ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 7, July 2014**

DOI: 10.15662/ijareeie.2014.0307040

# A Novel Way of Deduplication Approach for Cloud Backup Services Using Block Index Caching Technique

Jyoti Malhotra[1] ,Priya Ghyare[2]

Associate Professor, Dept. of Information Technology, MIT College of Engineering, Pune, India[1]

PG Student [IT], Dept. of Information Technology, MIT College of Engineering, Pune , India[2]

**ABSTRACT:** Data Deduplication describes approach that reduces the storage capacity needed to store data or the data has to be transfer on the network. Cloud storage has received increasing attention from industry as it offers infinite storage resources that are available on demand. Source Deduplication is useful in cloud backup that saves network bandwidth and reduces network space Deduplication is the process by breaking up an incoming stream into relatively large segments and deduplicating each segment against only a few of the most similar previous segments. To identify similar segments use block index technique The problem is that these schemes traditionally require a full chunk index, which indexes every chunk, in order to determine which chunks have already been stored unfortunately, it is impractical to keep such an index in RAM and a disk based index with one seek per incoming chunk is far too slow. In this paper we describes application based deduplication approach and indexing scheme contains block that preserved caching which maintains the locality of the fingerprint of duplicate content to achieve high hit ratio and to overcome the lookup performance and reduced cost for cloud backup services and increase dedulpication efficiency.

**KEYWORDS:** Backup services, Caching, Data deduplication.

## I.INTRODUCTION

The explosive growth of the digital data, data deduplication has gained increasing attention for its storage efficiency in backup storage systems. Today, in the context of user data sharing platforms the challenges for large scale, highly redundant internet data storage is high. Due to this redundancy storage cost is reduces. Storage for this increasingly centralized Web data can be getting by its de duplication. Data deduplication describes a class of approaches that reduce the storage capacity needed to store data or the amount of data that has to be transferred over a network. These approaches detect coarse-grained redundancies within a data set, e.g. a file system; Data deduplication not only reduces the storage space requirements by eliminating redundant data but also minimizes the network transmission of duplicate data in the network storage systems. It splits files into multiple chunks that are each uniquely identified by a hash signature called a fingerprint. It removes duplicate chunks by checking their fingerprints, which avoids byte by byte comparisons. Mainly data deduplication focused on different terms like throughput, advance chunking schemes, other type of storage capacity and clustering method and system workload.

As data passes through a cache on its way to or from the storage/processing/networking device, some of the data is selectively stored in the cache. When an application or process later accesses data stored in the cache that request can be served faster from the cache than from the slower device. The more requests that can be served from cache, the faster is the overall system performance. There is a trade-off in cache cost and performance. Larger caches yield a higher cache hit rate and therefore better performance. Unfortunately, the hardware used for cache is generally more expensive than the hardware used for the storage, processing, or networking device. As a result, cache design is a trade-off between size and performance. The best caching algorithms yield a higher hit rate for a given cache size. The three most common types of caches are: CPU cache, used to speed up the processor; storage cache, intended to accelerate

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 7, July 2014**

**DOI: 10.15662/ijareeie.2014.0307040**

storage I/O; and web/ network cache, designed to improve responsiveness of web applications In this paper we introduces A framework for Application based deduplication scheme based on caching method gives the better lookup performance for index structure based on application system on cloud network so that it gives the various chunking methods and improve data transfer efficiency and save cost on cloud services. Mainly detection is focus on locality based approach that gives various block stored on cache memory containing fingerprint with chunk id and block number so that index lookup make easily and storage capacity increases. This locality information contains information regarding backup files of current data that used during compare backup data of one or more backup files. Update database if new data is available and store in container of database.

