

Augmenting Performance For Distributed Cloud Storage

Matthew B. Hancock
Department of Computer Science
Rensselaer Polytechnic Institute
Troy, NY USA
hancom@rpi.edu

Carlos A. Varela
Department of Computer Science
Rensselaer Polytechnic Institute
Troy, NY USA
cvarela@cs.rpi.edu

Abstract—The device people use to capture multimedia has changed over the years with the rise of smartphones. Smartphones are readily available, easy to use, and capture multimedia with high quality. While consumers capture all of this media, the storage requirements are not changing significantly. Therefore, people look towards cloud storage solutions. The typical consumer stores files within a single provider. They want a solution that is quick to access, reliable, and secure. Using multiple providers can reduce cost and improve overall performance. We present a middleware framework called Distributed Indexed Storage in the Cloud (DISC) to improve all aspects a user expects in a cloud provider. The process of uploading and downloading is essentially transparent to the user. The upload and download performance happens simultaneously by distributing a subset of the file across multiple cloud providers that it deems fit based on policies. Reliability is another important feature of DISC. To improve reliability, we propose a solution that replicates the same subset of the file across different providers. This is beneficial when one provider is unresponsive, the data can be pulled from another provider with the same subset. Security has great importance when dealing with consumers data. We inherently gain security when improving reliability. Since the file is distributed using subsets, not one provider has the full file. In our experiment, performance improvements are observed when delivering and retrieving files compared to the standard approach. The results are promising, saving upwards of eight seconds in processing time. With the expansion of more cloud providers, the results are expected to improve.

Keywords—distributed cloud storage; cloud computing; cloud services;

I. INTRODUCTION

The mobile phone market has continued to grow rapidly with no indication of slowing down. Followed by browsing the web, the most popular features on a smartphone are the camera and video capabilities. The storage space on a smartphone to hold these multimedia files is very limiting compared to a computer or cloud storage. This limitation influences users to look toward cloud storage options. Cloud storage is known for its simplicity and ease of use, the ability to share and communicate with friends, and the availability of files from anywhere with an internet connection. These features empower a user to be more productive.

Cloud storage is great on paper, but has its own limitations as well. Files stored with a cloud provider are susceptible to

compromised security, have poor network performance during the upload or download process, and are only accessible when the provider is operating. The problem that arises is how can we produce a more secure, more efficient, and fault tolerant approach while using multiple storage providers and leveraging a concurrent model. A distributed cloud storage approach provides an elegant solution that is transparent to the user. There are a number of techniques that tackle only a portion of the cloud providers shortcomings [1]. The majority of these techniques are geared toward the enterprise market, not allowing the consumer market to take advantage of the concurrent cloud model.

II. CLOUD STORAGE PROVIDER LIMITATIONS

Users by nature see cloud storage as a means for reliable backups, sharing of data, twenty-four hour access, and synchronization across multiple devices. As seen in Table I, a typical free account offers a small amount of storage space. Many users have accounts with multiple providers to avoid paying for more storage space. Utilizing a single cloud provider's resources proves to have many drawbacks.

Table I: Storage Providers Statistics as of 2014

Source	Free Storage	Paid Storage
Dropbox [2]	2GB	1TB \$9.99/month
Box [3]	10GB	100GB \$10.00/month
Google Drive [4]	15GB	100GB \$1.99/month 1TB \$9.99/month
One Drive [5]	15GB	100GB \$1.99/month 200GB \$3.99/month 1TB \$6.99/month

A. Increased Cost

The key reason to use cloud storage is the amount of extra space it provides. However, the provided free accounts are well below the sufficient amount of space needed to store files. Multimedia files are among the most popular file type to be stored in the cloud because they provide hassle free backup and easy sharing capabilities. Depending on the type and quantity of data, paid accounts are necessary for cloud storage to be effective. Budget must now be considered when

using cloud storage. The demand to increase communication bandwidth and maintenance to cushion the impact of occasional outages add yearly costs. As demands increase, it is likely cost will follow [6].

