International Journal of Emerging Technology in Computer Science & Electronics (IJETCSE)
ISSN: 0976-1353 Volume 25 Issue 4 – APRIL 2018.

ENHANCED DATA ARCHIVAL AND RETRIEVAL SYSTEM USING REED SOLOMON CODING

G.Brajith kumar [#1] and V.Karthi [*2]

[#] *PG Scholar, Dept. of CSE, J.K.K Nataraja college of engineering and technology komarapalayam, India*

[*] *Asst. Prof., Dept. of CSE, J.K.K Nataraja college of engineering and technology komarapalayam, India*

***Abstract*— Very large data sets within the range of megabytes to terabytes generated daily from checkpoint-and-restart processes are seen in today's scientific simulations. Reliability and durability are two important factors to build an archive storage system. We propose enhanced data archival and retrieval process using reed Solomon encoder process. The proposed algorithm composes of two phases, namely, data archival and data retrieval stage. The task of data archival process is to fetch data in a distributed manner. Then the selected data is encoded using reed Solomon coding systems. The encoded data is decoded only by authorized users. For the requested file, the cloud server checks the data blocks across multiple nodes and then retrieve the original source nodes. If any nodes get collapses, the reconstruction module repairs the nodes and also preserves the originality of the data. Experimental analysis proves the efficiency of the system in terms of data originality, reconstruction accuracy and data integrity.**

***Index Terms*— Reliability, Data archival, Encoded data, Multiple nodes, Data originality and Big data.**

I. INTRODUCTION

In the recent past, there has been a widespread growth in the use of cloud infrastructures. The major reason for this growth is that in general, it is more efficient and less expensive to host applications on the cloud. Since these considerations apply also to Big Data systems such as Hadoop, it is important to support them efficiently on the cloud. For example, a major factor driving the adoption of cloud technology is resource sharing [Cr2009]. Since the demands of applications are typically bursty, it is possible to share the same server resources between multiple applications, leading to lower costs. Other advantages include the ability to scale server resources rapidly, and to have large spare capacity [Ar2010] [1]. These factors indicate that it is important to support Big Data systems efficiently on cloud infrastructures.

To excel and to be more successful, the best way is to retrieve the large amount of data in the shortest possible time, and to take data driven decisions which on an average are 5% more productive the proponents and the users of this system are 6% more successful than others. World renowned MIT [2] has done an in depth study and found that undertakings/corporates which used big data analytics were in the top. This was borne out by the fact that companies that were in the top third tier of their industry in terms of the use of data driven decisions were, were more productive with effective use and maintenance of big data which gave them an edge over competitors. All this comes at a huge cost as both retrieval and maintaining such huge amount of data and then using it effectively comes at a huge cost. The best way to utilize big data analytics is to use the best method of data archiving and later onto use the cheapest and the most effective method of retrieving it.

The traditional approach to data archiving was to move the information to cheaper secondary storage, such as tapes and optical disks. With the advent of Big Data, traditional methods of data archival are being revaluated. Typically, once the data is archived, it is never accessed again. So, despite its tremendous potential value, in many ways, traditional archival systems spell death for data usability. A close second was a technological issue: dealing with what has become known as the three 'V's' of Big Data [3]: volume, velocity and variety. Unstructured data includes video, audio, images, weblogs, and so on. This data can then be retrieved whenever required. But data cannot be deleted as it needs to be retained for legal compliance or analytics. Many countries have laws requiring businesses to keep records for as long as five to seven years. Businesses often face the additional burden of maintaining legacy storage systems, solely for data accessibility.

The rest of the paper is organized as follows: Section II describes the related work; Section III presents the proposed work; Section IV presents the experimental analysis and concludes in Section V.

II. RELATED WORK

This section presents the prior techniques of our study domain. Hadoop Map Reduce is a large scale, open source software framework dedicated to scalable, distributed data intensive computing [4]. The framework breaks up large data into smaller parallelizable chunks and handles scheduling

- Maps each piece to an intermediate value
- Reduces intermediate values to a solution
- User-specified partition and combiner options Fault tolerant, reliable, and supports thousands of nodes and petabytes of data

- If you can rewrite algorithms into Mapreduce, and your problem can be broken up into small pieces solvable in parallel, then Hadoop's Map Reduce is the way to go for a distributed problem solving approach to large datasets
- Tried and tested in production
- Many implementation options.

