THE PERFORMANCE EFFICIENCY OF THE VIRTUAL HADOOP USING OPEN BIG DATA

Martin Lněnička, Jitka Komárková

Abstract: *Public sector institutions nowadays maintain a large amount of data from various domains. This data represents a potential resource that businesses and citizens can use to enhance their own datasets or which can be used to develop new products and public services. Open data support the emergence and realization of the big data potential. While it enhances the volume and velocity of available data, its main impact is on the variety of data sources. This paper deals with the deployment of the Virtual Hadoop for the processing of the open big data idea in the public sector. The first part of this paper is based on the literature review of the cloud computing, the distributed processing of data, big / open / linked data and theirs sources on the web. The primary aim of the Virtual Hadoop deployment is to test the performance efficiency using open big data in order to obtain the direction of the future research. The last part then introduces the most important findings and recommendations.*

Keywords: *Open big data, Virtual Hadoop, Data processing, Performance efficiency, Cloud computing, Public sector.*

JEL Classification: *C55, C63, H83, L86.*

Introduction

The volume of data being made publicly available increases every year. The emergence of big and open data use is yet another phase of the ongoing Information and Communication Technologies (ICT) revolution which resembles the previous technology-driven economic transitions [3]. However, public sector and governments are not the only possible sources of the open data. Others include businesses, research institutions and citizens. Thus, the open data can be distinguished into four categories: Open Government Data (OGD), Open Business Data (OBD), Open Citizen Data (OCD) and Open Science Data (OSD). In the age of Web 2.0 and social media new big data flows have emerged, those provided by the businesses, citizens as well as public sector institutions, themselves into the open big data era. Resulting adaptability and efficiency of this era the new opportunities and also threats for the public sector institutions are raised, forcing them to adapt to the new reality and adopt the open big data flows. In practice, gaining access to raw data, placing it into a meaningful context, processing the data and extracting valuable information from them is often extremely difficult. As a result, during the last couple of years different solutions have been developed to support the whole lifecycle of the open big data reuse i.e. data discovery, cleaning, integration, browsing and visualization [9], [12].

Open data are an important part of the transparent public administration. Collection and dissemination of information and data are key tools of government. Governments gather large amounts of data and hold significant national datasets. The public sector is charged with the responsibility of offering and providing effective and efficient services and maintaining infrastructure for citizens as well as businesses.

Large-scale data-intensive cloud computing with the MapReduce framework is becoming increasingly popular of many academic, government, and mostly industrial organizations. Apache Hadoop and its deployment model Virtual Hadoop, an open source project, is by far the most successful realization of MapReduce framework. While MapReduce is efficient and reliable for data-intensive computations, the excessive configuration parameters in Apache Hadoop impose unexpected challenges on running various workloads with a cluster effectively [25].

1 Problem formulation and the tools used

Public sector institutions have collected large amounts of data long before the age of digitization, e.g. during censuses, tax collection or welfare provision. ICT helps to handle these vast amounts of information and use them to find inefficiencies in public sector as well as to provide adequate evidence for policymakers. The concepts of open and linked data also have led to the necessity to process these large amounts of data very quickly to retrieve valuable information. With cheaper computational power and storage available in the form of scalable cloud solutions and large scale computational clusters, it has become possible to extend the techniques developed in the open big data processing to empower businesses and citizens and cost savings by developing innovations and solutions that improve the quality of public services.

The main goal of this paper is to introduce the benefits and risks of the open big data era and describe a solution to process these data with the use only open-source solutions, so they could be used in the public sector without additional fees and licenses. The aim of this paper is two-fold: firstly a review of the related works of different types of computational models and techniques for the data processing will be discussed. Finally, a case study in which will be proposed, implemented and evaluated the optimal performance of the Virtual Hadoop cluster to help process the open big data will be presented.

The research is mostly based on literature review of foreign and domestic resources which should lead to make recommendations on the definition and development of the open big data processing on the basis of study of the scientific publications in the field of the public sector, data processing, Apache Hadoop and performance evaluation. The case study consists of the deployment of a virtual cluster using MapReduce paradigm with the framework Apache Hadoop for the processing of the open big data using the standard WordCount algorithm which is used in most tutorials to MapReduce – e.g. in [10]. It reads text files and counts how often words occur. The last part then contains results and recommendations for the further research. The main tools used are Apache Hadoop 2.2.0, VirtualBox 4.3.8, Ubuntu Server 12.04 and Java 7.

