Emerging Technologies of Big Data

Mrs Vidyullata Shekhar Jadhav
Computer Application
V.P. Institute of Management Studies & Research, Sangli
vidyullatap@gmail.com

Abstract:
Big data is an all encompassing term for any collection of data sets so large or complex that it becomes difficult to process them using traditional data processing applications. Big data has three key strategic and operational challenges which include Information Strategy: need to tie together the power of information assets. Big data is causing enterprises to find new ways to leverage information sources to drive growth. There are number of emerging technologies that are helping users cope with and handle Big Data in a cost-effective manner these includes Column oriented databases, Map Reduce, Hadoop, Hive, Pig , Platfora

Keyword- Big data, Column oriented databases, Map Reduce, Hadoop, Hive, Pig,Platfora

I. INTRODUCTION

Big data is an all encompassing term for any collection of data sets so large or complex that it becomes difficult to process them using traditional data processing applications. Big data has three key strategic and operational challenges which include Information Strategy: need to tie together the power of information assets. Big data is causing enterprises to find new ways to leverage information sources to drive growth. Data Analytics: need to draw more insight from your big data analytics or large and complex datasets. It also needs to predict future customer behaviors, trends and outcomes. Enterprise Information Management: Information is everywhere – volume, variety, velocity – and it keeps growing. You need to manage access to growing extreme information management requirements and drive innovation in rapid information processing. There are number of emerging technologies that are helping users cope with and handle Big Data in a cost-effective manner.

II. EMERGING TECHNOLOGIES

A. Column-oriented databases

Column oriented databases generally work on columns. All columns are treated individually. Values of single column are stored contiguously. The goal of a Column-oriented databases is to efficiently write and read data to and from hard disk storage in order to speed up the time it takes to return a query. In a Column-oriented databases, all the column 1 values are physically together, followed by all the column 2 values, etc. Following example shows how data is stored in column oriented database.

Suppose a sample database table student with information

Roll_no	Class	Name
1	MCA I	Raj
2	MCA II	Ram
3	MCAI	Geeta
4	MCAII	Datta

Then in Column-oriented databases the data would be stored like this:1, 2, 3, 4;MCA I,MCA II,MCAI,MCAIII; Raj,Ram, Geeta, Datta;

Advantages of column oriented technology are

- High performance on aggregation queries (like COUNT, SUM, AVG, MIN, MAX)
- Highly efficient data compression and/or partitioning
- True scalability and fast data loading for Big Data
- Accessible by many 3rd party BI analytic tools
- Fairly simple systems administration

B. Map Reduce

Map Reduce is a programming model and an associated implementation for processing and generating large data sets. Users specify a map function that processes a key/value pair to generate a set of intermediate key/value pairs, and a reduce function that merges all intermediate values associated with the same intermediate key. Any Map Reduce implementation consists of two tasks:

- The "Map" task, where an input dataset is converted into a different set of key/value pairs, or tuples;
- The "Reduce" task, where several of the outputs of the "Map" task are combined to form a reduced set of tuples.

Simple map reduce flow contains following steps

- The input data can be divided into n number of chunks depending upon the amount of data and processing capacity of individual unit.
- Next, it is passed to the mapper functions. Please note that all the chunks are processed simultaneously at the same time, which embraces the parallel processing of data.
- After that, shuffling happens which leads to aggregation of similar patterns.
- Finally, reducers combine them all to get a consolidated output as per the logic.

This algorithm embraces scalability as depending on the size of the input data, we can keep increasing the number of the parallel processing units.

C. Hadoop

Hadoop is a core component of a Modern Data Architecture, allowing organizations to collect, store, analyze and manipulate massive quantities of data on their own terms—regardless of the source of that data, how old it is, where it is stored, or under what format. Hadoop consist of number of modules and each is designed with a fundamental assumption that hardware failures are commonplace and thus should be automatically handled in software by the framework.Following are some common modules:

- Hadoop Common: The common utilities that support the other Hadoop modules.
- Hadoop Distributed File System (HDFS™): A distributed file system that provides high-throughput access to application data.
- Hadoop YARN: A framework for job scheduling and cluster resource management.
- Hadoop MapReduce: A YARN-based system for parallel processing of large data sets.

D. Hive

Hive is data warehouse software facilitates querying and managing large datasets residing in distributed storage. Hive provides a mechanism to project structure onto this data and query the data using a SQL-like language called HiveQL. At the same time this language also allows traditional map/reduce programmers to plug in their custom mappers and reducers when it is inconvenient or inefficient to express this logic in HiveQL.

Hive, allows SQL developers to write Hive Query Language (HQL) statements that are similar to standard SQL statements; now you should be aware that HQL is limited in the commands it understands, but it is still pretty useful. HQL statements are broken down by the Hive service into MapReduce jobs and executed acros a Hadoop cluster.

