UNIVERSITI PUTRA MALAYSIA

WORKFLOW SYSTEM FOR MAPREDUCE IN CLOUD ENVIRONMENT

MUNTADHER SAADOON WADI

FSKTM 2017 6

WORKFLOW SYSTEM FOR MAPREDUCE IN CLOUD ENVIRONMENT

By

MUNTADHER SAADOON WADI

Thesis Submitted to Universiti Putra Malaysia,

in Fulfilment of the Degree of Master Computer Science

July 2017

Abstract of thesis presented to the Senate of Universiti Putra Malaysia in fulfillment of the requirements for Master of Computer Science

WORKFLOW SYSTEM FOR MAPREDUCE IN CLOUD ENVIRONMENT

By

MUNTADHER SAADOON WADI

July 2017

ABSTRACT

The magnitude of data generated and shared by businesses, public administrations, industrial sectors and scientific research, has increased immeasurably. Apache Hadoop is an open source software framework, which enables a scalable and distributed processing of high volumes of data. MapReduce together with its Hadoop implementation has been widely adopted in many practical applications. A common practice nowadays is to implement MapReduce applications in a high-performance infrastructure, such as cloud computing. A cloud platform can deploy and manage Hadoop clusters. However, there are tasks required advanced knowledge in computer science and cloud computing when using MapReduce technology that prevent the usage of current technologies and software solutions. For example, MapReduce deployment and maintenance, data integration with Hadoop distributed file system or MapReduce job submission. A MapReduce workflow system is one of the solution that could assist MapReduce and Hadoop developers. Besides, it provides a user-friendly execution platform that encapsulating complexity of data analysis steps. In this research, a new workflows system is developed to facilitate the use of collaborating, coordinating and executing operations of MapReduce programs with a graphical user interface based on Hadoop cloud cluster. The experimental results indicate that the developed workflow system can achieve good speed in performance. It is believed that the workflow system is an ideal stereotype for MapReduce and it will play an important role in the era of big data applications in cloud computing.

Abstrak tesis yang dikemukakan kepada Universiti Putra Malaysia sebagai memenuhi keperluan Ijazah Sarjana Sains Komputer

SISTEM ALIRAN KERJA BAGI MAPREDUCE DI DALAM PERSEKITARAN CLOUD

By

MUNTADHER SAADOON WADI

Julai 2017

ABSTRAK

Magnitud data yang dijana dan dikongsi oleh sektor perniagaan, pentadbiran awam, sektor industri dan penyelidikan saintifik, telah meningkat dengan ketara. Apache Hadoop adalah kerangka kerja perisian sumber terbuka, yang membolehkan pemprosesan boleh skala dan teragih bagi data berjumlah tinggi. MapReduce bersama dengan pelaksanaan Hadoop telah digunakan secara meluas di dalam kebanyakkan aplikasi praktikal. Amalan biasa pada masa kini adalah untuk melaksanakan aplikasi MapReduce di dalam infrastruktur berprestasi tinggi seperti pengkomputeran awan. Pelantar awan boleh meletak atur dan mengurus gugusan Hadoop. Walau bagaimanapun, terdapat tugasan yang memerlukan pengetahuan lanjut di dalam bidang sains komputer dan pengkomputeran awan apabila menggunakan teknologi MapReduce yang boleh menghalang penggunaan teknologi dan penyelesaian perisian sedia ada. Sebagai contoh, peletak aturan dan penyelenggara MapReduce, integrasi data dengan sistem fail teragih Hadoop atau penyerahan kerja MapReduce.

Sistem aliran kerja MapReduce adalah salah satu kaedah penyelesaian yang dapat membantu pembangun MapReduce dan Hadoop. Selain itu, ianya menyediakan satu pelantar pelaksanaan yang mesra pengguna merangkumi langkah analisis data yang kompleks. Dalam kajian ini, satu sistem aliran kerja baru dibangunkan untuk memudahkan penggunaan untuk kolaborasi, koordinasi dan pelaksanaan operasi program MapReduce dengan antara muka pengguna grafik berasaskan Hadoop gugusan awan. Hasil eksperimen menunjukkan bahawa sistem aliran kerja yang dibangunkan dapat mencapai kelajuan yang baik di dalam aspek prestasi. Adalah dipercayai bahawa sistem aliran kerja adalah stereotaip yang ideal untuk MapReduce dan ianya akan memainkan peranan penting di dalam era aplikasi data bersaiz besar dalam pengkomputeran awan.

ACKNOWLEDGEMENT

First of all, all thanks and praise goes to God for His will and for bestowing me with health, time and maturity of mind to study.

