



## Analysis of NoSQL and relational database performance

Akhil Tandon, Dr. Abhay Kumar Agrawal

Kamla Nehru Institute of Technology, Sultanpur, Uttar Pradesh, India

### Abstract

NoSQL data stores are widely used to store and retrieve possibly large amounts of data, typically in a key-value format. There are many NoSQL types with different performances, and thus it is important to compare them in terms of performance and verify how the performance is related to the database type. In this paper we provide comparison between the two leading type of Database storage components prevailing in the industry (one is relational database and the second is NOSQL database). The Database is largely concerned with managing massive amount of data in a consistent, stable, repeatable and quick manner. The prominent features of both relational as well as non relational databases have been specified which form the basis of the comparison between the two types of database. This comparison allows users to choose the most appropriate database according to the specific mechanisms and application needs.

**Keywords:** NoSQL, Database, stable, repeatable and quick manner

### 1. Introduction: Database Categories

There are two categories of Databases, which are Relational (uses SQL for performing queries) and the other one is Non Relational (NoSQL). The Relational Model is based on Mathematical Theory (set theory, relational theory) whereas the Non-Relational Databases may or may not have a single groundwork mathematical theory. Relational model is beneficial when it comes to reliability, flexibility, robustness, scalability requirements but in order to cater to the needs of modern applications Where the data is huge and generally unstructured. Non-relational databases show true signs of usability here and further categories in four types which we discussed in previous chapters. Our study comprises of studying various relational and non-relational databases for storing and retrieval of large amount of data. The attributes and features of both are listed in the following columns [14, 17, 19].

#### 1.1 NoSQL

NoSQL( Not only SQL) is a Non Relational Databases that do not store data in tables, the schema is not fixed and have very simple data model, they can handle unstructured data such as documents, email, multimedia and social media efficiently. In the previous chapter we classify the NoSQL databases. For our experiment we use CouchDB, which belongs to the category of Document Oriented Database [10, 12].

#### 1.2 Relational Database

A relational database is a collection of data items organized in formally described tables from which data can be accessed or reassembled in many different ways. Relational Database is a set of tables referred to as a relation with data category described in columns similar to spreadsheets. Each row contains a unique instance of data for the corresponding data category. While creating a relational database domain of

possible values along with constraints are applied to the data. It is the relation between the tables that makes it a 'relation' table [13].

### 2. Experimental Setup

- **Hardware Requirement-** Processor: Intel Core™ i5 2.50 GHz, RAM: 4GB, HDD: 500 GB
- **Software Requirement-** CouchDB Database and MYSQL, Windows 8

### 3. Performance of Different Database Category

#### 3.1 NoSQL (CouchDB)

The performance of couchDB was measured for four different commands, for performing the experiment above mentioned experimental setup was prepared [1].

##### 3.1.1 Insert Operation

When we perform the insertion operation in CouchDB we found that the time of insertion increase when we increase the number of records. The details are shown in Table 1.

**Table 1:** Performance of Insertion of Records in CouchDB

No of Records	Time of Execution in CouchDB (Milliseconds)
1 Record	00:00:00:005
100 Record	00:00:00:010
500 Record	00:00:00:025
1000 Record	00:00:00:045
5000 Record	00:00:00:200
10000 Record	00:00:00:600

##### 3.1.2 Delete Operation

When we perform the delete operation in CouchDB we found variation in the time of deletion of records when we increase the number of records. The details are shown in Table 2.

**Table 2:** Performance of Deletion of Records in CouchDB

No of Records	Time of Execution in CouchDB (Milliseconds)
1 Record	00:00:00:008
100 Record	00:00:00:006
500 Record	00:00:00:010
1000 Record	00:00:00:023
5000 Record	00:00:00:064
10000 Record	00:00:00:150

### 3.1.3 Select Operation

When we perform the selection of records operation in CouchDB we found that the time of selection increases when we increase the number of records. The details are shown in Table 3.

