NoSQL & MongoDB
CS 623: Database Management Systems

Group Members
Agarwal, Kritika
Jadhav, Mahesh Dilip
Yelgati, Aditya

Professor. Mustafa, Shahed

Summer 2022
Pace University
New York, NY

NoSQL: Introduction

NoSQL databases are non-relational databases that use documents to store data unlike the tables used in Relational databases. NoSQL has the meaning “Not only SQL” which means that data can be retrieved without using SQL also. It supports more than SQL language.

NoSQL databases have a flexible structure. They are highly scalable systems and very robust. NoSQL databases can be used to store unstructured, semi-structured, or structured data. NoSQL databases provide support for horizontal scaling by adding more data nodes of the same configuration to the existing nodes for data storage or processing as the data size grows. NoSQL databases give priority to the performance of the system and provide high flexibility in database design

Examples of NoSQL databases provided by different tech companies are:

1. DynamoDB by Amazon
2. CosmosDB by Microsoft
3. Cassandra by Facebook.

Classification of NoSQL Databases

The different types of NoSQL databases are:

1) Key-Value databases:

The key-value database is the simplest and most basic type of NoSQL database. Every data object database is stored in the form of key-value pair. The attribute of the data element is called "key” and the value corresponding to that key is stored in “value”. The data is stored in a hash table in a Key-value pair where each key is unique, and the value can be a number, string, Boolean, JSON, or BLOB (Binary Large Objects). Key-value databases are used as collections, dictionaries, associative arrays, etc.

Application: The applications of Key-value databases are shopping carts, user preferences, and user profiles.

Example: The most commonly used key-value databases are Redis, Memcached, and Couchbase.

KEY	VALUE
Name	"Thompson, Rich"
Employee_ID	"78784903"
Department	"Sales"
Team	"NYC"
BandLevel	"5C"
CostCenter	"8789"
Supervisor	"Bill, Chen"

2) Column-oriented databases:

Column Oriented databases store the data in a form of columns. Each column consists of data elements of a homogeneous category. It is very useful when real-time data is required for analytics that may be social-networking sites, live cricket matches, and so on. Column-oriented databases read column data without consuming large memory by storing all unwanted data. Column-oriented databases give high performance on aggregation functions like SUM, AVG, COUNT, and so on as the data is readily available to perform these operations.

Application: Column-based databases are used to manage Data warehouses, Business Intelligence, CRM, and real-time analytics.

Example: Examples of Column-oriented databases are HBase, Cassandra, Hypertable, and Amazon DynamoDB.

Row Oriented

Airport_Code	City	State	Name	No_of_Terminals
JFK	New York	NY	John F. Kennedy International Airport	8
EWR	Newark	NJ	Newark Liberty International Airport	3
LGA	New York	NY	LaGuardia Airport	4

Column Oriented

Airport_Code	City
JFK	New York
EWR	Newark
LGA	New York

Airport_Code	State
JFK	NY
EWR	NJ
LGA	NY

Airport_Code	Name
JFK	John F. Kennedy International Airport
EWR	Newark Liberty International Airport
LGA	LaGuardia Airport

Airport_Code	No_of_Terminals
JFK	8
EWR	3
LGA	4

3) Document databases:

Documents are the fundamental idea in document databases. Document databases stores data in the form of a document. The documents are mostly in XML and JSON format. In Document Based databases, all data relative to an object is stored as a single instance. They provide embedded documents, which are self-describing, hierarchical tree structures that comprise maps, collections, and scalar values.
Document databases like MongoDB have immensely powerful and rich query language and constructs, with which a simpler move from relational databases is possible.

Application: Document databases are widely used in CMS systems, blogging platforms, and E-commerce web applications to manage the huge volume of media and document-based data.

