A survey on data stream, big data and real-time

Eliza H.A. Gomes* and Patrícia D.M. Plentz

Department of Informatics and Statistics (INE),
Federal University of Santa Catarina (UFSC),
Florianópolis, SC, Brazil
Email: eliza.gomes@posgrad.ufsc.br
Email: patricia.plentz@ufsc.br
*Corresponding author

Carlos R. De Rolt

Centre of Management and Socioeconomic Science (ESAG),
State University of Santa Catarina (UDESC),
Florianópolis, SC, Brazil
Email: rolt@udesc.br

Mario A.R. Dantas

Department of Computer Science (DCC),
Federal University of Juiz de Fora (UFJF),
Juiz de Fora, MG, Brazil
Email: mario.dantas@ice.ufjf.br

Abstract: Real-time concept is being widely used by a society that seeks to speed communications, decisions and their daily activities. Even though this term is not used with the necessary conceptual precision, it makes clear the importance that time exerts on computer systems. Nowadays, the big data scenario, this concept is important and used with different meanings, which can define failure or successful of applications. This article aims to present a systematic literature review on the topics of data stream, big data and real-time. For this, we developed a protocol revision in which were determined research questions, the search term, the search source and the inclusion and exclusion criteria of articles. After an extensive study, we classify the articles selected in seven categories according to real-time concept used. Finally, we present a discussion that shows that there is not convergence on real-time concepts in the big data literature.

Keywords: real-time; big data; data stream; stream processing; big data stream tools; time constraint.

Reference to this paper should be made as follows: Gomes, E.H.A., Plentz, P.D.M., De Rolt, C.R. and Dantas, M.A.R. (2019) 'A survey on data stream, big data and real-time', *Int. J. Networking and Virtual Organisations*, Vol. 20, No. 2, pp.143–167.

Biographical notes: Eliza H.A. Gomes is a PhD student of the Graduate Program of Computer Science (PGCC) in the Department of Informatics and Statistics (INE), Technology Centre (CTC), Federal University of Santa Catarina (UFSC), Brazil. Currently, she is working in the Research Laboratory of Distributed Systems (LaPeSD) supervised by Prof. Mario Dantas and cosupervised by Prof. Patricia Plentz. Her interesting researches areas are: distributed systems, operating systems, computer network and real-time.

Patrícia D.M. Plentz is currently an Associate Professor of Computer Science at the Federal University of Santa Catarina (UFSC), Brazil. She received her PhD and MSc degree from UFSC in 2008 and 2002, respectively, and her BSc in Computer Science from University of Cruz Alta (UNICRUZ) in 2000. Her research interests include real-time distributed systems, forecast of deadline missing and temporal constraints of mobile robots.

Carlos R. De Rolt is an Associate Professor at the State University of Santa Catarina (UDESC), coordinator of LabGES – Laboratory Management Technologies CCAUDESC which operates in research in the areas of information management technologies in dynamic business networks, time stamping, security, electronic documents, digital signature and public key infrastructure. In 2013 coordinated the development of a management model for cluster of innovation in the areas of nanotechnology in the technological pole of FlorianÃ³polis. In 2014, he realised his postdoctorate at UNIBO – University of Bologna in a project that aims to develop the organisational and information security issues in virtual communities of WBAN e-health systems, big data and mobile crowdsensing in participatory management of smart cities.

Mario A.R. Dantas is a Professor in the Department of Computer Science (DCC) at the Exact Sciences Institute (ICE), Federal University of Juiz de Fora (UFJF) and in the Graduate Program in Computer Science (PPGCC), at the Technology Centre (CTC), Federal University of Santa Catarina (UFSC), with a PhD in Computer Science from the University of Southampton (UK). He is a Visiting Professor at the University of Western Ontario (Canada) and a Senior Visiting Researcher in Riken (Japan). He is the author of a hundred scientific articles, dozens of chapters in books and three books. He has advised numerous undergraduate, specialisation, master and doctorate research works. He has acted as a consultant on various projects with industry in the areas of IoT, networking, distributed systems, and high-performance environments

1 Introduction

We are currently witnessing the era of big data wherein large volumes of heterogeneous data is generated (Sagiroglu and Sinanc, 2013), besides having an increasing need to provide faster processing and analysis of this data. Recent studies of major IT corporations, for example, Cisco (Pepper and John, 2016) and IBM (IBM, 2016), indicate that are generated daily 2.5 exabytes of data. It is estimated at 2020 this number reached of 40 yottabyte, which means about 5,200 gigabytes for every person on earth. This large amount of information should have predominantly originated portion of Internet of Things (IoT) approach.

New techniques and technologies are being developed aiming to meet the big data characteristics, known as 5Vs (velocity, variety, volume, value and veracity). Tools such

as MapReduce that perform batch processing are being gradually substituted by data stream processing tools to meet demand (Shahrivari, 2014).

Data stream can be understood as the receiving and continuous processing of the data. Financial applications, network monitoring, security, telecommunications data management, web applications and sensor networks are examples of applications that use data stream (Babcock et al., 2002). These kind of applications may perform the computation within a temporal constraint – deadline, computation time, response time, and other restrictions (Liu, 2000). However, despite the importance of temporal constraints for data stream systems, to the best of our knowledge the state of the art lacks of a comprehensive and in-depth analysis of time constraints accomplishments for data stream domain.

Tools such as Apache Spark and Apache Storm are widely used because they offer low latency and faster processing of data stream. Apache Storm also offer real-time processing. The difference between fast processing and real-time processing should be clarified because even though real-time term is not used with the necessary conceptual precision, it makes clear the importance that time behaviour exerts on computer systems.

Before tackle this context, this survey proposes to carry out a systematic review with the goal to present approaches of data stream in distributed and parallel environments considering big data and real-time aspects, and thus confront the real-time concepts introduced by articles studied.

This survey is organised such as following: in Section 2 we briefly present the concepts discussed by this survey (big data, data stream and real-time). The methodology developed and implemented are presented in the Section 3. In Section 4 we discussed and confronted the real-time concepts presented in the Section 2 with the selected articles. Lastly, in Section 5 we presented the conclusions and intentions of future works.

2 Overview

In this section we briefly introduce some concepts related to *Data Stream*, *Big Data* and *Real-Time*. In addition, we present a discussion about the implementation of concepts in real cases, citing as examples research related to smart cities.

2.1 Data Stream

In data stream systems, the data arrive as a continuous, infinite, fast and variable sequence in time (Safaei, 2016). Examples of data stream applications are: financial applications, network monitoring, security, telecommunications data management, web applications and sensor networks (Babcock et al., 2002).

Data streams differs from conventional relational model of storing several ways (Babcock et al., 2002):

- the data stream elements arrive online
- the system has no control the order in which the data elements arrive
- data streams are potentially unlimited in size
- once the data stream element is processed it is discarded or stored.

In view of this, a new software system class has been developed, stream processing engines, specifically to support large volumes and low latency stream processing applications (Stonebraker and Zdonik, 2005).

2.2 Big Data

According to Lee et al. (2015a) big data can be conceptualised as a collection of data sets so large and complex that it becomes difficult to process, manipulate and analyse data in a good time. To solve this issue, various techniques and big data tools have been developed and used in ordinary applications (Inacio and Dantas, 2014). The big data techniques involve a number of disciplines including: statistics, data mining, machine learning, neural networks, social network analysis, signal processing, pattern recognition, optimisation methods and visualisation approaches (Philip Chen and Zhang, 2014). On the other hand, tools are necessary so that it is possible to build a big data platform to capture, process, store, analyse and visualise large volumes of data. To Gomes et al. (2016) a structured platform with correct tools determines positively the efficient use of data.

Big data has five characteristics known as the 5 Vs (Demchenko et al., 2013) and (Xhafa et al., 2015b), that are:

1 *Volume*: the amount of data is very large compared to conventional computing systems.

2 *Velocity*: is a characteristic required for big data and all processes (Sagiroglu and Sinanc, 2013).

3 *Variety*: this characteristic refers to data and source heterogeneity. Data can be structured or unstructured and can have various formats.

4 *Veracity*: the data is checked to avoid possible errors and missing values. Furthermore, the data are processed and the results should be reliable.

5 *Value*: it refers to the value that the data can bring to the intended process, activity or analysis.

According to the presented concepts, big data stream can be understood as the continuous, infinite and fast arrival of large volumes of complex data and heterogeneous, that seek efficient storage and rapid processing and analysis.

