COMPSCI 597: Big Data and NOSQL Databases

Assignment 5

Trisca, Gabriel

October 24, 2013

In brief, explain the Role of Region Servers in the HBASE architecture.

Region Servers, as their name indicates, take care of *regions*. Each region represents a certain amount of storage that contains acertain amount of *keys* and their associated values, growing or shrinking depending on the *key space*, or the number of keys. These servers are managed by *ZooKeeper* –which is equivalent to a *Namenode* in HDFS or a *Jobtracker* in MapReduce, which re-assigns resources, resizes regions when needed and maintains a list of current region servers and their range of keys.

When a client wants to read some key, it consults Zookeeper first to resolve the right region server that contains said key, and from that point on it communicates directly with the region server. To achieve higher performance, clients will cache the key offsets for the region servers they query to skip the Zookeeper altogether. If the regions change or a region server fails, an error occurs and the client is required to query the Zookeeper again for new mapping of `key` → `region server`.

For the case of writes, the client will send the data to the region server that corresponds to the key to be saved. The region server then stores the key and its value in one of its region's memory –which acts as a buffer, and flushes it to the filesystem when full. At the same time, these changes to the data are stored in the Write-Ahead-Log –which is equivalent to a transaction log in traditional databases, to maintain transactional data and recover operations in case of region failure. In consequence, when the data is persisted in memory and not flushed, faster read times can be achieved.

In brief, explain the use of Scanners in HBASE

ResultScanners in HBASE are just like Scanners in Java, with the only difference that while the Java version accepts any `InputStream` –buffered or not, HBASE Scanners will ask the Region Server for new rows every time. They fulfil a role similar to cursors in traditional databases in which a *resultset* is retrieved and then traversed sequentially.

The most common way to utilize scanners in HBASE is to pass a `Scan` object to the method `HTable.getScanner`. This `Scan` object can contain filters for specific families and columns in a table (families are groups of columns), increase the number of rows to fetch to increase performance and set the start and end keys, among other possibilities.

In short, a `ResultScanner` is the equivalent to a cursor that combines both the query that generates some *resultset*, and the cursor itself.