# Personal Cloud Storage Performance Benchmarking

**Kartik Kapoor[1], Kushagra Verma[2], Kushal Jain[3]**

[1-3]*Student, Dept. of Computer Science, Indian Institute of Information Technology, Sri City*

---

**Abstract -** *Cloud-based storage can offer advantages like data security, scalability, and availability. As cloud storage technologies advance and become available to the masses, users are offered with numerous options of storage technologies, facing a dilemma: which personal cloud platform to use. Many of these platforms provide customers with various features and low-cost storage. While many users are attracted by these offers, other important aspects, such as underlying architecture and synchronization performance, are mostly unknown given the proprietary design of most services. This paper proposes a methodology to analyze and benchmark personal cloud storage services. The implications of different design choices on the performance are assessed by executing a series of benchmarks.*

***Key Words***: Personal Cloud, Benchmarking, Client-side capabilities, Performance, Google Drive, Dropbox, OneDrive

## 1. INTRODUCTION

With personal cloud storage becoming more and more popular among users, many companies are starting to offer their own cloud services. Many people are being attracted by these cloud storage services as they provide many features like low-cost storage, no hardware setup, data durability, and synchronization of data among multiple devices. These companies are trying to attract new customers by offering a number of new features, low-cost storage, or even some free storage to the new customers. Despite the increasing interest in cloud storage services, users have very little knowledge about the underlying architecture and design choices and their implications on the end-user performance.

Our goal is to compare these services based on various performance parameters. For that, we developed certain methodologies and carried out experiments. Our experiments help understand the architecture used by these services, client capabilities implementation, etc. The results of these experiments can be used to determine which cloud service is best for a particular type of usage.

Our experiments include a series of benchmark tests, each designed to analyze a specific capability of the cloud storage service. We use this benchmark to determine differences in client-side software features, data center placements, etc. Then we use the results to analyze their implications on the overall user experience. These experiments are designed and executed from the perspective of an average user from India. For this reason, we chose the three most popular cloud storage services in India, namely, Google Drive, Dropbox, and OneDrive.

Our results reveal interesting insights into how different the performance of each of these services is due to the implementation or lack of certain infrastructure and design choices. No clear winner can be declared based on the results as each of the services handles user data very differently and each one is suitable for a different type of user and different workloads.

## 2. METHODOLOGY

In this section, we explain the methodology followed by us to benchmark the performance of personal cloud storage services. We designed a tool that generates synthetic workloads and obtains performance figures. The tool is composed of two parts (i) a computer which runs the benchmarking application; and (ii) a virtual machine that runs the testing application.

We set up a Linux server which runs the benchmarking application and hosts a virtual machine that runs the testing application. For this experiment, the testing application was run in Windows 10 Home version. The server configuration consists of an Intel i7 processor and 8GB of RAM. The server is connected to 1 GB/s Ethernet Network. We ensured that the server performance and internet connectivity is not a bottleneck.

The benchmarking application simulates real-world usage by creating synthetic workloads. It then measures the performance of the cloud storage service and outputs the performance figures. The testing application receives benchmarking parameters as input which describes the series of operations to be performed. The benchmarking application generates specific workloads to simulate real-world usage. These files are then synchronized to the cloud by the testing application. The exchanged traffic between the testing application and the cloud is then monitored to compute performance metrics.

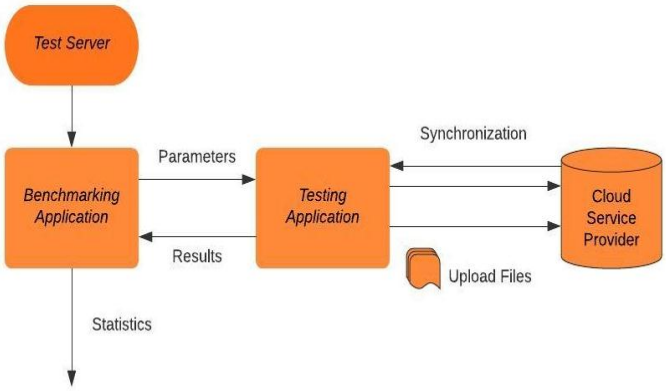


**Figure -1**: Test setup

### 2.1 Geolocation of Data Centers

Each personal cloud storage provider has many data centers spread around the world in order to ensure data durability and availability. The geolocation of these data centers has strong implications on the performance of the storage service.

