A Catholic Research on Big Data and Hadoop Environment

Sushma Lakshkar
M.Tech Student
RCEW Jaipur

Geet Kalani
Assistant Professor
RCEW Jaipur

Vinod Todwal
Assistant Professor
RCEW Jaipur

ABSTRACT

In recent years, there are various sources that generates data in petabyte and terabyte, known as big data and its generated by human, machine, sensor etc. So the solution leds with big data is Apache Hadoop has attracted strong attention because of its applicability on processing for the large data sets. This paper present the review about the big data and it's characteristics and also the types of the open source tools environment like HADOOP. The objective of paper to identify the power of the Hadoop on the big data and motivation behind the new research and outlines to address to Apache Hadoop also includes the programming paradigm that is Map-Reduce.

General Terms

Big Data, Map Reduce, Hadoop Distributed File System

Keywords

Bigdata, ApacheHadoop, Map-Reduce,distributed systemHDFS,MPI,G-Hadoop,Gfram,3VBigDatamodel, 5 VBigData model.

1. INTRODUCTION

In the area of advance computing, efficient data storage mechanism, high speed processing, quick accessing and large dataset manipulation techniques are some of the challenging task. Interestingly, the speed of generating new datasets has been increased due to implementation of worldwide web services. In spite of these web facilities, globaly adapted social networking and immense business aspirations are two other significant aspects of big data generation in this century. One of the much acclaimed social networking websites i.e. Facebook producing as well as handling an impressive digital data on daily basis. The international enterprises i.e. Google, Yahoo and Amazon are well known for their services and managing the valuable public information. These information have been stored either in structured or in non-structured data formats. Nowadays, the very well acknowledged term 'Big Data' has become more meaningful and insightful for these enterprises. In fact, big data is about more than just the "bigness" of the data [10]. Notably, the Big Data has been characterized by conceptual models i.e. 3v and 5v models. The former 3v model only depicts about the speed, type and value, whereas letter is about speed, value, size, type and accuracy of the Big Data. In the field of Big Data sciences the term 'data analysis' is concerned about the utility and effectual processing of the stored or generated information in real time or post time frameworks. The data processing capabilities highly depends on the acceptable time (i.e. scheduling of the task), integrity as well as consistency of data, constrained resources and a specific problem index. The parallel and distributed computing provides a better solution in the Big Data storage, manipulation, and processing. Multitudinous algorithmic solutions and frameworks have been released to facilitate a high level compatibility to the users [1].The open source is Hadoop, based on the novel approach developed by the 'Doug Cutting' and 'Mike Cafarella'in the year of 2005[14].The Hadoop facilitates a congested solution package to data storage, manipulation, and retrieval of the information from the large datasets. The Hadoop framework received a great attention because of its inescapable properties such as scalability, data integrity, fault tolerance, failure recovery due to replication of the files and ease-to-use way. The Message Passing Interface (MPI) is a well-known programming paradigm to deal with in parallel computing environment.

2. LITERATURE REVIEW

2.1 Big Data

Big Data contains large-volume, complex and growing large data sets with multiple, self-governed sources. Big data processing is the fast growth of demands on storage, computation and communication in data centers [15]. Big data is result of globalization of the business, partnerships, value networks, social networks, and the large amount of information flow across and within enterprises, more and more businesses are interested in utilizing big data [16].

2.1.1 Big Volume

Data sizes are exponentially increasing Stonebraker considers the Big Volume property as important and challenging for two types of analytics: "small analytics" and "big analytics". Small analytics include smaller operations such as running SQL quries (count, sum, max, min, and avg), while big analytics are more complex operations that can be very expensive on very large datasets, such as clustering, regressions, and machine learning.

2.2.2 Big Variety

Structured data comes from various forms like text, audio, video, graphics semi structured data like XML files to very large library files objects. The primary challenge is to integrate all these types of data, and manage them in an efficient way, because integration will provide the speed up in retrieval process of data.

2.2.3 Big Velocity

For many organizations, the most challenging aspect of big data is not exclusively the large volume. it is rather, How fast to process data to meet demands.

2.2.4 Big Veracity

Veracity shows the accuracy and correctness of the big data because information retrieval fast as well as the accurate. Behind any information management practice lies the core doctrines of Data Quality, Data Governance, and Metadata Management, along with considerations for Privacy and Legal.

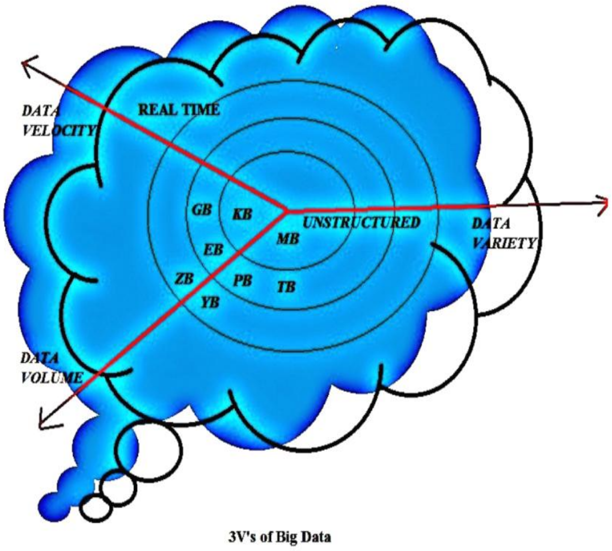


Fig 1 3Vs Big Data Model

2.2.5 Big Value

The ability to understand and manage these sources, and then integrate them into the larger Business Intelligence ecosystem can provide previously unknown insights from data and this understanding leads to Big Data – Value**.**