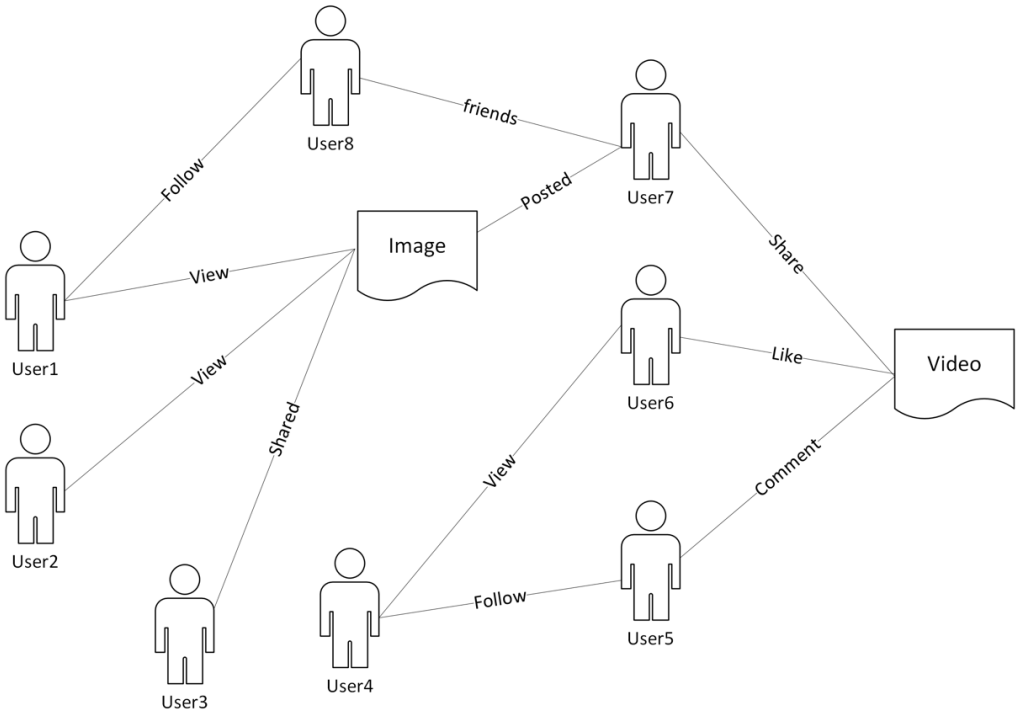
Example: Some Examples of document databases are CouchDB, MongoDB, OrientDB, etc.

4) Graph Databases:

A graph database is another type of NoSQL database that stores the data in the form of edges and nodes. An element is represented by a node that may be connected with another node. The connections between elements are called links or relationships and are stored as edges in the database. Graph databases are useful for forming and storing the relationship of the data. A graph database is optimized to capture and search the connections between data elements.

Application: Use cases included are fraud detection, social networks, and knowledge graphs.

Example: Neo4J, InfiniteGraph, OrientDB, or FlockDB are some of the popular graph databases commonly used.



Characteristics of NoSQL Databases

1. Scalability: NoSQL databases provide a highly scalable database system. It means they can be scaled horizontally. They are designed to scale out by using distributed clusters instead of scaling up by adding expensive servers. On the other hand, Relational databases cannot be scaled horizontally and require a new expensive server to scale up.

2. Flexible Data Models: Unlike relational databases where data schema is fixed, the NoSQL database offers the creation of a flexible schema design for storing the data which can be easily altered without any downtime of the database system. They are ideal for data models where the data is structured, semi-structured, or may be unstructured.

3. High-performance: NoSQL databases give higher performance as the same data is available on multiple nodes or there are replicas of the same data. So even if the system failure occurs at one cluster other remains unaffected and data is continuously served by the database server.

4. Fast Queries: Data in NoSQL databases are stored in an optimized way that promotes faster Query execution. Also, there are no joins available in NoSQL databases, unlike the SQL ones.

5. Distributed: NoSQL database has a distributed architecture which means the data is clustered in the form of nodes. There is no master and slave instead database has no control unit. The data is available continuously as multiple nodes are serving the same data.