Million thanks to Novia Indriaty Admodisastro being my supervisor during master degree. She has provided assistance, guidance and constructive comments during the process of completing this thesis.

I would like to infinitely thank and appreciate my father, Saadoon Wadi Najim, mother, Jinan Naaem, siblings and friends who always support, inspire, and encourage me to complete this project.

Special thanks to Lina, who gives me the power to always look forward and never give up.

Finally, to all UPM staff, thanks you for your facilitation. I would like to acknowledge to any individual who are not mentioned here for his/her irreplaceable helps and cooperation.

APPROVAL

Thesis submitted to the Senate of University Putra Malaysia and has been accepted as fulfillment of the requirement for Master of Computer Science (Software Engineering).

Supervisor,

Dr.Novia Indriaty Admodisastro

Department of Software Engineering

Faculty of Computer Science and Information Technology

Universiti Putra Malaysia

Date: 10/07/2017

DECLARATION

I declare that the thesis is my original work except for the quotations and citations, which have been duly acknowledged. I also declare that it has not been previously, and is not concurrently, submitted for any other degree at Universiti Putra Malaysia or at any other institution.

MUNTADHER SAADOON WADI

Date : 10/07/2017

TABLE OF CONTENT

LIST OF TABLES

Table **Page**

LIST OF FIGURES

LIST OF ABBREVIATIONS

Abbreviations	Meaning
UPM	Universiti Putra Malaysia
IT	Information Technology
NIST	National Institute of Standards and Technology
OSI	Open Source Software
UML	Unified Modeling Language
HTML	HyperText Markup Language
CSS	Cascading Style Sheets
VM	Virtual Machine
HDFS	Hadoop Distributed File System
MR	MapReduce
ES2	Amazon Elastic Compute Cloud
SaaS	Software as a Service
PaaS	Platform as a Service
IaaS	Infrastructure as a Service
JavaEE	Java Enterprise Edition

CHAPTER 1

INTRODUCTION

1.1 Overview

According to Hashem et al., (2015) the constant rise in the amount and detail of data occupied by organizations, such as the increase of social media, Internet of Things (IoT), and multimedia, has produced an unusual flow of data formats structured, semi-structured and unstructured. Big data cites awareness from the academia, government, and enterprises. Big data are categorized by three characters: (1) data are infinite, (2) data cannot be classified into traditional relational databases, and (3) data are produced, grown, and processed rapidly. Besides, big data is transforming health-care, physics, manufacturing, economics, marketing, and finally, the society. The velocity at which new data are being produced is tremendous. A major challenge for researchers and practitioners is that this increased rate eclipses their ability and knowledge to design proper cloud computing platforms for data analysis and update-intensive workloads (Hashem et al. 2015).

Cloud computing is one of the hugely significant turns in latest Information and communications technology and service for enterprise software. In addition, it is a robust architecture to achieve large-scale and heterogeneous computing. It has been demonstrated quite successful in big data applications. The major step is to drive jobs from traditional servers to cloud servers. Cloud servers provide higher services to users rather than a service that contributed by personal machines or small servers. The two significant benefits of cloud computing are scalability and efficiency (Ma et al. 2016). MapReduce is the most common parallel programming model to analysis vast amount of data on Hadoop clusters in cloud platforms. MapReduce is the native programming language of several tools that have been developed on the top of Hadoop ecosystem such as Apache Pig (Olston et al. 2008), Apache Hive (Thusoo et al. 2009), and code translation applications (Li et al. 2016; Zhang et al. 2013). These tools have significantly improved the productivity of writing MapReduce programs. However, in practice, auto-generated MapReduce programs have to be observed for many queries that are often extremely inefficient compared to native MapReduce programs by experienced programmers (Olston et al. 2008). In addition, executing MapReduce programs on cluster architecture poses considerable challenges for IT research laboratories that interesting in using MapReduce programming model (Chung et al. 2014; Ko, Park, and Version 2017; Schönherr et al. 2012) . In order to address this problem, several recent workflow systems have emerged specifically for MapReduce programming model to facilitate the use of MapReduce in cloud platform, for example, Cloudgene (Schönherr et al. 2012), CloudDOE (Chung et al. 2014) and cl-dash (Hodor et al. 2016).

In this research, we intend to design and develop a workflow system for MapReduce program in Hadoop cloud cluster. The main objective of the workflow system is to facilitate the use of collaborating, coordinating and executing operations of MapReduce programs with a graphical user interface based on Hadoop cloud cluster.

