JCSE International Journal of Computer Sciences and Engineering Open Access
Research Paper Vol.-6, Issue-10, Oct. 2018 E-ISSN: 2347-2693

A Novel Method to Improve Data Deduplication System

K. H. Walse

Dept. of Computer Science and Engineering, Anuradha Engineering College, S.G.B. Amravati University, Chikhli, India

[*]*Corresponding Author: kwalse1234@gmail.com, Tel.: +91-9689271947*

Available online at: www.ijcseonline.org

Accepted: 8/Oct/2018, Published: 31/Oct/2018

***Abstract*—**In large organizations same data is stored on the different places by different users. This will occupy the storage space. In the duplicate removal process one can remove the file duplicate with the original file and make space empty for the further storage. It works by eliminating redundant data and ensuring that only one unique instance of the data is actually retained on storage. The data deduplication technique works by tracking each data file and eliminate each file that it found more than one copy of it in the storage. There are many techniques for deduplication. Our proposed algorithm depends on reducing the data before it's stored in the storage or backup. Basically the procedure is the system analyses the data before storing it by one of mechanism for checking like hash value. If the system found the same data is stored already, ignore the data or document else store the data and save its analysis for future processing. There are many advantages by using this technique. No need for extra storage space.

***Keywords*—** data deuplication, classification, storage, hashing

I. INTRODUCTION

Nowadays the population growth has affected the world in an undesirable way. Increasing population has lead to the problem of accommodation, pollution, etc. Comparing these with our computer world, increasing amount of data has led to problem of inefficient utilization of storage space, performance, cost, etc. In computer world data comes from various sources and in various forms. Data source may be an individual computer, mobiles, tablets or even servers. Also data occurred in the form of structured/unstructured files, compressed files, images, sounds, videos and many more. Because data is growing at faster speed and it is copying in a very short period of time, managing it is a great and intelligent task. Considering these issues experts are looking for different techniques to manage such data. There are many techniques used for eliminating the redundant data in the storage. One of the best techniques is the deduplication data. Deduplication is an intelligent compression technique which reduces duplicate data and saves a huge amount of space [1].
Deduplication technology is normally used to reduce the space and bandwidth requirements of services by eliminating unnecessary data and storing only a single copy of them.
This technology gives benefits by many features like data backup and archival, no need of maintaining hardware resources, greater data accessibility, etc. Deduplication is most effective when multiple users outsource the same data to the cloud storage. Also Deduplication technique takes advantage of data similarity to identify the same data and reduces the storage space [2]. This technique is categorized in to two different approaches: deduplication over encrypted data and deduplication over unencrypted data [2]. It contain various deduplication strategies as client Side or server side deduplication, also file level or block level deduplication. In term of size there are two types of deduplication:

File Level Deduplication: File level deduplication in which it discovers repetition between different files and eliminate these repetition to reduce capacity demand
Block Level Deduplication: Block level deduplication in which it discovers and eliminate repetition between data block. Also file is divided into smaller fixed size or variable size block. Fixed size blocks reduce the calculation of block boundaries, while using variable size block it provide better deduplication capability [3].

Deduplication strategies as client side and server side deduplication are categorized according to the host where deduplication happens. In server side deduplication, firstly all files are uploaded to the storage server which then deletes the duplicates. Clients are unaware of deduplication because duplicates are deleted after uploading the file. This strategy saves storage but not bandwidth. In client side deduplication, a client uploading a file and that file is first check by server to see the existence of this file on server database. Repetitions are not uploaded. This strategy saves storage and bandwidth [4]. Deduplication ratio is used to measure the effectiveness of deduplication. Deduplication ratio is defined as “the number of bytes input to a data deduplication process divided by the number of bytes output” [4].