2 Related work and background

The growth of data sources and the ease of access that ICT affords, also brings new challenges on data acquisition, storage, management and analysis. Traditional data management platforms, analysis systems and tools are still based on the Relational Database Management System (RDBMS). However, such RDBMSs only apply to structured data, other than semi or unstructured data, and are increasingly utilizing more and more expensive hardware. It is apparently that the traditional RDBMSs could not handle the huge volume and heterogeneity of the big data [7]. Tien in [22] compares major differences between the big data approach and the traditional data management approach with the four

stages (acquisition, access, analytics, application) and three elements (focus, emphasis, scope).

Data held by the public sector institutions has a great reuse potential. Buchholtz et al. [3] in their study estimates that aggregate direct and indirect economic impacts from use of the open big data across the whole EU28 economy are of the order of billions EUR annually. The resulting economic gains can be put into three broad categories: resource efficiency improvements through reducing the information concerning resource waste in production, distribution and marketing activities, product and process improvements through innovation based on R&D activities, day-to-day process monitoring and consumer feedback, management improvements through evidence based, data-driven decision making.

Kucera and Chlapek [13] then present a set of benefits that can be achieved by publishing OGD and a set of risks that should be assessed when a dataset is considered for opening up. They introduce these benefits of OGD: increased transparency, improved public relations and attitudes toward government, increased reputation of a public sector instituion, transparent way of informing the general public about infringement of legislation, improved government services, improved government data and processes, better understanding and management of data within public sector bodies, supporting reuse, increasing value of the data, stimulating economic growth, minimizing errors when working with government data, easier translations and less requests for data. A set of possible risks to OGD publication contains: publication of data against the law, trade secret protection infringement, privacy infringement, risk to the security of the infrastructure, publication of improper data or information, publication of inaccurate data, misinterpretation of the data, absence of data consumers, subjects less willing to cooperate, overlapping of data and increased number of requests for data. Kalampokis et al. [12] in their paper from 2013 claim that the real value of OGD will unveil from performing data analytics on top of combined statistical datasets that were previously closed in disparate sources and can now be linked to provide unexpected and unexplored insights. To support this claim, authors described the OGD analytics concept along with its technical requirements, which can be later extended with Apache Hadoop.

The most recent survey about the term big data is well conducted by [7], where authors present the general background of big data and review related technologies, such as distributed approach, Internet of Things, data centres, and Apache Hadoop. They also introduce the terms big data generation and acquisition, big data storage and big data applications including big data analysis, which are closely related to the topic of this paper. The main work by Dean and Ghemawat [8] describes the file system implemented by Google called the Google File System (GFS), which handles the big data operations behind the Google services. Also Vilas's findings lend support to the claim that the high performance computing platforms are required, which impose systematic designs to unleash the full power of the big data [23]. Along similar lines, Lin et al. [17] develop the idea that processing amounts of data requires computational power far beyond the capability of an individual personal computer; it requires a more powerful resource such as a cluster supercomputer.

A comparison of approaches to large-scale data analysis can be found in Pavlo et al. [19] or Chen et al. [7]. The evidence supporting the use of the big data for analytics and the improvement of the decision-making process may lie in the findings of Power [20], who proposes to identify use cases and user examples related to analysing large volumes of semi and unstructured data.

Since Apache Hadoop is an open source project, many optimizations have been applied to improve its performance. The work by Li et al. [16] on optimally tuning MapReduce platforms contributed an analytical model of I/O overheads for MapReduce jobs performing incremental one-pass analytics. Although their model does not predict total execution time, it is useful in identifying three key performance parameters: chunk size (amount of work assigned to each map task); external sort-merge behaviour and number of reducers. Yang et al. [25] then concentrated on the relationships between workload characteristics and corresponding performance under different Apache Hadoop configurations. They selected a suite of benchmarks representing a large range of important applications and derived several configuration metrics that influence the workload performance. They identified critical metrics using principal component analysis, which significantly reduce the complexity of performance modelling. Some of them will be used later in this paper.

