HPCC System and its Future Aspects in Maintaining Big Data

Tarun Goya
Dept. of Computer Science,
Govt. Engineering College Bikaner, Bikaner, Rajasthan, India
tarungoyal.it@gmail.com
Ved Prakash Upadhyay
Dept. of Information Technology
Govt. Engineering College Bikaner, Bikaner, Rajasthan, India
tarungoyal.it@gmail.com
Bhartendu Shrimali
Dept. of Information Technology
Govt. Engineering College Bikaner, Bikaner, Rajasthan, India
tarungoyal.it@gmail.com

----------ABSTRACT----------

This research paper will make a complete overview of HPCC system and its future aspects in the field of maintaining big data. HPCC (High Performance Computing Cluster) system is developed and designed by LexisNexis. The main benefit of HPCC system is it have its own language ECL which make it faster and better than its competitors. Definitely HPCC will be a future player in the field of big data. As big data is the data in huge amount and maintenance such huge data is the main problem of the today's world.

Keywords—HPCC, big data, clusters, cloud computing, Hadoop, ECL and ROXIE.

I. INTRODUCTION

Today we live in the era of data, where we can find all the information in the form of data. As a result of information explosion, many organization have large amount of data, which they have collected from different sources and are stored in massive datasets. Now they data need to be processed and analyzed to provide business intelligence, to improve product and service for customers, to make improvement in working of organization or to meet other internal data processing requirement.

These are all data-intensive computing requirements which can be only addressed by a scalable system which is based on some hardware clusters of commodity servers which are coupled with system software to provide a distributed file storage system, online query capability, parallel programming development tools, job execution environment, instant response, and parallel application processing.

For the last few years we were solely dependent on Apache HADOOP, an open-source software framework for distributed processing and distributed storage of big data on clusters of commodity hardware. But now we have HPCC (High-Performance Computing Cluster), which is also known as DAS (Data Analytics Supercomputer) [1], is an open source, data-intensive computing system platform developed by lexis nexis.

The HPCC system incorporates a software architecture which is implemented on commodity computing clusters to provide high-performance, data-parallel processing for the application by the utilization of big data. The HPCC System provides all of the capabilities in an integrated, easy to implement and use, open source high-performance computing environment [1]. The HPCC system includes a system configurations to support both parallel batch data processing called THOR and high performance online query applications using indexed data files called ROXIE. The HPCC system also having a data-centric declarative programming language for parallel data processing called ECL[7].

II. OVERVIEW OF HPCC SYSTEM

A. HPCC Platform and Architecture

The architecture of HPCC system is having two distinct cluster processing environment. The first platform is called data refinery who is responsible for consuming large amount of data, Transform and Load

processing of the raw data, transforming, linking and indexing the data. This data refinery is known as THOR [4] [6].

The second platform is called Roxie and functions as a rapid data delivery engine [6]. It is just like an online high-performance structured query platform, which analysis data.

The HPCC system incorporates both the Thor and Roxie clusters and this layer is known as supercomputer layer. Also it have some other parts known as middleware layer, end user services, management tools, external communication layer, ECL Watch and auxiliary components.

B. *Cluster Types*

1) THOR: The thor cluster is very similar in its working, file system, environment of execution and capabilities to the HADOOP MapReduce, but it also offers significantly better performance in the same equivalent configuration [6]. In Thor platform we are having slave and master nodes. There is one master node and n slave nodes. In addition to these two nodes, here also a common and an auxiliary component are needed to the implementation of a complete HPCC system processing environment. Here in the Thor cluster we can execute multiple jobs in parallel for the programs which are written in ECL.

Here the distributed file system we used in the Thor cluster is record oriented and it is somewhat different from the block format used in a Map Reduce clusters. All the records that we use may be fixed length or in variable length, and also they support a variety of standards and also having custom formats which includes child datasets. When we want to load a file in Thor cluster we first transfer it to a landing zone from the real location of file. When we finish the transfer of file we have to do a process called “spraying” is used to partition the file and then we load the file to the nodes of a Thor cluster. It is very similar to Map which we do in MapReduce. To get back the output of a stored file we have to do a process called “dispraying”. The indexed file which are generated in the Thor cluster are directly copied to the ROXIE cluster which support online queries[4][5].

Here we also have a server known as Dali server in which we store the Name services and storage of metadata about files including record format information of THOR distributed file system.

2) ROXIE: The Roxie cluster is nothing but an online query processing platform, which consists a configurable number of peer-coupled nodes. Here in Roxie cluster DFS is a distributed index based file system which use a structure of B+ tree for storage of data. The indexes are directly copied from the Thor cluster to the Roxie cluster. Here the data associated with the logical index keys are added to the index structure as payload. The payload can have any type of structured or unstructured data which are supported by ECL language. Also the index keys may be multivariate and multi field too[4].

In Roxie cluster we use the concept of agents and servers. The server process will wait for a query request by the web interface and then determines the nodes and associated agents with it processes that are having the data we needed for the query[4]. The Roxie query request can also be submitted by SOAP calls, HTTP protocol request from a web application, or through a socket connected to it directly. The every Roxie request have its ECL query program associated with it too. The size of Roxie cluster is always smaller then Thor cluster and vary according query processing throughput and response time requirements.

C. *ECL Language*

Several well-known big data companies understand the need of a separate language for data processing for example, in MapReduce we use C++ and while in Yahoo’s open source project we used java and also we used some other language for different language for different platforms[3].

