Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 82 (2016) 99 – 106

Procedia Computer Science

Symposium on Data Mining Applications, SDMA2016, 30 March 2016, Riyadh, Saudi Arabia

Performance Comparison of Spark Clusters Configured Conventionally and a Cloud Service

Hameeza Ahmed[a], Muhammad Ali Ismail[a,]*, Muhammad Faraz Hyder[a], Syed Muhammad Sheraz[a], Nida Fouq[a]

[a]*High Performance Computing Centre (HPCC), Department of Computer & Information Systems Engineering, NED University of Engineering & Technology University Road, Karachi-75270, Pakistan*

Abstract

Apache Spark is an open source cluster computing technology specifically designed for large scale data processing. This paper deals with the deployment of Spark cluster as a cloud service on the OpenStack based cloud. HiBench benchmark suite is used to compare the performance of Spark cluster as a service and conventional Spark cluster. The results clearly depict how Spark as a cloud service gives more promising outcomes in terms of time, effort and throughput.

Keywords: OpenStack; Spark Cluster; HiBench; Cloud Service; Benchmark;

1. Introduction

With the recent developments in technology, there are numerous sources which are contributing towards the generation of huge amount of data. These sources include now sensor equipped areas, RFID, social networks, large-scale eCommerce, phones, credit cards, atmospheric science, medical records, trains, buses, biological, astronomy, genomics, military surveillance, video archives, photography archives, and the internet of things. Big data means data that’s *too fast, too big,* or *too hard* for existing tools to process in fact it refers to datasets whose size is beyond the ability of classical database software tools to manage, capture, store, and analyze. Big data is not a single technology, but it involves many of existing fundamental concepts such as parallel processing, distributed file systems, in-memory databases virtualization, and many more [1], [8].

Big data computing is a big challenge in the present era in various aspects including storage, processing, management, analytics, visualization etc. Amongst this data processing is the most challenging aspect. In order to

doi:10.1016/j.procs.2016.04.014

process such huge quantity of data, there exists a variety of programming frameworks namely Apache Hadoop, Apache Spark, HPCC, HPCC Systems (High Performance Computing Cluster), Storm, Lambdoop etc [2].

Apache Spark is an open source big data programming model. It was started as a research project in the AMPLab at UC Berkeley. It is a general-purpose cluster computing engine with libraries for streaming, machine learning, and graph processing. Additionally, it has APIs in Java, Python, and Scala. Spark is an open source big data framework primarily designed for three major objectives namely ease of use, sophisticated analytics, and speed. The processing speed of Spark is increased due to its feature of in-memory cluster computing [3], [4], [5].

With the complexities and challenges involved in big data computing, the need for large computational infrastructure, expensive software, and effort is also raised. Cloud computing provides the ultimate solution to these problems. It does so by providing on-demand resources and charging as per the actual resource consumption. Besides, it allows the infrastructures to be scaled down and up rapidly, adjusting the system to the actual demand [6]. Cloud computing is powerful enough to perform complex and massive-scale computing. It eliminates the need of onsite maintenance of expensive dedicated space, computing hardware, and software [7].

The primary objective of this paper is the deployment of Apache Spark cluster as a cloud service (SAAS) on OpenStack cloud. There are several benefits of providing Apache Spark as SAAS namely scalability, backup and restore facility, ease of use, high speed, increased throughput, lower cost and many others [8]. The work being presented in this paper makes an in-depth analysis of the performance of Spark cluster as a SAAS. It does so by comparing the results of a Spark cluster configured as cloud service with the conventional one. The comparison is done using a complete big data benchmark suite known as HiBench. In order to perform the comparison, total nine benchmarks are executed on both the implementations of Spark cluster. These benchmarks include four separate categories namely Micro benchmarks, Web Search, Machine Learning and Analytical Querying. The benchmarks namely Sort, WordCount, Sleep, and TeraSort are included in micro benchmarks category. The Web Search benchmarks include PageRank while the Machine Learning category is comprised of Bayesian Classification. The analytical query involves three benchmarks namely Hive Join, Scan and Hive Aggregate respectively [9]. The final results clearly depict how apache Spark cluster deployed on OpenStack dominates the conventional cluster both in terms of speed and throughput.

Rest of the paper is organized as follows: Section 2 discusses related work. Apache Spark cluster deployment both as a SAAS on cloud and the conventional cluster is shown in section 3. Section 4 provides description about the benchmark suite HiBench and performance metrics used in the experiment. A detailed comparison of Spark performance results is covered in Section 5. The final conclusion is presented in section 6.

2. Related Work

OpenStack and Apache Hadoop represent the largest open source communities. Their integration will benefit users of both of these communities. Considering this, there has been some efforts in this area most notable of them is the project SAHARA [11]. This integration manages and configures Hadoop cluster in the cloud. The project SAHARA has now been extended for Spark Support but currently Spark can only installed in standalone mode, with no YARN or Mesos support [12]. Our research is novel in the way that this research is succeeded in implementing Spark cluster as a service in an Openstack cloud in distributed mode with full YARN support.

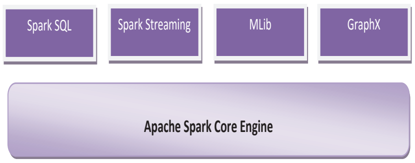

Fig. 1. Components of Spark [4]

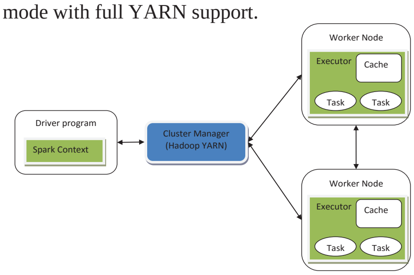

Fig. 2. Apache Spark Cluster Manager [4]