Performance Evaluation of the Illinois Cloud Computing Testbed

Ahmed Khurshid, Abdullah Al-Nayeem, and Indranil Gupta

Department of Computer Science
University of Illinois at Urbana-Champaign

Abstract. Cloud computing has the potential to bring sophisticated computing facility at the doorsteps of users. The performance of services hosted at a cloud generally depends on its computation, storage and networking models. Since the storage and communication costs in a data-intensive application usually dominate the overall cost, we analyze the performance of the storage and communication facilities of a cloud. Little work has been done to measure the performance of existing cloud facilities. In this project, we experiment on the Open Cirrus cloud computing research testbed located at the University of Illinois. We evaluate the performance of this cloud environment while providing service to users located at different parts of the world. We use PlanetLab and Emulab as part of our experiments to mimic distributed users and cloud facilities. We observe that the performance mostly vary because of the network characteristics between the cloud facility and its users. Performance also depends on internal cloud architecture. Based on our experiences, we suggest different possible enhancements of existing cloud computing environments.

1 Introduction

Existing commercial cloud computing infrastructures such as Google App Engine [7], Microsoft Azure [2], and Amazon Web Services (AWS) [17] have proved the impact that cloud computing can have on current and future computations. These systems offer different levels of computation, storage and networking models to cloud users for deploying large scale applications. Cloud users can use these facilities over the Internet from anywhere in the world. For example, Amazon's AWS offers a number of high-level managed services. Amazon's Elastic Compute Cloud (EC2) [5] presents a virtual computing environment to cloud users. Amazon also rents out different storage services, for example Simple Storage Service (S3) [16].

These commercial cloud providers usually claim fast and reliable services. For example, Amazon says *"the Amazon EC2 Service Level Agreement commitment is 99.95% availability for each Amazon EC2 Region"* [5]. Similarly it claims 99.99% availability for the stored data at S3. Amazon also says *"Amazon S3 must be fast enough to support high-performance applications. Server-side latency must be insignificant relative to Internet latency."* [16]. These providers largely maintain their promises on availability and performance. However, there had been disruptions in these services in the past. In February 2008 and July 2008, S3 was unavailable for 8-10 hours for different reasons [1]. Garfinkel [8] performed detailed evaluation of Amazon EC2 and S3. In that study, the author experienced a sudden and persistent reduction in throughput during the experiments. The author could not discover the actual cause as Amazon did not disclose detailed information about its internal architecture and possible changes in that architecture. The author also observed that users located at different parts of the world experienced varying levels of performance while accessing these AWS services.

These changes in performance and availability may happen due to internal cloud configurations and network connectivity between the cloud and its users. If a cloud user decides to host its mission-critical services at a public cloud, it might worry about these variations in performance and availability. Therefore, tracking the perceivable performance is necessary from a cloud user's perspective so that it can adapt to the changes or take corrective measures. A cloud user may also decide to host applications at multiple clouds to ensure high-availability. Based on the performance measurements, the user may decide to switch between clouds. Moreover, this knowledge of user perceived performance will allow cloud providers to formulate better pricing models and realistic Service Level Agreement (SLA). However, there has not been enough work in the literature that addresses these issues.

In this project, we perform a thorough analysis of a cloud's performance during data transfer. We conduct our experiments at an Open Cirrus [19] testbed. Open Cirrus is an open cloud computing testbed (CCT) that aims to foster design and development of services, tools and service management frameworks targeting large scale clusters. Its open nature gives researchers sufficient flexibility while dealing with different aspects present in large scale computing infrastructures. Open Cirrus is the outcome of the collaboration between different organizations, e.g. University of Illinois at Urbana-Champaign, HP Labs, Intel Research, Yahoo Research, etc. Our experiments are currently done at the Illinois site only. Since the knowledge of its internal architecture is readily available, we can analyze the results with more accuracy. For experimentation, the Illinois CCT is accessed from nodes present in PlanetLab [15] to emulate spatial diversity of users. We also use Emulab [6] to emulate another cloud facility.

From our experiments, we observe that data transfer performance of a cloud varies while serving different users. Much of these variations can be attributed to the network characteristics between the cloud and its users. We find that the distance between a cloud facility and a user is a significant factor in determining the performance. Internal cloud architecture, e.g., topology, file system, also affect the performance.