1.2 Problem Statement

The magnitude of data generated and shared by businesses, public administrations numerous industrial and not-to-profit sectors, and scientific research, has increased immeasurably (Sivarajah et al. 2016). Apache Hadoop is an open source software framework, which enables a scalable and distributed processing of high volumes of data (Vavilapalli et al. 2013). MapReduce together with its open-source implementation Hadoop has been widely adopted in many practical data processing applications (Dean & Ghemawat, n.d. 2008). A common practice nowadays is to process these data in a high-performance computing infrastructure, such as cloud (Hashem et al. 2015). Cloud platform has the ability to deploy and manage Hadoop clusters (Thaha et al. 2016). However, there are four major problems when emerging these fundamental technologies, which are listed below.

First, due to the tremendous volume of data sets being collected and analyzed daily, companies scale up hundreds of systems into their data centers to handle this growth of data (Computing et al. 2016). In addition, as long as data centers have increased numbers of systems, multiple configurations and maintenance requirements have to be increased concurrently. Resulting, traditional solutions for collecting and analyzing data are prohibitively expensive (Gates et al. 2009).

Second, Hadoop is an open source software framework which implements MapReduce model for controlling big data operations. It provides a Hadoop Distributed File System (HDFS) as the global file system running on a cluster (Shvachko 2010). According to Ma et al., 2016, setting up Hadoop cluster in a distributed environment is not an easy task. Developers have challenges in installing and configuring Hadoop cluster because of Hadoop complexity that leads them to spend more time and effort on Hadoop deployment.

Third, MapReduce is the best option being used as a parallel processing system framework. However, MapReduce is very low level and rigid. Besides, it requires developers to write custom codes that are complex to be reused and maintained (Li et al. 2016; Ma et al. 2016; Olston et al. 2008; Thusoo et al. 2009).

Fourth, MapReduce offers a simple dataflow programming model that appeals to many users especially who is familiar with Java programming language. Nevertheless, in MapReduce codes implementation, the complete simplicity of the Map-Reduce programming model drives to 1) MapReduce job optimization has to be manually manipulated by its developers. 2) Map-Reduce also lacks explicit support for combined processing of multiple data sets. 3) Frequently-needed data manipulation primitives like filtering, aggregation, and top-k thresholding, must be coded by hand (Gates et al. 2009).

As known, workflow systems can streamline the design and execution of workflows in high-performance computing settings such as local or distributed cloud clusters (Spjuth et al., 2015). Besides, these are many workflow systems and software tools attempted to solve this problem. However, there was a lack in performance and some features (e.g. collaboration, reusability, and run time interruption) were not fully realized in their current versions (Li et al. 2016; Ma et al. 2016; Schönherr et al. 2012).

1.3 Research Objectives

The objectives of this study are listed as below:

- To derive the characteristics of a workflow system for MapReduce programming model.
- To design and develop a workflow system for MapReduce in cloud environment.
- To evaluate the performance of the workflow system.

1.4 Research Scope

The research focuses on MapReduce programming model and big data analysis based on Hadoop cloud cluster. The cloud environment used to carry out the proposed system is OpenStack cloud software (Kumar and Parashar 2014). In addition, OpenStack will handle both big-data resources (e.g. Hadoop nodes) and the workflow system. The workflow system will be designed and developed as web-based application using Java EE. The following Figure 1.1 shows the major tools that will be used in this research.

