International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)
Web Site: www.ijettcs.org Email: editor@ijettcs.org
Volume 4, Issue 3, May-June 2015 ISSN 2278-6856

Comparative Analyses of Scheduler

Shivani Thakur[1], Rupinder Singh[2]

[1.]Research Scholar Dept. of Computer Science & Engg.
Chandigarh University, Ghauran, Mohali

[2.]Assistant Professor Dept. of Computer Science & Engg.
Chandigarh University, Ghauran, Mohali

Abstract

Hadoop is in large demand within the market currently days. As there large quantity of information is lying within the trade however there's not tool to handle it. Hadoop will enforce on low price hardware and might be utilized by giant set of audience on sizable amount of dataset. In hadoop map reduce is that the most vital part in hadoop. In this paper we have discussed the difference between Hadoop and RDBMS; different scheduling issues that why we need scheduling, and also explain the comparison among widely used schedulers.

Keywords: Map Reduce, Hadoop, HDFS, Scheduler.

1. INTRODUCTION

This is the time of digitalization and that we all board digital world. During this era we would like a lot of and a lot of facility for our life is being easier. Thus we tend to used completely different quite new things for entertaining our life exploitation audio, video and digital footage, through social stuff style of a Facebook Google+, victimization mobile to boot as devices, we have a tendency to used private or public sector that everyone this will increase the digitalization and make an oversized quantity of structured, semi structured and un-structured knowledge. Face book and Google+ or alternative social web site activity daily billions of information area unit being shared at an equivalent time. Through sensing element network, instrument and mo-bile devices additionally generates great amount of knowledge in few hours. In paper of the large information spectrum they same one interesting issue is IDC Terms this because the digital universe and IDC additionally predict that in year 2015 this digital universe is generates a fantastic eight letter bytes of massive information [1] [2]. Big knowledge merely sort of a smaller information however distinction is it's an oversized volume of information is in zeta bytes that's why it's known as huge data. giant information produce a obstacle to the corporate likewise as government organization for storage, capture, search, transfer, sharing, analysis, and image. To over comes this drawback apache introduced one free open source tool that is known as the Hadoop. Since 82% of information is in "unstructured" kind, it should be formatted in a very unique manner that creates it suitable for ulterior analysis and data processing [3]. Apache Hadoop is core platform for structuring huge knowledge, and cracks the problem of creating it helpful for analytics purpose. The rest of the article is organized as follows. In Section II, we present a concise description of Hadoop, HDFS architecture, RDBMS v/s Hadoop, describe the main problem of scheduling in map reduce. In Section III, mainly used scheduling algorithm in Hadoop map reduces will be described. In Section IV, we have a discussion and analysis of these algorithms and show the preemption, working, taxonomy, fairness. We conclude our paper in Section V.

2. Preliminary Study

2.1 Overview of Hadoop

Hadoop was created by Doug Cutting, the creator of Apache Lucene, the wide used text search library. Hadoop has its origins in Apache Nutch, an open supply net program, itself a locality of the Lucene project [3] [4]. Hadoop is highly available extensive data storage engine. The main importance of using hadoop is that you can merge processing and data storage. Hadoop is a, Java-based programming framework that supports the processing of huge information sets during a distributed computing setting and is a component of the Apache project sponsored by the Apache software system Foundation. It will provide plenty of needed strength and quality option to a distributed system as Hadoop provides low cost and reliable storage. HDFS is employed for information storage and Map scale back is employed for process the information in Hadoop. Hadoop is also a cross-platform framework that supports information intensive distributed cloud applications paying attention on information method. Hadoop is meant supported the Map scale back programming model. This model divides the applications into many tiny tasks to be distributed and executed on the Hadoop resources. The goal of Hadoop is to supply economical and high performance process of massive information applications [5]. Hadoop consist two components mainly as hadoop map reduce and HDFS [6] [7].

