***Database Comparison (MongoDB vs Cassandra)***

Cassandra database could be used to store and handle large data than using MongoDB. In order to prove respective features, resilience, availability, scalability, fault tolerance and consistency, each database is explained below,

***MongoDB:***

***MongoDB*** is a document-oriented database and JSON query language where data stored as key-value pair in document. It doesn’t replace relational database because mongo DB doesn’t follow any ACID properties, normalization and consistency and it stores any kind of data unlike RDBMS.

Below example shows how data stores in mongoDB.

{

student\_Id: 111,

student\_name: ‘Arvind’,

student\_course: ‘MIT’,

Unit: ‘Database’

}

Here, key is ‘student\_id’ and value is ‘111’, hence it is called as key value pair. In comparison with RDBMS,

In MongoDB,

TABLES are called as COLLECTIONS,

Database called as Database,

ROWS are called as DOCUMENT.

The main advantage of MongoDB is, it follows schema less model approach, rich data model, secondary indexes, native aggregation, replication and high availability. It can perform dynamic queries using document-based query language and supports deep query ability & JSON query language. It is faster than traditional database system but *slower than other NOSQL database.*

***Cassandra:***

***Cassandra*** is the column-oriented NoSQL database. It is distributed, decentralized and open source database system used to handle large number of data in multiple servers. In addition to that, In Cassandra, data are structured and provides no single point failure with highly available services.

In Cassandra, data are stored as column family which consists of set of related columns. Each column family is inserted as rows in the database and it can be identified using unique row-id. The column values are indexed with row-id, column name and timestamp.

Cassandra is linearly scalable and fault tolerance and better performance as multiple master nodes are available, data lost in one node could be retrieved from another node. Hence it follows peer to peer architecture approach. More data can write at once as more master nodes available and each master node can accept different writes in parallel.

***Steps to implement merging process:***

***Strengths and Weakness of MongoDB and Cassandra:***

|  |  |
| --- | --- |
| ***MongoDB*** | ***Cassandra*** |
| ***Strengths:***   * It provides better and rich expressive *data model*. * *Secondary indexes* are first class construct, so it is easy to query and index on any property of data stored in the database. * Provides built in aggregation framework. * No need to use external tools for aggregation. | ***Weakness:***   * It does not provide rich *data model* than MongoDB as it follows traditional model table structure. * *Secondary indexes* are supported and are limited to single column and equality comparisons. * Doesn’t provide built in aggregation framework. * Provides external tools like Apache spark, Hadoop etc., to perform aggregation framework. |
| ***Weakness:***   * Cluster can have down time and provides delay in taking any input when master node goes down as there is one master node and multiple slave nodes. * Failure on master node takes time to elect any slave node as master node. * Master node accepts only input and slave node accepts only output, so data is written in slave nodes and it is pass through master node. * Cannot handle heavy load data input as it has only single master node. | ***Strengths:***   * No down time on cluster and no delay takes place in taking any input as Cassandra has multiple master node. * Failure on one master node provides no impacts as other master node takes in charge. * Since Cassandra has multiple master node, it is used to input data in other nodes. * Handles heavy load data input as it has multiple master node. |

***Valid explanation on choosing Cassandra Database to use:***

MongoDB and Cassandra provide equal weightage on its own advantages and disadvantages. Depending on the size of the data, database can be chosen.

Even though MongoDB handle lot of highly unstructured data with its flexible data structure because of its one Master and multiple slave architecture, there will limited write speeds due to down time. To overcome this and to handle larger amount of data with multiple servers, Cassandra can be used whereas MongoDB can also be used for large data but Cassandra provides no single point failure and cluster down than MongoDB *therefore it is better to use Cassandra as it is more resilient and highly available than MongoDB*.