# Bryan Wong

# Professor Mushkatblat

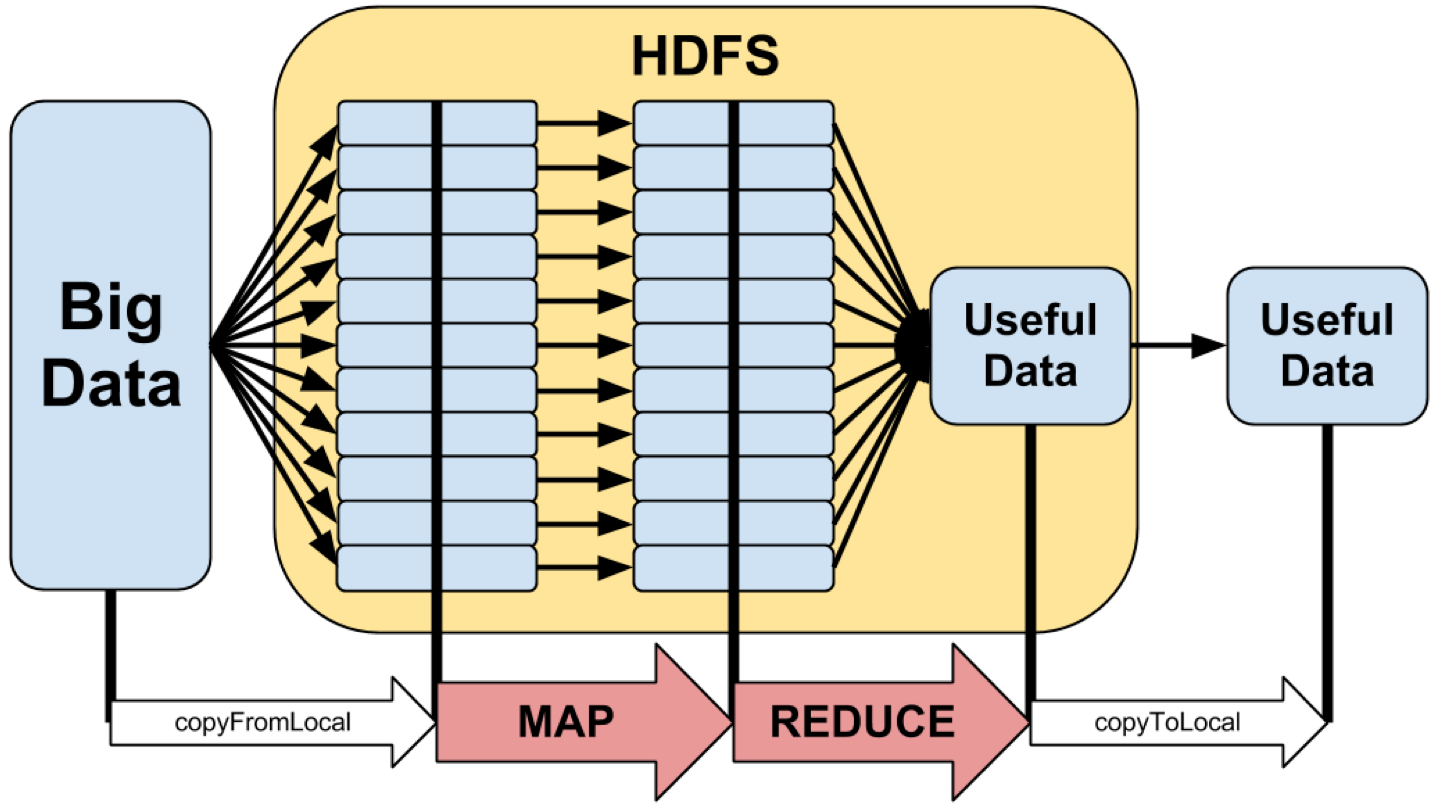
Spring 2017

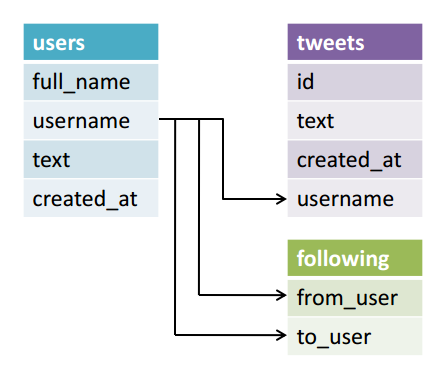
Final Exam Paper

Relational Database vs. Hadoop

Relational databases have been around for quite some time, since the 1970s when first proposed by E. F. Codd (Relational Model). Relational Database Management Software (RDBMS) is software used to maintain these relational databases. The type of computer language used in these systems is called SQL, which stands for Structure Query Language. Hadoop is an open source framework written in Java used for distributed processing and storage of very large set of data on clusters of computers. Differences between them will be discussed as well as in terms of is high-level architecture, ACID and CAP theorem, its functionalities and uses in the industry.

The database industry continues to flourish with innovations. Just in the last couple of years these database systems seem to scale new possibilities with revolutionized performance. These new possibilities include encryption, compression, the movement of select database logic into storage to speed functions such as row selection, compression which can dramatically increase the amount of data cached in memory and the speed of disk scans for large queries and the use of very large multi-terabyte flash memories.