Based on these experiments, we now have a better understanding of a cloud facility. We believe there are several ways to improve current cloud computing technology. We present a number of suggestions in this regard. For example, many of the existing technologies and practices may not be well suited for future data-intensive applications. At present, the distributed file system at the Illinois CCT does not support parallel data access. This is a limiting factor and therefore, the distributed file system must be modified to support scalable, parallel access to the storage. Moreover, current commercial clouds do not support any performance-oriented pricing for data transfer. In this report, we suggest the need for a pricing model that considers both data size and transfer throughput. We plan to work further on this pricing model in our future work. We also emphasize the importance of cache utilization and automated system monitoring facility in existing cloud environments.

2 Related Works

In order to make computational facilities easily accessible, cloud computing has emerged as an important research topic in recent years. Till now, most of the research works related to cloud computing focused on developing efficient and scalable data processing frameworks such as Hadoop [10]. There has been little work on analysis of performance of cloud computing infrastructures as experienced by users from different parts of the world. Garfinkel [8] experimented on the performance of different cloud services provided by Amazon and their interoperability. The author measured the data throughput and transactions-per-second (TPS) for read/write operations between S3 and different end-hosts located either in EC2 or at remote locations. The author found that the best performance in data throughput is achieved if the data is kept completely inside Amazon's cloud environment. Data transfer throughput for different locations varied significantly, although all of those locations were capable of achieving higher data transfer rates (as claimed by the author). It was also observed that the size of the transferred objects influenced data transfer throughput. Since Garfinkel did not know the exact cloud configurations at Amazon, he could not discover the root cause of this performance variation. We follow similar measurement methodology to evaluate the performance of the Illinois CCT.

Significant amount of works have been done on available bandwidth measurement to locate bottlenecks in the Internet [11]. Since different cloud users access the cloud facilities from different parts of the world, their data transfer throughput can vary due to differences in network characteristics. Varying link capacity, routing anomalies, network congestion and link instability can cause variations in available bandwidth. Paxson [14] studied end-to-end routing behavior in the Internet from different locations. According to this study, the growth of the Internet is causing more and more routing and network anomalies. This situation along with heterogeneity in network connections cause geographically dispersed hosts to experience variations in available network bandwidth. Strauss *et al.* [18] expressed the importance of available bandwidth estimation and presented a light-weight tool for measuring the same. Although we focus on the performance variations at cloud infrastructures, these studies are complementary to our project to isolate the effect of network characteristics on the performance.

Numerous works have also been done on system-level monitoring of large scale infrastructures [12], [21]. Currently, Zenoss [21] is used at the Illinois CCT. Similarly, application-specific monitoring for cloud environments are also supported by different monitoring tools, e.g. Chukwa [3]. These monitoring tools present operational status of the cloud infrastructure to the system administrators, but not to the cloud users.

3 Open Cirrus testbed at Illinois

The Illinois CCT [20] consists of 128 HP DL160 compute nodes with dual quad core CPUs, 16GB of RAM, and 2TB of disk space (Figure 1). The cluster also has an additional 4 network-attached storage nodes having 288TB of space. Each of these compute and storage nodes has a separate 1Gb/s and 10Gb/s link respectively to different switches.

The testbed is partitioned into two separate logical clusters. The first cluster is named Altocumulus and is used for running MapReduce [4] jobs using Hadoop [10]. It consists of 64 compute nodes running Hadoop Distributed File System (HDFS) on approximately 96TB of space. The second cluster, named Cumulonimbus, consists of the remaining 64 compute nodes. This cluster is dedicated for systems research and provides the capability to run experiments on dedicated hardware. The storage facility allocated to Cumulonimbus is accessed through Network File System (NFS).

Currently, we access only 16 compute nodes in the Cumulonimbus cluster and 5TB of networked storage at one storage node. This storage facility is provided as a mounted directory at each of the compute nodes accessible to us.

