Information Systems 47 (2015) 98–115

Contents lists available at ScienceDirect

Information Systems

journal homepage: www.elsevier.com/locate/infosys

The rise of “big data” on cloud computing: Review and open research issues

Ibrahim Abaker Targio Hashem [a,*], Ibrar Yaqoob [a], Nor Badrul Anuar [a], Salimah Mokhtar [a], Abdullah Gani [a], Samee Ullah Khan [b]

[a] *Faculty of Computer Science and information Technology, University of Malaya, 50603 Kuala Lumpur, Malaysia*
[b] *NDSU-CIIT Green Computing and Communications Laboratory, North Dakota State University, Fargo, ND 58108, USA*

ARTICLE INFO

Article history:
Received 11 June 2014
Received in revised form
22 July 2014
Accepted 24 July 2014
Recommended by: Prof. D. Shasha
Available online 10 August 2014

Keywords:
Big data
Cloud computing
Hadoop

ABSTRACT

Cloud computing is a powerful technology to perform massive-scale and complex computing. It eliminates the need to maintain expensive computing hardware, dedicated space, and software. Massive growth in the scale of data or big data generated through cloud computing has been observed. Addressing big data is a challenging and time-demanding task that requires a large computational infrastructure to ensure successful data processing and analysis. The rise of big data in cloud computing is reviewed in this study. The definition, characteristics, and classification of big data along with some discussions on cloud computing are introduced. The relationship between big data and cloud computing, big data storage systems, and Hadoop technology are also discussed. Furthermore, research challenges are investigated, with focus on scalability, availability, data integrity, data transformation, data quality, data heterogeneity, privacy, legal and regulatory issues, and governance. Lastly, open research issues that require substantial research efforts are summarized.

Contents

[*] Corresponding author. Tel.: +60 173946811.
E-mail addresses: targio@siswa.um.edu.my (I.A.T. Hashem), ibraryaqoob@siswa.um.edu.my (I. Yaqoob), badrul@um.edu.my (N.B. Anuar), salimah@um.edu.my (S. Mokhtar), abdullah@um.edu.my (A. Gani), samee.khan@ndsu.edu (S. Ullah Khan).

http://dx.doi.org/10.1016/j.is.2014.07.006
0306-4379/© 2014 Elsevier Ltd. All rights reserved.

1. Introduction

The continuous increase in the volume and detail of data captured by organizations, such as the rise of social media, Internet of Things (IoT), and multimedia, has produced an overwhelming flow of data in either structured or unstructured format. Data creation is occurring at a record rate [1], referred to herein as big data, and has emerged as a widely recognized trend. Big data is eliciting attention from the academia, government, and industry. Big data are characterized by three aspects: (a) data are numerous, (b) data cannot be categorized into regular relational databases, and (c) data are generated, captured, and processed rapidly. Moreover, big data is transforming healthcare, science, engineering, finance, business, and eventually, the society. The advancements in data storage and mining technologies allow for the preservation of increasing amounts of data described by a change in the nature of data held by organizations [2]. The rate at which new data are being generated is staggering [3]. A major challenge for researchers and practitioners is that this growth rate exceeds their ability to design appropriate cloud computing platforms for data analysis and update intensive workloads.

Cloud computing is one of the most significant shifts in modern ICT and service for enterprise applications and has become a powerful architecture to perform large-scale and complex computing. The advantages of cloud computing include virtualized resources, parallel processing, security, and data service integration with scalable data storage. Cloud computing can not only minimize the cost and restriction for automation and computerization by individuals and enterprises but can also provide reduced infrastructure maintenance cost, efficient management, and user access [4]. As a result of the said advantages, a number of applications that leverage various cloud platforms have been developed and resulted in a tremendous increase in the scale of data generated and consumed by such applications. Some of the first adopters of big data in cloud computing are users that deployed Hadoop clusters in highly scalable and elastic computing environments provided by vendors, such as IBM, Microsoft Azure, and Amazon AWS [5]. Virtualization is one of the base technologies applicable to the implementation of cloud computing. The basis for many platform attributes required to access, store, analyze, and manage distributed computing components in a big data environment is achieved through virtualization.

Virtualization is a process of resource sharing and isolation of underlying hardware to increase computer resource utilization, efficiency, and scalability.

The goal of this study is to implement a comprehensive investigation of the status of big data in cloud computing environments and provide the definition, characteristics, and classification of big data along with some discussions on cloud computing. The relationship between big data and cloud computing, big data storage systems, and Hadoop technology are discussed. Furthermore, research challenges are discussed, with focus on scalability, availability, data integrity, data transformation, data quality, data heterogeneity, privacy, legal and regulatory issues, and governance. Several open research issues that require substantial research efforts are likewise summarized.

The rest of this paper is organized as follows. Section 2 presents the definition, characteristics, and classification of big data. Section 3 provides an overview of cloud computing. The relationship between cloud computing and big data is presented in Section 4. Section 5 presents the storage systems of big data. Section 6 presents the Hadoop background and MapReduce. Several issues, research challenges, and studies that have been conducted in the domain of big data are reviewed in Section 7. Section 8 provides a summary of current open research issues and presents the conclusions. Table 1 shows the list of abbreviations used in the paper.

2. Definition and characteristics of big data

Big data is a term utilized to refer to the increase in the volume of data that are difficult to store, process, and analyze

Table 1
List of abbreviations.

Abbreviations	Full meaning
ACID	Atomicity, Consistency, Isolation, Durability
ASF	Apache Software Foundation
DAS	Direct Attached Storage
Doc	Document
DSMS	Data Stream Management System
EC2	Amazon Elastic Compute Cloud
GFS	Google File System
HDDs	Hard Disk Drives
HDFS	Hadoop Distributed File System
IaaS	Infrastructure as a Service
ICT	Information Communication Technology
IoT	Internet of Things
IT	Information Technology
JSON	JavaScript Object Notation
KV	Key Value
NAS	Network Attached Storage
NoSQL	Not Only SQL
OLM	Online Lazy Migration
PaaS	Platform as a Service
PDF	Portable Document Format
RDBMS	Relational Database Management System
SAN	Storage Area Network
SQL	Structured Query Language
SDLM	Scientific Data Lifecycle Management
S3	Simple Storage Service
SaaS	Software as a Service
URL	Uniform Resource Locator
XML	Extensible Markup Language

through traditional database technologies. The nature of big data is indistinct and involves considerable processes to identify and translate the data into new insights. The term “big data” is relatively new in IT and business. However, several researchers and practitioners have utilized the term in previous literature. For instance, [6] referred to big data as a large volume of scientific data for visualization. Several definitions of big data currently exist. For instance, [7] defined big data as “the amount of data just beyond technology's capability to store, manage, and process efficiently.” Meanwhile, [8] and [9] defined big data as characterized by three Vs: volume, variety, and velocity. The terms volume, variety, and velocity were originally introduced by Gartner to describe the elements of big data challenges. IDC also defined big data technologies as “a new generation of technologies and architectures, designed to economically extract value from very large volumes of a wide variety of data, by enabling the high velocity capture, discovery, and/or analysis.” [10] specified that big data is not only characterized by the three Vs mentioned above but may also extend to four Vs, namely, volume, variety, velocity, and value (Fig. 1, Fig. 2). This 4V definition is widely recognized because it highlights the meaning and necessity of big data.

The following definition is proposed based on the above-mentioned definitions and our observation and analysis of the essence of big data. *Big data is a set of techniques and technologies that require new forms of integration to uncover large hidden values from large datasets that are diverse, complex, and of a massive scale.*

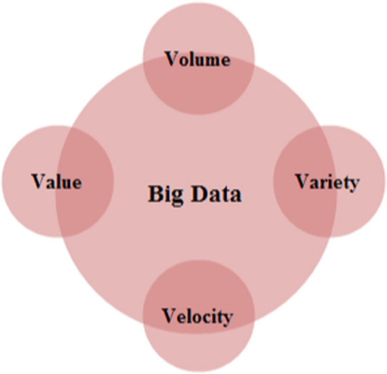

Fig. 1. Four Vs of big data.

(1) ***Volume*** refers to the amount of all types of data generated from different sources and continue to expand. The benefit of gathering large amounts of data includes the creation of hidden information and patterns through data analysis. Laurila et al. [11] provided a unique collection of longitudinal data from smart mobile devices and made this collection available to the research community. The aforesaid initiative is called mobile data challenge motivated by Nokia [11]. Collecting longitudinal data requires considerable effort and underlying investments. Nevertheless, such mobile data challenge produced an interesting result similar to that in the examination of the predictability of human behavior patterns or means to share data based on human mobility and visualization techniques for complex data.

(2) ***Variety*** refers to the different types of data collected via sensors, smartphones, or social networks. Such data types include video, image, text, audio, and data logs, in either structured or unstructured format. Most of the data generated from mobile applications are in unstructured format. For example, text messages, online games, blogs, and social media generate different types of unstructured data through mobile devices and sensors. Internet users also generate an extremely diverse set of structured and unstructured data [12].

(3) ***Velocity*** refers to the speed of data transfer. The contents of data constantly change because of the absorption of complementary data collections, introduction of previously archived data or legacy collections, and streamed data arriving from multiple sources [9].

(4) ***Value*** is the most important aspect of big data; it refers to the process of discovering huge hidden values from large datasets with various types and rapid generation [13].

2.1. Classification of big data

Big data are classified into different categories to better understand their characteristics. Fig. 2 shows the numerous categories of big data. The classification is important because of large-scale data in the cloud. The classification is based on five aspects: (i) data sources, (ii) content format, (iii) data stores, (iv) data staging, and (v) data processing.

Each of these categories has its own characteristics and complexities as described in Table 2. Data sources include internet data, sensing and all stores of transnational information, ranges from unstructured to highly

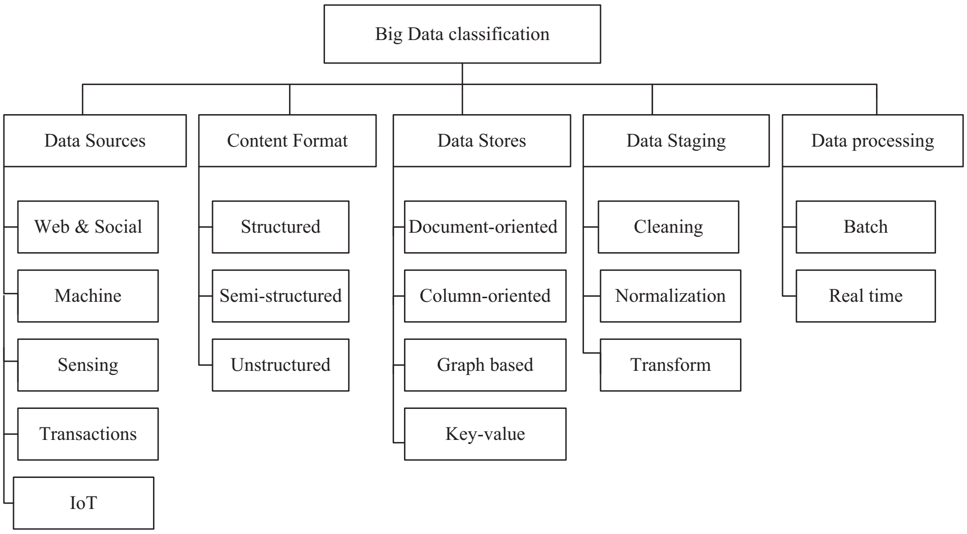

Fig. 2. Big data classification.

structured are stored in various formats. Most popular is the relational database that come in a large number of varieties [29]. As the result of the wide variety of data srouces, the captured data differ in zise with respect to redundancy, consisteny and noise, etc.