B. Limited Bandwidth

A major source of performance problems for cloud services is the communication time between the client computer and the web server in the cloud [6]. The bandwidth to upload or download files from the cloud are throttled to help improve overall performance among all active users. For instance, a user has 1.5GB of data stored with Dropbox who throttles the network bandwidth to 1.5MB/s. This would require 17.06 minutes assuming full bandwidth throughout the entire process.

C. Decreased Security

Storage providers are getting hacked more frequently than ever before. With many enhancements in security and firewall protection, simple attacks like phishing or brute force can lead to unauthorized access to millions of users sensitive information. Many attacks are blocked, but a few are granted unauthorized access to user accounts and files. Storing all files in one location, in an unencrypted form for some providers, shows to be a security flaw.

D. Unpredictable Availability

There are two kinds of provider outages known as permanent and temporary. A permanent outage is when a cloud provider goes out of business while a temporary outage occurs when the cloud provider is unavailable for a span of time [6]. Here, we are referencing temporary outages.

Twenty-four hour access is the ideal expectation, but unanticipated errors are likely to occur. A simple outage can occur from a network problem or a database corruption problem. Both cases prevent a user from accessing or creating new files. During one incident in August 2014 [7], Microsoft encountered a problem with their Visual Studio Online Service that resulted in five-and-a-half hours of outage time. This instance hindered users productivity worldwide.

III. DISTRIBUTED STORAGE AND MANAGER

We developed Distributed Indexed Storage in the Cloud (DISC) to be an expandable framework that follows the concurrent model for communicating and interfacing with cloud storage providers. With each given file, DISC will follow a set of user policies to mitigate some of the limitations to using a single cloud storage provider. These policies can be used to configure how DISC works for different scenarios and preferences.

A. User Policies

1) *Millionaire Policy*: Users want to access their files one-hundred percent of the time. However, this does not come without a cost involved. This model is based on a user who is willing to spend extra money in return for more space. With the extra space, we replicate the files across all cloud storage providers. Higher replication increases the probability the requested file will be accessible from one of the n providers.

2) *Security Policy*: Moving data from the local source to a public cloud opens the possibility for data to be vulnerable and intercepted during transfer. DISC takes an approach where each sequential byte is stored with a different cloud provider. Splitting a file is determined using the following equation:

$$CSP_i = F_{byte}, \quad i = F_{byte} \bmod CSP_{total} \quad (1)$$

This controls which byte of the original file F_{byte} is stored with which cloud storage provider CSP_i . Each byte location is calculated using the total number of cloud storage providers CSP_{total} .

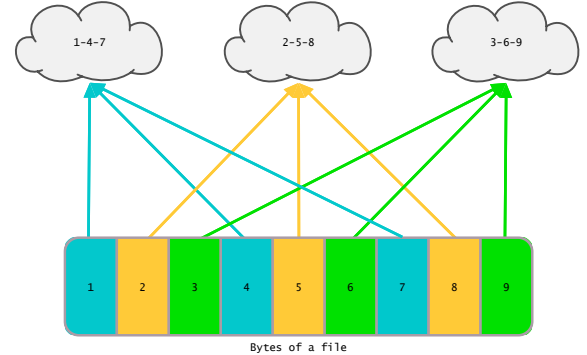


Figure 1: Security Policy Splitting

3) *Availability Policy*: Performance is an important factor to the user experience. To improve performance, a file must be split into multiple files, now rendering each split file unreadable. The availability policy separates the files, similar to the security policy, while creating usable and viewable partitions. Figure 2 shows an experiment using two cloud providers implementing the availability policy.

Taking the even and odd pixels from the original image creates two new images. The user will always see the combined image when both providers are operating. In case one provider fails to operate, the image is still viewable in its original aspect-ratio. To achieve the same aspect-ratio, we replace unreachable portions of the image with pixels in close proximity. This policy can be used for videos by splitting the video based on its frames rather than pixels.