The author [5] stated the importance of some of the technologies that handle Big Data like Hadoop, HDFS and Map Reduce. The author suggested about various schedulers used in Hadoop and about the technical aspects of Hadoop. The author also focuses on the importance of YARN which overcomes the limitations of Map Reduce. The author [6] have surveyed various technologies to handle the big data and there architectures. They discussed about big data characteristics (volume, variety, velocity, value, veracity) and various advantages and a disadvantage of these technologies. They have also discussed an architecture using Hadoop HDFS distributed data storage, real-time NoSQL databases, and MapReduce distributed data processing over a cluster of commodity servers. The main goal was to make a survey of various big data handling techniques those handle a massive amount of data from different sources and improves overall performance of systems. The author continue with the Big Data definition and enhance the definition given in [7] that includes the 5V Big Data properties: Volume, Variety, Velocity, Value, Veracity, and suggest other dimensions for Big Data analysis and taxonomy, in particular comparing and contrasting Big Data technologies in e-Science, industry, business, social media, healthcare. With a long tradition of working with constantly increasing volume of data, modern e-Science can offer industry the scientific analysis methods, while industry can bring advanced and fast developing Big Data technologies and tools to science and wider public.

The author [8] stated the need to process enormous quantities of data has never been greater. Not only are terabyte - and petabytes scale datasets rapidly becoming commonplace, but there is consensus that great value lies buried in them, waiting to be unlocked by the right computational tools. In the commercial sphere, business intelligence, driven by the ability to gather data from a dizzying array of sources. Big Data analysis tools like Map Reduce over Hadoop and HDFS, promises to help organizations better understand their customers and the marketplace, hopefully leading to better business decisions and competitive advantages. The author [9] stated there is a need to maximize returns on BI investments and to overcome difficulties. Problems and new trends mentioned in this article and finding solutions by combination of advanced tools, techniques and methods would help readers in BI projects and implementations. BI vendors are struggling and doing continuous effort to bring the technical capabilities.

The author [10] describes the concept of Big Data along with 3 Vs, Volume, Velocity and variety of Big Data. The paper also focuses on Big Data processing problems. These technical challenges must be addressed for efficient and fast processing of Big Data. The challenges include not just the obvious issues of scale, but also heterogeneity, lack of structure, error -handling, privacy, timeliness, provenance, and visualization [11], at all stages of the analysis pipeline from data acquisition to result interpretation. These technical challenges are common across a large variety of application domains, and therefore not cost –effective [12] to address in the context of one domain alone. They described a Hadoop which is an open source software used for processing of Big Data. The author [13] proposed system is based on implementation of Online Aggregation of Map Reduce in Hadoop for ancient big data processing. Traditional MapReduce implementations materialize the intermediate results of mappers and do not allow pipelining between the map and the reduce phases [14]. This approach has the advantage of simple recovery in the case of failures, however, reducers cannot start executing tasks before all mappers have finished. As the Map Reduce Online is a modeled version of Hadoop Map Reduce, it supports Online Aggregation and stream processing [15], while also improving utilization and reducing response time.

III. PROPOSED WORK

This section presents the working process of our proposed systems. An enhanced reed Solomon coded archival technique is proposed. This work concentrates on facilitating high resiliency for frequently used data. The proposed model composes two stages, namely, archival and retrieving stage.

A. Archival stage:

Archival stage is the first process that consists of data fetching, reed Solomon encoder and distribution. Each process holds its own unique characteristics. The steps involved in archival storage are:

a) Data fetching process helps to select the files from the server and then fed as input to the data program.
b) Reed Solomon encoder: The selected file is segmented into multiple blocks for easy analysis purpose. Parity block is generated and take data inputs. Then, the coded blocks are generated.
c) Distribution phase: The coded blocks are distributed over multiple nodes. By doing so, the reliability of the data system is achieved.

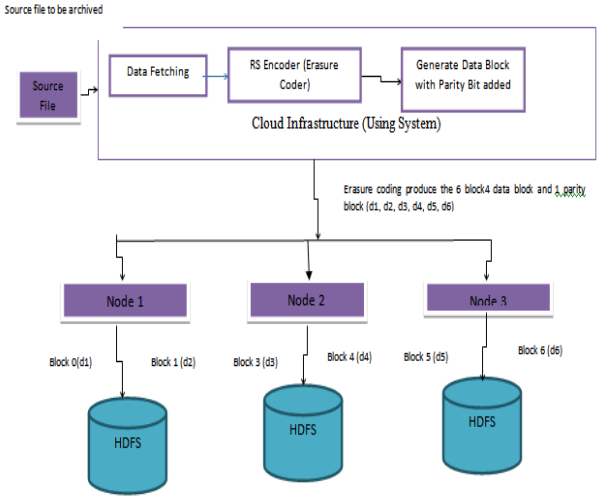

Fig.1 Process involved in data archival systems

B. Retrieving stage:

Retrieving stage is the second stage that composes of user request, cloud server and RS Decoder.

a) User request: The users have to register with the system, so as to assess the data operations such as reading, writing and sharing services. The user data will be stored in an index manner. Based on the entered keywords, the data is requested to the cloud server.

b) Cloud server: Based on the received requests, the cloud server start checking the nodes. It begins by checking each data blocks.

c) RS Decoder: Once the appropriate data blocks are found, the data block is decoded. Thus, the decoded data is send to the user.

d) Error phase: In some cases, the data block is collapsed. The original file has to be available for all cases. The construction of the node should be wiser for tolerating the errors. Thus, the concept of distributed blocks is used for avoiding the errors. Erasure coding technique is used for reconstructing the collapsed data.

