NoSQL

IIIT-DWD

INDIAN INSTITUTE OF INFORMATION TECHNOLOGY

3rd floor, IT Park, Hubballi- Dharwad, Karnataka 580029

November 2019

Name: - Rajendra Prajapat Roll No: - 17BCS024

Under the supervision of Dr. Uma Seshadri

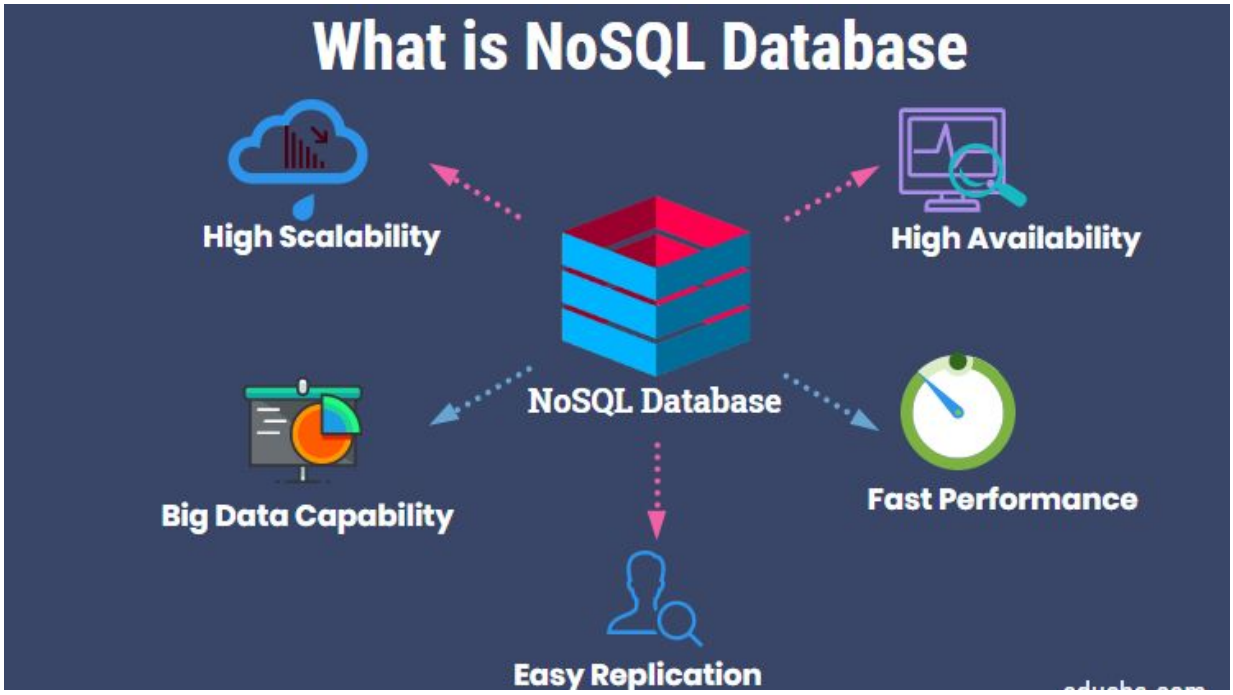
1. Introduction

1. Scope of the Topic:

Today's web, mobile, and IoT applications need to:

- Support large numbers of concurrent users
- Deliver highly responsive experiences to a globally distributed base of users
- No downtime
- Handle semi and unstructured data
- Rapidly adapt to changing requirements with frequent updates and new features

The new enterprise technology architecture needs to be far more agile than ever before and requires an approach to real-time data management that can accommodate unprecedented levels of scale, speed, and data variability. Relational databases are unable to meet these new requirements, and enterprises are therefore turning to NoSQL database technology. NoSQL databases are widely used in many famous enterprises such as Google, Yahoo, Facebook, Twitter, Taobao, Amazon, and so on.