Google File System is a distributed file system. It is developed by Google to efficient, reliable access to data using large cluster .The basic millstone to development of hadoop is GFS (Google file system).Google file system consists the three working unites: GFS client, GFS master and GFS chunk server. Files are divided into chunks of 64 MB GFS is designed and optimized to run on data centers to provide extremely high data throughputs, low latency and survive individual server failures. Inspired by GFS, the open source Hadoop Distributed File System (HDFS).

2.2 Apache HADOOP

The distributed and parallel computing has a fundamental role in the data processing and information retrieval from large data sets. Software techniques development and hardware compatibility play an important role such as commodity hardware is very decisive to maintain the degree of cluster. The hardware cope up with having a better performance helps to customize the all over estimation of the cluster setup with respect to the cost. Hadoop is a framework that has been designed to take the aforementioned customization advantages i.e. storage, processing, and handling of the faults in cluster. Apache Hadoop is emerging as the obvious choice for managing and big unstructured and structured data. Hadoop helps to improve data search quality and user experience. Hadoop framework came in to the scenario in 2004, which provides an easy and reliable implementation of distributed computing [9]. In this paper the taxonomical hierarchy of the Apache Hadoop organization has been followed. Hadoop is a data storage and data processing framework for large file systems. Hadoop consist of two different working units first the HDFS (Hadoop distributed file system) and second is the programming paradigm Map-Reduce. The HDFS works like as master-slave architecture [7][13][14].

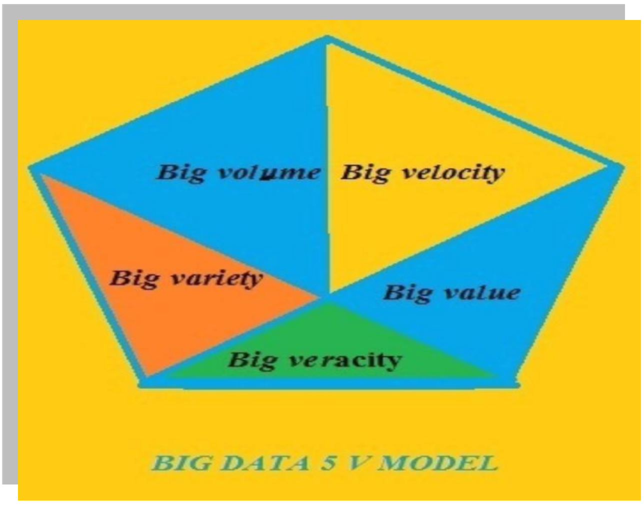


Fig 2 5V Big Data Model

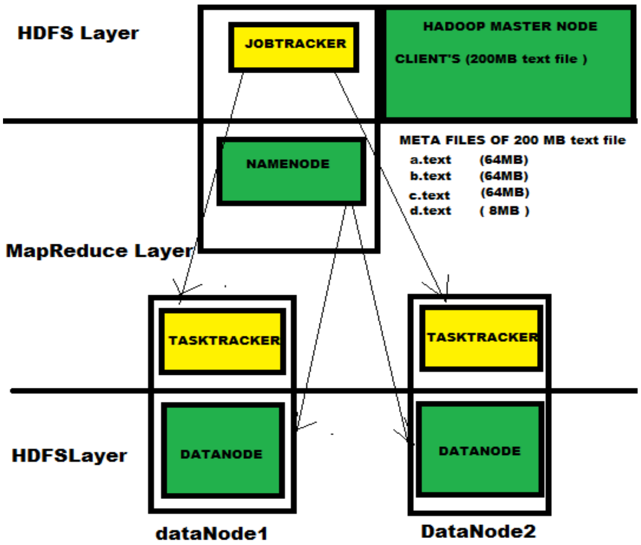


Fig3. HDFS Layer Architecture

Notably, the 'NameNode', 'secondary Node' and 'Job tracker' are three master Nodes. The 'DataNode' and 'Task Tracker' is slave nodes of NameNode and for job tracker respectively [14]. Map-Reduce is an efficient programming model that allow data processing string as alive. Map reduce consists scheduling, data flow and resource allocation at level of storage [1]. In case of BigData, the storage, replication then random access from the database is crucial. Nonetheless the open source frameworks help to Add-on contribution to grown up the power of the software products which directly assist to the researchers. As an open source ecosystem the Apache Hadoop is very important in computing power enhancements [14]. Hadoop can be categorized into four folds first the Map- Reduce which consists scheduling, resource allocation and flow of data, Second the data storage & manipulation, Third the open source contributions, Lastly the data security and energy management [1].