2.3 Real-Time

Despite of research on real-time systems to be a classical area of computer science (Shaw, 1989), (Liu, 2000), (Stankovic et al., 2012), the term ‘real-time’ have been used without the necessary conceptual precision in other areas, such as data stream and big data. Currently, we can see that many real-time concepts are being discussed and accepted in the academy as synonym of fast computing. This subsection aims to present three concepts of real-time respectively applied to digital real-time signal processing, real-time systems and real-time data stream.

To Kuo et al. (2013) digital real-time signal processing is demand of hardware and software designed to complete a predefined task within a specified period of time. Smith

(1999) defines that in the real-time processing, the output signal is produced at the same time the input signal is to be acquired.

Stankovic and Ramamritham (1990) define real-time system as a system in which its correct operation not only depends on the logic computation response but also the time in which these results are presented. The real-time system tasks must have deadline that is a timing constraint imposed by the system application (Plentz et al., 2012). This deadline can be hard, soft or firm (Stankovic et al., 2012). Hard deadline means that it is essential to the security of the system that the deadline is always met. Soft deadline means that tasks can be performed after the deadline, implying the reduction system quality. On the other hand, firm deadline means that the results obtained after the deadline has no value.

Stonebraker and Zdonik (2005) present eight characteristics that a software system must provide for real-time processing of data stream and, thus, generate processing high volumes of data and low latency. Among the eight requirements, we can cite:

- *Keep the data moving*: the data stream processing must to be continued without the storage operation.
- *Generate predictable outcomes*: there must be guarantees that the processing results are deterministic and repeatable.
- *Guarantee data safety and availability*: data integrity must be maintained and the data stream processing system must be highly available.
- *Process and respond instantaneously*: must be processing of large volumes of data stream with low latency.

2.4 Discussion of the three elements

The use of the three elements (data stream, big data and real-time) has generated searches that manipulate large volumes of continuous data and expect fast results or at a predetermined time, as seen in Subsection 3.3. Smart cities are an area of research increasingly exploited and whose main characteristic is the use of large volumes of continuous data. However, the use of real-time is still a challenge due to the diversity of data sources. However, real-time, both based on the concept of fast response and temporal constraint, in smart city could generate more efficient results for those who use this type of approach.

ParticipACT Brazil project (Gomes et al., 2016) is an example of a smart city that uses data from two different types of data sources, crowdsensing and utilities, to direct managers in solving urban problems. It is an ongoing project with an initial objective to obtain the floating population of a tourist city through crowdsensing campaigns and data obtained from rubbish collection company. The purpose of the crowdsensing campaigns is to obtain information from the citizens and tourists of the city and the rubbish collection company the variation of the garbage production throughout the year and thus calculate the floating population of the city.

The use of real-time in solving the problem proposed by ParticipACT Brazil project could generate fast results, as the data was received, or in a pre-determined time interval, which makes useful this approach. However, the use of time constraint or fast response is directly related to factors such as user availability and agility in sending the data, as well as the consistency of the data which makes its use a challenge.

3 Research methodology

The development this research work was characterised by adopting a systematic literature review (SLR), based on the procedures suggested by Kitchenham and Charters (2007). Procedures adopted to develop this research were divided into three phases:

- *Planning the review* (Subsection 3.1): it consists the specification of the research questions and the development and validation of the research protocol.
- *Conducting the review* (Subsection 3.2): it consists the identification and selection of papers.
- *Analysis and classification of data* (Subsection 3.3): it consists of the results obtained from the analysis of studies related to the research question.

The following subsections present the details of the review procedure stages.

3.1 Planning the review

Initially, a review protocol has been defined in order to specify methods to be used for its accomplishment (Kitchenham and Charters, 2007). Table 1 presents the SLR protocol defined for this research.

In the first stage, *research questions*, those questions in which the research will be related are defined. The first research question aims to select and to present the articles that have proposals related to data stream, big data and real-time topics. The second research question seeks to analyse and to classify real-time concepts described by the authors. In the *strategy* stage sources and search terms used for the research are determined. In the *selection criteria* stage, inclusion and exclusion criteria of articles are defined. In other words, articles that meet the inclusion criteria are selected and articles that address exclusion criteria are excluded. Lastly, in the *quality selection criteria* stage the quality requirements of the articles are defined.

3.2 Conducting the review

This phase consists of the collection of articles that meet the revision protocol described in the previous section. As can be seen in Table 1, we choose Scopus (Elsevier, 2016) database to carry out this research. Our motivation for this decision was the great acceptance of this database by the scientific community and the large amount of journals and conferences indexed on its base. The search term was created based on the three areas present in the first research question and its writing variations. The three areas are: data stream, big data and real-time.

The articles search procedure began with a general research with only the use of the term without any criteria of inclusion or exclusion, which returned 224 articles. The Figures 1 and 2 present an analysis of the results of the general search. This analysis justifies our choices to the inclusion and exclusion criteria of articles, with regard to the subject areas, since 88.4% of researches are published in the computer science area and 22.3% in the engineering area (Figure 1). Furthermore, in relation to document type, 69.2% of researches are conference paper and 18.3% are article and they are articles and conference papers (Figure 2).

Table 1 Review protocol

Steps		*Description*
Research questions	1	How many articles have processing large amounts of data stream (Big Data Stream) whose answer this processing should be in real time, with time constraints or near real time?
	2	What are real-time concepts presented by the authors?
Strategy	*Resource*: Scorpus (Elsevier, 2016)	
	Search terms: TITLE-ABS-KEY(((“Big Data”) OR (“Big Data Stream*”)) AND ((“Stream* Process*”) OR (“Data Stream* Process*”) OR (“Data Stream*”) OR (“Stream* Data Process*”)) AND ((“Real-Time”) OR (“Real Time”) OR (“Time Constraint”))	
Selection criteria	*Studies included*:	
	1	Articles published without period constraint
	2	Articles published in English
	3	Subject Areas: Computer Science and Engineering
	4	Document Type: Conference paper and Article
	5	Source Type: Conference Proceedings and Journals
	6	Peer Reviewed
	Studies excluded:	
	1	Articles that do not meet the inclusion criteria
	2	In Press
Quality selection criteria	1	Articles that do not address video stream or throughtput evaluation.
	2	Articles that present a clear proposal.
	3	Articles in which real-time, data stream and big data are part of the proposal.

Figure 1 Quantity of articles by subject without inclusion or exclusion criteria (see online version for colours)

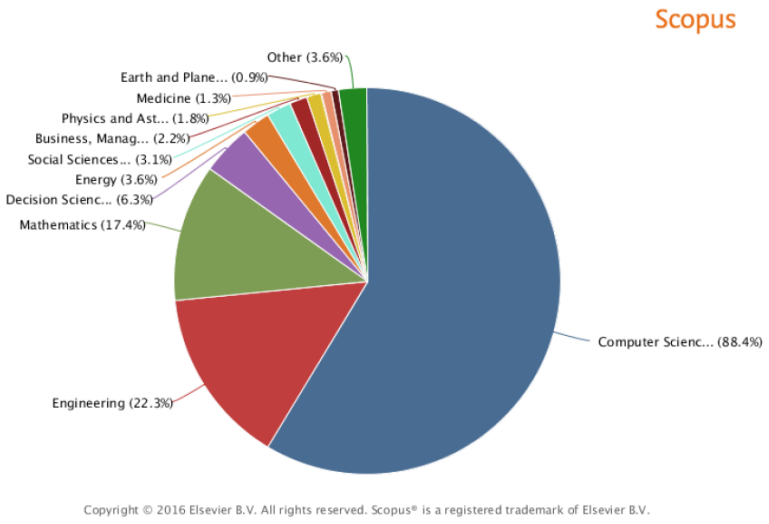
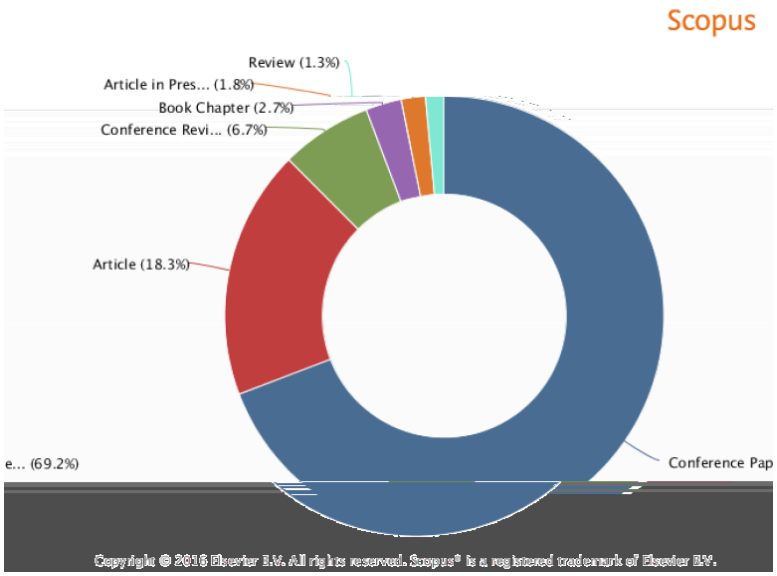


Figure 2 Quantity of articles by document type without inclusion or exclusion criteria (see online version for colours)



Then, the first selection of the articles was conducted by applying the inclusion and exclusion criteria described in the revision protocol. This first selection returned 121 articles. Then we performed a second selection that consisted of reading the titles and abstracts of articles and we obtained as result 84 articles. From these articles, the third selection was carried out through a careful reading of these works, respecting the Quality Selection Criteria. This last selection resulted in 59 articles which answered our first research question.

