

SQL & NoSQL Database Study for Internet of Things

Sharvari Rautmare, Dr. D. M. Bhalerao

Student, Department of Electronics and Telecommunication, Sinhgad College of Engineering, Pune, India

Department of Electronics and Telecommunication, Sinhgad College of Engineering, Pune, India

ABSTRACT: Internet of Things (IoT) is the new emerging technology. With the rapid developments in this technology, in near future, IoT applications are going to produce tremendous amount of data. Most of the IoT research work is inclined towards manufacturing low-cost and energy-efficient hardware, as well as the communication technologies that provide thing-to-thing connectivity. However, the solutions to manage and utilize the massive volume of data produced by these objects are yet to mature. Traditional database systems used Relational database management systems (RDBMS). RDBMS has till now supported all the user requirements along with simplicity, robustness, flexibility, scalability, performance. But the main limitation they are facing is their static schema which is making RDBMS not suitable for IoT applications. On the other hand, NoSQL databases emerging in market have claimed to perform better than SQL database. The NoSQL databases are non-relational, schema free, no joins, easy replication support, horizontally scalable, etc. So, which database will be suitable for a particular IOT application is of our interest. Hence, for an IoT application, comparison study is done for both SQL & NoSQL database technologies.

KEYWORDS: Internet of Things (IoT), Database, SQL, NoSQL, RDBMS

I. INTRODUCTION

Internet of Things (IoT) is currently in developing stage. IoT refers to any thing around us which is connected to internet and is able to communicate. Usually sensors are attached to the things like temperature sensor connected to oven, fridge or just to measure a room temperature. IoT is network of such things which may communicate with each other or with user. IoT is having application in various domains like smart city, industrial, medical services, etc. As, the application area is wide, the type of data generated by these objects is heterogeneous. It is said that, data generated by these IoT applications will be big data which can be further analyzed to make the optimum use of resources. Also, the use of IoT applications will increase the real-time data, which all together makes the information storage and accessing more difficult and challenging. Several new technologies and database management systems (DBMS) are emerging to handle the data growth and to improve the system performance. Even though SQL databases have been the most popular database systems so far, question arises whether the traditional relational databases fit well with all the new system and performance requirements. A new class of database referred to as NoSQL databases are emerging in market. NoSQL databases store data very differently from the traditional relational database systems. They are meant for data of schema-free structure, can be easily distributed providing high scalability and availability. These properties are actually needed to realize the vision behind Internet of Things data.

II. LITERATURE REVIEW

Authors [1] proposed the design, implementation and evaluation of three data retrieval approaches for Multi-Sensor Passive RFID Tags. These approaches were designed based on the Data-in-ID approach. Alternate, Sequential & parallel sensor sampling scheduling schemes proposed were compared with parameters time delay, power consumption, communication range & read rate. Limitations to this study were the limited data and limited number of sensors used. In another work, application specific research is done for IoT domain. Internet of things is heterogeneous system having wide application domains. Application oriented research is being done as application demands are different.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

BoyiXu, Li Da Xu, Senior Member, IEEE, HongmingCai, Cheng Xie, Jingyuan Hu, and Fenglin Bu [2] have proposed ubiquitous data accessing method for IoT based Information System for emergency medical services. The study concluded that the Method of Ubiquitous data accessing is effective in a distributed heterogeneous data environment for supporting data accessing timely and ubiquitously in a cloud and mobile computing platform. In UDA-IoT model, heterogeneous IoT data are encapsulated in unified format of resources with unique URI so as to be accessed ubiquitously. The UDA-IoT is significant to support decision-making in emergency medical services. Similar to this study, application specific studies are being done, because of heterogeneous format of data in each and every application of IOT. For this heterogeneous data generated by IoT applications, advanced database systems are required to support the system. The main function of data acquisition, Data storage and data analysis are managed by the Database management system. Advances in the DBS are discussed by authors in their study [3]. Advanced Database Systems such as the parallel DBS, the mobile DBS, the multimedia DBS and distributed DBS are explained which can be beneficial for the development of IoT. Comparison of various database systems SQL & NoSQL [4] is done for modest-sized structured data. Results were based on the runtime of both the databases. Similar such comparative study [5] between NoSQL and SQL database is done with structured and unstructured data. Here, author has concluded that it is difficult to state the supremacy of either database system without considering the specific application of data definition and data querying. New advanced NoSQL database claims to be better suited for the heterogeneous IoT generated data. The question arises whether to go for this new technology or to use the more reliable traditional relational database system. In this paper, comparison study of SQL and NoSQL database is done.

III. DATABASE MANAGEMENT SYSTEMS

Database is defined as organized collections of data. Although when using the term database we refer to the entire database system, the term actually refers only to the collection and the data. The system which handles the data, transactions, problems or any other aspect of the database is the Database Management System (DBMS). Traditional method was relational database and the new advanced one is non-relational database which are SQL and NoSQL respectively. Both these databases are equal potential competitors of each other with both having several pros and cons. This study is to find out the suitable database for IoT applications. Some basic characteristics of database are given by the CAP theorem which explains Consistency, Availability and Partitioning:

- Consistency: means that once an update operation is finished, everyone can read that latest version of the data from the database and such system is consistent system. While, a system in which, updated data can not be seen by all users at once is known as eventually consistent.
- Availability: is achieved if the system always provides continuous operation, normally achieved by deploying the database as a cluster of nodes, using replication or partitioning data across multiple nodes so if one node crashes, the other nodes can still continue to work.
- Partition Tolerance: A database system which can operate even if one node of its nodes fails or is inaccessible. This is done by redirecting all queries to the failed node to some other active node of that system.