2.3 Map Reduce

Map reduce is a programming paradigm which is associated with BigData owing to its simple and fast processing features. It allows Hadoop to efficiently process the large datasets. Map reduce programming has not comprised new programming models, instead of that it gains attention due to its potential efficacy to deal with large datasets. Map-reduce is basically designed to work Hadoop Distributed File System (HDFS). It works with the master-slave architecture of the HFDS. Apache Hadoop automatically optimizes the execution of Map Reduce task. Because the location of the meta files manage by the NameNode then JobTracker request to assist the location of the meta data in DataNode from the cluster. Map reduce has an ability to process a program in couple of minutes. In Map reduce programming it fundamentally deals with the task scheduling of the jobs. In Hadoop systems the prime objective of the task scheduling or locality aware algorithms is to reduce the overhead of the data during the map and reduce steps [4]. Clearly, Hadoop works to reduce the collision during the data transmission in its framework. The data locality into the homogeneous and dedicated environment can be done in easy way, because that data node has the same configured systems in cluster. Whereas in Heterogeneous environment it may consists different configured system that creates difficulty to achieve good data locality [4]. Map-Reduce have a scheduling as well as the data flow activities [1]. In order to achieve a better data allocation task of the job as well as the load balancing, enormous scheduling algorithms have been proposed. The Task scheduling is one the important technologies of Hadoop framework. The allocations of task and resource utilization are directly related to the overall performance of Hadoop.

2.4 Scheduling Approach

The task scheduling algorithms also affects resource utilization of the system. The allocation of task is performed by job tracker; it creates an instant from name node for taking location of the meta data into the cluster. Job tracker is responsible to assign task of the job. Whereas the assignment of the task of the job is performed by task scheduling Algorithms. It is noteworthy that the default approach of scheduling is First in First out (FIFO) [4]. FIFO algorithm facilitates with the simplicity of the Map-Reduce tasks. FIFO provides less work load on the server due to the order of the task presented in a queue form. The prime disadvantage of FIFO is to deal with different needs by different operations for the job execution. FIFO scheduling problems have been resolved by Facebook and yahoo by putting efforts to design new algorithm to achieve cluster capacity [1]. It is worth to mention about the fair scheduling and capacity scheduling. The fair scheduling makes work group from the job pool. If resources are not sufficient to share, it will assign the extra resources to each job evently. The fair scheduling improves degree of resource utilization of the system. Whereas, the capacity scheduling put the jobs into the multiple queues. The formation of multiple queues depends on the conditions and capacity of the system for each queue. Based on the case specific requirement to achieve quick resource allocation the capacity scheduling uses the redundant resources for that specified queue it will improves the low utilization rate of resources. Now the client will ensure the queue sets and selection of the groups to queue for that job. The Capacity Scheduler is not responsible for automatic settlement of the queues for job execution. By taking advantages from the Weighted Round Robin (WRR) algorithm that has small overhead and easy to be implementing, Jilan Chen et al (2013) proposed a novel approach named Improved Weighted Round Robin (IWRR) scheduling algorithm. The weight is not fixed into the IWRR. Weight update roles help to assign weight according to the small and large task from the job. The basic idea will remain constant during allocation of resources that small weight job will get resource assignment just after large weight job. Weight update roles are able to reduce workload. The Longest Approximate Time to End (LATE) Algorithm is basically for the heterogeneous environment of computing [1]. LATE basically follows priority of the task, selection of fast working node. The LATE used to increase the robustness of node. Some limitations are associated with the LATE to address these limitation the researcher proposed SAMR approach same as to heterogeneous computing environment. fast node which is execute task fast and slow node that execute task in long time ,it’s all are dependence on the memory, resources available to each node to finish the task. The difference generated due the heterogeneous environment. 25% fast working of the SAMR scheduler in compare to Hadoop scheduler.[7]

A different approach about the heterogeneity and homogeneity of the cluster by using information about the system, classification of the jobs and available resource is presented by Seyed Reza Pakize (2014), who presented a comprehensive view on Hadoop scheduling algorithms. The regular analyses on the task scheduling algorithms have the improvement factor in terms of priority and execution time of the task. The data flow during the scheduling is a challenge for researchers and developers. Data flow is a process of the data that employed by Map-reduce framework. Map-reduce consist of two functions ‘map’ and ‘reduce’. It is important to note that these functions are primitive in nature. The Map-reduce divides job into ‘map’ and ‘reduce’ task. Initially input data loaded into HDFS are partitioned into 64mb or 128mb memory blocks. Simultaneously the replications also exist within the assignment of the data block into the cluster. Record reader can read the file that file is form of key–value pairs .Record reader is predefined interface in Hadoop framework. The number of Recorded reader will assigned same as number of mappers.How much mappers will assign that will decide according to the number of Input Splits of the client’s file. The number reducers will be equal to the number of Mapper are present into Hadoop to execute requested file from the client of the cluster .The reducer starts as soon as there are enough map output to start a reduce task. The one to one shuffling has been finished during the map to reduce phases then sorting phase will be executed and the last result will be stored in HDFS.