Increasing the total number of cloud providers will produce better results. In most cases, only one provider will be nonfunctional. Simple math shows that as the denominator

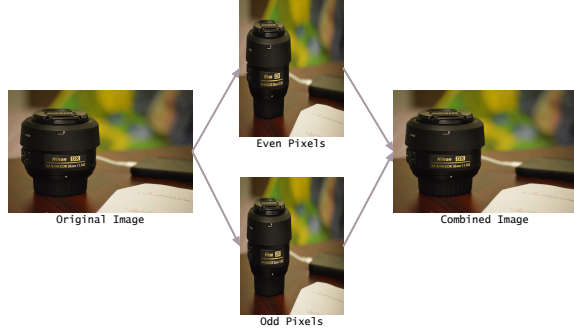


Figure 2: Availability Policy Experiment

approaches infinity, the viewable image approaches one-hundred percent. The percentage P of the original image can be determined with the following equation where CSP_{online} is the number of operating providers:

$$P = \frac{CSP_{online}}{CSP_{total}} \times 100 \quad (2)$$

4) *Budget Policy*: More storage space is available, but only at a premium cost. This policy tackles the problems for a user with limited space. DISC will compress the file until reaching a specified tolerance. With each compression, DISC will alter the file size by half. This is achieved by removing a quarter of the row pixels and a quarter of the column pixels. Unlike other policies, this results in a permanent file reduction.

5) *Low Replication Policy*: Rather than replicating the files among all storage providers, this policy calculates the number of replicas to create using a subset of the total storage providers. Replication is a valued feature, but their budget may be constrained. We can now increase the probability the file is accessible, while reducing the total storage space. The number of replicas R is determined using the following equation:

$$R = \left\lceil \frac{CSP_{total} - 1}{2} \right\rceil \quad (3)$$

The file must be stored in at least one location. Therefore, we subtract one from the total cloud providers since we cannot make a replica on the same storage provider.

B. Data Throughput Management

During an upload or download, bandwidth is constantly changing, which can enhance or hinder the time taken to process the request. Using a feedback loop, we can alter which cloud providers will receive more of the data and which will receive less. After each iteration, calculations are performed and if the difference in throughput between cloud providers falls below a threshold, DISC will alter the 1:1 ratio to be skewed. The cloud provider that performs better, will receive more data to process while the contrasting provider receives less data. This alteration will level out extreme cases where one cloud provider is performing poorly, requiring excess throughput times.

1) *Throughput Model*: Each cloud provider gets its own thread container, encapsulating multiple pieces of information including a file queue, internal variables, and methods. A file queue simply holds a list of filenames to be uploaded or downloaded. Supervising all these threads is a Manager. The file queue is accessible to the Manager to allow migration of files between threads. Internal variables and methods are all accessible to the Manager, but cannot be changed outside of the thread container.

Within the Manager is a list of threads and methods to facilitate the migration procedures. The Manager is called upon, each time a file is processed. At this time, statistics from each thread are gathered followed by calculations to determine whether one cloud provider should gain or lose a file. To improve performance, there are three times as many files as cloud providers. The extra files, which are of equal size, allow the Manager to detect early on if one cloud provider has slow performance in which a migration procedure will occur.

2) *Migration Procedure*: A migration procedure is the process of altering the responsibility of a cloud provider by removing one file from the file queue and transferring this file to another file queue in a different thread. A migration only occurs if one of the four migration rules is satisfied. These decisions are based on two statistics stored within each thread consisting of the number of files processed and the time taken to process the most recent file.

There are two scenarios that cause the Manager to leave all thread containers unchanged. The first occurs when the thread with the lowest processing time and the thread that called the Manager, are the same. No improvements can be made knowing that thread is performing the fastest. This will occur more likely when threads start to exit as time increases. The next scenario uses statistics from the number of files processed. Completing a file increases the number of files processed by one. However, if the lowest files processed is zero, there is not enough information to give a good indication that the other provider is performing slowly. This information can only infer that the thread finished first and the other threads may or may not finish close behind.

File migration occurs during two similar scenarios. The first scenario leverages the time taken to process the most recent file from each thread. The minimum time among all threads is compared to the time taken from the thread who called the Manager. The difference between these two times must be greater than a tolerance level. This tolerance level is set at 1.5 seconds. It is not advantageous to migrate if the difference is too small. The next scenario occurs when the lowest number of files processed is zero and the thread that called the Manager has completed processing two or more files. In contrast to before, we can confidently say we need to migrate. Having a 0:2+ ratio means one provider has completed two files in the same time another provider has completed zero. This is a strong indication that one provider

is operating at a subpar performance.

