International Journal of Electrical Electronics & Computer Science Engineering
Special Issue - TeLMISR 2015, ISSN : 2348-2273
Available Online at www.ijeecse.com

Business Intelligence by Cloud Computing Using Hadoop Single Node Cluster

Ritu Aggrawal[1], Gautam Kumar[2]
[1]ritu.aggrawal78@gmail.com, [2]gkbhardwaj88@gmail.com

Abstract: ***Today in IT industry data is following as a wave of flood. The data can be categorised by structured, semi-structured and unstructured, collectively such type of data is know as Big Data. The data is the king of knowledge discovery process but to manage storage and analysis of big data is too expansive and complex. One of the great solutions to do business intelligence by cloud computing to reduce the cost and ease of data center to reduce workload. The hadoop is the big data analysis platform that is configured as single node cluster on amazon web services(AWS) to seggregate business value data and move the data in cloud. I demonstrated the work of Hadoop on AWS with the help of customer social Behaviour Application (CBA).***

Keywords: AWS, Hadoop, Big data, Cloud computing, BI, CBA

I. INTRODUCTION

The cloud computing is one of the emerging technology to do business intelligence in cost effective way which work on the concept of "pay-per-use". The source of big data are social media, sensor data, system log files, web log files etc. Almost all industry facing the problem storage management, segregation of business value data from big data, accelerating query on data and cost to manage infrastructure of big data. IBM SmartCloud offering the software- as -a – services to analysis of social media data but it is too expansive than doing business intelligence in house-premise[1].

Hadoop is a single leader in big data platform management that is based on cluster computing. San Francisco, manages a private Hadoop cluster for data processing but it is too expansive[1]. The implementation of hadoop single node cluster on CloudStack may reduce the cost to do big data analysis on cloud computing.

II. BIG DATA

The Big data warehouses, webpages log files, blogs, tweets, audio and video streams are generating a massive amount of complex data. Such types of data is known as big data Discovering useful knowledge from huge datasets requires smart and scalable analytics services, programming tools, and applications. Big data analysis uses data mining algorithms to discover useful information. Cloud computing can be served as both data analysis and data storage of big data analytical application (BDA'Apps) [2].

III. BIG DATA HISTORY

The history of big data started many years ago before the latest buzz word big data in the IT industry. In 1941 first time encountered the rate of growth in volume of data. Below is few milestones of data becomes big year by year.

IV. CLOUD BASED BIG DATA ANALYTICS

Big data refers to large amount of data sets that can not be managed by the conventional relational database system. It is measured in 3v (volume, velocity, varity). Combining bid data analytical and knowledge discovery techniques can generates new insights in shorter times. Few cloud-based big data analytics tools are available and becomes common within few years. There are two current solution available in the market today. One solution is based on open source and others are proprietary solutions provided by companies such as Google, IBM, EMC, BigML, Splunk Storm, Kognitio, and InsightsOne. As more such platforms emerge, but open source platform is most prevalent and useful. One of the open source solution is apache Hadoop.

V. BIG DATA ANALYTICS SERVICE MODEL

There are three big data analytics service model named software as a service (SaaS), platform as a service (PaaS) and Infrastructure as a service (IaaS) to implement big data analytics in cloud environment.

A. *Big data analytics software as a service:* provides th complete advanced data mining algorithms and knowledge discovery tools as a service to the end user over the Internet.

B. *Big data analytics platform as a service:* provides platform to build big data analytics application.

C. *Bid data analytics Infrastructure as a service:* provides virtualized resources to run data analytics application.

Big data cloud service model	Features	Users
Big Data analytics software as a service	Single and complete data mining application offered as a service	End users, analytics managers, data analysts

Big data cloud service model	Features	Users
Big data analytics platform as a service	Data analysis suite for programming or developing high-level applications.	Data mining application developers, data scientists
Big data analytics infrastructure as a service	A set of virtualized resources for running data analysis applications	Data mining programmers, data management developers, data mining researchers

VI. HADOOP ECOSYSTEM

Hadoop is an open source software developed by Apache Software Foundation. It is basiclly know for MapReduce programming and Hadoop distributed file system(HDFS). It is used for large scale data set processing. There are following component of Hadoop.