There are more than 225 kinds of NoSQL databases.So, the problem is, how to choose an appropriate NoSQL database for a specific enterprise based on their requirements, is very important because the change of database may affect the enterprise performance of the business

operations. Paper one introduces basic concepts, compares the data formats and features, and lists the actual products for every category of NoSQL databases.

NoSQL aims to solve the needs of high concurrent read-write, efficient mass data storage and access, database scalability and high availability. In these large-scale concurrent systems, cluster cache and data consistency become the focus of attention. Paper two proposes a fast and effective approach to estimate the solution of data consistency.

paper 3, first, describes the physical database tuning of the Oracle Relational database and then compared it with the NoSQL Graph database. Relational database performance is increased by up to 50% due to the physical database tuning technique (Tablespaces). Besides, the physical database tuning approach of relational database NoSQL graph database performed better in all our proposed scenarios.

2. Literature Study:

- Users of SQL databases are required to handle large amounts of varied nature of data by increasing the system capacity like CPU, RAM, etc. Relational databases are vertically scalable databases and can process the data at a certain limit. NoSQL databases are horizontally scalable databases and can process huge amounts of data efficiently. Trustful transactions, better data integrity, and security are the main focus and strength of the relational databases. The BASE approach is used by the NoSQL databases and can easily manage various forms of massive data. Handling big data through NoSQL databases required lower cost and minimum overhead. The new server can be added easily in the cluster environment due to the horizontally scalable characteristics of NoSQL databases. In the cluster commodity hardware is used to store big data.
- Eric Brewer presented the CAP theorem and stand for Consistency, Availability and Partition Tolerance. Amazon-like large companies implemented and adopted the CAP theorem. In CAP, consistency describes how to bring the system in a consistent state after performing some writes operations by the system. Availability describes, after performing the writes operations, the updated data must always be highly available to the users. Partition tolerance describes if the data is spread and distributed over the various nodes in a network the system must respond and be able to continue its operations. The C (Consistency) and P (Partition Tolerance) of the CAP theorem are used by the SQL databases. NoSQL databases use the A (Availability) and P (Partition Tolerance) of the CAP theorem. A and P of the CAP theorem is used by the Amazon Dynamo.
- The rigid schema structure of RDBMS with the passage of time becomes more mature. In the mature rigid schema, to bring any changes if necessary is very hard but possible. NoSQL databases do not follow the rigid schema structure. The schema of NoSQL databases is flexible and developed gradually. NoSQL databases can also handle the problems of the NULL value of stored data in RDBMS.

- Z. Parker, S. Poe, S. V. Vrbsky Compared the performance of Microsoft SQL Server relational database and MongoDB NoSQL database. MongoDB performed well than the SQL server in a simple query, insert and update. SQL server performed better for aggregate queries and queries with secondary attributes.

2. New proposed Solutions - Research papers

2.1 Paper 1: A study of NoSQL Database for enterprises

- Year of publication: 2018
- The Journal name: IEEE

THE CATEGORIES OF NoSQL DATABASES:

A. Wide Column Store

This NoSQL database has a complex table schema described as follows:

a. A Row Key is an identification that has a unique value used to identify a specific record, similar to the primary key of relation in RDB.

b. A Timestamp (abbreviated as ts) is an integer used to identify a specific version of a data value.

Table Name: Products_Inventory

Row Key	ts	Column family Products	Column family Inventory
P001	t1	Products:classes = "TV"	
	t2	Products:title = "LG 55 inch 4K LED TV"	
	t3	Products:descriptions = "TBD"	
	t4	Products:price = "27000"	
	t5		Inventory:quantity = "10"
	t6		Inventory:place = "1A"
P002	t7	Products:classes = "Laptop"	
	t8	Products:title = "ASUS FX503VD i7 gaming laptop"	
	t9	Products:descriptions = "TBD"	
	t10	Products:price = "32000"	
	t11		Inventory:quantity = "20"
	t12		Inventory:place = "2A"

c. At least one Column Families that have the format of "Family: Qualifier = Value," where "Family" is the name of a Column Family, "Qualifier" is the name of a Column Qualifier, and "Value" is a real value of a Column Qualifier stored in text.

d. According to the statistics of the DB-Engines Ranking Website, Apache Cassandra and Apache HBase are more widely discussed ones of Wide Column Store databases.