6. Large Data storage: NoSQL databases can handle large volumes of data that are used in big data applications, cloud storage of Media data, and real-time analytics. Also, the data is readily available to be used despite its huge size.
7. Developer Friendly: NoSQL databases help developers to create data models which are very similar to real-time objects being used in the application. NoSQL databases provide highly functional APIs and data types for storing. For Example, NoSQL provides support to store data types like graphs, images, files, documents, and other media. This feature is widely used in real-time applications which require complex data structures.

Applications of NoSQL Databases

1. Real-Time Big data Processing: Big data refers to a collection of data that are huge in volume and too large to be processed. NoSQL database systems are used for storing and retrieval of Big Data in real-time which requires stream processing, analysis of historical data, and many more operations.
2. Internet of Things: In today's world, huge volumes of data are generated by smartphones, tablets, security devices, electronics, and other home appliances. Relational databases cannot support such huge data. NoSQL databases can store this huge data and allow concurrent access keeping in mind the performance of the system.
3. E-commerce: NoSQL is widely used in e-commerce applications due to its flexible schema and large datasets for storing dynamic products. E-commerce requires scalable systems that can incorporate a large number of users and provides high performance.
4. Social Media websites: Social Media websites generate tons of data each day and are an excellent application of NoSQL databases. Facebook, Instagram, and Twitter rely heavily on NoSQL databases for the huge amount of data generated by user activities like Posts, comments, stories, advertisements, and so on.

MongoDB: Introduction

MongoDB is a cross-platform open-source non-relational database used for high-volume data storage. Data is represented in the form of JSON. MongoDB stores data in the form of collections and documents, unlike relational databases which use tables and rows. Each document in a MongoDB database is stored in the form of key-value pairs. Data is represented in the form of JSON (JavaScript Object Notation).

In MongoDB, we can store nested data which allows the creation of complex relations between the data. This feature helps in the efficient retrieval of data as compared to SQL which involves multiple joins.

MongoDB Representation in JSON format

MongoDB stores data in the form of JSON. JSON stands for JavaScript Object Notation. It is a simple and lightweight text-based data format storing data in the form of key-value pairs using strings. JSON is a language-independent data format and is widely used to transmit data between web applications and servers. It is very easy for a machine to parse the data represented in the form of JSON.

Example of JSON representation for an Employee Object:

{

“_id”: “12345”,

“name”: ”John Denver”,

“dateOfBirth”: “19-04-1990”,

“age”: “35”,

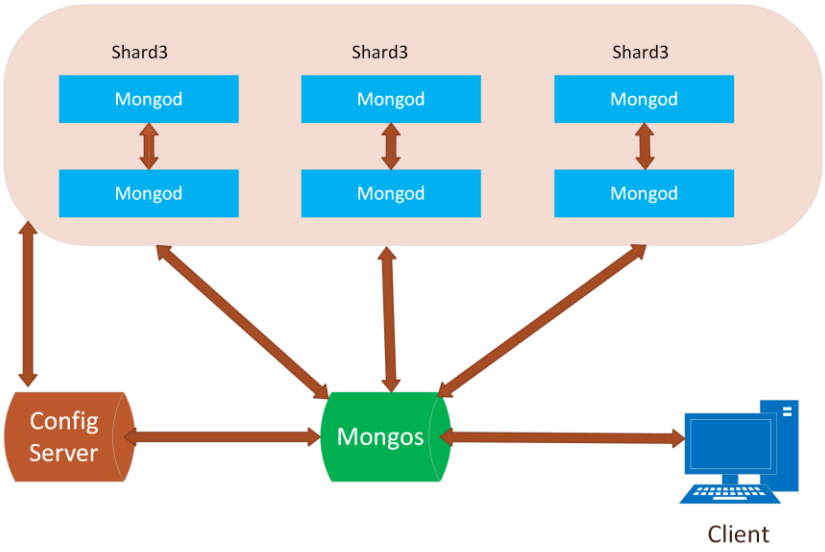
‘’designation”: ”Software Developer”

}

The data types supported by MongoDB JSON are:

- Number
- Array
- Boolean
- String
- Date
- Hash
- Timestamp

Key Components of MongoDB Architecture



1. Database: The database is called the physical container for storing the data.
2. Collection: The group of database documents is called a collection. The database can store many collections in a single database.
3. Document: A single record is called a document. Each document consists of key-value pairs of data elements. A collection comprises multiple documents.

