ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

A Review on Processing Big Data

Dr. Shoban Babu Sriramoju

Associate Professor, Dept. of Computer Science and Engineering, Varadha Reddy Engineering College, Warangal, India

Abstract- Big Data has become a known buzzword to public at large which represents huge amount of data in terabytes. Processing such huge data is not possible with conventional environments. Therefore new environments came into existence with data centers and distributed file systems. Hadoop is a well known framework for processing big data using its new programming model known as MapReduce. However, later on many improvements came into existence with the help of other frameworks such as Sailfish, Haloop and AROM besides many other frameworks. In this paper we focused on the three important frameworks such as Sailfish, Haloop and AROM in terms of their performance and characteristics such as Velocity, Volume and Variety. This paper provides insights into big data processing that can help reader to have fundamental knowledge about Big Data, processing it and advantages derived from big data.

Keywords – Big data, Hadoop, Haloop, big data processing

I.INTRODUCTION

Big Data refers to huge amount of data that is generally in peta bytes. Big Data is a buzzword that is associated with volumes of data that cannot be processed in traditional environments [12]. Big Data and its processing are attracting lot of attention recently. Hadoop is open source software that makes use of MapReduce for distributed processing. It has attracted worldwide attention for processing big data in the real world [1]. Distributed parallel processing is done by Hadoop and over few years it has become a reliable system for processing big data. Big data processing should have certain characteristics such as volume, velocity and variety. Hadoop is able to handle volume and variety while real time systems need the velocity as well. The challenges in big data processing in real time include handling of streams that come with certain velocity, parallel processing of data, and even correlation [2]. Big data can transform economy in business, government and other aspects. It has its impact on the society [3], [10], [16]. From big data real value of businesses can be obtained. However, right kinds of tools are required in order to make the exercise fruitful [4]. Strategy to handle big data in organizations will become part of the ICT strategies of the organizations [5]. Big data processing has certain phases that include data acquisition and recording, information extraction and cleaning, data representation, aggregation and integration, data modeling, analysis and query processing, and interpretation. The challenges in big data processing include heterogeneity and incompleteness, scale, timeliness, privacy and human collaboration [6]. Effective use of big data can result in financial performance. Exploiting data is required in order to prioritize business activities [7]. Bit data problems are encountered by big enterprises [8]. The frameworks that process big data follow "NoSQL" principle [9]. Big data does mean different for different people. IBM's survey concludes what is Big Data is for people. It is a great scope of information, new kinds of data and analysis, real time information, data influx from new technologies, non traditional forms of media, large volumes of data, the latest buzzword, and social media data [10]. There are three Vs that characterize the buzzword big data [11]. They are velocity, volume and variety. Big data is measured in terabytes as shown in figure 1.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

Fig. 1 – Illustrates how big data is measured [11]

When big data is transformed into big value, organizations can witness growth in various dimensions faster [13], [15]. MapReduce is the new programming model that is used for generating and processing huge data known as big data [14]. Working with big data is made simple with application frameworks such as Twister, Haloop, Spark, Apache Hama, Boom, Twitter Storm, and GraphLab [17]. For storing big data Solid State Drives (SSDs) are used that have NAND flash memory as they provide high speed performance and can ensure ROI [18]. Big data processing with cutting edge technologies can save companies and government lot of money [19]. The results of big data processing can be used in various sectors such as financial services, education, health, agriculture, and so on [20].

Ghit, Losup and Epema [21] studied big data processing with MapReduce programming model. They optimized the processing of MapReduce further to process volumes of data more efficiently. Suzumura [22] studied processing of big data in large networks. They have worked on various applications pertaining to big data processing with bench mark datasets. Ji *et al*. [23] explored processing of big data in cloud computing environments. Their experiments are on semi-structured cloud storage, non-structural storage, and distributed file system. They found challenges in big data management, computation and analysis, and security. Bu, Howe, and Ernst [24] explored Haloop which is a variant of Hadoop that uses MapReduce programming paradigm for big data processing. Their results revealed that Haloop is able to reduce query runtime. Dawei and Lei [25] explored data processing technologies and data analysis on big data. Liu [26] studied infrastructure required for processing big data. They discussed emerging infrastructures including GPUs (Graphics Processing Units). McGowan [27] focused on various techniques such as PROC SQL, SAS macros, indexes and data step. Rao, Ramakrishnan, and Silberstien [28] explored Sailfish which is a framework for processing big data. Sailfish provides features like auto-tuning. Tran, Skhiri, Lesuisse, and Zimanyi [29] explored a framework known as AROM that can process big data using functional programming and data flow graphs. Maheswaran, Asteris and Papailiopoulos [30] focused on developing erasure codes for big data. They opined that Reed-Solomon codes that

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

are existing incur high repair cost and developed new erasure codes. The remainder of this paper explores some of the important papers in the literature on big data.

