ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 5, Special Issue 2, January 2018

Decentralized Erasure Coded Archival Method for Cloud File Systems

L. Maria Michael Visuwasam[1], Siddartha Mondal[2], S. Sudharsan[3], B. SunilKumar[4], A. Suriya[5]
[1]Assistant Professor-SG, Department of Computer Science and Engineering
[2,3,4,5]Students, Department of Computer Science and Engineering
Rajalakshmi Institute of Technology, Chennai.
mariamichael.l@ritcennai.edu.in[1]
siddart.mondal05@gmail.com[2], sundaramraghavan67@gmail.com[3], mailtosunil.25@gmail.com[4], suriyarm38@gmail.com[5]

***Abstract*:This paper reviews the existing disk-based archival storage systems which are not adequate for Hadoop clusters because of the cognitive content of data replicas and also the programming model of map-reduce. An oversized range of nodes lead to a high possibility of failures caused by unreliable components, software system glitches, machine reboots, maintenance operations. To guarantee high reliability and availability in the presence of various types of failures, data redundancies are commonly used in cluster storage systems. This paper counters and contrasts various methodologies used to solve the problems that arise in disk based archival system.**

Keywords- Disk based archival system, Hadoop clusters, map-reduce, unreliable, cluster storage systems.

I. Introduction

Petabytes of data are nowadays stored in large distributed storage systems [12] like the Google File System (GFS) [13], the Hadoop Distributed File System (HDFS) [38], and the Microsoft Azure Storage [39]. In 2012, two millions of search queries received per minute by 2014, that number has more than double. Every minute, 2.5 million pieces of content shared in Facebook, 50 thousand applications downloaded by Apple users, 200 million messages sent via Emails [40]. Such a massive amount of data demands large-scale storage systems maintained in data centers. With an increasing number of storage nodes installed, the data storage cost goes up dramatically.

Hadoop is an open programming framework based on java in which the processing of huge data sets are carried out in a Parallel and distributed computing environment. It makes Use of the commodity hardware Hadoop is Highly Scalable and Fault Tolerant. Hadoop runs in cluster and eliminates the use of a Super computer. Hadoop is the widely used big data processing engine with a simple master slave setup.

Big Data in most companies are processed by Hadoop by submitting the jobs to Master. The Master distributes the job to its cluster and process map and reduces tasks sequentially. But nowadays the growing data need and the competition between Service Providers leads to the increased submission of jobs to the Master. This Concurrent job submission on Hadoop forcesus to do Scheduling on Hadoop Cluster so that the response time will be acceptable for each job.

This survey studies various methods and approaches to efficiently store and retrieve data from an Hadoop distributed file system. An overview of systems is given and the different steps and parameters involved in building an efficient, fault tolerant and cost effective, system is developed. In section II, data replication storage issues are tackled by issuing codes to the data blocks that help in retrieving rarely accessed data 3X-Replica system is reviewed and is implemented for frequently accessed data, rarely accessed blocks are retrieved in the same way as being currently practiced. In section III with the advent of new retrieval techniques, pipelined erasure codes for quick data archival and fault tolerance frameworks for an energy-efficient storage is implemented. In section IV, the challenges in design of archival system are discussed and scope for this research are discussed in section V. The Approaches used and future possibilities are discussed in section VI.

II. EARLIER METHODOLOGIES USED IN STORAGE SYSTEMS

David G. Cameron Etal [1] have elaborately explained an environment suitable for the simulation of realistic Grid scenarios and the evaluation of Grid optimization algorithms and also pondered upon several strategies in scheduling and replica optimization and presented results showing their performance in the tests with the Grid simulator OptorSim. OptorSim is a simulation package written in java, developed to mimic the structure of a real Data Grid and study effectiveness of replica optimization algorithm in systems. They have displayed that the choice of strategies used, can alter the throughput of Grid jobs, and in some cases the strategies used affect the extent to which the Grid resources

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 5, Special Issue 2, January 2018

are exploited. In particular, their experiments show that Queue Access Cost, a scheduling algorithm which takes into account both the file access cost of jobs and the workload of computing resources, is the most effective at optimizing [33] computing and storage resources and reducing the average time to execute jobs. They have also proven that an effective and archival suitable choice of data replication strategy can enhance grid performance [32] for most situations, particularly with large numbers of jobs, the economy-based strategies and developed have the greatest effect, regardless of the presence of background (non-Grid) network traffic.