**Table 3:** Performance of Selection of Records in CouchDB

No of Records	Time of Execution in CouchDB (Milliseconds)
1 Record	00:00:00:007
100 Record	00:00:00:009
500 Record	00:00:00:025
1000 Record	00:00:00:040
5000 Record	00:00:00:220
10000 Record	00:00:00:300

### 3.1.4 Update Operation

When we perform the updating operation in CouchDB, we found that the time of updating the records increases when we increase the number of records. The details are shown in Table 4.

**Table 4:** Performance of Updation of Records in CouchDB

No of Records	Time of Execution in CouchDB (Milliseconds)
1 Record	00:00:00:007
100 Record	00:00:00:010
500 Record	00:00:00:063
1000 Record	00:00:00:050
5000 Record	00:00:00:260
10000 Record	00:00:00:500

## 3.2 Relational Database (MYSQL)

The performance of MYSQL database was measured for four different commands, for performing the experiment above mentioned experimental setup was prepared again [18, 20].

### 3.2.1 Insert Operation

When we perform the insert operation in MYSQL, we found the variation in time of insertion when we increase the number of records. The details are shown in Table 5.

**Table 5:** Performance of Insertion of Records in MYSQL

No of Records	Time of Execution in MYSQL (Milliseconds)
1 Record	00:00:00:460
100 Record	00:00:00:112
500 Record	00:00:00:200
1000 Record	00:00:00:396
5000 Record	00:00:00:800
10000 Record	00:00:02:056

### 3.2.2 Delete Operation

When we perform the delete operation in MYSQL, we found

the variation in time of deletion when we increase the number of records. The details are shown in Table 6.

**Table 6:** Performance of Deletion of Records in MYSQL

No of Records	Time of Execution in MYSQL (Milliseconds)
1 Record	00:00:00:095
100 Record	00:00:00:026
500 Record	00:00:00:075
1000 Record	00:00:00:091
5000 Record	00:00:03:172
10000 Record	00:00:05:231

### 3.2.3 Select Operation

When we perform the selection operation in MYSQL, we found the variation in time of selection of records when we increase the number of records. The details are shown in Table 7.

**Table 7:** Performance of Selection of Records in MYSQL

No of Records	Time of Execution in MYSQL (Milliseconds)
1 Record	00:00:00:095
100 Record	00:00:00:004
500 Record	00:00:00:007
1000 Record	00:00:00:008
5000 Record	00:00:00:032
10000 Record	00:00:00:060

### 3.2.4 Update Operation

When we perform the updating operation in MYSQL, we found the variation in time of updating records when we increase the number of records. The details are shown in Table 8.

**Table 8:** Performance of Updation of Records in MYSQL

No of Records	Time of Execution in MYSQL (Milliseconds)
1 Record	00:00:00:045
100 Record	00:00:00:056
500 Record	00:00:00:067
1000 Record	00:00:00:170
5000 Record	00:00:03:110
10000 Record	00:00:05:735

## 4. Analysis of performance of different database category

### 4.1 Analysis of Insert Operation

The time difference if we execute the query for 1 record is 455 millisecond, for 100 records is 102 milliseconds, for 500 records is 174 milliseconds, for 1000 records is 351 milliseconds, for 5000 record is 600 milliseconds and for 10000 records the time difference is 1:456 milliseconds. Here we notice that the difference of time is increases when the number of records grows in insert operation. So this indicates that if you have large number of records then use NoSQL database.

### 4.2 Analysis of Delete Operation

The time difference if we execute the query for 1 record is 87 milliseconds, for 100 records is 20 milliseconds, for 500 records is 65 milliseconds, for 1000 records is 68 milliseconds, for 5000 record is 03seconds and 188

milliseconds and for 10000 records the time difference is 5seconds and 240 milliseconds. Here we notice that the delete operation is faster in CouchDB as we compared with MYSQL database. We also noticed that the difference of time is increases when the number of records grows in delete operation. So if you have large number of records then use NoSQL database.

