

Josh Newlands - 1602713

by Joshua Newlands

FILE	12810_JOSHUA_NEWLANDS_JOSH_NEWLANDS_- _1602713_178152_1972892745.PDF (109.26K)		
TIME SUBMITTED	12-APR-2019 11:58PM (UTC+0100)	WORD COUNT	1635
SUBMISSION ID	104713317	CHARACTER COUNT	8557



Investigating Emerging Web Technologies

M6X01096

Lecturer: Mike Dacey

Josh Newlands, 1602713
josh.newlands@outlook.com

Table of Contents

1. Introduction	2
2.Redis Database Management System	2
3.Relational database vs NoSQL databases	3
3.1 Traditional SQL	3
3.2 NoSQL databases	3
4.References	4

1. Introduction

The advancement of convenient technologies that can be used to access the internet has led to the popularity of the internet rising higher than it has ever been. The internet is being used for services, data transfer, instant chat and to search information. This expansive use of the internet leads to an insurmountable increase in the amount of data that must be managed, with the increase of information to be managed comes the need for an adaptable web technology to handle this data. One of the web technologies that stands to make an impact in this field is Redis.[6]

2. Redis Database Management System

Redis is a database management system that uses key-value storage as its primary database model to improve the overall performance of database management, it is different from traditional SQL as it does not use the relational database model. Key Value stores give the developer the ability to sort through data based on a key and value pair. The simplicity of which gives key value stores large performance and easy scalability. Redis can also use Document store, Graph DBMS, Search engine, Time Series DBMS as secondary database models.[7]

An example of where redis could be used with the key-value store model could be with a webservice like twitter, the key would be the twitter ID whilst the value would be all the information about the tweet such as the message and who wrote it.

Web servers have seen an increase in use that stems from the large availability of the internet in recent times, the use of smartphones to connect to the internet on the go as well as it being common place for families to have a home internet connection. It wouldn't be far stretched to say that most people are connected to the internet almost 24/7. With this Increase in users comes big data associated with site traffic.[7]

Many companies make big data a focus point, being able to store, access, and analyse this data is a necessity for companies that make it online. The traditional method of manipulating this data comes from the standard SQL relational database management system such as Oracle. Using the traditional data manipulation system can be quite costly in particular to young businesses, it requires a competent database administrator as well as some expensive servers to keep the data. Scaling up using this system can also be both costly and difficult.[4]

Redis is a candidate for having the potential to largely impact the future of web development as it offers a simplistic form of information management that boasts impressive speed as well as being compatible with an impressive number of programming languages. As well as being open sourced it can also run on BSD, Linux, OS X, and Windows servers. Redis is also benchmarked as the world's fastest database.[3]

Redis is able to be used as a search engine, time-series database, graph database, JSON store.

3. Relational database vs NoSQL databases

3.1 Traditional SQL

Traditional SQL databases like Oracle use the relational database model. The relational approach to databases is widely used for its ACID compliance. SQL uses table-based databases which allows for relationships to be created between tables, to enforce features such as, referential integrity and reducing anomalies. [4,5]

ACID compliance is the features of the database that need atomicity, consistency, isolation, and durability. These features of a relational database are what makes the SQL databases good for data that is structured and unchanging. [7]

SQL databases are best used in complex query-intensive environments due to the structured query language having the strongest capabilities, NoSQL does not match up in this regard. This also contributes to the scaling of SQL databases; they tend to be vertically scalable in the sense of more CPU power, RAM, and storage results in handling a bigger load. [5]

To set up a relational database model requires a competent database administrator to set up relational tables. As well as having big expensive servers when there is a lot of data involved, these factors can increase the expensiveness of setting up as well as restrict expansion. There is a lot of support available for SQL including professional services and expert help, unlike with NoSQL where the support would mainly come from the community as there are fewer experts available in that field. [4]

Relational databases are extremely well tested and reliable with set transaction processes making them reliable for secure and integral databases. When data is hierarchical and relational, the SQL becomes best suited due to its robustness, secure transactions, and complex query handling. A typical use case for SQL databases may be through a booking system involving money, since this would require atomicity to ensure that the transaction either wholly completes or not at all. In scenarios such as these, it would also do to have a relational table to show how the customer relates to orders or to venues, as well as having complex queries in place that ensure that there are no double bookings involved. The structured query language excels at ensuring these factors take place reliably, since reliability is preferred over speed in this matter; SQL relational databases are ideal.

