International Journal of Modern Agriculture, Volume 9, No. 4, 2020
ISSN: 2305-7246

EXPLORING NOSQL DATABASES IN MEDICAL IMAGE MANAGEMENT

Timmana Hari Krishna[1], Dr C. Rajabhushanam[2], Dr C. Rajabhushanam[3]

[1]Dept. of Computer Science and Engineering, Bharath University, Chennai, India

Harikrishna.timmana@gmail.com

[2]Dept. of Computer Science and Engineering, Bharath University, Chennai, India

Harikrishna.timmana@gmail.com

[3]Dept. of Computer Science and Engineering, Bharath Institute of Science & Technology, Bharath Institute of Higher Education and Research Chennai, India

Sivaraman2006@gmail.com

Abstract

Medical Imaging technology is a revolution in Health care industry over the past 3 decades. It allows doctors to notice disease prior and progress patient outcomes. It produces huge semi-structured and unstructured image data as health records which does not contain any fixed schema. It entails better data model for storage and retrieval. NoSQL (Not only SQL) databases are considerably suitable to store and retrieve the Big Data. Hence proposing NOSQL database to store Medical health records. This paper intends to compare the different NOSQL databases in the view of imaging technology

Key words: NoSQL databases, Medical Imaging Technology, HBase, MongoDB, Cassandra

Introduction

In medical informatics, storage and retrieval of medical images is very vital and there is an intense change in data storage methodology in the past decade in every extent. For each curative procedure, there will be a medical image involved as it is part of Medical informatics.

Medical imaging technologies are very significant in medical diagnostics and therapeutics. So there will be extensive increase in the volume of Medical images stored in the future. Experts estimated that, 30% of world's storage will be related to health informatics, and primarily the medical image data in the future. Researchers estimated that the medical imaging systems market will grow to $49 billion in 2020[1, 2]. Medical imaging systems are classified into basic modalities and advanced modalities. General X-ray, mammographic X-ray and ultrasound are the examples for basic modalities. Computed tomography (CT), magnetic resonance (MR), and molecular imaging are the examples for advanced modalities. To reduce the complexity and cost of storage of medical images [3], it is necessary to find an improved solution which helps in efficient archiving of the medical images. Currently, the stored of medical images volume has surpassed 1 Exabyte mark, which yields medical imaging into Big Data world [4]".

Now days, several technologies are developed to lever voluminous data i.e. big data. Medical images are treated under big data category due to volume and variety [5, 6]. Due to variety in data requires change in storage and retrieval which leads to NoSQL. Cassandra, Neo4j, Couch DB, MongoDB and Terrastore are examples for open source NoSQL databases. These databases are capable in handling large amount of unstructured data and semi structured data along with structured data. They provide high performance in data read and write. These databases are being adopted by IT industry and research community. Medical Images are categorized under unstructured data, and be maintained using NoSQL databases certainly. In this paper we discuss and compare the performances of three NoSQL databases, namely HBase, MongoDB and Cassandra with respect to storing of

International Journal of Modern Agriculture, Volume 9, No. 4, 2020
ISSN: 2305-7246

medical images.

This paper is structured as follows: In Section II, we discussed the existing system related to No SQL to storage and retrieval of images. In section III, the necessity and the advantages of NoSQL to store images are discussed. We also compared the different types of NoSQL databases with their functionalities. Further in section IV we projected the implementation details of image storage and retrieval related to HBase, MongoDB and Cassandra, Next in section V we mentioned the system setup detail with experiment results. Finally, section VI affords our conclusions.

Existing Systems Survey

NoSQL is adopted by the industry so rapidly. Under several scientific and research contexts [7,8], It had been compared with relational property. Rascovsky et al. proposed and implemented CouchDB based medical archiving system. The authors recommend that document databases are extremely significant to store, retrieve DICOM files. Also suggested to use Document databases to store the metadata of DICOMM images [8,10], In [8] Luís A. Bastião Silva et al, compared MongoDB and CouchDB in medical images storage and retrieval, and concluded that Mongo DB performance was better than CouchDB. D.R.Rebecca and Dr.I.E. Shanthi [9] compared the storage and retrieval of Medical images in MYSQL with Mongo DB. The results show that the Mongo DB performance was better than MySQL. Also proved that NoSQL is well suitable for storing unstructured data. Generally, Medical images are used to store by stored using a RDBMS based solution called. Various disadvantages of storing medical images in a RDBMS based archive is discussed in [9, 10]. In [10], it is proved that Mongo DB performed better. In the context of medical image storage and retrieval, we definite to compare the different NoSQL databases

Handling Medical Images

The NoSQL databases are very much suitable to store larger size images. It needs to move the medical image providers to cloud for The time has come where the medical image providers are moving to the cloud for their storage requirements [13]. The whole scenario of medical imaging is based on RDBMS [9, 10] is a bad fit compared to cloud based storage systems. [11, 12]. It is required to find an enhanced NoSQL database which stores Medical Images effectively. In this paper, the performances of three NoSQL databases are compared.