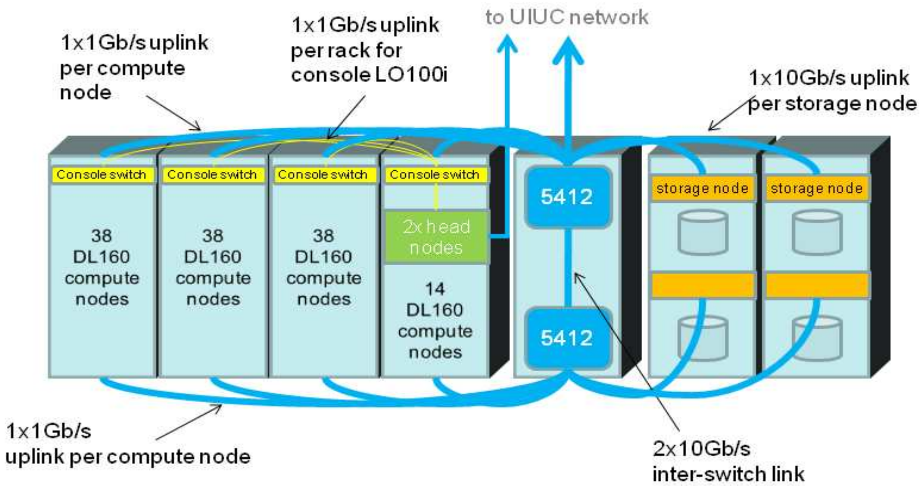


Fig. 1. Architecture of the Illinois CCT

3.1 Limitations

The Illinois CCT is still not matured enough compared to other commercial clouds. It does not support any virtualization. Thus in some respects, it still does not support elastic computing facilities like Amazon EC2. Although the network-attached storage facilities available in this testbed mimic the Amazon S3 service, currently this storage service is not directly accessible from any external computer. This storage facility can be accessed only via a compute node acting as a gateway to the storage nodes. Considering these limitations, the Illinois CCT may not represent a commercial cloud. Still, we believe that some of our experimental results can be used to detect the root cause of performance variations and degradations.

4 Experiment with the Storage Facility of the Illinois CCT

Cloud computing facilities are largely used for data-intensive applications, for example log processing or storing data at the cloud. User-perceived data transfer throughput is an important metric for the effectiveness of these applications. We, therefore, have implemented a throughput measurement program to measure the storage and network performance of the Illinois CCT.

As mentioned earlier, storage nodes are accessed through certain gateway (or compute) nodes. Hence, the internal network characteristic can affect the storage performance, in addition to the external network (the Internet) characteristics. In this project, we have set up experiments to observe the impact of network characteristics on the storage performance.

This section describes different experiments we performed to measure the storage performance of the Illinois CCT.

4.1 Internal Cloud Storage Access from Compute Nodes

This experiment shows the performance of internal storage access. The results are also significant in analyzing the relative effect of the internal cloud performance on user experience. In this experiment, we transfered data between 16 compute nodes and the only storage node allocated to us. In one case, the gateway nodes accessed the storage node sequentially and separately. In another case, all compute nodes accessed the storage node concurrently.

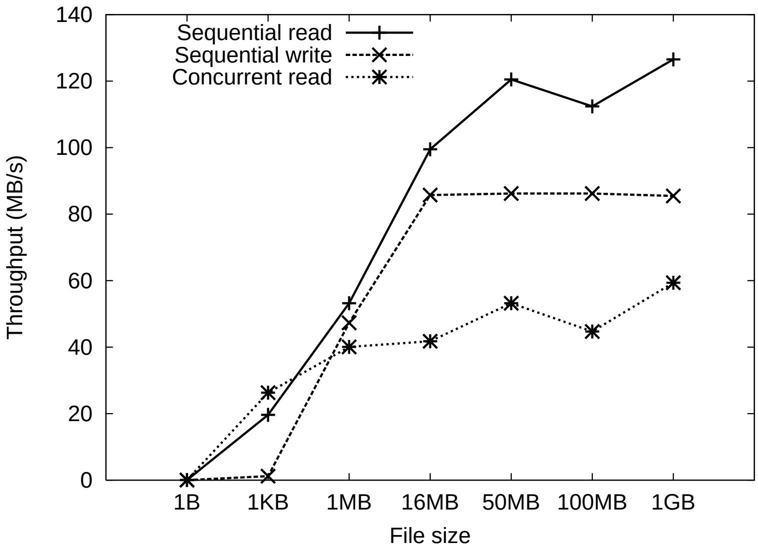


Fig. 2. Relative performance of read-write operations within the cloud

Figure 2 shows the average data throughputs between the compute nodes and the storage node. We transfered files of sizes 1B, 1KB, 1MB, 16MB, 50MB, 100MB, and 1GB.