Furthermore, there are also researches on identifying design factors of specific application or areas that can improve the performance of Apache Hadoop, e. g. Jiang et al. [11] have conducted an in-depth performance study of MapReduce and as an outcome of this study, they identified some factors that can have significant performance effect on the MapReduce framework. Almeer in [2] considered the trend in time consumption with the increase in the volume of data, and tried to show the difference in run time between a single PC implementation and the parallel Apache Hadoop implementation. More precisely, the author tested the performance of some parallel image filtering algorithms, which ran well when the size of the input image was not large in the comparison of the default value of block size in Apache Hadoop.

Lai et al. [15] introduced how cloud computing can make a breakthrough by proposing a multimedia social network dataset on Apache Hadoop platform and compared and verified the performance efficiency of this platform with the different hardware parameters. The impact of network speed in the cluster computations is discussed in [24]. Other related work of Schätzle et al. [21] investigated the efficiency of Apache Pig and Hadoop for large Resource Description Framework (RDF) datasets. More information about the quality of service attributes and performance metrics for evaluation of cluster and cloud architectures and services can be found in the work of Garg et al. [6].

3 Cloud computing and the distributed data processing

Cloud computing techniques take the form of distributed computing by utilizing multiple computers to execute computing simultaneously on the service side over the Internet. Businesses and citizens no longer require large capital outlays in hardware to deploy their service. Instead they access the hardware and system software provisioned by data centres in a pay-as-you go manner [25]. It is a viable alternative to improve the scalability and high availability of applications. Cloud-based applications typically feature elasticity mechanisms, namely the ability to scale-up or down their resource use depending on user demand [26].

The MapReduce framework fits well this model since it is highly parameterized and can be configured to use as many resources as an administrator deems cost-effective for a particular job [15]. GFS, a scalable and reliable distributed file system for large data sets and BigTable, a scalable and reliable distributed storage system for sparse structured data were the first pioneers [26]. MapReduce parallel programming model and its open-source clone Apache Hadoop, a computing cluster formed by low-priced hardware, have attracted the interest of both industrial and public sector environments in implementing

scalable and fault-tolerant data-intensive applications. The Apache Hadoop framework is aligned with the transparency citizens expect from good government.

A number of storage abstractions and models are being proposed in the context of cloud computing. Microsoft Azure, for example, provides abstractions such as Table, Blob, and Queue. Amazon provides the Simple Storage Service, Elastic Block Storage, and a key/blob store. MapReduce itself depends on the GFS and the corresponding Hadoop implementation uses the Hadoop Distributed File System (HDFS). More detailed information about cloud computing and software services can be found in [1].

Distributed data processing is a method of organizing data processing that uses networked computers in which data processing capabilities are spread across the network. In this kind of processing, specific jobs are performed by specialized computers which may be far removed from the user and/or from other such computers. It provides greater scalability, allows greater flexibility in structure, more autonomy, however, it requires more network administration resources, incompatibility of components, difficulty of controlling information resources and more redundancy. This method is increasing because dramatically reduced hardware costs, improved user interfaces and new frameworks like MapReduce [4], [10].

4 Open big data and the data catalogs

Open data are a piece of content or data if anyone is free to use, reuse, and also redistribute it – subject only, at most, to the requirement to attribute and share-alike. Most of the open data are actually in raw form [9], [13]. Linked data describes a method of publishing structured data so that it can be interlinked and become more useful. It requires a standard mechanism for specifying the existence and meaning of connections between items described in this data using web technologies such as HTTP, RDF and Uniform resource identifiers (URIs) [9]. The concept of big data is usually defined by the volume, velocity and variety. In many areas volumes of available facts are higher than before, they are also expanding quicker than ever, come from many more sources and materialize in many different forms than small, well-structured datasets from the past [3].

Open big data have a great potential for reuse but in order to turn this potential into actual benefits it is necessary for potential users to be able to easily find the data of their interest. Thus, the open big data catalog is a tool that can significantly improve discoverability of the free available datasets. Data catalogs can be divided into the following groups [14]:

- Local – data catalog owned by cities/towns or with only city/town coverage,
- regional – data catalogs owned by a regional authority (county government or federal state government) or with regional coverage,
- national – data catalog owned by a central government institution or with nationwide coverage,
- international – data catalog owned by an international institution or with the international coverage.