Medical Image data comes from different heterogeneous systems. It is in the form of structured, semi-structured and unstructured and huge in the volume. No SQL databases are proficient in handling this kind data. They do not follow a strict schema as they are non-relational. It follows more flexible data model

The advantages of NoSQL databases over RDBMS are i) Highly available with redundancy across multiple locations ii) It is cloud enabled and runs over multiple data centers iii) it has low latency read with high write speed. iv) It is highly scalable and easy to add another processing power and storage.

A single machine or cluster difficult to hold and handle the voluminous data. The design goals to leverage an inexpensive hardware cluster to handle big images effectively and efficiently is as follows:

- Data storage can be distributed to multiple machines instead of centralized system. Giant files can be chunked into small files and stored in multiple nodes.
- Data should be warehoused in flexible schema structure which can be altered easily when it is required.
- Data processing has to be done as isolated subsets and combined them after processing to produce the results for effective bandwidth utilization

NoSQL databases are categorized into a) Key-Value store b) Column-oriented data store c) Document-oriented data store and d) Graph databases. Table 1 shows the comparison between different types of NoSQL databases in terms of performance, scalability, flexibility and complexity

International Journal of Modern Agriculture, Volume 9, No. 4, 2020
ISSN: 2305-7246

Table 1 Feature comparison of No SQL databases

Data model	**Performance**	**Scalability**	**Flexibility**	**Complexity**
Key–value store	high	high	high	none
Column-oriented store	high	high	moderate	low
Document-oriented store	high	variable (high)	high	low
Graph database	variable	variable	high	high

Column-oriented store varies with row-oriented data bases with respects to a) performance b) storage and c) ease of schema modification

Implementation:

The code implementation of storing and retrieve image from different databases is as follows:

i) Apache Cassandra: It is a highly scalable and highly available distributed database with no single point of failure [14]. It provides high performance and designed to handle large amounts of data across many servers. If storing large objects in Cassandra has not handled prudently, it may cause extreme heap pressure and hot spots. Netflix provides Astyanax api to store the image in to Cassandra database concurrently by splitting in to chunks. It provides utility classes that address this issues by splitting up large objects into multiple keys and handles fetching them in random order to reduce hot spots.

The sample code as shown in Fig Store the image in Cassandra as chunks [15]

```
ObjectMetadata imgMd = ChunkedStorage.newWriter(
          cassandraChunkedStorageProvider,
          objName, imageInputStream)
          .withChunkSize(0x1000)
          .withConcurrencyLevel(8)
          .withTtl(60).call();
```

Sample code to retrieve the image in Cassandra

```
ObjectMetadata imgMd =ChunkedStorage.newInfoReader( cassandraChunkedStorageProvider,
objName).call();
ByteArrayOutputStream os = new ByteArrayOutputStream( imgMd.getObjectSize()
.intValue());
imgMd = ChunkedStorage.newReader(cassandraChunkedStorageProvider, objName,
byteArrayOutputStream).withBatchSize(11)
.withRetryPolicy(new ExponentialBackoffWithRetry(250,20))
.withConcurrencyLevel(2).call();
```

ii) Apache HBase: Open source, non-relational, distributed database modeled database which is developed based on Google's Big Table Concept. It fits to key-value workloads with high volume random read and write access patterns used for basic use cases. It serves the image files either storing them in itself or storing the image. Handling of more images is complex and it depends on the name node memory size [17]. The column

International Journal of Modern Agriculture, Volume 9, No. 4, 2020
ISSN: 2305-7246

qualifiers of single column are data and type where data column could store either the path or the actual image bytes and type would store the image type (png, jpg, tiff, etc.). It helps for sending the correct mime type over the wire when returning the image. Cloudera's Kestelyn mentioned that "HBase provides a record-based storage layer that enables fast, random reads and writes to data, complementing Hadoop by emphasizing high throughput at the expense of low-latency I/O