3. Cloud computing

Cloud computing is a fast-growing technology that has established itself in the next generation of IT industry and business. Cloud computing promises reliable software, hardware, and IaaS delivered over the Internet and remote data centers [30]. Cloud services have become a powerful architecture to perform complex large-scale computing tasks and span a range of IT functions from storage and computation to database and application services. The need to store, process, and analyze large amounts of datasets has driven many organizations and individuals to adopt cloud computing [31]. A large number of scientific applications for extensive experiments are currently deployed in the cloud and may continue to increase because of the lack of available computing facilities in local servers, reduced capital costs, and increasing volume of data produced and consumed by the experiments [32]. In addition, cloud service providers have begun to integrate frameworks for parallel data processing in their services to help users access cloud resources and deploy their programs [33].

Cloud computing "is a model for allowing ubiquitous, convenient, and on-demand network access to a number of configured computing resources (e.g., networks, server, storage, application, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction" [34]. Cloud computing has a number of favorable aspects to address the rapid growth of economies and technological barriers. Cloud computing provides total cost of ownership and allows organizations to focus on the core business without worrying about issues, such as infrastructure, flexibility, and availability of resources [35]. Moreover, combining the cloud computing utility model and a rich set of computations, infrastructures, and storage cloud services offers a highly attractive environment where scientists can perform their experiments [36]. Cloud service models typically consist of PaaS, SaaS, and IaaS.

- PaaS, such as Google's Apps Engine, Salesforce.com, Force platform, and Microsoft Azure, refers to different resources operating on a cloud to provide platform computing for end users.
- SaaS, such as Google Docs, Gmail, Salesforce.com, and Online Payroll, refers to applications operating on a remote cloud infrastructure offered by the cloud provider as services that can be accessed through the Internet [37].
- IaaS, such as Flexiscale and Amazon's EC2, refers to hardware equipment operating on a cloud provided by service providers and used by end users upon demand.

The increasing popularity of wireless networks and mobile devices has taken cloud computing to new heights because of the limited processing capability, storage capacity, and battery lifetime of each device [126]. This condition has led to the emergence of a mobile cloud computing paradigm. Mobile cloud facilities allow users to outsource tasks to external service providers. For example, data can be processed and stored outside of a mobile device [38]. Mobile cloud applications, such as Gmail, iCloud, and Dropbox, have become prevalent recently. Juniper research predicts that cloud-based mobile applications will increase to approximately 9.5$ billion by 2014 [39]. Such applications improve mobile cloud performance and user experience. However, the limitations associated with wireless networks and the intrinsic nature of mobile devices have imposed computational and data storage restrictions [40,127].

Table 2
Various categories of big data.

Classification	Description
Data sources	
Social media	Social media is the source of information generated via URL to share or exchange information and ideas in virtual communities and networks, such as collaborative projects, blogs and microblogs, Facebook, and Twitter.
Machine-generated data	Machine data are information automatically generated from a hardware or software, such as computers, medical devices, or other machines, without human intervention.
Sensing	Several sensing devices exist to measure physical quantities and change them into signals.
Transactions	Transaction data, such as financial and work data, comprise an event that involves a time dimension to describe the data.
IoT	IoT represents a set of objects that are uniquely identifiable as a part of the Internet. These objects include smartphones, digital cameras, and tablets. When these devices connect with one another over the Internet, they enable more smart processes and services that support basic, economic, environmental, and health needs. A large number of devices connected to the Internet provides many types of services and produces huge amounts of data and information [14].
Content format	
Structured	Structured data are often managed SQL, a programming language created for managing and querying data in RDBMS. Structured data are easy to input, query, store, and analyze. Examples of structured data include numbers, words, and dates.
Semi-structured	Semi-structured data are data that do not follow a conventional database system. Semi-structured data may be in the form of structured data that are not organized in relational database models, such as tables. Capturing semi-structured data for analysis is different from capturing a fixed file format. Therefore, capturing semi-structured data requires the use of complex rules that dynamically decide the next process after capturing the data [15].
Unstructured	Unstructured data, such as text messages, location information, videos, and social media data, are data that do not follow a specified format. Considering that the size of this type of data continues to increase through the use of smartphones, the need to analyze and understand such data has become a challenge.
Data stores	
Document-oriented	Document-oriented data stores are mainly designed to store and retrieve collections of documents or information and support complex data forms in several standard formats, such as JSON, XML, and binary forms (e.g., PDF and MS Word). A document-oriented data store is similar to a record or row in a relational database but is more flexible and can retrieve documents based on their contents (e.g., MongoDB, SimpleDB, and CouchDB).
Column-oriented	A column-oriented database stores its content in columns aside from rows, with attribute values belonging to the same column stored contiguously. Column-oriented is different from classical database systems that store entire rows one after the other [16], such as BigTable [17].
Graph database	A graph database, such as Neo4j, is designed to store and represent data that utilize a graph model with nodes, edges, and properties related to one another through relations [18].
Key-value	Key-value is an alternative relational database system that stores and accesses data designed to scale to a very large size [19]. Dynamo [20] is a good example of a highly available key-value storage system; it is used by amazon.com in some of its services. Similarly, [21] proposed a scalable key-value store to support transactional multi-key access using a single key access supported by key-value for use in G-store designs. [22] presented a scalable clustering method to perform a large task in datasets. Other examples of key-value stores are Apache Hbase [23], Apache Cassandra [24], and Voldemort. Hbase uses HDFS, an open-source version of Google's BigTable built on Cassandra. Hbase stores data into tables, rows, and cells. Rows are sorted by row key, and each cell in a table is specified by a row key, a column key, and a version, with the content contained as an un-interpreted array of bytes.
Data staging	
Cleaning	Cleaning is the process of identifying incomplete and unreasonable data [25].
Transform	Transform is the process of transforming data into a form suitable for analysis.
Normalization	Normalization is the method of structuring database schema to minimize redundancy [26].
Data processing	
Batch	MapReduce-based systems have been adopted by many organizations in the past few years for long-running batch jobs [27]. Such system allows for the scaling of applications across large clusters of machines comprising thousands of nodes.
Real time	One of the most famous and powerful real time process-based big data tools is simple scalable streaming system (S4) [28]. S4 is a distributed computing platform that allows programmers to conveniently develop applications for processing continuous unbounded streams of data. S4 is a scalable, partially fault tolerant, general purpose, and pluggable platform.

4. Relationship between cloud computing and big data

Cloud computing and big data are conjoined. Big data provides users the ability to use commodity computing to process distributed queries across multiple datasets and return resultant sets in a timely manner. Cloud computing provides the underlying engine through the use of Hadoop, a class of distributed data-processing platforms. The use of cloud computing in big data is shown in Fig. 3. Large data sources from the cloud and Web are stored in a distributed fault-tolerant database and processed through a programing model for large datasets with a parallel distributed algorithm in a cluster. The main purpose of data visualization, as shown in Fig. 3, is to view analytical results presented visually through different graphs for decision making.

Big data utilizes distributed storage technology based on cloud computing rather than local storage attached to a computer or electronic device. Big data evaluation is driven by fast-growing cloud-based applications developed using

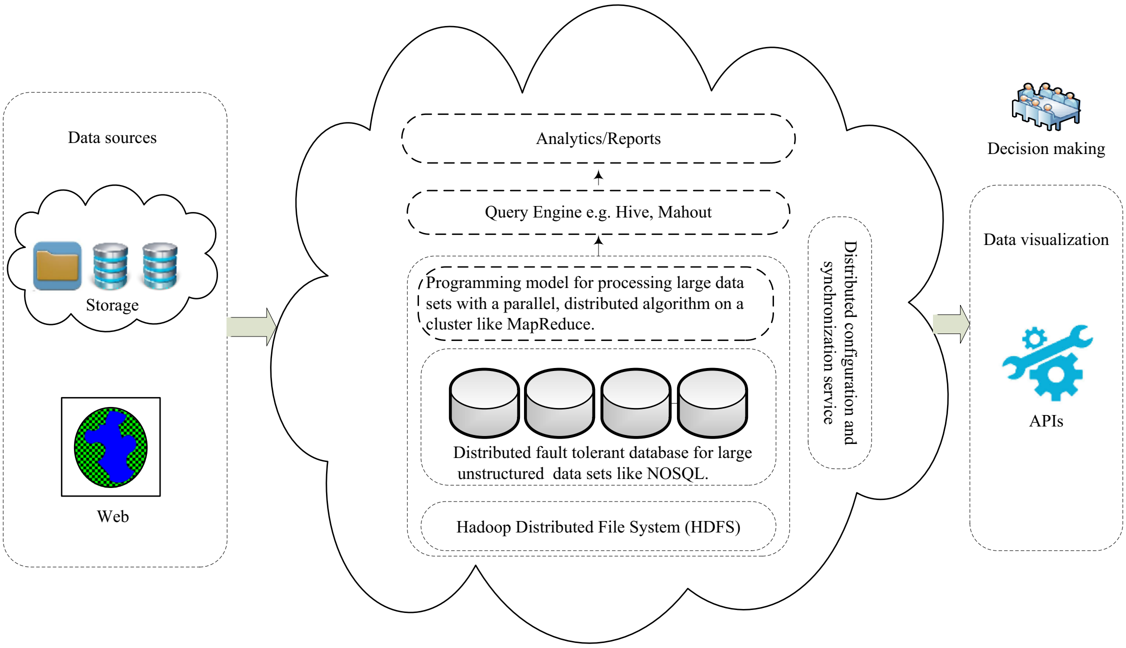
Data sources
Storage
Web
Analytics/Reports
Query Engine e.g. Hive, Mahout
Programming model for processing large data sets with a parallel, distributed algorithm on a cluster like MapReduce.
Distributed fault tolerant database for large unstructured data sets like NOSQL.
Hadoop Distributed File System (HDFS)
Distributed configuration and synchronization service
Decision making
Data visualization
APIs

Fig. 3. Cloud computing usage in big data.

Table 3
Comparison of several big data cloud platforms.