Classification of the selected sources of the open big data can be seen in the Tab. 1. It extends the results of Heath and Bizer in [9], where authors divided these sources of data into the categories of geographic data, media data, government data, libraries and education,

life sciences data, retail and commerce, user generated content and social media. As well as the official public and private sector sponsored portals, there are numerous unofficial sources of the open big data, usually compiled by citizens, communities or aggregators. To facilitate interoperability between data catalogs published on the web, the World Wide Web Consortium (W3C) published an RDF vocabulary named Data Catalog Vocabulary (DCAT). By using DCAT to describe datasets in data catalogs, publishers increase discoverability and enable applications easily to consume metadata from multiple catalogs [5].

Tab. 1: Classification of the selected open data catalogs

Category	Example of the selected catalog
data aggregators	http://datacatalogs.org/, http://knoema.com/, Google Public data explorer, Junar, DataMarket etc.
OGD and international governmental organization's data	USA - http://www.data.gov/, UK - http://data.gov.uk/, DE - https://www.govdata.de/ etc. EU - http://publicdata.eu/, UN - http://data.un.org/, WBG - http://datacatalog.worldbank.org/ etc.
OSD	https://www.opensciencedatacloud.org/publicdata/, http://statistics.ucla.edu/
news data	API's of The New York Times, The Guardian Data Blog, iDnes.cz etc.
sports data	http://www.pro-football-reference.com/, http://sportsdatabase.com/, http://developer.espn.com/
social data	The best place to get social data for an API is the site itself: Instagram, Facebook, Twitter, GetGlue, Foursquare, pretty much all social media sites have their own API's.
weather data	http://www.wunderground.com/, http://www.weatherbase.com/, http://openweathermap.org/
spatial data	http://www.openstreetmap.org/, http://www.iscgm.org/, http://www.geonames.org/, http://gcmd.nasa.gov/, https://www.sharegeo.ac.uk/
digitized data from libraries and e-books	http://arxiv.org/, http://www.lib.powerdata.ir/, https://www.bookshare.org/, https://openlibrary.org/, http://www.widernet.org/egranary/ etc.

Source: Authors

5 Case study

For this case study was used Apache Hadoop in the fully-distributed mode created on the virtual machine as the Virtual Hadoop cluster, because it is easier to deploy for the testing purposes of the performance efficiency. Authors could use a pseudo-distributed mode, however, this feature is rather useful for the basic development and testing (writing some code (script) that uses the services and check if it runs correctly). VirtualBox 4.3.8 was used to setup the Virtual Hadoop cluster. Ubuntu Server 12.04 was the main operating system of the cluster's members. Since the main machine had a 6-core

processor, the virtual cluster with the maximum of 4-nodes was created. The hardware and software used can be seen from the Tab. 2.

Firstly a virtual machine in VirtualBox had to be created and configured with the required hardware parameters and settings to act as a cluster node (specially the network settings). This virtual machine was then cloned as many times as there will be nodes in the Virtual Hadoop cluster. Only a limited set of changes were needed to finalize the node to be operational (the hostname and IP address had to be defined).

Tab. 2: The default configuration of the main machine

Hardware	Processor	AMD FX-6300 VISHERA
	Number of Cores	6
	Threads per Core	1
	Memory Capacity	8 GB
	Disk Capacity	1 TB
	Network	100 Mbps
Software	Virtual Machine	VirtualBox 4.3.8
	Operating System	Ubuntu Server 12.04
	Java Virtual Machine	Java 7
	Hadoop Release	Hadoop 2.2.0

Source: Authors

The input data file had 750 MB in total and it was the text data file in csv format. More open big data for the testing purposes can be found in the data catalogs in Tab 1. Open big data can be also distinguished into categories such as transport, education, environment, public finances, geospatial etc. and selected data mining or text mining methods and statistical analyses are performed on these data. The size of these data files is then typically in tens to hundreds of GBs, which is more suitable to choose the Apache Hadoop platform for the open big data processing using commodity PCs. This will be presented in the authors' future papers. However, for this case study is the file size of 750 MB sufficient.

