Michael Elder

Professor Johnson

CS-300-ON

9/4/2022

Non-Relational Data Storage and Retrieval Systems

Over the course of this program, we have learned a lot about Relational Database Management Systems and this paper will make it our first time we have researched a non-relational data storage system. Non-Relational Database Management Systems have their pros and cons, and I am going to be writing about NoSQL databases and graph databases. I find this topic interesting because I have heard of NoSQL databases before I started this program. I will be discussing the NoSQL and graph database's intended use for what it's designed to solve, examples of each database, advantages and disadvantages, etc.

Microsoft Azure says, "NoSQL databases are interchangeably referred to as "nonrelational," "NoSQL DBs," or "non-SQL" to highlight the fact that they can handle huge volumes of rapidly changing, unstructured data in different ways than a relational (SQL) database with rows and tables ("NoSQL Database - What Is NoSQL? | Microsoft Azure.")." NoSQL technologies have been around since the 1960's but they have been being used a lot more frequently now due to the growth of data and its practicality in real life. It has a lot more flexibility and adaptability than regular Relational Database Management Systems which is very important for certain types of data that don't necessarily need to be in a heavily structured database. NoSQL allows you to store data easier and run queries a lot faster because the data isn't in the format of a Relational Database Management System such as a row-based table structure that connects related data. The main difference between a Relational Database Management System and NoSQL Database is the way that the database is modeled. Some of the features of NoSQL databases are flexible schemas, horizontal scaling, fast queries to the data model, and ease of use for developers. Microsoft Azure also states that there are four different types of NoSQL databases: Document Based No SQL Databases, Key Value Databases, Wide-Columned Based Databases, and Graph-Based Databases.

The first type of NoSQL database is the document-based NoSQL database. The document-based database allows you to organize documents into groups and queries on any attribute within a document. The second type of NoSQL database is the key-value database that contains a key value with a value that its associated with but it can unknown. The third NoSQL database is a wide-columned base database which allows you to store data and query across columns in the database. The last NoSQL database is a graph-based database which is modeled on nodes and their interconnected relationships to other different nodes. A good example of this would be social networking and all of the relationships with different users. The NoSQL databases are more used for more unstructured data that can be easily changed. They are used for apps, analytics, database migration, and content management. They were designed to process in volume from a lot of sources and from different formats. Currently, this is a very important feature because all of our different applications and databases will provide an enormous amount of data and with regular Relational Database Management Systems there will not be enough memory available for some complex databases that need more memory.

With internet and technology progressing, NoSQL databases are becoming a lot more popular than they used to be due to the ease of using the database and the available memory. The No-SQL databases are very useful for cloud computing which is becoming more and more popular. Cloud gaming allows you to play a game without downloading it. This makes you think how it can do that. It has to be downloaded somewhere else and then have the software on your certain system to remotely access it and play it remotely without having it downloaded directly on your system. This is just one example of what NoSQL databases can do for cloud computing. The problems NoSQL databases were designed to solve the big data issue by allowing the content of the database be unstructured. This allows more flexibility and scalability in a NoSQL database. This is very important for bigger companies or companies that have a lot of data or could becoming into a lot of changes in your database.

The main advantages of NoSQL Databases are performance, scalability, and flexibility. NoSQL Databases perform better than relational database management systems because they are formatted into different tables. You would have to compare the data to multiple tables which would make the query time slower. With NoSQL Databases, all of the data is unstructured and in a single database which allows the computer to work faster analyzing the data and returning your queried data. NoSQL databases are also easy to grow because they can support a great amount of data. As for SQL or Relational Databases, you would have to upgrade your CPU and RAM to

better process and query the data in your database because of its size. As for the NoSQL database, the process for adding more data is cheap and easy. Flexibility is the final advantage I will be talking about and this allows the data to be unstructured which in turn allows the run time to be quicker because the database isn't looking for relationships with other data and tables. It is searching the unstructured data and returning exactly what you are asking for which makes for quicker computing time.

The cons for NoSQL databases are that they are not mature, require multiple databases, and they contain huge databases. They are not mature because SQL is more widely used and invested in because it has been more popular to use. More people know SQL than they do NoSQL, so this brings up issues with hiring and support for any issues that come up with your database. It required multiple databases depending on what you are trying to do with the data and what schema you would like to use. As listed above, you can have different database types depending on what kind of data you have or what you would like to do with the data such as the key-value database, document-based database, wide-column database, and graph database. NoSQL databases are very large and this can lead to different issues such as data quality and security. NoSQL databases aren't designed to remove data duplication, so this makes the database even larger with some of the data throwing off your queries because some of it could be repeated information. Security is another issue with these databases because it is hard to provide security for such a large database that is constantly changing.