For anyone with a SQL or relational database background, this section will look very familiar to you. As with any database management system (DBMS), you can run your Hive queries in many ways. You can run them from a command line interface (known as the Hive shell), from a Java Database Connectivity (JDBC) or Open Database Connectivity (ODBC) application leveraging the Hive JDBC/ODBC drivers, or from what is called a Hive Thrift Client. The Hive Thrift Client is much like any database client that gets installed on a user's client machine (or in a middle tier of a three-tier architecture): it communicates with the Hive services running on the server. You can use the Hive Thrift Client within applications written in C++, Java, PHP, Python, or Ruby (much like you can use these client-side languages with embedded SQL to access a database such as DB2 or Informix).

Hive looks very much like traditional database code with SQL access. However, because Hive is based on Hadoop and MapReduce operations, there are several key differences. The first is that Hadoop is intended for long sequential scans, and because Hive is based on Hadoop, you can expect queries to have a very high latency (many minutes). This means that Hive would not be appropriate for applications that need very fast response times, as you would expect with a database such as DB2. Finally, Hive is read-based and therefore not appropriate for transaction processing that typically involves a high percentage of write operations.

E. Pig

Pig was initially developed at Yahoo! to allow people using Hadoop® to focus more on analyzing large data sets and spend less time having to write mapper and reducer programs. Like actual pigs, who eat almost anything, the Pig programming language is designed to handle any kind of data—hence the name!

Pig is made up of two components: the first is the language itself, which is called PigLatin, and the second is a runtime environment where PigLatin programs are executed.

This course begins with an overview of Pig. It explains the data structures supported by Pig and how to access data using the LOAD operator. The next lesson covers the Pig relational operators. This is followed by the Pig evaluation functions, as well as math and string functions.

Steps to use Pig Program

1. The first step in a Pig program is to LOAD the data you want to manipulate from HDFS.
2. Then you run the data through a set of transformations (which, under the covers, are translated into a set of mapper and reducer tasks).
3. Finally, you DUMP the data to the screen or you STORE the results in a file somewhere.

F. Platfora

Platfora was founded in 2011 by Ben Werther. Werther studied computer science at Stanford University.[1] Prior to founding Platfora, he worked at There Inc., Siebel Systems, Microsoft, and Greenplum. Platfora is one of several new big data analytics companies that industry analysts expect to compete with established firms including SAP, IBM, SAS, and Oracle, whose older methods of data analysis and visualization are currently more time consuming.

WibiData- WibiData is a combination of web analytics with Hadoop. It allows web sites to better explore and work with their user data, enabling real-time responses to user behavior, such as serving personalized content, recommendations and decisions.

III. REFERENCES

[1] Introduction-to-pi. (n.d.). Retrieved december 21, 2014, from bigdatauniversity.com: http://bigdatauniversity.com/bdu-wp/bdu-course/introduction-to-pig/

[2] 10-emerging-technologies-for-big-data. (2012, December 4). Retrieved December 5, 2014, from

http://www.techrepublic.com: http://www.techrepublic.com/blog/big-data-analytics/10-emerging-technologies-for-big-data/

[3] Anderson, P. b. (2012, July 24). column-oriented-database-technologies. Retrieved December 13, 2014, from www.dbbest.com: http://www.dbbest.com/blog/column-oriented-database-technologies/

[4] apache. (n.d.). Apache Hive TM. Retrieved december 21, 2014, from hive.apache.org: https://hive.apache.org/

[5] emerging technologies for Big Data. (2014, March 14). Retrieved December 15, 2014, from cio.economictimes.indiatimes.com: http://cio.economictimes.indiatimes.com/news/big-data/10-emerging-technologies-for-big-data/32791624

[6] Ghemawat, J. D. (2004.). MapReduce: Simplified Data Processing on Large Clusters . Sixth Symposium on Operating System Design and Implementation , 3-13.

[7] hive. (n.d.). Retrieved DECEMBER 21, 2014, from www-01.ibm.com: http://www-01.ibm.com/software/data/infosphere/hadoop/hive/

[8] map-reduce-algorithm. (2014, May 19). Retrieved December 13, 2014, from www.thegeekstuff.com: http://www.thegeekstuff.com/2014/05/map-reduce-algorithm/

[9] pig. (n.d.). Retrieved December 21, 2014, from www-01.ibm.com: http://www-01.ibm.com/software/data/infosphere/hadoop/pig/

[10] Platfora. (2014, December 16). Retrieved December 21, 2014, from en.wikipedia.org: http://en.wikipedia.org/wiki/Platfora

[11] Rouse, M. (2005). columnar-database. Retrieved 2015, from searchdatamanagement.techtarget.com: http://searchdatamanagement.techtarget.com/definition/columnar-database

[12] What Is Apache Hadoop? (2014, December 12). Retrieved december 21, 2014, from hadoop.apache.org: http://hadoop.apache.org/#What+Is+Apache+Hadoop%3F