Goodies:

1. Both Cassandra and HBase are open-source NoSQL Database. Both databases can manage extremely large data sets due to high linear scalability and handle non-relational data.
2. There is internode communication in both Cassandra and HBase. However, Cassandra uses Gossip Protocol for it. After this, the data will transfer from one node to another. In other words, we replicate the data. For this internode communication, HBase relies on Zookeeper Protocol. In this, one node acts as the boss through which all the other nodes get the data.

Background:

**HBase**: open-source, non-relational, distributed database modeled after Google’s BigTable. Initial release in March 2008.

1. Highlight:
   1. Based on Master Slave Architecture Model (https://en.wikipedia.org/wiki/Master/slave\_(technology))
   2. Runs on top of the Hadoop Cluster. Technologies like Apache Hive can be put on top of Hadoop for providing SQL-like interface for data query and analysis
2. Wiki: “HBase is a column-oriented key-value data store and has been idolized widely because of its lineage with Hadoop and HDFS”
3. Wiki: “HBase runs on top of HDFS and is well-suited for faster read and write operations on large datasets with high throughput and low input/output latency”

**Cassandra**: free and open-source, distributed, wide column store, NoSQL database management system. Cassandra takes inspirations from both Amazon’s Dynamo and Google’s BigTable

1. Highlight:
   1. Based on Active – Active Node Model (https://en.wikipedia.org/wiki/High-availability\_cluster)
   2. Reliability: “treats failures as the norm rather than the exception”
      1. All nodes are in sync with each other. Every node contains a copy of the entire data within the cluster. In case of any node failure, client can still read data from rest available nodes
   3. Efficiency
   4. Scalable
2. Facebook released Cassandra as an open-source project on Google code in July 2008
3. CQL (Cassandra Query Language) is Cassandra’s query language but having syntax same as SQL. It supports all major OS like Linux, Unix, OSX, and windows
4. Why are they created at the first place?
   1. **HBase**: Apache HBase began as a project by the company *Powerset* out of a need to process massive amounts of data for the purpose of natural language search
   2. **Cassandra**: Facebook designed Cassandra to fulfill the storage needs of the Inbox Search problem
      1. Inbox Search enables users to search through their Facebook Inbox. The system was required to handle a very high write throughout, billions of writes per day, and also scale with the number of users
5. Commons between HBase and Cassandra:
   1. Open-source
   2. High throughput

Architecture (Photo):

1. SPoF (Single Point of Failure): Cassandra has a masterless architecture while HBase is master-based:
   1. When a master goes down in HBase system, although the cluster still have some working time due to the fact that an HBase client communicate directly with the slave-server without contacting the master, HBase is incompetent with Cassandra in this case because its cluster is always available. Therefore, if a user cannot afford any downtimes, Cassandra is preferable than HBase
2. Data-replication
   1. Because Cassandra tries hard to ensure availability, it replicates sync data between nodes which creates data inconsistency problem. Hence, from another angle, if a user relies heavily on data consistency, a stronger consistent HBase system, which writes data only to one place, is superior than Cassandra.
3. Cassandra supports both data management and storage by itself, while HBase is only designed for data management and typically relies on HDFS for data storage.
4. DR (Disaster recovery): DR is possible in HBase if two master nodes are configured. In Cassandra, since it has Active – Active Node model, all nodes contain the copy of the same data. A recovery from disaster is easy.
5. Faster and more consistent writes for HBase because it writes only on one server