As shown in the figure, the read operations are always better than the write operations when the compute nodes accessed the storage node sequentially. This improved performance can mostly be attributed to NFS caching. Although we randomized file accesses, we could not completely avoid the caching effect. As a result, both server and client-side caching influenced the storage performance during read operations. Write operations achieved lower throughput compared to read operations as there is no effect of caching during the write operations. In case of reads, it is evident that file sizes of 50MB or more give the highest achievable throughput; for write this optimum value is 16MB. Moreover, Figure 2 shows that write performance becomes almost constant for file sizes beyond 16MB. This can be attributed to write characteristics of the storage elements. In case of concurrent reads, we observe degraded performance compared to both sequential read/write operations. As all the 16 compute nodes (each having 1Gb/s link) access the storage facility (having 10Gb/s link) simultaneously, it creates contention for the available bandwidth and results in reduced throughput.

4.2 Read Operations by Distributed Users

In order to simulate distributed users, we measure the end-to-end data transfer throughput from the Illinois CCT to five remote PlanetLab nodes located in different continents: North America (mit.edu, ucla.edu, uchicago.edu), Asia (naist.jp) and Europe (unibo.it). These nodes are almost symmetrically placed with respect to the Illinois CCT. We assume that users access the cloud storage from these distributed locations by connecting to one of the gateways of the Illinois CCT.

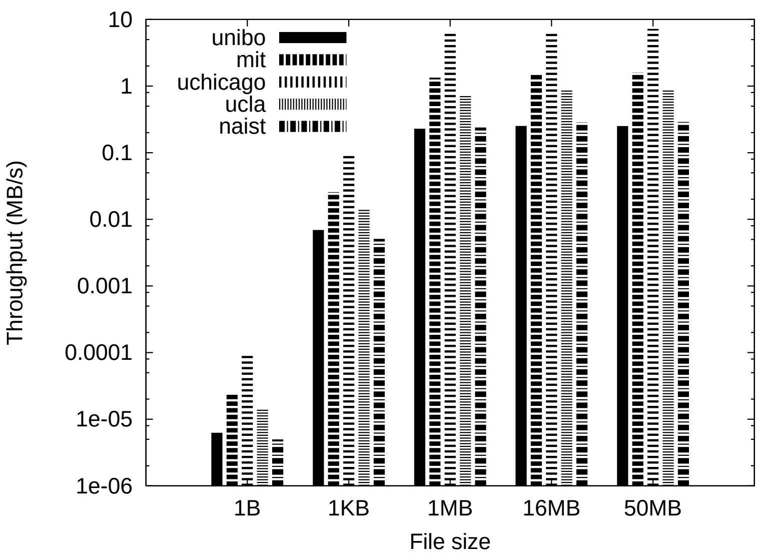


Fig. 3. Relative performance of sequential read operations from the Illinois CCT to distributed PlanetLab nodes

Currently, we have tested with files of sizes 1B, 1KB, 1MB, 16MB, and 50MB. We have transferred a total of 1875 files from the Illinois CCT to different PlanetLab nodes. We have not used larger files because of the bandwidth usage limit of 5.4GB/day on each PlanetLab slice. This limitation has also prevented us to perform fine grained performance measurement on hour-by-hour basis.

Figure 3 shows the end-to-end data transfer throughput from the Illinois CCT to these PlanetLab nodes. In general, we observe that the node at uchicago.edu, being closest to the Illinois CCT, experiences the highest end-to-end throughput and the node at naist.jp experiences the lowest end-to-end throughput. This result indicates that throughput is inversely proportional to the distance from the Illinois CCT. However, the corresponding read throughputs from the storage nodes to the gateway nodes does not vary significantly in our experiments. Therefore, we may assume that these variations in end-to-end throughput are mostly due to the network characteristics between the Illinois CCT and these remote nodes. Cloud architecture or file size do not have any influence on these variations. Figure 3 also shows that throughput is maximized by using files of size 1MB or larger.

5 Experiment with Distributed Cloud Facilities

Currently, lack of standardization in cloud computing APIs prevents cloud users from deploying their services at multiple cloud facilities. Such methods can be quite advantageous for cloud users as it ensures fault-tolerance of the offered services and improves availability. This also allows cloud users to move their data and/or tasks to a cloud that gives better performance. In [9], Greenberg *et al.* emphasized on the use of multiple cloud data centers and geo-distributed computational facilities.