II.RELATED WORKS

Processing Big Data with Haloop

Bu, Howe, and Ernst [24] proposed a modified variant of Hadoop that can process data iteratively on large clusters. It extends MapReduce and also provides various capabilities to it including caching mechanisms, loop aware task scheduler and other features. As real time applications need to process huge amount of data in terms of data mining and data analysis Hadoop came into existence. In Hadoop the main programming model is known as MapReduce which is suitable for processing big data. Hadoop is a distributed file system that supports processing huge amount of data in terabytes or more in distributed environments such as cloud computing. As MapReduce is already a scalable and efficient programming model that is improved further. Another tool that has been focused is dryad which is also a popular platform for processing big data. MapReducing programming model is already being used by many companies to process huge data. They include Yahoo, Facebook, Google and so on. Hadoop is an open source MapReduce framework that has been improved and presented as Haloop. The architecture of Haloop includes loop aware task scheduler, and caching mechanisms besides other common requirements as there in Hadoop [24]. The architecture of Haloop is as shown in figure 2.

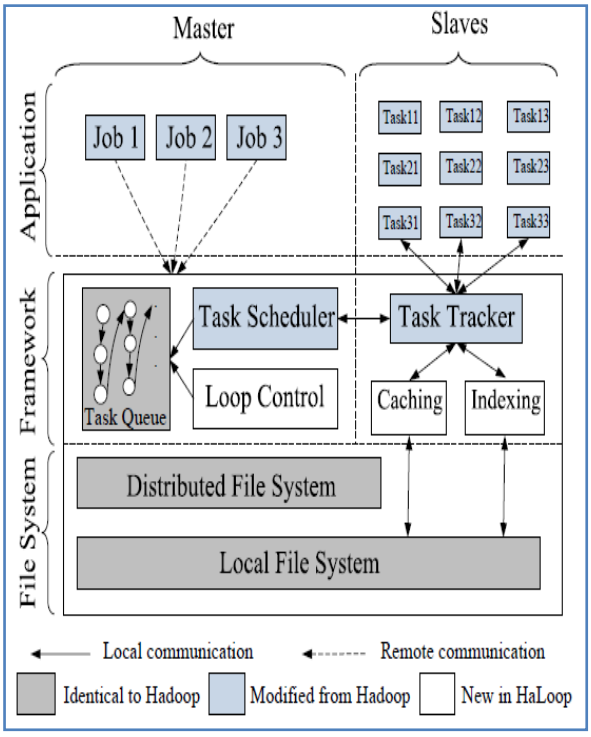

Fig. 2 – Architectural overview of Haloop [24]

As can be viewed in figure 2, the Haloop architecture accommodates many components. Broadly it has three layers such as file system, framework and application. In the file system layer there are two file systems. They are local file system and distributed file system. The local file system takes care of local storage while the distributed file system takes care of storage in multiple machines in order to manage big data processing. In the framework layer task tracker, task scheduler is important components. The task tracker is able to communicate with local file system that makes use of indexing and caching features in order to improve the processing performance. The task scheduler is different from

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

Hadoop here as it is supported by a loop control. It does mean that the task scheduler is loop aware for high performance. Task queue is used to maintain queue of tasks for processing efficiently [24]. Caching is very important in the framework which reduces number of hits to the file system. Caching and indexing are the two important features used by task tracker in order to show high performance of processing big data. The task scheduler is in master node while the task tracker is slave node. The master node takes jobs and gives to slave nodes. The slave nodes process the data and give result back to master node. This way data is processed in parallel to support big data. In the application layer, the master node manages jobs while the slaves manage tasks. Actually the jobs are divided into tasks and the tasks are performed by slave nodes. The master nodes only delegate the jobs to slave nodes in different ways. For instance the master can invoke slaves in either sequential or parallel fashion or it may use combination of both based on the workload. The master node communicates with the framework in order to get jobs done. With respect to iterative work there is a fundamental difference between Hadoop and Haloop that is Haloop is loop aware while the Hadoop is not [24]. This fact is visualized in figure 3.

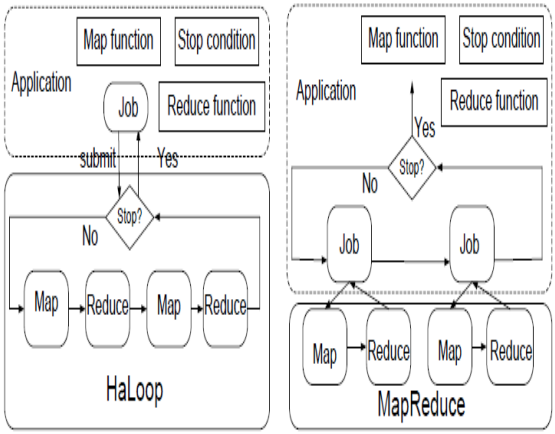

Fig. 3 – Difference between Hadoop and Haloop in iterative processing [24]

As can be seen in figure 3, it is evident that the Hadoop is not loop aware. It continues iterative work until jobs are completed while the Haloop is loop aware that knows how many times the loop has to be iterated. Other main difference is that the Haloop has caching mechanisms that help in improving speed of big data processing further. It focuses on reducer input cache, and reducer output cache for both caching and indexing. Haloop also has mapped input cache in order to map input data and tasks. PageRank is used by both frameworks [24].

