Improving Scalability and Performance of Application

K. Saranya, N. Anandh

Muthayammal Engineering College, Rasipuram, Rasipuram, Tamil Nadu, India

ABSTRACT

Now-a-days, most of the enterprise and web application is using the conventional database which is Oracle DB. Oracle Database is uses a table structured data storing methodology. Because of that accessing tables to get the relevant data will be faster. But if the table column needs to be updated or a new column needs to be added then it adds up extra queries and additional work to the developer. MongoDB is an open-source document database, and leading NoSQL database. MongoDB is written in c++ . MongoDB is a cross-platform, document oriented database that provides, high performance, high availability, and easy scalability. MongoDB works on concept of collection and document. Collection is a group of MongoDB documents. It is the equivalent of an RDBMS table. A collection exists within a single database. Collections do not enforce a schema. Documents within a collection can have different fields. Typically, all documents in a collection are of similar or related purpose. A document is a set of key-value pairs. Documents have dynamic schema. Dynamic schema means that documents in the same collection do not need to have the same set of fields or structure, and common fields in a collection's documents may hold different types of data.AngularJs is advanced javascript which helps in many ways to access the mongodb directly from the front end. And it help the application to gain scalability and performance.

Keywords : RDBMS, Oracle DB, DBA, NoSQL, MongoDB, AngularJS

I. INTRODUCTION

In today's world all the business are being done in online and with help of IT technologies. Those applications can be categorized into two as Enterprise application and web application. All these application can be varied based on the data which it consumes and the nature of the websites. Most of the application uses the conventional databases like Oracle DB which is nothing but a table structured DB. Because of that accessing tables to get the relevant data will be faster. But if the table column needs to be updated or a new column needs to be added then it adds up extra queries and additional work to the developer. The Oracle RDBMS stores data logically in the form of tablespaces and physically in the form of data files ("datafiles"). Tablespaces can contain various types of memory segments, such as Data Segments, Index Segments, etc. Segments in turn comprise one or more extents. Extents comprise groups of contiguous data blocks. Data blocks form the basic units of data storage. A DBA can impose maximum quotas on storage per user within each tablespace.

To replace the conventional database one of finest database emerged into the market as MongoDB , MongoDB is an open-source document database, and leading NoSQL database. MongoDB is written in c++ . MongoDB is a cross-platform, document oriented database that provides, high performance, high availability, and easy scalability. MongoDB works on concept of collection and document. Collection is a group of MongoDB documents. It is the equivalent of an RDBMS table. A collection exists within a single database. Collections do not enforce a schema. Documents within a collection can have different fields. Typically, all documents in a collection are of similar or related purpose. A document is a set of key-value pairs. Documents have dynamic schema. Dynamic schema means that documents in the same collection do not need to have the same set of fields or structure, and common fields in a collection's documents may hold different types of data.

Angular Js is javascript which will be used to access the mongodb from the front end. The conventional strategy was like accessing the database through the backend. But with the help of the angularJs the database can be accessed from front end. The AngularJS library works by first reading the HTML page, which has embedded into it additional custom tag attributes. Angular interprets those attributes as directives to bind input or output parts of the page to a model that is represented by standardJavaScript variables. The values of those JavaScript variables can be manually set within the code, or retrieved from static or dynamic JSON resources. Since we use the angularjs it'll improve the application's scalability and performance.

II. LITERATURE SURVEY

2.1. MONGODB ENABLES FACILITIES NEVER POSSIBLE WITH THE TRADITIONAL

RELATIONAL DATABASES:

MongoDB is the database for today's applications, enabling you to:

- Leverage data and technology to maximize competitive advantage
- Reduce risk for mission-critical deployments
- Accelerate time-to-value
- Dramatically lower total cost of ownership

With MongoDB, you can build applications that were never possible with traditional relational databases[6]. Here's how.

Fast, Iterative Development. Scope creep and changing business requirements no longer stand between you and successful project delivery. A flexible data model coupled with dynamic schema and idiomatic drivers make it fast for developers to build and evolve applications. Automated provisioning and management enable continuous integration and highly productive operations. Contrast this against static relational schemas and complex operations that have hindered you in the past.

Flexible Data Model. MongoDB's document data model makes it easy for you to store and combine data of any structure, without giving up sophisticated validation rules, data access and rich indexing functionality. You can dynamically modify the schema without downtime. You spend less time prepping your data for the database, and more time putting your data to work.

Multi-Datacenter Scalability. MongoDB can be scaled within and across geographically distributed data centers, providing new levels of availability and scalability. As your deployments grow in terms of data volume and throughput, MongoDB scales easily with no downtime, and without changing your application. And as your availability and recovery goals evolve, MongoDB lets you adapt flexibly, across data centers, with tunable consistency.

Integrated Feature Set. Analytics and data visualization, text search, geospatial, in-memory performance and global replication allow you to deliver a wide variety of real-time applications on one technology, reliably and securely. RDBMS systems require additional, complex technologies demanding separate integration overhead and expense to do this well.

Lower TCO. Application development teams are more productive when they use MongoDB. Single click management means operations teams are as well. MongoDB runs on commodity hardware, dramatically lowering costs. Finally, MongoDB offers affordable annual subscriptions, including 24x7x365 global support. Your applications can be one tenth the cost to deliver compared to using a relational database.

Long-Term Commitment. MongoDB Inc and the MongoDB ecosystem stand behind the world's fastest-growing database. 10M+ downloads and 2,000+ customers including over one third of the Fortune 100. Over 1,000 partners and greater investor funding than any other database in history. You can be sure your investment is protected.

2.2. MONGODB USING DATA MODELS (Data as Documents)

MongoDB stores data as documents in a binary representation called BSON (Binary JSON)[10]. Documents that share a similar structure are typically organized as collections. You can think of collections as being analogous to a table in a relational database:

documents are similar to rows, and fields are similar to columns.