	Google	Microsoft	Amazon	Cloudera
Big data storage	Google cloud services	Azure	S3	
MapReduce	AppEngine	Hadoop on Azure	Elastic MapReduce (Hadoop)	MapReduce YARN
Big data analytics	BigQuery	Hadoop on Azure	Elastic MapReduce (Hadoop)	Elastic MapReduce (Hadoop)
Relational database	Cloud SQL	SQL Azure	MySQL or Oracle	MySQL, Oracle, PostgreSQL
NoSQL database	AppEngine Datastore	Table storage	DynamoDB	Apache Accumulo
Streaming processing	Search API	Streaminsight	Nothing prepackaged	Apache Spark
Machine learning	Prediction API	Hadoop+Mahout	Hadoop+Mahout	Hadoop+Oryx
Data import	Network	Network	Network	Network
Data sources	A few sample datasets	Windows Azure marketplace	Public Datasets	Public Datasets
Availability	Some services in private beta	Some services in private beta	Public production	Industries

virtualized technologies. Therefore, cloud computing not only provides facilities for the computation and processing of big data but also serves as a service model. Table 3 shows the comparison of several big data cloud providers.

Talia [41] discussed the complexity and variety of data types and processing power to perform analysis on large datasets. The author stated that cloud computing infrastructure can serve as an effective platform to address the data storage required to perform big data analysis. Cloud computing is correlated with a new pattern for the provision of computing infrastructure and big data processing method for all types of resources available in the cloud through data analysis. Several cloud-based technologies have to cope with this new environment because dealing with big data for concurrent processing has become increasingly complicated [42]. MapReduce [43] is a good example of big data processing in a cloud environment; it allows for the processing of large amounts of datasets stored in parallel in the cluster. Cluster computing exhibits good performance in distributed system environments, such as computer power, storage, and network communications. Likewise, Bollier and Firestone [44] emphasized the ability of cluster computing to provide a hospitable context for data growth. However, Miller [45] argued that the lack of data availability is expensive because users offload more decisions to analytical methods; incorrect use of the methods or inherent weaknesses in the methods may produce wrong and costly decisions. DBMSs are considered a part of the current cloud computing architecture and play an important role to ensure the easy transition of applications from old enterprise infrastructures to new cloud infrastructure architectures. The pressure for organizations to quickly adopt and implement technologies, such as cloud computing, to address the challenge of big data storage and processing demands entails unexpected risks and consequences.

Table 4 presents several related studies that deal with big data through the use of cloud computing technology. The table provides a general overview of big data and cloud computing technologies based on the area of study and current challenges, techniques, and technologies that restrict big data and cloud computing.

Table 4
Several related studies that deal with big data through the use of cloud computing technology.

Reference	Title of paper	Objectives
[46]	"Data quality management, data usage experience and acquisition intention of big data analytics"	To propose a model for the acquisition intention of big data analytics
[47]	"Big Data Analytics Framework for Peer-to-Peer Botnet Detection Using Random Forests"	To develop open-source tools, such as Hadoop, to provide a scalable implementation of a quasi-real-time intrusion detection system
[48]	"MERRA Analytic Services: Meeting the Big Data Challenges of Climate4 Science through Cloud-enabled Climate Analytics-as-a-Service"	To address big data challenges in climate science
[49]	"System of Systems and Big Data Analytics – Bridging the Gap"	To demonstrate the construction of a bridge between System of Systems and Data Analytics to develop reliable models
[50]	"Symbioses of Big Data and Cloud Computing: Opportunities & Challenges"	To highlight big data opportunity
[51]	"A Special Issue of Journal of Parallel and Distributed Computing: Scalable Systems for Big Data Management and Analytics"	To address special issues in big data management and analytics
[52]	"Smarter fraud investigations with big data analytics"	To investigate smarter fraud with big data analytics
[53]	Moving Big Data to the Cloud: An Online Cost-Minimizing Approach	To upload data into the cloud from different geographical locations with minimum cost of data migration. Two algorithms (OLM, RFHC) are proposed. These algorithms provide optimization for data aggregation and processing and a route for data.
[54]	"Leveraging the capabilities of service-oriented decision support systems: putting analytics and big data in cloud"	To propose a framework for decision support systems in a cloud
[32]	"Cloud Computing and Scientific Applications — Big Data, Scalable Analytics, and Beyond"	To review some of the papers published in Cloud Computing and Scientific Applications (CCSA2012) event
[41]	"Clouds for Scalable Big Data Analytics"	To discuss the use of cloud for scalable big data analytics
[55]	"Cloud Computing Availability: Multi-clouds for Big Data Service"	To overcome the issue of single cloud
[56]	"Adapting scientific computing problems to clouds using MapReduce"	To review the challenges of reducing the number of iterative algorithms in the MapReduce model
[57]	"p-PIC: Parallel Power Iteration Clustering for Big Data"; Journal of Parallel and Distributed Computing	To explore different parallelization strategies
[58]	"Cloud and heterogeneous computing solutions exist today for the emerging big data problems in biology"	To review cloud and heterogeneous computing solutions existing today for the emerging big data problem in biology

Table 5
Summary of Organization case studies from Vendors.

Case	Business needs	Cloud service models	Big data solution	Assessment	Reference
SwiftKey	Language technology	IaaS	Amazon Elastic MapReduce	Success	[59]
343 Industries	Video game developer	IaaS	Apache Hadoop	Success	[60]
redBus	Online travel agency	IaaS, PaaS	BigQuery	Success	[61]
Nokia	Mobile communications	IaaS	Apache Hadoop, Enterprise Data Warehouse	Success	[62]
Alacer	Big data solution	IaaS	Big data algorithms	Success	[63]

5. Case studies

Our discussion on the relationship between big data and cloud computing is complemented by reported case studies on big data using cloud computing technology. Our discussion of the case studies was divided into two parts. The first part describes a number of reported case studies provided by different vendors who integrate big data technologies into their cloud environment. The second part describes a number of case studies that have been published by scholarly/academic sources.

5.1. Organization case Studies from vendors

Customer case studies from vendors, such as Google, Amazon, and Microsoft, were obtained. These case studies show the use of cloud computing technologies in big data analytics and in managing the increasing volume, variety, and velocity of digital information. We selected this collection of five cases because they demonstrate the extensive variety of research communities that use cloud computing. Table 5 summarizes the case studies of big data implemented by using existing cloud computing platforms.

5.1.1. A. SwiftKey

SwiftKey is a language technology founded in London in 2008. This language technology aids touchscreen typing by providing personalized predictions and corrections. The company collects and analyzes terabytes of data to create language models for many active users. Thus, the company needs a highly scalable, multilayered model system that can keep pace with steadily increasing demand and that has a powerful processing engine for the artificial intelligence technology used in prediction generation. To achieve its goals, the company uses Apatche Hadoop running on Amazon Simple Storage Service and Amazon Elastic Compute Cloud to manage the processing of multiple terabytes of data. By using this new solution, SwiftKey is able to scale services on demand during peak time.

5.1.2. B. 343 Industries

The Halo is science fiction media franchise that has grown into a global entertainment phenomenon. More than 50 million copies of the Halo video games have been sold worldwide. Before launching Halo 4, the developers analyzed data to obtain insights into player preferences and online tournaments. To complete this task, the team used Windows Azure HDInsight Service, which is based on the Apache Hadoop big data framework. The team was able to provide game statistics to tournament operators, which used the data to rank players based on game play, by using HDInsight Service to process and analyze raw data from Windows Azure. The team also used HDInsight Service to update Halo 4 every week and to support daily e-mail campaigns designed to increase player retention. Organizations can also utilize data to make prompt business decisions.

5.1.3. C. redBus

The online travel agency redBus introduced Internet bus ticketing in India in 2006, thus unifying tens of thousands of bus schedules into a single booking operation. The company needed a powerful tool to analyze inventory and booking data across their system of hundreds of bus operators serving more than 10,000 routes. They considered using clusters of Hadoop servers to process the data but decided that the system would take considerable time and resources to maintain. Furthermore, the use of clusters of Hadoop servers would not provide the lightning-fast analysis needed by the company. Thus, redBus implemented GoogleQuery to analyze large datasets by using the Google data processing infrastructure. The insights rapidly gained through BigQuery have made redBus a strong company. By minimizing the time needed for staff members to solve technical problems, BigQuery helps improve customer service and reduce lost sales.

5.1.4. D. Nokia

Nokia is a mobile communications company whose products comes to be an integral part of the people live. Many people around the world use Nokia mobile phones to communicate, capture photos and share experiences. Thus, Nokia gathers and analyzes large amounts of data from mobile phones. However, in order to support its extensive use of big data, Nokia relies on a technology ecosystem that includes a Teradata Enterprise Data Warehouse, numerous Oracle and MySQL data marts, visualization technologies, and Hadoop. Nokia has over 100 terabytes of structured data on Teradata and petabytes of multistructured data on the Hadoop Distributed File System (HDFS). The HDFS data warehouse allows the storage of all semi/multistructured data and offers data processing at the petabyte scale.

5.1.5. E. Alacer

An online retailer was experiencing revenue leakage because of unreliable real-time notifications of service problems within its cloud-based e-commerce platform. Alacer used big data algorithms to create a cloud monitoring system that delivers reactive and proactive notifications. By using cloud computing with Alacer's monitoring platform, the incident response time was reduced from one hour to seconds, thus dramatically improving customer satisfaction and eliminating service level agreement penalties.

5.1.5.1. Case studies from scholarly/academic sources. The following case studies provide recent example of how researchers have used cloud computing technology for their big data projects. Table 6 details the five case report studies which explored the use of cloud for big data.

5.1.5.2. Case study 1: cloud computing in genome informatics. Reid et al. [64] have investigated the growth of next-generation sequencing data in laboratories and hospitals. This growth has shifted the bottleneck in clinical genetics from DNA sequence production to DNA sequence analysis. However, accurate and reproducible genomic results at a scale ranging from individuals to large cohorts should be provided. They developed a Mercury analysis pipeline and deployed it in the Amazon web service cloud via the DNAnexus platform. Thus, they established a powerful combination of a robust and fully validated software pipeline and a scalable computational resource that have been applied to more than 10,000 whole genome and whole exome samples.

5.1.5.3. Case study 2: mining Twitter in the cloud. Noordhuis et al. [65] used cloud computing to analyze of large amounts of data on Twitter. The author applied the PageRank algorithm on the Twitter user base to obtain user rankings. The Amazon cloud infrastructure was used to host all related computations. Computations were conducted in a two-phase process: in the crawling phase, all data were retrieved from Twitter. In the processing phase, the PageRank algorithm was applied to compute the acquired data. During the crawling stage, the author web crawled a graph containing 50 million nodes and 1.8 billion edges, which is approximately two-thirds of the estimated user base of Twitter. Thus, a relatively cheap solution for data acquisition and analysis is implemented by using the Amazon cloud infrastructure.

5.1.5.4. Case study 3: scientific data processing. Zhang et al. [66] developed a Hadoop-based cloud computing application that processes sequences of microscopic images of lives cells by using MATLAB. The project was a collaboration between groups in Genome Quebec/McGill University in Montreal and at the University of Waterloo. The goal was to study the complex molecular interactions that regulate biological systems. The application, which was built on the basis of Hadoop, allows users to submit data processing jobs in the cloud. The authors used a homogeneous cluster to conduct initial system development and proof-of-concept tests. The cluster comprises 21 Sun Fire X4100 servers with

Table 6
Summary of case studies from scholarly/academic sources.