B. Document Store:

a. This NoSQL database stores data with files of semi-structured documents which have specific formats such as XML (eXtensible Markup Language) or JSON (JavaScript Object Notation)

b. According to the statistics of the DB-Engines Ranking Website, MongoDB and Couchbase are more widely discussed ones of Document Store databases.

```
[
   {
      "c_no": "C001",
      "title": "management",
      "credits": 3,
      "instructor": "Amy"
   },
   {
      "c_no": "C002",
      "title": "economics",
      "credits": 3,
      "instructor": "Ben"
   }
]
```

Fig. 1. An example of a data file in Document Store.

C. Key-Value Store:

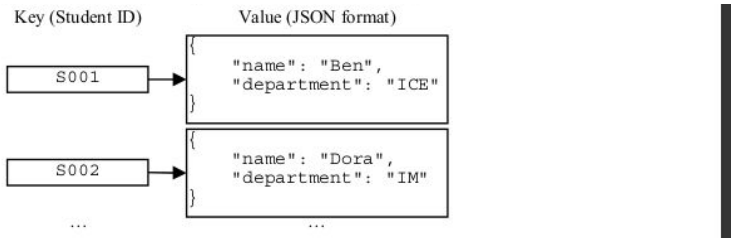
a. The data in this NoSQL database is stored with the format of "Key -> Value" where

   i. "Key" is a string used to identify the unique "Value;"

   ii. "Value" is the real data which can be number, string, or a JSON file, etc.;

   iii. The user can search for a certain "Value" by a specific "Key."

b. According to the statistics of the DB-Engines Ranking Website, both Redis and DynamoDB are more widely discussed ones of Key-Value Store databases.

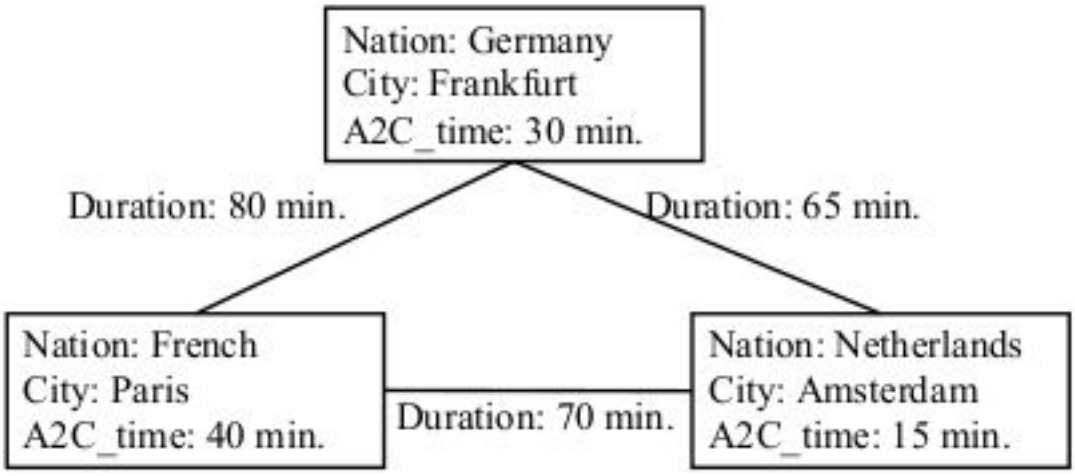


D. Graph Databases:

a. This NoSQL database stores data based on a graphic structure.

b. According to the statistics of the DB-Engines Ranking Website, Neo4J and OrientDB are more widely discussed ones of Graph Database databases.

c. ArangoDB stores graph by holding edges and nodes in separate collections of documents. A node would be represented as any other document store, but edges that link two different nodes hold special attributes inside its document; a _from and _to attributes.