Pramod K. Meher and Jagdish C. Patra [2] introduced a new technique for secure distributed storage and dissemination of digital images using Chinese remainder theorem (CRT). The proposed technique not only involves significantly low computational complexity but also imposes various stages of difficulty to an intruder who intends to extract the original image from the encrypted sub-blocks. The proposed CRT-based technique is, therefore, useful for secure archival of large volume of image data and sharing of digital images by the users from different locations.

Mansaf Alam and Kashish Ara Shakil [3], the researchers have mentioned the helping developers of BDA Apps for cloud deployments. In their paper they have proposed a light-weight approach for distinguishing and surfacing variations between pseudo and large-scale cloud deployments, their approach makes use of the readily-available however scarcely used execution logs from these platforms. They have done a case study on 3 representative Hadoop-based BDA Apps and have shown that their approach can efficiently and quickly direct the attention of BDA App developers to the main variations between the two deployments.

The research work [7] outline challenges in analyzing big data for both data at rest and data in motion. They have described two kinds system for big data which is at rest namely NoSQL systems for interactive data serving environments and systems for large scale analytics based on MapReduce paradigm, such as Hadoop. They have discussed that the NoSQL systems are designed to have a simpler key-value based data model having in-built sharing, so this work in a distributed cloud based environment.

In a research paper [8], it has been proved that the cost of using 1000 machines for 1 hour, is the same as using 1 machine for 1000 hours, this means that a Hadoop job's performance can be improved potentially, while incurring the same cost and this proved Hadoop is put organized to achieve parallelism.

Zujie Ren Etal [4] provided a highly scalable and flexible platform for managing high scale data sets. The core of MapReduce lies in its ability to distribute data into different nodes and process it in a parallel fashion which is not transparent to its users. Hadoop [17] is an open source implementation of MapReduce [5] which has garnered immense popularity amongst different organizations such as facebook, Yahoo and twitter for running their data intensive [36] applications. Its popularity can be attributed to its ability to provide a highly scalable, parallel and fault tolerant framework that supports automatic distribution and computation across a cluster of commodity hardware [4]. Thus, hadoop is well equipped as a platform for carrying out large scale data analysis.

In [15] the authors simulated various replication and caching strategies in a tier-model Grid environment. In [16] they combined replication strategies with different scheduling algorithms and found that when using any replication strategy, taking into account the location of data when scheduling vastly improves the overall job performance. They found, however, that there was no marked difference in the choice of replication algorithm, perhaps because replication took place at the level of entire file sets (one file set defining a job) rather than individual files.

EDGSim [9] was designed to simulate the performance of the EU Data Grid but focuses on the optimization of scheduling algorithms. Analysis showed thatdata location was important in the scheduling decision, but no replication of data was taken into account.

In [11] a replication algorithm is tested which uses a cost function to predict cost effectiveness of replicas. It is found to be more effective in reducing average job time than the case where there is no replication. The simulation architecture used was based on a hierarchical model where leaf client nodes ran jobs but higher nodes contained all the storage resources, in contrast to the EU Data Grid architecture [35].

Erasure coding is widely implemented in distributed storage systems [14]-[17]. It is desirable for its efficient use of

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 5, Special Issue 2, January 2018

storage space given the same data availability [17]. There are several erasure coding techniques namely Reed-Solomon [19], LDPC [20], Tornado [21], LT [22], Raptor [23], each having its own class of applications. For instance, in a Bit Torrent - like file sharing system, the Tornado code is used to generate encoded blocks without the need to predefine the number of blocks to generate. With Tornado code, the receiver keeps receiving encoded blocks until it can decode to get the original file. Usually the number of encoded blocks required to decode in this case is high. In applications where this number needs to be small, Reed-Solomon is more appropriate.