IV. PERFORMANCE EVALUATIONS

In the multiple experiments for uploading and downloading files, we use two instances of Dropbox and two instances of Google Drive. The network performance is measured as 68.86Mbps download and 12.45Mbps upload. Throughout the various tests, we use a 20.9MB image file. The purpose for a high resolution image is to compare overall quality with performance of splitting or rescaling the image.



Figure 3: Experimental Results

Seen in Figure 3a and Figure 3b, there is no particular order in which the cloud providers completed their request which occurred during all test cases. In the security-minded test scenario, we are able to decrease the total upload time by 35% and a decrease of 13% in download time. The majority of the time is used to concatenate downloaded files in the correct order to reproduce the original image. Only considering the download time, there is a 65% decrease. Along with stronger security, we gain significant improvements in upload and a slight gain in download performance.

Another test scenario was performed using the availability policy seen in Figure 3c and Figure 3d. We again see improvements in upload and download performance when removing the time taken to process the image. Comparing the overall time, upload time decreases by 17% and download increases by 112%. In this experiment, the algorithm is

manipulating the individual pixels which requires more time to preserve the RGB values.

Figure 3e and Figure 3f prove the benefits of using a Manager. Comparing all four iterations, there is no way of predicting which cloud provider will process how many files. This is the result from an inconsistent bandwidth problem. We see how Google performs slow during one iteration, but performs well during a different iteration. These sudden changes are caught by the Manager and corrected.

Using DISC without a Manager allowed for a wide range of total processing times. This difference is considered wasted time. In an ideal scenario, zero seconds are wasted and all cloud providers require the same amount of time to complete processing the file. By using a Manager and migrating files between file queues, we are able to decrease the range in processing times.

V. CONCLUSION

Current approaches to cloud storage have shown to be inadequate, leveraging only a single provider. Security vulnerabilities, poor performance, and unpredictable reliability are some shortcomings to using a single provider. Knowing cloud providers throttle the bandwidth, we can leverage the concurrent cloud storage model to take full advantage of the local user bandwidth by processing files concurrently. We have presented a middleware framework known as Distributed Indexed Storage in the Cloud (DISC) that leverages this model. As a work in progress, we are investigating a better tolerance level for migration and the footprint this framework has on resource constrained devices.

ACKNOWLEDGMENT

A special thanks to Shigeru Imai and Antwane Mason for reviewing the content of this paper.

REFERENCES

- [1] D. Slamanig, C. Hanser, "On Cloud Storage and the Cloud of Clouds Approach," in *Proc. of the 7th Int. Conf. for Internet Technology and Secured Transactions*, 2012, pp. 649-655.
- [2] Dropbox Inc., San Francisco, CA, USA, "Plans - Dropbox," Available: <https://www.dropbox.com/plans/>. Accessed: Dec. 29th, 2014.
- [3] Box Inc., Los Altos, CA, USA, "Plans and Pricing — Box," Available: <https://www.box.com/pricing/>. Accessed: Dec. 29th, 2014.
- [4] Google Inc., Mountain View, CA, USA, "Buy and manage storage plans - Drive help," Available: <https://support.google.com/drive/answer/2375123?hl=en> Accessed: Dec. 29th, 2014.
- [5] Microsoft Inc., Redmond, WA, USA, "OneDrive Plans," Available: <https://onedrive.live.com/about/en-us/plans/>. Accessed: Dec. 29th, 2014.
- [6] W. Kim, S. Kim, E. Lee, S. Lee, "Adoption Issues for Cloud Computing," in *Proc. of the 7th Int. Conf. on Advances in Mobile Computing and Multimedia*, 2009, pp. 2-5.
- [7] G. Keizer. "Customers praise Microsoft's 'no BS' explanation of cloud service outage," Network World Inc., Framingham, MA, USA. [Online]. Available: <http://www.networkworld.com>