The input data file was then divided into the data files of 50, 100, 200 and 500 MB. Every test for the required configuration was repeated 15 times. Tab. 3 presents the comparison of the performance efficiency (as processing times) of the WordCount algorithm using one to four computers in the Virtual Hadoop cluster with different levels of computer hardware specifications. As the table indicates, while processing 200 a 500 MB files, the cluster made up of three and four nodes is almost twice faster compared to the single or two nodes cluster. While large files are processed, the cluster made up of multiple nodes fulfils its utility with the co-operation of these nodes. The factor which has the greatest influence on the performance efficiency is the memory capacity. On the other hand, the disk capacity has an inconclusive effect on the processing times as well as the network.

Tab. 3: The hardware performance efficiency of the Virtual Hadoop cluster – the analysis of 50,100, 200, 500 MB files 1/2

The hardware configuration of a single member in the virtual cluster	**Number of the nodes in the virtual cluster** (one is always a Master, the other ones are Slaves)															
	1				**2**				**3**				**4**			
Memory Capacity (512 MB),	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB
Disk Capacity (100 GB), Network (100 Mbps).	41,5 s	56,7 s	73,6 s	125,2 s	32,8 s	43,2 s	56,6 s	94,1 s	26,4 s	34,3 s	44,5 s	71,7 s	20,2 s	25,1 s	41,2 s	66,7 s
Memory Capacity (256 MB),	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB
Disk Capacity (100 GB), Network (100 Mbps).	56,1 s	77,7 s	103,2 s	200,3 s	42,3 s	57,1 s	77,2 s	143,1 s	32,5 s	43,6 s	58,7 s	103,2 s	24,2 s	31,4 s	53,6 s	93,4 s
Memory Capacity (128 MB),	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB
Disk Capacity (100 GB), Network (100 Mbps).	58,5 s	82,2 s	109,7 s	212,7 s	44,3 s	60,4 s	81,7 s	151,4 s	33,8 s	46,3 s	62,6 s	109,8 s	25,3 s	33 s	56,9 s	99,3 s
Memory Capacity (512 MB),	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB
Disk Capacity (100 GB), Network (10 Mbps).	51 s	70,3 s	88,3 s	149 s	40,7 s	54,4 s	68,5 s	112 s	33,3 s	43,6 s	55,2 s	88,2 s	25,9 s	32,1 s	53,1 s	86 s
Memory Capacity (256 MB),	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB
Disk Capacity (100 GB), Network (10 Mbps).	68,9 s	96,4 s	123,5 s	238,1 s	52,5 s	71,9 s	93,1 s	170,2 s	40,9 s	55,3 s	72,8 s	126 s	30,9 s	40,4 s	69,1 s	120,5 s
Memory Capacity (128 MB),	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB
Disk Capacity (100GB), Network (10 Mbps).	72,1 s	101,9 s	131,6 s	253,1 s	54,5 s	75,7 s	98,6 s	180,2 s	42,6 s	58,4 s	77,3 s	134,8 s	32,3 s	42,4 s	73,3 s	128,1 s
Memory Capacity (512 MB),	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB
Disk Capacity (10 GB), Network (100 Mbps).	50,7 s	65,3 s	78,2 s	129,4 s	39,7 s	49,2 s	59,4 s	96 s	31,7 s	39,3 s	46,8 s	73,5 s	24,2 s	27,1 s	43,3 s	68 s
Memory Capacity (256 MB),	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB
Disk Capacity (10 GB), Network (100 Mbps).	66,5 s	88,7 s	109 s	207,5 s	49,3 s	63,1 s	80,8 s	145,7 s	37,1 s	47,7 s	60,1 s	105 s	27,3 s	32,9 s	55,1 s	94,9 s
Memory Capacity (128 MB),	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB
Disk Capacity (10 GB), Network (100 Mbps).	67,9 s	89,5 s	111 s	209 s	44,3 s	65,6 s	83,1 s	152,8 s	39,8 s	50,1 s	64,5 s	111,9 s	29,1 s	36,5 s	59,1 s	100,8 s

Source: Authors

Tab. 3: The hardware performance efficiency of the Virtual Hadoop cluster – the analysis of 50,100, 200, 500 MB files 2/2