E. MultiModel Databases

a. The data format of this NoSQL database contains more than two data formats of the other categories of NoSQL databases
b. According to the statistics of the DB-Engines Ranking Website, OrientDB and ArangoDB are more widely discussed ones of Multimodel Databases databases.
c. OrientDB contains the data formats of Object Database, Document Store, Graph Database, and Key-Value Store.
d. ArangoDB contains the data formats of Document Store, Graph Database, and Key-Value Store

F. Object Databases:

a. This NoSQL database combines the functions of object-oriented programming languages and traditional databases.
b. each rectangle is an object that includes both data items and data processing functions.
c. According to the statistics of the DB-Engines Ranking Website, db4o and Versant are more widely discussed ones of Object Databases databases.

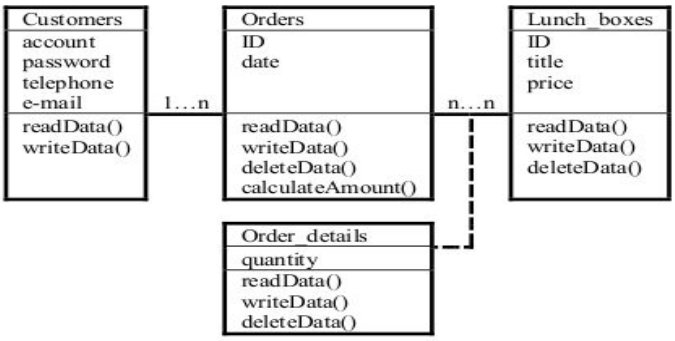


Fig. 4. An example of a Class Diagram in Object Databases.

G. Grid & Cloud Database Solutions:

a. This NoSQL database stores recent access data in Random Access Memory (RAM) and uses Grid Computing to speed up the time of access data from a database.
b. According to the statistics of the DB-Engines Ranking Website, Hazelcast and Oracle Coherence are more widely discussed ones of Grid & Cloud Database Solutions databases.

H. XML Databases:

a. The files stored in this category of NoSQL databases are based on the XML format.
b. According to the statistics of the DB-Engines Ranking Website [13], Oracle Berkeley DB and BaseX are more widely discussed ones of XML databases

```
<?xml version="1.0" encoding="UTF-8"?>
<Courses>
   <Course c_no="C001">
        <title>AI</title>
        <credits>3</credits>
        <remark>Teaching in English</remark>
   </Course>
   <Course c_no="C002">
```

I. Multidimensional Databases:

a. The data in this NoSQL database is stored in a multidimensional array in order to analyze the value of each array element.
b. According to the statistics of the DB-Engines Ranking Website, Intersystems Cache and GT.M are more widely discussed ones of Multidimensional Databases databases.

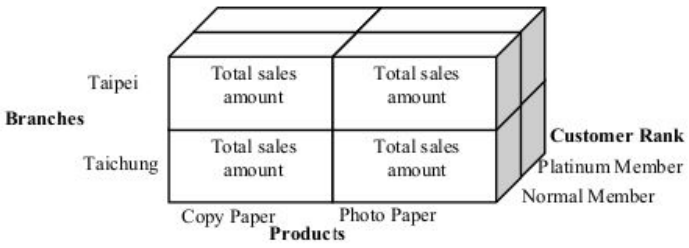


Fig. 6. An example of a three-dimensional array in Multidimensional Databases.

J. Multivalue Databases:

TABLE II
AN EXAMPLE OF A DATA TABLE IN MULTIVALUE DATABASES.

Table Name: Students

SID	Name		Society
	First_name	Last_name	
S001	Cindy	Lin	{Pop music,Choir}
S002	Eric	Wu	{Choir, Poetry}
S003	Peggy	Lu	{Computer, Guitar}

a. This NoSQL database is suitable for storing data of multivalued attributes or composite attributes.
b. According to the statistics of the DB-Engines Ranking Website, the jBASE and Model 204 Database are more widely discussed ones of Multivalue Databases databases.

