MongoDB

History

MongoDB database application is an application developed by MongoInc. This organization was founded by Dwight Merriman, Eliot Horowitz and Kevin Ryan in 2007. The fist MongoDB application is introduced to the public in a form of a platform between the periods of 2007 and 2008. Next MongoInc released their first nosql database in 2009. In 2017 MongoDB was listed in NASDAQ. MOngoDB version 6 was released in 2022. The title of the application defines the nature itself. Humongousness of the application allows users to store a large quantities of data without worrying about the structure of the application. MongoDB databases are used mainly in IoT devices, mobile applications, real-time analysis, personalizations, catalog management and content management. Enterprises such as Google, Facebook and ebay deal with their growing data using MongoDB. Let’s examine the marvel behind the MongoDB databases starting from the next chapter.

Concepts

The collections in MongoDB acts as tables in relational databases while a document in a nonrelational database represents a row. A document consists of various fields and these fields are labeled as columns in relational databases. Values are stored as key value pairs in MongoDB. Documents use the BSON format when storing values. The key features of MongoDB are aggregation, gridfs, sharding, document oriented, replication, shchema less database, indexing, AD HOC Queries and high performance. MongoDB uses Data Modeling to store data. The relations that exists between entities and the way the data is stored is defined as data modeling. It helps to build logical databases that are efficient. These databases offer low storage requirements, grants efficient data retrieval and limits redundancy. Through the data modeling process, the following objectives are archived.

High Data Quality

Comprehension of dataflow and characteristics

Development and Maintenance

Performance

Data models are categorized into three types based in the specificity/detail and and they are conceptual, logical and physical data model. One to One, One to many and Many to Many are relationships types that are used in data models. Embedded and Reference data models are two methods used for creating a model. Logical or mathematical operations operations are executed in MongoDB by using query operators. Query and projection, update and aggregation pipeline operators are examples for query operators.

Regular expressions in nosql queries are used to match patterns in a document. List given below point out reasons for using regular expressions.

Match text and define search patterns in sequences of characters.

Retrieve data of a undefined field.

Locate small subsets of data within a collection.

The database administors are able to view only the necessary data in a document using the projection feature of MongoDB. Projection feature comes in handy when removing unnecessary fields from query results, retrieving indexed query results without fetching full documents and filtering data without causing any database performance issues. Sorting method in MongoDB defines the order in which query returns the matching documents in collection. The number of retrieving documents can be limited by using the limit method. Indexes in mongoDB aid in query data in an efficient manner. Avoidance of collection scans, effectivity of indexing strategies and searching efficiency are achieved through the indexing mechanism. Single field, compound and multi key index are examples for indexes. Creation of a single index in a document is done by using single field index while references to multiple fields in a document is obtained through compound index. The values stored in arrays are indexed by multi key indexes. The data retrieval speed is enhanced by the advanced indexing feature. Geospatial, text and hashed indexing are techniques used in advanced indexing. Geospatial coordinate data are handled by 2d indexes and 2d sphere indexes. Text search queries are empowered by text indexes. Entries with hashes are indexed using hash indexes. Gaining accurate result through processing multiple documents by selcting data from a collection id defined as aggregation. Aggregation in mongoDB is performed using group by clause and complex aggregation operations. It is performed to group values of multiple documents together and to fetch nested data to perform complex operations. In addition it is used to filter and sort documents and analyzing the changes in data. Match, group, sort and project are aggregation pipeline stages. Operations in aggregation are sum, average, minimum, maximum and push. Replication is the process of cloning same data across multiple MongoDB servers. The intentions behind replication are listed below.

Data redundancy and high availability.

Making copies of data across servers.

Performing data backups and recoveries.

Replication is performed to gain the availability of data and to dampen the effects of a single server loss.MongoDB manages replication through replica sets. Collection of MongoDB nodes are defined as MongoDB nodes. A minimum of three MongoDB nodes are required to form a replica set. All the writing operations are performed by the primary node while secondary nodes that are under the root node, copy data from the primary node. When a primary node fails, it will be elected as a secondary node. The benefits and drawbacks of replication are discussed in the table given below.

|  |  |
| --- | --- |
| Advantages | Disadvantages |
| Availability | Higher usage of storage and network bandwidth |
| Scalability | Higher levels of complexity in operation and configuration and monitoring. |
| Enhanced Performance | Lags in propagation. |
| Disaster Recovery | Challenges in maintenance. |
| Continuous Operations | Consistency Issues. |

The list given below highlights the limitations of replication.

Difficulties in handling data and query traffic.

Unability of MongoDB instances to manage write operations.

Unability of outsizing memory in regards to large datasets.

Higher costs.