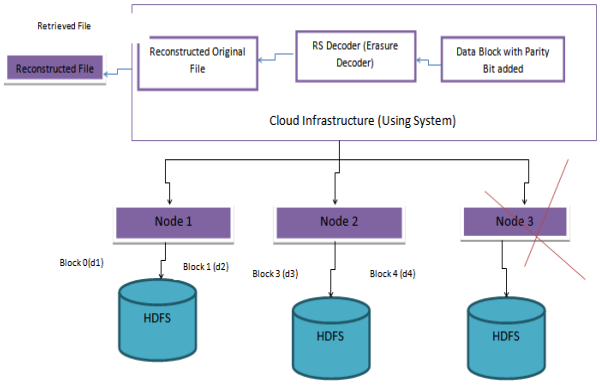

Fig.2 Process involved in Retrieving system

IV. EXPERIMENTAL RESULT AND ANALYSIS

This section depicts the experimental analysis of our proposed work. The proposed model is

Fig.3 Components presents in data archival process

Fig.4. Selecting the files and encoding the data

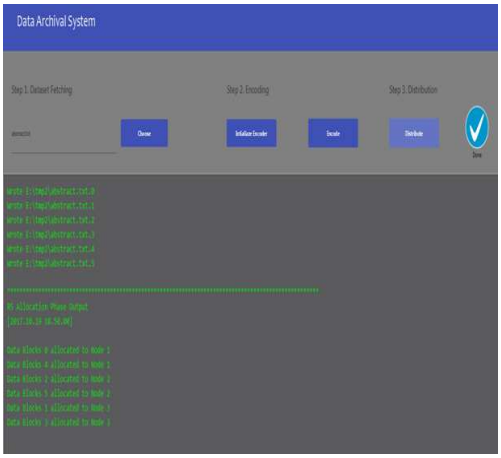

Fig.5. Data block creation and the nodes selection

Fig.6. Execution of nodes

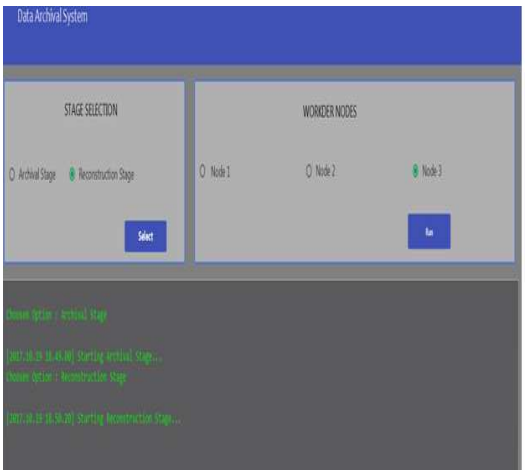

Fig.7. Components in reconstruction stage

Fig.8. Requesting the files

Fig.9. Searching operation is performed over the data blocks for reconstructing the data