K. Event Sourcing:

a. This NoSQL database is suitable for storing events that occurred in the past in order to track the status of a specific event.
b. According to the statistics of the DB-Engines Ranking Website, Event-Store is the most widely discussed database of Event Sourcing databases.

L. Time Series Databases (TSDBs):

a. This NoSQL database is designed to handle time-series data.
b. According to the statistics of the DB-Engines Ranking Website, Informix Time Series Solution and influx data are more widely discussed ones of TSDBs databases.

M. Scientific and Specialized DBs:

a. This category of NoSQL databases is designed to solve scientific and professional issues.
b. For example, BayesDB allows users who have not been statistically trained to solve basic science problems, and GPUdb is a database suitable for distributed computing.

N. Other NoSQL related databases:

a. The NoSQL databases in this category seem to be able to be categorized into several other categories mentioned earlier, but the official website of NoSQL database categorizes them into this special category without giving any explanation for the characteristics of this category of NoSQL databases.
b. Therefore, we have no way to know why this category is needed and the reasons why these NoSQL databases are assigned to this category.
c. According to the statistics of the DB-Engines Ranking Website, eXtremeDB is the most widely discussed database in this category of NoSQL databases.

The Principle of Database Selection:

If an enterprise prepares to choose a NoSQL database, it must understand the following questions according to the cultures and characteristics of the enterprise.

1. Understand the current problems, goals, and challenges of the corporate operations database.
2. Determine to continue to use current RDB or to change to use NoSQL databases according to business requirements and features of NoSQL databases.

3. If changing to use NoSQL databases, select a suitable category of NoSQL databases based on the features and formats of the enterprise's operating data.
4. When selecting a specific NoSQL database, we can first find out the NoSQL databases that are most frequently discussed on the Internet according to the statistics and evaluation of the DB-Engines Ranking website. Finally, based on the advantages and disadvantages of these databases and enterprise's needs, we can select the most appropriate NoSQL database.

2.2 Paper 2: Cache and Consistency in NoSQL

- year of publication: 2010
- The Journal name: IEEE

Section 1 presents cluster cache, it introduces a distribute cache model msmc in the non-relational database with MemCached and discusses how to read and modify data in order to maintain data consistency and here it references Google App Engine. Section 2 details the data consistency, it introduces a data consistency mechanism, Paxos, a very efficient transaction processing consistency algorithm, which can deal with the consistency management well.

CLUSTER LAYER CACHE

High-performance web applications generally use a distributed in-memory data cache in place of robust persistent storage for some tasks. Replication of the database among several servers removes the performance bottleneck of centralized database. The solution requires low-latency communication protocols between servers to implement update consistency protocols.

memcached brings the minimal impact to the internet. Fig. 1 shows an abstract view of the architecture of distributed system, which reducing the system load.

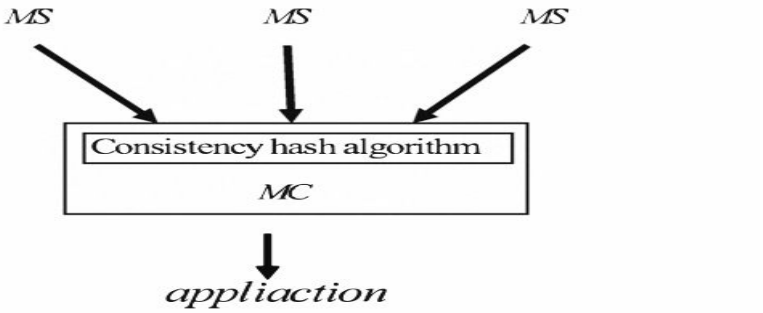


Figure 1. The architecture of distributed system

- Memcached has two main components: the server and client
- In a Memcached query, mc first calculated the hash value of the given key to determining the kV's location in the ms.
- When ms is confirmed, the client will send a query request to the corresponding ms.

- Then the ms will find the exact data. Because there is no interaction and multicast protocol among them, so the interaction of Memcached brings a minimal impact on the internet.