When a cloud user has access to multiple cloud facilities, it can selectively decide where to run the computation and where to put the associated data. As shown in Figure 4, if a cloud user decides to use the computational facility of Cloud 1 but experiences low throughput while communicating with this cloud, it may choose to send the data to Cloud 2 so that data transfer during computation can take place using the high performance path between Cloud 1 and Cloud 2. Users can also select which cloud to use based on their geographical locations. But before that, cloud users need to be aware of the data transfer throughput that can be achieved between two clouds. Currently, we only have access to the Illinois CCT. As we do not have access to other cloud facilities, we employ an alternative strategy to perform the aforementioned experiments. We use Emulab to emulate a cloud computing facility. Instead of accessing the data from local Emulab machines, we measure the data transfer throughput while accessing the networked storage of Emulab. However, Emulab is not built to provide cloud computing facility, rather is more suited for other systems and networking research. Thus, results obtained from these experiments may not necessarily portray the actual performance characteristics of cloud-to-cloud communication. Still, we are able to have some insight on data transfer characteristics between two infrastructures hosting a cluster of machines shared by multiple users. In addition to measuring the data

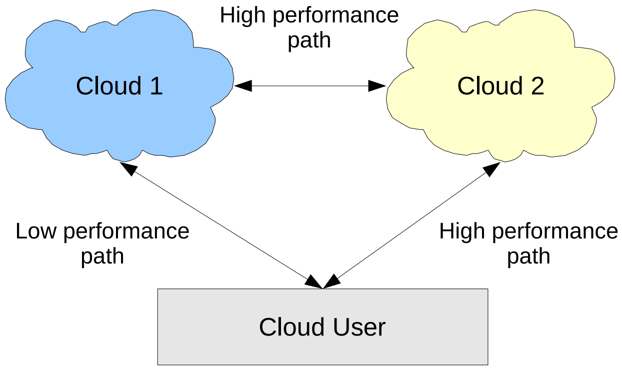


Fig. 4. Cloud-to-Cloud communication

transfer throughput between the Illinois CCT and Emulab, we also measure data transfer throughput when distributed users (i.e., PlanetLab nodes) access the Emulab facility.

5.1 Read Operations from the Illinois CCT to Emulab

Figure 5 gives the inter-cloud data transfer throughput between the Illinois CCT and Emulab. It is clear from this figure that the average throughput does not change significantly for any file larger than 16MB. We have not considered any concurrent operations in this experiment. Comparing both Figure 5 and 2, it is evident that the inter-cloud network characteristics dominates the throughput, even though their internal storage may have better performance.

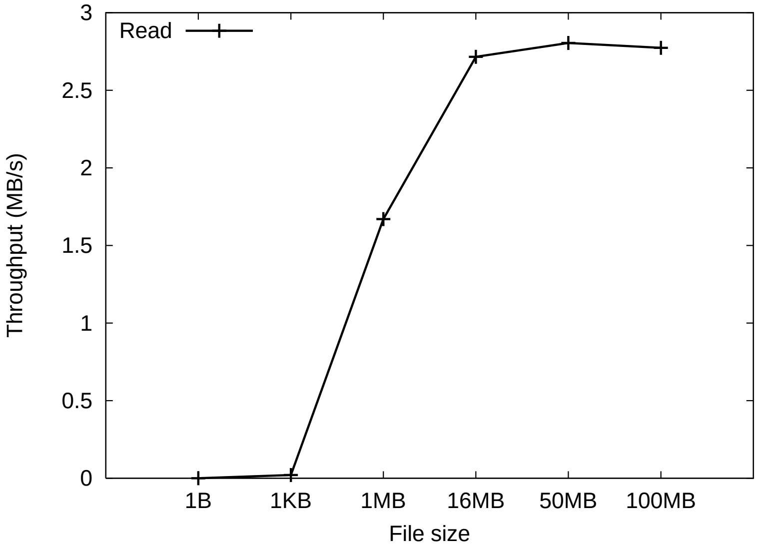


Fig. 5. Relative performance of sequential read operations at the Illinois CCT by Emulab nodes

