Setup and Configuration of MapReduce in a Cloud Environment

Adam Pasqua Blaisse*, Martin Berlove† and Jie Wu*
*Department of Computer and Information Sciences, Temple University, USA
†Department of Computer Science and Engineering, Lehigh University, USA
Email: {adam.blaisse, jiewu}@temple.edu {mbb415}@lehigh.edu

***Abstract*—In this paper we will look at how we went about setting up our clusters. We will work on implementing Apache's open-source software, Hadoop, onto virtual machines in our TCloud Cluster. We will also be implementing Apache's Hadoop in our Net Cloud clusters. Both of these clusters will be running on the open-source software named Eucalyptus. We will discuss installation and configuration of the open-source Eucalyptus cloud onto our Net Cloud cluster. We will also discuss the writing of scripts for both the TCloud Cluster and the Net Cloud cluster, which will setup Hadoop in both the multi-node and single-node configurations. Finally, we will touch on the documentation that we worked on, and we will explain why we did it. Once we have finished presenting what were able to accomplish in both of the clusters available for use, we will discuss our desires for future works, with the paper's conclusion following immediately after.**

***Keywords*—MapReduce, Hadoop, Eucalyptus, OpenFlow.**

I. Introduction

The MapReduce Paradigm was presented by Google's Jeffrey Dean and Sanjay Ghemawat in 2004 [1]. The concept is that a job will consist of two types of tasks. The first is called Map tasks. These tasks take in data and create key-value pairs. The second type of tasks are Reduce tasks. These tasks take as input the outputted key-value pairs from the map tasks. In the MapReduce paradigm, each map task takes in a part of the file and creates a key-value pair (i.e <word,1 >). The number of map tasks are decided based on the size and number of files that need to be read. Then a Reducer will take in the output from a number of mappers. It will then combine pairs based the the keys. The output of the Reducer will look like <word,41 ><word,27 ><word,21 >. One of the most popular implementations of MapReduce is called Hadoop. Hadoop is an open-source implementation written in java by the Apache software foundation [2]. It is used in production by large companies such as Yahoo, and Facebook. Amazon EC2 is a very commonly used cloud service that allows users to start virtual machines in the cloud with different sizes and images[3]. Eucalyptus is an open-source cloud computing environment that is compatible with Amazon's EC2 environment[4]. We decided to use Eucalyptus since it is similar to Amazon's EC2 environment. We will work on setting up the Hadoop MapReduce software on the Eucalyptus cloud environment, as seen in 1.

II. Accomplishments

A. Work within TCloud

TCloud is an experimental HPC cloud environment. TCloud is run on Eucalyptus which is installed on CentOS.

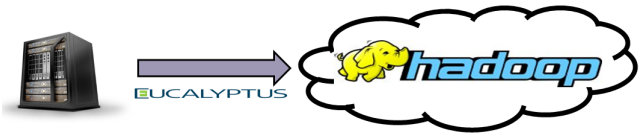

Fig. 1. Hadoop Implemented on Virtual Machines running in the Eucalyptus cloud environment running on 33 servers in the Net Cloud cluster.

Eucalyptus is made to be compatible with Amazon's AWS cloud environment. The TCloud cluster is made of twelve Dell R614 cloud servers for a total of ninety six conventional CPU cores. Each server is interconnected with 4-way redundant 10GB Ethernet and 2-way redundant infiniband [5]. Since this environment is similar to and compatible with the widely-used Amazon AWS environment, and since this environment was already installed and working, we decided to use it for the initial experimentation.

- We worked on setting up and connecting different types of virtual machines inside the Eucalyptus environment that are of different sizes and run on different linux operating systems.
- We next worked on installing Hadoop onto a single virtual machine so that we had a single-node Hadoop cluster configured and running.
- After this, we worked on setting up a multi-node Hadoop cluster on these four new virtual machines.
- Once we were able to get the Hadoop Clusters working on the four virtual machines, we then looked into the different sizes of virtual machines available in the Eucalyptus software.
- We decided on four sizes of machines, and attempted to set up Hadoop on clusters comprised of these types of virtual machines. It took some time to get this working, as different sizes of machines made us change some of the ways we went about things due to space limitations. We were eventually able to get the Hadoop clusters set up on all the four different types of machines.
- Once we had the clusters working on the different sizes of machines, we then worked on creating scripts that would automatically set up Hadoop in the virtual machines. The first script that we wrote was a script to set up a single-node Hadoop system on a machine.

