



Model to Analyse Network and for Proactive node Management Implemented in Hadoop environment

Anjana Pandey
Deptt of IT
UIT RGPV Bhopal

Mahesh Pawar
Deptt of IT
UIT RGPV Bhopal

Abstract: The paper involves designing of a model that does two tasks. First is to perform analyses of the network and second is to manage nodes in the network in a very proactive manner. Analyses of the existing network are done by discovering network topology using a network monitoring tool called OpenNMS. This tool helped us discover whether a node is working or not by regularly collecting signals or heartbeat from the nodes. The S.M.A.R.T. tool and the topology discovered from this tool can be used to perform effective network administration by managing network traffic. It can also help us perform load balancing by guiding necessary decisions that are to be made about node management, as and when required. The other aspect of the model that deals with proactive node management accomplishes its objective by recognizing hardware failures. This is done by analysing the data coming from all nodes regularly. The data, that comes in the form of log files, involves many parameters like temperature, seek error rate, raw read error rate etc. The model inspect for any failures in a Node by comparing the values coming from node with the threshold values of failure. This helps the model detect the network, CPU and hard disk failures. In this way the model informs the network administrator about the problems that may occur well in advance so that he can take appropriate actions.

Keywords: Hadoop, Apache, OpenNMS, ZenMap, S.M.A.R.T tool

I. INTRODUCTION

The Apache Hadoop is an open-source software framework for storage and large-scale processing of data-sets on clusters of commodity hardware. It is used to develop open-source software for reliable, scalable, distributed computing[1]. It is a framework that allows for the distributed processing of large data sets across clusters of computers built on commodity hardware using a simple programming model. It provides the HDFS that stores data on the compute nodes, providing very high aggregate bandwidth across the cluster [2]. The Hadoop network can be deployed on a variety of scales. In the context of high-volume data processing, the limiting factor is the rate at which one can transfer data between the nodes or the bandwidth which is also dependent on the physical topology of the network [3]. In order to configure Hadoop clusters to meet varying needs in terms of size, processing power, reliability, availability etc. it is important to keep track of the current topology of the Hadoop network and track the status of the network services in Hadoop [4]. This is one of the modules in this paper. The other module in this paper is to keep track of the hardware failures in the nodes in the Hadoop cluster and alert that node about the same. Data pertaining to various hardware components is collected from the nodes in the Hadoop cluster. This data is then analysed to alert the node about the status of its hardware components based upon the threshold values. The objective of the paper is to build a Hadoop compatible failure monitoring system for discovering network topology and diagnosing the hardware that are not functioning correctly. One are working on a Hadoop framework that will enable nodes to identify failures on their hardware using Hadoop log, System log and various OS hardware diagnosing utilities. Such events include block device failures, file system-level failures, memory error corrections, PCI device failures, network errors etc. The Hadoop log files provide hints about failed hardware.

The rest of the paper is organised as follows: related work explain in section II. Hadoop architecture and tools are

described in section III. Proposed methodology is described in section IV. An experimental results are presented in section V and concluding remarks are given in section VI.

II. RELATED WORK

Apache Hadoop is solution for big data .It deals with complexities of high volume, velocity and variety of data. In research paper "An experimental approach towards big data for analyzing memory utilization on a Hadoop cluster using HDFS and Map Reduce" describes that Hadoop consist of two pieces, first Map Reduce which is the processing part of Hadoop and second is data part of Hadoop called HDFS (stores all the data and files). The researcher analyzed the variables for storage and processing of the data on a Hadoop cluster [5]. Storage and processing using Map reduce algorithm and Pig tool was studied.

In research paper "Design and performance evaluation for Hadoop clusters on virtualized environment" says that Amazon already released the Hadoop on Xen-Virtualization as Elastic Map Reduce, but VM downgrade its performance, so it may possible to build a performance model and examine how the performance is affected while changing VM configuration. By using Hadoop on Amazon EC2, one can study of resources consumption information for Map Reduce application [6]. So, that one can find the efficient way by using Map Reduce Algorithm. In research paper "The performance evaluations for Hadoop in diverse system architectures" that Hadoop framework works on 32bit Linux operating system which may less utilize the resource, so it may possible to create 64bit operating system[14].