III.EXPERIMENTS AND FUTURE WORK

Experiments are made to know the main differences between Hadoop and Haloop in terms of iterations, PageRank, and descendent query performance. The performance of the PageRank mechanisms of both is presented in figure 4.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

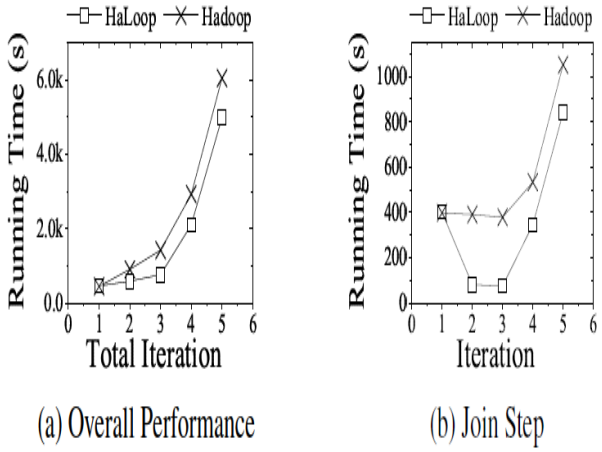

Fig. 4 – Overall performance and Join Step performance of Hadoop and Haloop [24]

As seen in figure 4, iterations and running time of both Hadoop and Haloop with respect to PageRank algorithm and performance is presented. Horizontal axis shows number of iterations while the vertical axis shows the time taken in seconds. The results revealed that Haloop's performance is better that that of Hadoop [24]. With respect to descendent query performance of both Hadoop and Haloop, figure 5 visualized it.

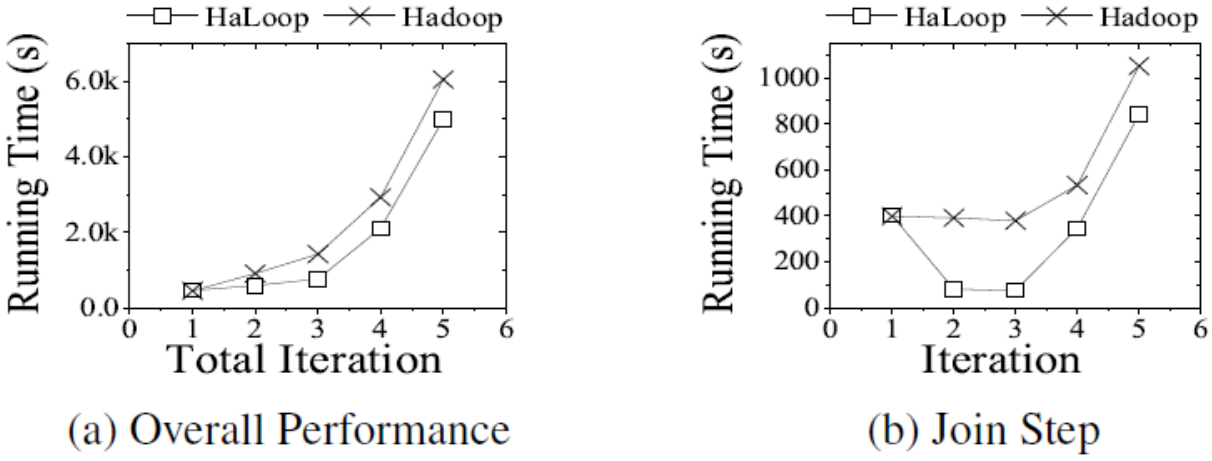

Fig. 5 – Descendent query performance comparison [24]

As can be seen in figure 5, it is evident that the join step and over all performance of Hadoop and Haloop are presented. The iterations and running time are analyzed. From the results it can be understood that the Haloop shows better performance when compared to Hadoop. As future work the authors wanted to implement an evaluation engine for a simplified datalog in Haloop [24].

Large Scale Data Processing with Sailfish

Rao, Ramakrishnan, and Silberstien [28] presented a new framework for processing big data known a Sailfish. It also uses MapReduce layer. However, its design is improved to enhance Map and Reduce phases of the new programming paradigm suited for big data processing. They have built sailfish in such way that it can improve 20% faster

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

performance when compared with Hadoop besides supporting a new future known as auto-tuning. They studied many frameworks and presented Sailfish. The frameworks they studied include MapReduce, Dryad, Hadoop, and Hive. The authors have improved the functionality of MapReduce in their framework by name Sailfish [28]. The map tasks and reduces tasks and their functionality has been improved as shown in figure 6.