Primarily deduplication technique in which server stores only single copy of every file, regardless of how many client asked to store that file. File stored by all client simply use links of file or single copy of file which is stored at server side. Furthermore, if copy of file is already stored at server then client do not need to upload the same file again to server, thus saving bandwidth as well as storage. In deduplication system firstly client sends server only hash value of the file then server check the hash value to see that hash value already exists in its database or not. If that hash value is not in database then server asked for entire file to client and if hash value already exists in database then it tells client that there is no need to send file. Then server marks the client as owner of that file, form that point there is no difference between the client and the original party who has upload the file [5]. Deduplication is the method to avoid having to store the identical data many times. Deduplication influences the fact that large data sets often show high repetition. Example of deduplication includes common email attachments, financial records, with common headers and semi-identical fields, and popular media content like music, video [6].

II. DATA DEDUPLICATION

Data de-duplication has many forms. Different organizations may use multiple de-duplication strategies. It is very essential to understand the backup and backup challenges, when selecting de-duplication as a solution. Data de-duplication has three types. Compression, single-instance storage, sub-file de-duplication.

A. Compression

The data compression mechanism work by reducing the size of the file to save the storage. The word reducing means removing some binary digits from the file. The compression technique compresses all files even if it is duplicated. Because of the data size are reduced so the processing speed decrease, that means the overall speed will increase and the time to load or store data are decreasing. Data compression are limited though it has been available for many years, it becomes isolated to each particular file. For example, data compression cannot identify and remove duplicate files, but will independently compress each of the files [7].

B. Single-Instance Storage

Single-Instance Storage removes multiple copies of any file. Single-instance storage (SIS) environments can detect and eliminate redundant copies of identical files. As name suggests it keeps only single Instance or copy of data and pointers are created for all other users who own the same file. In Single-instance storage systems, content of files are checked to determine if the file to be uploaded is identical to an existing file or not. The number of files that are stored as unique at cloud, on the basis of file content in Single Instance system, there may be large amount of redundancy in that file or files. For example, a new date inserted into the title slide of a presentation of some files, this is very small amount of change in huge files but considered as different files to be stored without further de-duplication.

C. Sub-file De-Duplication

If redundant data exists in separate files not needed to be identical files, that redundancy can be avoided with the help of Sub-file. Sub-file de-duplication implementation has two types. Fixed-length sub-file de-duplication uses fixed length of data to search for the duplicate data within the files. Fixed-length segments are simple in design, but miss many opportunities to discover redundant sub-file data. For example, when name of person is added to certain file’s tile page—the whole content of the file will shift, resulting the failure of the de-duplication to detect equivalencies. Means, small change or addition in file may cause non equivalencies. Variable-length implementations are usually not corresponding to segment length. Variable-length implementations match data segment sizes to the naturally occurring duplication within files, vastly increasing the overall de-duplication ratio [7].

III. RELATED WORK

Nagapramod Mandagere, Pin Zhou [8], presented a system demystifying data deduplication which aims to provide a comprehensive taxonomy and experimental evaluation using real-world information. Demystifying Data Deduplication gives the basic idea of Deduplication and where it can be applied and basic algorithms used. Problem of this method is same technique been applied over all types of data which increases backup window time. In this results show that between different deduplication techniques the space savings varies by about 30%, the CPU usage differs by almost 6 times and the time to reconstruct a deduplicated file can vary by more than 15 times [8].

Earlier study done by Jiang and Fenge [9] provide a deduplication system which produced increased latency and reduces throughput, high data transmission costs which result in a large backup window. Researchers developed a novel technique called SAM, A semantic aware multi-tiered source deduplication framework that first combines the global file-level deduplication and local chunk level deduplication. They also considered file level semantic attributes like file locality, file time stamps, file size, file type which are used to find redundant data. The performance evaluation proved that they achieved high ratio of deduplication efficiency, reduced throughput shortens the backup time by an average of 38.7% during backup operation [9][10].

Earlier study done by D. Harnik, B. Pinkas, and A. Shulman-Peleg discusses the attacks that develop client-side deduplication, by allowing an attacker to gain access to random size files of other users based on a very small hash signatures of these files. To overcome such attacks, it introduce the notion of proofs-of-ownership (PoWs), which lets a client efficiently prove to a server that that the client holds a file, rather than just some short information about it [11].