The research paper "Astro: A predictive model for anomaly detection and feedback-based scheduling on Hadoop" describe that in Hadoop, Astro which consist of predictive model detects anomalies and give feedback based scheduling [15]. In research paper "Dealing with Small Files Problem in Hadoop Distributed File System" describes that Hadoop distributed file system (HDFS) is meant for storing

large files but when large number of small files needs to be stored, HDFS has to face few problems as all the files in HDFS are managed by a single server. So, there are various methods have been proposed to deal with small files problem in HDFS. This paper gives comparative analysis of methods which deals with small files problem in HDFS [16].

In research paper "ProMARTES: Accurate network and computation delay prediction for systems" perform analysis on Cycle-accurate performance analysis open-source tool for CBRTDS Fusion of network and processing analysis methods. Support both analytical and simulation analysis techniques. Application of the proposed framework to an autonomously navigating robot system. Proposal of an effective combination of scheduling and simulation analysis techniques[17].

The research paper "Topological characteristics of simple and complex normal fault networks" in this A range of topology analysis is proposed to characterize normal fault networks. The 12 fault networks studied can be divided into 3 main groups. The networks can be further subdivided into 7 types. A broad link between topology characteristics and cause of network complexity[18]. Therefore, Hadoop is set of open source framework that work together to provide common set of services such as storage and processing of large data.

III. HADOOP ARCHITECTURE

Hadoop is an open-source software framework for storage and large-scale processing of data-sets on clusters of commodity hardware [7]. There are mainly five building blocks inside this runtime environment (from bottom to top) as shown in Figure1:

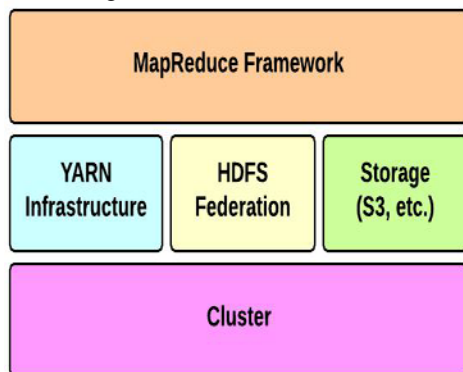


Figure1 Block diagram of Hadoop Architecture

The cluster is the set of host machines (nodes). Nodes may be partitioned in racks. This is the hardware part of the infrastructure. The YARN Infrastructure (Yet another Resource Negotiator) is the framework responsible for providing the computational resources (e.g., CPUs, memory, etc.) needed for application executions. HDFS i.e. Hadoop distributed file system is used for big data processing and analyses [8]. Two other important elements are:

The various tools used for discovering network topology and analyzing the log files are which includes: OpenNMS, ZenMap, S.M.A.R.T tool

A. Open NMS

Open NMS is a free and open-source enterprise grade network monitoring and network management platform. It

is developed and supported by a community of user and developers as well as by the Open NMS Group [9].

B. Zen Map

Zenmap is the official Nmap security scanner GUI. It is a multi-platform open source application". The scans which are used frequently can be saved as profiles so that it would become easy when one wish to run them repeatedly.

C. S.M.A.R.T. TOOL

S.M.A.R.T. (Self-Monitoring, Analysis and Reporting Technology) is a monitoring system for computer hard disk drives. Depending upon the statistics provided by S.M.A.R.T, the user is advised to replace the hard drive in order to avoid loss of data. It is also used by the hard drive manufacturers in order to discover various failures on the hard drives [10].

IV. PROPOSED METHODOLOGY

The model provides a Hadoop framework for running applications on large clusters built on commodity hardware. Hadoop framework transparently provides applications, both reliability and data motion. 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.

A. Improvements made

In addition to the previous work done on this topic, our model provides HDFS framework that stores data on the compute nodes, providing very high aggregate bandwidth across the cluster. The model's ability to help the network administrator make intelligent decisions about network load balancing by providing failure notifications in advance is an added advantage of this model. The following section gives an insight into the implementation process followed to develop this paper.

B. Implementation

The current implementation of Hadoop, the only service available for discovering network topology is a topology script which identifies the rack id of the node, if pass the IP address of the node as argument. This model however has scope for hardware failure detection as the Name Node would notice that if the Data Node is not sending heartbeat signals, then after a certain time period (10 minutes by default) it considers the node as dead, at which point it will re-replicate the blocks that were on the failed Data Node using replicas stored on other nodes of the cluster. We propose a system in which one detect the network topology map of the Hadoop cluster using a Linux tool named Open NMS. The IP Addresses of the Data Nodes are dynamically fed into the tool with the help of a script which parses the addresses from the Hadoop logs generated in the Name Node [13]. One also monitor the status of services in the Hadoop Data Nodes like SSH, SMTP, SNMP, etc. This functionality is performed with the help of Open NMS tool. For hardware failure detection, one propose a mechanism in which one can diagnose failures in a more detailed manner like block device failures, file system-level failures, network

errors, hard disk failure etc. The major part of the model however is hadoop configuration that is mentioned below.