We can envisage that ms can also be divided, every ms can be treated as a physical node, within each physical node, there will be a variable number of virtual nodes running according to the available hardware capacity of the physical node. So the query content is divided into several parts, which distributed in different ms.

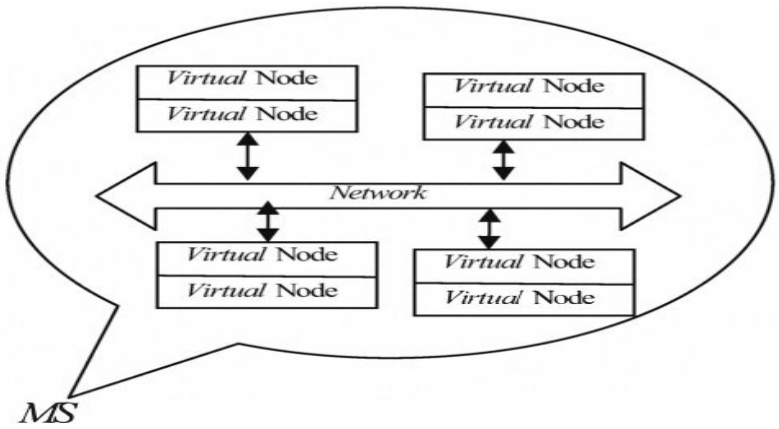


Figure 2. The ms-mc structure

Let's assume that our system is based on the masters-slaves system, there are a master server and a large number of slave servers. Each server includes an mc and ms, and we use the Memcache caching mechanism in each server. Each mc is a client when you need a query from a proxy server, and the proxy server you choose will check if the content you query is in the ms, and if the ms can't find the required content, and then turn the next proxy server. Some NoSQL databases use index and query processing mechanism in local database (node), and we can use a DHT indexing scheme that enables the proxy server to directly deal the cache query. In this case, we can use query processing program of the system to broadcast the query to all other nodes on the DHT, each node completes the local search, and then return the result to the system processor as a response. In DHT search is parallel in all nodes.

DATA CONSISTENCY:

A. Consistent Hashing

- As discussed in the previous paper, we know that there are many types of NoSQL databases, such as Wide Column/Column Families, Document Store, Key ValuelTuple Store, and Eventually Consistent Key-Value Store, etc; our data needs to partition and replicated using consistent hashing, and
- consistency is facilitated by object versioning, we need a suitable consistent hashing algorithm to ensure the consistency of replicas.
- Consistency among replicas is maintained by a quorum-like technique during updates and a decentralized replica synchronization protocol. the initial distributed architecture use

hash() mod n, but its weakness is that if a single-point failure, the system can not automatically recover.

- In order to solve the single point of failure, we use hash() mod (n/2), so anyone has two options which can be selected by the client, but this program still has problems: load imbalance, in particular, if one of the servers failures, the other server has excessive pressure.
- Inconsistent hashing, the output range of the hash function is treated as a fixed circular space (means the largest hash value comes around to the smallest hash value like 1,2,3,4,5,6,1,2...).
- Each node in the system is assigned a random ID within this space which represents its position on the ring.
- Each data item identified by the ID is assigned to a node by hashing using hash function the data item's key to yield its position on the ring, and then walking in the ring clockwise to find the first node with a position larger than the item's position.
- In Consistent Hashing, as mentioned above, we can use n servers and we can divide each server into several virtual nodes, each server can be seen as a physical node and all the virtual nodes connects to the ring.
- Then we assigned the virtual nodes to the consistory hash ring randomly, so the users through the hash algorithm to find the first v-node in the circle, and if the v-node failures, we can use the next v-node with a clockwise way.
- As Fig. shows that if the hash value located in v-node a, but v-node a crashed, then we will use v-node b. If all the v-nodes fail, then turn to the next v-node of the next physical-node.

