Role of Graph Databases in Social Networking Sites: A Performance Comparison between Graph Database Neo4j and Relational Database Mysql in Social Networking Sites

Rida Fatima[#1], Adeel Ahmed[#2]

Department of Computer Science
SZABIST, Karachi, Pakistan
[1]rfatima.87@gmail.com
[2]adeel.ahmed@szabist.edu

***Abstract* — for a long period Relational Databases were the king for data storage, data retrieval and data manipulation sites and applications. SQL is structured query language used to retrieve and manipulate data. Relational database applications are usually fast and effective but if there are many relationships that require Joins between large tables then their efficiency becomes low and sometimes relational databases fail to retrieve data. In short, relational database is best for applications that have small and fixed number of relationships and it does not support the applications that require continuous database schema changes for example Social networking applications. Nowadays, there is more interest in graph database. Graph databases can easily handle huge records of data as they do not require costly joins. As they do not depend on static schemas, they are best for managing data with dynamic schemas. Social networking sites have multiple connections in between objects. Social networking sites have tree based database structures. Graph databases are like tree based structures too. So, this study is about the comparison of Graph databases Neo4j with the most popular relational database, Mysql in the field of social networking applications and to provide the best performance results.**

Keywords; Relational Databases, Graph Databases, Social Networking, SQL, Mysql, Neo4j.

I. INTRODUCTION

Large scale information storage and retrieval is becoming today's need for any system. Data is separated into portions of information, normally arranged or designed in a special way. All systems or software has two types: data and programs. Programs are set of commands that deal with data and manipulate data. Datum is the singular of data but data is used as singular and plural both.

A database is an organized collection of knowledge in which data can easily be fetched, organized and updated. Databases can be categorized by content type: numeric, text and images. The database idea has developed since the 1960s to decrease difficulties in designing, structuring, and managing difficult systems. It has grown with database management systems that handle databases effectively and efficiently.

A relational database is a set of data items that is maintained as in properly defined tables from which data can be fetched or updated in many ways without reorganizing the database tables. E. F. Codd invented relational database at IBM in 1970.

A graph consists of nodes, edges, and properties to represent and save information. A graph can be a single node; it also has records that are called properties. In the start, a node could only have a single property but with time it grew to large amount of properties. There are relationships between nodes; relationships also have properties to store values. Relationships arrange nodes into a structure that allows the graph to act like a list, map, tree or composite entity.

We can query the Graph by traversal; traversal makes a path from the initial node to correlated nodes by using provided instructions and these instructions can be questions like, "All music liked by my friends that I have not listened to yet".

We can also query a specific node or relationship using its property without traversing the whole graph. We can use an index to retrieve information, for example "find out the detailed information of my friend whose name is Albert". Graph databases can easily handle huge records of data as they do not usually require costly joins. As they do not depend on static schema, they are best for managing data with dynamic schemas. Social networking sites have

multiple connections in between objects. Social networking sites have tree based database structures. Graph databases are like tree based structures too. So, this study is about the comparison of Graph databases Neo4j with one of the most popular relational databases, Mysql, in the field of social networking applications and to provide the best performance results.

Neo4j provide ACID transactions. Neo4j forces all operations to execute in one transaction that can alter the information. Neo4j can easily scale in size from application to application. Neo4j is so fast that it can execute millions of traversal operations in one second.

II. Literature Review

In the 1970s, the first relational database management system was created and it became one of the most popular system for storing and manipulating data among educational and commercial fields. Relational databases have small scale databases like Microsoft Access and also have large scale databases like Microsoft SQL Server, Oracle and Mysql etc. After the popularity of the internet and it being used everywhere as a part of life, data storage needed to increase and enhance its size and interconnectivity. Data storage in graph structure is the best solution for this [3].

In this era, there are other database structures being used by various applications which are not focusing on the relational model. Google BigTable is fast and has a huge scale. It has facility to act as a column oriented database or as a row oriented database. Dynamo is a highly distributed key value storage system used by Amazon [3]. Cassandra is also a key value storage system developed by the Facebook team. Cassandra also has the BigTable characteristics of row and column orientation [5].

In current years, research on data storage as in graph form has been increased. After studying social networking sites, analyzing the huge internet use and strong interaction between people, the importance of graph structure for data storage systems increased [1].