2.2. Map Reduce: A Programming model

Map Reduce is generated by map and reduce Operations in practical languages, like Lisp. This model abstracts computation issues through two functions i.e. map and reduces [8]. All issues developed in this way is parallelized mechanically. Essentially, the Map Reduce model permits users to write Map Reduce elements with functional-style code. These elements are then composed as a dataflow graph with fastened dependency relationship to explicitly specify its correspondence. Finally, the Map Reduce runtime system will transparently explore the correspondence and schedule these elements to distributed resources for execution. All information processed by Map Reduce are within the type of key/value pairs. The execution happens in 2 phases. In the 1st section, a map perform is invoked once for each input key/value combine and it will generate output key/value pairs as intermediate results. Within the second one, all the intermediate results are incorporated and grouped by keys [9]. The reduce perform is termed once for each key with associated values and produces output values as final results. We use Hadoop elements to perform job execution, file staging and advancement info storage and use the files replace the information to store datasets. Within the files every line will be seen as a group action, every item is separated with white space. The datasets in files are split into smaller segments mechanically when hold on in HDFS and therefore the map perform is executed on every of those knowledge segments. At first, the candidate sub item are place out with the counts variety after the execution of map perform, then the frequent item sets are found when the execution of map perform [10].

Input /Output	Map function	Reduce function
Input: Key/value pairs	Key: line No; Value: one row of data	Key: candidate item sets; Value:1
Output: Key/value pairs	Key: candidate item set; Value:1	Key: Frequent sub items; Value: support

Fig. 1. Key/value pairs for map function and reduce function.[10]

Input data partition count the items in the prune
Key value candidate set at the the candi-
operation of map -date & summ
-ary the same Item to get data set

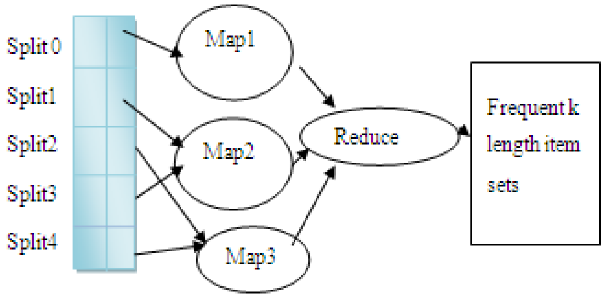


Fig. 2. Data flow of Map Reduce [10]

2.3 Hadoop Distributed File System

The Hadoop Distributed file system (HDFS) could be a distributed file system designed to run on trade goods hardware. it\'s several similarities with existing distributed file systems. However, the variations from different distributed file systems are vital [11]. HDFS is extremely fault-tolerant and is intended to be deployed on affordable hardware. HDFS provides high output access to application knowledge and is appropriate for applications that have giant knowledge sets. HDFS relaxes a couple of POSIX necessities to alter streaming access to classification system knowledge. HDFS was originally designed as infrastructure for the Apache Nutch net computer programmer's project. HDFS is currently associate Apache Hadoop subproject [12].

HDFS features a master-slave design. Associate HDFS cluster consists of one Name Node, a master server that manages the filing system namespace and regulates access to files by shoppers. Additionally, there are a unit variety of Data Nodes, typically one per node within the cluster, that manage storage hooked up to the nodes that they run on. HDFS exposes a filing system namespace and permits user information to be keeping in files. Internally, a file is split into one or additional blocks and these blocks area unit keep in a very set of Data Nodes. The Name Node executes filing system namespace operations like gap, closing, and renaming files and directories [10] [11]. The Name Node executes classification system namespace operations like closing, new, and renaming files and directories. It conjointly determines the mapping of blocks to Data Nodes. The Data Nodes square measure answerable for serving scans and writes requests from the file system's purchasers. The Data Nodes conjointly perform block creation, deletion, and replication upon instruction from the Name Node [11].

The Name Node and Data Node square measure items of code designed to run on goods machines. These machines usually run a GNU/Linux package [HYPERLINK \l "Dua12" 12].