A. *MapReduce:* A distributed data processing model and execution environment that runs on large clusters of commodity machines.

B. *HDFS:* A distributed filesystem that runs on large clusters of commodity machines.

C. *Pig:* A data flow language and execution environment for exploring very large datasets.Pig runs on HDFS and MapReduce clusters.

D. *Hive:* A distributed data warehouse. Hive manages data stored in HDFS and provides a query language based on SQL for querying the data.

E. *HBase:* A distributed, column-oriented database. HBase uses HDFS for its underlying storage, and supports both batch-style computations using MapReduce and point queries (random reads).

F. *ZooKeeper:* A distributed, highly available coordination service. ZooKeeper provides primitives such as distributed locks that can be used for building distributed applications.

G. *Sqoop:* A tool for efficiently moving data between relational databases and HDFS.

VII. HADOOP DISTRIBUTED FILE SYSTEM

HDFS is a file system designed for storing very large files with streaming data access patterns, running on clusters of commodity hardware. The default data block size of HDFS is 64MB. An HDFS cluster has two types of node operating in a master-worker pattern: a namenode (the master) and a number of datanodes (slaves). The namenode manages the file system namespace. It maintains the file system tree and the metadata for all the files and directories in the tree. The namenode also knows the datanodes on which all the blocks for a given file are located, however, it does not store block locations persistently, since this information is reconstructed from datanodes when the system starts. They store and retrieve blocks when they are told to (by clients or the namenode), and they report back to the namenode periodically with lists of blocks that they are storing. Name Node decides about replication of data blocks. In a typical HDFS, block size is 64MB and replication factor is 3 (second copy on the local rack and third on the remote rack). The Fig 2 shown architecture distributed file system HDFS [3].

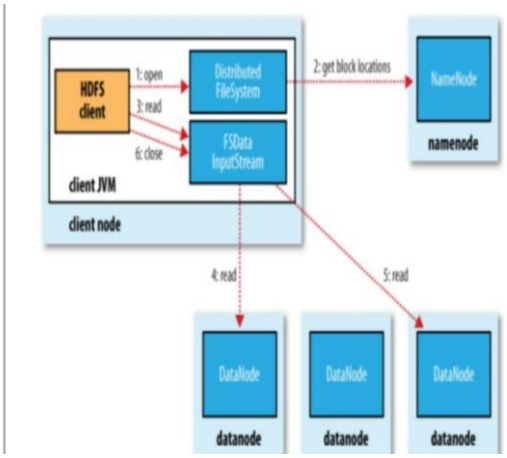


Fig 2 HDFS [3].

VIII. MAPREDUCE PROGRAMMING MODEL

MapReduce is a data processing or parallel programming model introduced by Google. In this model, a user specifies the computation by two functions, Map and Reduce. In the mapping phase, MapReduce takes the input data and feeds each data element to the mapper. In the reducing phase, the reducer processes all the outputs from the mapper and arrives at a final result. The figure3 [4] shows the process of map-reduce programming.

At the highest level in classic map-reduce a job runs with the four independent entities [3]:

*The client***,** which submits the MapReduce job.

The jobtracker, which coordinates the job run. The jobtracker is a Java application whose main class is JobTracker .

The tasktrackers, which run the tasks that the job has been split into. Tasktrackers are Java applications whose main

class is TaskTracker .

The distributed filesystem (normally HDFS), which is used for sharing job files between the other entities. The figure4[3] shows the complete process of map-reduce programming model.