The hardware configuration of a single member in the virtual cluster	**Number of the nodes in the virtual cluster** (one is always a Master, the other ones are Slaves)															
	1				**2**				**3**				**4**			
Memory Capacity (512 MB),	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB
Disk Capacity (10 GB), Network (10 Mbps).	57,7 s	78,7 s	95,4 s	157, 9 s	45,1 s	59,9 s	74,6 s	120, 9 s	36,9 s	48,4 s	59 s	92,6 s	28,4 s	35 s	57,4 s	90,3 s
Memory Capacity (256 MB),	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB
Disk Capacity (10 GB), Network (10 Mbps).	77,9 s	107, 9 s	133, 5 s	252, 7 s	58,2 s	79 s	101, 5 s	183, 8 s	45,4 s	61,4 s	77,9 s	133, 3 s	34,1 s	43,8 s	74,6 s	126, 5 s
Memory Capacity (128 MB),	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB	50 MB	100 MB	200 MB	500 MB
Disk Capacity (10 GB), Network (10 Mbps).	81,3 s	114, 2 s	142, 1 s	268, 3 s	60,5 s	83,2 s	107, 5 s	194, 6 s	47,3 s	64,8 s	82,7 s	141, 6 s	35,6 s	46,2 s	79,2 s	134, 5 s

Source: Authors

The most efficient hardware configuration in the meaning of the processing time was chosen from the Tab. 3 and the different Apache Hadoop configuration metrics were set for the following models as can be seen in the Tab. 4. The default value of the concrete metric is bold and these values have already been measured in the Tab. 3. To simplify this goal, the 200 MB size file was selected as an example. The corresponding configuration metrics of Apache Hadoop are located in hdfs-site.xml (dfs.blocksize, dfs.replication) and mapred-site.xml (maximum.map.tasks, maximum.reduce.tasks). More information about these metrics and how they affect the performance efficiency can be found in [10] or [25].

Tab. 4: The software performance efficiency of the Virtual Hadoop cluster – the analysis of 200 MB file

The software configuration of Apache Hadoop	**The hardware configuration of a single member in the virtual cluster**											
The range for Apache Hadoop configuration metrics (default value is bold)	Disk Capacity 100 GB											
	Network 100 Mbps											
	Memory 128 MB				Memory 256 MB				Memory 512 MB			
dfs.blocksize	1 PC	2 PCs	3 PCs	4 PCs	1 PC	2 PCs	3 PCs	4 PCs	1 PC	2 PCs	3 PCs	4 PCs
32 MB	126,1 s	92,3 s	70,1 s	62,6 s	127,9 s	94,1 s	71 s	63,3 s	92 s	69,6 s	53,4 s	48,9 s
64 MB	109,7 s	81,7 s	62,6 s	56,9 s	103,2 s	77,2 s	58,7 s	53,6 s	73,6 s	56,6 s	44,5 s	41,2 s
128 MB	105,1 s	79,1 s	60,5 s	55,3 s	100,5 s	74,3 s	55,2 s	49,1 s	69,3 s	52,5 s	41,1 s	37,7 s
dfs.replication	1 PC	2 PCs	3 PCs	4 PCs	1 PC	2 PCs	3 PCs	4 PCs	1 PC	2 PCs	3 PCs	4 PCs
1	97,6 s	72,7 s	53,2 s	46,7 s	87,8 s	65,6 s	48,5 s	44,4 s	61,8 s	47,5 s	36,9 s	34,2 s
2	105,2 s	78,6 s	59,8 s	53,9 s	94 s	70,3 s	52,8 s	48,2 s	67,5 s	51 s	39,6 s	36,5 s
3	109,7 s	81,7 s	62,6 s	56,9 s	103,2 s	77,2 s	58,7 s	53,6 s	73,6 s	56,6 s	44,5 s	41,2 s
maximum.map.tasks	1 PC	2 PCs	3 PCs	4 PCs	1 PC	2 PCs	3 PCs	4 PCs	1 PC	2 PCs	3 PCs	4 PCs
2	109,7 s	81,7 s	62,6 s	56,9 s	103,2 s	77,2 s	58,7 s	53,6 s	73,6 s	56,6 s	44,5 s	41,2 s
3	106,2 s	82 s	60,1 s	54 s	95,8 s	77 s	55,4 s	50,1 s	67,7 s	55,2 s	41,8 s	38,9 s
4	99,8 s	82,3 s	60,9 s	52,8 s	93 s	75,9 s	56 s	48,4 s	66,9 s	55 s	42 s	37,5 s
maximum.reduce.tasks	1 PC	2 PCs	3 PCs	4 PCs	1 PC	2 PCs	3 PCs	4 PCs	1 PC	2 PCs	3 PCs	4 PCs
2	109,7 s	81,7 s	62,6 s	56,9 s	103,2 s	77,2 s	58,7 s	53,6 s	73,6 s	56,6 s	44,5 s	41,2 s
3	104,2 s	82 s	61,1 s	55,7 s	98 s	79,1 s	55,9 s	51,6 s	70,2 s	57 s	42,9 s	39,8 s
4	115,3 s	88,3 s	62,4 s	58,5 s	101,5 s	81 s	56,8 s	50,5 s	70 s	58,2 s	41,8 s	38,6 s