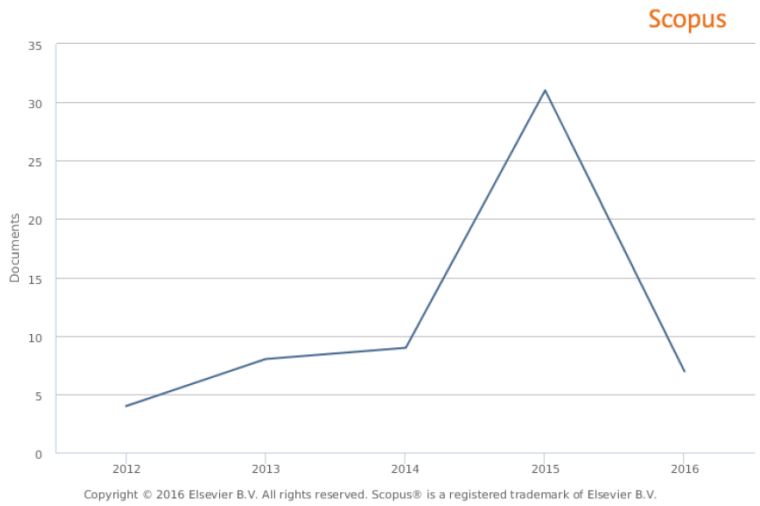
With the third selection, it was possible to analyse the concepts used by the authors to describe the meaning of real-time on their research and answered the second research question. The analysis of the articles is described in more detail in the next subsection.

3.3 Analysis and classification of data

In this section we present the analysis and the classification obtained from the 59 selected articles. Firstly, we show the analysis, through graphical representation, the amount of articles published per year, by country, by type of document and by search area. After that, we provide a classification according to the concept of real-time presented by the authors based on the research proposal of each article in order to group the selected research.

As can be seen in Figure 3, the first publications occurred in 2012 (four articles). The largest number of publications occurred in 2015 with 31 articles, and the year 2016 has seven articles, until the date this research. In the Figure 4 we can see that of ten countries that have most publications on the subject of this survey, the United States holds 17 articles, followed by China with eleven and Germany with six publications. The majority of the articles were published in conferences, totalling 52, as can be seen in Figure 5. On the other hand, the Figure 6 shows that 93.2 % of the articles were published in the Computer Science area.

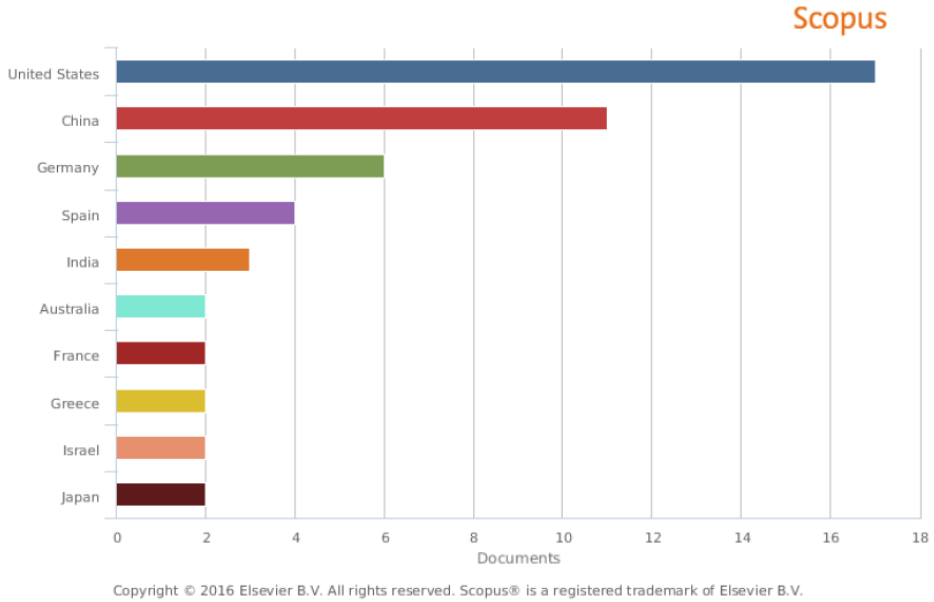
Figure 3 Quantity of articles published by years (see online version for colours)



The Figure 7 shows a tag cloud with the keywords found in the selected articles. In tag cloud, the size of tags represents the number of times these words are cited in keywords of the articles. The words that are part of the search term of this survey are the most cited, which are: 'data', 'processing', 'real-time', 'big' and 'stream'.

Table 2 shows the ten keywords that are most cited in the select articles. The keywords 'data stream', 'big data' and 'real-time', which are part of our search term, appear each one in sixth, first and tenth places, respectively. The first keyword received ten citations, second keyword 49 citations and the third keyword received eight citations.

Figure 4 The ten countries that have most publications (see online version for colours)



In addition, we study each article and find a tendency in the definitions established by the authors regarding the concept of real time and the use of this approach in their research.

Figure 5 Articles published by type (see online version for colours)

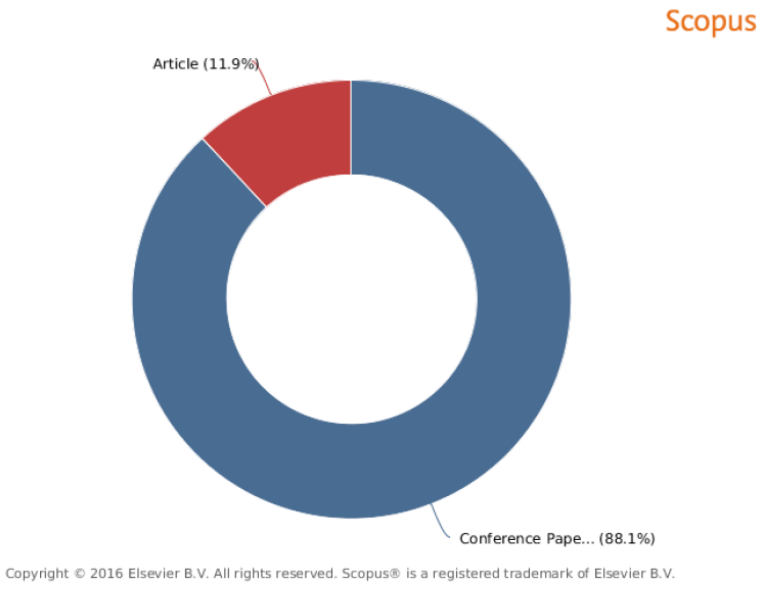
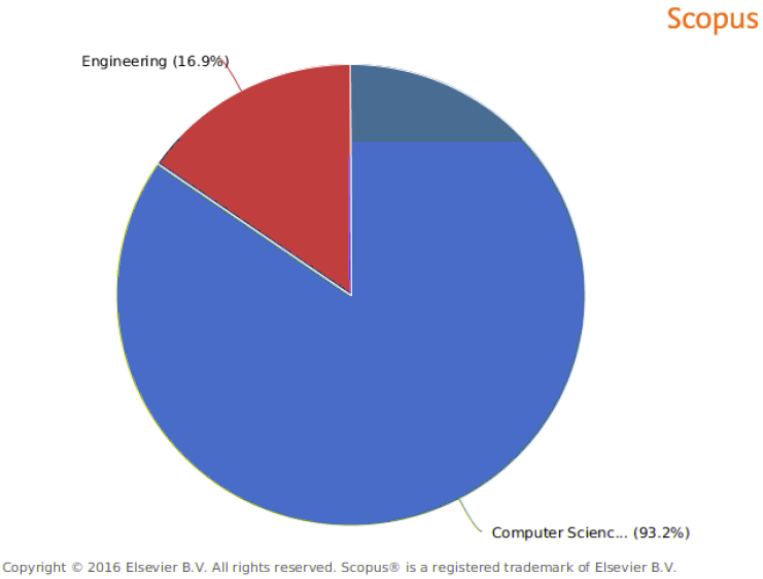