Lizhe Wang et al (2012) presented the design and implementation of G-Hadoop, a Map-reduce framework that aims to perform large scale distributed computing across multiple clusters. G-Hadoop replaces the native distributed file system by the G-farm file system. A wide area network and storage area network (SAN) is the dedicated network for G-Farm and storage respectively. The prime limitation with Hadoop NameNode is that replica generation of the data at NameNode cannot be generated. This limitation has been called as single point failure of the Hadoop. The single point failure can be avoid by the G-Hadoop. The peer-to-peer message routing and information dissemination structure in which the single point failure of the Name node can be avoided has been presented by the authors [Lizhe Wang et al (2012)].

2.5 Storage and Replication

BigData is not only concerned about the processing and sizing of the large datasets, but also pertained to the accessibility of that datasets with high confidentiality in real time. The Apache Hadoop facilitates the data storage in a layered fashion by arranging the data in blocks followed by racks. The HDFS is the core component of Hadoop system which monitors and manages the complete data storage mechanism. It is worthwhile to state that the ‘failure’ and ‘recovery’ are the two fundamental events in huge data storage. Hadoop 0.20 does not support automatic recovery in case of a NameNode failure. This is well know and recognized single point of failure in Hadoop, becuase NameNode haven't replication of the client's file. [Reason for not support automatically] NameNode are more likely to fail due to the mis-configuration, network issues, and bad behavior amongst clients than actual hardware mistakes. The secondary node or the checkpoint Node also has copy of the NameNode metadata[14]. The HDFS architecture consists master node (NameNode) ,slave node (DataNode). The 'NameNode' is accountable for splitting and managing of the data into Input splits. In such a way the 'NameNode' helps to locate the storage positions of the meta files in HDFS. NameNode spilt client's file into Input splits by 64MB/128MB memory block size. NameNode by default generates three replica of each meta file and locates at DataNode. The NameNode always try to manage storage of replicated Meta files in nearest DataNodes into the cluster [7]. DataNode generates alive report or Heart beat within every three seconds for NameNode. The NameNode identifies the failure in DataNode through the heart beat. Map-reduce program is executed by 'Job tracker' and 'Task tracker. Job tracker assigns the task of the job to the task tracker [14][7]. A significant contribution has been made by several researchers on replication and storage issue. The Yuan Dong [9] proposed a data placement strategy based on the k-means clustering algorithm. The main motivations behind it for load balancing to improve the work perform into the Hadoop framework and increasing the efficiency of the data accessing. HDFS has the load balancing procedure which can balance the storage load on each machine. When the clients uploaded a file, analyzing request of file will uploading and place the relevant documents to the same storage node. In the resettlement process, there should be ensure load balancing [9] simultaneously network admin should try to reduce network cost during program execution and data migration at the stage of the follow-up replica's adjustment. Data placement affect all over reliability of Hadoop framework. Pre-placement process always guides the data processing in fast manner.

Lu Lu et al [2014] proposed A decoupled Map Reduce computing-storage system for cloud computing environment and present the load aware data placement strategy also implementation design to data placement by the virtual machine. In this paper author proposed the comparison between physical storage and virtual storage in reference of the load balancing and high availability of the data. The "Morpho" is proposed as a modified version of the Hadoop [5].The morpho consists process of task execution through virtual machine same as Hadoop and storage keeping through the physical storage .When Map Reduce computation will performing the map tasks can get meta data file directly from physical machines storage without any extra data transfers .Morpho achieves high performance due to virtual placement. The virtual placement increases the resistant factor of the resource utilization. The Morpho facilitates decoupling of computation and storage, VM placement and data perception mechanism after decoupling. The decoupling of the Morpho creates more possibilities to do work with virtual machines in the context of BigData. In context of hardware failure and recovery it takes times to remake. Additionally author proposed the stress about sophisticated locality aware algorithms for data into the framework to achieving high performance computing.

Hadoop may work using a replacement from the file system but when changes have been done changing into the HDFS. Lizhe Wang et al [2013] proposed the new file system . G-Hadoop works like a Hadoop but different storage system that is Gfram . Gfram use MPI application so map reduce task can work with MPI application. To migrate HDFS to GFram is not a silly think it is so hard because for that need to generate extra replicas of the data. To implementation of Gfram into Hadoop plug-in is used for it. To make G-Hadoop fully functional there will be requirement to manage network to extrude the compressive results for analysis the performance of two same frameworks.

In Apache Hadoop HDFS has been designed to support sequential queries but if same data will exercise with the queries it's just about the unnecessary data transfer. HDFS is not suitable for the small storage. In Hadoop a single change in file. Hadoop will again do exercise on replica generation and placement of data into the DataNode.

3. IMPLEMENTATION & RESULT

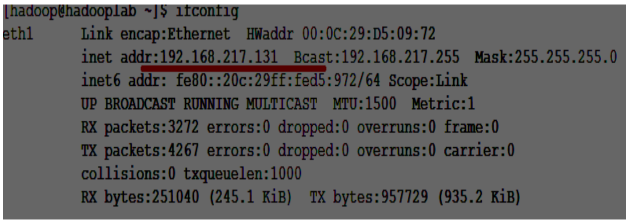
3.1 Lab details

3.1.1 VM is running on CentOS 6.2

Use the user name and password

3.1.2 Get the VM's IP address which we will use

Run ifconfig and make note of you ip address



```
[hadoop@hadooplab ~]$ ifconfig
eth1      Link encap:Ethernet  HWaddr 00:0C:29:D5:09:72
          inet addr:192.168.217.131  Bcast:192.168.217.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fed5:972/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:3272 errors:0 dropped:0 overruns:0 frame:0
          TX packets:4267 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:251040 (245.1 KiB)  TX bytes:957729 (935.2 KiB)
```

3.1.3 Configure your VM's hosts file

sudo vi /etc/hosts

Change the following ip address to the one obtained in the previous step

192.168.217.131 hadooplab.bigdataleap.com hadooplab

Save and exit the file. And verify if the settings are working file by running the following command.

ping hadooplab.bigdataleap.com

If you are getting reply from the VM, then it is configured properly.