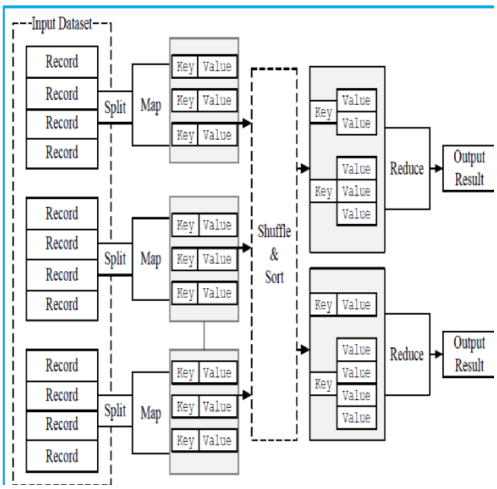


Fig 3 MapReduce programming model

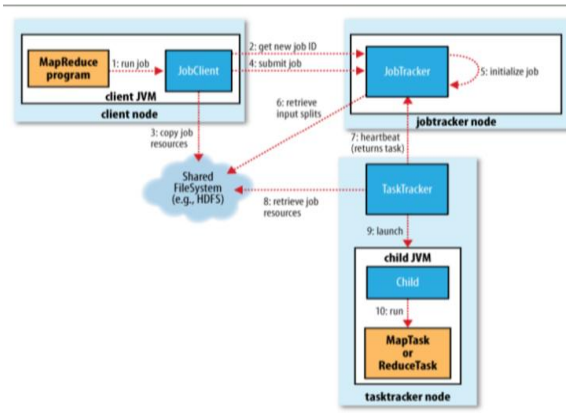


Fig 4 Architecture of MapReduce

There are following steps of MapReduce process execution:

Job Submission: The submit() method on Job creates an internal JobSummitter instance and calls sub mitJobInternal() on it. When the job is complete, if it was successful, the job counters are displayed. Otherwise, the error that caused the job to fail is logged to the console. The job submission process implemented by JobSummitter does the following:

- Asks the jobtracker for a new job ID
- Checks the output specification of the job.
- Computes the input splits for the job.
- Copies the resources needed to run the job, including the job JAR file, the configuration file, and the computed input splits, to the jobtracker's filesystem in a directory named after the job ID.
- Tells the jobtracker that the job is ready for execution.

Job Initialization: Initialization involves creating an object to represent the job being run, which encapsulates its tasks, and bookkeeping information to keep track of the tasks' status and progress. To create the list of tasks to run, the job scheduler first retrieves the input splits computed by the client from the shared filesystem.

Task Assignment: Tasktrackers run a simple loop that periodically sends heartbeat method calls to the jobtracker. Heartbeats tell the jobtracker that a tasktracker is alive, but they also double as a channel for messages. As a part of the heartbeat, a tasktracker will indicate whether it is ready to run a new task, and if it is, the jobtracker will allocate it a task, which it communicates to the tasktracker using the heartbeat return value.

Task Execution: First, it localizes the job JAR by copying it from the shared filesystem to the tasktracker's filesystem. It also copies any files needed from the distributed cache by the application to the local disk; Second, it creates a local working directory for the task, and un-jars the contents of the JAR into this directory. Third, it creates an instance of TaskRunner to run the task.

Progress and Status Updates: the task reports its progress and status (including counters) back to its application master every three seconds.

*Job Completion:*As well as polling the application master for progress, every five seconds the client checks whether the job has completed. The figure5[4] shows the sequence diagram of map-reduce working model.

