Backup Optimization Method for Cloud Backup System

Boris, Atayan

National Polytechnic University of Armenia
Yerevan, Armenia
e-mail: borisn70@gmail.com

ABSTRACT

The incremental backup method is proposed for Cloud backup system to minimize the cost of data restore and to optimize the multiple backup processes of data that had been previously backed up. The overhead of backup processing is decreased by operation of incremental backups in between the operations of a full backup. The optimal time scheme for incremental and full backups in Cloud Backup System is derived.

Keywords

Cloud backup, data restore, data backup, incremental backup, backup optimization

1. INTRODUCTION

Due to the rapid development of cloud technology nowadays cloud services operating in that field are providing convenient tools for users to store their backup data in the Cloud while spending fewer money resources compared to traditional backup storages.

Reasonable data transfer capacity and improvements are turning cloud backup technology into a widely used alternative option to portable media, for example, tape.

As of now, there are two prominent ways to deal with cloud data backup: Software-as-a-Service (SaaS) and distributed cloud storage. As another option to onsite software, backup SaaS is a Web-based application facilitated and worked at Web area and accessed via a browser-based interface. SaaS backup is usually described as a model with shared, scalable architecture that can host multiple users with virtually separated data.

Cloud storage services are a mix between of onsite and remote software components. Usually, somewhat big IT organizations have onsite control of their software and hardware infrastructure for backup (e.g., data centers housing network and storage resources). On the other hand, when using Cloud backup services, customers are charged on an actual consumption basis, based on capacity or bandwidth, etc.

Overall, there are several advantages to using cloud backup:

- Productivity and reliability. Cloud providers use cutting age technology, such as disk-based backup, compression, possibly encryption, server virtualization, storage virtualization, etc. In addition to that, many providers offer 24/7 monitoring, management, and support that may not be afforded by many companies to administer their backup infrastructure on-site. Also, there is no need to think about or manage software and hardware upgrades and migrations, the burden of the backup infrastructure completely lies on the service provider.
- Ease of scalability. Cloud backup users can take the advantage of the unlimited scalability backup infrastructure without being charged for it upfront. In fact, the pay-as-you-go model is significantly convenient for backup that paying for full local architecture. This use model helps to have a predictable and efficient management of capacity growth and operational costs.
- Decreases recovery time of small data backups. For example, a recovery from tape takes much more complicated steps: an operator would need to recall the tape, load it, locate the data and recover it. In contrast, file recovery from cloud storage is faster because it doesn't require extra steps such as tape handling or seek time. Data to be recovered is located and downloaded over the Internet which saves time and eliminates the need for a local infrastructure.
- Accessibility. Cloud backup is more attractive to customers that couldn't afford the investment and maintenance of a disaster recovery infrastructure. Offsite data copies that in case of cloud backup are accessible from any place and device where Internet connection is available, provide more data insurance if any regional disaster happens.

Advancements in cloud technology and backup are promoting exciting development in cloud backup. Cloud backup technology provides undeniable benefits to organizations with limited IT resources and budget, including the abovementioned advantages.

Of course, as in any other technology area, some disadvantages also exist. One of the most crucial obstacles in cloud backup systems is the problem of multiple backup of unchanged data that had been already backed up. The cause of this problem is the absence of checks on database and file levels. It is obvious that backup process of large amounts of data is the most sensitive of the described problem. The effective incremental backup scheme is adopted in cloud backup system to solve this issue. The proposed solution is based on effective planning and time schedule of incremental backups.

There are two main methods which are used for backup data inside the cloud storage: full and incremental backups. When we are saying a full backup, we mean a complete data copy which includes all the files or databases inside the system. The main disadvantage of the full backup is the cost. It takes a very long time for archiving besides the storage for the backup data is quite large. To overcome this main disadvantage, it is suggested to use an incremental backup. The advantage of an incremental backup is that it takes the least time to finish. Incremental backup provides a faster method of backing up data than repeatedly running full backups. During an incremental backup, only files changed since the most recent backup are included. That is where it gets its name: each backup is an increment for a previous backup. The disadvantage of incremental backups that affects the overall system performance the most is that, usually, it is time-consuming to restore [1]. Let’s suppose full backups are done once in a week, incremental backups are done once in a day. Now, if the data from Thursday needs to be recovered, first Monday's full backup needs to be restored. After that, Tuesday's and Wednesday's backups will be recovered. There is definitely a possibility of a situation that some of the intermediate backups will be not available or corrupted. In that case, it will be impossible to perform the full restoration.

Conspicuously, the time estimation for a data recovery depends on the interval of full backups. For instance, if a full backup is executed every five days, the recovery operation can be done by utilizing all contents of data updated for the past five days. A complete backup of a mass data might need a voluminous overhead of data transfer. In this situation, it would be obligatory to consider the time interval and backup methods from the viewpoints of cost and reliability.