3.1.4 Configure your laptop/desktop's windows' hosts file

Run notepad in administrator mode. Note: right click on notepad icon and run it as administrator Then do File -> Open from notepad and go to the directory C:\Windows\System32\drivers\etc And open the hosts file in the notepad Add the following line as the last line in the file Note: MAC users should update their /etc/hosts file to add VM's hostname and IP address
192.168.217.131 hadooplab.bigdataleap.com
(Note: change IP to your VM's IP Address)
Save and exit the file. And verify if the settings are working file by running the following command in the windows command shell.

ping hadooplab.bigdataleap.com

If you are getting reply from the VM, then it is configured properly.

3.1.5 Know the directories available in the VM for hands on exercises

Go to lab directory is available in /home/hadoop and list the directories available inside it.

cd /home/hadoop/lab

/home/hadoop/lab contains the following directories and will be used for the following purposes.

export PATH=$PATH:$HIVE_HOME/bin

export PIG_INSTALL=/home/hadoop/lab/software/pig-0.12.0

export OOZIE_HOME=/home/hadoop/lab/software/oozie-4.0.0

export PATH=$PATH:$PIG_INSTALL/bin:$OOZIE_HOME/bin

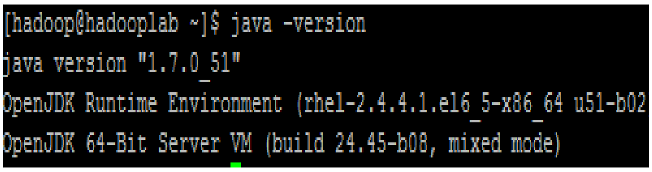
export PATH

- Save and exit .bash_profile
- run following command

. .bash_profile

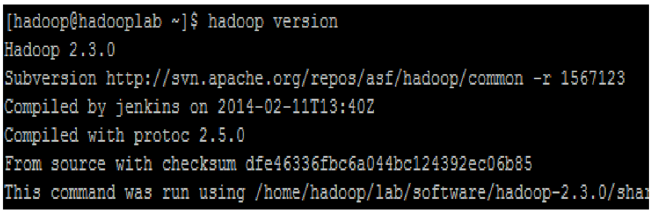
- Verify whether variable are defined or not by typing ***export*** at command prompt
- Check the following versions

java –version



```
[hadoop@hadooplab ~]$ java -version
java version "1.7.0_51"
OpenJDK Runtime Environment (rhel-2.4.4.1.el6_5-x86_64 u51-b02
OpenJDK 64-Bit Server VM (build 24.45-b08, mixed mode)
```

hadoop version



```
[hadoop@hadooplab ~]$ hadoop version
Hadoop 2.3.0
Subversion http://svn.apache.org/repos/asf/hadoop/common -r 1567123
Compiled by jenkins on 2014-02-11T13:40Z
Compiled with protoc 2.5.0
From source with checksum dfe46336fbc6a044bc124392ec06b85
This command was run using /home/hadoop/lab/software/hadoop-2.3.0/shar
```

3.1.6 Configuring pseudo-distributed mode

Go to conf directory of hadoop installation folder

cd /home/hadoop/lab/software/hadoop-2.3.0/etc/hadoop

The following files are available in the reference folder of the lab distribution files on your windows or mac machine.

Modify core-site.xml

```
 <configuration>
    <property>
        <name>fs.defaultFS</name>
        <value>hdfs://localhost:9000</value>
    </property>
</configuration>
```

Modify hdfs-site.xml

```
<configuration>
    <property>
        <name>dfs.replication</name>
        <value>1</value>
    </property>
</configuration>
```

Modify yarn-site.xml

```
<configuration>
    <property>
        <name>yarn.nodemanager.aux-services</name>
        <value>mapreduce_shuffle</value>
    </property>
</configuration>
```

Modify mapred-site.xml

```
<configuration>
    <property>
        <name>mapreduce.framework.name</name>
        <value>yarn</value>
    </property>
</configuration>
```

3.1.7 Copy the 64 bit libraries

Copy the 64 bit native libraries

Go to the following directory

cd /home/hadoop/lab/downloads/lib64bit/

cp libhadoop.so.1.0.0 $HADOOP_INSTALL/lib/native/

cp libhdfs.so.0.0.0 $HADOOP_INSTALL/lib/native/

3.1.8 Configure JAVA_HOME

Go to ***/home/hadoop/lab/software/hadoop-2.3.0/etc/hadoop*** directory

export JAVA_HOME=/usr/lib/jvm/jre-1.7.0-openjdk.x86_64

Enter the above line at the beginning of all the following files:

□ hadoop-env.sh

□ mapred-env.sh

□ yarn-env.sh

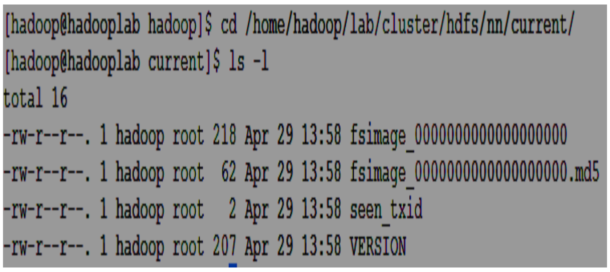
i.Format the namenode

Enter the following command at prompt

hdfs namenode –format

- Go to /home/hadoop/lab/cluster/hdfs/nn/current directory and verify whether all files have been created. o fsimage (file system image) and it's md5 file (fingerprint)

o VERSION (contains unique cluster, layout version and other details…)



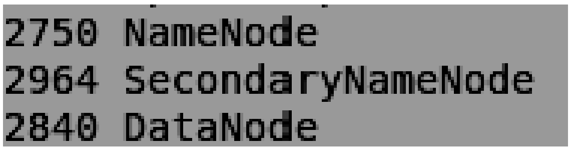
```
[hadoop@hadooplab hadoop]$ cd /home/hadoop/lab/cluster/hdfs/nn/current/
[hadoop@hadooplab current]$ ls -l
total 16
-rw-r--r--. 1 hadoop root 218 Apr 29 13:58 fsimage_0000000000000000000
-rw-r--r--. 1 hadoop root  62 Apr 29 13:58 fsimage_0000000000000000000.md5
-rw-r--r--. 1 hadoop root   2 Apr 29 13:58 seen_txid
-rw-r--r--. 1 hadoop root 207 Apr 29 13:58 VERSION
```

3.1.9 Start HDFS and YARN services

Go to */home/hadoop/lab/software/hadoop-2.3.0/sbin* directory and type the following command

./start-dfs.sh

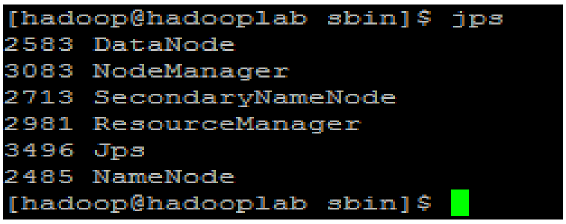
Note: verify if all the following three processes have started by typing *jps* command



```
2750 NameNode
2964 SecondaryNameNode
2840 DataNode
```

And then type the following command

./start-yarn.sh
Run jps and verify if all the following processes are running



```
[hadoop@hadooplab sbin]$ jps
2583 DataNode
3083 NodeManager
2713 SecondaryNameNode
2981 ResourceManager
3496 Jps
2485 NameNode
[hadoop@hadooplab sbin]$
```

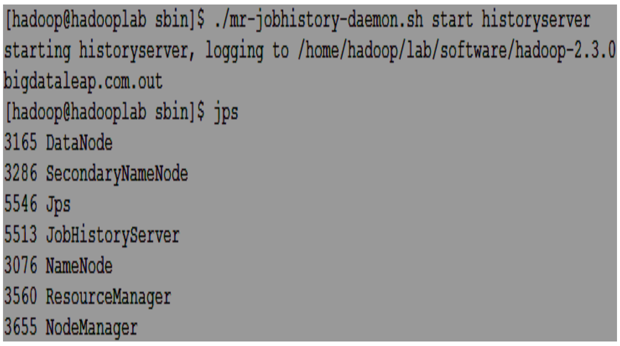
If all five processes are running, then hadoop is up and running

□□ Run the history server, which will provide information about completed jobs

Go to ***/home/hadoop/lab/software/hadoop-2.3.0/sbin*** directory and type the following command

./mr-jobhistory-daemon.sh start historyserver

And run jps to confirm if the history server is started or not.



```
[hadoop@hadooplab sbin]$ ./mr-jobhistory-daemon.sh start historyserver
starting historyserver, logging to /home/hadoop/lab/software/hadoop-2.3.0
bigdataleap.com.out
[hadoop@hadooplab sbin]$ jps
3165 DataNode
3286 SecondaryNameNode
5546 Jps
5513 JobHistoryServer
3076 NameNode
3560 ResourceManager
3655 NodeManager
```

3.2 HDFS Setup

3.2.1 Verify what all files are available in the hdfs file system

hadoop fs –ls /

3.2.2 Copy files into HDFS

Create the following HDFS directories

hadoop fs –mkdir /lab

hadoop fs –mkdir /lab/mr

hadoop fs –mkdir /lab/hive

hadoop fs –mkdir /lab/pig

hadoop fs –mkdir /lab/sqoop

Check directories in HDFS

hadoop fs -ls /lab

Copy files from linux directory */home/hadoop/lab/data* to HDFS directory /lab/mr

hadoop fs -copyFromLocal */home/hadoop/lab/data/*txns /lab/mr

hadoop fs -copyFromLocal */home/hadoop/lab/data/*custs /lab/mr

hadoop fs -copyFromLocal /home/hadoop/lab/data/words /lab/mr/

hadoop fs –ls /lab/mr/

- Go to the following directory on the linux machine

cd /home/hadoop/lab/cluster/hdfs/dn/current/BP-*/current/finalized

and verify the blocks have been created.