Source: Authors

6 Results and discussion

The main findings show, that the processing with the block size of 32 MB is about 15-20% slower than the default value. However, the processing with the block size of 128 MB is only 3-7% faster than the default value. With the memory capacity of 512 MB it is about 10% faster. These results also provide confirmatory evidence that the memory capacity has the greatest influence on the performance efficiency. The default setting for the replication factor means, that 3 copies of all data would be distributed around the file system. However, this level of redundancy is necessary only in the case to prevent loss of data in the event of failures. In this case study, the processing only with one copy of the open big data is about 20% faster than 3 copies of all data. The effect of the number of map and reduce tasks in the Virtual Hadoop cluster is more or less inconclusive. Nevertheless, based on the literature review, it is closely related to the number of processor's cores and the number of nodes (PCs) in the cluster.

This case study demonstrates that Apache Hadoop platform has a potential for the processing of the open big data and solving complex analytics problems. It helps programmers to concentrate on the essence of their problems. The practical value for further innovations in global society for better cooperation and continually increasing development lies in the open big data analytics. On the other hand, Apache Hadoop requires to be tuned for optimal performance according to the problem and the data available. Therefore, it is important to focus on the Apache Hadoop's programming model and the use of the appropriate algorithms (scripts), which can be used with these types of the open big data. Some of them can be already found e. g. in [10]. However, the performance efficiency of the Virtual Hadoop cluster may also vary from the previous requests. This can be due to processor differences, or the other workloads.

Conclusion and future work

The availability of the open big data enabled by the recent hardware and software advances and complemented by the shift towards more openness of the public sector provides yet another example of the ICT revolution persistence. In this paper, authors used the sources of the Virtual Hadoop cluster to simplify the description of the performance model of a small cluster, which can be used for the open big data analytics in the public sector. The next step will be the deployment of these findings in the fully-distributed mode with the commodity PC hardware. The other option of the future research should be the MapReduce implementation on top of a cloud operating system. Liu and Orban in [18] studied this issue and showed that their implementation of MapReduce in cloud run faster the Apache Hadoop. Also the use of more complex algorithms already implemented on this platform such Apache Mahout may be a way.

Acknowledgement

This contribution was supported by SGSFES_2014003 fund.

References

[1] AHSON, S. A.; ILYAS, M. *Cloud Computing and Software Services: Theory and Techniques*. Boca Raton, FL: CRC Press, 2011. 442 p. ISBN 978-143-9803-158.

[2] ALMEER, M. H. Hadoop MapReduce for Remote Sensing Image Analysis. *International Journal of Emerging Technology and Advanced Engineering*, 2012, vol. 2, no. 4, pp. 443-451.

[3] BUCHHOLTZ, S.; BUKOWSKI, M.; ŚNIEGOCKI A. *Big and open data in Europe: A growth engine or a missed opportunity*. Varšava: demosEUROPA, 2014. 113 p. ISBN 978-83-925542-1-9.

[4] BUYYA, R.; BROBERG, J.; GOŚCIŃSKI, A. *Cloud computing: Principles and Paradigms*. Hoboken, N.J.: Wiley, 2011. 637 p. ISBN 978-0-470-88799-8.

[5] Data Catalog Vocabulary (DCAT). *World Wide Web Consortium (W3C)* [online]. 2014 [cit. 2014-05-15]. Available at WWW: http://www.w3.org/TR/vocab-dcat/

[6] GARG, S. K.; VERSTEEG, S.; BUYYA, R. A framework for ranking of cloud computing services. *Future Generation Computer Systems*, 2013, vol. 29, no. 4, pp. 1012-1023.