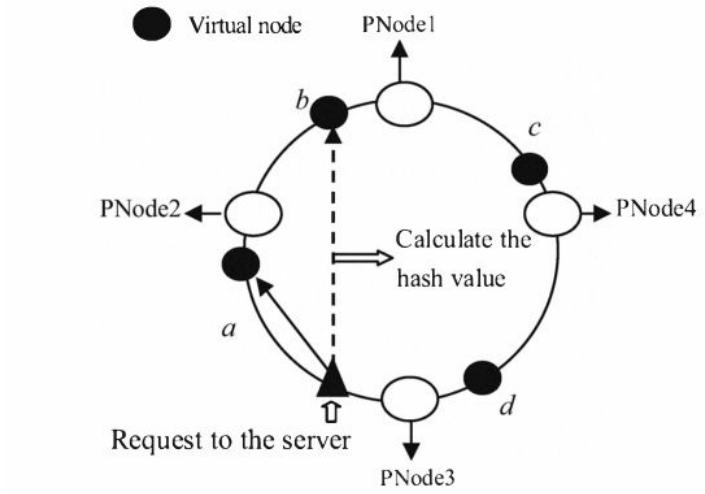


Figure 3. Example of hashing search

B. consistent hashing way: Paxos

- Every time, Paxos instance will be assigned a number which needs to increment, each replica doesn't accept the proposal which is smaller to the greatest number in the current.
- Once a value v is passed by a replica, then this value can not be changed, that have no Byzantine question. Take an example, If you plan to organize an event in college and If

the participant accepts the plan, he can't change his idea no matter who asks him to change. If most participants accept the plan, and the proposal can be adopted.

2.3 Paper 3: SQL Database with physical database tuning technique and NoSQL graph database comparisons

- year of publication: 2019
- The Journal name: IEEE

Physical database design:

- Physical design or schema tuning of any database management (DBMS) system consists of the process in which logical data has transformed into the physical structure. Frontier design is a method that presents the benefits of virtualization in the database environment.

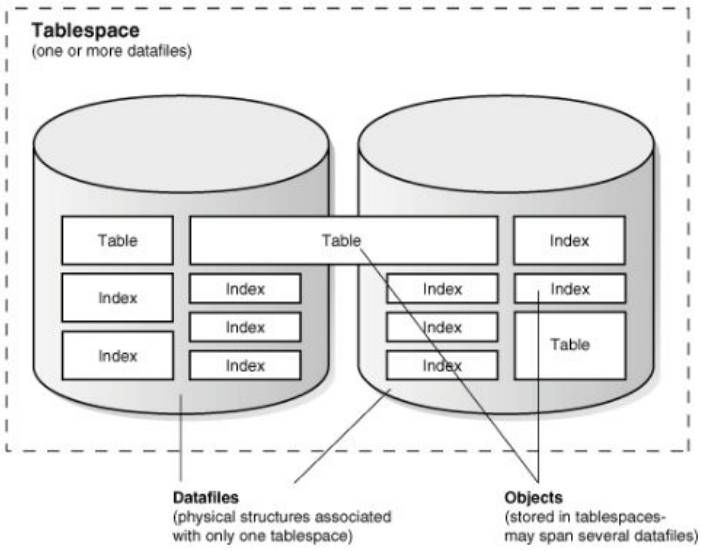


Fig. 2. Data files and Tablespaces.

Proposed Method For Comparison:

The proposed methodology consists of a SQL database (ORACLE) with a physical database tuning technique called tablespaces and NoSQL graph database. The Oracle relational database performance can be increased with the physical database tuning. We compared the same databases without the physical database tuning technique of the relational database. The NoSQL graph database performed better than the Oracle SQL database with its default configuration. By using the physical database tuning technique, we can increase the performance
of the Oracle relational database.

Experiment:

- Used the medical dataset for our desired experiments to evaluate and analyze the proposed research process flow.
- Used the Neo4j (3.03) community edition and Oracle 11g enterprise edition for our desired experiments.
- The healthcare case study Medicare schema consists of the following primary objects: Patient Table, Dependent Table, Patient Visit Table, Patient Issued Medicine table, Patient History table, Medical Staff table, and Patient Appointment table.
- The following properties are used by the objective benchmark:
    - Characteristics of Scalability
    - Requirements of Disk Space
    - Predefined Queries Set

Results and Evolution of experiment:

- In the Neo4j graph database performed better than Oracle a relational database for the first 5 queries.