Figure 6 Articles published by subject (see online version for colours)



Therefore, we have developed a classification of the real-time concepts composed of seven categories, which are:

- *It shows the time in which the system must answer*: the articles belonging to this category clearly show the time in which the system should respond. In this case, the understanding of the authors about real-time concept is that to obtain real-time analysis or processing it is necessary to determine the computation time of these executions.
- *Real-time is the guarantee of low latency processing*: the articles included in this category showed that a processing is considered real-time if it is possible to ensure its low latency.
- *Real-time is the immediate response*: for the authors of articles that belong to this category, real time is defined as an execution whose response time is so fast that can be compared with immediate response.
- *Continuously data processing on the fly*: this category considers articles which show that real-time is the continuous processing of the data as they come to be consumed.
- *Data stream is real-time*: in this category, the authors broach that to be data stream processing then, consequently, is dealing with real time, since data stream is processed in real-time.
- *It utilises real time processing tools*: the articles were included in this category because they utilise applications that perform real-time processing of large volume of data, or somehow near from real-time processing. Apache Spark and Apache Storm are the most applications cited in the set of articles analysed. Most of the articles belonging to this category did not present a clear and formal concept of real-time, but showed that their proposals have this approach since they used applications that process data stream in real time.
- *Does not clearly present the concept of real-time*: the articles of this category does not show clearly the concept of real time to its proposal, but they inserted real-time keywords in its proposal.

Figure 7 Tag cloud of the keywords used in the selection articles (see online version for colours)

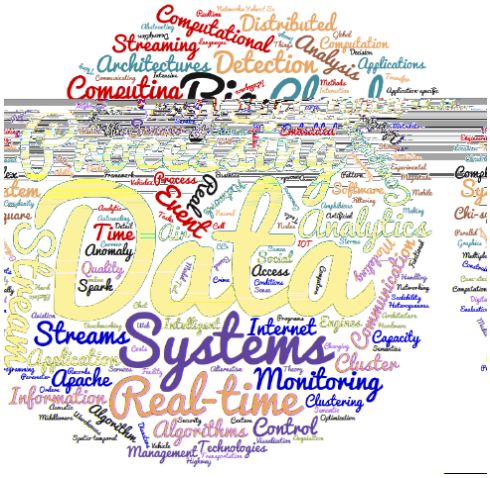


Table 2 The ten most cited keywords

Keywords	*Number of citations*
Big data	49
Data handling	21
Data communication systems	19
Data mining	15
Stream processing	13
Data stream	10
Internet	9
Algorithms	8
Digital storage	8
Real-time	8

Figure 8 Quantity of article for each category

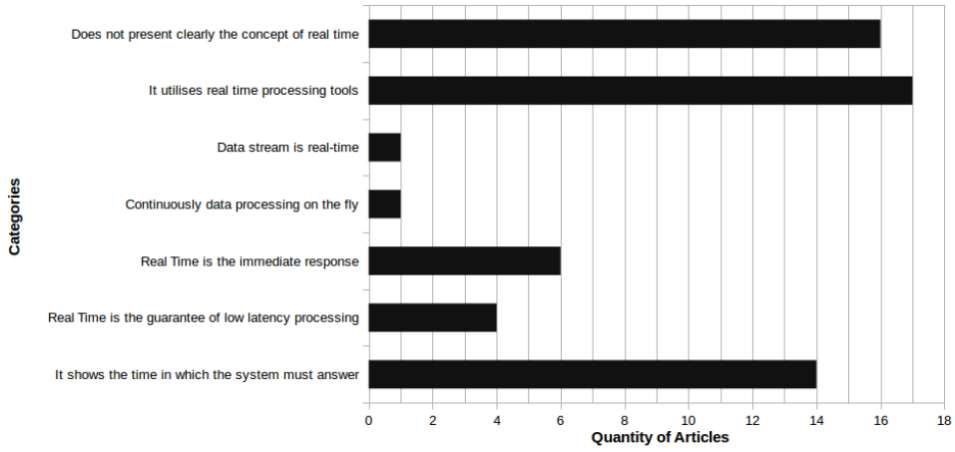


Table 3 presents the seven developed categories, the articles that compose them and the research proposal of each article as well as the scenarios used to carry out the experiments.

Figure 8 presents the number of articles belonging to each category. As can be seen, the categories It utilises real time processing tools, Does not present clearly the concept of real time and It shows the time in which the system must answer have most articles, containing 17, 16 and 14, respectively.

The next section correlates the classification provided with the concepts presented in Section 2.

Table 3 Classification of article

Categories	*Articles*	*Contribution and relevant information*	*Information about experiments*
It shows the time in which the system must answer	Chen and Kang (2015)	It is designed a real-time map-reduce framework which supports a non-pre-emptive periodic task model and a schedulability test based on the earliest deadline first algorithm.	For performance evaluation data analytic benchmarks are adapted to model periodic real-time data analysis tasks.
	Antonić et al. (2016)	It is presented an ecosystem for mobile crowd sensing which relies on cloud-based publish/subscribe middleware to acquire sensor data from mobile devices and perform near real-time processing of big data streams.	The experiments were carried out with real user traces collected during an air quality measurement campaign in the City of Zagreb.
	Kridel et al. (2015)	It is proposed a dynamic real-time modelling based upon automated models, which adapt to real-time customer behaviour via model feedback loops. It used Apache Kafka and Apache Spark technologies.	Mobile marketing campaigns.
	Lee et al. (2015b)	It is developed a data processing framework based on Apache Hadoop and Pig for analysing the number of collected records as well as the amount of energy supply.	The experiment was carried out analysing energy supplied by vehicles.
	Zaharia et al. (2013)	It is proposed a model for distributed streaming computation that enables fast recovery from both faults and stragglers without the overhead of replication.	To evaluate the proposal it was used both several benchmark applications and two real applications: a commercial video distribution monitoring system and a machine learning algorithm for estimating traffic conditions from auto-mobile GPS data.
	Zacheilas et al. (2015)	It is proposed an approach to detect sudden changes in the input rate or latency increases, and determine an appropriate system policy.	The experiment was carried out with a real traffic monitoring application that analyses bus traces from the city of Dublin.
	Lohrmann et al. (2015)	It is proposed a strategy to provide latency guarantees while minimising resource consumption.	The experimental evaluation was carried out with a micro benchmark and real-world social media data.
	Wang et al. (2015)	It is proposed a real-time road traffic monitoring system that uses GPS data collected through wireless communication in probe vehicles.	The experiment was carried out with data collected from taxis during one day.
	Truong et al. (2015)	It is proposed a scalable system for storing and mining massive the real-time vehicle location data.	For the experiments were collected GPS records from vehicles equipped with GPS in Ho Chi Minh city.

Table 3 Classification of article (continued)

Categories	*Articles*	*Contribution and relevant information*	*Information about experiments*
It shows the time in which the system must answer	Lau and Tham (2012)	It is proposed a framework for detecting and responding to real-time anomalies in spatial-temporal contexts.	For the experiments were used real-time GPS information on the National University of Singapore (NUS) internal shuttle buses plying the roads on the campus.
	Basanta-Val et al. (2015)	It is proposed to increase the predictability of the stream processing model, in which Apache Storm is inspired, including real-time capabilities.	For the experiment a testing infrastructure on a local area network was developed.
	Xiao and Aritsugi (2013)	It is designed a model to introduce OLAP functionality for multi-dimensional event stream analysis and reuse schemes of sub-expressions among nested event pattern queries for high throughput of CEP systems.	The experiments were carried out within StreamBase system.
	Ayhan et al. (2013)	It is built a data warehouse that contains summary, historical and detail aviation data to support tactical and strategic decision making.	The implementation was carried out using Aircraft Situation Display to Industry (ASDI) data archiving by Boeing Advanced Air Traffic Management (AATM).
	Verner et al. (2012)	It is presented a framework, scheduler and a performance estimation tool for a compute engine running on a heterogeneous architecture (CPU and GPU) that aims to process vast numbers of data streams under hard real-time constraints.	To evaluate the proposal was used the AES-CBC encryption kernel on thousands of streams with realistic distribution of rates and deadlines.
Real-time is the guarantee of low latency processing	Fernandez et al. (2015)	It is proposed a real-time big data architecture to collect, maintain and analyse massive volumes of images related with automatic glasses detection.	The article did not present experimental results.
	Gaunitz et al. (2015)	It is presented an approach for a distributed information system which will be capable to analyse and process vast amounts of data in real-time.	The system was developed for logistics companies.
	Braik et al. (2016)	It is presented an approach to detect pattern in real-time over large streams of events.	The experiments were carried out based in the Cdiscount requirements.
	Wickramaarachchi et al. (2015)	It is proposed a scalable solution which perform real-time data processing and graph updates to enable low latency graph and analytic on large evolving social networks.	The experiments were carried out using a large Twitter data set.