Case	Situation/context	Objective	Approach	Result
1	Massively parallel DNA sequencing generates staggering amounts of data.	To provide accurate and reproducible genomic results at a scale ranging from individuals to large cohorts.	Develop a Mercury analysis pipeline and deploy it in the Amazon web service cloud via the DNAnexus platform.	Established a powerful combination of a robust and fully validated software pipeline and a scalable computational resource that have been applied to more than 10,000 whole genome and whole exome samples.
2	Given that conducting analyses on large social networks such as Twitter requires considerable resources because of the large amounts of data involved, such activities are usually expensive.	To use cloud services as a possible solution for the analysis of large amounts of data.	Use PageRank algorithm on the Twitter user base to obtain user rankings. Use the Amazon cloud infrastructure to host all related computations.	Implemented a relatively cheap solution for data acquisition and analysis by using the Amazon cloud infrastructure.
3	To study the complex molecular interactions that regulate biological systems.	To develop a Hadoop-based cloud computing application that processes sequences of microscopic images of live cells.	Use Hadoop cloud computing framework.	Allows users to submit data processing jobs in the cloud
4	Applications running on cloud computing likely may fail.	Design a failure scenario	Create a series of failure scenarios on a Amazon cloud computing platform	Help to identify failure vulnerabilities in Hadoop applications running in cloud.

two dual-core AMD Opteron 280 CPUs interconnected by gigabit Ethernet.

5.1.5.5. Case study 4: failure scenario as a service (FSaaS) for Hadoop Clusters. Faghri et al. [67] have created a series of failure scenarios on a Amazon cloud computing platform to provide Hadoop service with the means to test their applications against the risk of massive failure. They developed a set failure scenarios for Hadoop clusters with 10 Amazon web service EC2 machines. These types of failures could happen inside Hadoop jobs include CPU intensive, namely I/O-intensive and network-intensive. Thus, running such scenario against Hadoop applications can help to identify failure vulnerabilities in these applications.

6. Big data storage system

The rapid growth of data has restricted the capability of existing storage technologies to store and manage data. Over the past few years, traditional storage systems have been utilized to store data through structured RDBMS [13]. However, almost storage systems have limitations and are inapplicable to the storage and management of big data. A storage architecture that can be accessed in a highly efficient manner while achieving availability and reliability is required to store and manage large datasets. The storage media currently employed in enterprises are discussed and compared in Table 7.

Several storage technologies have been developed to meet the demands of massive data. Existing technologies can be classified as direct attached storage (DAS), network attached storage (NAS), and storage area network (SAN). In DAS, various hard disk drives (HDDs) are directly connected to the servers. Each HDD receives a certain amount of input/output (I/O) resource, which is managed by individual applications. Therefore, DAS is suitable only for servers that are interconnected on a small scale. Given the aforesaid low scalability, storage capacity is increased but expandability and upgradeability are limited significantly. NAS is a storage device that supports a network. NAS is connected directly to a network through a switch or hub via TCP/IP protocols. In NAS, data are transferred as files. Given that the NAS server can indirectly access a storage device through networks, the I/O burden on a NAS server is significantly lighter than that on a DAS server. NAS can orient networks, particularly scalable and bandwidth-intensive networks. Such networks include high-speed networks of optical-fiber connections. The SAN system of data storage is independent with respect to storage on the local area network (LAN). Multipath data switching is conducted among internal nodes to maximize data management and sharing. The organizational systems of data storages (DAS, NAS, and SAN) can be divided into three parts: (i) disc array, where the foundation of a storage system provides the fundamental guarantee, (ii) connection and network subsystems, which connect one or more disc arrays and servers, and (iii) storage management software, which oversees data sharing, storage management, and disaster recovery tasks for multiple servers.

7. Hadoop background

Hadoop [73] is an open-source Apache Software Foundation project written in Java that enables the distributed processing of large datasets across clusters of commodity. Hadoop has two primary components, namely, HDFS and MapReduce programming framework. The most significant feature of Hadoop is that HDFS and MapReduce are closely related to each other; each are co-deployed such that a single cluster is produced [73]. Therefore, the storage system is not physically separated from the processing system.

Table 7
Comparison of storage media.

Storage type	Specific use	Advantages	Limitations	Reference
Hard drives	To store data up to four terabytes	Density, cost per bit storage, and speedy start up that may only take several seconds	Require special cooling and high read latency time; the spinning of the platters can sometimes result in vibration and produce more heat than solid state memory	[68]
Solid-state memory	To store data up to two terabytes	Fast access to data, fast movement of huge quantities of data, start-up time only takes several milliseconds, no vibration, and produces less heat than hard drives	Ten times more expensive than hard drives in terms of per gigabyte capacity	[69]
Object storage	To store data as variable-size objects rather than fixed-size blocks	Scales with ease to find information and has a unique identifier to identify data objects; ensures security because information on physical location cannot be obtained from disk drives; supports indexing access	Complexity in tracking indices.	[70]
Optical storage	To store data at different angles throughout the storage medium	Least expensive removable storage medium	Complex; its ability to produce multiple optical disks in a single unit is yet to be proven	[71]
Cloud storage	To serve as a provisioning and storage model and provide on-demand access to services, such as storage	Useful for small organizations that do not have sufficient storage capacity; cloud storage can store large amounts of data, but its services are billable	Security is the primary challenge because of data outsourcing	[72]

Table 8
Summary of the process of the map/reduce function.

Mapper (key1, value1)→List [(key2, value2)]
Reducer [key2, list (value2)]→List (key3, value3)

HDFS [74] is a distributed file system designed to run on top of the local file systems of the cluster nodes and store extremely large files suitable for streaming data access. HDFS is highly fault tolerant and can scale up from a single server to thousands of machines, each offering local computation and storage. HDFS consists of two types of nodes, namely, a namenode called "master" and several datanodes called "slaves." HDFS can also include secondary namenodes. The namenode manages the hierarchy of file systems and director namespace (i.e., metadata). File systems are presented in a form of namenode that registers attributes, such as access time, modification, permission, and disk space quotas. The file content is split into large blocks, and each block of the file is independently replicated across datanodes for redundancy and to periodically send a report of all existing blocks to the namenode.

MapReduce [43] is a simplified programming model for processing large numbers of datasets pioneered by Google for data-intensive applications. The MapReduce model was developed based on GFS [75] and is adopted through open-source Hadoop implementation, which was popularized by Yahoo. Apart from the MapReduce framework, several other current open-source Apache projects are related to the Hadoop ecosystem, including Hive, Hbase, Mahout, Pig, Zookeeper, Spark, and Avro. Twister [76] provides support for efficient and iterative MapReduce computations. An overview of current MapReduce projects and related software is shown in Table 9. MapReduce allows an unexperienced programmer to develop parallel programs and create a program capable of using computers in a cloud. In most cases, programmers are required to specify two functions only: the map function (mapper) and the reduce function (reducer) commonly utilized in functional programming. The mapper regards the key/value pair as input and generates intermediate key/value pairs. The reducer merges all the pairs associated with the same (intermediate) key and then generates an output. Table 8 summarizes the process of the map/reduce function.

The map function is applied to each input (key1, value1), where the input domain is different from the generated output pairs list (key2, value2). The elements of the list (key2, value2) are then grouped by a key. After grouping, the list (key2, value2) is divided into several lists [key2, list (value2)], and the reduce function is applied to each [key2, list (value2)] to generate a final result list (key3, value3).

7.1. MapReduce in clouds

MapReduce accelerates the processing of large amounts of data in a cloud; thus, MapReduce, is the preferred computation model of cloud providers [86]. MapReduce is a popular cloud computing framework that robotically performs scalable distributed applications [56] and provides an interface that allows for parallelization and distributed computing in a cluster of servers [12]. Srirama

Table 9
Current MapReduce projects and related software.

Reference	Software	Brief description
[77]	Hive	Hive offers a warehouse structure in HDFS
[78]	Hbase	Scalable distributed database that supports structured data storage for large tables
[79]	Madout^TM	Mahout is a machine-learning and data-mining library that has four main groups: collective filtering, categorization, clustering, and parallel frequent pattern mining; compared with other pre-existing algorithms, the Mahout library belongs to the subset that can be executed in a distributed mode and is executable by MapReduce
[80]	Pig	Pig framework involves a high-level scripting language (Pig Latin) and offers a run-time platform that allows users to execute MapReduce on Hadoop
[81]	Zookeeper^TM	High-performance service to coordinate the processes of distributed applications; ZooKeeper allows distributed processes to manage and contribute to one another through a shared hierarchical namespace of data registers (z-nodes) similar to a file system; ZooKeeper is a distributed service with *master* and *slave* nodes and stores configuration information
[82]	Spark^TM	A fast and general computation engine for Hadoop data
[83]	Chukwa	Chukwa has just passed its development stage; it is a data collection and analysis framework incorporated with MapReduce and HDFS; the workflow of Chukwa allows for data collection from distributed systems, data processing, and data storage in Hadoop; as an independent module, Chukwa is included in the Apache Hadoop distribution
[76]	Twister^TM	Provides support for iterative MapReduce computations and Twister; extremely faster than Hadoop
	MAPR	Comprehensive distribution processing for Apache Hadoop and Hbase
	YARN	A new Apache–Hadoop–MapReduce framework
[84]	Cassandra	A scalable multi-master database with no single point of failure
[85]	Avro	The tasks performed by Avro include data serialization, remote procedure calls, and data passing from one program or language to another; in the Avro framework, data are self-describing and are always stored with their own schema; this software is suitable for application to scripting language, such as Pig, because of these qualities.

Table 10
Summary of several SQL interfaces in the MapReduce framework in related literature.

Author(s)	Title of paper	Result/techniques/algorithm	Objective/description
[89]	"Jaql: A scripting language for large scale semi-structured data analysis"	Jaql	Declarative query language designed for JavaScript Object Notation
[90]	"Tenzing an SQL implementation in the MapReduce framework"	Tenzing	An SQL query execution engine
[91]	"HadoopDB: an architectural hybrid of MapReduce and DBMS technologies for analytical workloads"	HadoopDB	Comparison between Hadoop implementation of MapReduce framework and parallel SQL database management systems
[92]	"SQL/MapReduce: A practical approach to self-describing, polymorphic, and parallelizable user-defined functions"	SQL/MapReduce	Provides a parallel computation of procedural functions across hundreds of servers working together as a single relational database
[77]	"Hive - A Warehousing Solution Over a Map-Reduce Framework"	Data summarization and ad hoc querying	Presents an open-source warehouse Hive solution built on top of Hadoop
[80]	"Pig latin: a not-so-foreign language for data processing"	Pig Latin	The software takes a middle position between expressing tasks using the high-level declarative querying model in the spirit of SQL and the low-level/procedural programming model using MapReduce
[93]	"Interpreting the data: Parallel analysis with Sawzall"	Sawzall	Sawzall defines the operations to be performed in a single record of the data used at Google on top of MapReduce

et al. [56] presented an approach to apply scientific computing problems to the MapReduce framework where scientists can efficiently utilize existing resources in the cloud to solve computationally large-scale scientific data. Currently, many alternative solutions are available to deploy MapReduce in cloud environments; these solutions include using cloud MapReduce runtimes that maximize cloud infrastructure services, using MapReduce as a service, or setting up one's own MapReduce cluster in cloud instances [87]. Several strategies have been proposed to improve the performance of big data processing. Moreover, effort has been exerted to develop SQL interfaces in the MapReduce framework to assist programmers who prefer to use SQL as a high-level language to express their task while leaving all of the execution optimization details to the backend engine [88]. Table 10 shows a summary of several SQL interfaces in the MapReduce framework available in existing literature.