Redundant data blocks maintenance is critical in many large-scale reliable distributed storage systems [18]. In these systems [23], server failures are inevitable and the lost encoded blocks need to be replaced and recovered to maintain the data availability and integrity over time. Several techniques have been proposed to make the maintenance more efficient. For example, FARM [24] stored multiple copies of encoded blocks but because of its randomness in choosing where to store these copies, the time to search for the missing blocks might be significant and sometimes inefficient. CFS [25] distributes encoded blocks across a number of servers which are organized in a Chord [26] overlay network.

III. RECENT TRENDS IN STORAGE SYSTEMS

Goodson et al. proposed a coding scheme that handles Byzantine and crash-stop failures [6]. In particular, they design a consistency protocol for asynchronous environments via versions. Their protocol bares similarities with our progressive scheme in that it uses RS codes and accounts for data integrity. However, it does not address the communication and computation aspects of progressive data retrieval when there are failures in the distributed storage system. Instead, coded fragments are retrieved one time, and deemed one of three states: complete, incomplete, or repairable, depending on the degree of errors in the system.

Lluis Pamies – Juarez, Etal [27] Distributed storage systems used in data centers have started to adopt a hybrid strategy for redundancy, where replicas of the newly inserted data are created, while erasure codes are preferred for archival of the same data once it does not need to be regularly accessed anymore, but still needs to be preserved. The number of replicas is then reduced as the data gets older. The use of erasure coding for archival [34] increases the fault tolerance of the system while reducing storage overheads with respect to replication [5], though replication remains so far the best form of redundancy for new data since it is likely to be frequently manipulated.

The most relevant related work is that of Fan et al. [12], who proposed to distribute the task of erasure coding using the Hadoop infrastructure, as MapReduce tasks [37]. Any individual object is however encoded at a single node, and hence the parallelism achieved in their approach is only at the granularity of individual data objects. Decentralized erasure coding has also been explored in the context of sensor networks [10].

In [28] the authors proposed a resource efficient scheduler that processes MapReduce jobs into heterogeneous storage containers. STAS introduces the use of tag jobs, heterogeneous shared-queues, heterogeneity-aware STAS manager and use of heterogeneous storage media as container volume. STAS reduces processing latency of locating and pairing dataset into heterogeneous storage media and optimizes the consumption of resources in Hadoop cluster.

In [29] the authors proposed a general energy-efficient framework that can be integrated with different fault tolerance mechanisms. Additionally, the study used an economic evaluation, which takes into consideration the cost of hardware failure in applying energy saving schemes, proposed by Chen et al. In order to evaluate the capability of fault-tolerance mechanisms, they implemented the solutions on a simulator.

In [30] the authors argued for the need for a caching system specifically developed for erasure-coded data, providing high availability with low latency and without the need to store full object replicas. They designed and implemented Agar, a caching system tailored for erasure-coded data, and explained how it integrates with a typical storage system. Agar uses a dynamic programming approach inspired by the Knapsack problem to optimize the cache configuration under a given workload. They also compared their prototype with the LFU and LRU strategies and showed that Agar consistently outperforms them, obtaining 16% – 41% lower latency.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 5, Special Issue 2, January 2018