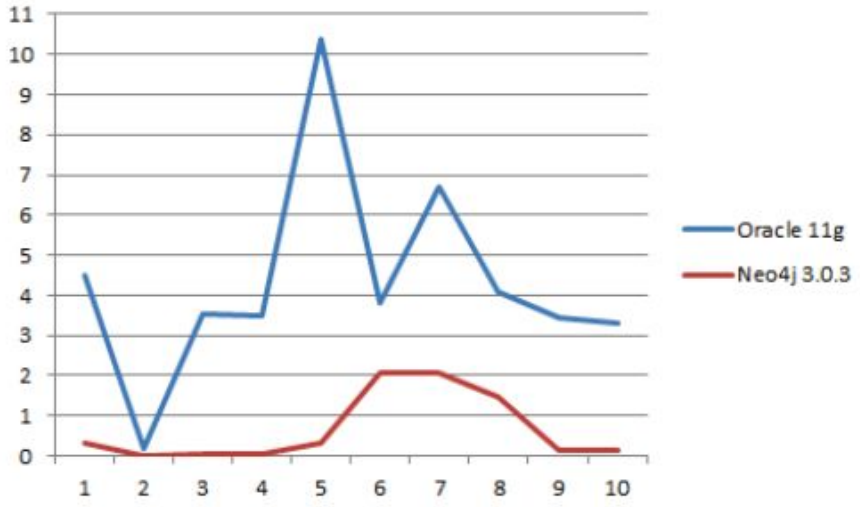


Fig. 9. SQL database without Physical database tuning technique and Neo4j. X-axis represents executed queries and Y-axis represents time in seconds taken by each executed query

-

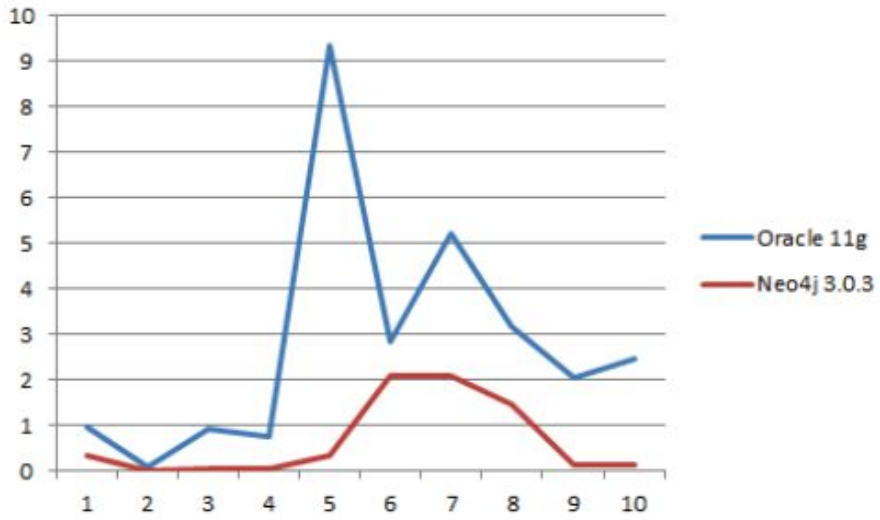


Fig. 10. SQL database with Physical database tuning technique and Neo4j. X-axis represents executed queries and Y-axis represents time in seconds taken by each executed query

3. Glossary/References

1. https://www.guru99.com/nosql-tutorial.html
2. https://neo4j.com/developer/graph-db-vs-nosql/
3. SOLID IT Team (2018) DB-Engines Ranking [Online]. Available: https://db-engines.com/en/ranking
4. https://www.couchbase.com/resources/why-nosql
5. https://en.wikipedia.org/wiki/Paxos_(computer_science)