The traditional RDBMS focused on consistency and supports following ACID properties:

- Atomicity: the transactions are either fully successful or unsuccessful. Partially accomplished transactions are discarded.
- Consistency: If a transaction fails, system reverts the transaction and goes back to the previous stable state. So, system remains stable before and also after the transaction.
- Isolation: Transactions are performed without any interference and are processed independently.
- Durability: All the committed transactions are saved in logs and will not be lost. This helps to recover the system in case of abnormal terminations.

Traditional database system i.e. SQL, focused on the consistency characteristic. But, there are applications which demand availability and partition tolerance over consistency. For such applications, instead of ACID properties, BASE properties are implemented: Basic Availability, Soft state, Eventual Consistency. The system following BASE properties is not strictly consistent. But, all the updates or modifications will be available eventually. In such systems, clients may encounter an inconsistent data as updates are in progress (during replication process), the data will eventually reach the expected consistent state. Even though relational databases have been considered the classic kind of databases for years, NoSQL databases have been using the CAP theorem as an argument against the traditional ones.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

As system scale is getting larger, it is difficult to leave out the partition tolerance. In the end, the goal is to find the best combination of consistency and availability to optimize specific applications. The point of NoSQL databases is to focus on availability first then consistency, while SQL databases follow the reverse order.

IV. SQL AND NOSQL DATABASE

SQL Database: Traditional database systems for storage have been SQL database which follows the relational model. In the relational model, data are stored in the tables which are interrelated. Indexing mechanism supported by SQL is used to speed up various operations like reading or creating views which can join data from multiple tables. There are many relational databases available such as MySQL, Oracle and SQLServer. On the other hand, NoSQL is non-relational database management system which supports flexible schema to store the data. It is definitely different than traditional relational database system which provides structured schema in form of tables. NoSQL databases supports large scale of data storing by providing distributed data stores. Hence, these NoSQL databases are supposed to be suitable for storing the data generated by IoT applications. These NoSQL databases grew in popularity because of the ease of access, speed, and high scalability, distributed computing. Most of the NoSQL databases are based on storing key value pairs. Commonly used NoSQL database categories are:

- Document Store: In this type, database is nothing but collection of documents. Each document contains multiple name fields and unique document ID. Documents can be stored in specific formats such as XML, JSON and BSON. It is the most popular type of NoSQL database as it can avoid complex join operations, rich set of query capabilities and high scalability.
- Key-value stores: Data is stored in pair of key & value. The value is block of data & Key is used to access the data. It is easy to build, provides easy scalability and gives good performance.
- Graph databases: Nodes and edges are the primary concept in graph databases. Nodes correspond to entities such as a user or a sensor and edges represent the relations between the nodes.

Table: Differences in SQL and NoSQL

	SQL	NoSQL
Model	Relational database system	Non-relational database system
Data-storage	Stores data in tables having rows and columns	Stores data in different forms like document, Column stores, Graphs, Key-value pairs
Schema	SQL data has fixed schema and changing it is extremely hard and mostly avoided	NoSQL is schema-free database and so it allows user to change the data schema easily
Data structure	Suitable for structured data	More suited for semi-structured or unstructured data
New fields	Adding new fields in the table may require altering the schema	New fields can be added with much more ease
Scalability	Scales well vertically	More suited for horizontal scaling
Querying language	Use powerful declarative language for querying	Use JSON data objects for querying
Transactions	Supports ACID transactions	ACID transactions support varies per solution
Consistency	Strong consistency	Eventual consistency