TABLE I : COMPARISON OF VARIOUS METHODS USED IN STORAGE SYSTEMS

Proposals	*Used Methodology/ Algorithms*	*Advantages*	*Performance*	*Cost*	*Limitation*
Pipelined Erasure Codes for Fast Data Archival in Distributed Storage Systems [41]	Rapid RAID codes	Distributes the network and computing load of single-object encodings among different nodes and provides fast archival without compromising.	Uses cluster of 50 nodes and a set of Amazon EC2 instances, usually fast.	Storage cost high.	Distributed storage systems often store newly introduced data using replication. Slows down the whole archival process.
Optimizing Remote File Access for Parallel and Distributed Network Applications [42]	Smart File Objects(SFO)	The SFO uses application and network information to adaptively pre fetch and cache needed data in parallel with the execution of the application to mitigate the impact of the network.	Pre fetches data using network information, faster than RAID codes.	Cost very high, special hardware required for optimizing remote file access.	The SFO hides all details of the remote file from the application
T*: A Data-Centric Cooling Energy Costs Reduction Approach for Big Data Analytics Cloud [43]	A Data-Centric T* Cost Reduction Approach	T* takes a novel, data-centric approach to reduce cooling energy costs and to ensure thermal-reliability of the servers.	Working is data centric, performance on par with RAID not as fast as SFO.	Cost higher than RAID and SFO, dedicated hardware required to ensure thermal reliability of servers.	Bandwidth can be 8-20x higher than inter-rack bandwidth in these clusters and the huge data sizes of Big Data, render sending data to computations infeasible.
Highly Reliable Two-Dimensional RAID Arrays for Archival Storage. [44]	2D RAID Arrays	The data will remain largely unmodified once they are stored. For this reason, write rates are a much less important issue than in conventional storage systems.	Stored data largely unmodified, performs faster than RAID.	Costlier than RAID, expensive technology such as flash drives are used unlike disk drives in RAID.	The reason for the shut-down is the high read load that is necessary since super groups are quite large and the system is not expected to support both workloads.

IV. CHALLENGES IN DESIGN OF ARCHIVAL SYSTEM

- Large scale implementation is not viable with the current hardware modules.
- High speed storage modules are expensive and not scalable.
- Cold data retrieval has to undergo decoding of the encoded data, results in performance slow down.
- Data block failures are still prevalent in unavoidable circumstances.

V. SCOPE FOR THIS RESEARCH

The scope of this research is to develop a data reconstruction system to deal with block failure issues on Hadoop clusters. Grouping and pipelining strategies are applied to the reconstruction system to speed up the reconstruction process.

VI. APPROACHES USED AND FUTURE WORK

Over the past two decades, there have been varying size of data that are uploaded every second in internet. To process that data efficiently various approaches are carried out to tackle the problem of performancedegradationdue to dumping of overloaded data. There have been differing advantages and limitations in each method that were carried out. Table I lists a few of the recent work in this field, while outlining

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com
International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 5, Special Issue 2, January 2018

the advantages, limitations and the methods used. These methods have dealt with complexities and dependencies like failures caused by unreliable components, software glitches, machine reboots, maintenance operations, etc. As a future research direction,

TABLE II : COMPARISON OF ARCHIVAL SYSTEMS				
STORAGE SYSTEMS	ARCHIVAL	ERASURE CODE	HDFS	PIPELINE
DeepStore [45]	✓	✗	✗	✗
Venti [46]	✓	✗	✗	✗
ScaleRS [47]	✗	✓	✗	✗
[D+P]cd [48]	✓	✓	✗	✓
RapidRAID [41]	✓	✓	✗	✗
HDFS-RAID [49]	✗	✓	✓	✗
HDFS-EC [50]	✗	✓	✓	✗
aHDFS	✓	✓	✓	✓

a data reconstruction system to deal with block failure issues on Hadoop clusters can be developed. To optimizereconstruction performance, a way of choosingintermediate parity blocks to be kept in the local key-value store will be carried out.

VII. CONCLUSION

The paper analyzed existing disk-based archival storage systems in HDFS clusters and does not discuss any implementation results. Based on the limitations in the existing archival process, this paper proposes an enhanced and fault tolerant data archival framework using latest technology like erasure codes, agar, HDFS grouping, HDFS pipelining, etc. to improve Big Data processing in MNCs.

ACKNOWLEDGMENT

We are grateful to extend our sincere gratitude to our organization and to Mr. L. Maria Michael Visuwasamfor sharing the pearls of wisdom to complete this project.