HDFS Filesystem statistics

hdfs dfsadmin –report

Gives you detailed report of the hdfs system including

□ total capacity allocated, used, available

□ no offiles, block

Checking health of files in HDFS

Gives you detailed report of hdfs files (All files or a specific files)

hdfs fsck /

hdfs fsck /lab/mr/txns -files -blocks -locations

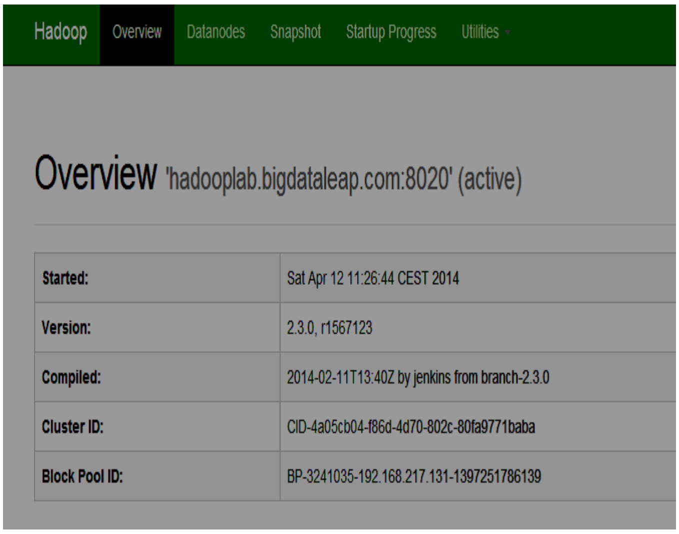
Gives you detailed report of the file that is specified

□ Total numberof blocks and their size

□ Under replicated or missing blocks, if any

3.3 HDFS Web UI

Open your browser & enter the following URL [31]



Overview 'hadooplab.bigdataleap.com:8020' (active)

Started:	Sat Apr 12 11:26:44 CEST 2014
Version:	2.3.0, r1567123
Compiled:	2014-02-11T13:40Z by jenkins from branch-2.3.0
Cluster ID:	CID-4a05cb04-f86d-4d70-802c-80fa9771baba
Block Pool ID:	BP-3241035-192.168.217.131-1397251786139

4. CONCLUSION & FUTURE WORK

This concludes that Apache Hadoop has overcome its initial phase with a lot of unstable issues. The open source framework is growing solid and functional. The performance issues addressing by the Hadoop Distributed File System. The Map reduce playing smart role with Hadoop as computing paradigm Map reduce is not new but with the Hadoop is shine its attention to award Hadoop. The understandings and growth of Hadoop and map reduce has consolidated because users are able to understand why Hadoop map reduce are designed. Task scheduling is considered a crucial in context of the Hadoop performance. Some papers present solutions covering area to cloud resource allocation .the storage is another intersection point to study HDFS. The intersection point is achieved by study the Morpho because the Morpho just proposed difference and new always to data placement and data processing. In order to improve perform into the resource allocation, replica generation several approaches make modifications in how data decoupled and how replication has been completed .some area although not directly related to the framework. Various contributions are including themes such as energy efficient algorithms, cooling of the Hadoop energy management. Energy management is catching the speech in research topic as the numbers of data centers are rapidly increased. With new technologies of the storage becoming cheaper, the intersection among storage cloud computing green Hadoop will probably deserve upcoming developments exploration of the new challenges.

5. ACKNOWLEDGMENT

This research paper is made possible through the help and support from everyone Especially, please allow us to dedicate our acknowledgment of gratitude toward the Department of Information Technology and Computer Science And Engineering, R.C.E.W Bankrota, jaipur for providing necessary support and lab facility.

6. REFERENCES

[1] IvaniltonPolato a,b,n, ReginaldoRé b, AlfredoGoldman a, FabioKon a A comprehensive rview of Hadoop research—A systematic 46 (2014)1–25

[2] Seyed Reza Pakize, A Comprehensive View of Hadoop MapReduce Scheduling Algorithms VOL. 2, NO. 9, SEPTEMBER 2014, 308–3179 (2014)

[3] Gothai E, Balasubramanie P, A Novel Approach For Partitioning In Hadoop Using Round Robin Technique, 20th May 2014. Vol. 63 No.2 © 2005 - 2014

[4] Kamble Ashwini ,Kanawade Bhavan, A Brief on MapReduce Performance Volume 1 Issue 1 (April 2014)

[5] Lu Lu, Xuanhua Shi ∗, Hai Jin, Qiuyue Wang, Daxing Yuan, Song Wu, Morpho: A decoupled MapReduce framework for elastic cloud computing , 36 (2014) 80–90

[6] Kirandeep Kaur1, Khushdeep Kaur2, An Improved Longest Approximate Time to End Algorithm using Dynamic Cloud Sim 2319-7064 .2013