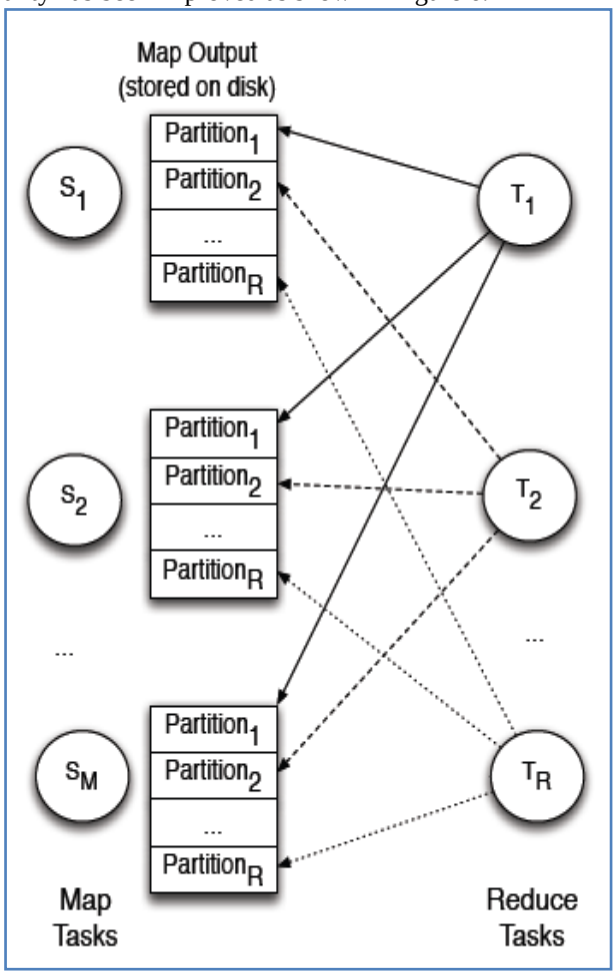

Fig. 6 – Map and Reduce tasks [28]

As can be seen in figure 6, the map tasks and reduces tasks work together. A reduce task gets inputs from map task. The number of retrievals is proportional to number of map and reduces tasks. When compared with other frameworks, Sailfish improves performance and also has auto tuning functionality. With regard to performance the Sailfish is able to improve 20% performance. This fact was known when it was tested with benchmark datasets at Yahoo. Especially its performance is higher than that of Hadoop. It aggregates outputs of map tasks as part of its design principles for efficient data management. It exploits parallelism automatically and has the capability of auto-tuning. With this facility it and handle burst in data and skew in intermediate data and able to decouple the sorting process of intermediate data. The intermediate data handling is very important in big data processing. Sailfish has good support for intermediate data processing. In fact, there is impact on intermediate data size and tuning. When compared with Hadoop's performance Sailfish is better to handle intermediate data when is beyond 16TB. In order to reduce computational overhead, the Sailfish has batching data for disk I/O concept [28]. Figure 7 shows the batching data for disk I/O.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

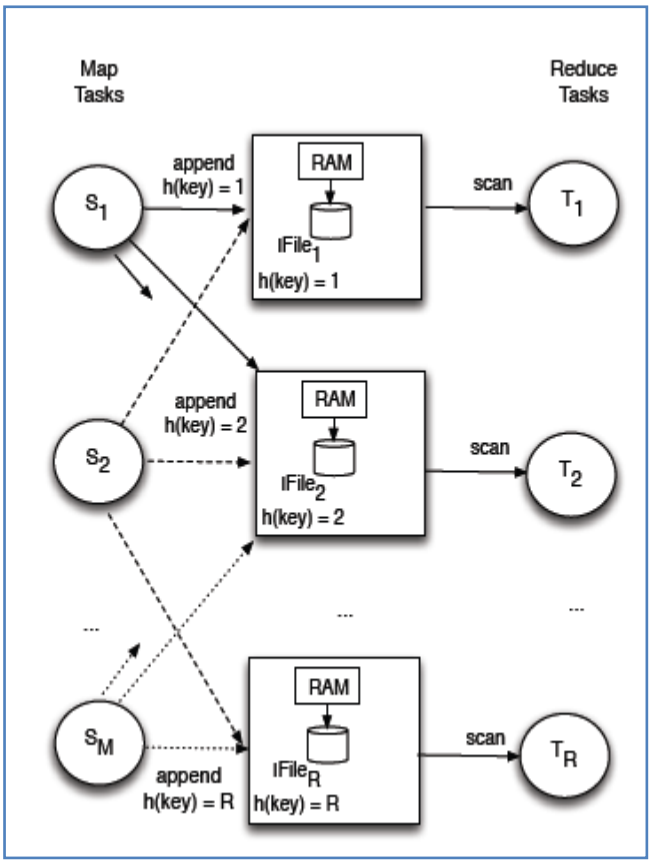

Fig. 7 – Batch processing in Sailfish [28]

As can be seen in figure 7, overhead of the batch processing is reduced in Sailfish. It is achieved as it can process intermediate data well. It commits data to disk once map output is done successfully. Thus Sailfish can improve batch processing performance. Using I files is very important future of Sailfish [28]. The dataflow in Sailfish is as presented in figure 8.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