8. Research challenges

Although cloud computing has been broadly accepted by many organizations, research on big data in the cloud remains in its early stages. Several existing issues have not been fully addressed. Moreover, new challenges continue to emerge from applications by organization. In the subsequent sections, some of the key research challenges, such as scalability, availability, data integrity, data transformation, data quality, data heterogeneity, privacy and legal issues, and regulatory governance, are discussed.

8.1. Scalability

Scalability is the ability of the storage to handle increasing amounts of data in an appropriate manner. Scalable distributed data storage systems have been a critical part of cloud computing infrastructures [34]. The lack of cloud computing features to support RDBMSs associated with enterprise solutions has made RDBMSs less attractive for the deployment of large-scale applications in the cloud. This drawback has resulted in the popularity of NoSQL [94].

A NoSQL database provides the mechanism to store and retrieve large volumes of distributed data. The features of NoSQL databases include schema-free, easy replication support, simple API, and consistent and flexible modes. Different types of NoSQL databases, such as key-value [21], column-oriented, and document-oriented, provide support for big data. Table 11 shows a comparison of various NoSQL database technologies that provide support for large datasets.

The characteristics of scalable data storage in a cloud environment are shown in Table 12. Yan et al. [57] attempted to expend power iteration clustering (PIC) data scalability by implementing parallel power iteration clustering (p-PIC). The implementation considers two key components, namely, similarity matrix calculation and normalization and iterative matrix–vector multiplication. The process begins with the master processor indicating the beginning and ending indices for the remote data chunk. Therefore, each processor reads data from the input file and provides a similarity sub-matrix by performing the following calculation.

$Ai\,(r,c)=\frac{x_r.x_c}{\|xr\|2\|xc\|2}$, where $r\neq c$//from the input [57]

$Ai\,(r,\,:)=Ai\,(r,:)/\sum Ai\,(r,c)$//normalizes by row sum [57]

The master processor collects all row runs from the other processors and concatenates them into an overall row sum. Each processor that interacts with the main processor updates its vector by performing matrix–vector multiplication.

Wang et al. [95] proposed a new scalable data cube analysis technique called HaCube in big data clusters to overcome the challenges of large-scale data. HaCube is an extension of MapReduce; it incorporates some of MapReduce's features, such as scalability and parallel DBMS. The experimental results provided in the study indicated that HaCube performs at least 1.6× to 2.8× faster than Hadoop in terms of view maintenance. However, some improvements in performance, such as integrating more techniques from DBMS (e.g., indexing techniques), are still required.

8.2. Availability

Availability refers to the resources of the system accessible on demand by an authorized individual [98]. In a cloud environment, one of the main issues concerning cloud service providers is the availability of the data stored in the cloud. For example, one of the pressing demands on cloud service providers is to effectively serve the needs of the mobile user who requires single or multiple data

Table 11
Comparison of NoSQL databases.

Feature/ capability	NoSQL database name									
	DynamoDB	Redis	Voldemort	Cassandra	Hbase	MangoDB	SimpleDB	CouchDB	BigTable	Apache Jackrabbit
Storage type	KV	KV	KV	KV	KV	Doc	Doc & KV	Doc	CO	Doc
Initial release	2012	2009	2009	2008	2010	2009	2007	2005	2005	2010
Consistency	N/A	✓	N/A	N/A	✓	✓	N/A	N/A	✓	✓
Partition Tolerance	N/A	✓	✓	✓	✓	✓	N/A	✓	✓	N/A
Persistence	✓	✓	✓	✓	✓	✓	✓	N/A	✓	✓
High Availability	✓	✓	✓	✓	✓	✓	✓	N/A	✓	✓
Durability	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Scalability	High	High	High	High	High	High	High	High	High	High
Performance	High	High	High	High	High	High	High	High	High	High
Schema-free	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Programming Language	Java	Ansi-C	Java	Java	Java	C++	Erlang	Erlang	C, C++	Java
Platform	Linux	Windows, Linux, OS X	Windows, Linux, OS X	Windows, Linux, OS X	Windows, Linux, OS X	Windows, Linux, OS X	Windows, Linux, OS X	Windows, Linux, OS X	Windows, Linux, OS X	Windows, Linux, OS
Open Source	X	✓	✓	✓	✓	✓	×	✓	×	✓
Developer	Amazon	Salvatore Sanfilippo	LinkedIn	ASF	ASF	10gen	Amazon	ASF	Google	Apache

ASF=Apache Software Foundation, Doc=Document, KV=Key-Value, N/A=No Answer, ✓=Support, ×=Not support.

Table 12
Characteristics of scalable data storage in a cloud environment.

Reference	Characteristic	Advantage	Disadvantage
[96]	DBMS	Faster data access Faster processing	Less attractive for the deployment of large-scale data Limited
[20]	Key Value	Scales to a very large size Limitless	
[97]	Google file system (GFS)	Scalable distributed file system for large distributed data-intensive applications Delivers high aggregate performance File data is stored in different chunk servers	Garbage collection could become a problem Performance might degrade if the number of writers and random writers increases
[74]	Hadoop distributed file system (HDFS)	Stores large amounts of datasets Uses a large cluster	

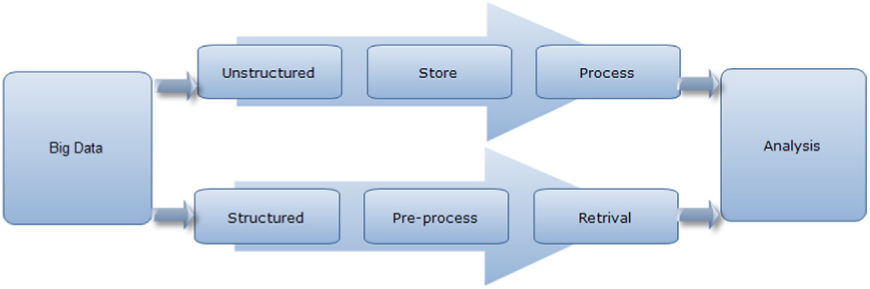

Fig. 4. Transforming big data for analysis.

within a short amount of time. Therefore, services must remain operational even in the case of a security breach [98]. In addition, with the increasing number of cloud users, cloud service providers must address the issue of making the requested data available to users to deliver high-quality services. Lee et al. [55] introduced a multi-cloud model called "rain clouds" to support big data exploitation. "Rain clouds" involves cooperation among single clouds to provide accessible resources in an emergency. Schroeck et al. [99] predicted that the demand for more real time access to data may continue to increase as business models evolve and organizations invest in technologies required for streaming data and smartphones.

8.3. Data integrity

A key aspect of big data security is integrity. Integrity means that data can be modified only by authorized parties or the data owner to prevent misuse. The proliferation of cloud-based applications provides users the opportunity to store and manage their data in cloud data centers. Such applications must ensure data integrity. However, one of the main challenges that must be addressed is to ensure the correctness of user data in the cloud. Given that users may not be physically able to access the data, the cloud should provide a mechanism for the user to check whether the data is maintained [100].

8.4. Transformation

Transforming data into a form suitable for analysis is an obstacle in the adoption of big data [101]. Owing to the variety of data formats, big data can be transformed into an analysis workflow in two ways as shown in Fig. 4.

In the case of structured data, the data is pre-processed before they are stored in relational databases to meet the constraints of schema-on-write. The data can then be retrieved for analysis. However, in unstructured data, the data must first be stored in distributed databases, such as HBase, before they are processed for analysis. Unstructured data are retrieved from distributed databases after meeting the schema-on-read constraints.

8.5. Data quality

In the past, data processing was typically performed on clean datasets from well-known and limited sources. Therefore, the results were accurate [102]. However, with the emergence of big data, data originate from many different sources; not all of these sources are well-known or verifiable. Poor data quality has become a serious problem for many cloud service providers because data are often collected from different sources. For example, huge amounts of data are generated from smartphones, where inconsistent data formats can be produced as a result of heterogeneous sources. The data quality problem is usually defined as "any difficulty encountered along one or more quality dimensions that render data completely or largely unfit for use" [103]. Therefore, obtaining high-quality data from vast collections of data sources is a challenge. High-quality data in the cloud is characterized by data consistency. If data from new sources are consistent with data from other sources, then the new data are of high quality [104].

8.6. Heterogeneity

Variety, one of the major aspects of big data characterization, is the result of the growth of virtually unlimited different sources of data. This growth leads to the heterogeneous nature of big data. Data from multiple sources are generally of different types and representation forms and significantly interconnected; they have incompatible formats and are inconsistently represented [105].

In a cloud environment, users can store data in structured, semi-structured, or unstructured format. Structured data formats are appropriate for today's database systems, whereas semi-structured data formats are appropriate only to some extent. Unstructured data are inappropriate [105] because they have a complex format that is difficult to represent in rows and columns. According to Kocarev and Jakimoski [110], the challenge is how to handle multiple data sources and types.

8.7. Privacy

Privacy concerns continue to hamper users who outsource their private data into the cloud storage. This concern has become serious with the development of big data mining and analytics, which require personal information to produce relevant results, such as personalized and location-based services [105]. Information on individuals is exposed to scrutiny, a condition that gives rise to concerns on profiling, stealing, and loss of control [106].

Currently, encryption is utilized by most researchers to ensure data privacy in the cloud [107,108]. Encryption algorithms are usually written in the form of transformations, such as $Y=E_Z(X)$ [109], where (X) refers to plaintext, (Y) is a cryptogram, and (Z) is the secret key. Encryption algorithms have a special case called block algorithms as proposed by Kocarev and Jakimoski [110], where E_Z is defined as $f_{Z:}\ f_{Z:\ X,}\ X=[0, 1\ldots\ldots\ldots..,2m-1]$, and $m=64$.

Xuyun et al. [111] discussed the problem of preserving the privacy of intermediate datasets in cloud computing; they argued that encrypting all intermediate datasets in the cloud is neither computationally effective nor cost effective because much time is required to encrypt or decrypt data. The researchers also performed experiments to reduce the cost of encryption by investigating which part of the intermediate datasets must be encrypted and which part must not.

Fan and Huang [112] proposed a variant of symmetric predicate encryption in cloud storage to control privacy and preserve search-based functionalities, such as undecrypt and revocable delegated search. Therefore, controlling the lifetime and search privileges of cloud data could become easy for the owner of the cloud storage.