The distribution process of data in MongoDB is designated as Sharding. A single dataset is stored in multiple databases. Sharding process adds more memory and processing units or ram into a single server. Sharding is associated with two types of scaling methods and they are vertical and horizontal scaling. Adding more resources to the server is defined as vertical scaling while horizontal declares of adding more processing units or physical machines to the server or database. Sharding uses a an architecture and this architecture uses sharding clusters. A sharding cluster is formed out of multiple shards, mongos processes and configuration servers. Shard key distributes data among the shards and a shard key is automatically created by the database. The sharding process creates a cluster. A cluster of mOngoDB instances forms at least three servers. The list below addresses the pros of sharding.

Increased storage capacity.

Increased read/write throughput.

High availability.

Facilitates horizontal scaling.

The way multiple documents are logically connected to each other in MongoDB is represented in relationships. Database structures are refined by creating relationships between documents. In addition they link up the entities in a database and make execution times shorter. Relationships in documents are categorized into embedded and reference relationships. Reference relationships are achieved through many to many relationships while embedded relationships are achieved through one to oen and one to many relationships. Files that are greater than 16MB are stored using GridFS driver in mongoDB. GridFS mechanism divides large files into smaller chunks and store the as separate documents. The list given below addresses the probable causes of using GridFS.

Giving the privilege of storing files that are greater than 16MB.

Giving the access to a portion of a file without loading the whole file.

Storing and syncing files and metadata across distributed systems.

By default GridFS uses fs files and fs chunks to store the file’s metadata and the chunks. The data processing paradigm for condensing large volumes of data into useful aggregated results is defined as Map Reduce. This techniue is mostly used for large datasets. Map, reduce and query functions are executed MapReduce method. Referencing documents stored in multiple collections or databases is performed by using the DBref option. Ref, ID and DB parameters are used with the DBRef option. Cover queries use indexed fields to gain efficient results without searching a whole collection. Given below are scenarios when cover queries comes in handy.

When all the fields in query are part of an index.

When all the fields returned in query are in the same index.

When there are no null values in fields.

All data which require an update is stored within a single document to maintain the atomicity. There are certain constraints regarding the atomicity of MongoDB and they are listed below.

Zero support for multi document atomic transactions.

Only allows to perform atomic operations in a single document.

Atomocity is maintained and executed at the document level.

Set, inc, push, pull and rename are commands used in performing atomic operations. High throughput operations that insert retrieve and delete documents based on insertion order are stored as fixed collections under the capped collections. When collection size reach over its maximum threshold, it starts to delete the old data autonomously. Due to the following reasons capped collections are used.

To make sure that recently entered data stay in the database.

Automate the old data removal process.

Store cached data that needs regular refreshing.

Capped collections has it’s own characteristics and they are listed below.

Delete operations not allowed to be performed unless it is removed automatically.

All elements in a collection should have an equal size.

No indexing is required because of working queues.

Helps to maintain log files to troubleshoot any errors.

Mechanisusms such as authorization and authentication safe guards the MongoDB database from unauthorized access. MongoDB gives the capability to data expert to create users, grant access control for the created users and authenticate users.

Sectionizing a collection of related commands which should be executed with similar session options are defined as session commands.Examples for session commands are abortTransaction, commitTransaction, endSessions, killAllSessions, killAll SessionByPatterns, refreshSession, killSessions and startSession. Undoing modifications and restore the database back to its normal condition is performed by abortTransaction command. The commitTransaction command saves the changes made by operations in a multi-document transaction before ending the transaction. Before terminating a session, the endSession command overides the timeout window. Ongoing operations in a session is terminated by using the killSession command. Any sessions that are twined with declared patterns is terminated by the killSessionbyPattern command. MongoDB tools such as bsondump, mongodump, mongoexport, mongofiles, mongoimport, mongorestore, mongostat and mongotop helps data engineers in the process of backing up and restoring the databases. MongoDB Atlas is a cloud based data storage service. Management and monitoring of MongoDB deployments in enterprises are carried out by the MongoDB Ops Manager tool. MongoDB databases have pro and cons. They are discussed in the next section thoroughly.

Advantages and Disadvantages

The MongoDB inherits it’s benefits and drawbacks and they are highlighted in the table below.

|  |  |
| --- | --- |
| Advantages | Disadvantages |
| Open source | Issues related to consistency. |
| Easy to use | Consumes a lot of resources. |
| Highly flexible | Less capabilities in creating relations. |
| Advanced security features | Complexity of managing transactions. |
| High Availability | Less compatibility |
| Reliable Indexing | Maintenance and management |
| Flexible Schema |  |
| High Performance |  |

The next section discuss about application of MongoDB’s concepts towards the Nexart’s web application.