Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 132 (2018) 518–524

Procedia Computer Science

www.elsevier.com/locate/procedia

International Conference on Computational Intelligence and Data Science (ICCIDS 2018)

High Performance Parallel Computing with Cloud Technologies

R. Kannadasan*, N. Prabakaran, P. Boominathan, A.Krishnamoorthy, K. Naresh and G. Sivashanmugam

School of Computer Science and Engineering
VIT University, Vellore, India.

Abstract

Cloud is referred to as a collection of infrastructure services, such as Infrastructure as a service (IaaS) and Platform as a service (PaaS), which are made available to us for utilization by various organizations in which the key factor is virtualization of data as it allow the user to manage, handle and compute a large number of tasks very easily.By referring to Cloud technologies we mean runtime such as Hadoop, Dryad and other Map Reduce frameworks. In this paper we would analyse the above mentioned software's and techniques for the cloud system by comparing them on the basis of its processing speed, its data handling capacity , the nature of user friendliness. We would discuss large scale data analysis using different implementations on the above mentioned tools and after that we would give a performance analysis of these tools on the given implementation like Cap3 , HEP ,Cloudburst.

Keywords: Map reduce, hadoop,cloud technology,Cap3,HEP,cloudburst.

*Corresponding Author: kannadasan.r@vit.ac.in

10.1016/j.procs.2018.05.004

1. Introduction

Cloud storage and its related technologies are the two diverse categories of technologies dealing with common notion of Cloud Computing.Cloud refers to a group services like Platform-as-a-Service (PaaS), Infrastructure-as-a-service (IaaS), etc in which virtualization plays a very vital and important role. Whereas Cloud Technology means different runtimes software based on cloud like Hadoop, Dryad , and communicating frameworks like [1] HDFS(Hadoop Distributed File System), Amazon S3,and many more.In the Present scenario, availability of various open source cloud based infrastructure software's like Nimbus, Eucalyptus, etc allows various organizations to develop their own private clouds to increase the efficiency of resource utilization of present computation facilities. Cloud technologies addition has created a new trend in parallel computing.

Various applications gained from cloud technologies, among which data or computation related applications hold the most importance. The large part of data and highly intense computational applications are found in field of particle physics, finance, information retrieval and many more [2].

This work has been divided into three parts, the first part deals with various cloud technologies, second deals with its applications and at the last section deals with the results which shows the analysis on various cloud technology software developed.

2. Hadoop

Apache Hadoop is one of the distributed file based system which has the capability to handle terabytes of information and data [3] .The way in which it different from any other distributed file system is its nature to be flaw - tolerant. It stores data across various different machines in a redundant way such that it decrease the risk of data loss. It can be run on even one computer to hundreds of computer and is based majorly on Google Map Reduce Framework. It major components are HDFS, YARN, Namenode , Datanode , Map Reduce. The architecture followed by Hadoop is of the nature of Master-slave. In it the master is called NameNode whose sole aim is to manage namespace of file system in addition to it also regulate the control to the other files while managing file renaming and closing operations. The datanode is a set of blocks which contain which contain many files which are splitted in it internally .There is mostly a single DataNode corresponding to a node in a Cluster. The role of DataNode is to read the client request as well as write the client request. Yarn stands for Yet Another Resource Navigator which is a framework for allocating resources of a computer for execution. YARN is highly scalable in nature and results in better cluster utilization.