We suggest a method which reiterates a full backup on a short cycle. This method assures a prompt data recovery; however, it is obvious that the operation cost will be remarkably incremented. This is considered an old backup method. Rather than this method, we are suggesting a newly adopted way for an incremental backup. Suggested method can be used for massive data backup and will reduce the overhead of backup operation.

2. The scheme of incremental backup in cloud backup system

A full backup operation will obtain all the information in a second iteration. On the other hand, an incremental backup requires only modified data after the full backup. Keeping this in mind as an overhead of an incremental backup is small, we can execute it in a small-time interval between full backups. Concepts of full and incremental backups are shown in Fig.1. [2]

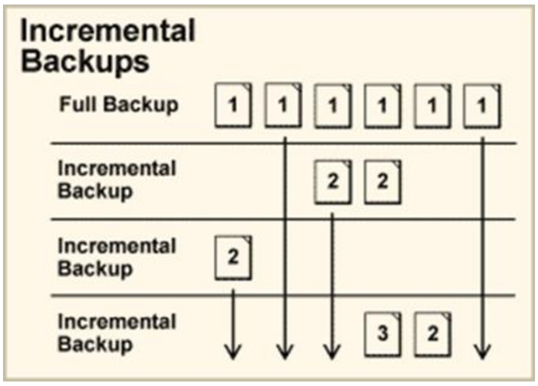

Fig.1 Incremental backup scheme

A backup method with incremental and full backups is described in this paper which solves the above mentioned issue in Cloud Backup System. Some schemes concerning system recovery mechanisms have been proposed in [3, 4], especially discussions on optimization problems of system checkpoints when failures occur. The ordinary backup operation cost is analyzed for an incremental backup and its optimization is discussed. Then, the expected cost of a backup is calculated and an optimal interval of full backup is discussed to minimize it.

In this paper, we are presenting a mixed backup process with both full and incremental backups. In the first round a full backup process is executed and at every T intervals of the backup, for example, after $T = 3$ incremental backups a full backup is processed. However, if the amount of the updated files exceeds a threshold value K_F, suppose the amount of the total files of the system is N, the update rate can be calculated with the following formula:

$$R = 100K_B/N\,\% \qquad (1)$$

and a full backup process can be executed regardless of the number of incremental backups. Consequently, after a full backup is processed and the situating information of a data is initialized, an incremental backup is executed again [5]. During the incremental backup process, newly updated trucks are captured and only the changes made since the last incremental backup are included in the current backup, so all updated data after a full backup become new. Therefore, the number of updated trucks at the r backup is generally more than that of incipiently updated trucks after the second backup.

Now, let us consider some other input variables that may have an impact on backup process and its optimization. There are two main Cloud storage specific metrics that we need to consider:

- bandwidth, particularly, upload rate,
- storage, how much storage is available on Cloud server to contain a backup.

At this phase, we will not focus much on storage, as it may affect to the decision to do incremental or full backup only in the situation when a decision is made to upgrade incremental backup to a full one, but there is not enough storage to hold it. In this case, at some point of time in the near future, the backup process will stop because of the out of space server. On the other hand, the bandwidth is an important metric of Cloud Backup System that can affect backup type decision. If the network upload speed at time point of the j^{th} backup is lower than a threshold value K_B, then incremental backup should be done, regardless of number or updated files, as we aim to an overall faster backup process, which means lower latency time for user (less waiting time to have backup uploaded on server).

Let's denote the number of updated files in the backup system with M_j.The total number of files updated at the j^{th} backup is given by the following equation:

$$Z_j = \sum_{i=1}^{j} M_j \qquad (2)$$

We can calculate the overhead of an incremental backup when a number of updated trucks is x with the cost function $c(x)$. In this case, the x is increasing and the cost of a full backup is a constant cf. Obviously, we can see that from the idea of the threshold of updated files that $c(K_F) = cf$. The cost of the i^{th} incremental backup, when the number of updated files in backup system has not exceeded a threshold value K_F and upload rate $B_i \geq K_B$, is:

$$h_i = c(Z_i), \qquad Z_i \leq K_F,\ B_i \geq K_B\ (i = 1,2,\dots,T-1) \quad (3)$$

The cost of operations of incremental backup repeated j times after a full backup is:

$$H = \sum_{i=1}^{j-1} h_i\ (i = 1,2,\dots,T-1) \qquad (4)$$

So, the idea here is that in case of $1 \leq j \leq T$, a full backup is executed when the number of updated files at j time exceeds a threshold value K_F and bandwidth B_j is higher than lower bound value K_B.