Table 3 Classification of article (continued)

Categories	*Articles*	*Contribution and relevant information*	*Information about experiments*
Real-time is the immediate response	Tsirakis et al. (2015)	It is presented a tool for monitoring and analysis of opinions about user defined entities thus creating a Reputation Management System.	The tool uses social media data.
	Meng-Meng et al. (2014)	It is proposed a dynamic task scheduling approach for real-time stream processing platforms.	To carry out the experiments were used tweets as input data obtained by Twitter's API.
	Pan et al. (2013)	It is proposed a scalable and elastic streaming solution that analyses mobile broadband statistics and traffic patterns in real-time, and provides information for real-time decision making.	Experiments were performed based on two cases: statistical analysis and traffic prediction.
	Xhafa et al. (2015b)	It is implemented and evaluated the Yahoo!S4 for Big Data Stream processing.	Experiments were carried out using a real source of data generated from real-time updating a large number of international flights from FlightRadar24.
	Peng et al. (2015)	It is proposed a system that implements resource-aware scheduling within Storm.	To evaluate the proposal was used a variety of micro-benchmark Storm topologies and production topologies used by Yahoo!.
	Huang et al. (2015)	It is proposed a general real-time stream recommender system built on Storm from three aspects: system, algorithm and data.	To evaluate the proposal two applications were used: news recommendation (TencentRec News) and E-commerce recommendation (YiXun an e-commerce website).
Continuously data processing on the fly	Chardonnens et al. (2013)	It is presented a case study using Apache Storm to perform real-time integration and trend detection.	For experiments were used Twitter and Bitly streams.
Data stream is real-time	Hu et al. (2015)	It is proposed an algorithm based on grid and density for clustering data streams in a parallel distributed environment.	For parallel cluster analysis of data streams is used the Map-Reduce framework.
It utilises real-time processing tools	Gokalp et al. (2015)	It is proposed a generic and scalable architecture to process a large number of real-time queries. It is used Apache Storm, Apache Kafka, HBase and Zookeeper technologies.	Scenario 1: the system calculates the average of sensor's readings and write the average into HBase. Scenario 2: the system compares the latest readings of two sensors and notify to the user.
	Zhao et al. (2014)	It is implemented a real-time stream processing platform under cloud environments: SpeedStream, based on real-time stream processing model directed graph. It is used Apache Storm and Zookeeper technologies.	In the experiment was processed real data flow in a production environment: the processing of different type of data stream.
	Feng et al. (2015)	It is proposed a memory capacity model to calculate the need heap size for Samza-based applications.	In the experiment was used input data stream that represents LinkedIn production traffic patterns.

Table 3 Classification of article (continued)

Categories	*Articles*	*Contribution and relevant information*	*Information about experiments*
It utilises real-time processing tools	Samosir et al. (2016)	It is proposed a comparison of the streaming platforms: Apache Storm, Apache Spark and Apache Samza technologies.	For the experiment was used a batch dataset acquired from the Monash Institute of Railway Technology (IRT) team.
	Kalashnikov et al. (2015)	It is developed Cerrera system, a cloud-based data stream processing and analysing platform. It is used Apache Zookeeper, Apache Storm and Apache Kafka.	For the experiment was used two use cases: earthquake detection with Twitter and emotion and finances.
	Zang (2015)	It is proposed an approach to dynamically rebalance the resource used in Storm according to real-time data rate, a tool named DBalancer.	For the experiment was measured the effectiveness of DBalancer in CPU and memory utilisation.
	Schaefer and Manoj (2015)	It is proposed policy framework to modify an Apache Storm cluster and improve privacy policies and security gateway.	In the experiment was carried out performance comparisons between the default Storm version and the customised trusted Storm version proposed.
	Sawyer and O'Gwynn (2014)	It is proposed A data pipeline design for delivering real-time ingest and responsive queries using Apache Accumulo.	The experiment focus on web traffic captured from web proxy server log files.
	Zhao et al. (2013)	It is proposed a structure and computing environment needed for analysis of large-scale GPS data which was collected. It is used Apache Kafka, Apache Storm and Spring technologies.	For the experiment data was downloaded from the datatang.com, including more than twenty thousand taxis' GPS location data.
	Maarala et al. (2015)	It is proposed a big data platform for low latency traffic flow analysis based on real-time, high velocity data. It is used Apache Spark, Apache Flume and HDFS.	The analysis takes GPS events as an input, calculates traffic flow indicator values in real-time and stores results to HDFS.
	Xhafa et al. (2015a)	It is proposed to investigate such issue by processing a real life Big Data Stream using a heterogeneous cluster. It is used Yahoo!S4.	For the experiment real data stream received from FlightRadar24 global flight monitoring system was used.
	Eskandari et al. (2016)	It is proposed a hierarchical adaptive scheduling algorithm that employs graph partitioning algorithms in a hierarchical manner. It is used Apache Storm and Zookeeper technologies.	For the experiment was used a benchmark called Test Throughput Topology to have the same data transfer rates and used Top Trending Topics on Twitter, a real-word topology to evaluate the method.
	Saravanan et al. (2013)	It is proposed to localise to use the twitter data for finding out the specific issues/topics that spreads in every location. It is used Apache Storm technology.	For the experiment was used large volumes of big data from real stream of tweets generated by twitter API.

Table 3 Classification of article (continued)

Categories	*Articles*	*Contribution and relevant information*	*Information about experiments*
It utilises real-time processing tools	Mylavarapu et al. (2015)	It is proposed a real-time hybrid intrusion detection system using CC4 neural network and misuse based attack detection using multi-layer perceptron implementation.	The entire simulation was performed inside Apache Storm cluster with ISCX 2012 Intrusion dataset.
	Solaimani et al. (2015)	It is implemented a real-time Chi-square test based anomaly detection framework using Apache Spark. It is used Apache Kafka technology too.	The experiment was carried out in laboratory through monitoring of VMware data.
	Dong et al. (2015)	It is proposed a elderly health monitoring platform which introduces memory-based computation framework Spark to carry out the analysis of the data clustering.	To carry out the experiment was given to user/client a sensor collector to the collected of data.
	Mayhew et al. (2015)	It is developed concepts and technologies as part of the Behavior-Based Access Control (BBAC) to perform accurate calculations of trustworthiness of actors and documents.	The current prototype combines big data batch processing to train classifiers and real-time stream processing to classifier observed behaviours at multiple layers.
Does not present clearly the concept of real-time	Mo and Wang (2012)	It is presented a flexible near real-time high performance and high availability solution with indexes for data.	The test collection was randomly generated document stream.
	Amatriain (2013)	It is presented different approaches to deal with large streams of data in order to extract information for personalising services.	Netflix Prize was used to carry out the experiments.
	Skodzik et al. (2015)	It is presented a system based on the modified Peer-to-Peer network Kad combined with the standardised protocol CoAP to transmit and interpret big amounts of data reassembling data fragments from the UDP packets.	Three different scenarios were used to evaluated the proposal: Dummy UDP packets, using the CoAP protocol and using CoAP block-wise transfer.
	Gibson et al. (2015)	It is presented an European FP7 project that utilises social media to enhance the ability of law enforcement agencies, emergency responders and citizens to react in time of crisis.	Two examples are presented to show how the project works: summarising crisis information from social media postings.
	Golab and Johnson (2014)	It is proposed to review recent research and open problems in data stream warehousing, including motivating applications, system architectures and performance optimisation.	It was not carried out experiments in this article.
	Bouillet et al. (2012)	It is implemented and deployed a call detail record processing application using the IBM InfoSphere Streams middleware.	It was not carried out experiments in this article.