Hadoop and Relational Database Management Systems (RDBMS) retrieve, process and store data differently. RDBMS have been around for quite some time whereas Hadoop is a new proven concept. The processing of all this data in a certain period of time become really important as customer data and storage capacities are growing enormously. Hadoop is an open source framework built on the language of Java. What’s great about it is, that it is quite scalable and can support demanding high performance applications. With the Hadoop framework, it is now possible to store very large amounts of data on the file systems of multiple computers. Up to thousands of computers can be connected as nodes and locally use their CPU, memory and processing power. If you are familiar with Google’s “MapReduce” algorithm then you might want to know that the Hadoop framework is based on it. What makes Hadoop popular for companies is that it can handle processing BIG data. Companies such as Facebook, AOL, IBM, ImageShack, and Yahoo are some of the big names that currently use Hadoop.

A RDBMS, is built on a Database Management System (DBMS) that stores information in the form of a table, which has rows and columns. The language SQL, as mentioned above is the language used to store, and retrieve the data from the tables. What makes a database relational is that is stores the relationships between these tables in various forms as column entries a table that will serve to reference another. An important feature of a rational database is that a single system has many tables and relationships between these tables so information is classified into their own entities. They are also stores in a normalized or simplified way while maintaining a relationship between tables with primary/foreign keys. The main difference between the two is that RDBMS works well with structured data, whereas Hadoop works well with both structures and unstructured data. Hadoop is ideal for processing large amount of data that cannot be defined in a relationship amongst other data while RDBMA works better when the data is defined such as data types, relationships among data and constraints.

Hadoop follows with the CAP theorem, where the system can’t guarantee Consistency, Availability and Partition Tolerance all at the same time. When developers realize that they can’t have all three at the same time, they will compromise and choose “Availability” and “Partition Tolerance” firs, which is commonly known as “eventual consistency”. Consistency is all the nodes see the same data at the same time, Availability is that every request receives a response about whether it succeeded or failed and Partition Tolerance is the system continues to operate despite arbitrary partitioning our to network failures. Relational databases follow the ACID theorem where “ACID” is a group of properties that guarantee that database transactions are processed concurrently. A stands for Atomicity, which requires each transaction to be “all or nothing”. C for Consistency, ensures that any transaction will bring the database from one valid state to another. I for Isolation, which ensures that the concurrent execution of transactions results in a system state that would be obtained if they were executed one after another, and last D for Durability, which ensures that once the transaction has been committed, it will remain so. Basically, what it does it makes sure that all the users can have the same view of the data.

An example use case of Hadoop is that many ideas that originated from the internet were behind the open source Hadoop, like Google and Yahoo. They used the system for their massive search engines that farms that search the web into local files. They use this data and process it, for you and I to be about to search it. These algorithms that are used by these search engines determine the information that closely matches the search terms and the ensure that the quality of the web page it delivers are specialized, sophisticated, and highly proprietary. Today, relational database are used just about everywhere you go. Once example would be a patient record system at a nearby hospital. Where the hospital keeps tracks of each patient and each patient has its own unique set a data. Items such as patients address to hair color, and body weight. All this data is keep in tables in a database for the employees of the hospital to quickly search for.

Both Hadoop and RDBMS are similar in the way they collect information from massive server farms, but each have their own differences to make them efficient in specific applications. Hadoop would be ideal for a business that deals with search engines and the speed of access to the information, companies like Google and Yahoo. RDBMS are times faster than Hadoop when it relates to querying predefined types, more notably like banking systems and small businesses, for speed and reliability. It comes down to what type of job needs to be accomplished, and how efficient and reliable it can be.

Works Cited

"ACID Implementation." Wikipedia. Wikimedia Foundation, n.d. Web. 18 May 2016.

Brewer, Eric. "CAP Twelve Years Later: How the "Rules" Have Changed." InfoQ. Computer Magazine, n.d. Web. 18 May 2016.

"CAP Theorem." Wikipedia. N.p., n.d. Web. 18 May 2016.

Cassandra. "CAP Theorem - Hadoop Online Tutorials." Hadoop Online Tutorials. Hadoop Online Tutorial, 27 Jan. 2016. Web. 18 May 2016.

"Difference between Hadoop and RDBMS." WikiDifference. N.p., n.d. Web. 18 May 2016.

Mishra, Smita. "Understanding Storage in the Hadoop Cluster." Understanding Storage in the Hadoop Cluster. SearchSolidStateStorage, 14 Aug. 2015. Web. 18 May 2016.

Oracle. "1 Introduction to Oracle Database." Introduction to Oracle Database. Oracle, n.d. Web. 18 May 2016.

Oracle. "Hadoop and NOSQL Technologies and the Oracle DatabaseO." Hadoop and NoSQL Technologies and the Oracle Database (n.d.): n. pag. Oracle. An Oracle White Paper, Feb. 2011. Web.

"Relational Database." Wikipedia. Wikimedia Foundation, n.d. Web. 18 May 2016.

"Relational Databases for Dummies." Code Envato Tuts+. N.p., n.d. Web. 18 May 2016.