## II. LITEARTURE SURVEY

The increasing popularity of the cloud backup services has a great attention to the industry. cloud backup services has become a cost effective choice for data security of personal cloud environment [8] and also for improving deduplication efficiency, Yinjin Fu, et.al [4] In this paper, introduce ALG dedupe system used for to combine local and global deduplication for maintain effectiveness. Proposed system gives the optimize performance of lookup performance and used for personal cloud environment and reduce system overload. Existing method that are introduces for deduplication technology for backup service only focus on removing redundant data from transmission during backup operation to reduce backup time and there is no attention in restore time. Yujuan Tan, et.al [5] this paper introduces CAB Architecture that captures the casual relationship among dataset used in backup and restore operation. It is integrated into existing backup system. This Architecture remove the redundant data from transmission not only backup operation but also restore operation and improve the backup and restore performance and also reduce both the reduction ratio. Dongfang Zhao, et.al[1] This paper presents a distributed storage of middleware, Called as HyCache+, used compute nodes, which allows I/O to the high bi section bandwidth of the high speed interconnect to the parallel computing systems. HyCache+ gives the POSIX interface to end users with the memory class I/O throughput and latency, and transparently exchange the cached data with the existing slow speed but high capacity networked attached storage. This caching approach shows 29X speedup over the traditional LRU algorithm. Deduplication on primary storage system is rarely used because of the disk bottleneck problem [9].There has been many different ways to solve the index lookup problem these effort have typically been limited to backup systems. Dirk Meister, et.al [2] this method is try to capture the locality information of a backup run and use this in the next backup run to predict future chunk requests. Using this method less I/O operation is needed and gives the better performance of lookup problem than Zhu performances that overcome in BLC approach. Wildani, et.al [3] the hands technique that used in this paper reduces the amount of memory index storage and making primary deduplication cost effective. This technique use the fingerprint cache and LRU caching algorithm and working set is calculated for fingerprint making group of Fingerprint so that single entry of fingerprint is accesses into memory index cache due that number of cache hit ratio increases.

## III. PROPOSED SYSTEM MODEL

Data Deduplication has emerged as an attractive lossless compression technology that has been employed in various network efficient and storage optimization systems so that we proposed A new approach for Application based Deduplication for cloud backup services using Block Locality Caching contain backup data as shown in figure 1.From backup files as input files having redundant or copied data files that want to deduplicate and for improve storage efficiency this system uses different chunking method base on file type. Files are filtered because of containing tiny files having less than 10 KB Size. So that after making group of files in MB Get filter and then different chunking strategy is used in this system. Chunk with file type are then deduplicate by calculating hash value name as fingerprint using different hash algorithm this fingerprint is then stored in container of cloud having new entries. Fingerprint which we stored for finding duplicate copies are get indexing by using block locality method index entries are name by their block number and chunk id. All this information is stored in block and blocks are stores in cache. If we search fingerprint in block and a match is found, the block for the file containing that chunk of fingerprint is updated and point to the location of the existing chunk of fingerprint. If there is no match, then new fingerprint is stored based on the container management in the cloud, the metadata for the associated file is updated to point to it and a new entry is

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 7, July 2014**

**DOI: 10.15662/ijareeie.2014.0307040**

added into the application aware index to index the new chunk of fingerprint. Due to this performance of system increases and system overload is reduced.

1. Different chunking methods
   Depending on file type whether the file is compressed file or uncompressed files. Chunking can be done. Mainly chunking is the process where files are break into chunk with same size or variable size. For this system use different chunking methods for whole file chunking as file contain compressed file and static and dynamic chunking on uncompressed files. And for identify chunk boundaries use Rabin hash fingerprint. So we use intelligent Chunker for chunking method.

2. Application based deduplicator for hashing techniques
   By generating fingerprint containing hash value and finds duplicate chunks on cloud. For the uncompressed file we use MD5 for dynamic chunking and for static SHA are used.

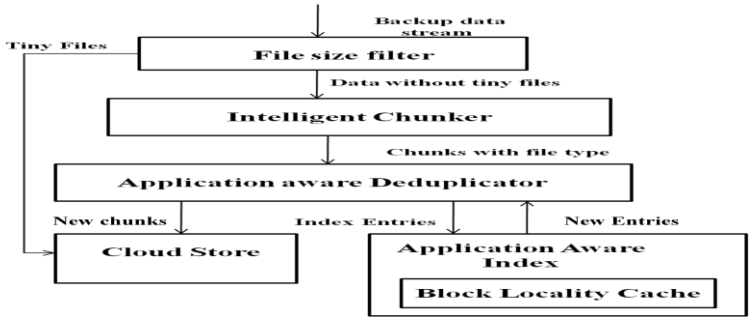


Figure.1 Architecture of application based Deduplication using block caching

## IV. RESULT AND DISCUSSION

1. Hash generation performance

   In the fig 2 shows that comparison of hashing algorithm for generating hash value of fingerprint contain file in MB size and time shows the comparison of hashing algorithm.The hash fingerprint method can reduce system overload on cloud system. We observed the performance of Rabin hash MD5 and SHA hashing algorithm from whole file chunking and static chunking i.e. SC and content defined chunking i.e. CDC with 4 kb chunk size. Rabin hash gives low computational overhead than MD5 and SHA and running time of Rabin hash and MD5 is less than of SHA. Due to less collision of files, SHA performance is better than MD5 and Rabin hash. So that purpose we use SHA for pdf files and MD5 for text files and Rabin hash for whole file.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 7, July 2014**