Moreover, in case of $j = T$, a full backup is executed at scheduled interval T without regard to the number of updated files. The following formula is used to calculate the overall cost of 1 backup cycle, which means a full backup and number of incremental backups until the next full backup:

$$H = c_f + \sum_{j=1}^{T-1} h_j \qquad (5)$$

The result is the optimal value of $T*$ which gives the minimal cost for the whole chain of backups. To find out the $T*$ the following inequality should be solved:

$$\sum_{j=0}^{T-1} h_j \geq c_f \qquad (6)$$

It is obvious, that if $h_1 \geq c_f$, which means that the cost of the first incremental backup is more than the cost of a full backup, then $T* = 1$ (no incremental backup is needed, perform full backup).

In this paper we have stressed out the time minimization of the backup process, but research shows that the mentioned method is also effective for improving other resource consumption metrics such as storage and in some cases processor time. Introducing incremental backups to the backup process, the overall storage consumption will be lower, as the size of some backup archives will be smaller than when compared with a size of full backup archives. For many Cloud services, saving storage means saving money resources, which is important both for the user and for the backup service provider. Regarding processor time, there may be some cases when the computational cost of incremental backup will be lower than the cost of full backup. In this case, it will be possible to also save on processor time costs.

The following numerical example will show how this method works. The constant input values are:

$$N = 20,\ K_F = 15,\ K_B = 128\ Kb/s$$

Let us define the cost calculation function as follows:

$$c(x) = S_A * V * x \qquad (7)$$

where S_A is an average size of one updated file and V is a constant chosen to indicate the speed of backup operation for 1 KB, we choose $V = 0.001$s. Example files and their sizes are given in Table1.

Table 1. Files in Cloud backup system with random sizes.

$File\ No$	$File\ Size\ (KB)$	$File\ No$	$File\ Size\ (KB)$
1	5	11	8
2	10	12	18
3	12	13	22
4	17	14	40
5	12	15	19
6	13	16	17
7	20	17	5
8	35	18	10
9	17	19	10
10	23	20	25

To simulate a backup process, random numbers of updated files, the average size of one updated file and upload rate are chosen. Those values are given in Table 2.

Table 2. Random numbers of updated files and bandwidth

j	M_j	$S_A(KB)$	$B_j(Kb/s)$
0	10	12.4	256
1	5	10	256
2	2	16	512
3	15	15.4	64
4	7	18.4	128
5	4	13.75	128
6	1	5	256
7	16	15.31	256

Using (7) we can calculate the cost of backup operation at the time *j* (Table 3).

Table 3. Cost of backup operation at time *j*

j	0	1	2	3	4	5	6	7
$C(M_j)$	0.124	0.05	0.032	0.231	0.1288	0.055	0.005	0.244

The cost of one full backup:

$$c_f = 16.4 * 0.001 * 20 = 0.328$$

The sum of the costs from Table 3:

$$\sum_{j=0}^{3} C(M_j) = 0.477 > 0.328$$

so $T*$ might be 3, but $B_3 = 64 < K_B$, so $T* = 4$.

3. CONCLUSION

The incremental backup process of Cloud Backup System was analyzed in terms of costs of full and incremental backup operations. The optimization method of incremental backup schedule was presented which was based on cost calculations which in their turn depend on a threshold value of updated files K_F and lower bound of upload rate K_B. Used method helped to evaluate the total expected cost of ordinary incremental backups and to minimize it.

It is important to understand, that minimizing the cost of backup operation doesn't necessarily mean minimizing the number of incremental backups between full backups. For example, there could be a situation where full backups are done weekly by schedule. But, using this method the backup administrator might identify that weekly full backups are not optimal, instead, full backup schedule could be shifted to be done once in a month with incremental backups on weekly basis.

Described method also has a positive impact on Cloud storage and performance consumption, particularly, the resource consumption is decreased in comparison with old backup schemes.

At each point of time the number of updated files and upload rate in Cloud Backup System are random, so as a result, the proposed method will be used in future work to create a backup process model using the theory of stochastic processes. Optimal $T*$ intervals of backups will be calculated based on examples of some probability distributions used for describing updated file numbers and upload rate. Examples of optimal incremental backup schedules for Cloud Backup system will be discussed, as well as the effects of probability distribution mean and variance on the interval $T*$.

REFERENCES

[1] Techtarget.com, "Full, incremental or differential: How to choose the correct backup type", *Online*, 2008

[2] Versionbackup.eu, "Backup management", *Online*.

[3] G.M. Lohman, J. A. Muckstadt, "Optimal Policy for Batch Operations: Backup, Checkpointing, Reorganization, and Updating", *ACM TODS*, vol.2, no.3, pp.209-222, 1977.

[4] J.W. Young, “A First Order Approximation to the Optimum Checkpoint Interval”, *Comm. ACM*, vol. 17, no. 9, pp. 530-531, 1974

[5] R.E. Barlow, F. Proschan, “Mathematical Theory of Reliability”, *Wiley*, 1965.