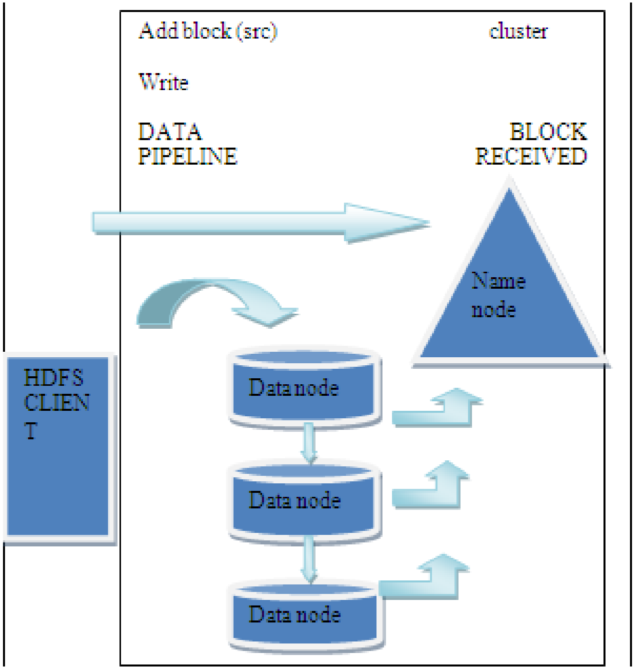


Fig 2.3. Hadoop Architecture [11]

An HDFS a new creates a brand new file by giving its path to the Name Node. For every block of the file, the Name Node returns a list of Data Nodes to host its replicas. The consumer then pipelines information to the chosen Data Nodes, that eventually make sure the creation of the block replicas to the Name Node [11].

2.4. RDBMS v/s Hadoop

RDBMS fetching and storing the data faster than DBMS.SQL command were also used in RDBMS .By using these command user easily retrieve the data but it not an easy to learn whole command and also not suitable when we need to store and retrieve large amount of complex data .As the data of companies increases day- by -day and also the number of users and concurrency of user also increased rapidly [13].We found that we need a new database system in which we stored terabyte and pet byte amount of data effectively, to improve this Big data and Hadoop are rapidly emerging as the preferred solution where information is simply too massive from numerous sources in several type [14].The following table show the difference among Hadoop and RDBMS in different – different scenario.

Table 2.1 RDBMS v/s Hadoop

	RDBMS	**Hadoop**
Work	Work strictly in structure data	Work well both in structure as well as unstructured data
Processing	Point queries, OLTP processing	Batch, OLAP , Analytical queries
Programming Model	Declarative (SQL)	C/C+, JAVA OR extensive through pig and Hive
Node	–	Add extra node which will be self managed
Schema	Inflexible schema	Flexible schema
Property	Scale-up	Scale-out
Function	key-value pair	Record
Expense	Very expensive	Cost effect
Fault Tolerance	Coarse grained technique	possible

2.5 Scheduling issues

There are III necessary scheduling problems in map reduce like fairness, locality, and synchronization. Locality may be terribly crucial issues disturb the performance in an exceedingly shared cluster surrounding, due to limited network bandwidth [8]. It is defined because the distance between the computers files node and task-assigned node. The thought of information locality in Map Reduce may be a time once computer hardware tries to assign Map tasks to slots out there on machines during which the underlying storage layer holds the input meant to be processed [15]. Fairness finiteness have trade-offs between the neighborhood and need between the map and reduce phases .When every Map Reduce job has about calculated equal share of the nodes and also the input files are spread in distributed classification system, some map processes ought to load knowledge from the networks. This causes an excellent degradation in outturn and response time. Synchronization overhead could affect the fairness [15] [16] [17] [18]. To resolve these necessary problems different- different scheduling algorithm have been proposed in the past decade in the next section we describe the most widely used scheduling algorithm.

3. Scheduling algorithm and related work

The issues that are described above in scheduling issues is a critical aspect of map reduce. Many algorithm is used to address these problems with different and approaches .some is used to improve the data locality and some is used to make efficient synchronization.

3.1. FIFO scheduling algorithm

In the earliest Hadoop Map Reduce computing architecture, the essential job sort is massive batch jobs that a single user submits, thus Hadoop use inventory accounting (First in 1st out) rule in early planning algorithm [19]. The jobs of all users are brought up just one queue. According to the priority level and also the time sequence when they are submitted, the complete job queues are scanned, and then a satisfactory job is chosen to execute. FIFO is simple, the value of entire cluster planning method is Less. Only single type of job is basically designed for FIFO so when multiple users at the same time run multiple sorts of jobs, performance are going to be comparatively low. As the usage rate of Hadoop platform is progressively high, the demand is additionally magnified [20]. The FIFO algorithmic rule tends to decrease the general performance of the platform and the utilization of system resources, and generally even affect the implementation of jobs.