3.2 NoSQL databases

MongoDB is a NoSQL document-oriented database. This type of database has increased in popularity due to its ability to deal with big data, unlike relational databases which have their limits in terms of massive storage. [2]

Relational database management systems require expensive equipment to scale up to large databases, whilst MongoDB and NoSQL databases use multiple servers with different techniques to operate with big data at normal costs; they can even use cloud storage without function being comprised, this allows for cheaper equipment to be used for testing. A NoSQL database is not as integral as a relational model database but it does have many advantages when used for the correct scenarios. Schema-less links to local programming languages flexible as its in JSON format. This allows it to add types of information unlike in SQL databases, for instance you could define a car as a type Vauxhall. Due to this schema-less feature, NoSQL databases are best suited for hierarchical data structures in particular with the key-value pair data model. [6]

Whilst there is fewer expert help available for NoSQL databases, there is a large community available for information and the set-up of a NoSQL database is comparatively easier to do than

relational databases. A noSQL database is used instead of a relational database to prevent the database from becoming a the bottleneck of the system as with large amounts of data noSQL databases perform better than relational databases. noSQL databases typically require a lot less preparation than relational databases allowing a faster system development stage.[1]

In noSQL databases indexing works differently than in SQL databases, rather than having a tree hierarchy of the structure of the relational data noSQL points to parts of documents of which the information resides making a faster result.[6]

MongoDB is free to use and easy to install and manage. MongoDB supports embedded documents and indexing which provides increased processing. Has some common Relational database features as well that make mongoDB adaptable and scalable whilst being able to perform Dynamic queries, sorting, secondary indexes, rich updates, easy aggregation.

Nosql becomes suitable for businesses that are fresh starting out due to its cheaper start up cost and excel in scalability. As well as this the nosql database becomes suitable for businesses that have non hierarchical data that may need schemaless types of data. If the queries that are required to be performed on the database is non complex but numerous then the nosql database can shine in ensuring that the database does not become a bottleneck in the system speed. Use Case scenario may see the nosql database being used on social websites such as twitter or facebook, utilizing key-value pairs to keys to information about posts. This is suitable since most twitter like websites would have some basic search functions as well as a lot of fetch requests, in order to handle fetching this amount of data from the number of connections that may be up database speed would become essential, this as well as the sheer scale of the data involved proves a noSQL database would be suitable in this situation.

4.References

- [1]B. Jose and S. Abraham, "Exploring the merits of nosql: A study based on mongodb", 2017 International Conference on Networks & Advances in Computational Technologies (NetACT), 2017. Available: 10.1109/netact.2017.8076778 [Accessed 1 April 2019].

 - [2]D. Maltby, "Big Data Analytics", University of Texas at Austin School of Information, 2014.

 - [3]R. Cattell, "Scalable SQL and NoSQL data stores", ACM SIGMOD Record, vol. 39, no. 4, p. 12, 2011. Available: 10.1145/1978915.1978919 [Accessed 1 April 2019].

 - [4]S. Binani, A. Gutti and S. Upadhyay, "SQL vs. NoSQL vs. NewSQL- A Comparative Study", Communications on Applied Electronics, vol. 6, no. 1, pp. 43-46, 2016. Available: 10.5120/cae2016652418 [Accessed 2 April 2019].

 - [5]S. Agarwal, k. Rajan and K. Rajana, "Analyzing the performance of NoSQL vs. SQL databases for Spatial and Aggregate queries", 2017. Available:
-

<https://pdfs.semanticscholar.org/01bd/f45802014e39ec60926fce6627a45aba0bf8.pdf>. [Accessed 3 April 2019].

[6]V. Abramova and J. Bernardino, "NoSQL databases", Proceedings of the International C* Conference on Computer Science and Software Engineering - C3S2E '13, 2013. Available: 10.1145/2494444.2494447 [Accessed 3 April 2019].

[7]M. Mohamed, O. Altrafi and M. Ismail, "Relational vs. NoSQL Databases: A survey", International Journal of Computer and Information Technology, vol. 3, no. 3, 2014. Available: <https://pdfs.semanticscholar.org/37f3/9efa3bc92028b7d867cdb8a9165f2a03843a.pdf>. [Accessed 3 April 2019].

FINAL GRADE

40/100

GENERAL COMMENTS

Instructor

Eval emergent tech (20) 17

Eval of NoSQL (30) 23

Typescript application (20) 0

REST web service (20) 0

Tech demo (10) 0

The emergent technology that you chose was Redis. You have provided some interesting arguments for its future growth, but you did not provide good examples of the types of (future) applications it might be used for.

The NoSQL database you chose was Redis. Again a nice analysis but I would like to have seen some examples of its usage.

No code provided.

PAGE 1

PAGE 2

PAGE 3



Comment 1

Why is this relevant?

PAGE 4
