Implementation and performance test of cloud platform based on Hadoop

Jingxian Xu, Jianhong Guo* and Chunlan Ren
TSL School of Business and Information Technology, Quanzhou Normal University, Quanzhou, China
*Corresponding author e-mail: yjsxjx@126.com

Abstract. Hadoop, as an open source project for the Apache foundation, is a distributed computing framework that deals with large amounts of data and has been widely used in the Internet industry. Therefore, it is meaningful to study the implementation of Hadoop platform and the performance of test platform. The purpose of this subject is to study the method of building Hadoop platform and to study the performance of test platform. This paper presents a method to implement Hadoop platform and a test platform performance method. Experimental results show that the proposed test performance method is effective and it can detect the performance of Hadoop platform.

1. Introduction
At present, big data has become the focus of research, there are many solutions for big data. Hadoop is an open source project of the Apache Foundation, which has accumulated a large number of users, and it has also been widely recognized in the industry. Many well-known enterprises have applied Hadoop to their own business domain, these well-known enterprises include Alibaba, Tencent and so on. Hadoop, as an implementation of cloud computing technology, allows users to implement their application logic over the Hadoop framework. One of the goals of cloud computing is to provide high availability computer resources at the lowest possible cost. Hadoop can handle thousands of computer nodes and huge amounts of data, and can automatically handle job scheduling and load balancing, so it's the perfect tool for cloud computing. And Hadoop has the advantages of low cost and high security, so it is very meaningful to study the implementation of Hadoop platform and the performance of the test platform. This paper first studies the implementation of Hadoop platform on the virtual machine, then carries out the benchmark test of Hadoop platform, and finally analyzes the test results. This topic uses the virtual environment to build the cloud platform, this kind of construction way has many advantages, these advantages are to save money and can test the performance of the platform.

2. Implementation of Hadoop Cloud Platform
This article introduces the method of installing and configuring Hadoop in the Linux operating system environment. The installed version of the Linux operating system is Centos 6.5, the installed version of Java is 1.7, and the installed version of Hadoop is 2.7. The main steps for installation and configuration are described below:
2.1. **Server node planning**
The cluster consists of 4 nodes: 1 Master and 3 Slave. First of all, the virtual software VMware is installed on the host, and then 4 computer nodes are cloned by Vmware. The 4 nodes are composed of 1 Master node and 3 Slave nodes. You can install Master nodes first, then use VMware to clone 3 Slave nodes, and configure 4 nodes at the same time, so that the virtual nodes can communicate with each other. The four nodes are installed with the CentOS 6.5 system and have the same user hadoop. Master nodes are responsible for managing distributed data and decomposing tasks; 3 Salve nodes are responsible for storing distributed data and performing tasks. Specific server node planning is shown in table I.

| Host Name | Running Main Process | Assigned IP   | Installed software |
|-----------|----------------------|---------------|-------------------|
| Master    | NameNode, DataNode,  | 192.3.30.1    | Jdk, hadoop       |
|           | DFSZKFailoverController, NodeManager, JournalNode |              | zookeeper         |
| Slave1    | DataNode, NodeManager, JournalNode,QuorumPeerMain,ResourceManager | 192.3.40.1    | Jdk, hadoop       |
|           |                      |               | zookeeper         |
| Slave2    | DataNode, NodeManager, JournalNode,QuorumPeerMain,ResourceManager | 192.3.40.2    | Jdk, hadoop       |
|           |                      |               | zookeeper         |
| Slave3    | DataNode, NodeManager, JournalNode,QuorumPeerMain,ResourceManager | 192.3.40.3    | Jdk, hadoop       |

2.2. **Configure host**
In order to successfully build Hadoop cluster and realize the information transfer among nodes in the cluster, it is necessary to modify the hostname and configure the network environment first. Modify the host name to become Master, the command is HOSTNAME=master, and the Slave node settings refer to the above command. In the network environment for configuring Hadoop clusters, you need to add the IP and hostname of all machines in the /etc/hosts file. In this way, the communication between Master and all Slave machines can not only communicate via IP, but also communicate through the host name. Therefore, add the following contents in the end of the "/etc/hosts" file on all machines:192.3.30.1, master, 192.3.40.1, slave1, 192.3.40.2, slave2, 192.3.40.3, slave3.