Li et al. [113] proposed a flexible multi-keyword query scheme (MKQE) that significantly reduces the maintenance overhead during keyword dictionary expansion. MKQE considers the keyword weights and user access history to generate query results. MKQE improves the performance of multi-keyword ranked query over encrypted data to prevent information leakage and solve the data indexing problem.

Squicciarini et al. [114] presented a three-tier data protection architecture to provide multiple levels of privacy to cloud users. Bhagat et al. [115] investigated the issue of social networks, such as Facebook and Twitter, in which users share sensitive information over the Internet. They presented a method to deal with privacy leakages of an anonymous user's information. Itani et al. [116] presented privacy as a service model that involves a set of security protocols to ensure the confidentiality of customer data in the cloud.

Agarwal and Aggarwal [117] proposed a privacy measure based on differential entropy. Differential entropy $h(A)$ of a random variable A is defined as follows [119]:

$$H(A)=\int_{\Omega A} fA\,(a)\log_2 fA\,(a)\,da$$

where ΩA is the domain of "A." $h(A)=\log_2 a$ is a measure of uncertainty inherent in the value of "A" proposed to randomize variable "A" between 0 and (A). Therefore, the random variable with less uncertainty than "A" in [0, 1] has negative differential entropy, whereas the random variable with more uncertainty has positive differential entropy. An overview of privacy preservation and their proposed solutions, techniques, and limitations are presented in Table 13.

8.8. Legal/regulatory issues

Specific laws and regulations must be established to preserve the personal and sensitive information of users. Different countries have different laws and regulations to achieve data privacy and protection. In several countries, monitoring of company staff communications is not allowed. However, electronic monitoring is permitted under special circumstances [120]. Therefore, the question is whether such laws and regulations offer adequate

Table 13
Overview of privacy preservation in a cloud.

References	Proposed solution	Technique	Description	Limitation
[117]	Reconstruction algorithm for privacy-preserving data mining	Expectation–maximization algorithm	Measurement of privacy preservation	Efficiency of randomization
[114]	Three-tier data protection architecture	Portable data binding	Addresses the issue of privacy caused by data indexing	Protection from malicious attempts
[118]	Privacy-preserving layer (PPL) over a MapReduce framework		Ensure data privacy preservation before data are further processed by MapReduce subsequence tasks	Integration with other data processing
[111]	Upper bound privacy leakage constraint-based	Privacy-preserving cost-reducing heuristic algorithm	Identify which intermediate datasets need to be encrypted	Efficiency of the proposed technique

protection for individuals' data while enjoying the many benefits of big data in the society at large [2].

8.9. Governance

Data governance embodies the exercise of control and authority over data-related rules of law, transparency, and accountabilities of individuals and information systems to achieve business objectives [121]. The key issues of big data in cloud governance pertain to applications that consume massive amounts of data streamed from external sources [122]. Therefore, a clear and acceptable data policy with regard to the type of data that need to be stored, how quickly an individual needs to access the data, and how to access the data must be defined [50].

Big data governance involves leveraging information by aligning the objectives of multiple functions, such as telecommunication carriers having access to vast troves of customer information in the form of call detail records and marketing seeking to monetize this information by selling it to third parties [123].

Moreover, big data provides significant opportunities to service providers by making information more valuable. However, policies, principles, and frameworks that strike a stability between risk and value in the face of increasing data size and deliver better and faster data management technology can create huge challenges [124].

Cloud governance recommends the use of various policies together with different models of constraints that limit access to underlying resources. Therefore, adopting governance practices that maintain a balance between risk exposure and value creation is a new organizational imperative to unlock competitive advantages and maximize value from the application of big data in the cloud [124].

9. Open research issues

Numerous studies have addressed a number of significant problems and issues pertaining to the storage and processing of big data in clouds. The amount of data continues to increase at an exponential rate, but the improvement in the processing mechanisms is relatively slow. Only a few tools are available to address the issues of big data processing in cloud environments. State-of-the-art techniques and technologies in many important big data applications (i.e., MapReduce, Dryad, Pregel, PigLatin, MangoDB, Hbase, SimpleDB, and Cassandra) cannot solve the actual problems of storing and querying big data. For example, Hadoop and MapReduce lack query processing strategies and have low-level infrastructures with respect to data processing and management. Despite the plethora of work performed to address the problem of storing and processing big data in cloud computing environments, certain important aspects of storing and processing big data in cloud computing are yet to be solved. Some of these issues are discussed in the subsequent subsections.

9.1. Data staging

The most important open research issue regarding data staging is related to the heterogeneous nature of data. Data gathered from different sources do not have a structured format. For instance, mobile cloud-based applications, blogs, and social networking are inadequately structured similar to pieces of text messages, videos, and images. Transforming and cleaning such unstructured data before loading them into the warehouse for analysis are challenging tasks. Efforts have been exerted to simplify the transformation process by adopting technologies such as Hadoop and MapReduce to support the distributed processing of unstructured data formats. However, understanding the context of unstructured data is necessary, particularly when meaningful information is required. MapReduce programming model is the most common model that operates in clusters of computers; it has been utilized to process and distribute large amounts of data.

9.2. Distributed storage systems

Numerous solutions have been proposed to store and retrieve massive amounts of data. Some of these solutions have been applied in a cloud computing environment. However, several issues hinder the successful implementation of such solutions, including the capability of current cloud technologies to provide necessary capacity and high performance to address massive amounts of data [68], optimization of existing file systems for the volumes demanded by data mining applications, and how data can be stored in such a manner that they can be easily retrieved and migrated between servers.

9.3. Data analysis

The selection of an appropriate model for large-scale data analysis is critical. Talia [41] pointed out that obtaining useful information from large amounts of data requires scalable analysis algorithms to produce timely results. However, current algorithms are inefficient in terms of big data analysis. Therefore, efficient data analysis tools and technologies are required to process such data. Each algorithm performance ceases to increase linearly with increasing computational resources. As researchers continue to probe the issues of big data in cloud computing, new problems in big data processing arise from the transitional data analysis techniques. The speed of stream data arriving from different data sources must be processed and compared with historical information within a certain period of time. Such data sources may contain different formats, which makes the integration of multiple sources for analysis a complex task [125].

9.4. Data security

Although cloud computing has transformed modern ICT technology, several unresolved security threats exist in cloud computing. These security threats are magnified by the volume, velocity, and variety of big data. Moreover, several threats and issues, such as privacy, confidentiality, integrity, and availability of data, exist in big data using cloud computing platforms. Therefore, data security must be measured once data are outsourced to cloud service providers. The cloud must also be assessed at regular intervals to protect it against threats. Cloud vendors must ensure that all service level agreements are met. Recently, some controversies have revealed how some security agencies use data generated by individuals for their own benefit without permission. Therefore, policies that cover all user privacy concerns should be developed. Traditionally, the most common technique for privacy and data control is to protect the systems utilized to manage data rather than the data itself; however, such systems have proven to be vulnerable. Utilizing strong cryptography to encapsulate sensitive data in a cloud computing environment and developing a novel algorithm that efficiently allows for key management and secure key exchange are important to manage access to big data, particularly as they exist in the cloud independent of any platform. Moreover, the issue with integrity is that previously developed hashing schemes are no longer applicable to large amounts of data. Integrity verification is also difficult because of the lack of support, given remote data access and the lack of information on internal storage.

10. Conclusion

The size of data at present is huge and continues to increase every day. The variety of data being generated is also expanding. The velocity of data generation and growth is increasing because of the proliferation of mobile devices and other device sensors connected to the Internet. These data provide opportunities that allow businesses across all industries to gain real-time business insights. The use of cloud services to store, process, and analyze data has been available for some time; it has changed the context of information technology and has turned the promises of the on-demand service model into reality. In this study, we presented a review on the rise of big data in cloud computing. We proposed a classification for big data, a conceptual view of big data, and a cloud services model. This model was compared with several representative big data cloud platforms. We discussed the background of Hadoop technology and its core components, namely, MapReduce and HDFS. We presented current MapReduce projects and related software. We also reviewed some of the challenges in big data processing. The review covered volume, scalability, availability, data integrity, data protection, data transformation, data quality/heterogeneity, privacy and legal/regulatory issues, data access, and governance. Furthermore, the key issues in big data in clouds were highlighted. In the future, significant challenges and issues must be addressed by the academia and industry. Researchers, practitioners, and social science scholars should collaborate to ensure the long-term success of data management in a cloud computing environment and to collectively explore new territories.

Acknowledgment

This paper is financially supported by the Malaysian Ministry of Education under the University of Malaya High Impact Research Grant UM.C/625/1/HIR/MoE/FCSIT/03.

References

[1] R.L .Villars, C.W. Olofson, M. Eastwood, Big data: what it is and why you should care, White Paper, IDC, 2011, MA, USA.

[2] R. Cumbley, P. Church, Is Big Data creepy? Comput. Law Secur. Rev. 29 (2013) 601–609.

[3] S. Kaisler, F. Armour, J.A. Espinosa, W. Money, Big Data: Issues and Challenges Moving Forward, System Sciences (HICSS), 2013, in: Proceedings of the 46th Hawaii International Conference on, IEEE, 2013, pp. 995–1004.

[4] L. Chih-Wei, H. Chih-Ming, C. Chih-Hung, Y. Chao-Tung, An Improvement to Data Service in Cloud Computing with Content Sensitive Transaction Analysis and Adaptation, Computer Software and Applications Conference Workshops (COMPSACW), 2013 IEEE 37th Annual, 2013, pp. 463–468.

[5] L. Chang, R. Ranjan, Z. Xuyun, Y. Chi, D. Georgakopoulos, C. Jinjun, Public Auditing for Big Data Storage in Cloud Computing – a Survey, Computational Science and Engineering (CSE), 2013 IEEE 16th International Conference on, 2013, pp. 1128–1135.

[6] M. Cox, D. Ellsworth, Managing Big Data For Scientific Visualization, ACM Siggraph, MRJ/NASA Ames Research Center, 1997.

[7] J. Manyika, M. Chui, B. Brown, J. Bughin, R. Dobbs, C. Roxburgh, A.H. Byers, Big data: The next frontier for innovation, competition, and productivity, (2011).

[8] P. Zikopoulos, K. Parasuraman, T. Deutsch, J. Giles, D. Corrigan, Harness the Power of Big Data The IBM Big Data Platform, McGraw Hill Professional, 2012.

[9] J.J. Berman, Introduction, in: Principles of Big Data, Morgan Kaufmann, Boston, 2013, xix–xxvi (pp).

[10] J. Gantz, D. Reinsel, Extracting value from chaos, IDC iView (2011) 1–12.

[11] J.K. Laurila, D. Gatica-Perez, I. Aad, J. Blom, O. Bornet, T.-M.-T. Do, O. Dousse, J. Eberle, M. Miettinen, The mobile data challenge: Big data for mobile computing research, Workshop on the Nokia Mobile Data Challenge, in: Proceedings of the Conjunction with the 10th International Conference on Pervasive Computing, 2012, pp. 1–8.