REFERENCES

[1] David G. Cameron, Rub´en Carvajal-Schiaffino, A. Paul Millar, Caitriana Nicholson, Kurt Stockinger, Floriano Zini, Evaluating Scheduling and Replica Optimisation Strategies in OptorSim, University of Glasgow, Glasgow, G12 8QQ, Scotland 2 ITC-irst, Via Sommarive 18, 38050 Povo (Trento), Italy 3 CERN, European Organization for Nuclear Research, 1211 Geneva, SwitzerlandM. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[2] Pramod K. Meher and Jagdish C. Patra" A New Approach to Secure Distributed Storage, Sharing and Dissemination of Digital Image" School of Computer Engineering Nanyang Technological University, Singapore 639798.

[3] Mansaf Alam and Kashish Ara Shakil, Big Data Analytics in Cloud environment using Hadoop,Department of Computer Science, Jamia Millia Islamia, New Delhi.

[4] Zujie Ren; Jian Wan; Weisong Shi; Xianghua Xu; Min Zhou, "Workload Analysis, Implications, and Optimization on a

Production Hadoop Cluster: A Case Study on Taobao," in Services Computing, IEEE Transactions on , vol.7, no.2, pp.307-321, 2014.

[5] "LsPS: A Job Size-Based Scheduler for Efficient Task Assignments in Hadoop," in Cloud Computing, IEEE Transactions on , vol.3, no.4, pp.411-424, Oct.-Dec. 1 2015, Yi Yao; J.Tai; Bo Sheng; Ningfang Mi.

[6] G.R. Goodson, J.J. Wylie, G.R. Ganger, and M.K. Reiter, "Efficient Byzantine-Tolerant Erasure-Coded Storage," Proc. Int'l Conf. Dependable Systems and Networks, pp. 135-144, July 2004.

[7] Gupta, Rajeev, Himanshu Gupta, and Mukesh Mohania. "Cloud computing and big data analytics: what is new from databases perspective?." Big Data Analytics. Springer Berlin Heidelberg, 2012. 42-61.

[8] L. X. Bin Fan, Wittawat Tantisiriroj and G. Gibson, "Disk reduce: Replication as a prelude to erasure coding in data-intensive scalable computing," Carnegie Mellon Univsersity, Parallel Data Laboratory, Tech. Rep. Technical Report CMU-PDL-11-112, 2011.

[9] P.Crosby. EDGSim. http://www.hep.ucl.ac.uk/ pac/EDGSim/.

[10] A. Dimakis, V. Prabhakaran, and K. Ramchandran, "Decentralized erasure codes for distributed networked storage," IEEE/ACM Transactions on Networking, vol. 14, 2006.

[11] H. Lamehamedi, Z. Shentu, B. Szymanski, and E. Deelman. Simulation of Dynamic Data Replication Strategies in Data Grids. In Proc. 12th Heterogeneous Computing Workshop (HCW2003), Nice, France, April 2003. IEEE-CS Press.

[12] J. Wang, P. Shang, and J. Yin, "Draw: A new data-grouping-aware data placement scheme for data intensive applications with interest locality," in Cloud Computing for Data-Intensive Applications. Berlin, Germany: Springer, 2014, pp. 149–174.

[13] S.Ghemawat, H.Gobioff and S.T.Leung,"The Google file system"ACM SIGOPS Operating Syst. Rev., vol.37, no.5, pp.29–43, 2003.

[14] R. Rodrigues and B. Liskov, "High Availability in DHTs: Erasure Coding vs. Replication", in Proc of IPTPS, 2005.

[15] R. Bhagwan, S. Savage, and G. Voelker, "Replication Strategies for Highly Available Peer-to-Peer Storage Systems", in Proc. of MMCN, 2002.

[16] J. Plank, "Erasure codes for storage applications", in Tutorial, FAST2005: 4th USENIX Conference on File and Storage Technologies, 2005.

