1.What is NoSQL data base?

NoSQL ("Not Only SQL") represents a completely different framework of databases that allows for high-performance, agile processing of information at massive scale. In other words, it is a database infrastructure that as been very well-adapted to the heavy demands of big data.

The efficiency of NoSQL can be achieved because unlike relational databases that are highly structured, NoSQL databases are unstructured in nature, trading off stringent consistency requirements for speed and agility.

NoSQL centers around the concept of distributed databases, where unstructured data may be stored across multiple processing nodes, and often across multiple servers. This distributed architecture allows NoSQL databases to be horizontally scalable; as data continues to explode, just add more hardware to keep up, with no slowdown in performance.

The NoSQL distributed database infrastructure has been the solution to handling some of the biggest data warehouses on the planet – i.e. the likes of Google, Amazon etc.

NoSQL database examples include Hadoop, Hbase, MongoDB, BigTable, Redis, RavenDB Cassandra, HBase, Neo4j and CouchDB.

2.How does data get stored in NoSQl database?

Key-value stores are the simplest NoSQL databases. Every single item in the database is stored as an attribute name (or 'key'), together with its value.

Apache Hive organizes data into tables. This provides a means for attaching the structure to data stored in HDFS. It is a non-relational column oriented distributeddatabase that runs on top of HDFS.

It is a NoSQL open source database which stores data in rows and columns.

3. What is a column family in HBase?

In the HBase data model columns are grouped into column families, which must be defined up front during table creation. Column families are stored together on disk, which is why HBase is referred to as a column-oriented data store.

4.How many maximum number of columns can be added to HBase table?

There is a limit to the number of column families in HBase. There is one MemStore(Its a write cache which stores new data before writing it into Hfiles) per Column Family, when one is full, they all flush.

The more you add column families there will be more MemStore created and Memstore flush will be more frequent. It will degrade the performance.

5.Why columns are not defined at the time of table creation in HBase?

Column families are part of the schema of the table. We can add them at runtime with an online schema change. The reason column families are part of the schema and would require a schema change is that they profoundly impact the way the data is stored, both on disk and in memory.

6.How does data get managed in HBase?

Just like in a Relational Database, data in HBase is stored in Tables and these Tables are stored in Regions. When a Table becomes too big, the Table is partitioned into multiple Regions. These Regions are assigned to Region Servers across the cluster.

7.What happens internally when new data gets inserted into HBase table?

All HBase data is stored in HDFS files. Region Servers are collocated with theHDFS DataNodes, which enable data locality (putting the data close to where it is needed) for the data served by the RegionServers. HBase data is local when it is written, but when a region is moved, it is not local until compaction.