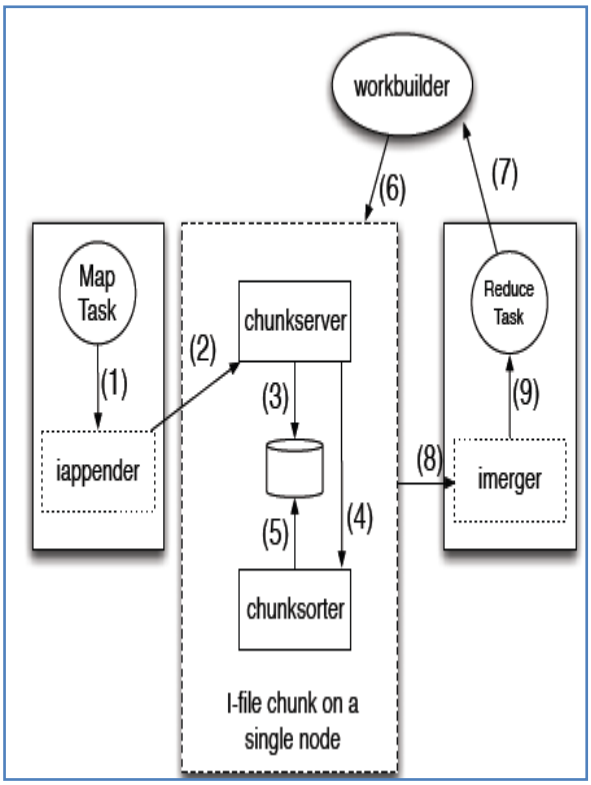

Fig. 8 – Illustrates data flow in Sailfish [28]

As seen in figure 8, it is evident that the data flow is coordinated by iappender, chunksorter, chunkserver, imerger and work builder. First of all the Map Task is given iappender, then after processing it hands it over to chunk server which stores it in database. Then the chunksorter component takes the data in the form of I-files and the processed outputs are given to imerger which merges the data and gives the resultant output to Reduce Task. That will be coordinated by workbuilder [28].

IV.EXPERIMENTS AND FUTURE WORK

Experiments are made to know the performance of Hadoop and Sailfish for big data processing. The Hadoop performance with various intermediate data sizes is presented in figure 9. The comparison results of Hadoop and Sailfish are presented in figure 10, 11 and 12.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

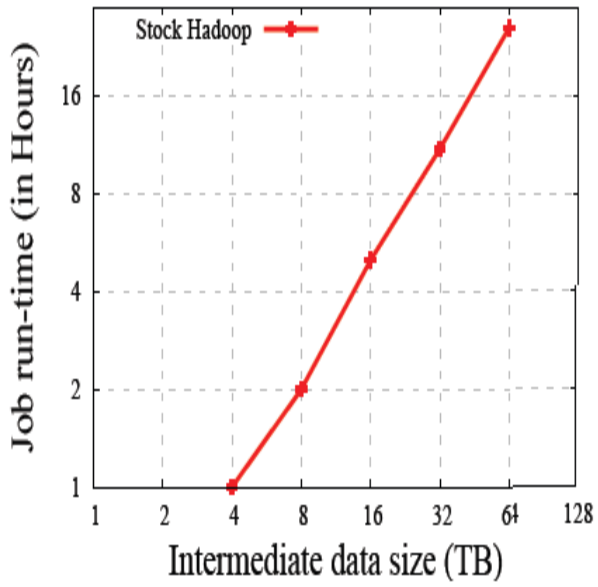

Fig. 9 – Hadoop performance with intermediate data size [28]

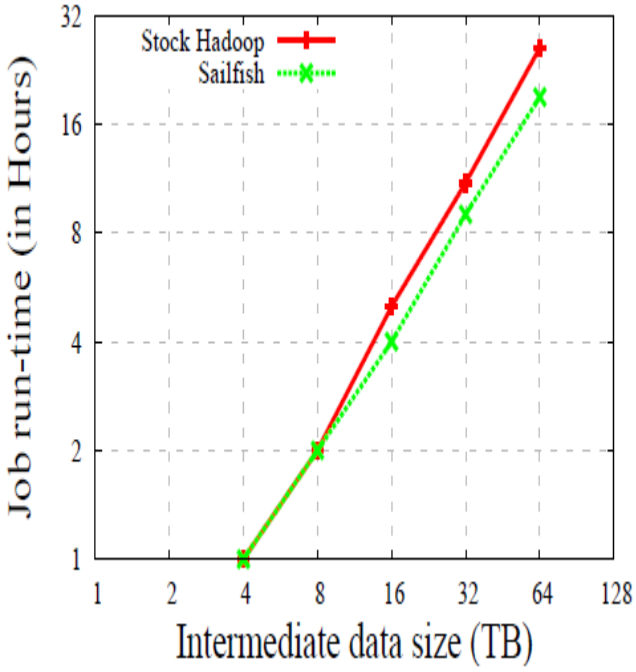

Fig. 10 – Performance comparison between Hadoop and Sailfish [28]

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

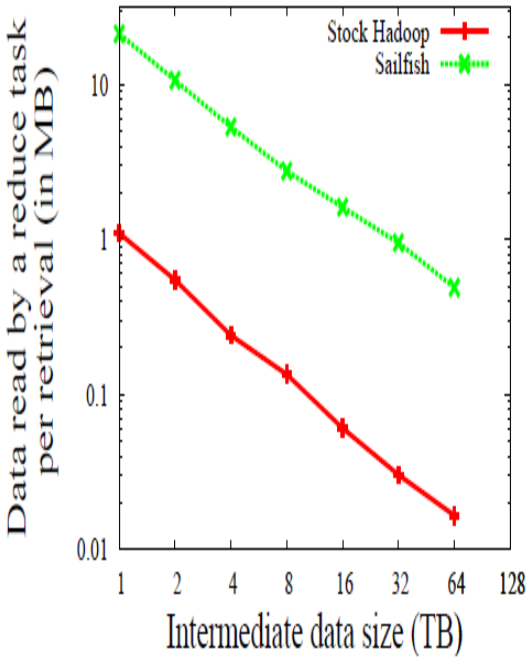

Fig. 11 – Comparison between Hadoop and Sailfish with reduce task and intermediate data size [28]

