IMPACT: International Journal of Research in Applied, Natural and Social Sciences (IMPACT: IJRANSS)
ISSN(P): 2347-4580; ISSN(E): 2321-8851
Vol. 4, Issue 8, Aug 2016, 101-106

HADOOP: FUTURE TALK ABOUT BIG DATA STORAGE

KAILASH CHANDER[1] & POONAM[2]

[1] Research Associate, Department of Science & Technology, HARSAC, Government of Haryana, India

[2] Resource Person, Department of Computer Science, Government P.G. College, Jind, Haryana, India

ABSTRACT

This paper presents a review on modern technology called HADOOP, which is used for managing very large amount of data. Multi Peta-byte Data sets becomes challenge for companies to process effectively and efficiently. Conversation about Big Data for very long without running into the elephant in the room is not possible. It is complex to have the data at distributed locations to process. For this a solution is needed i.e. open Source Apache License: HADOOP. It stores enormous data sets across distributed clusters of servers and then running "distributed" analysis applications in each cluster. Data applications will continue to run even when individual servers or cluster fails. Hadoop is almost completely modular, that allow swap out almost any of its components for a different software tool due to the flexibility of architecture.

KEYWORDS: Hadoop, Mapreduce, HDFS, Petabyte, Hortonworks, Sqoop

INTRODUCTION

Hadoop is an open source framework for processing, storing and analyzing massive amounts of distributed unstructured data. Originally created by Doug Cutting at Yahoo®, Hadoop was inspired by MapReduce, a user-defined function developed by Google in early 2000s for indexing the Web. It was designed to handle petabytes and Exabyte's of data distributed over multiple nodes in parallel. Hadoop is now a project of the Apache Software Foundation, where hundreds of contributors continuously improve the core technology. Fundamental concept: Rather than banging away at one, huge block of data with a single machine, Hadoop breaks up Big Data into multiple parts so each part can be processed and analyzed at the same time. Hadoop is used for searching, log processing, recommendation systems, analytics, video and image analysis, data retention? It is used by the top level apache foundation project, large active user base, mailing lists, users groups, very active development, and strong development teams.

COMPONENTS OF HADOOP

The Hadoop is consists of various components as described below:

Hadoop Distributed File System

HDFS, the storage layer of Hadoop, is a distributed, scalable, Java-based file system adept at storing large volumes of unstructured data.

MapReduce

MapReduce is a software framework that serves as the compute layer of Hadoop. MapReduce jobs are divided into two (obviously named) parts. The “Map” function divides a query into multiple parts and processes data at the node level. The “Reduce” function aggregates the results of the “Map” function to determine the “answer” to the query.

Hive

Hive is a Hadoop-based data warehousing-like framework originally developed by Facebook. It allows users to write queries in a SQL-like language called HiveQL, which are then converted to MapReduce. This allows SQL programmers with no MapReduce experience to use the warehouse and makes it easier to integrate with business intelligence and visualization tools such as Microstrategy, Tableau, Revolutions Analytics, etc. Pig Latin is a Hadoop-based language developed by Yahoo. It is relatively easy to learn and is adept at very deep, very long data pipelines (a limitation of SQL)

HBase

HBase is a non-relational database that allows for low-latency, quick lookups in Hadoop. It adds transactional capabilities to Hadoop, allowing users to conduct updates, inserts and deletes. EBay and Facebook use HBase heavily.

Flume

Flume is a framework for populating Hadoop with data. Oozie is a workflow processing system that lets users define a series of jobs written in multiple languages such as Map Reduce, Pig and Hive then intelligently link them to one another.

Sqoop

Sqoop is a connectivity tool for moving data from non-Hadoop data stores such as relational databases and data warehouses into Hadoop.

HCatalog

HCatalog is a centralized metadata management and sharing service for Apache Hadoop.

BigTop

BigTop is an effort to create a more formal process or framework for packaging and interoperability testing of Hadoop’s sub-projects and related components with the goal improving the Hadoop platform as a whole.

Oozie

Oozie allows users to specify, for example, that a particular query is only to be initiated after specified previous jobs on which it relies for data are completed. Flume is a framework for populating Hadoop with data.

Ambari

Ambari is a web-based set of tools for deploying, administering and monitoring Apache Hadoop clusters. Its development is being led by engineers from Horton works, which include Ambari in its Horton works Data Platform.

Avro

Avro is a data serialization system that allows for encoding the schema of Hadoop files. It is adept at parsing data and performing removed procedure calls. Mahout is a data mining library. It takes the most popular data mining algorithms for performing clustering, regression testing and statistical modeling and implements them using the Map Reduce model.