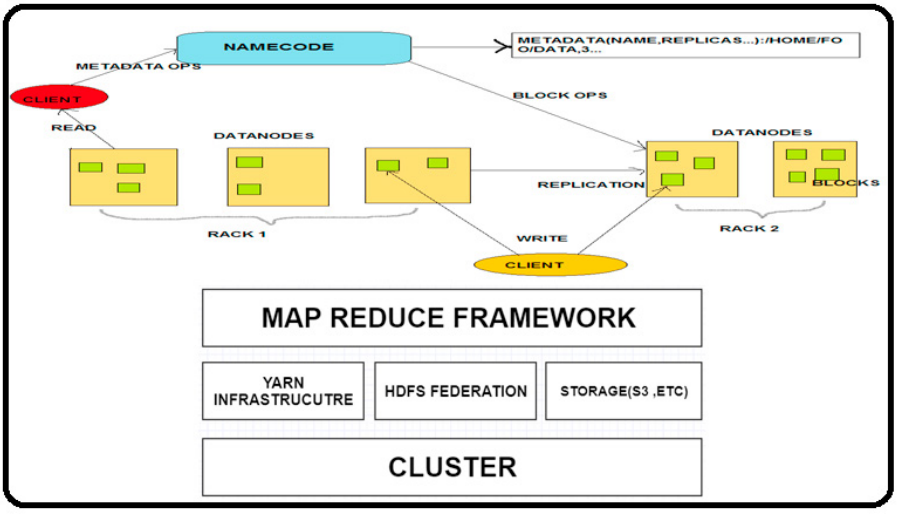

Fig. 1. Map Reduce Architecture

2.1 CGL Map Reduce

It is a streaming type of MapReduce which is executed in Java resulting in eliminating most of the overhead linked

with communication of a file resulting in sending the file directly to the consumer via producer. It offer various improvements to MapReduce [4] like its support in long running map but majorly it offer very fast data transfer from pub broker network. In the starting phase of it the map and reduce task are both configured and we can load any data for map and reduce task. The above stages handle all the task related to data processing. It has a framework which transfer result directly from map to reduced task. Then there is a merged state which convert the result derived from reduced state into a single value. The user is given access to the result of merged state .It have the additional facility of distribution of variable data set to all map based tasks which is very helpful for data analysis. The data partitioning and data distribution is handled by user who can read data from local disk as well as shared file system. The computation of CGL can be both single step as well as iterative .In iterative the change which is there from other than the normal method is that in it the user has the capability to call another iteration of MapReduce by analyzing the result of merge state but this time there is no need to configure the MapReduce task as it is managed by framework already [5] .

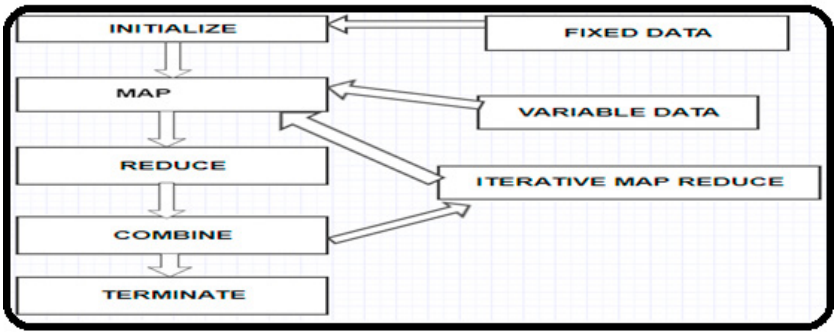

Fig. 2. CGL Map Reduce

2.2 Dryad

It's a system and also set of language extensions that enable a new developed programming mode for distributed computing on a large scale [6]. It basically defines a domain-specific language, which bases on C++ libraries, a general purpose distributed execution for parallel applications. It combines vertices and channels to develop a Dataflow graph. It then runs the applications by executing the graph's vertices on computers present communicating with each other using TCP pipes. Dryad has been developed to be quite expressive. It completely beats other computing frameworks like Google's map reduce [7]. It handles the work of various difficult issues of developing distributed application on a large scale like creating and managing jobs, monitoring and visualizing jobs, fault tolerance, accounting, transporting data between graph's vertices and other examples. It has a build up which allows programmers to utilize the resources of a cluster or data centres for various parallel running programs. It gives its programmers to use thousands of machines where every machine is made up of multiple cores and for this the programmer doesn't need to anything that concerns concurrent programming [8].

2.3 MPI

It stands for Message Passing Interface which offers guideline to developers and the users of message passing libraries. It is based on MPI forums in which there are more than 40 organization which started working on it from the year 1992.Their aim is to build a standard that is flexible and efficient in nature which could be use for the purpose of message sharing and writing of the message passing programs [9]. It is a de-facto standard. In it the data is shared in cooperative processes which are executed on various distributed system. Currently the latest version of the MPI which we are currently using is MPI-3.x.MPI has the capability to run on Hybrid, Distributed memory as well as shared memory. In most of the runtime cases MPI is deployed in computation clusters in which there are a group of nodes which are connected using a high speed network connection resulting in low latencies during communication. The benefit of using MPI standard over other are that it is highly portable which mean that mean we do not need to make any amend to our original source code of our application even when we switch platforms plus there are various running implementation of it present such as OpenMPI, MPICH,LAM.