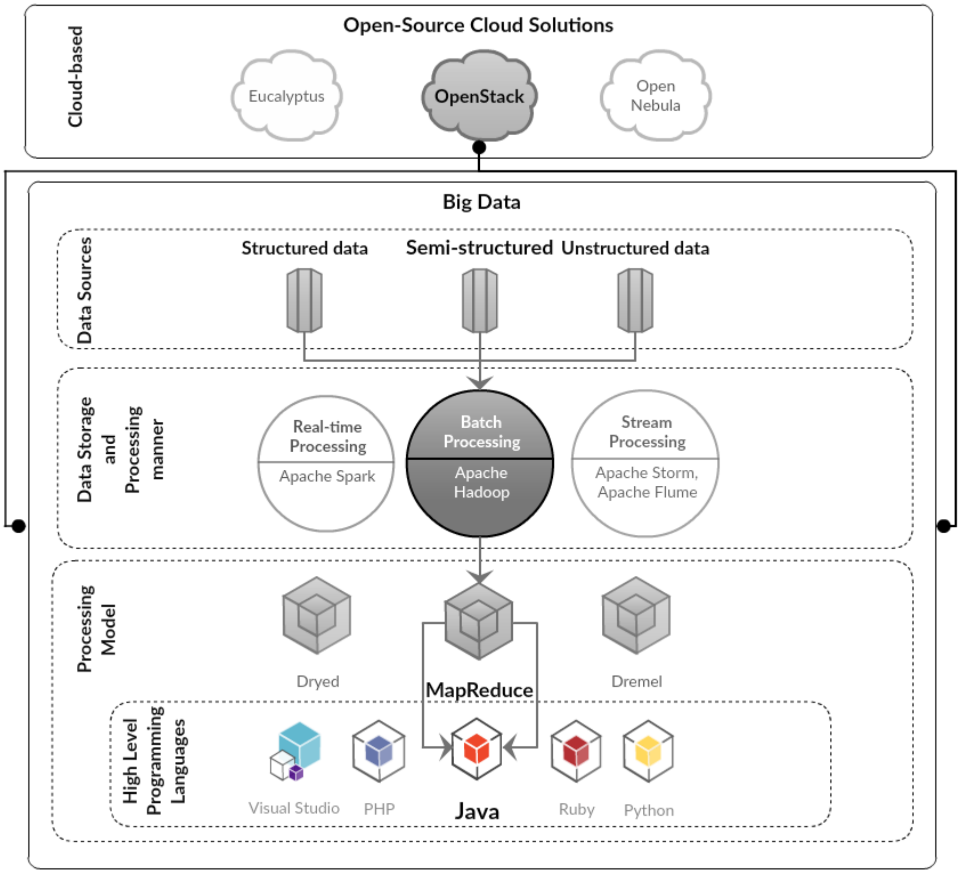

Figure 1.1: *Research Scope*

1.5 Thesis Organization

This thesis is divided into six chapters. The following paragraphs provide a brief description of the remaining chapters of this thesis: Chapter 2 discusses workflow systems and code generation tools based on Hadoop that are used for MapReduce model in order to simplify the use of complex workloads. Chapter 3 introduces different phases of the research methodology of this research. Chapter 4 describes the design, implementation and proves the concepts of the proposed workflow system. Chapter 5 demonstrates the performance of the developed workflow system. Chapters 6 present the conclusion together with thesis limitation and identify some areas for future work.

REFERENCES

Anon. n.d. *OpenStack Open Source Cloud Computing Software*. Retrieved November 3, 2016 (http://www.openstack.org).

Chung W-C, Chen C-C, Ho J-M, Lin C-Y, Hsu W-L, Wang Y-C, et al. (2014) CloudDOE: A User-Friendly Tool for Deploying Hadoop Clouds and Analyzing High-Throughput Sequencing Data with MapReduce. PLoS ONE 9(6): e98146. https://doi.org/10.1371/journal.pone.0098146

Computing, High Performance et al. 2016. "BIG DATA DEVELOPMENT PLATFORM FOR ENGINEERING." *IEEE International Conference on Big Data (Big Data)* 2699–2702.

Dean, Jeffrey and Sanjay Ghemawat. n.d. "MapReduce: Simplified Data Processing on Large Clusters". *Communications of the ACM* Volume 2008 107-113

Deelman, Ewa, Dennis Gannon, Matthew Shields, and Ian Taylor. 2009. "Workflows and E-Science : An Overview of Workflow System Features and Capabilities." *Future Generation Computer Systems* 25(5):528–40. Retrieved (http://dx.doi.org/10.1016/j.future.2008.06.012).

Ferreira, Rafael et al. 2017. "A Characterization of Workflow Management Systems for Extreme-Scale Applications." *Future Generation Computer Systems*. Retrieved (http://dx.doi.org/10.1016/j.future.2017.02.026).

Gates, Alan F. et al. 2009. "Building a High-Level Dataflow System on Top of Map-Reduce: The Pig Experience." *Vldb '09* 1–12. Retrieved (papers3://publication/uuid/08DB3582-AD5E-47BB-A370-589985B45F92).

Hashem, Ibrahim Abaker Targio et al. 2015. “The Rise Of ‘big Data’ on Cloud Computing: Review and Open Research Issues.” *Information Systems* 47:98–115. Retrieved (http://dx.doi.org/10.1016/j.is.2014.07.006).

Hodor, Paul, Amandeep Chawla, Andrew Clark, Lauren Neal, and Booz Allen Hamilton. 2016. “Sequence Analysis Cl-Dash : Rapid Configuration and Deployment of Hadoop Clusters for Bioinformatics Research in the Cloud.” *Bioinformatics* 32(October 2015):301–3.