To geolocate the data centers of the cloud servers, we have devised a set of experiments. First, a list of DNS names of all the servers contacted by our testing application is compiled. To determine the front-end IP address of these servers, the DNS names are resolved using more than 1000 open DNS resolvers spread around the world [1]. As the cloud services rely on the DNS to distribute workload, it helps us to check if load balancing techniques are implemented to route customers from different places to different IP addresses.

For each IP address, we have to determine the geographic location of the server. Since the data provided by popular geolocation databases is known to be unreliable [2], we make use of a simple methodology that includes:

(i) Official information: may be provided by the server owner
(ii) Airport Codes: by performing a reverse DNS lookup, the information retrieved often embeds airport codes
(iii) Server Round Trip Time: the shortest Round Trip Time (RTT) from multiple PlanetLab nodes is measured. The server returning the minimum RTT is considered to be the closest location to the storage server.
(iv) Using Traceroute: it helps us to get the domain names of intermediate routers, whose geolocation can then be determined using the above-listed methods.

These methodologies can provide an estimation of actual geographical location with a precision of about a hundred kilometers.

### 2.2 Checking Implementation of Capabilities

Personal cloud storage applications implement several client-side capabilities to optimize storage usage and to speed up synchronization. These capabilities include the adoption of different methods like:

(i) Chunking (splitting the data into smaller size units)
(ii) Bundling (transmitting multiple small files as a single object)
(iii) Deduplication (avoiding re-transmission of content already available on the servers)
(iv) Delta Encoding (transmitting only the modified portions of a file)
(v) Data Compression (reduction in the number of bits needed to represent data)

For each case, we designed specific tests to determine if the given capability is implemented. Our testing application produces specific batches of files to test a specific capability. The exchanged traffic is then monitored and analyzed by the benchmarking application to determine how the service operates.

### 2.3 Performance Benchmarking

Now, we check how the geolocation of the data centers and the system capabilities influence synchronization performance. To do this, we have designed several benchmarks covering a variety of synchronization scenarios. These benchmarks use varying

(i) number of files; (ii) file sizes and (iii) file types, to cover different synchronization scenarios. All these files are created at run-time by the testing application.

Each experiment is repeated 20 times per service. The server is kept idle for at least 10 min between experiments to avoid creating aberrant workloads to the servers. It is important to note that all the tests are conducted in the same controlled environment, from a single location. The results may vary while measuring from different locations.

### 2.4 Selected Personal Cloud Services

We mainly focus on 3 services for this experiment. We have selected the most popular[3] and most used services in India which are Google Drive [4], Dropbox [5], and OneDrive [6].

## 3. SYSTEM ARCHITECTURE

### 3.1 Protocols

All the listed clients exchange traffic using HTTPS. All the services use separate servers for client management and data storage. This can be determined by monitoring the

traffic exchange when the client (i) starts; (ii) is sitting idle; and (iii) synchronizes files.

We noticed some differences among applications during login and idle phases. This is mainly due to two reasons. Firstly, after logging in the applications check if any content has to be updated. One can notice that OneDrive takes up 100 kB in total as it contacts multiple Microsoft servers during login, as opposed to Dropbox and Google Drive which require very little data to connect to the respective servers. Secondly, after login is completed, the application keeps exchanging data with the cloud servers. We noticed that Google Drive polls the servers every 40 seconds, as opposed to Dropbox and OneDrive which poll the server every minute.

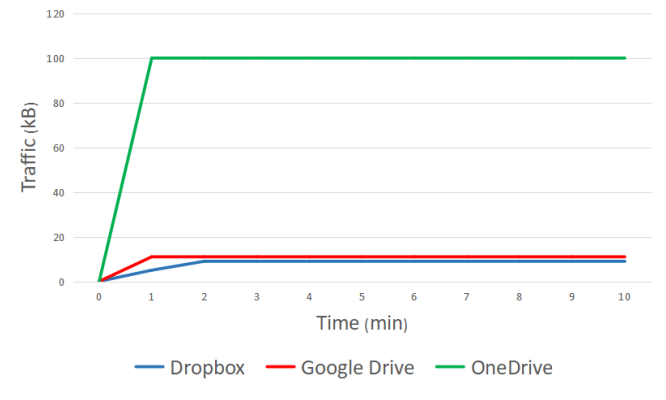


**Chart -1**: Background traffic while idle

### 3.2 Data Center Locations

Dropbox uses its own servers for client management and Amazon servers for storing the user data. OneDrive relies on Microsoft's data centers and Google Drive relies on Google's data centers.

With our experiments, we found out that OneDrive makes use of three different Microsoft data centers situated in Mumbai, Pune, and Chennai. Google Drive has one data center situated in Mumbai. Dropbox does not have any data centers in India. It makes use of its data centers situated in the USA.