5.2 Read Operations from Emulab to Distributed Users

We have measured the end-to-end data transfer throughput from Emulab to our five remote PlanetLab nodes. Here we have also tested with files of sizes 1B, 1KB, 1MB, 16MB, and 50MB. Figure 6 shows the end-to-end data transfer throughput from the Emulab nodes to

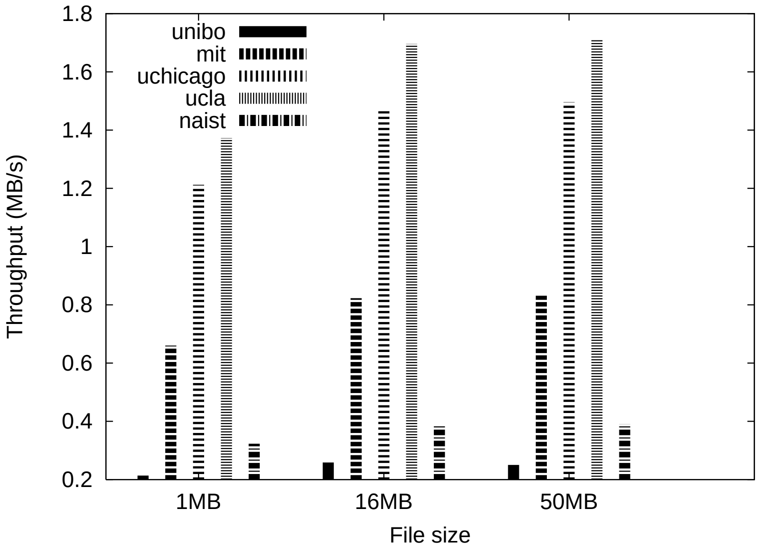
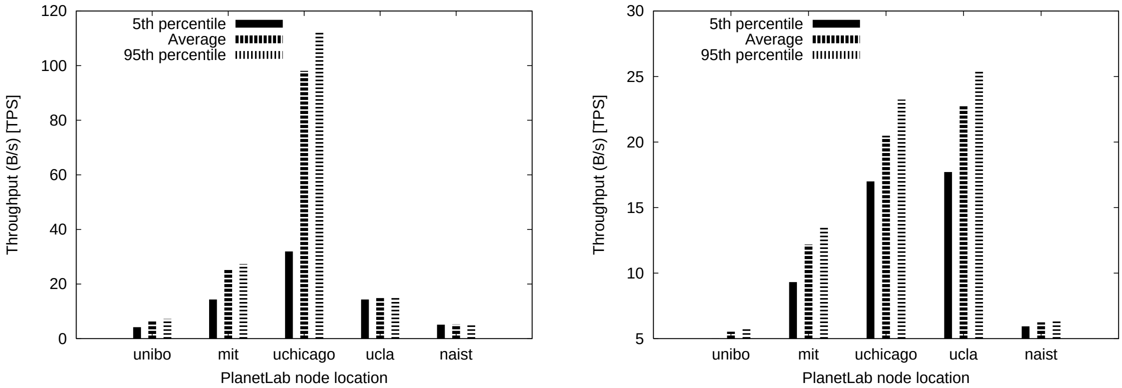


Fig. 6. Relative performance of sequential read operations at Emulab by distributed PlanetLab nodes

the PlanetLab nodes. Again, the performance variations for different PlanetLab nodes can be explained by considering their geographical distance from the Emulab facility at University of Utah. In this case, ucla.edu is located closer to Emulab than other four PlanetLab nodes and hence experiences the highest end-to-end throughput. On the other hand, the node at unibo.it, being the farthest PlanetLab node from Emulab, experiences the lowest end-to-end throughput. This result also indicates the dominant effect of network characteristics, such as bandwidth and latency, on end-to-end data transfer throughput.

5.3 Comparison between Transactions per Second (TPS) of the Illinois CCT and Emulab

TPS corresponds to the maximum number of requests that can be managed per second by a cloud facility. Figure 7 compares the TPS of the Illinois CCT and Emulab. To calculate TPS, we consider the throughput of 1B data transfer from PlanetLab nodes to the Illinois CCT and Emulab. In our experiments, we initiate every transaction after the successful completion of the previous one. Hence, the throughput of 1B files is a measure of transaction speed. Variations in bandwidth among different links are not going to affect the throughput of 1B file transfers. So, TPS values are solely determined by the network latency between the cloud facility and its (distributed) users. Figure 7 supports this argument. In this figure, PlanetLab nodes nearer to the Illinois CCT or Emulab achieve higher TPS values than distant ones. Therefore, geographic distance between a cloud facility and its users dictates the data transfer performance heavily.



