1. What are the advantages of using HBase over HDFS?

From my opinion, HDFS is distributed file system that handle file but HBase in other hand, is a distributed columnar NoSQL database that build on top of HDFS, enjoying all HDFS capability.

HDFS is write once read many designs but hbase support record level manipulation. Hbase rely on region and region server to have the ability for random read and write into HDFS file server. HDFS in other hand reading and writing entire file even if portion of data is needed.

HBase perform better in handling small data, like row:cell. HDFS is meant to store data in file format (default 128mb). It is unwise to store file/data smaller than default size because it will make the name node congested with metadata (https://blog.cloudera.com/the-small-files-problem/).

HDFS is rely on map reduce which is batch processing. Every operation in map reduces perform scanning through the whole file. Hbase has rowkey that acts like partition for the data. This allow hbase to lookup the data instead of scanning through the whole file, thus more real-time processing capability.

Hbase has version number in every cell to identify record timestamp. This to tell any modification to the data. HDFS overwrite everything in the file.

2. Discuss the data model used in HBase.

- table – A table is defined as the main structure of data, table consist of column family only, not column and data type like string, int, timestamp. Collection of rows.
- row – A row is single record, controlled by rowKey. A row is partitioned horizontally by collection of column families.
- column – A column is collection of key value pair. Every record for row is place accordingly by column. A column can be sparse.
- column family – collection of columns.
- cell – lowest granularities support by hbase. Tuple form if (Key, Value, version number)
- region – subset rows of column family in table. To store data into hdfs, a table with one of the column family are stored and partition by group of row. The data is called region to be assigned by region family to be stored into Data Node of HDFS.

3. List out the main function for each of the key components in HBase architecture.

- region – Rows of record in table column family divided into smaller chunk of data to be stored into HDFS Data Node.
- hmaster – only one instance, coordinate region server, assign region go to which region server. Contain table metadata.
- region server – many instance, manage region read and write
- zookeeper – coordinate between hmaster, region server and client. The communication channel for hmaster, region server and client.
- client – access data in hbase through zookeeper.

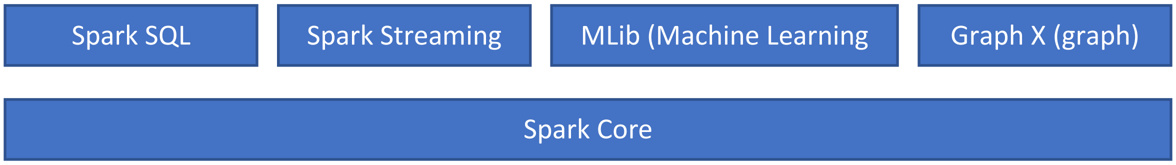
4. Discuss the differences between Spark and Hadoop MapReduce.

Hadoop Map Reduce is a batch processing engine. Spark does both real-time processing and batch processing.

Hadoop map reduce load, process and pass the data in disk but spark perform processing and communication in ram. However, spark also offer the processing in disk as well.

Hadoop map reduce can only be written in java, which hard for data scientist. Spark in support multiple API and can be written in Java, Scala, Python and R.

5. Draw a figure to show the five components in Spark ecosystem. Briefly explain the function(s) for each of them.



Spark SQL | Spark Streaming | MLib (Machine Learning | Graph X (graph)

Spark Core

Spark core – control everything, manage each node resources and can process in node ram instead of disk.
Spark SQL – sql interface to query spark
Spark Streaming – data ingestion (read/write data), counterpart for flume. Data is constantly steam into RDD and being process
MLib – inbuilt machine learning library. Similar like sklearn in python, Spark MLib contain all the library needed for performing Machine Learning instead of re-written everything from sketch.
Graph X – to store/retrieve the data in graphical format. Mainly provide the capability complex record relation and indexing.