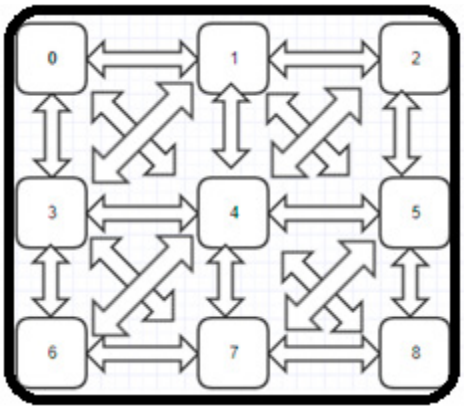

Fig. 3. CGL Map Reduce

2.4 HEP

It is short for High Energy Physics(HEP). It is a field of branch which deal in the study of particle that constitute matter and radiation and study the interaction between the particles. The aim of the study is to find the basic and most fundamental block that form matter[10]. This analysis need to store large amount of data and calculations and need to store huge experimental results of the experiment and evaluate the result. Since the input is large no of files which are in a binary format we cannot directly use the Dryad .

To overcome the problem we have to divide the data to each node of the cluster manually and we have to use a data partition that consists of only file name present in a given node. Beginning with the analysis of data we firstly apply a function that is programmed in ROOT script to all of the input files. Since it can analyse various files at a same time we use a homomorphic filter .We then use a Apply operation that can be applied to a complete data set and result in outputs of multiple nature. After that a apply root method is used which iterate over group of data while executing script resulting in a histogram. The whole process is now repeated over this partially created histogram till we get a single histogram.

It compared the result of all of the application that are Hadoop ,Dryad ,CGL Map reduce by changing the amount of data .We found out that the Hadoop has high overhead in compare to other due to its storing technique. In Hadoop the data was placed in the IU data capacitor while in dryad and CGL the data was accessed from local disk and there the data was distributed before sending the data for computation purposes. Moreover the Hadoop and CGL Map reduce directly pass the histogram to the reducer where the data is stored in local disk and evaluated while in Dryad the partial histogram were stored in shared directory and its output were combined with that of second step in a separate analysis in fig. 4(a).

2.5 CAP3

CAP3 was developed by Xiaoqiu Huang is a DNA based Sequence assembly program. It performs many important assembling steps which include overlap computations, making of multiple sequence alignments and generating consensus sequences from them on a given set of gene sequences and uses base quality values for computing these. It reads its inputs which is nothing but a group of genes in the FASTA file format [12]. Input has a collection of files which are processed separately by the CAP3 program. The program also utilizes the constraints on forward-reverse direction to rectify assembly errors and link coatings.

For our study we have stored the input file name and other parameters in a external location, as these are only needed by Dryad for execution. It divided the input data files equally among the nodes of cluster while creating it we also stored the data partition information in every node containing the information about original data file names ,furthermore a metadata is created which point to each division of the data present in the node. The found out that Dryad uses no of data partition to calculate no of vertices for each node rather than CPU core for execution it automatically assume that PLINQ will automatically handle the CPU core division [14]. Since our data file were only the name it was not able to estimate the time it would take for evaluation .We found out a solution to this by changing the way the data is portioned by changing the data partition to 2 file name per record where 2 is because of presence of dual core in our computer. For further improvement the no of data file name per file could be increased

depending on no of core you are using. But we found out that if even one of the execution file is taking a long time the other file go in the wait state and results in time lag.

Whereas this problem can be tackled by using Hadoop as maximum no of map and reduce task while also setting the minimum no for the same which result in increased in concurrence nature of CPU cores in Fig. 4(b).

2.8 Cloud Burst

Cloud-Burst is one of the latest developed highly sensitive algorithm in parallel computing which is seed-and-extend read-mapping algorithm for optimized mapping of single-end next generation sequence of data of given referenced genomes. It's a map reduce implementation for short-gene alignment. It has the option to filter the alignments to report the best read among the given read. Cloud burst is a highly scalable algorithm. It has many advantages like it provides the ease of working with it on large remote compute clouds and hence can map a large number of reads with high sensitivity in comparatively less time.