Earlier study by Jiang Yinjin et al. [12] proposed a application based source deduplication scheme Application Aware Deduplication or AA-deduplication technique reduces The data processing overhead by implementing an intelligent data chunking scheme and the adaptive used of hash function. In AA-deduplication architecture firstly files are filtered out by file size and chunks are created by using broken of backup data streams by an intelligent chunker using strategy called application aware chunking. Files with same type having data chunks are then deduplicated using application aware

Table 1: Existing Technique Summary

Sr. no	Paper title	Year	Author	Work description	Result
1	Demystifying Data Deduplication	2008	Nagapramod Mandagere, Pin Zhou, Mark A Smith, Sandeep Uttamchandani	Gives the basic idea of deduplication and where it can be applied and basic algorithms used.	Variable size hash consumes large number of CPU cycles. Increases window backup time.
2	SAM: A Semantic Aware Multi-Tiered Source De-duplication Framework for Cloud Backup	2010	Yujuan Tan, Hong Jiang, Dan Feng, Lei Tian, Zhichao Yan, Guohui Zhou	Focuses on reduction in backup window time and restore time.	Needs high restoration time. Reduces backup time by an average of nearly 38.7%.
3	Proofs of Ownership in Remote Storage Systems	2011	Shai Halevi, Danny Harnik, Benny Pinkas, Alexandra	Focuses on Rigorous security, Identify attacks, Bandwidth saving , Time saving	Performance measurements indicate that the scheme incurs only a small overhead as compared to naive client-side deduplication.
4	AA-Dedupe: An Application-Aware Source Deduplication Approach for Cloud Backup Services in the Personal Computing Environment	2011	Yinjin F, Hong Jian, Nong Xiao, Lei Tian, Fang Liu	Focuses on significantly reduce the computational overhead, increase the deduplication throughput and improve the data transfer efficiency	resulting in shortened backup window, increased power-efficiency and reduced cost for cloud backup services.
5	Data Deduplication Scheme for Cloud Storage	2012	Iuon-Chang Lin and Po-Ching Chien	A scheme that utilizes the capacity of cloud storage server and it	Resulting on less time on block verification and

				improved speed of data deduplication	computation time is also less.
6	Deduplication and Compression Techniques in Cloud Design	2012	Amrita Upadhyay, Pratibha R Balihalli, Shashibhushan Ivaturi and Shrisha Rao	goal is to reduce storage space and bandwidth usage during file transfer	minimizing bandwidth requirements

Deduplicator by considering their hash values that are stored in local disk. For file if match is found, file containing chunk having metadata is updated to point to existing chunk location. On other hand if there is no match, based on container management in the cloud the new chunk having metadata for associated file is updated to point to it and new entry is added to application aware index.

Earlier study done by Lin and Chien [13][14] developed a Signature based deduplication the recently used deduplication systems used in the cloud storage server spend too much time on examining duplicate blocks. In deduplication researchers proposed a scheme that utilizes the capacity of cloud storage server and also it improved speed of data deduplication. In signature based deduplication to insure the integrity of file a signature is used for each uploaded file. The scheme is derived from Zhang’s digital signature method with fault tolerance which is based on RSA cryptosystem

Amrita upadyay [15] presented deduplication and compression technique in cloud design its goal is to reduce storage space and bandwidth usage during file transfer. In this only copy of duplicate file is retained while other are eliminated. For deduplication the design is depend on multiple metadata structure. In deduplication and compression technique metadata determined the existence of duplicate files. Depending on file size it is clustered into bins. Then files are segmented, deduplicated, compressed and stored. Binning is the process by which it restricts number of segments and also its sizes so that it is optimum for each file size. In these deduplication process segments of file reduced to smaller chunks which are then easy to send over internet. Transfer unique compressed segments minimize bandwidth. This increase storage efficiency and also decrease in cost.