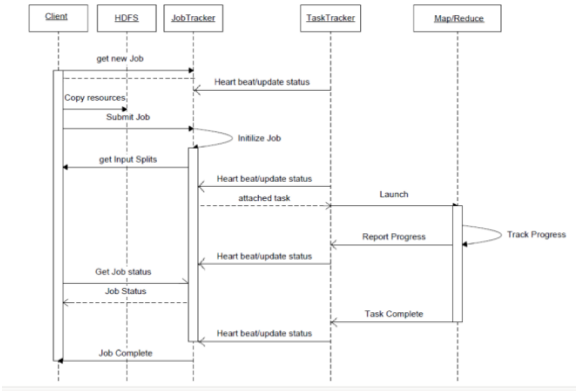
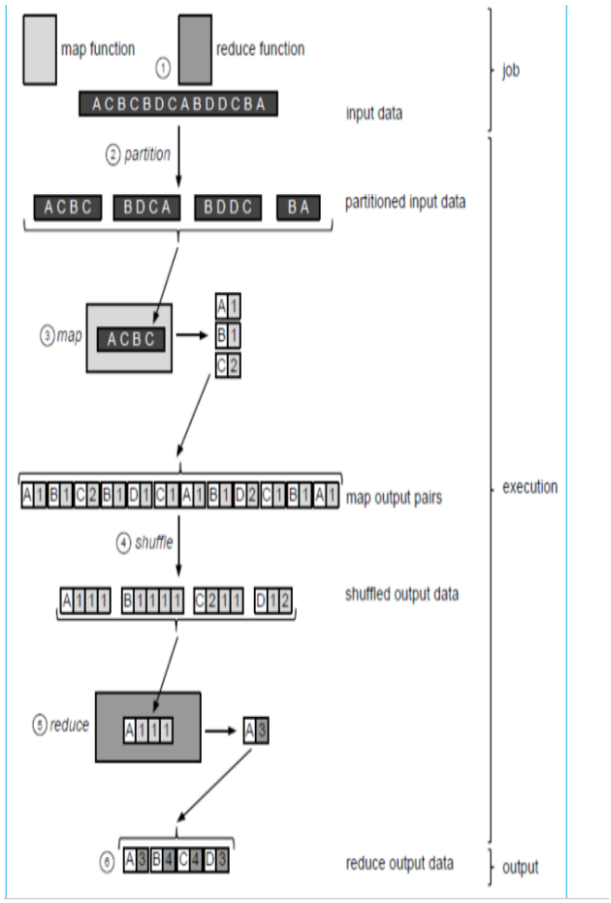


Fig 5- Sequence diagram of MapReduce model

IX. MAPREDUCE PROGRAMMING EXAMPLE OF WORD COUNT

A simple word count program using MapReduce counts the occurrence of word in a text. The figure 6 [4] shows the working of a word count program using MapReduce. It involves the following steps.

- Input data
- Partitions the data into 'n' no of blocks.
- For each block of data map function runs and extracts the collection of key-values pairs.
- The collected key-values pairs is shuffled for each



distinct keys and creates a collection containing corresponding values of each keys.

- For each key-collection a reduce function is applied and extracts a single keys-values for each key-collection.
- The collection of all key-values pairs is the output of MapReduce programm.

X. APPLICATION OF HADOOP

- Building search index at Google, Amazon, Yahoo
- Analyzing user logs, data warehousing and analytics
- Used for large scale machine learning and data mining applications
- Legacy data processing where it requires massive computational

XI. CONCLUSION

With the advancement in cloud computing big data analytics requires new tools, model and technology to implement the data mining and analysis algorithms in the cloud enviornment.

Big data analysis in cloud computing can reduce the cost to maintain infrastructure and storage.

It is very scalable to do business analytics in cloud enviornment on CloudStack.

XII. REFERENCES

[1]. In the Cloud, Big Data's a Big Deal, a SearchCloudComputing.com e-publication by www.techtarget.com.

[2]. Clouds for Scalable Big Data Analytics Domenico Talia University of Calabria, Italy, IEEE Computer society, 0018-9162/13/$31.00 © may 2013 IEEE.

[3]. Hadoop definitive guide 3rd edition.

[4]. Big Data Processing with Hadoop-MapReduce in Cloud Systems, International Journal of Cloud Computing and Services Science (IJ-CLOSER) Vol.2, No.1, February 2013, pp. 16~27 ISSN: 2089-3337.

[5]. Clouds for Scalable Big Data Analytics, Domenico Talia University of Calabria, Italy.

[6]. Apache CloudStack Cloud Computing Copyright © 2013 Packt Publishing, SBN 978-1-78216-010-6

[7]. Big Data Imparative Book by Soumendra Mohanty and Madhu Jagadeesh.

[8]. Retrived from http://www.ibm.com/IBM ReadBook.