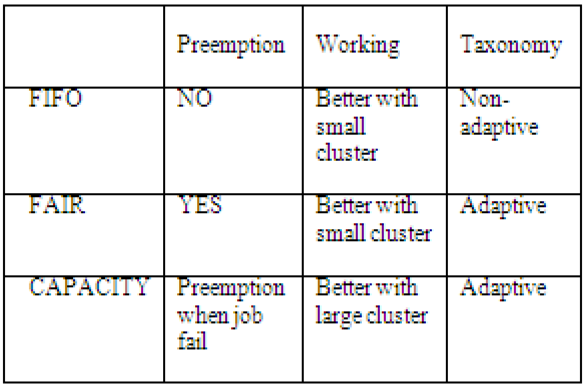
3.2 Fair scheduling algorithm

Fair programming can be a method of assignment resources to jobs such each job gets, on average, an equal share of resources over time [21]. If there is a single job running, the duty uses the complete cluster. Once alternative jobs are submitted, empty task slots are appointed to the new jobs, so every job gets roughly identical quantity of CPU time. It lets short jobs complete among an inexpensive time whereas not starving long jobs [22]. The objective of honest scheduling rule is to try and do a equal distribution of compute resources among the users/jobs within the system [23]. The scheduler truly organizes jobs by resource pool, and shares resources fairly between these pools. By default, there's a separate pool for each user. actuality computer hardware can limit the amount of synchronic running jobs per user and per pool. Also, it'll limit the number of coincident running tasks per pool Tao et al [24] introduced an improved truthful scheduling formula, that takes under consideration job characteristics and information locality, that decreases both information transfer and therefore the execution time of jobs. Fair scheduler covers some limitation of FIFO such as: it will works well in each tiny and large clusters and fewer complicated. This algorithm doesn't contemplate the duty weight of every node, that this is often a crucial disadvantage of it.

3.3. Capacity scheduling algorithm

Fair scheduler and capacity scheduling algorithm is very similar but capacity scheduler used queue instead of pool. Each queue is assigned to associate organization and resources square measure divided among these queues. Capacity scheduling puts jobs into various queues in accordance with the conditions, and allocates bound system capacity for every queue. If a queue has serious load, it seeks unallocated resources, then makes redundant resources allotted equally to every job [4]. It re-allocates the resources for empty queue to queues exploitation for increasing resource. Once jobs arrive therein queue, running tasks area unit completed and resources area unit given back to main queue. It additionally permits priority based totally programming of jobs in associate organization queue [25].Capacity programming algorithmic rule addresses the FIFO's disadvantage just like the low utilization rate of resources. The foremost advanced among III schedulers may be a very important downside in capacity algorithmic [26].

Table 3.1 Comparison between scheduler



	Preemption	Working	Taxonomy
FIFO	NO	Better with small cluster	Non-adaptive
FAIR	YES	Better with small cluster	Adaptive
CAPACITY	Preemption when job fail	Better with large cluster	Adaptive

4. Conclusion

In this paper we described the Hadoop Map reduce and its architecture .We try to work toward plotting the boundary to analyze hadoop v/s traditional database on the bases of fault tolerance, programmed model, node property. After the literature analysis it has been concluded that FIFO, FAIR AND CAPACITY most widely used scheduler in hadoop because they are adaptive in nature. After analyzing the big Data literature and hadoop scheduler based on working, preemption, taxonomy it has been found that hadoop is best approach in case of distributed data.

REFERENCES

[1]. John Gantz, David Reinsel, "Extracting Value from Chaos", IDC IVIEW, June 2011.

[2]. Contributing Authors, "Big Data Spectrum", Infosys Limited Banga-lore India, 2012

[3]. Inukollu, V. N., Arsi, S., & Ravuri, S. R. (2014). "Security Issues Associated With Big Data In Cloud Computing". International Journal of Network Security & Its Applications (IJNSA), 6(3).

[4]. Umesh V. Nikam, Anup W. Burange, Abhishek A. Gulhane, "Big Data and HADOOP: A Big Game Changer", International Journal of Advance Research in Computer Science and Management Studies, Volume 1, Issue 7, ISSN: 2321-7782, DEC 2013

[5]. White, T., 2012, "Hadoop: The Definitive Guide", ed. Third, Tokyo: Yahoo press.

[6]. Rasooli, Aysan, and Douglas G. Down. "A Hybrid Scheduling Approach For Scalable Heterogeneous Hadoop Systems." In High Performance Computing, Networking, Storage and Analysis (SCC), 2012 SC Companion: pp. 1284-1291. IEEE, 2012.