2.3. **SSH (Secure Shell) No password login authentication configuration**
In order to successfully build Hadoop cluster and realize the information transfer among nodes in the cluster, it is necessary to modify the hostname and configure the network environment first. Modify the host name to become Master, the command is HOSTNAME=master, and the Slave node settings refer to the above command. In the network environment for configuring Hadoop clusters, you need to add the IP and hostname of all machines in the /etc/hosts file. In this way, the communication between Master and all Slave machines can not only communicate via IP, but also communicate through the host name. Therefore, add the following contents in the end of the "/etc/hosts" file on all machines: (1) to generate two key files in the Master machine, the two files are id_rsa and id_rsa.pub, they are stored in the memory of the "/home/hadoop/.ssh" directory (2) public key file id_rsa.pub content is appended to the end of the file authorized_keys (3) modify the permissions of authorized_keys (4) copy the authorized_keys file from the Master to the Slave1 node. (5) on the Master node, through the SSH landing 3 other Slave nodes, if you do not need password authentication, then the Master node no password login settings successfully. (6) configuring SSH on Slave without password logging is the same as Master. After the configuration is successful, the Master node and the 3 Slave nodes can login to each other without the need to enter a password.
2.4. Java environment configuration
All nodes have to install JDK, first install the JDK at the Master node, and then install the JDK of the other Slave nodes to complete. The following describes the JDK environment configuration. To install JDK and configure environment variables, you need to log in as "root". After the /etc/profile file, add the contents of the three variables of the Java, which are "JAVA_HOME", "CLASSPATH", and "PATH".

2.5. Hadoop cluster configuration
All nodes must have Hadoop installed. First, Hadoop is installed at the Master node, and then Hadoop is installed on the other Slave nodes. To install and configure Hadoop, you need to log in as root. To configure Hadoop, you need to upload the hadoop-2.7.tar.gz file to the /usr folder of the Master machine and extract it. Seven configuration files are modified according to the actual situation of the system. These seven files are hadoop-env.sh, yarn-env.sh, core-site.xml, hdfs-site.xml, mapred-site.xml, yarn-site.xml. In the seven file, we will focus on configuring the hdfs-site.xml file, which is primarily responsible for setting up HDFS related information. Specific configuration information in the cloud computing platform is shown in table II.

| Parameter                | Value                  |
|--------------------------|------------------------|
| DFS. name.dir            | /home/hadoop/hadoop-2.7/dfs/name |
| Dfs.replication          | 3                      |
| dfs.blocksize            | 131072                 |
| dfs.namenode.handler.count | 20                    |

3. Benchmark tests on the Hadoop platform
The test can verify the correctness of the cloud platform and analyze the performance of the cloud platform. Therefore, testing is very important, but testing is often overlooked. In order to have a more comprehensive understanding of the cloud platform, and to find the bottleneck of the cloud platform, and to better improve the performance of the platform, 4 tests are performed on the Hadoop platform. The project intends to test the cloud platform using Hadoop’s own testing tools.

3.1. Mrbench test
In order to detect the efficiency of small job execution, mrbench performs a small job multiple times. This experiment uses mrbench program to test small jobs, and the result is shown in table III.

| Number of repeated implement | 10   | 20   | 50   | 200  | 500  |
|-----------------------------|------|------|------|------|------|
| execution time (s)          | 15.6 | 15.49| 15.43| 15.38| 15.36|

As you can see from the test results in Figure 1, the average time of job execution becomes steadily and slowly decreasing as the number of small jobs increases. From the results of the last two tests, it can be seen that these two times are not very different. Although the number of times has increased by more than 1 times, the execution time has not been reduced by 1 times. Explains that small jobs executed 500 times have reached the limit of this cluster.
3.2. WordCount test

The main function of the WordCount program is to count the number of occurrences of each word in the text file. The WordCount program divides the text files into small files according to certain rules and then inputs them to the Map task. In the Map task, the WordCount program just outputs the frequency of all the different words, and then the Shuffle module and the Reduce module work together to complete the word statistics. Therefore, the CPU resource requirements of the task are very small, and the results of the experiment also verify this. This experiment uses WordCount program to test the files of 100M, 300M, 500M, 1G and 2G respectively, and the result is shown in table IV.

| Size of statistics file | 100M | 300M | 500M | 1G | 2G |
|-------------------------|------|------|------|----|----|
| Normal execution time/s | 36   | 74   | 137  | 368| 852|

![Figure 2. WordCount test results](image)

In Figure 2, the test results show that the execution time of the WordCount program increases with the increase of data size. When the file is relatively small, the execution time increases slowly, the file
size reaches 1G or more, and the execution time increases faster. In general, the file size increases by 1 times, and the execution time is increased by more than 1 times.

3.3. TestDFSIO test
Hadoop has a number of benchmark programs that are packaged in the test program JAR file. TestDFSIO is one of these test programs, and TestDFSIO is used to test the I/O performance of HDFS. Most of the new cloud platform system failures are hard disks. By running I/O intensive tests, you can know the performance of the cluster’s hard disk. TestDFSIO uses the MapReduce job to complete the test, which is a convenient way to read and write files in parallel. Each file are read and written in a separate Map task, Map task can also be used for statistical processing of files, the final statistics data is accumulated in the Reduce task. The following experimental conditions are the same amount of data, but the number of files is different, the results shown in Table V and table VI.