Comparison of concepts used in MySQL and MongoDB

MySQL	MongoDB
Database Instance	MongoDB Instance
Schema	Database
Table	Collection
Row	Document
Column	Field
RowId	_id
Join	Embedded documents

Features of Mongo Database:

1. Schema-less database: Mongo supports a schema-less database design by storing multiple documents in a single collection. Each document may have different fields, sizes, and data types. This feature provides greater flexibility in designing the database.
2. Replication: MongoDB provides replication of data sets. Hence it provides high availability as it creates multiple replicas of the data and sends them to different servers. So, if there is a failure at one server, data can be retrieved from another server.
3. Load Balancing: Load-balancing is a key feature required in big applications. MongoDB supports high scale load balancing as it supports features like replication and sharding. It can handle multiple read and write requests concurrently for the same data.
4. High Performance: MongoDB provides high performance as it supports faster query processing. This is achieved through its features like indexing, replication, and scalability.
5. Sharding: MongoDB supports scalability through a feature called Sharding. Sharding is the process of distributing the data on multiple servers in data chunks using the shard key. The data chunks get evenly distributed in different shards across the physical servers.
6. Aggregation: MongoDB allows operations on a group of data to obtain the computed result. This is supported by three different methods of aggregation, called aggregation-pipeline, map-reduce, and single-purpose aggregation methods.

Comparison of Relational and MongoDB Database based on Queries

Query	Relational Database	MongoDB database
Create	CREATE TABLE table_name (col1, col2….)	db.createCollection(collection_name)
Insert	INSERT INTO(table_name) VALUES (val1, val2…)	db.collection_name.insert({key1:val1, key2:val2})
Select	SELECT col1,col2… FROM table_name WHERE (condition)	db.collection_name.find({condition})
Delete	DELETE FROM table_name WHERE (condition)	db.collection_name.remove({condition})

Future Trends in NoSQL Databases

1. Microservice Integration: Many software applications are following Microservice architecture which is scalable, flexible, and fault tolerant. Traditional databases do not

meet this requirement. Hence organizations are moving towards NoSQL databases like MongoDB, and DynamoDB which provides flexibility, and scalability to the application.

2. Updates in Cloud navigation: MongoDB had been doing different advancements in cloud services like Charts, Stitch, and Atlas. These enhancements are open source, which is easy to understand and implement. The new updates to upgrade the UI experience from the dashboard were centered around the improvement in the work process when MongoDB is utilized as an undertaking-level application.
3. Stitch meets GraphQL: With the developing prevalence among developers to interface with the Database utilizing GraphQL queries, it was nothing unexpected when MongoDB declared it would straightforwardly serve GraphQL queries from MongoDB. This element has been coordinated with Stitch and Realm.
4. Atlas search and Atlas Data Lake: The launch of the MongoDB cloud had a great deal to observe. From the most recent cycles in the Document dataset model in MongoDB 5.1 to the accessibility of Realm. However, the one that was quite anticipated was the accessibility of Atlas Data Lake and Atlas Search. Last year, Atlas Data Lake was pitched as an option in contrast to Hadoop.
5. MongoDB Realm: MongoDB procured the mobile data set organization in April of 2020 and coordinated it with MongoDB Stitch to launch the primary beta, MongoDB Realm. This has prompted numerous upgrades in Stitch itself while also giving an extraordinary stage to mobile data sets that are strengthening the future extent of MongoDB.

Security Features of MongoDB:

In 2022, MongoDB World introduced new security features that will aid businesses in achieving operational excellence. Some of these features are:

- Queryable Encryption: MongoDB will be the only database supplier that permits users to conduct expressive queries after Queryable Encryption is released as well as equality and range, prefix, suffix, substring, and other operations on totally random encrypted data.