[7]. Kurazumi, Shiori, Tomoaki Tsumura, Shoichi Saito, and Hiroshi Matsuo. "Dynamic Processing Slots Scheduling for I/O Intensive Jobs of Hadoop Map Reduce." In ICNC, pp. 288-292. 2012.

[8]. Dean, J., & Ghemawat, S. (2008). "Map Reduce: Simplified Data Processing On Large Clusters." Communications of the ACM, 51(1), 107-113.

[9]. Nanduri, Radheshyam, Nitesh Maheshwari, A. Reddyraja, and Vasudeva Varma. "Job aware scheduling algorithm for map reduce framework." In Cloud Computing Technology and Science, 2011 IEEE Third International Conference on, pp. 724-729.

[10]. Yang, X. Y., Liu, Z., & Fu, Y. (2010, June). Map Reduce as a programming model for association rules algorithm on Hadoop. In Information Sciences and Interaction Sciences (ICIS), 2010 3rd International Conference on (pp. 99-102). IEEE.

[11]. Shvachko, Konstantin, Hairong Kuang, Sanjay Radia, and Robert Chansler. "The hadoop distributed file system." In Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on, pp. 1-10. IEEE, 2010.

[12]. Duan, Ailing. "Research and application of distributed parallel search hadoop algorithm." In Systems and Informatics (ICSAI), 2012 International Conference on, pp. 2462-2465. IEEE, 2012.

[13]. Hu, Han, et al. "Towards Scalable Systems for Big Data Analytics: A Technology Tutorial." (2014)..

[14].https://software.intel.com/sites/default/files/article/402274/etl-big-data-with-hadoop.pdf}

[15]. Pastorelli, M., Barbuzzi, A., Carra, D., Dell'Amico, M., & Michiardi, P. (2013). Practical size-based scheduling for Map Reduce workloads. ArXiv preprint arXiv: 1302.2749.

[16]. Polato, Ivanilton, Reginaldo Ré, Alfredo Goldman, and Fabio Kon. "A Comprehensive View Of Hadoop Research—A Systematic Literature Review."Journal of Network and Computer Applications 46 (2014): 1-25.

[17]. Ekanayake, Jaliya, Shrideep Pallickara, and Geoffrey Fox. "Mapreduce for Data Intensive Scientific Analyses." In eScience, 2008. EScience'08.

IEEE Fourth International Conference on, pp. 277-284. IEEE, 2008.

[18]. D. Yoo and K.M Sim, "A Comparative Review Of job Scheduling For Map Reduce", In Cloud Computing and Intelligence Systems (CCIS), 2011 IEEE International Conference on IEEE, September 2011, pp. 353-358.

[19]. M. Isard, M. Budiu, Y. Yu, "Distributed Data-Parallel Programs from Sequential Building Blocks," Proceedings of the 2nd ACM SIGOPS European Conference on Computer Systems, ACM, 59-72. 2007.

[20]. "Hadoop home page." http://hadoop.apache.org/

[21]. Hadoop's Fair Scheduler. https://hadoop.apache.org/docs/r1.2.1/fair_sche Duler. [As accessed on 9 Feb. 2015]

[22]B.P Andrews and A. Binu. "Survey on Job Schedulers in Hadoop Cluster", IOSR Journal of Computer Engineering, Vol.15, NO. 1, Sep. Oct. 2013, pp. 46-50.

[23] The Apache Hadoop Project. http://www.hadoop.org.

[24] Y. Tao Y, Q. Zhang, L. Shi and P. Chen. "Job Scheduling Optimization for Multi-user Map Reduce Clusters", In: The fourth International symposium on parallel Architectures, algorithms and programming. IEEE; 2011. p. 213–17.

[25] N. Tiwari, "Scheduling and Energy Efficiency Improvement Techniques for Hadoop Map reduce: State of Art and Directions for Future Research (Doctoral dissertation, Indian Institute of Technology, Bombay Mumbai).

[26]http://hadoop.apache.org/docs/current/hadoop-yarn/hadoop-yarn-site/CapacityScheduler.html

[27] Pandey, Shweta, and Vrinda Tokekar. "Prominence of Map Reduce in Big Data Processing." In Communication Systems and Network Technologies (CSNT), 2014 Fourth International Conference on, pp. 555-560. IEEE, 2014.