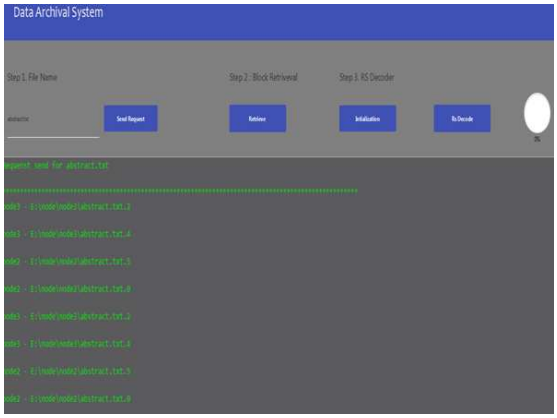

Fig.10 . Cloud server response to the requested file

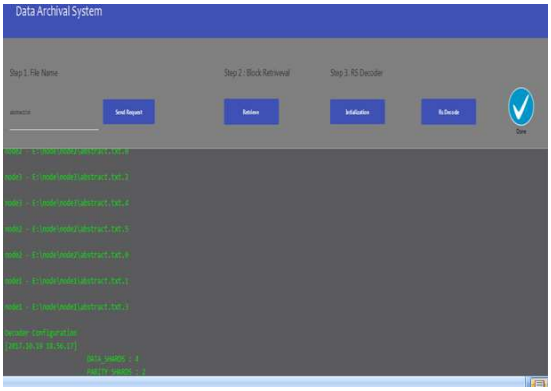

Fig.11. Decoder initialization process

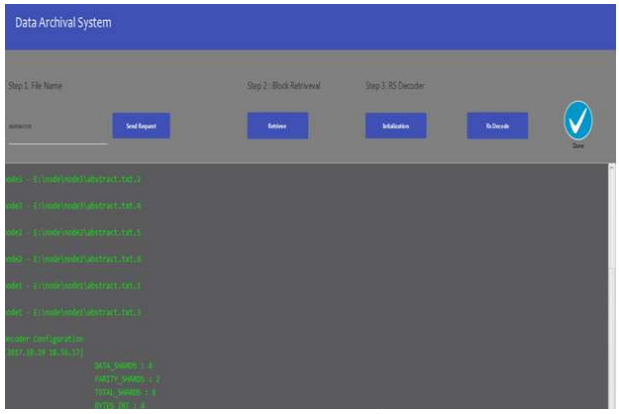

Fig.12. Decoding the data blocks for authorized users

V. CONCLUSION

Nowadays, the term big data has become very popular in Information Technology sector. Big-Data is a term for data sets that are so large or complex that traditional data processing tools are inadequate to process or manage them. Big data can be found in finance and business, banking, online and onsite purchasing, healthcare, astronomy, oceanography, engineering, and many other fields. This paper concentrates on enhanced data archival and retrieval process using reed Solomon encoder process. The proposed algorithm composes of two phases, namely, data archival and data retrieval stage. The task of data archival process is to fetch data in a distributed manner. Then the selected data is encoded using reed Solomon coding systems. The encoded data is decoded only by authorized users. For the requested file, the cloud server checks the data blocks across multiple nodes and then retrieve the original source nodes. If any nodes get collapses, the reconstruction module repairs the nodes and also preserves the originality of the data. Experimental results have shown the efficiency of our proposed systems.

REFERENCES

[1] James S. Plank, Erasure Codes for Storage Systems A Brief Primer, USENIX .login, Vol. 38 No. 6, 2013.
[2] Hsing-bung Chen, Ben McClelland, et al., An Innovative Parallel Cloud Storage System using OpenStack's Swift Object Store and Transformative Parallel I/O Approach, Los Alamos National Lab Science Highlights, 2013.
[3] Corentin Debains, Gael Alloyer, Evaluzation, Evaluation of Erasure-coding libraries on Parallel Systems, 2010.
[4] Peter Sobe, Parallel Reed/Solomon Coding on Multicore Processors, in Proceedings of International Workshop on Storage Network Architecture and parallel I/O, 2010.
[5] Babak Behzad, Improving parallel I/O auto tuning with performance modeling, in Proceedings of ACM International Symposium on High-performance Parallel and Distributed Computing (HPDC), 2014.
[6] Hsing-bung Chen, parEC – A Parallel and Scalable of erasure coding support in Cloud Object Storage Systems, Los Alamos National Lab.
[7] A. Varbanescu , On the Effective Parallel Programming of Multi-core Processors, Ph.D Thesis, Technische Universiteit Delft , 2010.
[8] William Gropp Ewing Lusk, Anthony Skjellum, Using MPI: Portable Parallel Programming with the Message-Passing Interface, The MIT Press, 2014.
[9] Hsing-bung Chen, Parallel Workload Benchmark on Hybrid Storage EcoSystem, Los Alamos national Lab.
[10] Adam Manzanares, John Bent, Meghan Wingate, and Garth Gibson, The Power and Challenges of Transformative I/O, in Proceedings of IEEE International Conference on Cluster Computing (CLUSTER), 2012.
[11] Hsing-bung Chen, Gary Grider, et al., Integration Experiences and Performance Studies of A COTS Parallel Archive System, in Proceedings of IEEE International Conference on Cluster Computing (CLUSTER), 2012.

[12] Gibraltar: A Reed-Solomon coding library for storage applications on programmable graphics processors, Concurrency and Computing: Practice and Experience, 2011.
[13] Jerasure: Erasure Coding Library, http://jerasure.org/
[14] Jianzong Wang and Lianglun Cheng, qSDS: A QoS-Aware I/O Scheduling Framework towards Software Defined Storage, in Proceedings of ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS), 2015.
[15] Peter Sobe, Parallel coding for storage systems - An OpenMP and OpenCL capable framework, Lecture Notes in Informatics, 2012.