Table 3 Classification of article (continued)

Categories	*Articles*	*Contribution and relevant information*	*Information about experiments*
Does not present clearly the concept of real-time	Qanbari et al. (2015)	It is developed a semantic gateway middleware that delivers automated and on-demand semantic annotations, labels and taxonomies for sensor data acquisition at scale.	The proposed model was evaluated with a real-world health-care data set.
	Schnizler et al. (2015)	It is presented a system architecture integrates multi-faceted sensing and distributed event detection to identify, label and increase confidence in detected incidents.	The experiments were carried out using nation-wide and city-level incident recognition scenarios.
	Nutzel and Zimmermann (2015)	It is described an approach to detect automatically events in real-time by analysing big data streams coming from a social network.	The experiments were carried out using social network Twitter.
	Cao et al. (2014)	It is presented a system that offers both real-time responsiveness as well as interactive outlier exploration in big data streams.	The experiments were carried out based on two scenarios: detect outliers in the STT data recording stock transactions from NYSE and the GMTI data recording information of moving objects.
	Mardani and Giannakis (2015)	It is proposed a real-time sketching scheme that exploits the correlations across data stream to learn a latent subspace.	The experiments were carried out using synthetic data.
	Nguyen and Jung (2014)	It is proposed a method for event detection in real-time, which is implemented using data aggregation technique on a distributed computing environment.	For experiment was used data collections from Twitter.
	Luts (2015)	It is proposed a method for semiparametric regression modeling for data sets that are horizontally partitioned over multiple hosts.	The experiments were carried out using real-life data set that are generated at 415 US airports in real-time.
	Chaolong et al. (2016)	It is employed the real-time graphic visualisation technology to improve the human-computer interaction and the efficiency and accuracy of the visual analysis of massive traffic data.	To carry out the experiments, traffic data of Chongqing was used.
	Riemer et al. (2014)	It is proposed a methodology and a technical approach for the composition and management of real-time processing pipelines.	Experiments were carried out based two use cases: a logistic use case and a personal monitoring use case.
	Cortés et al. (2015)	It is presented a preliminary analysis of the data flows generated by a typical IoT application.	To carried out the analysis it was used public workout traces extracted from the Endomondo sports tracker server.

4 Discussions

This section presents a comparative analysis of the concepts presented in the Section 2 with the classification suggested in Subsection 3.3. In carrying out this analysis it is possible to divide the classifications into three major groups. The first group refers to the category *It utilises real time processing tools*. In the second group are inserted the categories *Real-time is the immediate response*, *Real-time is the guarantee of low latency processing*, *Continuously data processing on the fly* and *Data stream is real-time*. Finally, the third group is the category *It shows the time in which the system must answer*.

Tools such as Apache Spark and Apache Storm are very used for the data processing stream since they offer low latency. Apache Spark is a specialised tool to perform data analysis fast for running programs (Shoro and Soomro, 2015). Spark supports in-memory computing, which enables the data query to be faster than disk-based mechanisms. On the other hand, Apache Storm is a real-time stream processing tool that ensures that the newly generated data processing with very low latency (Hussain and Rahim, 2015). According to the authors, real-time processing means that data is processed with high low latency.

Considering this, we can say that the articles belonging to the first group proposed the use of tools that ensure low latency and real-time data processing. This conceptualisation causes the first and second groups converge regarding the real time concept. The categories belonging to the second group basically refer to fast response time, low latency and fast and continuous data processing. The first and second groups, then, meets the requirements proposed for Stonebraker and Zdonik (2005), to real-time processing of data stream, shown above.

On the other hand, Stankovic et al. (1999) discuss nine errors concepts related to real-time. The misconceptions most common are: *real-time means speed* and *real-time computing is equivalent to fast computing*. Fast computing aims to minimise the average response time, however does not guarantee the temporal restriction. To Safaei (2016) velocity is necessary but not enough because a real-time system needs mechanisms as scheduling and real-time feedback control to satisfy the time restrictions.

The third group presents concepts defended by Stankovic and Ramamritham (1990), Stankovic et al. (1999) and Safaei (2016), as these articles are concerned with the processing velocity and low latency, however they differ from other groups because they determine the time at which the system must respond. In other words, the articles present in its proposal and implementation the time constraint that must be determined, since this is a real-time processing system.

According to what was discussed in this section, we can note the need to perform classification of real-time concepts. With this classification would be possible to analyse, effectively, the approaches used to conceptualise real-time.

5 Conclusions and future works

This research work carried out a systematic review with the aim of presenting studies related to data stream, big data and real-time. The research effort was to understand how these approaches are being used together and especially how the real-time concepts are been used.

After the development and execution of a revision protocol, we obtained 59 articles, of which conducted an analysis of publications and classification of articles according to the real-time concepts used. We divided the articles into seven categories, which were: It shows the time in which the system must answer, real-time is the immediate response, real-time is the guarantee of low latency processing, Continuously data processing on the fly, data stream is real-time, It utilises real time processing tools and Does not clearly present the concept of real-time. Of the selected articles, 16 did not present any definition of the real-time term, only its use in the proposal. Fourteen presented the concept of real-time and have proposed time constraint in system response and 29 only presented the concept of real-time as faster processing and low latency as well as have used tools that have the same concept.

We note that there is a tendency in the real-time utilisation and data stream in big data environments, in order to meet the characteristics of big data known as 5Vs. However, there is a lot divergence in the literature regarding the real-time concept. New concepts argue that real-time is fast response and low latency. On the other hand, classical concepts show that to obtain real-time is necessary to determine a response time, that is time constraint applied to the system. In face of this scenario, we can realise that the authors used the concept of real-time that best adapted to your proposal.

The main challenges encountered in the accomplishment of this research was the development of the review protocol and effective selection of articles. The development of the protocol review became a challenge because it was necessary to establish of efficient and correct manner the search terms and the criteria for inclusion and exclusion of articles such that systematic review generates good results. On the other hand, the selection of articles has become a hard process since it was necessary to select articles from a substantial amount of articles (i.e., 224).

As future works we pretend to propose a formal classification of the time constraints applications, based upon different requirements from different classes of applications. This proposal will target a mapping approach which will gather a specific threshold time constraints to a specific type of applications.

References

Amatriain, X. (2013) 'Big & personal: data and models behind netflix recommendations', *Proceedings of the 2nd International Workshop on Big Data, Streams and Heterogeneous Source Mining Algorithms, Systems, Programming Models and Applications - BigMine '13*, pp.1–6.

Antonić, A., Marjanović, M., Pripužić, K., and Podnar Žarko, I. (2016) 'A mobile crowdsensing ecosystem enabled by CUPUS: Cloud-based publish/subscribe middleware for the internet of things', *Future Generation Computer Systems*, Vol. 56, pp.607–622.

Ayhan, S., Pesce, J., Comitz, P., Sweet, D., Bliesner, S. and Gerberick, G. (2013) 'Predictive analytics with aviation big data', *Integrated Communications, Navigation and Surveillance Conference*, ICNS.

Babcock, B., Babu, S., Datar, M., Motwani, R. and Widom, J. (2002) 'Models and issues in data stream systems', in *Proceedings of the Twenty-first ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, pp.1–16.

Basanta-Val, P.N., Fernández-García, A.J.W. and Audsley, N.C. (2015) 'Improving the predictability of distributed stream processors', *Future Generation Computer Systems*, Vol. 52, pp.22–36.

Bouillet, E., Kothari, R., Kumar, V., Mignet, L., Nathan, S., Ranganathan, A., Turaga, D.S., Udrea, O. and Verscheure, O. (2012) *Experience Report: Processing 6 Billion CDRs/day - From Research to Production*, pp.264–267.

Braik,W., Morandat, F., Falleri, J-R. and Blanc, X. (2016) 'Real time streaming pattern detection for eCommerce', in *Proceedings of the 31st Annual ACM Symposium on Applied Computing - SAC '16*, pp.916–922.

Cao, L., Wang, Q. and Rundensteiner, E.A. (2014) 'Interactive outlier exploration in big data streams. *Proc. VLDB Endow*., pp.1621–1624.

Chaolong, J., Hanning, W. and Lili, W. (2016) 'Research on visualization of multi-dimensional real-time traffic data stream based on cloud computing', *Procedia Engineering*, Vol. 137, pp.709–718.