[12] D.E. O'Leary, Artificial intelligence and big data, IEEE Intell. Syst. 28 (2013) 96–99.

[13] M. Chen, S. Mao, Y. Liu, Big data: a survey, Mob. Netw. Appl. 19 (2) (2014) 1–39.

[14] B.P. Rao, P. Saluia, N. Sharma, A. Mittal, S.V. Sharma, Cloud computing for Internet of Things & sensing based applications, in: Proceedings of the Sensing Technology (ICST), 2012 Sixth International Conference on, IEEE, 2012, pp. 374–380.

[15] B. Franks, Taming the Big Data Tidal Wave: Finding Opportunities in Huge Data Streams with Advanced Analytics, Wiley. com John Wiley Sons Inc, 2012.

[16] D.J. Abadi, P.A. Boncz, S. Harizopoulos, Column-oriented database systems, Proc. VLDB Endow 2 (2009) 1664–1665.

[17] F. Chang, J. Dean, S. Ghemawat, W.C. Hsieh, D.A. Wallach, M. Burrows, T. Chandra, A. Fikes, R.E. Gruber, Bigtable: a distributed storage system for structured data, ACM Trans. Comput. Syst. (TOCS) 26 (2008) 4.

[18] P. Neubauer, Graph databases, NOSQL and Neo4j, in, 2010.

[19] M. Seeger, S. Ultra-Large-Sites, Key-Value stores: a practical overview, Comput. Sci. Media (2009).

[20] G. DeCandia, D. Hastorun, M. Jampani, G. Kakulapati, A. Lakshman, A. Pilchin, S. Sivasubramanian, P. Vosshall, W. Vogels, Dynamo: amazon's highly available key-value store, SOSP 41 (6) (2007) 205–220.

[21] S. Das, D. Agrawal, A. El Abbadi, G-store: a scalable data store for transactional multi key access in the cloud, in: Proceedings of the 1st ACM symposium on Cloud computing, ACM, 2010, pp. 163–174.

[22] F. Lin, W.W. Cohen, Power iteration clustering, in: Proceedings of the 27th International Conference on Machine Learning (ICML-10), 2010, pp. 655–662.

[23] R.C. Taylor, An overview of the Hadoop/MapReduce/Hbase framework and its current applications in bioinformatics, BMC Bioinf. 11 (2010) S1.

[24] A. Lakshman, P. Malik, The Apache cassandra project, in, 2011.

[25] E. Rahm, H.H. Do, Data cleaning: problems and current approaches, IEEE Data Eng. Bull. 23 (2000) 3–13.

[26] J. Quackenbush, Microarray data normalization and transformation, Nat. Genet. 32 (2002) 496–501.

[27] Y. Chen, S. Alspaugh, R. Katz, Interactive analytical processing in big data systems: a cross-industry study of MapReduce workloads, Proc. VLDB Endow. 5 (2012) 1802–1813.

[28] L. Neumeyer, B. Robbins, A. Nair, A. Kesari, S4: Distributed Stream Computing Platform, Data Mining Workshops (ICDMW), 2010 IEEE International Conference on, 2010, pp. 170–177.

[29] J. Hurwitz, A. Nugent, F. Halper, M. Kaufman, Big data for dummies, For Dummies (2013).

[30] M. Armbrust, A. Fox, R. Griffith, A.D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, M. Zaharia, A view of cloud computing, Commun. ACM 53 (2010) 50–58.

[31] L. Huan, Big data drives cloud adoption in enterprise, IEEE Internet Comput. 17 (2013) 68–71.

[32] S. Pandey, S. Nepal, Cloud computing and scientific applications — big data, Scalable Anal. Beyond, Futur. Gener. Comput. Syst. 29 (2013) 1774–1776.

[33] D. Warneke, O. Kao, Nephele: efficient parallel data processing in the cloud, in: Proceedings of the 2nd workshop on many-task computing on grids and supercomputers, ACM, 2009, p. 8.

[34] P. Mell, T. Grance, The NIST definition of cloud computing (draft), NIST Spec. Publ. 800 (2011) 7.

[35] A. Giuseppe, B. Alessio, D. Walter, P. Antonio, Survey cloud monitoring: a survey, Comput. Netw. 57 (2013) 2093–2115.

[36] T. Gunarathne, B. Zhang, T.-L. Wu, J. Qiu, Scalable parallel computing on clouds using Twister4Azure iterative MapReduce, Futur. Gener. Comput. Syst. 29 (2013) 1035–1048.

[37] A. O'Driscoll, J. Daugelaite, R.D. Sleator, 'Big data', Hadoop and cloud computing in genomics, J. Biomed. Inform. 46 (2013) 774–781.

[38] N. Fernando, S.W. Loke, W. Rahayu, Mobile cloud computing: a survey, Futu. Gener. Comput. Syst. 29 (2013) 84–106.

[39] R. Holman, Mobile Cloud Application Revenues To Hit $9.5 billion by 2014, Driven by Converged Mobile Services, in: The Juniper Research, 2010.

[40] Z. Sanaei, S. Abolfazli, A. Gani, R. Buyya, Heterogeneity in mobile cloud computing: taxonomy and open challenges, IEEE Commun. Surv. Tutor. (2013) 1–24.

[41] D. Talia, Clouds for scalable big data analytics, Computer 46 (2013) 98–101.

[42] C. Ji, Y. Li, W. Qiu, U. Awada, K. Li, Big data processing in cloud computing environments, Pervasive Systems, Algorithms and Networks (ISPAN), 2012,in: Proceedings of the 12th International Symposium on, IEEE, 2012, pp. 17–23.

[43] J. Dean, S. Ghemawat, MapReduce: simplified data processing on large clusters, Commun. ACM 51 (2008) 107–113.

[44] D. Bollier, C. Firestone, M, The Promise and Peril of Big Data, Aspen Institute, Communications and Society Program Washington, DC, USA, 2010.

[45] H. Miller, E, Big-data in cloud computing: a taxonomy of risks, Inf. Res. 18 (2013) 571.

[46] O. Kwon, N. Lee, B. Shin, Data quality management, data usage experience and acquisition intention of big data analytics, Int. J. Inf. Manag. 34 (3) (2014) 387–394.

[47] K. Singh, S.C. Guntuku, A. Thakur, C. Hota, Big data analytics framework for peer-to-peer botnet detection using random forests, Inf. Sci. (2014).

[48] J.L. Schnase, D.Q. Duffy, G.S. Tamkin, D. Nadeau, J.H. Thompson, C.M. Grieg, M.A. McInerney, W.P. Webster, MERRA Analytic Services: Meeting the Big Data challenges of climate science through cloud-enabled Climate Analytics-as-a-Service, Computers, Environment and Urban Systems, (2014).

[49] B.K. Tannahill, M. Jamshidi, System of systems and big data analytics – bridging the gap, Comput. Electr. Eng. 40 (2014) 2–15.

[50] J. Abawajy, Symbioses of Big Data and Cloud Computing: Opportunities & Challenges, (2013).

[51] S. Aluru, Y. Simmhan, A special issue of journal of parallel and distributed computing: scalable systems for big data management and analytics, J.Parallel Distrib. Comput. 73 (2013) 896.

[52] S. Hipgrave, Smarter fraud investigations with big data analytics, Netw. Secur. 2013 (2013) 7–9.

[53] Z. Linquan, W. Chuan, L. Zongpeng, G. Chuanxiong, C. Minghua, F.C.M. Lau, Moving big data to the cloud: an online cost-minimizing approach, IEEE J. Sel. Areas Commun. 31 (2013) 2710–2721.

[54] H. Demirkan, D. Delen, Leveraging the capabilities of service-oriented decision support systems: putting analytics and big data in cloud, Decis. Support Syst. 55 (2013) 412–421.

[55] S. Lee, H. Park, Y. Shin, Cloud computing availability: multi-clouds for big data service, Communications in Computer and Information Science 310 (2012) 799–806.

[56] S.N. Srirama, P. Jakovits, E. Vainikko, Adapting scientific computing problems to clouds using MapReduce, Futur. Gener. Comput. Syst. 28 (2012) 184–192.

[57] W. Yan, U. Brahmakshatriya, Y. Xue, M. Gilder, B. Wise, p-PIC: parallel power iteration clustering for big data, J. Parallel Distrib. Comput. 73 (3) (2012) 352–359.

[58] E.E. Schadt, M.D. Linderman, J. Sorenson, L. Lee, G.P. Nolan, Cloud and heterogeneous computing solutions exist today for the emerging big data problems in biology, Nat. Rev. Genet. 12 (2011) 224. (-224).

[59] Amazon, AWS Case Study: SwiftKey. ⟨http://aws.amazon.com/solutions/case-studies/big-data⟩, (accessed 05.07.14).

[60] Microsoft, 343 Industries Gets New User Insights from Big Data in the Cloud. ⟨http://www.microsoft.com/casestudies/⟩, (accessed 15.07.14).

[61] Google, Case study: How redBus uses BigQuery to Master Big Data. ⟨https://developers.google.com/bigquery/case-studies/⟩, (accessed 22.07.14).

[62] Cloudera, Nokia: Using Big Data to Bridge the Virtual & Physical Worlds. ⟨http://www.cloudera.com/content/dam/cloudera/documents/Cloudera-Nokia-case-study-final.pdf⟩, (accessed 24.07.14).

[63] Alacer, Case Studies: Big Data. ⟨http://www.alacergroup.com/practice-category/big-data/case-studies-big-data/⟩, (accessed 24.07.14).

[64] J.G. Reid, A. Carroll, N. Veeraraghavan, M. Dahdouli, A. Sundquist, A. English, M. Bainbridge, S. White, W. Salerno, C. Buhay, Launching genomics into the cloud: deployment of Mercury, a next generation sequence analysis pipeline, BMC Bioinf. 15 (2014) 30.

[65] P. Noordhuis, M. Heijkoop, A. Lazovik, Mining twitter in the cloud: A case study, Cloud Computing (CLOUD), 2010, in: Proceedings of IEEE 3rd International Conference on, IEEE, Miami, FL, 2010, pp. 107–114.

[66] C. Zhang, H. De Sterck, A. Aboulnaga, H. Djambazian, R. Sladek, Case study of scientific data processing on a cloud using hadoop, High Performance Computing Systems and Applications, Springer, 2010, 400–415.

[67] F. Faghri, S. Bazarbayev, M. Overholt, R. Farivar, R.H. Campbell, W.H. Sanders, Failure scenario as a service (FsaaS) for Hadoop clusters, in: Proceedings of the Workshop on Secure and Dependable Middleware for Cloud Monitoring and Management, ACM, 2012, p. 5.

[68] N. Leavitt, Storage challenge: where will all that big data go? Computer 46 (2013) 22–25.

[69] K. Strauss, D. Burger, What the future holds for solid-state memory, Computer 47 (2014) 24–31.

[70] K. Mayama, W. Skulkittiyut, Y. Ando, T. Yoshimi, M. Mizukawa, Proposal of object management system for applying to existing object storage furniture, System Integration (SII),2011 IEEE/SICE International Symposium on, IEEE, 2011, pp. 279–282.