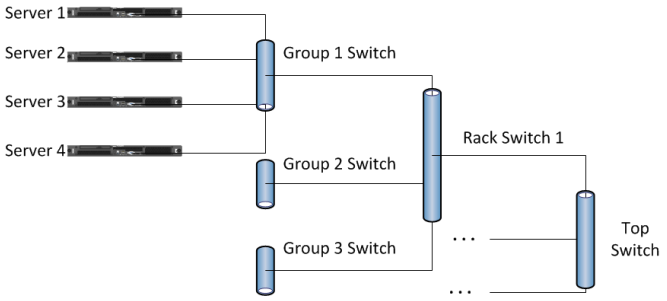

Fig. 2. Topology of Net Cloud Cluster Tree Setup

This allowed us to set up Hadoop and then simply manually change some configuration files to connect all the machines together.

- Finally, we worked on creating a script that would automate the setup and configuration of Hadoop on all the virtual machines at the same time.

B. Work within Net Cloud

After we had become comfortable with working on and with the virtual machines on TCloud, we went about setting up our own Eucalyptus cluster. We got thirty three Dell R210 Servers and twelve Cisco Small Business 300 series managed switches. Each switch has ten gigabit ports with the ability to drop any port to 100 or 10 megabit speed. These servers and switches are installed into three 42U racks. The servers and switches are connected in a tree structure as seen in Fig. 2.

- After we installed all of the hardware into the three racks, we went about installing CentOS onto the servers. We used CentOS because it is one of only a few versions of Linux that supports Eucalyptus.
- Once we had CentOS installed and connecting correctly, we then moved on Eucalyptus. We first went about installing Eucalyptus onto a single server.
- Once the setup and configuration was completed, we installed a few different images into the cluster to use when starting virtual machines. Since we only had one server in our Eucalyptus cluster, we could only start 2 virtual machines at a time.
- We then started a new virtual machine and installed Hadoop onto it; afterward, we started the second machine and worked on having a two-node Hadoop cluster.
- Next we worked on modifying both the single-node and multi-node scripts from the Tcloud cluster so that they will work on the Net Cloud cluster.
- Finally, we worked on adding documentation about what had been accomplished and how it was done.

III. Insights

We had a few different insights about working with Eucalyptus and Hadoop. The first is that Hadoop is very sensitive to having the correct host name in the configuration files. A second insight we had was that Open-Source software is useful but can be cumbersome and can often react poorly to system changes. A third insight, was that the often poor documentation of some open-source software can create more time being spent on making the software work then using the software, itself.

IV. Future Work

In the future, we hope to get all thirty three nodes added into the Net Cloud. Once the nodes are added into the Net Cloud, we plan to look into the effects of different-sized virtual machines on Hadoop. We hope to compare the sizes of the virtual machines in Eucalyptus to the virtual machines in Amazon's EC2. We will then look at ways of selecting the best cluster of virtual machines with a limited budget. The pricing of virtual machines will be based on the cost of different virtual machines in the Amazon EC2 environment. Since we control the cluster, we are hoping in the future to purchase a switch that supports a software-defined network such as Open Flow. One of the things we would like to do once we get Open Flow switches is to try different topologies and see what effects they have on Hadoop. We would also like to use Open Flow to try and implement an algorithm that would allow us to dynamically change the topology of the network. We believe that this could be very beneficial, as it would allow us to deal with peak demands and move bandwidth away as a connections become unpopular. In the future, we would also like to work on writing our own scheduler for Hadoop. We have a few different things we would like to consider adding in our own scheduler. The first thing would be the placement of the map and reduce tasks within a Hadoop cluster. We would like to write a scheduler that better selects the locations for both of these tasks. Finally, we would like to look at using machine learning techniques to better select the placement of map and reduce tasks within the Hadoop cluster.

V. Conclusion

In this paper, we have presented our works with two major Open-Source softwares. We talked about our works within our two research clusters. The first cluster, was our TCloud HPC Cluster, and we talked about the configuration of the cluster and our works we did within it. The second cluster was the Net Cloud cluster, and we talked about the configuration and how to set up the different parts of this cluster. We talked about what we were able to accomplish within these two clusters. We then talked about where we would like to go from here, and what we hope to accomplish in our future works. Finally, we looked into some of the insights that we felt we gained from working with both of the above two clusters, as well as open-source software in general.

Acknowledgements

Martin Berlove is supported by NSF/DoD REU Site, CNS 1156574.

References

[1] J. Dean and S. Ghemawat, "Mapreduce: Simplified data processing on large clusters 6th symposium on operating system design and implementation," *San Francisco*, 2004.
[2] [Online]. Available: http://hadoop.apache.org/
[3] [Online]. Available: http://http://aws.amazon.com/ec2/
[4] [Online]. Available: http://www.eucalyptus.com
[5] [Online]. Available: https://sites.google.com/a/temple.edu/tcloud/