Chardonnens, T., Cudre-Mauroux, P., Grund, M. and Perroud, B. (2013) 'Big data analytics on high Velocity streams: a case study', *2013 IEEE International Conference on Big Data*, pp.784–787.

Chen, L. and Kang, K.D. (2015) 'A framework for real-time information derivation from big sensor data', *Proceedings – 2015 IEEE 17th International Conference on High Performance Computing and Communications, 2015 IEEE 7th International Symposium on Cyberspace Safety and Security and 2015 IEEE 12th International Conference on Embedded Software and Systems, H*, pp.1020–1026.

Cortés, R., Bonnaire, X., Marin, O. and Sens, P. (2015) 'Stream processing of healthcare sensor data: studying user traces to identify challenges from a big data perspective', *Procedia Computer Science*, Vol. 52, pp.1004–1009.

Demchenko, Y., Grosso, P., De Laat, C. and Membrey, P. (2013) 'Addressing big data issues in scientific data infrastructure', *Proceedings of the 2013 International Conference on Collaboration Technologies and Systems, CTS 2013*, pp.48–55.

Dong, M., Huang, X., Bi, S., Zeng, X., Pang, N., Liu, H. and Tang, X. (2015) 'The elderly health monitoring platform based on spark', *2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, pp.514–519.

Elsevier (2016) *Scopus* [online] http://www.scopus.com (accessed August 2016).

Eskandari, L., Huang, Z. and Eyers, D. (2016) 'P-Scheduler: adaptive hierarchical scheduling in apache storm Leila', in *Proceedings of the Australasian Computer Science Week Multiconference on - ACSW '16*, pp.1–10.

Feng, T., Zhuang, Z., Pan, Y. and Ramachandra, H. (2015) 'A memory capacity model for high performing data-filtering applications in Samza framework', *Proceedings – 2015 IEEE International Conference on Big Data, IEEE Big Data 2015*, pp.2600–2605.

Fernandez, A., Casado, R. and Usamentiaga, R. (2015) 'A real-time big data architecture for glasses detection using computer vision techniques', *Proceedings - 2015 International Conference on Future Internet of Things and Cloud, FiCloud 2015 and 2015 International Conference on Open and Big Data,OBD2015*, pp.591–596.

Gaunitz, B., Roth, M. and Franczyk, B. (2015) 'Dynamic and scalable real-time analytics in logistics – combining apache storm with complex event processing for enabling new business models in logistics', *Proceedings of the 10th International Conference on Evaluation of Novel Approaches to Software Engineering*, pp.289–294.

Gibson, H., Andrews, S., Domdouzis, K., Hirsch, L. and Akhgar, B. (2015) 'Combining big social media data and FCA for crisis response', *Proceedings – 2014 IEEE/ACM 7th International Conference on Utility and Cloud Computing, UCC 2014*, pp.690–695.

Gokalp, M.O., Kocyigit, A. and Eren, P.E. (2015) 'A cloud based architecture for distributed real time processing of continuous queries', *Proceedings – 41st Euromicro Conference on Software Engineering and Advanced Applications, SEAA 2015*, pp.459–462.

Golab, L. and Johnson, T. (2014) 'Data stream warehousing', *Proceedings – International Conference on Data Engineering*, pp.1290–1293.

Gomes, E., Dantas, M.A.R., De Macedo, D.D.J., De Rolt, C., Brocardo, M.L. and Foschini, L. (2016) 'Towards an infrastructure to support big data for a smart city project', *Proceedings - 25th IEEE International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises, WETICE 2016*, pp.107–112.

Hu, W., Cheng, M., Wu, G. and Wu, L. (2015) 'Research on parallel data stream clustering algorithm based on grid and density', *Proceedings – 2015 International Conference on Computer Science and Mechanical Automation, CSMA 2015*, pp.70–75.

Huang, Y., Cui, B., Zhang, W., Jiang, J. and Xu, Y. (2015) 'TencentRec: real-time stream recommendation in practice Yanxiang', *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data - SIGMOD '15*, pp.227–238.

Hussain, I.M. and Rahim, S.T. (2015) 'Big Data analysis: apache storm perspective', *International Journal of Computer Trends and Technology*, Vol. 19, No. 1, pp.9–14.

IBM (2016) *Bringing Big Data to the Enterprise* [online] http://www-01.ibm.com/software/in/data/bigdata/ (accessed September 2016).

Inacio, E.C. and Dantas, M.A.R. (2014) 'A survey into performance and energy efficiency in HPC, cloud and big data environments', *International Journal of Networking and Virtual Organisations*, Vol. 14, No. 4, p.299.

Kalashnikov, D., Bartashev, A., Mitropolskaya, A., Klimov, E. and Gusarova, N. (2015) 'Cerrera: in-stream data analytics cloud platform', *2015 3rd International Conference on Digital Information, Networking, and Wireless Communications, DINWC 2015*, pp.170–175.

Kitchenham, B. and Charters, S. (2007) 'Guidelines for performing systematic literature reviews in software engineering', *Engineering*, Vol. 2, p.1051.

Kridel, D., Dolk, D. and Castillo, D. (2015) 'Adaptive modeling for real time analytics: the case of 'big data' in mobile advertising', *Proceedings of the Annual Hawaii International Conference on System Sciences*, March, pp.887–896.

Kuo, S.M., Lee, B.H. and Tian, W. (2013) *Real-Time Digital Signal Processing: Fundamentals, Implementations and Applications*. John Wiley & Sons.

Lau, J.K.S. and Tham, C.K. (2012) 'Hidden Markov Models for abnormal event processing in transportation data streams', *Proceedings of the International Conference on Parallel and Distributed Systems - ICPADS*, pp.816–821.

Lee, J., Park, G-L. and Park, C.J. (2015a) 'Charging facility monitoring stream analysis based on hadoop for smart grids', *International Journal of Software Engineering and Its Applications*, Vol. 9, No. 11, pp.153–162.

Lee, Y-j., Lee, M., Lee, M-y., Hur, S.J. and Min, O. (2015b) 'Design of a scalable data stream channel for big data processing', *2015 17th International Conference on Advanced Communication Technology (ICACT)*, pp.3–6.

Liu, J. (2000) *Real-Time Systems*, Prentice Hall.

Lohrmann, B., Janacik, P. and Kao, O. (2015) 'Elastic stream processing with latency guarantees', *Proceedings – International Conference on Distributed Computing Systems*, July, pp.399–410.

Luts, J. (2015) 'Real-time semiparametric regression for distributed data sets', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 27, No. 2, pp.545–557.

Maarala, A.I., Rautiainen, M., Salmi, M., Pirttikangas, S. and Riekki, J. (2015) 'Low latency analytics for streaming traffic data with Apache Spark', *Proceedings – 2015 IEEE International Conference on Big Data, IEEE Big Data 2015*, pp.2855–2858.

Mardani, M. and Giannakis, G.B. (2015) 'Online sketching for big data subspace learning', *2015 23rd European Signal Processing Conference, EUSIPCO 2015*, pp.2511–2515.

Mayhew, M., Atighetchi, M., Adler, A. and Greenstadt, R. (2015) 'Use of machine learning in big data analytics for insider threat detection', *MILCOM 2015 – 2015 IEEE Military Communications Conference*, pp.915–922.

Meng-Meng, C., Chuang, Z., Zhao, L. and Ke-Fu, X. (2014) 'A Task scheduling approach for real-time stream processing', *2014 International Conference on Cloud Computing and Big Data*, pp.160–167.

Mo, X. and Wang, H. (2012) 'Asynchronous Index Strategy for high performance real-time big data stream storage', *Proceedings – 2012 3rd IEEE International Conference on Network Infrastructure and Digital Content, IC-NIDC 2012*, pp.232–236.

Mylavarapu, G., Thomas, J. and Ashwin Kumar, T.K. (2015) 'Real-time hybrid intrusion detection system using apache storm', *Proceedings - 2015 IEEE 17th International Conference on High Performance Computing and Communications, 2015 IEEE 7th International Symposium on Cyberspace Safety and Security and 2015 IEEE 12th International Conference on Embedded Software and Systems*, H, pp.1436–1441.

Nguyen, D.T. and Jung, J.J. (2014) 'Real-time event detection on social data stream', *Mobile Networks and Applications*, Vol. 20, No. 4, pp.475–486.

Nutzel, J. and Zimmermann, F. (2015) 'Improved burst based real-time event detection using adaptive reference Corpora', *2015 3rd International Conference on Future Internet of Things and Cloud*, pp.512–518.

