UNIVERSITY OF SCIENCE AND TECHNOLOGY

COLLEGE OF GRADUATE STUDIES AND ACADEMIC ADVANCEMENT

Faculty of Computer Science and Information Technology

Distributed Query Processing on NoSql Couch DB and Redis DB using Medical dataset

By

Sundus Elfadil Ahmed Fadul

A Thesis

Submitted to the College of Graduate Studies and Academic Advancements
In PartialFulfillment of the Requirement for the Degree of Master of Science in Information Technology

Supervisor

Dr: Mohammed Bakri Bashir

August 2016

Abstract

When the software arise in previous years used relational databases to store the data and when the advanced programs appearance and access to the contents of the programs for large numbers of users become is a big data increase on web pages such as web 3.0 need to be characteristics does not provided in relational database so when the third generation of modern software have emerged is the ideal solution for traditional applications. One of the purposes of this research is to provide Introduction to non-relational data and the reasons that led to the emergence addition of the goals of this research single operation "distributed query" for data from non-relational distributed database on a number of computers and the possibility of measuring performance. And also of the goals of this research some of the previous work of the studies of non-relational data (NoSql) that depend on storage in the process and the different types of non-relational data stores. Use in this search of the most two famous of NoSQl DBMS Key value store "Redis", which a single key-value index for all the data call these systems key-value stores and CouchDB collections of documents . And also calculating the time of the operation depending on the data size and number of machine distributed to each data store redisDB and CouchDB.

المستخلص

عند ما ظهرت البرمجيات في الاعوام السابقه كانت تستخدم قواعد البيانات العلائقيه لتخزين البيانات وعند ظهور البرامج المتطور والوصول لمحتويات البرامج لاعداد كبيرة من المستخدمين اصبحت البيانات تزداد بشكل كبير علي صفحات الويب التي تحتاج الي خصائص لاتتوفر في قواعد البيانات العلائقيه لذلك عندما ظهرت برمجيات الجيل الثالث الحديث التي تعتبر الحل الامثل للتطبيقات التقليدية .أحد أغراض هذا البحث هى تقديم مقدمة تعريفيه عن قواعد البيانات الغير علائقية والأسباب التى أدت الى ظهورها بالأضافة الى ذلك من أهداف هذا البحث "عملية الاستعلام الموزع" او البحث عن البيانات من قواعد البيانات الغير العلائقية الموزعة علي عدد من الكمبيوترات و امكانية قياس الاداء . وايضا من اهداف هذا البحث تم تناول بعض الاعمال للدراسات السابقه لقواعد البيانات الغير العلائقيه التي تعتمد علي عملية التخزين فيها و علي انواع مختلفه من مخازن البيانات الغير العلائقية من اشهر اثنين منهم "Redis "Key value store التي تبني عملية التخزين فيها علي مفتاح واحد لجميع البيانات "Couchbase", document oriented base"التي تعتمد عملية تخزين البيانات فيها من النوع المستند . وايضا يتم عمل حساب زمن تنفيذ العمليه اعتمادا علي حجم البيانات وعدد الاجهزة المضافة وعدد مرات التشغيل بطريقه موزعه للكل قاعدة بيانات.

1.1 In troduction

Some of web applications in these days face a challenge of large scale of data and serving millions of users distributed over the world. Additionally, users expect the service to be always available, reliable, and with a high degree of consistency. on the other hand, the rapid increasing of the number of users and the amount of generated data, which should be stored in many servers that distributed over different locations[1]. The traditional relational database is increasingly incompatible with the demands of availability, scalability for massive data, fast data backup and recovery with large-scale web applications led this challenge to new concept of DBMS called NoSQL DBMS. NoSQL DBMSs have the ability to scale data over many nodes providing the needed levels of availability while keeping scalability within accepted levels and ignoring data consistency. This is a broad class of database management systems that differ from classic relational database management systems[2].

1.2Problem Background

Although many different types of database systems exist for decades, the relational database is the most famous database system. These relational database systems are developed, used, and optimized for decades and offer a solid solution for data storage in many different areas. Especially in the area of web applications, where the relational database used to be the standard database system for almost every application[3]. Additionally, recently a new trend is spotted in web development communities that don't utilizes the traditional relational databases management systems such as Oracle databases MySQL and Microsoft SQL Servers. the continues increment in the number of applications that use relational databases leads to problems because of deficits and problems in the modeling of Servers that are commonly used for websites[4]. Developers start considering alternative types of database systems for their data storage products. These DBMS's like CouchDB, MongoDB, Neo4j, Apache Cassandra, memcached, Redis, JADE, and Apache Hadoop are encountered more often in the context of web development instead of conventional relational databases. The above DBMS's could be categorizing to document stores, graph databases, key-value stores, object databases, and tabular data storage systems. The products are so-called NoSQL data storage systems [3].

There are two trends that bringing these problems to the attention of the international software community:

1. The exponential growth of the volume of data generated by users, systems and sensors, further accelerated by the concentration of large part of this volume on big distributed systems like Amazon, Google and other cloud services.

2. The increasing interdependency and complexity of data accelerated by the Internet, Web2.0, social networks and open and standardized access to data sources from a large number of different systems[4].

1.3 Problem Statement

Digital world is growing very fast and become more complex in the volume (terabyte to petabyte), variety (structured and un-structured and hybrid), velocity (high speed in growth) in nature. This refers to as 'Big Data' that is a global phenomenon. This is typically considered to be a data collection that has grown so large it can't be effectively managed or exploited using conventional data management tools classic relational database management systems (RDBMS) or conventional search engines. To handle this problem in traditional RDBMS are complemented by specifically designed a rich set of alternative DBMS suggested new technologies also known as NoSQL, for ―Not Only SQL, refers to an eclectic and increasingly familiar group of non-relational data management systems[4]. NoSQL DBMS is developed because the recent applications produce different kind of data and diverse from rational database in format with different query languages. Consequently, running distributed query processing is considered as significant issues which require study. This research study and compare the performance of the NoSQL DBMS's in distributed environments.

1.4 Research Objectives

1. Design and develop an user interface that able to retrieve from distributed NoSQL databases.
2. Compare the NoSQL DBMS's based distributed query processing.

1.5 Research Contribution

1. Comparative analysis of the recent work conducted to test the NosSQL database in distributed environments.

2. User interface can be used to execute user's queries over distributed databases.

3. Evaluate the performance NoSQL databases in term of response time of the Distributed queries.

1.6 The Scope of Research

Distributed query processing among the NOSQL DBMS and evaluate the performance and scalability by testing "query" operation.

1.7 Thesis Outlines

The thesis is organized in the following paragraphs.

Chapter 2 presents the literature review of related to NOSQL and NOSQL distributed query. The chapter started with general overview for NoSQL database by defining NoSQL and their types and characteristics of each type.

Chapter 3 prescribes the research methodology which used in the thesis.

Chapter 4 prescribes the test case dataset and operations testing interface implementation and the experiments and the results.

Chapter 5 illustrates conclusions and recommendations of the thesis.