International Journal of Innovative Research and Advanced Studies (IJIRAS)
Volume 3 Issue 4, April 2016

ISSN: 2394-4404

Data Deduplication For Cloud Backup Services Of Personal Storage Using Alg-Dedupe

Kiran Gawali

Sagar Pawar

Goraksh Pacharne

Rajendrakumar Kate

Dept. of Computer Engg.,
Govt. College of Engineering and Research Avasari (Kd.),
Pune, India

Prof. Danny J. Pereira

Guide

***Abstract:** so, it is the needed to implement a good backup and recovery plan. But the redundant nature of the backup data makes the storage a concern; hence it is necessary to avoid the redundant data present in the backup. Data de-duplication is one such solution that discovers and removes the redundancies among the data blocks. In this paper, a deduplication scheme is proposed that improves efficiency of data by reducing duplicated data and for that it uses application awareness concept. The application proposed is motivated on personal storage devices.*

***Keywords:** Application awareness, Cloud backup service, Chunking schemes, Data redundancy, Data deduplication, Deduplication efficiency.*

IJIRAS

I. INTRODUCTION

According to the present scenario, the backup has become the most essential mechanism for any organization. Backing up files can protect against accidental loss of user data, database corruptions, hardware failures, and even natural disasters. However, the large amount of redundancies which is found in the backups makes the storage of the backups a concern, thus utilizing a large of disk space. Data de-duplication comes as a rescue for the problem of redundancies in the backup. It is a capacity optimization technology that is being used to dramatically improve the storage efficiency. Data de-duplication eliminates the redundant data and stores only unique copy of the data. Here instead of saving the duplicate copy of the data, data de-duplication helps in storing a pointer to the unique copy of the data, thus reducing the storage costs involved in the backups to a large extent. It need not be applied in only backups but also in primary storage, cloud storage or data in flight for replication, such as LAN and WAN transfers. It can help organizations to manage the data growth, increase efficiency of storage and backup, reduce overall cost of storage, reduce network bandwidth and reduce the operational costs and administrative costs. The five basic steps involved in all of the data deduplication systems are evaluating the data, identify redundancy, create or update reference information, store and/or transmit unique data once and read or reproduce the data.

A. LITERATURE 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 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 global side i.e. data get reduced by system when actual data is going to store on server side 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.

II. PROPOSED SYSTEM

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 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 osf 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 added into the application aware index to index the new chunk of fingerprint.

III. ARCHITECTURAL OVERVIEW

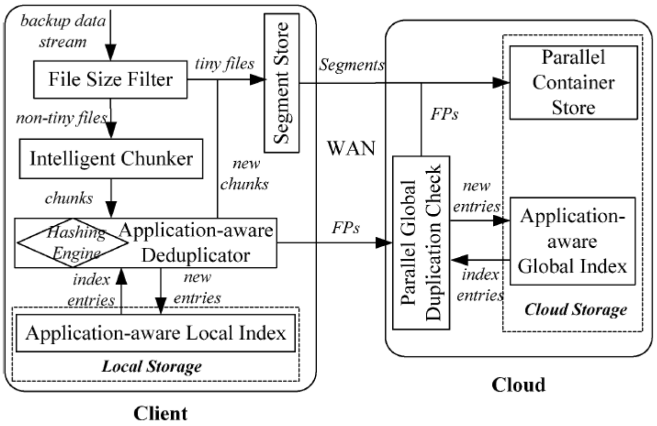

Figure 3.1: Architectural overview of the ALG-Dedupe design

A. FILE SIZE FILTER

Many files in the PC dataset are tiny(small) files. These files having size less than 10 KB, which consume less memory space with small percentage of the storage capacity. As shown in our statistical evidences in Section 2, about 60.3 percent of all files are tiny files, which uses only 1.7 percent of the total storage capacity of the dataset. To reduce the metadata overhead, ALG-Dedupe scheme filters these tiny files in the file size filter before the deduplication process actually starts, and these filtered tiny files grouped together into larger units of about 1 MB each in the segment store which alternatively increase the data transfer efficiency over WAN.

