBase tables. They are filtered out

of the HBase shell's list command, but they are in fact tables just like any other.

The .META. table keeps a list of all regions in the system. The .META. table structure is as follows:

* Key:
* Region key of the format ([table],[region start key],[region id])
* Values:
* info:regioninfo (serialized [HRegionInfo](http://hbase.apache.org/apidocs/org/apache/hadoop/hbase/HRegionInfo.html" \t "http://hbase.apache.org/0.94/book/_top) instance for this region)
* info:server (server:port of the RegionServer containing this region)
* info:serverstartcode (start-time of the RegionServer process containing this region)
* When a table is in the process of splitting two other columns will be created, info:splitA and info:splitB which represent the two daughter regions.
* The values for these columns are also serialized HRegionInfo instances. After the region has been split eventually this row will be deleted.
* The empty key is used to denote table start and table end. A region with an empty start key is the first region in a table. If region has both an empty start and an empty end key, it's the only region in the table

1. 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.

5. How read and write operation is performed in HBase.

**Read Operation:**

* A read request is sent to zookeeper from Hbase by the client.
* The zookeeper,gives the address for the META table.
* The process gets the region address of table from the META Table.
* From the specific HRegion, the process enters the BlockCache where data is present from previous read.
* If the table is found, the process returns to client with the data as result.
* If the table is not found, the process starts to search MemStore.
* If it is found, the process returns to client with the data as result.
* If the table is not found, the process moves forward in search of data within the HFile.
* The process takes required data from here and moves forward.
* The data is written in BlockCache, so that the next time, it can be instantly accessed by the client
* Finally the read process with required data will be returned to client along with acknowledgment.

**Write Operation:**

* When the client issues a Put request, the first step is to write the data to the write-ahead log, the WAL:
* Edits are appended to the end of the WAL file that is stored on disk.
* The WAL is used to recover not-yet-persisted data in case a server crashes.
* Once the data is written to the WAL, it is placed in the MemStore.
* Then, the put request acknowledgement returns to the client.