IV. PROPOSED FRAMEWORK

From previous studies of the way to optimizing the storage. We found that one of the best solutions is by deduplication or in another word eliminate the duplicated files, keep only unique data in storage or backup. There are many techniques for deduplication. Our proposed algorithm depends on reducing the data before it's stored in the storage or backup. Basically the procedure is the system analyzes the data before storing it by one of mechanism for checking like hash value. If the system found the same data is stored already, ignore the data or document. Else store the data and save its analysis for future processing. There are many advantages by using this technique. No need for extra storage space. The data domain is less bandwidth.

As seen in the flow diagram, the deduplication process begins with start. In starting it ask user to log in and for new user it necessary to register first. When users register successfully then next step is to enter mail id and password. Then system check mail id and password enter by user is valid or not. Now user is ready to start data deduplication process. Now if user with valid email id and password want to upload a file, to upload a file first is to select that particular file then with the help of hash value it check that file is available at server. If file is not available at server that means it is new file then file is uploaded to server successfully. And if user wants to upload a file is already available at server then it shows message that file is already available at server and you have access right, then there is no need to upload same file. Now user wants to send file or document to another user, the work is done by send document function. To send document first user required is the mail id of another user. Then user needs to select file to send and then select user which he/she wants to send document. With this process file is successfully send to the user.

The idea is using a hash algorithm attached to each data block. A SHA algorithm is used for this purpose. In this paper Message digest (MD5) is chosen. The reason of choosing MD5 because it 128bit length. Sha is more secure than MD5 because it is 160bit length, but the longer length means slower processing and more storage space is needed.

When file is processed by a hashing algorithm, a hash is created that represent the data. A hash is a bit string (128 bit for MD5) that represents the file processed. If you processed the same data through the hashing algorithm multiple times, the same hash value is created each time.MD5 hash functions to calculate the file's hash value and then pass the value to storage System. Compare the new hash value with the existing values. If it exists earlier in the system, system says, the content of the file is already available. No need to upload the file and it also check the number of links. If the hash value did not exist earlier, then it will ask the client to upload the file and update the logical path of that file.

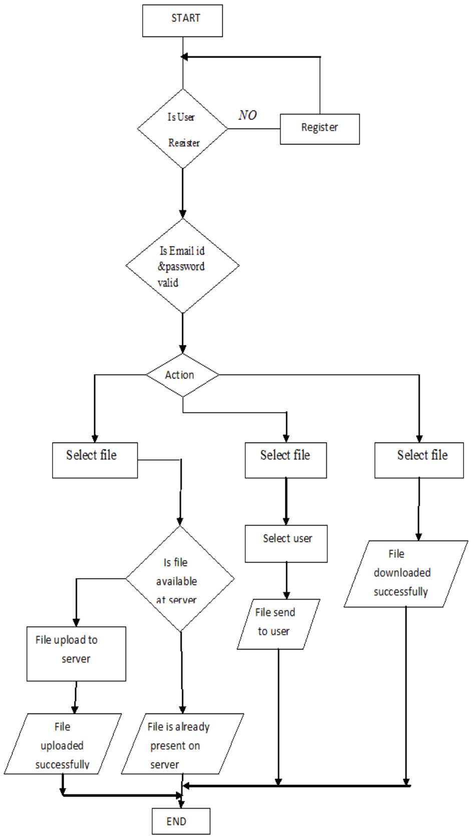
START
Is User Register
NO
Register
Is Email id &password valid
Action
Select file
Select file
Select file
Is file available at server
Select user
File downloaded successfully
File send to user
File upload to server
File uploaded successfully
File is already present on server
END

Fig 1: Flow chart of Data deduplication technique

V. RESULT

A set of sample files are taken for deduplication. If a new file comes, it is saved on storage. If a duplicated file comes, it is not saved, only index file is updated, the storage space is saved. Testing was done for different combinations of files and results were tabulated.

The graph fig. 3 shows time taken for MD5 algorithm i.e. time required calculating hash value of file. Here time required to calculate hash value of five files is considered and generated graph showing result for this. Here x axis represents file size and y axis represents time.