B. INTELLIGENT DATA CHUNKING

Data chunking schemes having different deduplication efficiency among different applications which differ greatly. Depending on the file type we divide files into three main categories: compressed files, static uncompressed files, and dynamic uncompressed files. The dynamic files are always editable means we can chang file frequently, while the static files are uneditable in common. To gain better deduplication efficiency between duplicate elimination ratio and deduplication overhead, we deduplicate compressed files with WFC, separate static uncompressed files into fix-sized chunks by SC with ideal chunk size, and break dynamic uncompressed files into variable-sized chunks with optimal average chunk size using CDC based on the Rabin fingerprinting to identify chunk boundries.

C. APPLICATION-AWARE INDEX STRUCTURE

An application-aware index structure for ALG-Dedupe is created. It has an in- RAM application index and small hash-table based on-disk indices divided by application type. According to the file type information, the incoming chunk is directed to the chunk index with the same file type. Each entry of the index stores a mapping from the fingerprint (fp) of a chunk or with its length (len) to its container ID (cid).As chunk locality exists in backup data streams , a small index cache is allocated in RAM to speedup index lookup by

IJIRAS

reducing disk I/O operations. The index cache is a key-value structure, and it is constructed by a doubly linked list indexed by a hash table. If the cache is full, fingerprints of those containers which are effective less in accelerating chunk fingerprint lookup are replaced. It make room for future prefetching and caching of index. ALG-Dedupe requires two application-aware chunk indices: a local index on the client side and a global index on the cloud side. Comparing with traditional deduplication mechanisms, ALG-Dedupe can achieve high deduplication throughput by looking up chunk fingerprints concurrently in small indices classified by applications rather than a single full, unclassified index for both local and global scenarios. Furthermore, a periodical data synchronization scheme is also proposed in ALG-Dedupe to backup the application-aware local index and file metadata in the cloud storage to protect the data integrity of the PC backup datasets.

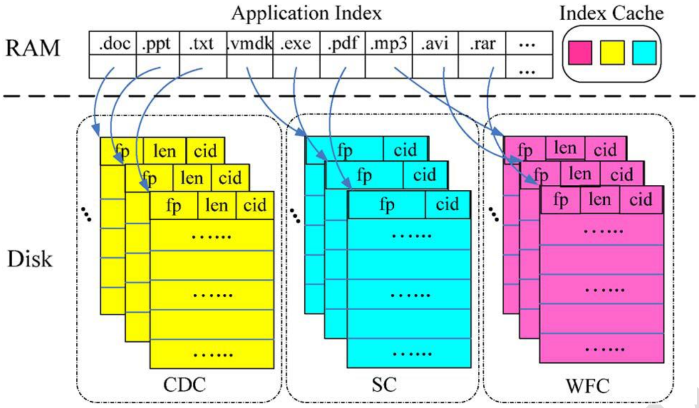

Figure 3.3: Application-aware index structure

D. APPLICATION-AWARE DEDUPLICATION

When file get divided into various chunks by intelligent chunker module, duplicated chunks will remove in application-aware deduplicator by creating chunk fingerprints in hash engine for detecting duplicate chunk in the local client as well as in server cloud. ALG-Dedupe strikes a good balance between alleviating computation overhead on the client side and avoiding hash collision to keep data integrity. At local level, extended 12-byte Rabin hash value as chunk fingerprint and MD5 value is used for global duplicate detection. A SHA-1 value of chunk is used as chunk fingerprint of SC in static uncompressed files and a MD5 value is used as chunk fingerprint of dynamic uncompressed files since chunk length is another dimension for duplicate detection in CDC-based deduplication in both cases local and global. To obtained high deduplication efficiency, the application-aware deduplicator first detects duplicate data in the application-aware local index respective to the local dataset with low deduplication latency in the PC client, and then checks local deduplicated data chunks with all data stored in the cloud by matching up fingerprints in the application-aware global index on the cloud side for high data reduction ratio. Only the unique data chunks after global duplicate detection are stored in the cloud storage with parallel container management.

IJIRAS

IV. CONCLUSION

ALG-Dedupe can accelerate backup efficiency and drive down IT costs. The implementation of Deduplication technique at client side, servers and also data centers will reduce the redundant data to much extent. Also the storage space will be utilized wisely by saving the only the unique data. Thus the availability of data and proper utilization of storage space can be managed with the ALG-Dedupe scheme. It combines local as well as global deduplication to equilibrium the effectiveness and latency of deduplication.

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 UCSCSSRC-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.