- We have used a modified procedure for it by using 3 way map reducing :
- Firstly take the data called seeds.
- Calculate alignment for each seed instead of using all of the alignment present for computation with a difference of t where t is set by the user before shuffling of data.
- From the above step we get the seed which are best possible aligned which would greatly increase the execution time as time spent by the algorithm in reduction phase is greatly reduced .
- Then we again calculate the alignment for each of new seed in addition with all previous seeds which would result in a even better alignment.
- The data is then finally sent to reducer which are further analysed.

We found out that Hadoop was faster than Dryad .This difference occur arise before the reduce phase as in Hadoop it divides the data into more partition in table 1 .For e.g. if there are 10 nodes Hadoop [15] will divide the data into 20 partition for dual core computer while Dryad will divide it in only 10 vertices which is half of the no of partition created by Hadoop thus the difference in speed. From the below graph we were able to choose the best technology for handling huge amount of data and it is CGL MapReduce and Hadoop which can be said to be of similar in performance and execution time .These algorithm were faster than the Dryad and has a good edge over it .This difference was found mainly due to storing mechansim of the algorithm and the data partition method of the techniques in Fig.4(c).

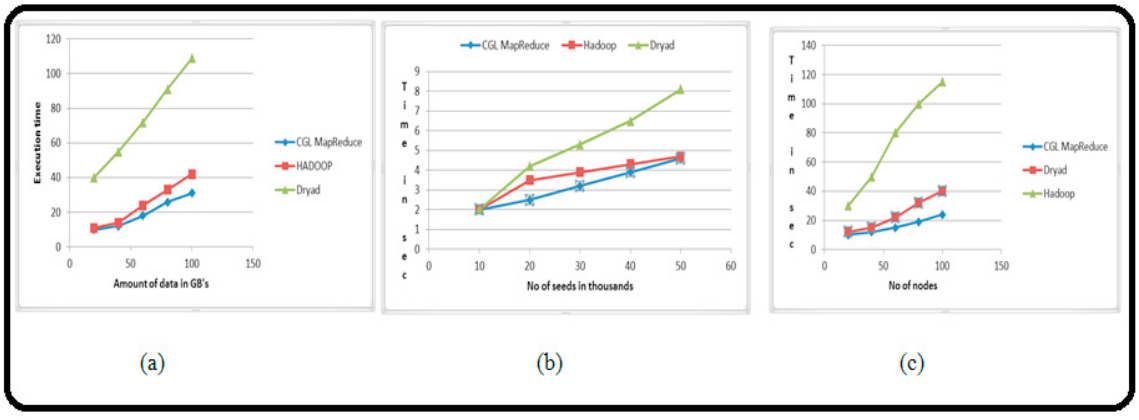

Fig. 4. (a) HEP (b) CAP3 (c) Cloudburst

3. Comparison for Different Technologies

Table 1.DifferentTechnologies : Hadoop,Dryad, CGL.

	Hadoop	Dryad	CGL Map reduce
Deployment	Master and worker nodes are present in a cluster to perform jobs like tracking(job or task).	Computing and data should be close so data storage and computing are arranged at the same node.	Same as dryad ,data storage and computation done at the same node.
Technology/Language support	Java or java services	Programmed by C#	C++, C, Python
Programming model	Map Reduce	Execution based on Directed Acyclic Graphs(DAG)	MapReduce with combine phase
Scheduling	FIFO scheduling	Optimizes graphs based on Network topologies	Data locality
Handling errors and failure	Again executes maps and reduces tasks	It again executes vertices of graphs	Not implemented
Monitoring	Web based user interface monitoring	-	Programming based progress monitoring
Execution environment	Future Grid, Linux cluster	Windows HPCS cluster	
Communication	Point to point using Http	TCP pipes Files	Content Distribution Network
Data storage	HDFS	NTFS, Cosmos DFS	Google, KFS, Amazon S3
Data handling	HDFS	Local disks or shared directories	Local disks or shared file system