C. Prerequisites

Hadoop is supported by GNU/Linux platform and its flavours. Therefore, it is required to install a Linux operating system for setting up Hadoop environment. In case, an OS other than Linux, then to install a Virtual box software in it and have Linux inside the Virtual box [12].

D. Hadoop Operation Modes

In a downloaded Hadoop, The Hadoop cluster can operate in one of the three supported modes:

Local/Standalone Mode

In this environment by default, it is configured in a standalone mode and can be run as a single java process.

Pseudo Distributed Mode

It is a distributed simulation on single machine. Each Hadoop daemon such as hdfs, yarn, MapReduce etc., will run as a separate java process. This mode is useful for development.

Fully Distributed Mode

This mode is fully distributed with minimum two or more machines as a cluster. This is very useful for managing large no of networks.

Before installing Hadoop into the Linux environment, one need to set up Linux using ssh (Secure Shell). Follow the steps given below for setting up the Linux environment.

E. Open NMS

Another important tool used in the model is Open NMS that is described as follows.

OpenNMS is a free and open-source enterprise grade network monitoring and network management platform. It is developed and supported by a community of user and developers as well as by the OpenNMS Group, offering commercial services, training and support. Discovery is the way by which Open NMS finds nodes (devices with a network address). Discovery can be either automatic or manual, depending on the preference. For smaller networks, manual addition of nodes is feasible, but larger networks will benefit significantly from automatic discovery.

Hence, there are three types of discoveries :

- I. Automatic Discovery
- II. Semi-automatic Discovery
- III. Manual Discoveries

F. Discovering Network Topology

Automatic Discovery

Before starting OpenNMS, configure OpenNMS with the relevant address ranges (typically one or more subnets, but the syntax is flexible). Optionally one can configure the local SNMP community strings, with significant flexibility in using different community strings for different address ranges or specific addresses. Once configured, start OpenNMS, and the network will be automatically discovered. The Discovery daemon sends out ICMP pings to

see if something responds on that IP address. If a response is received, a "New Suspect" event is generated and passed into the OpenNMS event system. The Capabilities daemon picks that event up and scans that IP address for available services that OpenNMS knows about. The node and it's services will be added to the OpenNMS database; monitoring and data collection will commence for that node.

Semi-automatic discovery

If full automatic discovery is not for one, OpenNMS will generate New Suspect events for any unknown nodes that send the OpenNMS server an SNMP trap. This means that with OpenNMS running, one can configure the SNMP agent on the host to send traps to the public community on the OpenNMS server, restart the SNMP agent (which typically causes a "Cold Start" trap to be sent), and OpenNMS will discover that node and schedule a capabilities scan.

Manual discovery

For some who desire more control, or who only have small network, individual nodes can be manually added. There are two significant ways to add nodes manually:

Via the web-interface, by entering the IP address, or Using the send-event.pl script to send a New Suspect event the second case for example, custom scripts can be written to obtain appropriate IP addresses from existing systems or databases, and feed these into OpenNMS.

The model used the semi-automatic discovery method to discover network topology and finally, the data coming from log files was fed into hadoop that analysed the data by comparing it with threshold values. The experiment was done on three different machines- DELL, TOSHIBA and HP. The performance of all these machines coming from different vendors was compared to reach a final conclusion. From existing systems or databases, and feed these into OpenNMS.

V. RESULT & ANALYSIS

Table 1 gives some sample vendor specific values of the hard disk models of the nodes involved in the cluster. The attributes which are listed in table 1 are as follows:

• Seek error rate

Rate of seek errors of the magnetic heads. Failures in mechanical positioning system such as failure due to thermal heating of hard disk, damage to a server can result in increase of seek errors. The seek error rate value depends on different vendors.

• Raw read error rate

This value of this attribute depends on read errors and disk surface condition. This indicates the rate of hardware read errors which occur when data is read from the disk surface. Lower value of this attribute is preferable.

• Reallocated Sector Count

Reallocated Sector Count provides the count of reallocated sectors. A sector is marked as "reallocated" when the hard drive finds a read/write/verification error. The process of transferring data to a special reserved area due to reallocation is called remapping and the reallocated sectors are called remaps. It indicates a count of remapped sectors. Higher value of reallocated sectors leads to decrease in the

read/write speed because the head of the drive is forced to seek to the remapped area when a reallocation sector is accessed. Lower value of this attribute is preferable.