  For businesses who need to perform expressive queries while also protecting their data, this is a major benefit. With Queryable Encryption, dealing with encrypted data requires less effort, which speeds up app development without compromising data security or compliance with data privacy laws.

- MongoDB Atlas Serverless: MongoDB has launched Atlas Serverless. Using Atlas serverless, the provisioning and maintenance of servers have been abstracted or hidden vi from the client or end user.

  As a result, there is no longer any cognitive strain associated with sizing and scaling infrastructure to meet application demand. With Atlas Serverless, you only pay for the resources you utilize as opposed to paying for unused ones. Atlas Serverless helps

businesses speed up the time to market and enhance developer and IT manager experiences by making provisioning simpler.

- MongoDB Atlas: MongoDB Atlas has built-in security measures. To integrate with your current security policies and compliance standards, enable enterprise-grade capabilities. Your data is secured with predefined security measures for authentication, and authorization.

- Authentication and Authorization: Verifying a client's identity is known as authentication. All clients must verify themselves while accessing MongoDB when authorization or access control is enabled. These elements have been created to assist enterprises in strengthening their data security.
  Authentication: It establishes a user's identification.
  Authorization: The confirmed user's access to resources and functions is determined by it.

Conclusion:

NoSQL databases' structures are adaptable. They have several benefits, such as being highly scalable, extremely robust, flexible data models, high performance, and developer friendly. NoSQL databases have various applications in social media, the internet of things (IoT), e-commerce, and data mining. Furthermore, MongoDB is a powerful document-oriented database that can be used to store large amounts of data. Additionally, it appears that the security offered by MongoDB's built-in capabilities is adequate. It includes features like MongoDB Atlas Serverless, Queryable Encryption, Authentication, and Authorization.

Moreover, Understanding the ongoing innovations and the market predominance of this simple to utilize data set makes us understand that the future extent of MongoDB shows a great deal of commitment.

References

Wahane, Deepa Suresh, and Prof. Mayuri Dhondiba Dendge. 2018. "Analysis on NoSQL: MongoDB Tool." *International Journal of Trend in Scientific Research and Development* Volume-2 (Issue-4): 1077–81. https://doi.org/10.31142/ijtsrd13089.

Mazurova, Oksana, Illia Syvolovskyi, and Olena Syvolovska. 2022. "NOSQL DATABASE LOGIC DESIGN METHODS for MONGODB and NEO4J." *Innovative Technologies and Scientific Solutions for Industries*, no. 2 (20) (June): 52–63. https://doi.org/10.30837/itssi.2022.20.052.

"NoSQL in Action - a New Pathway to Database." 2016. *International Journal of Science and Research (IJSR)* 5 (6): 872–77. https://doi.org/10.21275/v5i6.nov164282.

F. Shaikh, Nuzhat. 2018. "Data Migration from SQL to MongoDB." *HELIX* 8 (5): 3701–4. https://doi.org/10.29042/2018-3701-3704.

MongoDB. 2019. "The Most Popular Database for Modern Apps." MongoDB. 2019. https://www.mongodb.com/.

Rawat, Soumyaa. n.d. "NoSQL Databases: Characteristics, Types and Applications | Analytics Steps." Www.analyticssteps.com. https://www.analyticssteps.com/blogs/nosql-database-characteristics-types-and-applications.

"NoSQL Databases and Its Types: A Comprehensive Guide." n.d. Learn | Hevo. Accessed August 22, 2022. https://hevodata.com/learn/nosql-databases-and-its-types-a-guide/#s4.

"The Future Scope of MongoDB: Advantages, Improvements & Challenges [2022]." 2021. UpGrad Blog. January 8, 2021. https://www.upgrad.com/blog/the-future-scope-of-mongodb/.

"Future of MongoDB Could Be Brighter than Other NoSQL Engines." n.d. SearchDataManagement. Accessed August 22, 2022. https://www.techtarget.com/searchdatamanagement/feature/Future-of-MongoDB-could-be-brighter-than-other-NoSQL-engines.