[17] H. Weatherspoon and J. Kubiatowicz, "Erasure codes vs. Replication: A quantitative comparison", in Proc. IPTPS, 2002.

[18] R. Murty, G. Mainland, I. Rose, A. Chowdhury, A. Gosain, J. Bers, and M. Welsh, "CitySense: An Urban-Scale Wireless Sensor Network and Testbed", in IEEE International Conference on Technologies for Homeland Security, 2008.

[19] S. B. Wicker and V. K. Bhargava, ed., "Reed-Solomon Codes and Their Applications", in Piscataway, NJ: IEEE Press, 1983.

[20] B. Gaidioz, B. Koblitz, and N. Santos, "Exploring High Performance Distributed File Storage Using LDPC Codes".

[21] Ashish Khisti, "Tornado Codes and Luby Transform Codes", 2003.

[22] M. Luby, "LT Codes", in Proc. IEEE Foundations of Computer Science (FOCS), 2002. [23] A. Shokrollahi, "Raptor Codes", in IEEE Trans. on Information Theory, vol. 52, pp. 2551-2567, 2006.

[23] B. Chun, F. Dabek, A. Haeberlen, E. Sit, H. Weatherspoon, M. F. Kaashoek, J. Kubiatowicz, R. Morris, "Efficient replica maintenance for distributed storage systems ", in Proc. of NSDI, 2006, pp. 45–58.

[24] F. Dabek, M. F. Kaashoek, D. Karger , R. Morris , I. Stoica, "Wide-area cooperative storage with CFS", in ACM SIGOPS, 2001, pp 202–215.

[25] Q. Xin, E. L. Miller, T. J. E. Schwarz, "Evaluation of distributed recovery in large-scale storage systems", in Proc. of High performance Distributed Computing, 2004, pp 172–181.

[26] I. Stoica, R. Morris, D. Karger, M. Kaashock, and H. Balakrishman, "Chord: A scalable peer-to-peer lookup protocol for internet applications," in ACM SIGCOMM, San Diego, CA, August 2001, pp. 149–160.

[27] L.Pamies-Juarez, Anwitaman Datta and Frederique Oggier, Rapid RAID: Pipelined Erasure Codes for Fast Data Archival in Distributed Storage Systems, Nanyang Technological University (Singapore).

[28] Storage-Tag-Aware Scheduler for Hadoop Cluster Nawab Muhammad Faseeh Qureshi, Dong Ryeol Shin Member, IEEE, Isma Farah Siddiqui and Bhawani Shankar Chowdhry Senior Member, IEEE.

[29] A Study of Suitable Fault Tolerance Frameworks for an Energy-efficient Storage System, Shou-Han Chen, Tseng-Yi Chen, Hsin-Wen Wei, Nai-Yuan Jhang, Tsan-sheng Hsu, Wei-Kuan Shih, Department of Computer Science, National Tsing Hua University, Hsinchu, Taiwan ;Department of Electrical Engineering, Tamkang University, New Taipei City, Taiwan; Institute of Information Science, Academia Sinica, Taipei, Taiwan.

[30] C. Huang, H. Simitci, Y. Xu, A. Ogus, B. Calder, P. Gopalan, J. Li, and S. Yekhanin, "Erasure coding in windows azure storage," in Proceedings of the USENIX Annual Technical Conference (ATC), 2012.

[31] Khan, M.; Yong Jin; Maozhen Li; Yang Xiang; Changjun Jiang, "Hadoop Performance Modeling for Job Estimation and Resource Provisioning," in Parallel and Distributed Systems, IEEE Transactions on , vol.27, no.2, pp.441-454, Feb. 1 2016.

[32] "Assisting developers of big data analytics applications when deploying on hadoop clouds.", Shang, Weiyi, et al. Proceedings of the 2013 International Conference on Software Engineering. IEEE Press, 2013.