• Temperature Celsius

This attribute indicates the current internal temperature. Lower value of this attribute is preferable.

• Reallocated Event Count

This parameter indicates the count of remap operations (data is transferred from a bad sector to a special reserved disk area, which is known as spare area). The total number of attempts to transfer data from reallocated sectors to a spare area is indicated by the raw value. Both successful and unsuccessful attempts are counted. The Vendor specific values of the hard disk models of the nodes involved in the cluster.

TABLE1: Vendor specific values of the Hard disk models of the nodes involved in the cluster.

Attributes	DELL	TOSHIBA	HP
Seek Error rate	65	10	48
Raw read error rate	65	47	47
Relocated sector count	8	120	45
Temperature Celsius	58	60	63
Relocated event count	0	1	0

Change in Hard Disk Temperature

The sample data was collected from the Name Node and the Data Node which had the same manufacturer of the hard disk.

TABLE2: Change in Hard Disk Temperature

Time Instant	Temperature on Name Node	Temperature on Data Node
1	60	52
2	73	52
3	65	61
4	66	68
5	61	61

The following graph (Figure 2) represents the above values. Clearly, as temperature on Data node increases, temperature on name node also increases.

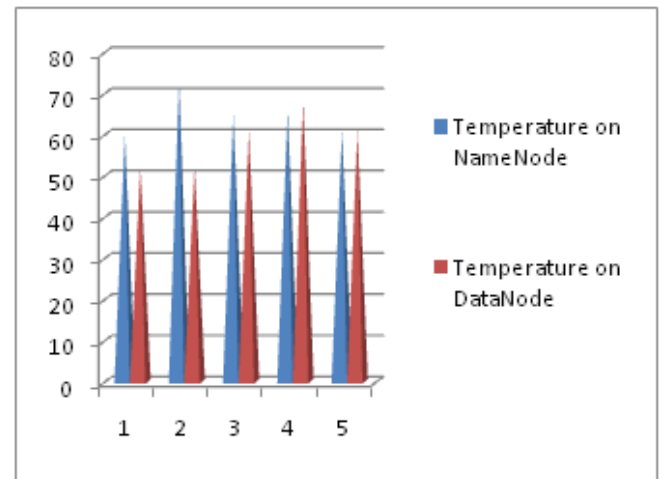


Figure 2: Graph for change in temperature in Name node and Data node

Change in temperature on change in number of nodes

Table 3 shows time taken by various no of nodes to respond. The graph (Figure 3) for this result shows that as the no of nodes increase, time taken to respond also increases.

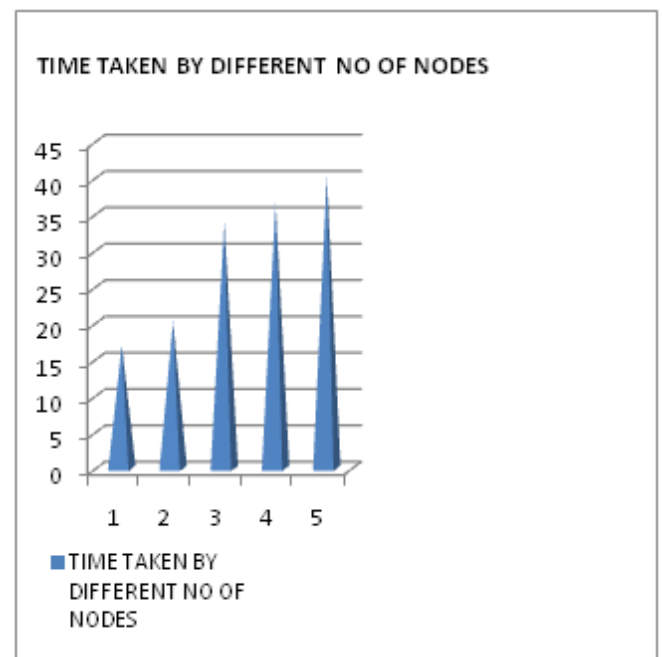


Figure 3: Graph for time taken by different nodes.

The Figure: 4 shows a representation of the all the above values. It clearly indicates that various companies that use different design methodologies for designing their processors give different values of the included parameters. While deciding on one particular system to be best is tough one can still conclude DELL performed better than others on an average.