Table 2: Represents the time taken to upload document

File name	File size	File type	Time to upload document(ms)
1	1mb	.ppt	5
2	1mb	.pdf	6
3	1mb	.pdf	4
4	2mb	.mp4	13
5	2mb	.ppt	10
6	2mb	.pdf	11
7	4mb	.pdf	15
8	4mb	.pdf	17
9	4mb	.pdf	16
10	6mb	.pdf	22
11	6mb	.pdf	20
12	6mb	.pdf	24
13	8mb	.mp4	26
14	8mb	.mp4	24
15	8mb	.mp4	25

The below graph Fig 2 shows the file uploading time i.e. the time required to upload the file. Here time required to upload five files is considered and generated graph showing result for this. Here x axis represents file size and y axis represents file uploading time.

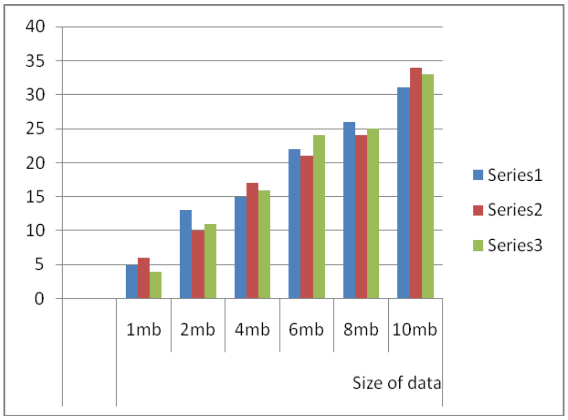

Fig 2: Computation time for upload

Table 3: represent time taken for MD5 algorithm

Sr. no	File type	File size	MD5time taken (ms)
1	.pdf	12kb	728
2	.pdf	18kb	755
3	.pdf	1mb	705
4	.mp4	4mb	925
5	.DOC	13mb	826

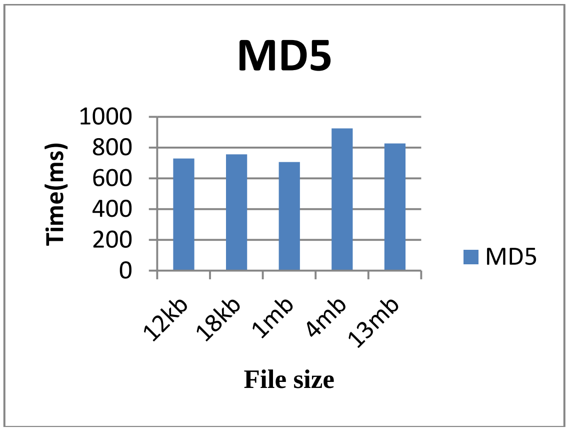

Fig 3: Computation time for MD5 algorithm

VI. Conclusion and Future Work

This paper presented broad research on data deduplication. Deduplication is storage optimization technique that avoids keeping duplicate copies of data. It focuses on the deduplication methods among other types of techniques. In this paper, a new method is proposed depending on the time of data arrival. The proposed system calculates hash values using hashing algorithms and stores them in efficient manner for better searching of index. The paper describe procedure for storing, accessing and deleting the files. In this work a method is implemented for removing the duplicate files using MD5 algorithm, ensuring the reduced time to deduplicate files being uploaded by the clients. This technique improve storage capacity and improve the performance by comparing the data before storing it using MD5 hash algorithm and store only the unique data file. In future, researcher plan to develop a framework which can optimize the performance measure like network bandwidth, high throughput, computational overhead, deduplication efficiency, backup window size. Along with this various other encryption techniques with different block sizes can be combined to obtain more efficient results.