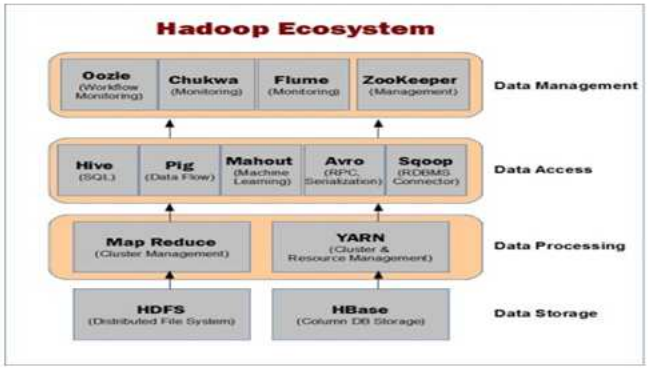

Figure 1: Components of Hadoop

WORKING OF HADOOP ARCHITECTURE

Hadoop is designed to run on a large number of machines that do not share any memory or disks. That means we can buy a whole bunch of commodity servers, slap them in a rack, and run the Hadoop software on each one. When we need to load all of the organization's data into Hadoop, the software bust that data into pieces and then spreads across different servers. There is no single place where the residency of data can be predicted. Hadoop keeps track of where the data resides. As there are multiple copy stores, if the server that goes offline or dies, it can be automatically replicated from a known good copy. In a centralized database system, it is similar to one big disk connected to four or eight or 16 big processors. But that is as much horsepower as you can bring to bear. In a Hadoop cluster, every one of those servers has two or four or eight CPUs. The indexing job can be run by sending the code to each of the dozens of servers in the cluster, and each server operates on its own little piece of the data.

Results are then delivered back in a unified whole. Architecturally, the reason to deal with lots of data is because Hadoop spreads it out. And the reason for asking complicated computational questions is because of all of these processors, working in parallel, harnessed together. Hadoop implements a computational paradigm named Map/Reduce, where the application is divided into many small fragments of work, each of which may be executed or re-executed on any node in the cluster. In addition, it provides a distributed file system (HDFS) that stores data on the compute nodes, providing very high aggregate bandwidth across the cluster. Both Map/Reduce and the distributed file system are designed so that node failures are automatically handled by the framework. Hadoop Common is a set of utilities that support the other Hadoop subprojects. Hadoop Common includes File System, RPC, and serialization libraries. The Fig-2 shows a muti-node Hadoop cluster structure.

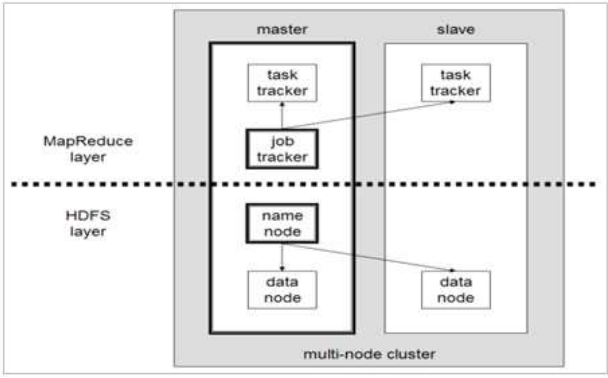

Figure 2: Multi-Node Hadoop Cluster

REQUIREMENT OF HADOOP

The Hadoop technology can be implemented in numerous applications. But some of the major applications are listed below:

- Batch data processing, not real-time / user facing (e.g. Document Analysis and Indexing, Web Graphs and Crawling).
- Highly parallel data intensive distributed applications.
- Very large production deployments (GRID) process lots of unstructured data. When the processing can easily be made parallel.

HADOOP USERS

The companies like Adobe®, Alibaba®, Amazon®, AOL, Facebook, Google and IBM are using Hadoop technology. Apache, Cloudera® and Yahoo® are the major contributors for Hadoop.

CONCLUSIONS

The Hadoop Distributed File System (HDFS) is a distributed file system designed to run on commodity hardware. Hadoop is designed to run on cheap commodity hardware, it automatically handles data replication and node failure. It does the hard work – we can focus on processing of data, Cost Saving and efficient and reliable data processing. It has many similarities with existing distributed file systems. However, the differences from other distributed file systems are significant. HDFS is highly fault-tolerant and is designed to be deployed on low-cost hardware. HDFS provides high throughput access to application data and is suitable for applications that have large data sets. HDFS relaxes a few POSIX requirements to enable streaming access to file system data. HDFS was originally built as infrastructure for the Apache Nutch web search engine project. HDFS is part of the Apache Hadoop Core project.

REFERENCES

1. Apache Hadoop! *(hadoop.apache.org)*
2. Hadoop on Wikipedia *(http://en.wikipedia.org/wiki/Hadoop)*
3. Free Search by Doug Cutting *(http://cutting.wordpress.com)*

4. Hadoop and Distributed Computing at Yahoo! *(http://developer.yahoo.com/hadoop)*
5. Apache Hadoop for the Enterprise *(http://www.cloudera.com)*
6. Siones, M. Tim (6 December 2011). "Scheduling in Hadoop". *ibm.com. IBM*. 20 November 2013.
7. Jones, M. Tim (6 December 2011). "Scheduling in Hadoop". *ibm.com. IBM*. 20 November 2013.
8. Murthy, Arun (2012-08-15). "Apache Hadoop YARN – Concepts and Applications". *hortonworks.com. Hortonworks*. 2014-09-30.
9. "Version 2.0 provides for manual failover and they are working on automatic failover:" *Hadoop.apache.org.*30 July 2013.
10. "Amazon Elastic MapReduce Now Supports Spot Instances". *Amazon.com. 2011-08-18.* 2013-10-17. "Apache Accumulo User Manual: Security". *apache.org. Apache Software Foundation.* 2014-12-03.
11. "Refactor the scheduler out of the JobTracker". *Hadoop Common. Apache Software Foundation.* 9 June 2012.
12. *www.tutorialspoint.com/hadoop/*
13. Andrew S. Tanenbaum Computer Networks Prentice Hall PTR New Jersey 2003
14. Ramjee Prasad An Introduction to OFDM John Wiley & Sons New York 2001.