[33] Kambatla, Karthik, Abhinav Pathak, and Himabindu Pucha. "Towards Optimizing Hadoop Provisioning in the Cloud." HotCloud 9 (2009): 12.

[34] Cuong Pham, Feng Zhang, Duc A. Tran" Maintenance-Efficient Erasure Coding for Distributed Archival Storage" Department of Computer Science, University of Massachusetts, Boston ‡ Distributed System Infrastructure Group, EMC Corporation, USA

[35] K.Ranganathan and I. Foster. Identifying Dynamic Replication Strategies for a High Performance Data Grid. In Proc. of the Int. Grid Computing Workshop, Denver, CO, USA, November 2001.

[36] Decoupling Computation and Data Scheduling in Distributed Data-Intensive Applications, K.Ranganathan and I. Foster. In Int. Symposium of High Performance Distributed Computing, Edinburgh, Scotland, July 2002.

[37] T. White, Hadoop-The Definitive Guide. Sebastopol, CA, USA: O'Reilly, 2009.

[38] D. Borthakur, "The hadoop distributed file system: Architecture and design, 2007," Apache Software Foundation, Forest Hill, MD, USA, 2012.

ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 5, Special Issue 2, January 2018

[39] B. Calder, et al., "Microsoft azure storage: A highly available cloud storage service with strong consistency," in Proc. 23rd ACM Symposium. Operating Syst. Principles, 2011, pp. 143–157.

[40] S. S. Miller, M. S. Shaalan, and L. E. Ross, "Correspondent-centric management email system uses message-correspondent relationship data table for automatically linking a single stored message with its correspondents," U.S. Patent 6 615 241, Sep. 2, 2003.

[41] RapidRAID: Pipelined Erasure Codes for Fast Data Archival in Distributed Storage Systems, Lluis Pamies-Juarez, Anwitaman Datta and Frederique Oggier, Nanyang Technological University (Singapore), Email: {lpjuarez, anwitaman, frederique}@ntu.edu.sg.

[42] Optimizing Remote file Access for Parallel and Distributed Network Applications, Jon B. Weissman, Mahesh Marina, Department of Computer Science, University of Minnesota, Twin Cities, jon@cs.umn.edu.

[43] T∗: A Data-Centric Cooling Energy Costs Reduction Approach for Big Data Analytics Cloud, Rini T. Kaushik, Klara Nahrstedt, University of Illinois, Urbana-Champaign, {kaushik1}@illinois.edu .

[44] Highly Reliable Two-Dimensional RAID Arrays for Archival Storage; Jehan-François Pâris, Thomas Schwarz, S. J., Ahmed Amer and Darrell D. E. Long. Computer Science Dept. University of Houston, Houston.

[45] L. L. You, K. T. Pollack, and D. D. Long, "Deep store: An archival storage system architecture," in Proc. 21st Int. Conf. Data Eng., 2005, pp. 804–815.

[46] S. Quinlan and S. Dorward, "Venti: A new approach to archival storage," in Proc. Conf. File Storage Technol., 2002, pp. 89–101.

[47] J. Huang, X. Liang, X. Qin, P. Xie, and C. Xie, "Scale-RS: An efficient scaling scheme for RS-coded storage clusters," IEEE Trans. Parallel Distrib. Syst., vol. 26, no. 6, pp. 1704–1717, Jun. 2015.

[48] J. Huang, Y. Wang, X. Qin, X. Liang, S. Yin, and C. Xie, "Exploiting pipelined encoding process to boost erasure-coded data archival," IEEE Trans. Parallel Distrib. Syst., vol. 26, no. 11, pp. 2984–2996, Nov. 2015.

[49] M. Sathiamoorthy, et al., "XORing elephants: Novel erasure codes for big data," Proc. VLDB Endowment, vol.6, no.5, pp 325–336, 2013.

[50] Xia, M. Saxena, M. Blaum, and D. A. Pease, "A tale of two erasure codes in HDFS," in Proc. 13th USENIX Conf. File Storage Technol., 2015, pp. 213–226.