DOI: 10.15662/ijareeie.2014.0307040

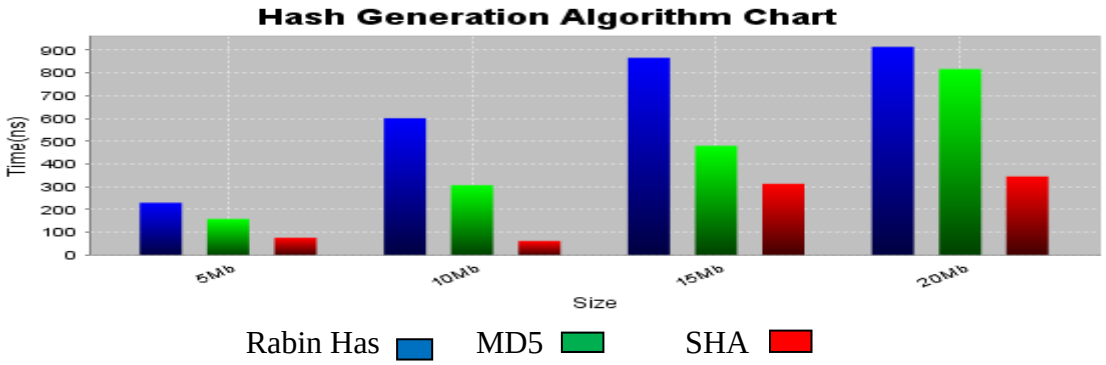


Figure.2 Comparison of hash generation

2. Deduplication ratio

In the fig 3 shows that Deduplication comparison of different chunk size from file contains 20 MB for text file and PDF file. As chunk size increases it affect Deduplication ratio in terms of time factor .For improving backup performance and Reduce the system overhead, improve the data transfer. Efficiency on cloud is essential. So that we use various chunking strategy such as CDC and static Chunking for improving running performance of the system and increase Deduplication ratio. Variation in chunk size affects Deduplication efficiency. Maintaining threshold size of chunk we get better Deduplication ratio pdf as well as text files.

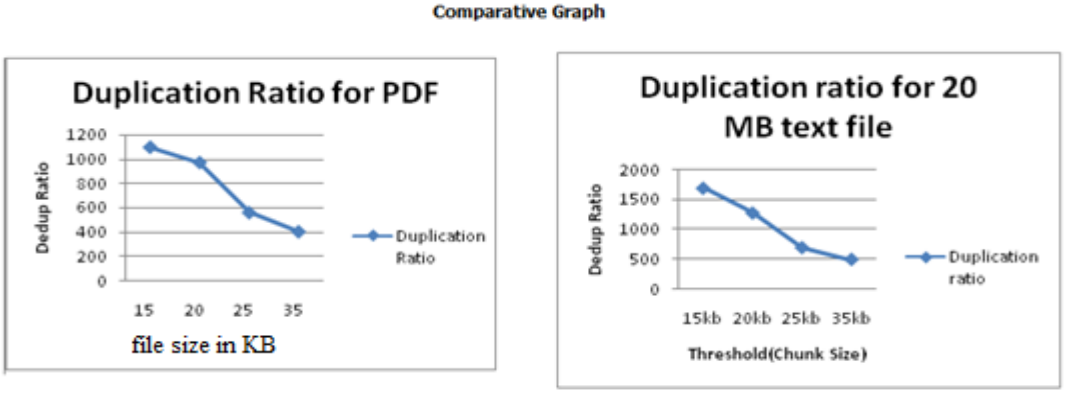


Figure.3 Comparison of Deduplication ratios

3. Cache performance

In the fig 4 shows the searching comparison of block cache and single cache with resp to file size and time in mili second. a new restore time caching and prefetching technique that exploits the perfect knowledge of future chunk accesses available when restoring a backup to reduce the amount of RAM required for a given level of caching at restore time. Because of modern disks relatively poor random I/O performance compared to sequential I/O, chunk fragmentation affect restore performance. Using block cache lookup performance of the system increase than single cache and system required less I/O operation so that searching time of the file required less time as compare to traditional system.