The improvement of huge systems such as the Internet, geological systems or dynamically created social network data storage systems, the need to graph like structure to store information has increased [2].

Nowadays, there are some applications that use the graph structure to store data to some extend but those systems have three major problems: (i) Data sources are continuously growing, (ii) There should be a standard query language to retrieve data by keywords search and other relational aspects and (iii) To combine data together coming from multiple sources and put results for difficult queries [2].

The secret behind the success of the relational model is its ease of use because of its simplicity it has control over the database world for many years. There is a lot of mathematical research done to make it more efficient and user friendly; non-functional query languages are its proof. However now, sometimes these query languages fail to retrieve data using joins between large scale tables [8]. For this, there is an open source graph database called Neo4j that has its own query language called 'Cypher'. Cypher is a human understandable language and can easily be understood by developers and operation professionals. Its keywords are based on English words and icons that a normal user can easily understand. Some popular keywords of the Cypher Query language are; START, MATCH, WHERE, RETURN [12].

III. Research Scope And Method

The scope of this research is to find out the better options in database structures for all network related sites and applications and to provide the best performance results.

This study is based on comparative and experimental research methods. First, we analysed the best database structure for social networking sites, then we compared it with one of the most popular relational databases, Mysql and provide the performance results.

IV. Design for Experiment

A. Measures

This assessment methodology is made for comparing objective benchmarks of Mysql and Neo4j databases according to the system requirements and experience. The objective comparison is done by measuring the processing speed by using some selected set of queries, and examining how scalable they are.

B. Relational Database (Mysql)

Currently Mysql is the king of database servers; it is the most popular database server among other relational databases. With the combination of PHP script, it is mostly used to develop dynamic, powerful and scalable server side applications. So we chose the most popular and powerful relational database system Mysql, to compare it with Neo4j for better comparison results.

.

C. Graph Database (Neo4j)

Neo4j is an open source java based graph database. Neo4j provide ACID transactions. Neo4j forces all operations to execute in one transaction that can alter the information. Neo4j can easily scale in size from application to application. Neo4j is so fast that it can execute millions of traversal operations in one second. So we chose Neo4j as the graph databases for this experiment.

We downloaded the latest Neo4j server from the official neo4j site. It is java based so it required a Java Runtime environment to run. We also downloaded a PHP wrapper for the Neo4j graph database 'REST' interface and

connected it with the Neo4j server using the provided details and then executed our queries using this wrapper. This wrapper required PHP version 5.3.1 and above.

D. Social Networking site databases (Facebook databases)

Using social networking sites, users can keep in touch with their current friends and can reconnect with old friends or can create new friends by matching the same interests and activities. A user can widen his social circle by connecting with friends of friends. Users can share their academic, professional and family oriented information as well. Users can share their thoughts, interests with other user and can get feedback on it. Recently, Facebook is one of the most popular social networking sites, so we chose the Facebook database schema in our experiment and listed down some queries which we will use for measures.

Following are some major tables of Facebook [13];

- Albums
- Applications
- Comments
- Events
- FriendList
- Groups
- Messages
- Pages
- Photos
- Posts
- Status messages
- Users
- Videos

V. EXPERIMENT (CASE STUDY)

Testing machine's CPU running at 3.Ghz with Intel Core 2 Duo processor and has 2 GB RAM.
We took a dataset of,

- 1000 content items (user comments on fan pages) [table name: Comments]
- 500 Fan Pages [table name: Fanpages]
- 200 Users have pages in their favorite fanpage set [table name: Users]
- A graph of fan relations between Users and Fanpages [table name: Favorite_Fanpages]

We logically designed the same databases for both competitors Mysql and Neo4j so that the accuracy of queries can be evaluated accurately.

A. *Speed Measures*

Query: For every user, select all comments of a user on his favorite fan pages.
We run the above query on both database servers for 10 times. Then we took the average time for both Mysql and Neo4j databases for correct and accurate results.

