MongoDB Application

Database

We have already acknowledge the issue face by the Nexart organization. In order to overcome the issue, the data engineers of Nexart change the architecture of the database and they moved towards a non relational database such as MongoDB. The data engineers understood that they have to do some alterations to meet the demands of the growing business. So they decided to migrate their data to MongoDB. The command given below creates a database in MongoDB database.

The creation of nexart database can be evaluated by executing the command given below.

In prior to the migration, the database engineers map entities and their attributes of the MSSQL database to MongoDB database. This process is discussed further in the collections section.

Collections

According to concepts of relational databases, all the entities are defined as tables. These tables were converted into collections in the MongoDB database. The tables in the MSSQL database inherited its own columns. These columns were mapped to the fields of the corresponding collections. Each table of the MSSQL database consists of 100 records per each. These records are mapped to documents in MongoDB database. According to this scenario clients, branches, products, orders and stock requests represents tables in relational databases while they are declared as collections in non-relational databases. The figure given below demonstrates the commands that should be executed to create the mentioned collections above.

The created collections can be verified by executing the command given below.

The next segment discuss about application of MongoShell in regards to the development of Nexart’s web application.

Mongo Shell

Before migrating data into the database, a database and tables should be created. While creating the tables primary keys, foreign keys and composite primary keys should be defined. So that it enables to map the relations between the entities. The database administrators in the past used a command line interface to interact with the application. This applies to the database management systems in the past as well. Before graphical user interface of MSSQL came to light, the administrators used MSSQL cmd to create databases, tables etc. The Shell of MongoDB performs the same role as the CMD of MSSQL. When creating tables they were required to provide a structure for the tables. Since MongoDB handles semi-structured and unstructured data well, there is no need to define the table structure. Even though the database, collections and column names should be named properly. So that it would help the software developers of IT department to populate the data of the database in the Nexart web application. Queries are inserted and executed in Mongo Shell to perform all the database operations. As the collection creation phase is completed, the migration process remains. Data experts of Nexart exported the records in the MSSQL database to json files. The list below depicts the data that are extracted from the tables.

Since the mongoDB documents are represented in BSON format, mongoDB is compatible with handling json files as well. In the next phase all json files are imported into the relevant collections of the Nexart database in MongoDB server using mongoimport tool. The progress of the data imports are given below.

Next the data engineers wanted to make sure wither the data is imported into the relevant collections.By executing the commands given below they verified import process of each collection.

Hence the records displayed at screen are limited to 20 per each iteration, they got the document statics of each collection by executing the commands given below.

There are differences between the query structures of relational databases and non-relational databases. With the growth of data, the database queries got complex and some admins were bored of typing same query again and again. In addition it took sometime to insert the query and human errors such as typos occurred as well. In addition only highly technical professionals of Nexart understood SQL/NoSQL while for other users of Nexart, didn’t comprehend SQL/NOSQL. So the developers of MSSQL came with a solution to address these issues.

Mongo Compass

The previous chapter brought up an issue regarding the usability of MSSQL CMD/Mongo Shell. As a remedy for this issue, MSSQL developed a graphical user interface along with a CMD. This resulted in less errors and query response times were improved. Even the the query is entered correctly, an admin can mistaking enter a wrong value in the field and make the data inefficient. So the admin is solely responsible for the validity of data. MongoDB Compass functions similarly as the MSSQL GUI. As a result of the development of mongoCompass, technical people who have an average knowledge about databases of Nexart do not need to worry entering data into the database. The image given below depicts the viewing of created Nexart database.

Earlier branch, client, product, order and stock request collections were created in the Nexart database. The image below shows the created collections visually in Mongo Compass.

The data engineers imported data into the collections. All imported data can be verified visually by the list given below.

People with average technical knowledge can enter data into the database via mongoDB compass’s user friendly graphical interface. Even though this graphical user interface had a limitation and it is addressed in the next chapter.