Cassandra, HBase and Mongo Db: Performance Evaluation of NoSQL Databases

ARIF HASAN[1], DR. P. SASIKALA[2]

[1] *Research Scholar (Computer Science & Application) Makhanlal University, Bhopal*

[2] *Professor, HOD New Media Technology Makhanlal University, Bhopal*

Abstract— Before starting social networking sites RDBMS was dominant database but as soon as social media came and for handling its unstructured data NoSQL database came into picture. NoSQL databases are four types column, key-value, document oriented and graph databases.In this research paper we will compare performances od Cassandra, Hbase and MongoDb NoSql databases

Indexed Terms-- Cassandra, Hbase, MongoDb, Performance Evaluation

I. INTRODUCTION

Cassandra is wide column open source . NoSql databases that is used in Fcebook, Twitter and many other Social Networking sites.[1] . Basic features of Cassandra are that it gives AP on CAP ,Cassandra performs fast reads and writes on database,Cassandra does not support complex type secondary index.Cassandra provides no point failure facility. Whereas Hbase is Column based NoSql database that provides range based scan and seamless scalability.Hbase also supports BigTable,Bloom filters and block cashes that play vital role in optimizing queries.[L. George, HBase: the definitive guide. O'ReiUy Media, Inc. , 2011.]where as MongoDb is distributed NoSql database system it was created to address the issues of internet ads.It provides full index support .MongoDB stores data in JSON/Binary JSON like .MongoDB Works on real time analytics and offer high speed logging and scalabilty.It also Offers CP from CAP Theorem and features of auto sharding for easy scalability [2] . Now one by one we will see each NoSQL database

II. CASSANDRA

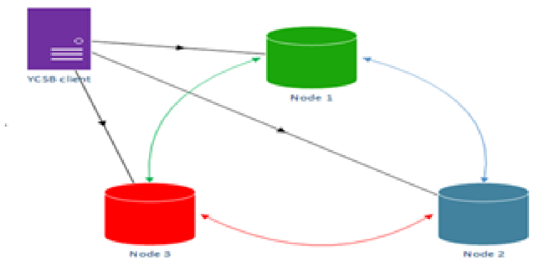

Figure 1 Cassandra Architeture

Cassandra belongs to wide column store NoSQL Family which provides an extended key value store method built on column oriented structure. Its architecture shown in the diagram is based upon ring topology in which every node is identical to others, gives guarantee that system has no point of failure. Each inserted record in the database has a associated hash value called token. The range of tokens is partitioned among the nodes to balance the ring.[3]Cassandra allows data replication among the nodes in the cluster by duplicating data from one node to another node in the ring User can use replication factor as a parameter set to control number of replica of each data Consistency level in Cassandra defines the number of replica that should respond to a data request It can be decided which node can communicate out side of the ring these are entry point**s.**

Cassandra has following properties

- Shared-nothing, master-master architecture,
- in-memory database with disk persistence.
- Key range based automatics data partitioning.
- Synchronous and asynchronous Cassandra replication across multiple data centers.
- High availability.
- Client interfaces include Cassandra Query Language (CQL), Thrift, and MapReduce.

- Largest known Cassandra cluster has over 300 TB of data in over 400-node cluster.
- It offers easy setup and maintenance (does not matter how big the dataset that you are setting)
- Flexible parsing and wide column requirements.[4]
- Not with multiple secondary indexes.
- It allows applications to write into any node anywhere and anytime.
- Automatic workload management and data balancing across the nodes
- Linearly scalable by just adding more nodes to the cluster.
- It can work as amazing, record-setting reliability at scale.
- Eventual consistency yields high availability.
- It offers Wide-column flexibility.
- It also offers minimal administrative tasks at scale