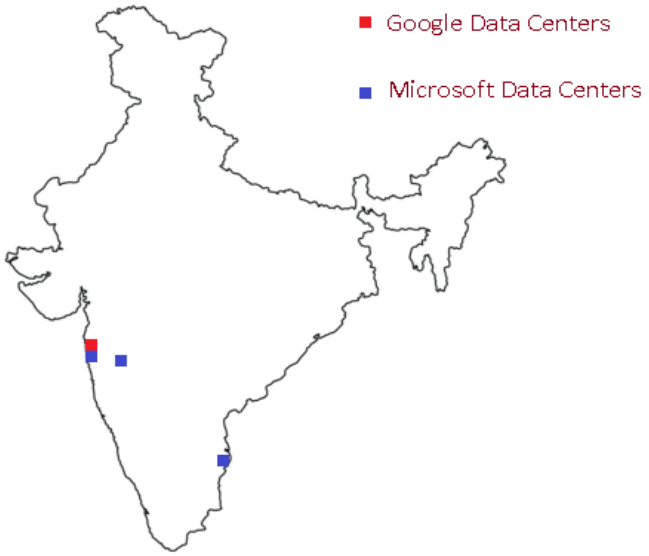


**Figure -2**: Data Center locations in India

## 4. CLIENT-SIDE CAPABILITIES

### 4.1 Chunking

We designed our first test to understand how large files are processed by different cloud storage services, whether they are exchanged as single objects, or split into chunks. We can determine this by monitoring throughput during the upload of files differing in size. Our experiments show that Dropbox uses 4 MB chunks while Google Drive uses 8 MB chunks. OneDrive uses variable chunk sizes.

Chunking simplifies upload recovery in case of failures, which may be beneficial for users connected to slow and unstable networks.

### 4.2 Bundling

Our second test is designed to understand how cloud storage services process a batch of files that are small in size. When transferring a batch of files, they can be bundled and pipelined to reduce the transmission latency. Our tests are designed to monitor how services handle different batches of files. For this, we made 4 batches of files, containing 1, 10, 100, and 1000 files respectively. Each batch of files has the same size (1 MB).

Our experiments revealed different types of bundling strategies implemented by cloud services. Google Drive opens a separate TCP connection for each file, which limits the client performance when several files have to be synchronized. OneDrive reuses a single TCP connection, but submits files sequentially and waits for acknowledgments after each upload. We observed that only Dropbox implements a file-bundling strategy.

### 4.3 Data Deduplication

Upload capacity can be saved by identifying replicas of the data already present on the storage server and eliminating them from the client folder. To check whether the services implement client-side deduplication, we designed the following test: (i) a random file is inserted in a random folder; (ii) a replica is generated with a different name in a second folder; and (iii) the original file is copied to a third folder.

From the results, we observe that only Dropbox implements data deduplication. All other services upload the same data even if it is already available at the storage server.

### 4.4 Delta Encoding

Delta encoding is a way of storing or transmitting data that calculates differences between two files and allows the storage or transmission of only the difference between the two files. To identify whether services implement delta encoding, a set of files is generated such that there is only a small difference in each subsequent file. The changed content may be at any random position within the file. The files are then uploaded sequentially to replace the old file. Traffic exchange between the testing application and the cloud server is then monitored to draw conclusions.

Our experiments show that only Dropbox implements delta encoding. We observed that the amount of traffic increases when files are bigger than 4 MB-long chunks of Dropbox. This happens because the original content may be shifted, changing two or more chunks at once. Google Drive and OneDrive do not implement delta encoding and upload the whole file to replace the old one.

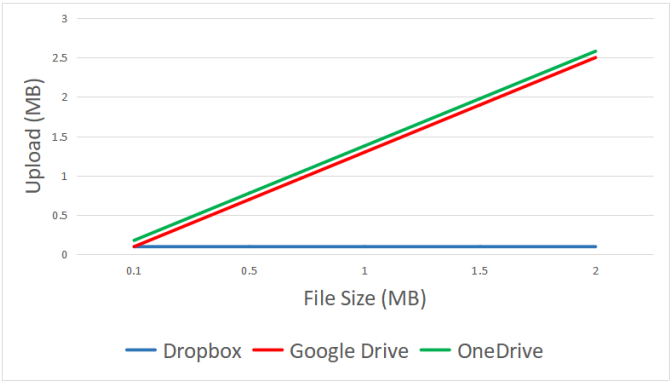


**Chart -2**: Delta Encoding Test

### 4.5 Data Compression

Data compression is a process of representing data using fewer bits than the original representation. Generally, compression can reduce storage and traffic requirements. To check whether the services use the compression capability, we use three distinct file sets. The first set consists of text files which are highly compressible. The second set of files contain pure random bytes so that no compression is possible. The third set consists of files with JPEG extension but is actually filled with text. Our experiments reveal that Dropbox and Google Drive compress files in sets one and two before transmission. In the case of the third set, Google Drive identifies JPEG files and does not perform compression, whereas Dropbox compresses all files irrespective of their content. OneDrive does not perform any kind of compression before file transfer.