Sample code to store the image in HBase

```
Configuration conf = HBaseConfiguration.create();

HTable table = new HTable(conf, "tst".getBytes());

Put put = new Put("row".getBytes());

put.add("C".getBytes(), "img".getBytes(),extractBytes("/tmp/sample/input.jpg"));

table.put(put);
```

Sample code to retrieve image from HBase

```
Get get = new Get("row".getBytes());

Result result = table.get(get);

byte[] arr = result.getValue("C".getBytes(), "img".getBytes());

OutputStream out = new BufferedOutputStream(new FileOutputStream( "/tmp/sample/output.jpg"));

out.write(arr);
```

MongoDB: It is Cross-platform document-oriented database system that deals JSON-like documents with dynamic schemas. It integration data as BSON (Binary Simple Object Notation) in certain types of applications simpler and faster.

There are many ways to store the images into data base.one is save them in a database as blob type and another is with folder structure which can retrieve them in fast and efficient way. MongoDB uses GridFS file system [16] to store and retrieve images or files in efficient manner. Below same code depicts the way to store and retrieve images using GridFS

```
GridFS gfsImg = new GridFS(mongoTemplate.getDb(), "img");

GridFSInputFile gFile = gfsImg.createFile(content);

gFile.setFilename(fileName);

gFile.setContentType(contentType);

gFile.save();

return gFile.getId()
```

Reading the image from the database is as easy as saving it. Obviously we require the identity of the image. In this case we assume that we have the ID.

```
GridFS gfsPhoto = new GridFS(mongoTemplate.getDb(), "img");
```

International Journal of Modern Agriculture, Volume 9, No. 4, 2020
ISSN: 2305-7246

```
GridFSDBFile img = gfsPhoto.findOne(new ObjectId(id));

InputStream stream = img.getInputStream();
```

Experiments and Discussion:

The Experiments are tested on Ubuntu distribution with 8GB RAM and 1TB hard disk runs on intel core i5 processor. The experimental code is implemented in JAVA. Databases used for the analysis are HBase 1.2.6, MongoDB 3.4.9 and Cassandra 3.11.1. We have tested the setup by storing and retrieving the images varies in size from 1 Mega Byte to 100 Mega Bytes. The mechanism to store and retrieve images related to three databases are mentioned below with code.

Endpoint has performed various experiments [17] on three different NoSQL Databases which is running on Amazon Web Services EC2 instances. For each experiment, new EC2 instance is used to reduce the impact of any "lame instance" or "noisy neighbor" in cloud environments. In performance view, there is no single winner among the NoSQL data bases or processing engines. It is completely depending on deployment and use cases. Fig 1 and 2 show the comparison between NOSQL databases with analogous and diversified operations..

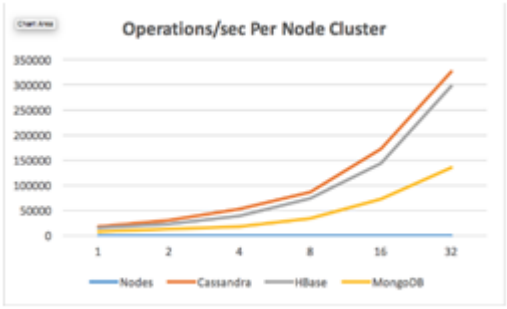

Fig 1- No SQL databases with analogous operations

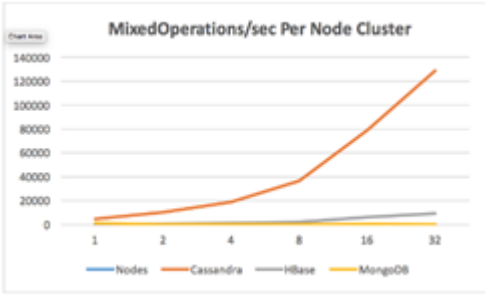

Fig 2- No SQL databases with diversified operations

It shows workload with the throughput/operations-per-second in Y- axis, the number of nodes used presented in X-axis. Table 1 and 2 show the results of each graph. Cassandra is the only database performing durable write operations HBase, and MongoDB perform non-durable write operations. Below results show that Cassandra performed 195 times faster than Mongo DB and six times faster than HBase for varied operational and analytic workloads.