#### 4.3 Analysis of select operation

The time difference if we execute the query for 1 record is 88 milliseconds, for 100 records is 5 milliseconds, for 500 records is 18 milliseconds, for 1000 records is 32 milliseconds, for 5000 record is 188 milliseconds and for 10000 records the time difference is 240 milliseconds. Here we notice that the select operation is faster in relational database MYSQL than in NoSQL database as MYSQL executes the query in less time. We also found that the difference up to 1000 record is 32 millisecond and then grows up at a fast speed. So if you have large number of records and the operation is select then use relational database.

#### 4.4 Analysis of update operation

The time difference if we execute the query for 1 record is 38 milliseconds, for 100 records is 46 milliseconds, for 500 records is 2 milliseconds, for 1000 records is 120 milliseconds, for 5000 record is 850 milliseconds and for 10000 records the time difference is 4 seconds and 235 milliseconds. Here we notice that the update operation is faster in NoSQL database as CouchDB executes the query in less time. We also notice that the difference of time is increases when the number of records grows in update operation. So if you have large number of records then use NoSQL database.

#### 5. Conclusion

Here we compare the two type of database and found that the NoSQL database is better in performance but the relational database is more structured and good when number records is less.

#### 6. References

1. Brown MC. Getting Started with CouchDB O Reilly Media, 2011
2. Abidi D. Overview of Oracle NoSQL Databases, 2011.
3. Lakshman A, Malik. The Apache Cassandra Project.
4. IBM. What is Big Data? Bringing Big Data to the Enterprise, 2012. [www.ibm.com](http://www.ibm.com)
5. Copper J. How entities and indexes are stored, 2009.
6. Google Inc Google Books [online] <http://books.google.com>
7. Online <http://docs.neo4j.org>
8. Garulli L. Orient DB Orient Technologies online, 2012. <http://www.orientdb.org>
9. Chudorow K, Dirolf M. Mongo DB. The definitive Guide. O Reilly Media.
10. Stefen Edlich. List of NOSQL Databases. <http://nosqldatabase.org>
11. Uma Bhat, Shraddha Jadhav. Moving towards Non Relational Databases.

12. Tudorica BG, Bucur C. A comparison between several NOSQL databases with comments and notes.
13. Data modeling and Data Analytics: A survey from a Big Data Perspective by Andre Ribeiro, Afonso Silva, Alberto Rodrigues da Silva, Journal of software Engineering and Applications. 2015; 8:617-634
14. Catell R. Scalable SQL and NoSQL Data Stores. ACM SIGMOD Record. 2011; 39:12-27. <http://dx.doi.org/10.1145/1978915.1978919>
15. Grolinger K, Higashino WA, Tiwari A, Capretz MAM. Data Management in Cloud Environments: NoSQL and NewSQL Data stores. Journal of Cloud Computing: Advances System and Applications. 2013; 2:22. <http://dx.doi.org>
16. Chang F, *et al.* Big table: A Distributed Storage System for Structured Data. Proceedings of the 7<sup>th</sup> Symposium on Operating Systems Design and Implementation (OSDI 06), Seattle. 2006; pp. 205-218.
17. Gyorodi C, Gyorodi R, Sotoc R. A comparative study of Relational and Non Relational Database Models in a Web Based Application. International Journal of Advance Computer Science & Application ISSN: 2158-107X(print) ISSN:2156-5570(online). 2015; 6(2):78-83
18. Jatana N, Puri S, Ahuja M, Kathuria I, Gosain D. A survey and comparison of relational and non relational database" IJERT, ISSN: 2278-0181. 2012; 1(6):1-5.
19. Bulos RD, Bonsol J, Diaz R, Lazaro A, Serra V. Comparative analysis of Relational and Non relational Database Models for simple queries in a web based application research.
20. Moniruzzaman ABM, Hossain SA. NoSQL Database: New era of Databases for Big Data Analytics, Classification, Characteristics and Comparison. International Journal of Database Theory and Application. 2013; 6:1-14.