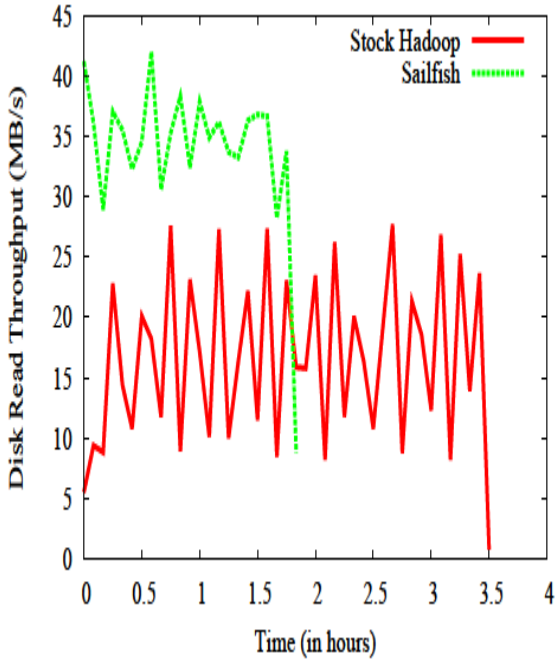

Fig. 12 – Disk read throughput comparison between Hadoop and Sailfish [28]

As seen in figure 10, 11, and 12 the performance of Sailfish is better than that of Hadoop as it has better handling of intermediate data and also auto-tuning functionality. Future direction of these authors is to evaluate Sailfish in real world environments and improve it further [28].

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

V.FUNCTIONAL PROGRAMMING AND DATA FLOW GRAPHS FOR BIG DATA PROCESSING

Tran, Skhiri, Lesuisse, and Zimanyi [29] presented a new framework for big data processing. Their framework is named as AROM. This tool makes use of functional programming and data flow graphs to process huge amount of data efficiently. The MapReduce model provided by Google is well known to the world but it has serious limitations. However models based on data flow graphs show improved performance. AROM is using such data flow graphs and functional programming for processing big data. The tool improves scalability in distributed environment. It handles pipelined tasks more naturally when compared with other frameworks. There are two models for processing big data. They are known as MapReduce and Data Flow Graphs (DFGs). The MapReduce model was originally proposed by Google [29]. This model is as shown in figure 13.

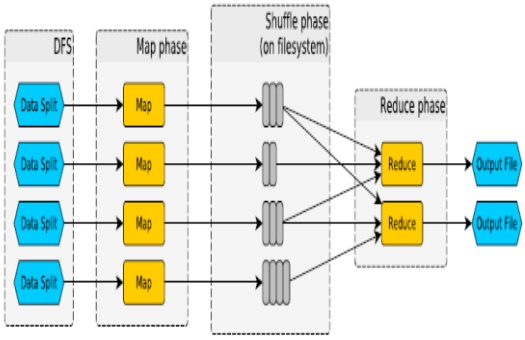

Fig 13 – MapReduce model for big data processing [29]

As can be seen in figure 13, the MapRedue model ha many phases involved. Important phases are Map phase and Reduce phase. First of all, the DFS component take bit data and splits the data. Such data is mapped in the Map phase. Afterwards, the maps are processed using Shuffle phase on the file system. Afterwards, the Reduce phase generates final output. The MapReduce model ha some drawbacks. They include mandated shuffle phase is not efficient, and joins are also cumbersome. The other programming model is known as DFG. Based example for DFG is Microsoft's Dryad. The pipelining in Dryad is better than that of MapReduce [29]. The pipelining process between the MapReduce and Dryad are provided in figure 14.

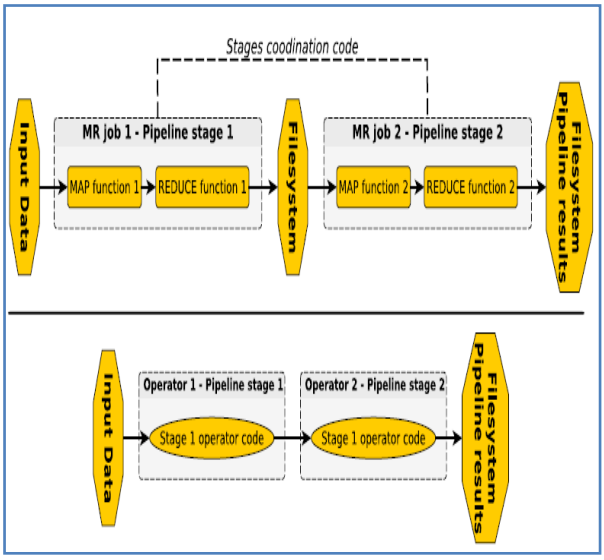

Fig. 14 – Comparison of pipelining between MapReduce and Dryad [29]

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

As can be seen in figure 14, the MapReduce approach for pipelining I given at the top while the same is provided at the bottom. In case of MapReduce, the first stage must be completed before going to second state. It makes it more time consuming. The Dryad case is different. The result of the first stage is streamlined directly to the second stage to improve processing performance. This is the advantage of Data Flow Graphs based processing. It too has drawbacks such as the freedom in its implementation may be misused and it is more general than MapReduce. The PageRank version of AROM ha been improved over that of MapReduce [29]. This is shown in figure 15.