Table 1 – Comparison between NOSQL data bases with same operations

International Journal of Modern Agriculture, Volume 9, No. 4, 2020
ISSN: 2305-7246

Nodes	Cassandra	HBase	MongoDB
1	18,683.43	15,617.98	8,368.44
2	31,144.24	23,373.93	13,462.51
4	53,067.62	38,991.82	18,038.49
8	86,924.94	74,405.64	34,305.30
16	173,001.20	143,553.41	73,335.62
32	326,427.07	296,857.36	134,968.87

Table 2 – Comparison between NOSQL data bases with diversified operations

Nodes	Cassandra	HBase	MongoDB
1	4,690.41	269.3	939.01
2	10,386.08	333.12	30.96
4	18,720.50	1,228.61	10.55
8	36,773.58	2,151.74	39.28
16	78,894.24	5,986.65	377.04
32	128,994.91	8,936.18	227.8

Conclusion:

In this paper we have discussed about existing methods to handle medical images. We have experimented with three different types of databases namely HBase, MongoDB and Cassandra. And compare them with respect to time in read and write operations. In this study, the results showed that Cassandra has fast write and read performance [14] with high linear scale performance among top NoSQL databases.

References

1. A DBA's Guide to NoSQL, Apache Cassandra, Datastax-2014.

2. Hari Krishna Timmana and Dr C Rajabhushanam," Mininet Implementation of SDN towards Network Softwarization", International Journal Of Innovative Research In Management, Engineering And Technology,Vol. 2, Issue 5, May 2017

3. Shashank tiwari, professional Nosql, John Wiley & Sons, Inc. 2011 Available at http://www.frost.com/prod/servlet/press-release.pag?docid=268728701

4. Frost & Sullivan: U.S. Medical Imaging Informatics Industry Reconnects with Growth in the Enterprise Image Archiving Market

5. Dan McCreary,Ann Kelly, Making Sense of NoSQL Databases, Manning Publications,2014.

6. N. V. Chawla and D. A. Davis, "Bringing big data to personalized healthcare: A patient-centered framework," Journal of general internal medicine, vol. 28, pp. 660-665, 2013

7. Rajat Aghi, Sumeet Mehta, Rahul Chauhan, Siddhant Chaudhary and Navdeep Bohra,A comprehensive comparison of SQL and MongoDB databases,International Journal of Scientific and Research Publications, Volume 5, Issue 2, February 2015,ISSN 2250-3153

8. Luís A. Bastião Silva, Louis Beroud, Carlos Costa and José Luis Oliveira,Medical imaging archiving: a comparison between several NoSQL,978-1-4799-2131-7/14/$31.00 ©2014 IEEE

International Journal of Modern Agriculture, Volume 9, No. 4, 2020
ISSN: 2305-7246

9. D.Revina Rebecca, I.Elizabeth Shanthi,A NoSQL Solution to efficient storage and retrieval of Medical Images,International Journal of Scientific & Engineering Research, Volume 7, Issue 2, February 2016,ISSN 2229-5518

10. Simón J. Rascovsky, MD, and et.al,Use of CouchDB for Document-based Storage of DICOM Objects

11. Katarina Grolinger1, Wilson A Higashino, Abhinav Tiwari and Miriam AM Capretz,Data management in cloud environments: NoSQL and NewSQL data stores,journal of cloud Computing, Springer Open journal, 2013

12. J.Antony John Prabu, Dr.S Britto Ramesh Kumar,Issues and Challenges of Data Transaction Management in Cloud Environment

13. http://docs.datastax.com/en/cassandra/2.1/cassandra/gettingStartedCassandraIntro.html

14. https://github.com/Netflix/astyanax/wiki/Chunked-Object-Store

15. "Cassandra vs. MongoDB vs. Couchbase vs. HBase" available at https://www.datastax.com/nosql-databases/benchmarks-cassandra-vs-mongodb-vs-hbase

16. Mongo DB api site Available at http://api.mongodb.com/

17. Apache HBase Documentation Available at http://hbase.apache.org/

18. Sivaraman, K., Dr. K.P. Kaliyamurthie, Cloud Computing in Mobile Technology, Journal of Chemical and Pharmaceutical Sciences, 2016

19. Priya. N, Sridhar. J, Sriram M, Mobile large data storage security in cloud computing environment- a new approach, Journal of Chemical and Pharmaceutical Sciences, Vol. 9, Issue 2, April-June 2016