[7] Lizhe Wanga,b,∗, Jie Taoc, Rajiv Ranjan d, Holger Martenc, Achim Streit c, Jingying Chene, Dan Chena,∗∗, G-Hadoop: MapReduce across distributed data centers for data-intensive computing 29 (2013) 739–750

[8] Vidyasagar S.D, A Study on “Role of Hadoop in Information Technology era” Volume : 2 | Issue : 2 | Feb 2013 • ISSN No 2277 - 8160

[9] Shaochun Wu ,Xiang Shuai, Liang Chen, Ling Ye, Bowen Yuan, A replica pre-placement strategy based on correlation analysis in cloud environment (CCIS 2013)

[10] Jilan Chen, Dan Wang and Wenbing Zhao, A Task Scheduling Algorithm for Hadoop Platform, VOL. 8, NO. 4, APRIL 2013

[11] Shreyas Kudale1, Advait Kulkarni2, Asst. Prof. Leena A. Deshpande3, Predictive Analysis Using Hadoop: A Survey, Vol. 1, Issue 8, October 2013

[12] Hortonworks, Community Driven Apache Hadoop Apache Hadoop Basics May 2013 ©

[13] S. Chandra Mouliswaran And Shyam Sathyan*, Study On Replica Management And High Availability In Hadoop Distributed File System (Hdfs), Vol 2 / Issue 2 / 2012 / 65-70

[14] Http://Www.Apache.Org

[15] Sarannia,N.Padmapriya, Survey On Big Data Processing In Geo Distributed Data Centers Vol 4, Issue 11, November 2014

[16] J. Gerard Wolff, (Member, Ieee) Cognitionresearch.Org, Menai Bridge, U.K. (Jgw@Cognitionresearch.Org) Big Data And The Sp Theory Of Intelligence ,Received October 20, 2013, Accepted March 27, 2014, Date Of Publication April 2, 2014, Date Of Current Version April 15, 2014. Digital Object Identifier 10.1109/Access.2014.2315297

[17] Madhury Mohandas & Dhanya P M ,Department Of Computer Science & Engineering, Rajagiri School Of Engineering & Technology, Cochin, Kerala, India, “Algorithm For Efficient Data Placement In Blobseer Architecture” International Journal Of Computer Science Engineeringand Information Technology Research (Ijcseitr) Issn 2249-683,Vol. 3, Issue 3, Aug 2013, 193-200

[18] Tao Gu, Chuang Zuo, Qun Liao, Yulu Yang and Tao Li “Improving MapReduce Performance by Data Prefetching in Heterogeneous or Shared Environments”, International Journal of Grid and Distributed Computing Vol.6, No.5 (2013), pp.71-82

[19] Md. Rezaul Karim1, Azam Hossain1, Md. Mamunur Rashid1, Byeong-Soo Jeong1, and Ho-Jin Choi2,” An Efficient Market Basket Analysis Technique with Improved MapReduce Framework on Hadoop: An E commerce Perspective

[20] Gothai E, 2balasubramanie P,” A Novel Approach For Partitioning In Hadoop Using Round Robin Technique”, Journal Of Theoretical And Applied Information Technology 20th May 2014. Vol. 63 No.2

[21] S. Chandra Mouliswaran And Shyam Sathyan, “Study On Replica Management And High Availability In Hadoop Distributed File System (Hdfs)”S. Chandra Mouliswaran And Shyam Sathyan. Et Al. / Journal Of Science / Vol 2 / Issue 2 / 2012 / 65-70

[22] Chia-WeiLeea, Kuang-YuHsieha, Sun-YuanHsieha,b,∗, Hung-ChangHsiaoa, “A Dynamic Data Placement Strategy for Hadoop in Heterogeneous Environments”, Big DataResearch1(2014)14–22

[23] Balaji Palanisamy, Aameek Singh, Ling Liu, Bhushan Jain,” Purlieus: Locality-aware Resource Allocation for MapReduce in a Cloud”

[24] Ivan Baev† Rajmohan Rajaraman‡ Chaitanya Swamy§,”Approximation Algorithms for Data Placement Problems

[25] R.Jemina Priyadarsini1, Dr.L.Arockiam2,” An Extensive Analysis On Task Scheduling Algorithms In Cloud Environments” (Ijetcas)”

[26] Medhat Tawfeek, Ashraf El-Sisi, Arabi Keshk and Fawzy Torkey,” Cloud Task Scheduling Based on Ant Colony Optimization” The International Arab Journal of Information Technology, Vol. 12, No. 2, March 2015.

[27] Nitesh MaheshwariRadheshyam Nanduri, Vasudeva Varma.”Dynamic Energy Ecient Data Placement and ClusterReconfigurationAlgorithmforMapReduceFramework”

[28] Phokham Nonava October 2014,” HDFS Blocks Placement Strategy

[29] Ivanilton Polatoa,b,⇜, Reginaldo R´eb, Alfredo Goldmana, Fabio Kona,” A Comprehensive View of Hadoop Research - A Systematic Literature Review”journal of network and computer applications volume ,46 november 2014, page no 1-25

[30] George Porter UC San Diego La Jolla,” Decoupling Storage and Computation in Hadoop with SuperDataNodes”.