III. HBASE

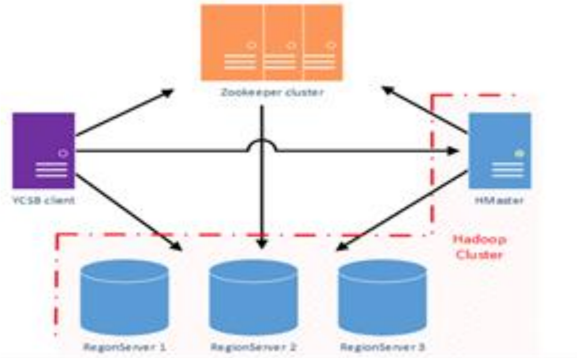

Figure 2 Hbase Architecture

It is based upon Hadoop Distributed File System(HDFS) and Hadoop Map Reduce Framwork.[5].It belongs to wide column store NoSQL Family, Hbase uses two supporting applications Hadoop**:** a distributed map-reduce framework that provide high throughput access to application data and also manages replication in Hbase, Zookeeper: It provides distributed configuration and synchronization service for large distributed System Hbase has main two types of nodes: The master That keep account for live node and which provides communication services. Regions are allocated to a node and split when they too big. Thus in Hbase data accumulation in node is non-uniform[6].Hbase has following properties

- Linear and modular scalability.
- Strictly consistent reads and writes.
- Automatic and configurable sharding of tables
- Automatic failover support between RegionServers.[7]
- Convenient base classes for backing Hadoop MapReduce jobs with Apache HBase tables.
- Easy to use Java API for client access.
- Block cache and Bloom Filters for real-time queries.
- Query predicate push down via server side Filters
- Thrift gateway and a REST-ful Web service that supports XML, Protobuf, and binary data encoding options
- Extensible jruby-based (JIRB) shell
- Support for exporting metrics via the Hadoop metrics subsystem to files or Ganglia; or via JMX
- HBase is a high-reliability, high-performance, column-oriented, scalable distributed storage system that uses HBase technology to build large-scale structured storage clusters on inexpensive PC Servers.
- The goal of HBase is to store and process large amounts of data, specifically to handle large amounts of data consisting of thousands of rows and columns using only standard hardware configurations.[8]
- Different from MapReduce's offline batch computing framework, HBase is random access storage and retrieval data platform, which makes up for the shortcomings of HDFS that cannot access data randomly.
- It is suitable for business scenarios where real-time requirements are not very high — HBase stores Byte arrays, which don't mind data types, allowing dynamic, flexible data models.[9]
- Hbase located on the structured storage layer.HDFS provides high-reliability low-level storage support for HBase.MapReduce provides high-performance batch processing capability for HBase. ZooKeeper provides stable services and failover mechanism for HBase. Pig and Hive provide HBase for high-level language support for data statistics processing, Sqoop provides HDB with available RDBMS data import function, which

makes it very convenient to migrate business data from a traditional database to HBase[10].

IV. MONGODB

Mongos:The only instances able to communicate outside the cluster.

Mongod**:** data nodes storing and retrieving data
Config –Server: The container of the metadata about the objects stored in the monogod. The Metadata is used in case of node failure. A cluseter allows only one or three config-server instances .each running component constitute node in a cluster. Replication achieved by sharding . Every shard is group of one or more nodes. The number of nodes in a shard decides replication factor. Nodes belonging to a same shard have same data.In each shard only one node is master can perform read and write operation where as all other nodes can perform only read operation and known as slave.[11]

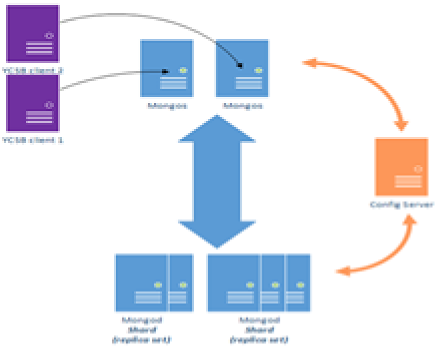

Figure-3 MongoDB Architecture

MongoDB has following Properties

- MongoDB is highly optimized for JSON, it stores data in flexible JSON-documents that means the columns may vary document to document and the data structure may be reformed over time.
- Easy to work with as the object mapping is done by the document model in the application code. The real-time aggregation, the indexing, and queries give significant ways to access and examine the data[12].