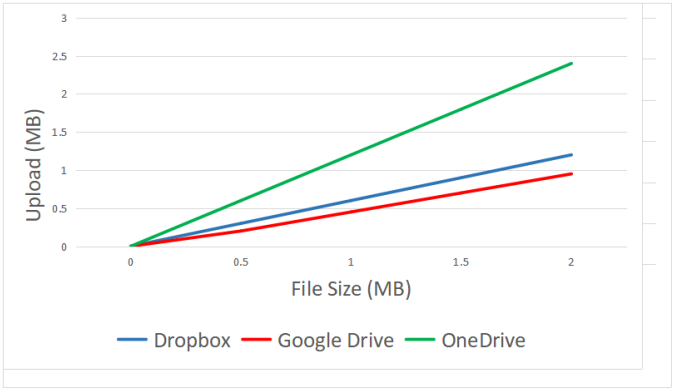


**Chart -3**: Uploading Random Text Files

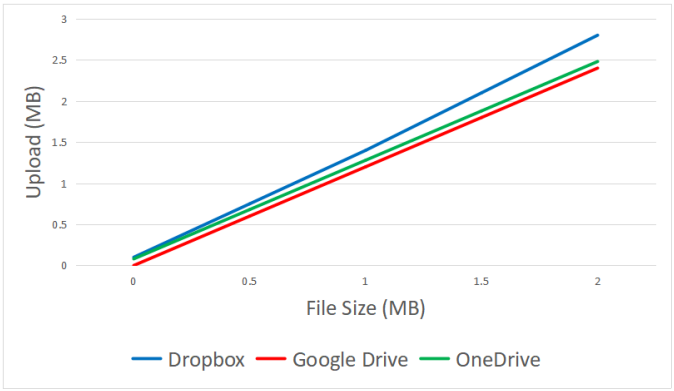


**Chart -4**: Uploading Random bytes

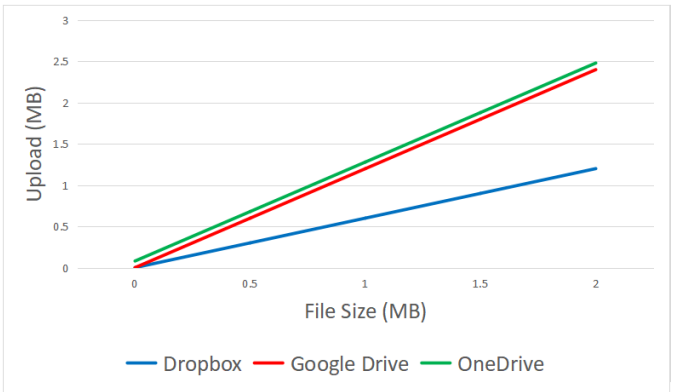


**Chart -5**: Uploading JPEG files

### 4.6 Summary

Table 1 summarizes the capabilities implemented by each service. It can be observed that Dropbox has implemented most of the client-side features to enhance synchronization speed. Google Drive implements only chunking and compression. OneDrive only implements chunking.

**Table -1:** Capabilities implemented by each service

| Capability | Dropbox | Google Drive | OneDrive |
|---|---|---|---|
| **Chunking** | 4 MB | 8 MB | variable |
| **Bundling** | yes | no | no |
| **Compression** | always | smart | no |
| **Deduplication** | yes | no | no |
| **Delta-encoding** | yes | no | no |

## 5. CLIENT PERFORMANCE

### 5.1 Startup Time

We first evaluate the time taken by each service before synchronization starts. This could reveal whether implementing advanced capabilities on the client side increases initial synchronization startup time.

We observed that Dropbox is the fastest service to start synchronizing single files. However, in the case of multiple files, the startup is slightly delayed because of the use of bundling strategy. OneDrive sits idle for at least 5 s before starting synchronizing files. Moreover, OneDrive gets slower as the number of files increases, as it does not implement any bundling strategy. Google Drive takes a bit less time than Dropbox when synchronizing single files. This may be due to the presence of Google's data center in India. Like OneDrive, Google Drive also gets slower with the increase in the number of files due to a lack of implementation of a bundling strategy.