Three NoSQL databases are Cassandra, HBase, and Dynamo. The NoSQL database Cassandra is used by many companies such as eBay, Facebook, Netflix, and Twitter. Cassandra is a column or wide-columned NoSQL database and it stores data in a column-based structure instead of a row-based structure like a Relational Database. It operates on a distributed system and uses data partitioning. The HBase is used by Adobe, Mozilla, and Netflix. The HBase is also a column or wide-columned NoSQL database like the Cassandra database. The last NoSQL database I will be talking about is the Dynamo database, and it is used by Amazon. The Dynamo is a key-value database that separates the data into certain keys and then each key is tied with something but it doesn't have to be chosen right away. For Amazon using this certain NoSQL database, you have to think about how large the database is and how much storage it contains. Amazon is the biggest online shopping store online and for Amazon to be using it along with other companies is truly remarkable.

TechTarget says, “a graph database is a NoSQL Database that stores data as a network graph. Graph databases are made up of nodes and edges, where nodes represent specific entities, while edges represent the connection between two nodes.” The difference between a graph database and a relational database is the way it stores the relationships between the entities. A graph database stores its relationships with records while relational database’s structure their relationships with table definitions. Graph databases don’t have a heavily-structured design, so all of the data has to be examined by itself to determine the structure. In a relational database, the structure is table based and that makes you tailor the data toward the table-based structure. Memgraph says, “the fundamental components of a graph database are nodes, relationships, labels, and properties. Nodes are the primary data points and some examples of nodes would be a person or place. The relationships refer to the connections between the nodes such as friendship, family, etc. The labels component is designed to identify the node. The last component is properties and these describe the node, and some examples of this could be name, nationality, sex, etc.

Graph databases have a lot of uses in our current economy and some industries that can use graph databases are financial services, manufacturing, government, data regulation and privacy, and marketing. Financial services can run queries on different customers and organizations to find out specific information on finances and to also check up on fraud. The graph databases ties all of the different data that goes into finances such as who the money is going to, what it is for, where the money is coming from, and other different relationships. Graph databases makes this a lot easier to quickly run a query to find out specific information. Graph databases are also used for manufacturing and this is helpful because it tracks materials, resources, personnel, etc. With the graph database, a project manager can run a query on how much they spent on a certain project or how much they spent on a certain material or person. This allows you to see what you have spent and potentially where you could save money through different suppliers. The next industry that could benefit from graph databases is the government. They could use them to detect tax fraud, get information on a criminal investigation, and contact tracing. Another industry that graph databases are useful for is cyber security or data regulation. The graph databases can track where the data is coming from and who is sending it. This allows you to block this security threat and similar threats like it. As you can see, graph databases are important to a lot of different organizations for a variety of different purposes, and a lot of organizations could potentially be using this tool to save money and time trying to figure out certain information that a graph database could easily show you once you get it set up.

Two types of graph databases are property graphs and RDF graphs. Property graphs are used to model relationships among data. The relationships tied with certain people or things provide the database with a wide variety of queries you can run to get different kinds of information. An RDF graph is a Resource Description Framework graph. The RDF graph is designed to publish data in a standard format with information that is related to one another. Both of these graph databases explore and analyze the relationships and behavior among data in the graph database. This allows you to discover hidden relations within your data and let you look at the metadata. There are two specific graph databases called JanusGraph and Neo4j. JanusGraph is an open-source database with integration that is used for big data analytics. Neo4j is a Network Exploration and Optimization 4 Java and this graph database has a high level or security and a lot of backups to secure your data in case of a power surge. Graph databases allow you to run queries easily without doing left or right joins like in SQL.

Some advantages of graph databases are establishing relationships with external sources, no joins required, query is dependent on tangible relationships, flexible and agile, and it is easily managed. Some disadvantages of graph databases are that they are slower, query language is platform dependent, and has a smaller user base. Overall, NoSQL databases and graph databases aren't trying to replace relational databases but trying to use different methods and programming languages to better analyze and compare data and results. I found this paper interesting to find out more about other databases rather than just the relational database.

Bibliography:

"NoSQL Database - What Is NoSQL? | Microsoft Azure." *Azure.microsoft.com*, azure.microsoft.com/en-us/resources/cloud-computing-dictionary/what-is-nosql-database/. Accessed 4 Sept. 2022.

"NoSQL Beginner Guide: Pros, Cons, Types, and Philosophy." *AltexSoft*, www.altexsoft.com/blog/nosql-pros-cons/.

Amini, Moreteza. *NoSQL Database Systems and Their Security Challenges NoSQL Database System and Their Security Challenges 12 Th ISC International Conference of Information Security and Cryptology (ISCISC'15) NoSQL Database Systems and Their Security Challenges. 2015*

"Graph Database vs. Relational Database: Key Differences." *SearchDataManagement*, www.techtarget.com/searchdatamanagement/feature/Graph-database-vs-relational-database-Key-differences

7 Use Cases for Graph Databases and Graph Analytics, *Oracle.com*

"What Is a Graph Database?" *Oracle.com*, 2021, www.oracle.com/autonomous-database/what-is-graph-database/#:~:text=There%20are%20two%20popular%20models%20of%20graph%20datbases%3a Accessed 6 Sept. 2022.

"Introduction to Graph Database on NoSQL." *GeeksforGeeks*, 25 Dec. 2021, www.geeksforgeeks,org/introduction-to-graph-database-on-nosql/. Accessed 6 Sept. 2022