ISSN (Print) : 2320 – 3765
ISSN (Online): 2278 – 8875

International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering

*(An ISO 3297: 2007 Certified Organization)*

**Vol. 3, Issue 7, July 2014**

**DOI: 10.15662/ijareeie.2014.0307040**

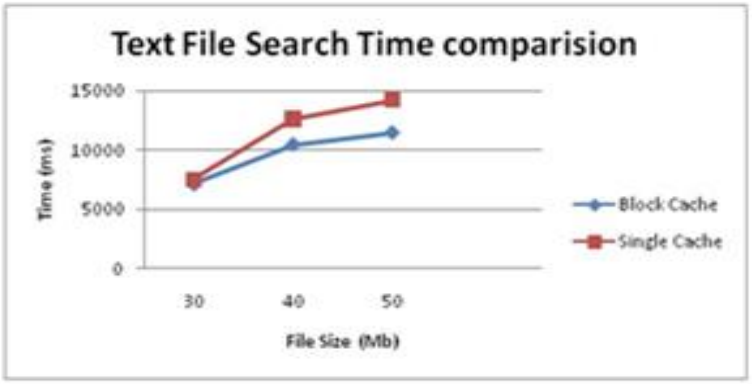


Figure.4 Comparison of searching time

## V. CONCLUSION AND FUTURE WORK

For cloud storage, using deduplication techniques and their performance and suggests a variation in the index of block level deduplication and improving backup performance and Reduce the system overhead, improve the data transfer efficiency on cloud is essential so that, We presented approach on application based deduplication and indexing scheme that preserved caching which maintains the locality of the fingerprint of duplicate content to achieve high hit ratio with the help of the hashing algorithm and improve the cloud backup performance. This paper proposed a novel variation in the deduplication technique and showed that this achieves better performance. Currently, optimized cloud storage has been tested only for text files and pdf files .In future, it can be further extended to use files of other type i.e. video and audio files.

## REFERENCES

[1] Dongfang Zhao, Kan Qiao, Ioan Raic,y ,"HyCache+: Towards Scalable High-Performance Caching Middleware for Parallel File Systems", Office of Science of the U.S. Department of Energy under contract DE-AC02-06CH11357, 2014.
[2] Dirk Meister, Jürgen Kaiser," Block Locality Caching for Data Deduplication**".** In Proceedings of the 11th USENIX Conference on File and Storage Technologies(FAST). USENIX, February 2013
[3] A. Wildani, E. L. Miller, and O.Rodeh. HANDS: A heuristically arranged non-backup in-line deduplication system. Technical Report UCSC-SSRC-12-03, University of California, Santa Cruz, March 2012
[4] Yinjin Fu, Hong Jiang, Nong Xiao, Lei Tian, Fang Liu,'' Application-Aware local global Source Deduplication for Cloud Backup Service of personal storage " IEEE International Conference on Cluster Computinges in the Personal Computing Environment (2012)
[5] Y. Tan, H. Jiang, D. Feng, L. Tian, and Z. Yan. CABdedupe: A causality-based deduplication performance booster for cloud backup services. In Proceedings of the 2011 IEEE International Parallel & Distributed Processing Symposium (IPDPS), 2011.
[6] Zhe SUN, Jun SHEN. DeDu: Building a Deduplication Storage System over Cloud Computing. In Proceedings of the 15th International Conference on Computer Supported Cooperative Work in Design, 2011
[7] Yinjin Fu, Hong Jiang, Nong Xiao, Lei Tian, Fang Liu,''AA-Dedupe: An Application-Aware Source Deduplication Approach for Cloud Backup Service " IEEE International Conference on Cluster Computinges in the Personal Computing Environment (2011)
[8]D. Harnik, B. Pinkas, and A. Shulman-Peleg. Side Channels in Cloud Services: Deduplication in Cloud Storage. IEEE Security and Privacy, 8(6):40–47, 2010
[9]B. Zhu, K. Li, and H. Patterson. Avoiding the disk bottleneck in the data domain deduplication file system. In FAST, 2008
[10] N. Mandagere, P. Zhou, M. A. Smith, and S. Uttamchandani. Demystifying data deduplication. In Companion '08: Proceedings of the ACM/IFIP/USENIX Middleware '08 Conference Companion, pages 12–17, New York, NY, USA, 2008. ACM
[11] HSU, W. W., SMITH, A. J., AND YOUNG, H. C. The automatic improvement of locality in storage systems. *ACM Transactions on Computer Systems 23*, 4 (2005), 424–473
[12] PATTERSON, R. H., GIBSON, G. A., GINTING, E., STODOLSKY, D., AND ZELENKA, J. Informed prefetching and caching. In *Proceedings of the Fifteenth ACM Symposium on Operating Systems Principles* (Dec. 1995), ACM Press, pp. 79–95.