

to prevent sending similar files or files segments to backup servers. Deduplication ~~cause to decrease~~ results in reduced in bandwidth usage for both ~~customer~~ customers and service providers, and maintains storage area space for service providers. For the service provider, these abilities ~~cause in save of~~ save on power consumptions and hard disk accesses. [7]. Data type, retention period, and number of users can affect the Deduplication effectiveness. [7]. Deduplication ratio like including 1:3 results in a 66 percent saving, and 1:10 to 1:500 result results in more than 90 percent disk and bandwidth savings. more than 90 percent. [7], [8]

اثر بخشی یا کارائی؟  
Comment [Reviewer9]:

To find similarities, hashes of files must be calculated with algorithms with algorithms like MD5 or SHA-1. Similar files have similar hash amount values. These hashes are stored in hash tables for further using use in hash compares comparisons. is stored in hash table.

File level Deduplication is a technique to find similarities between files. In this technique, by using file hash and compare comparing file hash with hash of files in the hash table, if file has been stored before, only a link to the original file will be stored if the file in question has been stored previously.

Block level Deduplication is a technique that file is segmented segments a file to fixed or variable sized blocks, then hash of these blocks is compared with hashes in the block hash table. In this technique technique, either fixed size or variable size blocks can be used. [7], [9], [10], [11]

با رفرنس نویسی ژورنال چک کنید، شاید بتوانید همه را با کاما در یک کروش قرار دهید.  
Comment [Reviewer 10]:

Target-Target-based Deduplication is a technique that in which the server does performs all the process of Deduplication, and the client only sends files. Client files. Hence, the client does not know anything anythings about Deduplication and its processes. [7]. All work will be done carried out in on the server. [7]. Advantages-The advantages of this technique are decrease of reduction in storage area space, and overload caused by calculating hash. On the other hand hand, more network bandwidth will be used because all files are sent to the backup server, more network bandwidth will be used.

Source-Source-based Deduplication is a technique that hash of files or blocks is calculated before is sent by the client client either sends sends either hash to the backup server for comparison, or compares a hash with other hashes in the internal hash table. [7]. In the second situation, the clients has its own cache table.

In Global Deduplication, hash of file or block will be sent to the backup server for further comparison. On the other hand hand, in local Deduplication, all processes of hash calculation and comparison will be done in carried out on the client side.

## 2. Literature Review

To this day, many Deduplication models have been proposed Based based on different techniques, that described in section two, many Deduplication model had been proposed.

Aa-dedup [1] is an application-application-aware model that combines source-source-based and files level Deduplication. In this technique, file types and size is doing plays an important role in Deduplication. [1], [10], [11]. This This model significantly reduces the computational overhead, increases the Deduplication throughput, and improves the data transfer transfer efficiency. Four. It addresses four efficiency-observation observations about Deduplication in cloud backup service service is noted by this model. First-The first observation shows that "The majority of storage space is occupied by a small number of compressed files with low sub-file redundancy" [1]. Second-The second observation is about static-Static chunking-Chunking (SC) [12] and content-Content defined-Defined chunking-Chunking (CDC) [13]. By this observation observation, SC method can outperform CDC method in Deduplication effectiveness for static application data and virtual machine.Third machine applications. The third observation shows that "The eComputational overhead for Deduplication is dominated by data capacity" [1]; and finally, the fourth observation show that shows that "The amount of data shared among different types of applications is negligible". [1]

Comment [Reviewer11]: File-level?

Based on some observations and analysis-analyses, Aa-dedup model is presented. first The first observation shows that large percentage of storage is occupied by a very small number of large files that have low sub-file redundancy a very small number of large files that have low sub-file redundancy occupy a large percentage of storage. This observation shows that a weak hash function is sufficient for a large file to hash collision avoidance. On avoidance. On the other hand, this model suggests that tiny file can be ignored in the Deduplication process [12] [14]; and finally, compressed files have low chunk level of data redundancy. [15]

Comment [Reviewer12]: ?

Comment [Reviewer13]: ?

In the second observation observation, the data shared between different types different types of application is noted. This observation shows that the amount of these data is negligible. So Hence, application-aware Deduplication can in fact improve Deduplication efficiency.