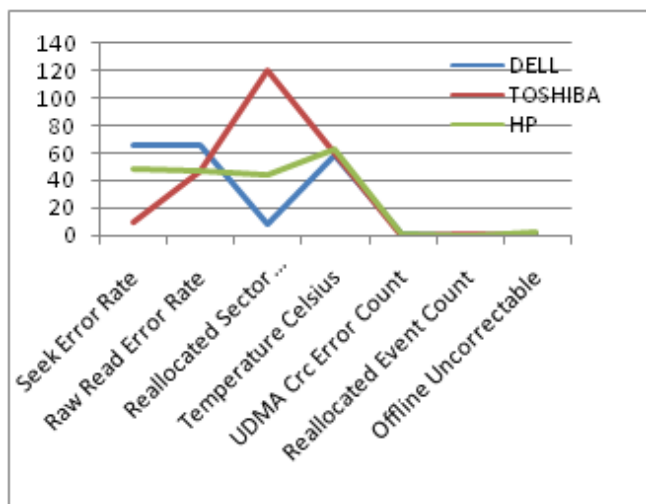


Figure 4: Graph for different attributes.

Table3: Time taken by different nodes

Number of Nodes	Time taken in seconds
0	0
1	17.01
2	20.23
3	33.85
4	36.52
5	40.23

VI. COUNCLUSION

A productive experiment was done to implement a Hadoop consistent framework that helps to determine network topology and does diagnoses of suspected hardware failures in Data Nodes within a Hadoop cluster. Based on the status of the various hardware components of the Data Nodes by using log files, the master node signals the Data Node using the open NMS tool so that it can take right actions. After determining the network topology using open NMS one can generated a topology map that displays only the alive nodes in our Hadoop cluster network. This was achieved by feeding the list of IP addresses of the live nodes which was generated by the scheduler script into the open NMS tool. The log files generated thus were introduced into Hadoop framework that analysed the values of various parameters coming from the data nodes and did proactive prediction

about data node failure by comparing the incoming data with threshold values.

VII. REFERENCES

- [1] AmritPal, SanjayAgrawal "An experimental approach towards big data for analyzing memory utilization on a hadoop cluster using HDFS and MapReduce" Networks & Soft Computing (ICNSC), 2014 First International Conference.
- [2] Masakuni Ishii, Jungkyu Han and Hiroyuki Makino "Design and Performance Evaluation for Hadoop Clusters on Virtualized Environmen" Information Networking (ICOIN), 2013 International Conference.
- [3] Apache Site "<http://hadoop.apache.org/common/docs/current/hdfsdesign.html>.
- [4] T.White, "Hdfs_Reliability" www.cloudera.com/HDFS_Reliability January 2008.
- [5] Log files in open nms <http://unicolet.orgt>
- [6] "OpenNms" <http://opennms.com>
- [7] "SSH," http://en.wikipedia.org/wiki/Secure_Shell
- [8] "Zenmap," <http://nmap.org/zenmap/>
- [9] "Running Hadoop on multi node cluster on Linux," <http://www.michael-noll.com/tutorials/running-hadoop-on-ubuntu-linux-multi-node-cluster>.
- [10] "ipconfigparameters," <http://www.cisco.com/en/US/docs/inte-networking>.
- [11] "Hadoop," <http://www.tutorialspoint.com/hadoop>.
- [12] "Open Nms" <http://sourceforge.net>
- [13] "TOM WHITE book "Hadoop: the definitive guide."
- [14] Yang Liu; Jie Yang; Yuan Huang; Lixiong Xu 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD) "The performance evaluations for Hadoop in diverse system architecture"
- [15] Chaitali Gupta; Mayank Bansal; Tzu-Cheng Chuang; Ranjan Sinha; Sami Ben-romdhane 2014 IEEE International Conference on Big Data (Big Data) "stro: A predictive model for anomaly detection and feedback-based scheduling on Hadoop"
- [16] Sachin BendeRajashree Shedge 2016 Procedia Computer Science "Dealing with Small Files Problem in Hadoop Distributed File System".
- [17] Konstantinos Triantafyllidis „Waqar Aslam Egor Bondarev Johan J. Lukkien Peter H.N. de With- Journal of Systems and Software July 2016 "ProMARTES: Accurate network and computation delay prediction for systems"
- [18] C.K. MorleyC.W. Nixon-Journal of Structural Geology March 2016 " Topological characteristics of simple and complex normal fault networks".