Similarly here in HPCC system we uses a different language called ECL (Enterprise Data Control Language). The ECL language is specially designed for data intensive application mostly used for entrepreneur and also some time in government use.

ECL language is the most important aspect of the HPCC system it makes the system more flexible and also make its capabilities increase. It is a data centric, high level and highly optimized language which allows the programmer to define that what the data processing result should be. Here in this language the execution is not determined by the order in the language statements came but from the transformation and dataflow sequence represented in the statements of the language ECL uses a syntax that is very similar to other familiar language but it increases the reusability and demote the coding. The code we use in ECL is 20 times lesser then what we use in C++ and java. The basic unit of an ECL code is known as attribute. Here an attribute can contain the complete query or a segment of code of the query[3].

1) Key benefits of ECL:

- ECL provide the functionality of parallel data processing and it reduces the complexity in computing large size of computing cluster.
- With ECL we can implement large amount of data because it is mainly designed for manipulation of large amount of data and query.
- ECL have a productivity improvement of 20 times in comparison to other programming language such as java and C++.
- ECL is a high level, parallel programming language which is powerful enough to do, information extracting, information storing, information retrieval, and linking of different records.
- ECL is very mature language but it is still making advancements in itself every day.

D. HPCC System Servers Middleware

In HPCC we have a number of system servers includes in system configuration, these servers provide a gateway from its two data cluster to the outside world[2]. These servers include the ESP server, ECL server, Dali server, DFU server and they all are known as middleware components because they reside in the middleware layer of HPCC system configuration.

III. HPCC PERFORMANCE

The performance of HPCC system is far better in comparison to HADOOP system and the one main reason behind it that HPCC system have a special language defined for it which increase its capabilities over HADOOP[9][10].

Also we find that a comparison done by Terabyte Sort Benchmark (which is managed by leading industry groups Microsoft and HP) gives us the result that for a 400 processing nodes cluster HADOOP system takes 6 minutes 45 seconds to create the test data while with same configuration of HPCC system finish this task with more efficiency and in less time of only 2 minutes and 35 seconds to create the same test.

So it is very clear that with same configuration the efficiency and speed of HPCC system is much better than the HADOOP and other similar systems.

IV. BENEFITS OF HPCC OVER HADOOP

We have few benefits of choosing HPCC system over HADOOP and these are listed below [9].

1) Programming language: The main benefit of choosing HPCC system over HADOOP is that it have a very powerful, extensible, declarative programming language called Enterprise Control Language. Which provide HPCC system more speed and increase its capabilities over HADOOP. LexisNexis claims that it enables the developer to use the data according to their wish not by giving the system step by step instructions.

2) ROXIE Delivery Engine: It increases the speed of completion of data queries. It allow user to run real time queries against HPCC. This seems to be an advantage over HADOOP which is a batch oriented and used for rear view mirror analysis. Roxie engine returns the queries back in just few seconds.

3) Easy To Implement And Truly Parallel: The HPCC system is very easy in implantation in comparison to HADOOP. Also the HPCC system is truly parallel in nature not like HADOOP where in HADOOP MapReduce every complex data have to wait for previous wait to complete the first one.

CONCLUSION

So it is very clear that in terms of the Big Data big picture, HPCC creates another Big Data "fork." From all over research we find out that HPCC system is very suitable for entrepreneurs who have to work with big data. It is a system with very powerful programming and having all the functionality to being accepted by big data world. So we can say that it is just like the future of big systems.

ACKNOWLEDGEMENT

We like to acknowledge our thanks to LexisNexis Risk Solutions for providing valuable data over internet related to HPCC systems developed by them for managing Big Data & its issues. We also want to thank Dr. Hardayal Singh Shekhawat, Associate Professor and Head, Department of Information Technology, Govt. Engineering College Bikaner, Bikaner, Rajasthan, India for providing us valuable suggestions during our work and also for continuously motivating us to do some work in the field of Big Data. Also want thank our friends and colleagues for their support and motivation.

REFERENCES

[1] B. Furht, and A. Escalante, "Handbook of Data Intensive Computing," Springer New York Dordrecht Heidelberg London, pp. 3-24, Dec 2011.

[2] A. M. Middleton, "HPCC systems:Introduction to HPCC," LexisNexis Risk Solutions, pp. 13-53, May 2011.

[3] J. Kelly, "LexisNexis HPCC Takes on Hadoop as Battle for Big Data Supremacy Heats up," Wikibon Blog, June 2011.
[4] D. Bayliss, "HPCC systems: Enterprise Control Language," LexisNexis Risk Solutions, pp. 3-5, May 2011.
[5] F. Tekiner and A. J. Keane, "Big Data Framework," 2013 IEEE International Conference, pp. 1494-1499, Oct. 2013.
[6] V. Shukla and P. K. Dubey, "Big Data: Moving Forward with Emerging Technology and Challenges," IJARCSMS, Vol. 2, Issue. 9, pp. 187-193, Sep 2014.
[7] R. Buyya, "High Performance Cluster Computing: Architectures and Systems," Prentice Hall PTR Upper Saddle River, NJ, USA, 1999.
[8] HPCC Systems [URL]: http://hpccsystems. com/.
[9] HPCC takes on Hadoop's Big Data Dominance [URL]:http://www.cio.com/article/2388441/big-data/hpcc-takes-on-hadoop-s-big-data-dominance.html
[10] Shilpa and M. Kaur, "Big Data and Methodology-A review," IJARCSSE, Vol. 3, Issue. 10, pp. 991-995, oct 2013.