Pan, L., Qian, J., He, C., Fan, W., He, C. and Yang, F. (2013) 'NIM: scalable distributed stream process system on mobile network data', *Proceedings – IEEE 13th International Conference on Data Mining Workshops, ICDMW 2013*, No. 1, pp.1101–1104.

Peng, B., Hosseini, M., Hong, Z., Farivar, R. and Campbell, R. (2015) 'R-storm: resource-aware scheduling in storm', in *Proceedings of the 16th Annual Middleware Conference on – Middleware '15*, pp.149–161.

Pepper, R. and John, G. (2016) *The Internet of Everything: How the Network Unleashes the Benefits of Big Data* [online] http://blogs.cisco.com/wp-content/uploads/GITR-2014-Cisco-Chapter.pdf (acessed September 2016).

Philip Chen, C.L. and Zhang, C.Y. (2014) 'Data-intensive applications, challenges, techniques and technologies: a survey on big data', *Information Sciences*, Vol. 275, pp.314–347.

Plentz, P.D.M., Montez, C. and Oliveira, R.S.d. (2012) 'Deadline missing prediction through the use of milestones', *Computing and Informatics*, Vol. 30, No. 4, pp.657–679.

Qanbari, S., Behinaein, N., Rahimzadeh, R. and Dustdar, S. (2015) 'Gatica: linked sensed data enrichment and analytics middleware for IoT gateways', *Proceedings – 2015 International Conference on Future Internet of Things and Cloud, FiCloud 2015 and 2015 International Conference on Open and Big Data, OBD 2015*, pp.38–43.

Riemer, D., Stojanovic, L. and Stojanovic, N. (2014) 'SEPP: semantics-based management of fast data streams', *Proceedings – IEEE 7th International Conference on Service-Oriented Computing and Applications, SOCA 2014*, pp.113–118.

Safaei, A.A. (2016) 'Real-time processing of streaming big data', *Real-Time Systems*.

Sagiroglu, S. and Sinanc, D. (2013) 'Big data: a review', *2013 International Conference on Collaboration Technologies and Systems (CTS)*, pp.42–47.

Samosir, J., Indrawan-Santiago, M. and Haghighi, P.D. (2016) 'An evaluation of data stream processing systems for data driven applications', *Procedia Computer Science*, Vol. 80, pp.439–449.

Saravanan, M., Sundar, D. and Kumaresh, V.S. (2013) 'Probing of geospatial stream data to report disorientation', *2013 IEEE Recent Advances in Intelligent Computational Systems, RAICS 2013*, pp.227–232.

Sawyer, S.M. and O'Gwynn, B.D. (2014) 'Evaluating accumulo performance for a scalable cyber data processing pipeline', *2014 IEEE High Performance Extreme Computing Conference*, HPEC 2014.

Schaefer, C. and Manoj, P.M. (2015) 'Enabling privacy mechanisms in apache storm', *Proceedings – 2015 IEEE International Congress on Big Data, BigData Congress 2015*, pp.102–109.

Schnizler, F., Liebig, T., Marmor, S., Souto, G., Bothe, S. and Stange, H. (2015) 'Heterogeneous stream processing for disaster detection and alarming', *Proceedings – 2014 IEEE International Conference on Big Data, IEEE Big Data 2014*, pp.914–923.

Shahrivari, S. (2014) 'Beyond batch processing: towards real-time and streaming big data', *Computers*, Vol. 3, No. 4, pp.117–129.

Shaw, A.C. (1989) 'Reasoning about time in higher-level language software', *IEEE Transactions on Software Engineering*, Vol. 15, No. 7, pp.875–889.

Shoro, A.G. and Soomro, T.R. (2015) 'Big data analysis: apache spark perspective', *Global Journal of Computer Science and Technology*, Vol. 15, No. 1, pp.7–14.

Skodzik, J., Danielis, P., Altmann, V., Konieczek, B., Schweissguth, E.B., Golatowski, F. and Timmermann, D. (2015) 'CoHaRT: a P2P-based deterministic transmission of large data amounts using CoAP', *Proceedings of the IEEE International Conference on Industrial Technology*, June 2015, pp.1851–1856.

Smith, S.W. (1999) *Digital Signal Processing*, 2nd ed., California Technical Publishing.

Solaimani, M., Iftekhar, M., Khan, L. and Thuraisingham, B. (2015) 'Statistical technique for online anomaly detection using Spark over heterogeneous data from multi-source VMware performance data', *Proceedings–- 2014 IEEE International Conference on Big Data, IEEE Big Data 2014*, pp.1086–1094.

Stankovic, J., Spuri, M., Ramamritham, K. and Buttazzo, G. (2012) 'Deadline scheduling for real-time systems: EDF and related algorithms', *The Springer International Series in Engineering and Computer Science*, Springer US.

Stankovic, J.A. and Ramamritham, K. (1990) 'What is predictability for real-time systems?', *Real-Time Systems*, Vol. 2, No. 4, pp.247–254.

Stankovic, J.A., Son, S.H. and Hansson, J. (1999) 'Misconceptions about real-time databases', *Computer*, June, Vol. 32, No. 6, pp.29–36.

Stonebraker, M. and Zdonik, S. (2005) 'The 8 requirements of real-time stream processing', *ACM SIGMOD Record*, Vol. 34, No. 4, pp.42–47.

Truong, H-L., Bui, D-K. and Tran, V-T. (2015) 'GPSInsights: towards an efficient framework for storing and mining massive vehicle location data', *Proceedings of the Sixth International Symposium on Information and Communication Technology – SoICT 2015*, pp.1–7.

Tsirakis, N., Poulopoulos, V., Tsantilas, P. and Varlamis, I. (2015) 'A platform for real-time opinion mining from social media and news streams', *2015 IEEE Trustcom/BigDataSE/ISPA*, pp.223–228.

Verner, U., Schuster, A., Silberstein, M. and Mendelson, A. (2012) Scheduling processing of real-time data streams on heterogeneous multi-GPU systems', *Proceedings of the 5th Annual International Systems and Storage Conference*, No. 2 pp.8:1–8:12.

Wang, F., Hu, L., Zhou, D., Sun, R., Hu, J. and Zhao, K. (2015) 'Estimating online vacancies in real-time road traffic monitoring with traffic sensor data stream', *Ad Hoc Networks*, Vol. 35, pp.3–13.

Wickramaarachchi, C., Kumbhare, A., Frincu, M., Chelmis, C. and Prasanna, V.K. (2015) 'Real-time analytics for fast evolving social graphs', *Proceedings – 2015 IEEE/ACM15th International Symposium on Cluster, Cloud, and Grid Computing, CCGrid 2015*, pp.829–834.

Xhafa, F., Naranjo, V. and Caball, S. (2015b) 'Processing and analytics of big data streams with Yahoo!S4', *Proceedings – International Conference on Advanced Information Networking and Applications, AINA*, April, pp.263–270.

Xhafa, F., Naranjo, V., Barolli, L. and Takizawa, M. (2015a) 'On streaming consistency of big data stream processing in heterogenous clutsers', *2015 18th International Conference on Network-Based Information Systems*, pp.476–482.

Xiao, F. and Aritsugi, M. (2013) 'Nested pattern queries processing optimization over multi-dimensional event streams', *Proceedings – International Computer Software and Applications Conference*, pp.74–83.

Zacheilas, N., Kalogeraki, V., Zygouras, N., Panagiotou, N. and Gunopulos, D. (2015) 'Elastic complex event processing exploiting prediction', *Proceedings – 2015 IEEE International Conference on Big Data, IEEE Big Data 2015*, pp.213–222.

Zaharia, M., Das, T., Li, H., Hunter, T., Shenker, S. and Stoica, I. (2013) 'Discretized streams: fault-tolerant streaming computation at scale', *Proceedings of the Twenty-Fourth ACM Symposium on Operating Systems Principles – SOSP '13*, Vol. 1, pp.423–438.

Zang, Z. (2015) 'DBalancer: a tool for dynamic changing of workers number in storm', *Iccsnt*, pp.142–145.

Zhao, J., Sun, Z. and Liao, Q. (2013) 'Implementation of K-means based on improved storm model', *International Conference on Communication Technology Proceedings, ICCT*, pp.728–732.

Zhao, L., Chuang, Z., Ke-Fu, X. and Meng-Meng, C. (2014) 'A computing model for real-time stream processing', *2014 International Conference on Cloud Computing and Big Data*, pp.134–137.