V. COMPARISION AMONG CASSANDRA, HBASE, MONGODB PERFORMANCE

The following are the comparisons between the three databases: MongoDB, Cassandra, and Hbase.

- Data availability**:** Cassandra supports multiple master clusters while MongoDB and HBase use a single master cluster. If a MongoDB or HBase master cluster fails, delays in large dataset processing and management usually follow.
- Scalability**.** MongoDB and HBase's single master node limit their write scalability. Cassandra's multiple master nodes enhances and increases scalability (model write speed).
- Data modelling**.** MongoDB is document and object-oriented. Cassandra and HBase use traditional tabular column and row structure.
- Query Language**.** MongoDB only supports JSON-like queries. HBase works well with Hive, a query engine for batch processing of big data. Cassandra has its own query language (CQL). Query intensive data management favors Cassandra.
- Data Aggregation**.** MongoDB has a built-in aggregation framework, Cassandra and HBase do not. MongoDB uses a built-in ELT-multi-stage pipeline that performs transformation of the MongoDB's JSON-like documents into aggregations
- NoSQL Schema. MongoDB does not always require a scheme, allowing for files of different structures to be stored, analyzed and interpreted. Cassandra and HBase are more stationary NoSQLs that are less flexible
- Supported programming languages. MongoDB leans more toward supporting the data executable within nodes with extension node.js and usually supports programming languages such as Java, PHP, C++, C#, Python, R, Scala, Ruby, MatLab, and Javascript. Cassandra does not support those languages operable in systems with node.js. HBase supports the fewest languages.
- Secondary Indexes. MongoDB possesses high-quality indexes. Cassandra only provides cursory assistance for secondary indexes. durability, it would be better to use relational databases such as PostgreSQL and MySQL, instead of MongoDB or Cassandra.

CONCLUSION

There is no doubt that MongoDB is one of the most popular open-source NoSQL databases, but wide column databases like Cassandra may provide better query performance and always-on capabilities. In case of DBaaS services, where you can offload the management and maintenance of the database to the provider, and the developer can simply focus on their application. HBase, in this context, is lacking, while MongoDB has very mature DBaaS offerings, like MongoDB Atlas. On the other hand, HBase can be a very good solution for write-heavy applications and enormous amounts of records.

REFERENCES

[1] J. Gantz and D. Reinsel, " The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east," IDC iView: IDCAnalyze the Future, vol. 2007, pp. 1-16, 2012

[2] P. Groves, B. Kay y ali, D. Knott, and S. Van Kuiken, " The 'big data' revolution in healthcare," McKinsey Quarterly, 2013.

[3] G. DeCandia, D. Hastorun, M. Jampani, G. Kakulapati, A. L akshman,A. Pilchin, S. Sivasubramanian, P. Vosshall, and W. Vogels, "Dynamo: amazon's highly available key - value store," in SOSP, vol. 7, pp. 205-220, 2007.

[4] L. George, HBase: the definitive guide. O'ReiUy Media, Inc. , 2011.

[5] G. Vaish, Gelling Started with Nosql. Packt Publishing, 2013.

[6] G. Vaish, Gelling Started with Nosql. Packt Publishing, 2013.

[7] B. G. Tudorica and C. Bucur, "A comparison between several nosql databases with comments and notes," in Roedunet International Conference (RoEduNet), 2011 1 0th, pp. 1-5, IEEE, 2011.

[8] R. Cattell, "Scalable sql and nosql data stores," ACM SIGMOD Record,vol. 39, no. 4, pp. 12-27, 2011.

[9] N. Developers, "Ne04j," Graph NoSQL Database [online], 2012

[10] 1. Han, E. Haihong, G. L e, and 1. Du, "Survey on nosql database," in Pervasive computing and applications (ICP CA), 20 11 6th international conference on, pp. 363-366, IEEE, 2011.

[11] 1. C. Anderson, 1. L ehnardt, and N. Slater, CouchDB: the definitive guide. O'Reilly, 2010.

[12] D. Bartholomew, "Sql vs. nosql," Linux Journal, vol. 2010, no. 195,p. 4, 2010.