[7] CHEN, M.; MAO, S.; LIU, Y. Big Data: A Survey. *Mobile Networks and Applications*, 2014, vol. 19, no. 2, pp. 171-209.

[8] DEAN, J.; GHEMAWAT, S. MapReduce: simplified data processing on large clusters. *Communications of the ACM*, 2008, vol. 5, no. 1, pp. 107-113.

[9] HEATH, T.; BIZER, C. *Linked Data: Evolving the Web into a Global Data Space*. Morgan & Claypool, 2011. 122 p. ISBN 978-160-8454-310.

[10] HOLMES, A. *Hadoop in Practice*. Shelter Island, NY: Manning, 2012. 511 p. ISBN 978-161-7290-237

[11] JIANG, D. et al. The performance of MapReduce: An in-depth Study. *Proceedings of the VLDB Endowment*, 2010, vol. 3, no. 1-2, pp. 472-483.

[12] KALAMPOKIS, E.; TAMBOURIS, E.; TARABANIS, K. Linked Open Government Data Analytics. *Electronic Government*. Springer Berlin Heidelberg, 2013, pp. 99-110.

[13] KUCERA, J.; CHLAPEK, D. Benefits and Risks of Open Government Data. *Journal of Systems Integration*, 2014, vol. 5, no. 1, pp. 30-41.

[14] KUČERA, J.; CHLAPEK, D.; NEČASKÝ, M. Open Government Data Catalogs: Current Approaches and Quality Perspective. *Technology-Enabled Innovation for Democracy, Government and Governance*. Prague, 26. 08. 2013 – 28. 08. 2013. Berlin: Springer Verlag, 2013, pp. 152–166.

[15] LAI, W. K. et al. Towards a framework for large-scale multimedia data storage and processing on Hadoop platform. *The Journal of Supercomputing*, 2014, vol. 68, pp. 488-507.

[16] LI, B. et al. A platform for scalable one-pass analytics using MapReduce. *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*. ACM, 2011. pp. 985-996.

[17] LIN, H.; MA, X.; FENG, W. Reliable MapReduce computing on opportunistic resources. *Cluster Computing*, 2012, vol. 15, no. 2, pp. 145-161.

[18] LIU, H.; ORBAN, D. Cloud MapReduce: A MapReduce Implementation on Top of a Cloud Operating System. *11th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGrid)*, 2011, pp. 464-474.

[19] PAVLO, A. et al. A Comparison of Approaches to Large-Scale Data Analysis. *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, 2009, pp. 165-178.

[20] POWER, D. Using 'Big Data' for analytics and decision support. *Journal of Decision Systems*, 2014, vol. 23, no. 2, pp. 222-228.

[21] SCHÄTZLE, A. et al. PigSPARQL: Übersetzung von SPARQL nach Pig Latin. *BTW*, 2011, pp. 65-84.

[22] TIEN, J. M. Big data: Unleashing information. *Journal of Systems Science and Systems Engineering*, 2013, vol. 22, no. 2, pp. 127-151.

[23] VILAS, K. S. Big Data Mining. *International Journal of Computer Science and Management Research*, 2012, vol. 1, no. 1, pp. 12-17.

[24] WRZUSZCZAK-NOGA, J.; BORZEMSKI, L. Comparison of MPI Benchmarks for Different Ethernet Connection Bandwidths in a Computer Cluster. *Computer Networks*. Springer Berlin Heidelberg, 2010, pp. 342-348.

[25] YANG, H. et al. MapReduce Workload Modeling with Statistical Approach. *Journal of Grid Computing*, 2012, vol. 10, no. 2, pp. 279-310.

[26] ZHANG, C. et al. Case Study of Scientific Data Processing on a Cloud Using Hadoop. *High performance computing systems and applications*. Springer Berlin Heidelberg, 2010, pp. 400-415.

Contact Address

Ing. et Ing. Martin Lněnička
doc. Ing. Jitka Komárková, Ph.D.
University of Pardubice, Faculty of Economics and Administration
Studentská 84, 53210 Pardubice, Czech Republic
Email: martin.lnenicka@gmail.com, jitka.komarkova@upce.cz
Phone number: +420 46603 6075, +420 46603 6070

Received: 19. 06. 2014
Reviewed: 25. 09. 2014, 20. 10. 2014
Approved for publication: 08. 04. 2015