| Single file size/Number of files | 10G/1 | 5G/2 | 2.5G/4 | 1G/10 | 400M/25 |
|----------------------------------|-------|------|--------|-------|---------|
| Write test throughput/(mb·s⁻¹)   | 30.17 | 14.68| 6.89   | 6.13  | 6.46    |
| Write test execution time/s      | 401.78| 452.33|519.89 | 549.92| 555.25  |

| Single file size/Number of files | 10G/1 | 5G/2 | 2.5G/4 | 1G/10 | 400M/25 |
|----------------------------------|-------|------|--------|-------|---------|
| Read test throughput/(mb·s⁻¹)    | 52.56 | 24.29| 4.28   | 3.93  | 3.35    |
| Read test execution time/s       | 254.61| 296.56|705.36 | 756.28| 802.39  |

**Figure 3.** Test the number of different TestDFSIO test results

Figure 3 test results show that the number of files increased from 2 to 4, the execution time increased faster, while the number of files is more than 4, the execution time has not increased significantly. These instructions show that the execution time increases significantly as the number of processing files increases, and when the Reducer parameters are close to the number of cluster nodes, the cloud platform performs faster.
3.4. TeraSort test
The TeraSort algorithm was created by Microsoft database expert Jim Gray. In 2008, Hadoop used the Terasort algorithm to sort 1TB data, taking 209 seconds, ranking first. The Terasort algorithm extracts the abstract from the data first, then distributes the Map output to the Reduce node, and finally completes the sort. The TeraSort program tests the performance of the Hadoop platform by sorting text files. The experiments sorted text files for 100M, 300M, 500M, 1G, and 2G, respectively. The test results are shown in table VII.

| Size of statistics file | 100M | 300M | 500M | 1G  | 2G  |
|-------------------------|------|------|------|-----|-----|
| Normal execution time/s | 33   | 68   | 141  | 288 | 731 |

Fig. 4 test results show that the execution time of Terasort program increases with the increase of data. When the amount of data processed is within 1G, the execution time increases slowly, and when the amount of data processed increases to 2G, the execution time increases very quickly.

4. Conclusion
This paper introduces the implementation of Hadoop cloud computing platform, and then carries out 4 benchmark tests on the Hadoop platform, and finally analyzes the data results. As you can see from the test, the advantages of the Hadoop platform are beginning to emerge as the amount of test data increases. The experiment is only implemented and tested on the virtual machine, and there is still a gap between the data and the reality. Because in actual production, Hadoop platform network size and server performance is very powerful. However, this project has played a role in the performance testing of the Hadoop platform and pointed out the following research directions. Through careful analysis and research of the above test results, we find that the performance of this Hadoop platform needs to be optimized. Hadoop platform can do further optimization research from the parameters configuration and implementation algorithm of Hadoop software architecture. After optimizing these aspects, the performance of the system will be greatly improved.

Acknowledgments
This work was financially supported by:
Fujian Natural Science Foundation of China (Number 2015J01286): Research on e-commerce chain based on Collaborative Perspective in cloud computing environment.
JK project of Fujian Provincial Department of Education (Number JK2014037): Research on e-commerce chain coordination based on Cloud Computing.
Curriculum Construction Project of Quanzhou Normal University: Demonstration Network Course Construction--ERP (Enterprise resource planning).

Students Innovation and Entrepreneurship Training Program Funded Projects of Quanzhou Normal University.

Fujian Province young and middle-aged teacher education research project (Number JAT170494): Research on university teaching resource sharing based on Cloud Computing.

References
[1] Yong Qi Han, Yun Zhang, Shui Yu. Research of Cloud Storage Based on Hadoop Distributed File System [J]. Applied Mechanics and Materials, 2014, 2987(513).
[2] Fan Sujuan, Tian Junfeng. Research and implementation of cloud computing platform based on Hadoop [J]. Computer technology and development, 2016, (07): 127-132.
[3] Xia Jingbo, Wei Zekun, Fu Kai, Chen Zhen. Overview of research and application of Hadoop technology in cloud computing [J]. Computer Science, 2016, (11): 6-11+48.
[4] Wei Dong Fei. Research on Hadoop Technology in Large Data and Cloud Computing [J]. Information Systems Engineering, 2017, (02): 58.
[5] Yao Ye, Chang Guangyan. Hadoop performance analysis based on cloud computing heterogeneous environment [J]. Computer and telecommunications, 2016, (10): 27-29.
[6] Yu Hui, Huang Yongfeng, Hu Ping. Design and implementation of Hadoop storage and management platform for microblogging public opinion [J]. Electronic Technology Applications, 2017, (03): 120-123+131.
[7] Pei Hao. Optimization and implementation of Hadoop platform based on GPU [J]. Fujian computer, 2017, (03): 41-42.
[8] Li Xiaojia, Dong Yanhua. Building and Application Research Based on Hadoop platform [J]. Fujian computer, 2017, (03): 132-133.
[9] Zhang Yan, Wang Wei. Optimization of cloud platform parameters based on Hadoop [J]. Journal of Shenyang Normal University (Natural Science Edition), 2017, (02): 234-239.
[10] Su Shupeng. Design and application of large data solution based on Hadoop [J]. Journal of Hechi University, 2017, (02): 89-93.