**ASSIGNMENT 13.3**

|  |
| --- |
| 1. **HFile** |
| File format for hbase. A file of sorted key/value pairs.  Both keys and values are byte arrays.  HFile index, which is proportional to the total number of Data Blocks. The total amount of memory needed to hold the index can be estimated as (56+AvgKeySize)\*NumBlocks.  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**   HRegionServer makes a set of HRegions available to clients.  It checks in with the HMaster. There are many HRegionServers in a single HBase deployment.  Regions are assigned to the region servers.  It is about 1000 regions that may belong to the same or different tables.  Region servers runs on an HDFS data node.  COMPONENTS :  Wal  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.  The .META. table keeps a list of all regions in the system.  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 "_top) instance for 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.  A region with an empty start key is the first region in a table.   1. **Zookeeper**   Apache ZooKeeper is an effort to develop and maintain an open-source server which enables highly reliable distributed coordination.  ZooKeeper is a centralized service for maintaining configuration information, naming, providing distributed synchronization, and providing group services.  Because of the difficulty of implementing these kinds of services, applications initially usually skimp on them ,which make them brittle in the presence of change and difficult to manage.   |  | | --- | |  | |  |  | |  |  | |  |  | |  |  | |  |  | |  |  |  1. **How read and write operation is performed in HBase.**   **Read operation :**   * Zookeeper has location for META table which is present in HRegion Server. * The process continues to HRegionServer and gets to META table. * Moving forward to a specific HRegion, the process enters the BlockCache where data is present from previous read. * If the table is not found, the process starts to search MemStore since data would have been written to HFile sometime back. If it is found, the process returns to client with the data as result. * The data taken from HFile is the latest read data and can be read by user again. Hence the data is written in BlockCache * When the data is written in BlockCache and all the search is completed, the read process with required data will be returned to client along with ACK(Acknowledgment)   **Write operation :**   * Instruction is directed to Write Ahead Log and first writes important logs to it. * Once the log entry is done, the data to be written is forwarded to MemStore which is actually the RAM of the data node. * All the data is written in MemStore which is faster than RDBMS * Later, the data is dumped in HFile, where the actual data is stored in HDFS. * Once writing data is completed, ACK (Acknowledgement) is sent to client as a confirmation of task completed. |
|  |
|  |
|  |