Security

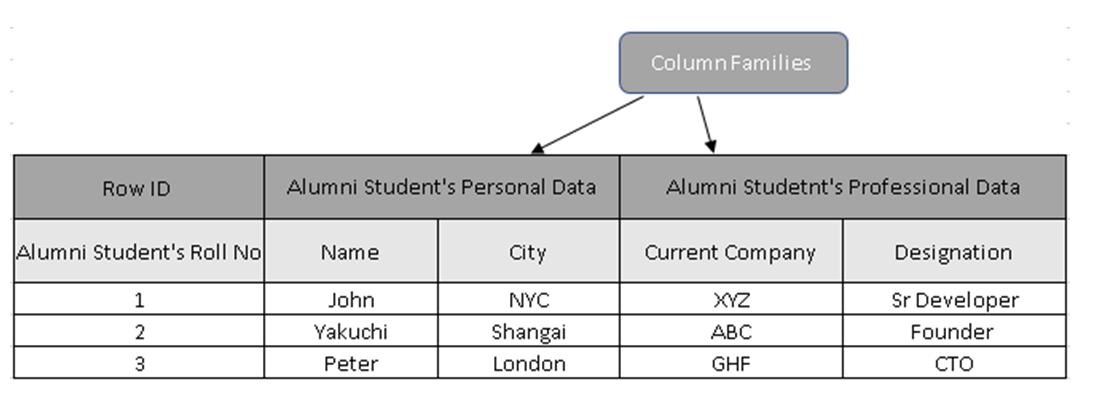
Similar to all NoSQL databases, both HBase and Cassandra have some security issues because making the effort to secure data tends to make the system heavy and drag down the performance.

Nevertheless, both databases still have some features to secure data. Beyond the authentication and the authorization that exist in both HBase and Cassandra, Cassandra has client to node encryption, while HBase ????

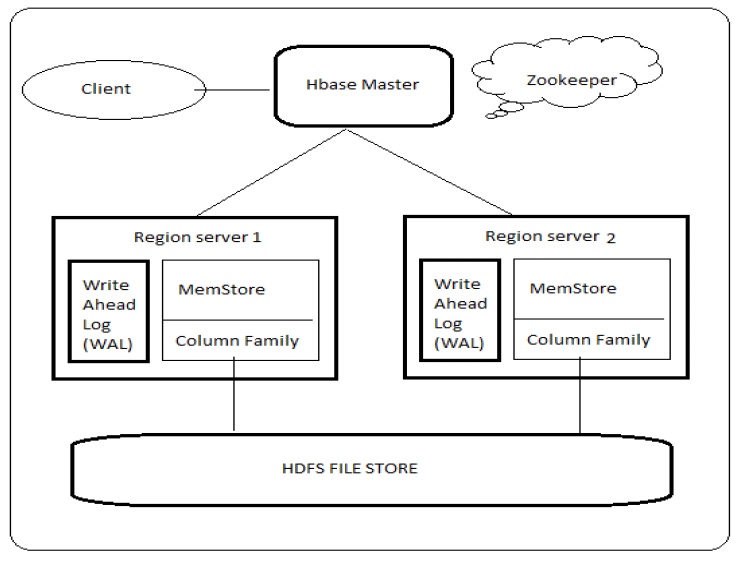
Application

Facebook moves from Cassandra to HBase (https://www.facebook.com/notes/facebook-engineering/the-underlying-technology-of-messages/454991608919/)

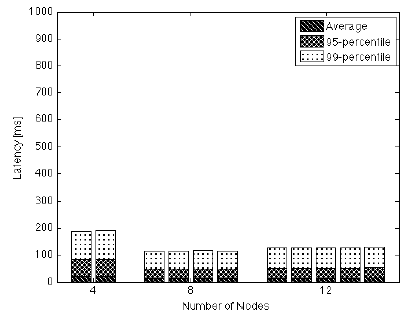
**Sample Table in HBase**



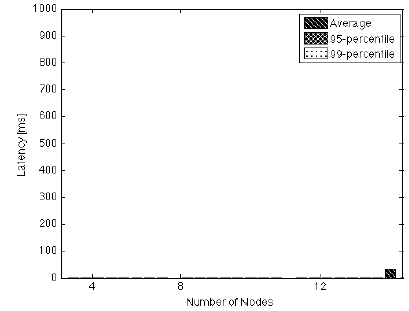
**HBase Data Model (Reference: Comparative Study of NoSQL Document, blablablab)**



**Cassandra write latency (Reference: Benchmaking Scalability and Elasticity of Distributed Database System)**

****

**HBase write latency (Reference: Benchmaking Scalability and Elasticity of Distributed Database System)**

****