4. Conclusion

After performing the experiments and research on various different cloud computing technologies we found that in most of the cases Hadoop has an edge over Dryad and CGL MapReduce .The execution time of Hadoop in most application oriented work is faster than the other techniques which are present this is mainly due to the reason that the no of partition created in Hadoop is greater than any other techniques which significantly reduce the time in the reduction phase of the computation. We also came to the conclusion that when the data is in huge amount CGL MapReduce and Hadoop are almost equivalently good but Dryad is very slower than both of these and this is mainly because of the storing mechanism of the Dryad as discussed earlier in the HEP application.Moreover in terms of user friendliness we found out that Hadoop offered more user friendliness and is easy to use in comparison to other techniques present. The one field in which more advancement can be done is that the online storage mechanism can be improved which will allow better handling and faster transfer time of the data for computation and analysing purposes.

References

1. Ekanayake J, Pallickara S, Fox G(2008). “Mapreduce for data intensive scientific analyses”,IneScience,. IEEE Fourth International Conference on 2008 Dec 7,IEEE: 277-284.
2. Pallickara S, Ekanayake J, Fox G. Granules(2009). “A lightweight, streaming runtime for cloud computing with support, for map-reduce”. InCluster Computing and Workshops: CLUSTER'09. IEEE International Conference on 2009 Aug 31, 1-10.
3. Ekanayake J, Fox G(2009). “High performance parallel computing with clouds and cloud technologies” InInternational Conference on Cloud Computing 2009 Oct:Springer, Berlin, Heidelberg 19: 20-38.
4. Murray DG, Hand S (2010). “Scripting the Cloud with Skywriting”. HotCloud, 22(1),10- 12.
5. Qiu X, Ekanayake J, Beason S, Gunarathne T, Fox G, Barga R, Gannon D (2009). “Cloud technologies for bioinformatics applications”. InProceedings of the 2nd Workshop on Many-Task Computing on Grids and Supercomputers 2009 Nov 16, 6.
6. Jiang W, Ravi VT, Agrawal G(2010). “A map-reduce system with an alternate api for multi-core environments”. InProceedings of the 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing 2010 May 17, 84-93.
7. Sehrish S, Mackey G, Wang J, Bent J. Mrap(2010) “ A novel mapreduce-based framework to support hpc analytics applications with access patterns”. InProceedings of the 19th ACM International Symposium on High Performance Distributed Computing 2010 Jun 21, 107-118.
8. Murray DG, Schwarzkopf M, Smowton C, Smith S, Madhavapeddy A, Hand S. CIEL (2011). “A universal execution engine for distributed data-flow computing”. InProc. 8th ACM/USENIX Symposium on Networked Systems Design and Implementation 2011 Mar 30 , 113-126.
9. Kannadasan R, Saleembasha MS, Emerson IA.(2016) “A Frame Work for Learning Drug Designing through Molecular Modelling Software Techniques and Biological Databases for Protein-Ligand Interactions.” International Journal of Engineering Research in Africa. 1(27): 326-343.
10. Prabakaran N, Kannan RJ.(2017) “Sustainable life-span of WSN nodes using participatory devices in pervasive environment”. Microsystem Technologies: 23(3):651-7.
11. Kannadasan R, Saleembasha MS, ArnoldEmerson I.(2015) “Survey on Molecular Cryptographic Network DNA (MCND) Using Big Data.” Procedia Computer Science: 1(50) :3-9.
12. Prabakaran N, Naresh K, Kannan RJ (2014). “Fusion centric decision making for node level congestion in wireless sensor networks”. InICT and Critical Infrastructure: Proceedings of the 48th Annual Convention of Computer Society of India,Springer, Cham: 321-329.
13. Borthakur D. The hadoop distributed file system(2007). “Architecture and design. Hadoop Project Website”,21.
14. Dittrich J, Quiané-Ruiz JA(2012). “Efficient big data processing in Hadoop MapReduce”, Proceedings of the VLDB Endowment,5(12): 2014-5.
15. O’Driscoll A, Daugelaite J, Sleator RD(2013). “ ‘Big data’, Hadoop and cloud computing in genomics”. Journal of biomedical informatics. 46(5):774-81.