References

[1] Jyoti Malhotra , Jagdish Bakal "A Survey and Comparative Study of Data Deduplication Techniques" International Conference on Pervasive Computing (ICPC)

[2] Junbeom Hur, Dongyoung Koo, Youngjoo Shin,And Kyungtae Kang "Secure Data Deduplication With Dynamic Ownership Management in Cloud Storage" IEEE Transactions on Knowledge and Data Engineering

[3] Jin Li, Xiaofeng Chen, Xinyi Huang, Shaohua Tang and Yang Xiang Senior Member, IEEE And Mohammad Mehedi Hassan Member,IEEE and Abdulhameed Alelaiwi Member,IEEE "Secure Distributed Deduplication Systems with Improved Reliability" IEEE Transactions on Computers

[4] Ms. Priyanka S. Savaji and Dr. K. H. Walse, "Survey on Data Deduplication system," In Proceedings International Conference-EECCMC 2018, pp. 1-7, 2018.

[5] Shai Halevi, Danny Harnik, Benny Pinkas, Alexandra Shulman-Peleg "Proofs of Ownership in Remote Storage Systems" October 17–21, 2011, Chicago, Illinois, USA2011ACM.

[6] Roberto Di Pietro , Alessandro Sorniotti "Proof of ownership for deduplication systems: A secure, scalable, And efficient solution" 2016 Elsevier

[7] Pawar P.R, Aarti Waghmare "Data Deduplication in Cloud Storage", National Conference on Advances in Computing 2015, pp 1-5, 2015.

[8] Nagapramod Mandagere, Pin Zhou, Mark A Smith, Sandeep Uttamchandani, "Demystifying Data Deduplication", 2008 ACM, pp 12-17, 2008.

[9] Yujuan Tan, Hong Jiang, Dan Feng, Lei Tian, Zhichao Yan, Guohui Zhou "SAM: A Semantic-AwareMulti-Tiered Source De-duplication Framework for Cloud Backup" 2010 39th International Conference on Parallel Processing.

[10] K. H. Walse , Dr. R. V. Dharaskar, Manisha V. Kharat ," Survey on Soft Computing Approaches for Human Activity Recognition", International Journal of Science and Research (IJSR), Vol. 6, 2017, pp 1328-1334, 2017.

[11] Shai Halevi, Danny Harnik, Benny Pinkas, Alexandra Shulman-Peleg," Proofs of Ownership in Remote Storage Systems", 2011 ACM, pp 491-500, 2011.

[12] Yinjin F, Hong Jian, Nong Xiao, Lei Tian, Fang Liu "AA-Dedupe: An Application-Aware Source Deduplication Approach for Cloud Backup Services in the Personal Computing Environment" In IEEE International Conference on Cluster Computing 2011.pp 112-120, 2011.

[13] Iuon-Chang Lin and Po-Ching Chien," Data Deduplication Scheme for Cloud Storage", International Journal of Computer, Consumer and Control (IJ3C), Vol. 1, 2012, pp 26-31, 2012.

[14] R. V. Dharaskar, Ph.D, Pravin R Futane, V. M. Thakare, Ph.D," Summarization of Own Contributory Efforts in the Field of Indian Sign Language Recognition System", Proceedings published by International Journal of Computer Applications NCIPET-2013, pp 26-30, 2013.

[15] Amrita Upadhyay, Pratibha R Balihalli, Shashibhushan Ivaturi and Shrisha Rao "Deduplication and Compression Techniques in Cloud Design" 2012 IEEE,pp 1-6, 2012.

Authors Profile

K. H. Walse is working as Professor at Anuradha Engineering College, Chikhli Distt. Buldana. Prior to this He was Principal at Shreeyash Polytechnic, Aurangabad during 2010-12. He was with Anuradha Engineering College as an associate Professor of Computer Science and Engineering where he leads the HCI Research Group(1996-2010) Prior to joining Anuradha Engineering College in 1996, he was an Lecturer of Computer Science and Engineering at SSGMCE, Shegaon (1994-1996).

He received his Ph.D. and Masters in Computer Science and Engineering from S.G.B. Amravati University, India. Prior to joining academia he spent 03 years working in industry both in India in development of computer networks. He had been a technical program Convener for ACM ICAC-2008.