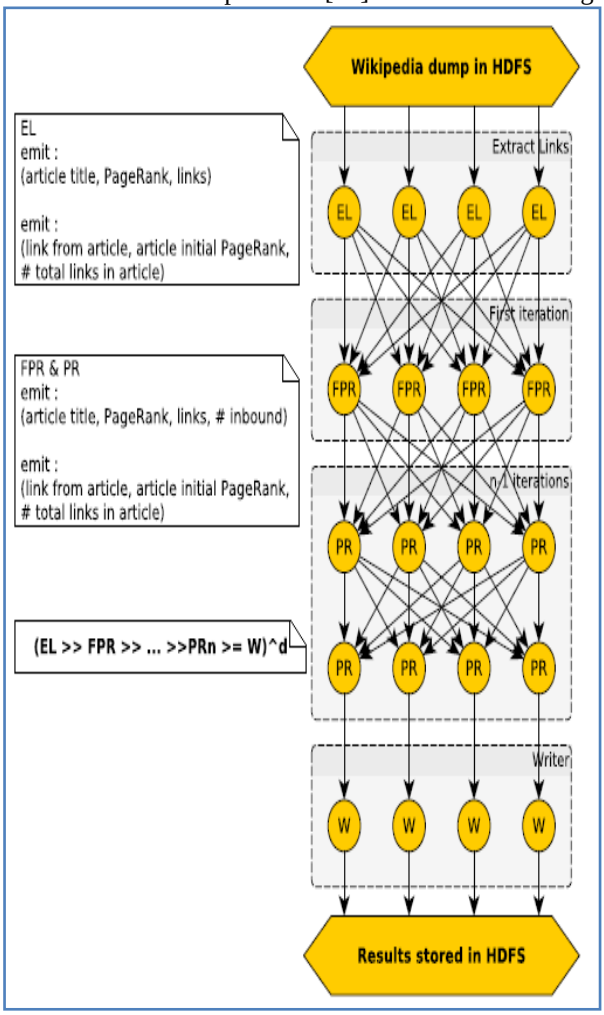

Fig. 15 – PageRank version of AROM [29]

As can be seen in figure 15, the architecture of PageRank process has more flexibility as it supports parallel processing of data. More parallel processing is achieved with its new architecture when compared with that of MapReduce [29].

VI.EXPERIMENTS AND FUTURE WORK

Experiments are made to know the difference between the two kinds of models to process big data. The models compared with the experiments are MapReduce and DFG. Experimental results show the PageRank computation over 5 iterations with MapReduce of Hadoop and DFG of AROM [29]. The experimental results are shown in figure 16.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

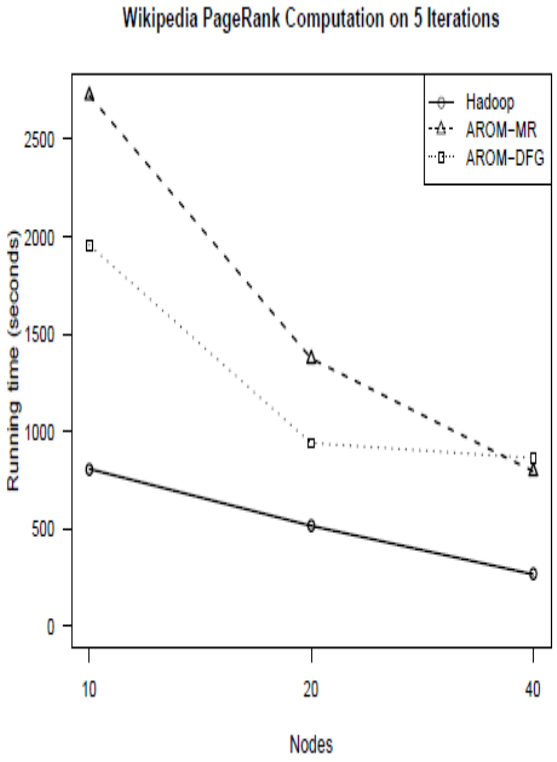

Fig. 16 – Number of nodes vs. running time of PageRank of Hadoop and AROM [29]

As shown in figure 16, the performance of AROM with DFG and AROM with MR show improved performance with MR of Hadoop. The future work to this is conceived as to improve scheduling process further with AROM for better performance [29].

VII.CONCLUSION

This paper has presented many details of Big Data and its characteristics with respect to Velocity, Volume and Variety. The need for big data processing and the real world advantages of processing big data in terms of generating revenues to companies and governments is discussed. Various frameworks for big data processing are reviewed in some details. This paper focuses more on three important frameworks that are used for processing big data. They include AROM which is based on functional programming and DFGs, Sailfish for large scale data processing and Haloop which is a variant of Hadoop. The essence of the paper is that there are many frameworks to process big data and the big data can lead to big advantages to companies and governments.