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

- A. **Data Model:** SQL is a relational database where data is stored in form of rows and columns in tables. The database design contains number of such tables linked together. In the table, columns represent different fields while rows are the records. This data model gives well organized structure of data but the design is rigid. Same table cannot be used to store different kind of information or insert a number where string is expected. On the other hand, NoSQL is non-relational database which uses data modelling types like document store, key-value, graph. Similar documents can be stored in a collection, which is analogous to SQL table. However, one can store any type of data in any document. SQL tables create a strict data structure/template, which decreases probability of user making mistakes. On the other hand, NoSQL is more flexible and forgiving, and is able store any type of data which can lead to consistency issue. But, this feature of storing any type of data in a document in NoSQL proves efficient in handling unstructured data like word, pdf, multimedia files. In IoT applications like Vehicular communication, Healthcare management where data structure cannot be predefined properly or data is heterogeneous, NoSQL definitely has the higher edge. But for simpler application fixed data structure will be beneficial to make data accessing faster.
- B. **Scalability:** Vertical scalability is an ability to scale up or to add resources to a single node in a system, typically involving the addition of CPUs or memory to a single computer. Application scalability refers to the improved performance of running applications on a scaled-up version of the system Traditional databases (SQL) were designed primarily to scale vertically by adding more power to a single node. Horizontal scalability is an ability to scale out or to add more nodes (servers) to the system so that system load can be shared. Size scalability is the maximum number of processors that a system can accommodate. Horizontal scalability is one of the main characteristics of NoSQL. For future expansions in any IoT application, where distribution of data amongst several servers is requires, use of NoSQL will be an better option. Practically, this allows expansion of hardware system in a stepwise manner rather than investing in high end module of hardware in beginning.
- C. **Data Retrieval:** To lookup data in SQL databases, user need to collect it from interlinked tables using JOIN statements and combine the results. Same process has to follow while writing. In NoSQL, each piece of data represents an object which contains all its related data especially in document store, which allows much faster read, write operations. This feature is mainly required while analyzing the data stored in the database or to take any action on specific record.
- D. **Transactions:** Relational databases including SQL support ACID properties and hence provide transaction reliability while NoSQL support BASE properties which are discussed above. NoSQL databases focus more on eventual consistency and provide availability and are more partition tolerant. For IoT applications which are in developing stage, data generation is going to increase day by day. So, ideally, database system which is partition tolerant is more suitable. Also, in IoT application where data is collected first and then analyzed later on, database can be eventually consistent. In this case, strong consistency is not a mandatory characteristic. Strong consistency will prove better suited in application scenarios like online reservation system, inventory system, retail store, banking where immediate updation of data is must.
- E. **Maturity of system:** Finally, one of the important issues is security and system problems. A NoSQL database system is a recent technology and hence is less mature as compared to SQL. Hence, NoSQL is more likely to generate new issues. Many problems arise with NoSQL, but most are just due to lack of knowledge. Developers have less experience with newer database systems, so mistakes are made. Also, using NoSQL just because it is a new technology, or because to avoid schema design inevitably, may lead to problems later. On the other hand, in SQL, most of the issues are all addressed as it is being used since several years now. Security concerns like authentication, data integrity, confidentiality, auditing and client communication are addressed in SQL while in NoSQL, they are yet to be integrated. In IoT, sensors are going to transfer a lot of sensitive information. So a secured channel for information exchange and a secure system to store this sensitive data is must. For application where data security and integrity is of high priority, better to go for a secured data storage.

International Journal of Innovative Research in Science, Engineering and Technology

(An ISO 3297: 2007 Certified Organization)

Vol. 5, Issue 5, May 2016

V. CONCLUSION

This paper compares two types of database systems namely SQL and NoSQL. SQL is a traditional relational database system which is being used since many years now and hence is more mature compared to NoSQL which is a non-relational database system. SQL has many advantages such as it is easy to use and setup, structured schema helps to avoid mistakes which may arise in case of flexible schema, strong consistency, gives secured storage. But, for IoT applications where data is going to be huge and is heterogeneous, it is not a suitable option. Whereas, NoSQL is capable of storing any type of data, is capable of handling the big data, also supports horizontal scalability, runs well on cloud platforms. But this NoSQL technology is still maturing and lacks security standards. Both the systems have pros and cons which makes the decision of choosing one system over other more complicated. A better performance can be achieved using either of the system if one can choose the database according to the needs of the application. Applications where sensor data is of one specific form or sensor is sending some sensitive data in such cases SQL should be the ideal choice. But in projects like Smart City where data requirements are unrelated, indeterminate or evolving, or in projects where future expansion is expected, use of NoSQL will be better option. To evaluate which database system performs better, research work must be application specific. Only then true results can be fetched. Future scope for this study can be focusing more on security vulnerabilities in NoSQL, analyze the techniques used in both systems to make data retrieval faster, calculating the response time of queries on the both databases and then comparing the results.

REFERENCES

- [1] Yang Su, Asanga Wickramasinghe, Damith C Ranasinghe, "Investigating Sensor Data Retrieval Schemes for Multi-Sensor Passive RFID Tags", Auto-ID Lab, The University of Adelaide, Adelaide SA 5005, Australia, 2015 IEEE International Conference on RFID
- [2] Boyi Xu, Li Da Xu, Senior Member, IEEE, Hongming Cai, Cheng Xie, Jingyuan Hu, and Fenglin Bu, "Ubiquitous Data Accessing Method in IoT-Based Information System for Emergency Medical Services", IEEE Transactions on Industrial Informatics, Vol. 10, No. 2, May 2014
- [3] Ion LUNGU, Manole VELICANU, Iuliana BOTHA, "Database Systems – Present and Future", Economic Informatics Department, Academy of Economic Studies, Bucharest, Romania, vol. 13, no. 1/2009
- [4] Zachary Parker, Scott Poe, Susan V. Vrbsky, "Comparing NoSQL MongoDB to an SQL DB"
- [5] Hammes, Dayne; Medero, Hiram; and Mitchell, Harrison, "Comparison of NoSQL and SQL Databases in the Cloud" (2014). SAIS 2014 Proceedings. Paper 12.