MongoDB documents tend to have all data for a given record in a single document, whereas in a relational database information for a given record is usually spread across many tables.

For example, consider the data model for a blogging application. In a relational database, the data model would comprise multiple tables such as Categories, Tags, Users, Comments and Articles. In MongoDB the data could be modeled as two collections, one for users, and the other for articles. In each blog document there might be multiple comments, multiple tags, and multiple categories, each expressed as an embedded array.

"Data as documents: simpler for developers, faster for users."

As a result of the document model, data in MongoDB is more localized, which dramatically reduces the need to JOIN separate tables. The result is dramatically higher performance and scalability across commodity hardware as a single read to the database can retrieve the entire document.

Unlike many NoSQL databases, users don't need to give up JOINs entirely. For additional analytics flexibility, MongoDB preserves left-outer JOIN semantics with the $lookup operator, enabling users to get the best of both relational and non-relational data modeling.

In addition, MongoDB documents are more closely aligned to the structure of objects in the programming language. This makes it simpler and faster for developers to model how data in the application will map to data stored in the database.

III. EXISTING SYSTEM

In the Existing system, The conventional database which is table structured DB like MYSQL is a popular open-source relational database management system (RDBMS) that is developed, distributed and supported by Oracle Corporation. Like other relational systems, MySQL stores data in tables and uses structured query language (SQL) for database access. In MySQL, you

pre-define your database schema based on your requirements and set up rules to govern the relationships between fields in your tables. In MySQL, related information may be stored in separate tables, but associated through the use of joins. In this way, data duplication is minimized.

3.1.1 DRAWBACKS IN EXISTING SYSTEM
RICH DATA MODEL AND DYNAMIC SCHEMA

Data models like json will be able to store as it is in the mongo db , but those can't be stored in the table structured database.

TIME AND COSE

Maintain the data severs are really costly when you compared with the mongodb and time to develop and maintaining requires a lot of time.

PROPOSED SYSTEM

In The Proposed System, MongoDB is an open-source database developed by MongoDB, Inc. MongoDB stores data in JSON-like documents that can vary in structure. Related information is stored together for fast query access through the MongoDB query language. MongoDB uses dynamic schemas, meaning that you can create records without first defining the structure, such as the fields or the types of their values. You can change the structure of records (which we call documents) simply by adding new fields or deleting existing ones. This data model give you the ability to represent hierarchical relationships, to store arrays, and other more complex structures easily. Documents in a collection need not have an identical set of fields and denormalization of data is common. MongoDB was also designed with high availability and scalability in mind, and includes out-of-the-box replication and auto-sharding.

ADVANTAGES

In our proposed aims at the scalability and performance of the conventional db can be improved by using the latest technologies like MongoDB and AngularJs together.
✓ Store the data as document like json formatted document. Hence adding a column doesn't need much work .it'll take the extra field if it is present.

- ✓ Files can be stored in a appropriate manner like it split the files into number of small chunks and store it in different chunk boxes. It'll helps to retrieve the files very soon when it is requested.

IV.SYSTEM MODULES

4.1 MODULES

- Developing a JAR for XML parsing
- Data Storage in MongoDB from JAR
- MongoDB setup
- Developing web Application using AngularJS
- Accessing the data from Mongodb using AngularJS
- GUI designing to showcase the fetched data

4.1.1 DEVELOPING A JAR FOR XML PARSING

In real life scenario the exacts data's will be in a xml representation. Hence presumed like that and proceeding with xml data and executable JAR will be created to get the xml from a location and it will parse the xml using document libraries which is a default one provided by the java. Parsing in the sense getting all the values related to DB and saving it in java bean , so that it can be converted to the DB object and it can inserted into the db directly.

4.1.2DATA STORAGE IN MONGODB FROM JAR

In this module, Mongodb connection will be created with the help of the mongodb libraries which is provided by the apache. Once it is created the parsed data which would've stored as a java bean. That relevant bean will be added in the mongodb object and some object related params would be set along with it . Then the data will be inserted into the mongo DB.
And also CRUD operation method and functionalities for the mongodb needs to be created in JAR . so that data can be inserted, deleted, updated and created again if we want.

V. CONCLUSION

Mongo DB will be the best option for the light weight application. Having a NoSql structure database which is document based database will be really helpful for the developers when they make the CRUD operation. And also we're using Angular Js for this project which really plays a tremendous role in this project. Since it is JavaScript which will be used to get the data from the MongoDB directly. Instead sending request to the backend server code and querying DB for the data is a time consuming process. But here Angular js simply uses a framework called mongoose and it will be connecting the database from the front end itself and it'll query for the data. Hence the application will be very light weight one and it won't be a much time consuming one when compared to the the SQL DB . So using MongoDB for storing the data and using Angular JS to fetch the data will leads to increase the scalability and performance of the application.

VI. REFERENCES

[1]. Alex Vanston(2012) ,'Complete Book on AngularJS ',Vol.4,pp.4593-4607.
[2]. Adam Freeman(2014) ,'Pro Angular JS(Experts Voice in Web application Developmetn)'In Proc. of ACM CCS.
[3]. Brad Green , Piva A, and Barni M (2014),' Angular JS ', vol. 4, No. 1 ,pp.477-887.
[4]. Brad Dayley (2013),' Node Js , Angular JS, MongoDB web application Development', Vol. 10, No. 4, pp. 12-21.
[5]. Elco Plugge, Griffith.R, Joseph A,.D, Katz R.H, Konwinski .A, Lee .G, Patterson D.A 2013, 'The Definitive guide to MongoDB: The NoSQL Database for cloud and Desktop Computing'. USB-EECS-pp.259-728.