[71] W. Hu, D. Hu, C. Xie, F. Chen, IEEE International Conference on A New Data Format and a New Error Control Scheme for Optical-Storage Systems, Networking, Architecture, and Storage, 2007, NAS 2007 2007, pp. 193–198.

[72] L. Hao, D. Han, IEEE Conference on The study and design on secure-cloud storage system, Electrical and Control Engineering (ICECE), 2011 International 2011, pp. 5126–5129.

[73] T. White, Hadoop: The Definitive Guide: The Definitive Guide, O'Reilly Media, Sebastapol, CA, 2009.

[74] K. Shvachko, K. Hairong, S. Radia, R. Chansler, The Hadoop Distributed File System, Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on, 2010, pp. 1–10.

[75] S. Ghemawat, H. Gobioff, S.-T. Leung, The Google file system, ACM SIGOPS Oper. Syst. Rev. ACM 37 (5) (2003) 29–43.

[76] J. Ekanayake, H. Li, B. Zhang, T. Gunarathne, S.-H. Bae, J. Qiu, G. Fox, Twister: a runtime for iterative mapreduce, in: Proceedings of the 19th ACM International Symposium on High Performance Distributed Computing, ACM, 2010, pp. 810–818.

[77] A. Thusoo, J.S. Sarma, N. Jain, Z. Shao, P. Chakka, S. Anthony, H. Liu, P. Wyckoff, R. Murthy, Hive: a warehousing solution over a map-reduce framework, Proc. VLDB Endow. 2 (2009) 1626–1629.

[78] L. George, Hbase: The Definitive Guide, O'Reilly Media, Inc., Sebastopol, CA, 2011.

[79] S. Owen, R. Anil, T. Dunning, E. Friedman, Mahout in action, Manning Publications Co., 2011.

[80] C. Olston, B. Reed, U. Srivastava, R. Kumar, A. Tomkins, Pig latin: a not-so-foreign language for data processing, in: Proceedings of the 2008 ACM SIGMOD international conference on Management of data, ACM, 2008, pp. 1099–1110.

[81] P. Hunt, M. Konar, F.P. Junqueira, B. Reed, ZooKeeper: wait-free coordination for internet-scale systems, in: Proceedings of the 2010 USENIX conference on USENIX annual technical conference, 2010, pp. 11–11.

[82] M. Zaharia, M. Chowdhury, M.J. Franklin, S. Shenker, I. Stoica, Spark: cluster computing with working sets, in: Proceedings of

the 2nd USENIX conference on Hot topics in cloud computing, 2010, pp. 10–10.
[83] A. Rabkin, R. Katz, Chukwa: A system for reliable large-scale log collection, in: Proceedings of the 24th international conference on Large installation system administration, USENIX Association, 2010, pp. 1–15.
[84] A. Cassandra, The Apache Cassandra project, in.
[85] S. Hoffman, Apache Flume: Distributed Log Collection for Hadoop, Packt Publishing Ltd., Birmingham, UK, 2013.
[86] X. Zhifeng, X. Yang, Security and privacy in cloud computing, IEEE Commun. Surv. Tutor. 15 (2013) 843–859.
[87] T. Gunarathne, T.-L. Wu, J. Qiu, G. Fox, MapReduce in the Clouds for Science, IEEE Second International Conference on Cloud Computing Technology and Science (CloudCom), 2010, pp. 565–572.
[88] S. Sakr, A. Liu, A.G. Fayoumi, The family of MapReduce and large-scale data processing systems, ACM Comput. Surv. (CSUR) 46 (2013) 11.
[89] K.S. Beyer, V. Ercegovac, R. Gemulla, A. Balmin, M. Eltabakh, C.-C. Kanne, F. Ozcan, E.J. Shekita, Jaql: a scripting language for large scale semistructured data analysis, Proc. VLDB Conf. (2011).
[90] L. Lin, V. Lychagina, W. Liu, Y. Kwon, S. Mittal, M. Wong, Tenzing a sql implementation on the mapreduce framework, (2011).
[91] A. Abouzeid, K. Bajda-Pawlikowski, D. Abadi, A. Silberschatz, A. Rasin, HadoopDB: an architectural hybrid of MapReduce and DBMS technologies for analytical workloads, Proc. VLDB Endow. 2 (2009) 922–933.
[92] E. Friedman, P. Pawlowski, J. Cieslewicz, SQL/MapReduce: a practical approach to self-describing, polymorphic, and parallelizable user-defined functions, Proc. VLDB Endow. 2 (2009) 1402–1413.
[93] R. Pike, S. Dorward, R. Griesemer, S. Quinlan, Interpreting the data: parallel analysis with Sawzall, Sci. Progr. 13 (2005) 277–298.
[94] R. Cattell, Scalable SQL and NoSQL data stores, ACM SIGMOD Record, 39 (4), ACM New York, NY, USA, 2011, 12–27.
[95] Z. Wang, Y. Chu, K.-L. Tan, D. Agrawal, A.E. Abbadi, X. Xu, Scalable Data Cube Analysis over Big Data, arXiv preprint arXiv:1311.5663 (2013).
[96] R. Ramakrishnan, J. Gehrke, Database Management Systems, Osborne/McGraw-Hill, New York, 2003.
[97] S. Ghemawat, H. Gobioff, S.-T. Leung, The Google file system, SIGOPS Oper. Syst. Rev. 37 (2003) 29–43.
[98] D. Zissis, D. Lekkas, Addressing cloud computing security issues, Futur. Gener. Comput. Syst. 28 (2012) 583–592.
[99] M. Schroeck, R. Shockley, J. Smart, D. Romero-Morales, P. Tufano, Analytics: The real-world use of big data, in, IBM Global Business Services, 2012.
[100] R. Sravan Kumar, A. Saxena, Data integrity proofs in cloud storage, in: Proceedings of the Third International Conference on Communication Systems and Networks (COMSNETS), 2011, pp. 1–4.
[101] R. Akerkar, Big Data Computing, CRC Press, 2013.
[102] T.C. Redman, A. Blanton, Data Quality for the Information Age, Artech House, Inc., Norwood, MA, USA, 1997.
[103] D.M. Strong, Y.W. Lee, R.Y. Wang, Data quality in context, Commun. ACM, 40, , 1997, 103–110.
[104] K. Weber, G. Rincon, A. Van Eenennaam, B. Golden, J. Medrano, Differences in allele frequency distribution of bovine high-density genotyping platforms in holsteins and jerseys, Western section American society of Animal science, 2012, p. 70.
[105] D. Che, M. Safran, Z. Peng, From big data to big data mining: challenges, issues, and opportunities, in: B. Hong, X. Meng, L. Chen, W. Winiwarter, W. Song (Eds.), Database Systems for Advanced Applications, Springer, Berlin Heidelberg, 2013, pp. 1–15.
[106] O. Tene, J. Polonetsky, Privacy in the age of big data: a time for big decisions, Stanford Law Review Online 64 (2012) 63.
[107] L. Hsiao-Ying, W.G. Tzeng, A secure erasure code-based cloud storage system with secure data forwarding, parallel and distributed systems, IEEE Transactions on, 23 (2012) pp. 995–1003.
[108] C. Ning, W. Cong, M. Li, R. Kui, L. Wenjing, Privacy-preserving multi-keyword ranked search over encrypted cloud data, INFOCOM, 2011 Proceedings IEEE, 2011, pp. 829–837.
[109] C.E. Shannon, Communication theory of secrecy systems*, Bell Syst. Tech. J. 28 (1949) 656–715.
[110] L. Kocarev, G. Jakimoski, Logistic map as a block encryption algorithm, Phys. Lett. 289 (4–5) (2001) 199–206.
[111] Z. Xuyun, L. Chang, S. Nepal, S. Pandey, C. Jinjun, A Privacy Leakage Upper Bound Constraint-Based Approach for Cost-Effective Privacy Preserving of Intermediate Data Sets in Cloud, Parallel and Distributed Systems, IEEE Transactions on 24 (2013) pp. 1192–1202.
[112] C.-I. Fan, S.-Y. Huang, Controllable privacy preserving search based on symmetric predicate encryption in cloud storage, Futur. Gener. Comput. Syst. 29 (2013) 1716–1724.
[113] R. Li, Z. Xu, W. Kang, K.C. Yow, C.-Z. Xu, Efficient multi-keyword ranked query over encrypted data in cloud computing, Futur. Gener. Comput. Syst. (2013).
[114] A. Squicciarini, S. Sundareswaran, D. Lin, Preventing Information Leakage from Indexing in the Cloud, Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on, 2010, pp. 188–195.
[115] S. Bhagat, G. Cormode, B. Krishnamurthy, D. Srivastava, Privacy in dynamic social networks, in: Proceedings of the 19th international conference on World wide web, ACM, Raleigh, North Carolina, USA, 2010, pp. 1059–1060.
[116] W. Itani, A. Kayssi, A. Chehab, Privacy as a Service: Privacy-Aware Data Storage and Processing in Cloud Computing Architectures, Dependable, Autonomic and Secure Computing, 2009. DASC '09, in: Proceedings of the Eighth IEEE International Conference on, 2009, pp. 711–716.
[117] D. Agrawal, C.C. Aggarwal, On the design and quantification of privacy preserving data mining algorithms, in: Proceedings of the Twentieth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, ACM, Santa Barbara, California, USA, 2001, pp. 247–255.
[118] Z. Xuyun, L. Chang, S. Nepal, D. Wanchun, C. Jinjun, Privacy-Preserving Layer over MapReduce on Cloud, in: International Conference on Cloud and Green Computing (CGC), 2012, pp. 304–310.
[119] D.P. Bertsekas, Nonlinear programming, (1999).
[120] C. Tankard, Big data security, Netw. Secur. 2012 (2012) 5–8.
[121] P. Malik, Governing big data: principles and practices, IBM J. Res. Dev. 57 (1) (2013) 1. (-1: 13).
[122] D. Loshin, Chapter 5 – data governance for big data analytics: considerations for data policies and processes, in: D. Loshin (Ed.), Big Data Analytics, Morgan Kaufmann, Boston, 2013, pp. 39–48.
[123] S. Soares, Big Data Governance, Sunilsoares, 2012.
[124] P.P. Tallon, Corporate governance of big data: perspectives on value, risk, and cost, Computer 46 (2013) 32–38.
[125] M.D. Assuncao, R.N. Calheiros, S. Bianchi, M.A. Netto, R. Buyya, Big Data Computing and Clouds: Challenges, Solutions, and Future Directions, arXiv preprint arXiv:1312.4722, (2013).
[126] Khan, Abdul Nasir, et al. BSS: block-based sharing scheme for secure data storage services in mobile cloud environment. The Journal of Supercomputing (2014) 1–31.
[127] Khan, Abdul Nasir, et al., Incremental proxy re-encryption scheme for mobile cloud computing environment, The Journal of Supercomputing 68 (2) (2014) 624–651.