Kaur, Simranjit and Sumesh Sood. 2016. “A Survey Paper on the Evaluation Criteria of Open Source Cloud Computing Solutions.” International Journal of Computer Science and Mobile Applications,Vol.4 Issue. 6, June- 2016, pg. 6-12 4:6–12.

Kiran, Mariam, Peter Murphy, Inder Monga, Jon Dugan, and Sartaj Singh Baveja. 2015. “Lambda Architecture for Cost-Effective Batch and Speed Big Data Processing.” *Proceedings - 2015 IEEE International Conference on Big Data, IEEE Big Data 2015* 2785–92.

Ko, Mi-eun, Young B. Park, and A.Apache Hadoop Version. 2017. “Automatic Generation of UML Model-Based Image Processing Source Code in Hadoop Platform.” I*nternational Conference on Platform Technology and Service* 2–5.

Kumar, Rakesh and Bhanu Bhushan Parashar. 2014. “Dynamic Resource Allocation and Management Using OpenStack.” cnsm (November):1–5.

Li, Bing, Junbo Zhang, Ning Yu, and Yi Pan. 2016. “J2M: A Java to MapReduce Translator for Cloud Computing.” *Journal of*

Supercomputing 72(5):1–18.

Ma, Zhiqiang, Shuangtao Yang, Zhida Shi, and Rui Yan. 2016. “Online Integrated Development Environment for MapReduce Programming.” *International Journal of u- and e- Service, Science and Technology* 9(6):399–408.

Mandal, Bichitra, Ramesh Kumar Sahoo, and Srinivas Sethi. 2015. “Architecture of Efficien T Word Processing Using Hadoop MapReduce for Big Data Applications.” *International Conference on Man and Machine Interfacing (MAMI)*.

Olston, Christopher, Benjamin Reed, Utkarsh Srivastava, Ravi Kumar, and Andrew Tomkins. 2008. “Pig Latin.” *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data - SIGMOD ’08* 1099. Retrieved (http://infolab.stanford.edu/~olston/publications/sigmod08.pdf%5Cnhttp://portal.acm.org/citation.cfm?doid=1376616.1376726).

Peng, Junjie et al. 2010. “Comparison of Several Cloud Computing Platforms.” *2nd International Symposium on Information Science and Engineering, ISISE 2009* 23–27.

Schönherr, Sebastian et al. 2012. “Cloudgene : A Graphical Execution Platform for MapReduce Programs on Private and Public Clouds.” *BMC Bioinformatics* 1–9.

Shvachko, K. 2010. “The Hadoop Distributed File System.” *IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST)* 1–10.

Shvachko, Konstantin, Hairong Kuang, Sanjay Radia, and Robert Chansler. 2010. “The Hadoop Distributed File System.” *2010 IEEE 26th Symposium on Mass Storage Systems and Technologies,* *MSST2010* 1–10.

Sidhu, Ravneet Kaur and Charanjiv Singh Saroa. 2016. “Efficient Batch Processing of Related Big Data Tasks Using Persistent MapReduce Technique.” *Proceedings of the Third International Symposium on Computer Vision and the Internet* 106–9. Retrieved (http://doi.acm.org/10.1145/2983402.2983431).

Sivarajah, Uthayasankar, Muhammad Mustafa Kamal, Zahir Irani, and Vishanth Weerakkody. 2016. “Critical Analysis of Big Data Challenges and Analytical Methods.” *Journal of Business Research* 70:263–86. Retrieved (http://dx.doi.org/10.1016/j.jbusres.2016.08.001).

Spjuth, Ola et al. 2015. “Experiences with Workflows for Automating Data-Intensive Bioinformatics.” *Bioinformatics* 1–12.

Thaha, Asmath Fahad, Anang Hudaya, Muhamad Amin, Subarmaniam Kannan, and Nazrul Muhaimin Ahmad. 2016. “Data Location Aware Scheduling for Virtual Hadoop Cluster Deployment on Private Cloud Computing Environment.” 103–9.

Thusoo, Ashish et al. 2009. “Hive - A Warehousing Solution Over a Map-Reduce Framework.” *Sort* 2:1626–29. Retrieved (http://portal.acm.org/citation.cfm?id=1687609).

Vavilapalli, V. K. et al. 2013. “Apache Hadoop Yarn: Yet Another Resource Negotiator.” *Annual Symposium on Cloud Computing* 5.

Zhang, Junbo, Dong Xiang, Tianrui Li, and Yi Pan. 2013. “M2M : A Simple Matlab-to-MapReduce Translator for Cloud Computing.” *Tsinghua Science and Technology* 18(1).