1) *Mysql:*

```
For all (User_ID in Users) {
   SELECT c.text
   FROM Comments c
   JOIN Favorite_Fanpages ff ON ff.FanPage_ID =
   c.FanPage_ID
   JOIN Fanpages fp ON fp.FanPage_ID = c.FanPag
   e_ID
   WHERE c.User_ID = User_ID AND ff.User_ID =
   User_ID
}
```

Using MySQL It took 152 Seconds to display the list of all 200 users' comments.

2)Neo4j:

```
For all (user in index){
   START user=node:node_auto_index(name =
   {user_name})
   MATCH user-[:fan_of]->fanpage<-[:posted_on]-
   otheruser
   RETURN fanpage.comments
 }
```

Using Neo4j It took 3 Seconds to display the list of all 200 users' comments.

B. Result

1- We logically designed the same databases for both competitors Mysql and Neo4j for accurate results.

2- We ran the selected query on both database servers for 10 times and took the average time for both Mysql and Neo4j databases for correct and accurate results.

Using MySQL, it took 152 Seconds to display the list of all 200 users' comments.

Using Neo4j, it took 3 Seconds to display the list of all 200 users' comments.

TABLE I.
SPEED MEASURMENT IN DATABASES

S No.	***Dataset***	***Mysql Speed***	***Neo4j Speed***
1	200 users	152 seconds	3,4 seconds

VI. CONCLUSION

Both database servers performed well in different ways but the graph database, Neo4j, returns better result against the structural queries and for the space test relational database Mysql required less space than Neo4j. If relational database applications have small scale relationship operations then they are usually fast and efficient but if there are many relationships that require Joins between large tables than their efficiency decreases and sometimes relational databases fail to retrieve data. In short, relational database is best for applications that have small and fixed number of relationships and does not support the applications that require continuous database schema changes for example Social networking applications. Graph databases can easily handle huge records of data as they do not require costly joins. As they do not depend on static schema, they are best for managing data with dynamic schemas. So, graph databases, especially Neo4j, is a better option for Social networking sites.

In future, I will design a bigger social networking site with complex relations and test more queries for better performance check.

ACKNOWLEDGEMENT

In the name of Allah the Almighty, All praise is to Him, Lord of the worlds and hereafter.

My profound regards and gratitude to Mr. Adeel Ahmed because without his able guidance and all the academic and material help that he provided this paper would not have turned into reality.

REFERENCES

[1] David Dominguez-Sal, Norbert Martinez-Bazan, Victor Muntes-Mulero, Pere Baleta, and Josep Lluis Larriba-Pey; "A Discussion on the Design of Graph Database Benchmarks".

[2] Norbert Martínez-Bazan, Victor Muntés-Mulero, Sergio Gómez-Villamor, "DEX: High-Performance Exploration on Large Graphs for Information Retrieval".*CIKM'07*, November 6–8, 2007, Lisboa, Portugal.

[3] Chad Vicknair, Michael Macias, Zhendong Zhao, Xiaofei Nan, Yixin Chen, Dawn Wilkins,"A Comparison of a Graph Database and a Relational Database A Data Provenance Perspective", 2010.

[4] Adrian Silvescu, Doina Caragea, Anna Atramentov;"Graph Databases".

[5] Avinash Lakshman, Prashant Malik; "Cassandra: a decentralized structured storage system"**,** 2010.

[6] Anton Dries, Siegfried Nijssen; "Analyzing graph databases by aggregate queries", 2010.

[7] Matthew Rowe, "Interlinking Distributed Social Graphs", LDOW2009, April 20, 2009, Madrid, Spain.

[8] Alberto O. Mendelzon, Peter T.Wood, "Finding Regular Simple paths in graph databases", (Amsterdam, The Netherlands, August 22–25, 1989.

[9] Walter Kriha, "NoSQL Databases", Stuttgart Media University.

R. Angles and C. Gutierrez, "Survey of graph database models", *ACM Comput. Surv.*, 40(1):1–39, 2008.

[10] F. Chang, J. Dean, S. Ghemawat, W. C. Hsieh, D. A. Wallach, M. Burrows, T. Chandra, A. Fikes, and R. E. Gruber, "Bigtable: A distributed storage system for structured data", ACM Trans. Comput. Syst., 26(2):1–26, 2008.

[11] Neo4j. Home. http://neo4j.org, 2012.

[12] Facebook.FQL. https://developers.facebook.com/docs/reference/fql/