REFERENCES

1. BogdanGhit¸ AlexandruIosup and Dick Epema (2005). *Towards an Optimized Big Data Processing System*. USA: IEEE. P1-4.
2. TAKAHASHI Chieko, SERA Naohiko, TSUKUMOTO Kenji, OSAKI Hirotatsu (2012). *OSS Hadoop Use in Big Data Processing*. USA: NEC TECHNICAL JOURNAL. p1-5.
3. Dibyendu Bhattacharya (2013). *ANALYTICS ON BIG FAST DATA USING REAL TIME STREAM DATA PROCESSING ARCHITECTURE*.us: EMC Proven Professional Knowledge Sharing. p1-34.
4. ToyotaroSuzumura (2012). *Big Data Processing in Large-Scale Network Analysis and Billion-Scale Social Simulation*. Tokyo: IBM Research. p1-2.
5. ChangqingJi, Yu Li, WenmingQiu, UchechukwuAwada, Keqiu Li (2012).*Big Data Processing in Cloud Computing Environments*. China: International Symposium on Pervasive Systems. p1-7.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 1, January 2014

6. LiranEinav and Jonathan Levin (2013). *The Data Revolution and Economic Analysis*. USA: Prepared for the NBER Innovation Policy and the Economy Conference. p1-29.
7. Emmanuel Letouzé (2011). *Big Data for Development: Challenges & Opportunities*. United Kingdom: UN Global Pulse. p1-47.
8. An Oracle White Paper (2013). *Oracle: Big Data for the Enterprise*. USA: Oracle .p1-16.
9. MaheswaranSathiamoorthy, MegasthenisAsteris and DimitrisPapailiopoulos (2013). *XORing Elephants: Novel Erasure Codes for Big Data*. Italy: Proceedings of the VLDB Endowment. p1-12.
10. Nam-Luc Tran and SabriSkhiri and Arthur Lesuisse and Esteban Zim´anyi (2012). *AROM: Processing Big Data With Data Flow Graphs and Functional Programming*. Belgium: Amazon. p1-8.
11. Yingyi Bu, Bill Howe, Magdalena Balazinska and Michael D. Ernst (2010).*HaLoop: Efficient Iterative Data Processing on Large Clusters*. USA: IEEE. p1-12.
12. SriramRao, Raghu Ramakrishnan and Adam Silberstein (2012). *Sailfish: A Framework For Large Scale Data Processing*. USA: Microsoft. p1-14.
13. Dawei Jiang, Gang Chen, Beng Chin Ooi, Kian Lee Tan, Sai Wu (2010).*epiC: an Extensible and Scalable System for Processing Big Data*. Singapore: CS. p1-12.
14. Michael Cooper & Peter Mell (2013). *Tackling Big Data*.us: National Institute of Standards and Technology. p1-40.
15. Michael Schroeck, Rebecca Shockley, Dr. Janet Smart, Professor Dolores Romero-Morales and Professor Peter Tufano (2012). *Analytics: The real-world use of big data*. USA: IBM Global Business Services. p1-20.
16. Eric Schmidt (2012). *Big data*.us: SAS. p1-30.
17. XIAO DAWEI, AO LEI (2013). *Exploration on Big Data Oriented Data Analyzing and Processing Technology*. us: IJCSI International Journal of Computer Science. p1-6.
18. Mike Ferguson (2012). *Architecting A Big Data Platform for Analytics*. England: Intelligent Business Strategies. p1-36.
19. Dr. Nathan Eagle"s (2010). *Big Data, Big Impact: New Possibilities for International Development*. Switzerland: The World Economic Forum. p1-10.
20. Ling LIU (2012). *Computing Infrastructure for Big Data Processing*. USA: IEEE. p1-9.
21. Yanpei Chen, Sara Alspaugh, Randy Katz (2010). *Interactive Analytical Processing in Big Data Systems: A CrossIndustry Study of MapReduce Workloads*. USA: IEEE. p1-12.
22. Chris Wilson and Jennifer Kerber (1979). *Demystifying Big Data: A Practical Guide To Transforming The Business of Government*. North American: TechAmerica Foundation. p1-40.
23. McKinsey (2011). *Big data: The next frontier for innovation, competition, and productivity*. US: MGI. p1-156.
24. Kevin McGowan (2013). *Big data: The next frontier for innovation, competition, and productivity. US: MGI. p1-156.*. USA: SAS Solutions on Demond .p1-16.
25. Jens DittrichJorgeArnulfoQuian´eRuiz (2012). *Efficient Big Data Processing in HadoopMapReduce*. USA: Proceedings of the VLDB Endowment. p1-2.

BIOGRAPHY

Dr Shoban Babu Sriramoju is an Associate Professor at Varadha Reddy Engineering College, Warangal . He received M.Tech degree in 2006 from Allahabad Agricultural University, Allahabad and Ph.D from Chandra Mohan Jha University, India. His research interests are Data Mining, Big Data, Web Technologies etc.