### 5.2 Completion Time

Next, we calculate the time taken by each service to complete the upload tasks. When synchronizing single files of 1 MB, the results are affected by the distance between our system and the data centers. We observed that Google Drive is the fastest in synchronizing files. It takes almost 300 ms to upload a 1 MB file. Despite having data centers in India, OneDrive needs almost 3 s to upload a 1 MB file, as opposed to Dropbox which requires only 1 s.

When multiple files are synchronized, the results are a bit different. Dropbox wins this time because of its bundling strategy. Google Drive and OneDrive struggle in this case due to the lack of implementation of bundling strategy.

The results with a moderate file size do not change much. Dropbox shows a small improvement in upload time due to the implementation of compression. Similarly, Google Drive manages to reduce upload time when sending a single file, due to its smart compression technique. However, it struggles when multiple files are synchronized. As OneDrive does not implement any compression technique, it is the slowest of all three.

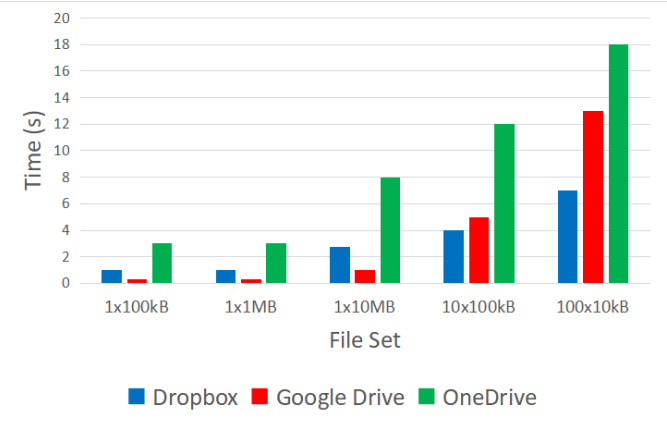


**Chart -6**: Upload times for different file sets

## 6. CONCLUSIONS

We presented several methodologies to analyze cloud service architectures, implementation of client capabilities, and their implication on the end-user performance. These methodologies were then applied to benchmark the three most popular cloud storage services from the perspective of an average Indian user. After analyzing the results of our experiments, we can say that each service handles user data differently. As all these services are so different in implementing features and processing user data, no clear winner can be declared.

Our experiments show how the architecture and implementation of client capabilities affect the end-user experience. Dropbox implements the majority of the listed client capabilities and tries to boost performance, but its lack of data centers in India greatly affects performance. On the other end of the spectrum lies OneDrive. Despite having three Microsoft data centers present in India, it struggles to perform well due to the lack of implementation of client capabilities. Google Drive lies somewhat in the middle of these two. It implements some client capabilities, which boosts performance, and utilizes one Google data center present in the country.

Our methodologies and benchmark tests prove to be useful to benchmark cloud storage services and make an

informed choice when looking for the most suitable cloud storage service provider for one's needs.

## REFERENCES

[1] I. Bermudez, S. Traverso, M. Mellia and M. Munafò, "Exploring the cloud from passive measurements: The Amazon AWS case," 2013 Proceedings IEEE INFOCOM, Turin, 2013, pp. 230-234, doi: 10.1109/INFCOM.2013.6566769.

[2] Ingmar Poese, Steve Uhlig, Mohamed Ali Kaafar, Benoit Donnet, and Bamba Gueye. 2011. IP geolocation databases: unreliable? SIGCOMM Comput. Commun. Rev. 41, 2 (April 2011), 53–56. DOI:https://doi.org/10.1145/1971162.1971171

[3] Google Trends. https://www.google.com/trends/

[4] Google Drive. https://www.google.com/drive/

[5] Dropbox. https://www.dropbox.com/

[6] OneDrive. https://onedrive.live.com/

[7] E. Bocchi, M. Mellia and S. Sarni, "Cloud storage service benchmarking: Methodologies and experimentations," 2014 IEEE 3rd International Conference on Cloud Networking (CloudNet), Luxembourg, 2014, pp. 395-400, doi: 10.1109/CloudNet.2014.6969027.

[8] Idilio Drago, Enrico Bocchi, Marco Mellia, Herman Slatman, and Aiko Pras. 2013. Benchmarking personal cloud storage. In Proceedings of the 2013 conference on Internet measurement conference (IMC '13). Association for Computing Machinery, New York, NY, USA, 205–212. DOI:https://doi.org/10.1145/2504730.2504762