(a) TPS at the Illinois CCT (b) TPS at Emulab

Fig. 7. Comparison between TPS of the Illinois CCT and Emulab to different PlanetLab nodes

6 Suggestions

Based on our experiences, we propose the following suggestions to enhance current cloud computing facilities.

1. *Distributed File System:* Distributed file systems used in the cloud must be scalable with the growing demand of data-intensive operations. These file systems must support parallelism as a mechanism to handle huge data transfer. Currently storage nodes at the Illinois CCT use NFSv3 as the distributed file system. This version of NFS does not support parallel data access. NFSv4 adds such parallelism by supporting the parallel NFS, pNFS [13]. In parallel NFS, meta-data is separated from actual data. It allows a clients to access the storage devices directly and in parallel. The pNFS architecture can eliminate the scalability and performance issues associated with NFS servers in deployment today, such as the one used in the Illinois CCT.

2. *Automated System Monitoring:* Since a cloud facility has a large number of compute and storage nodes, many system related problems are more likely to happen in different parts of the cloud. These problems may occur due to various reasons such as configuration errors, security bugs, etc. Automated, continuous monitoring of system metrics is necessary to detect and resolve these problems. The current monitoring tools of the Illinois CCT may not be sufficient for this purpose. For example, during the initial phase of our experiments, we found that three of the allocated compute nodes performed very poorly. We discussed this incident with the system administrator and confirmed that

these machines underwent networking reconfigurations. But these reconfigurations did not take place properly. As a result, their Ethernet port switched to 10Mb/s half-duplex mode instead of 1Gb/s full-duplex mode. Later on, rebooting those machines resolved the problem. While the solution to this problem was very straightforward, the problem was not automatically resolved with existing monitoring tools.

3. *Cache Utilization:* For data-intensive cloud computing, datasets are typically immutable. As a result, caching at both file servers and clients can significantly improve cloud performance. So, we believe that future cloud applications and infrastructures should focus on effective cache utilization.

4. *Pricing Model:* Currently, cloud storage providers charge at a flat rate based on the total amount of data transmission. For example, Amazon S3 charges $0.170 per GB per month for the first 10TB of data being transferred out of its cloud facility in United States. Amazon does not consider the data transfer performance as a deciding factor in the pricing model. We suggest that pricing should be a function of both data size and the throughput. For example, a cloud provider may decide to charge at a lower rate if the client experiences lower throughput. This initiative can encourage more distant users to buying the storage. It also implies more profit for the cloud provider, because the more users buy the storage capacity, the more money it can earn. However, there are certain challenges in implementing this pricing model. For example, both cloud provider and the user must agree on the perceived throughput to compute the cost of data transfer operations. Any of this entity may try to manipulate data transfer operations to gain undue advantages from this scheme. We have not explored this tussle between the cloud provider and the users in our experiments. We plan to continue on this topic in future.

7 Conclusion

In future, improved network facility and sophisticated software products will contribute to the development of cloud computing. It is important to provide quality service to users as long as the network between the user and the cloud infrastructure is free from bottlenecks and network anomalies. Our project aims to offer significant contributions to this cutting-edge research problem. Here, we perform a thorough evaluation of storage and communication performance of the Illinois CCT while serving distributed users. We find that both internal cloud configurations and network characteristics affect the data transfer throughput. It is quite likely that differences in performance of multiple clouds will make users to selectively choose better clouds to deploy data-intensive tasks. We believe that the results obtained from this project will provide important guidelines to design future cloud computing facilities and better service guarantees to the users.

References

1. M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. Katz, A. Konwinski, G. Lee, D. Patterson, A. Rabkin, I. Stoica, and M. Zaharia. Above the Clouds: A Berkeley View of Cloud Computing. http://berkeleyclouds.blogspot.com/.
2. Microsoft Azure. http://www.microsoft.com/azure/.
3. J. Boulon, A. Konwinski, R. Qi, A. Rabkin, E. Yang, and M. Yang. Chukwa: A Large-Scale Monitoring System. In *Proceedings of Cloud Computing and Its Applications*, 2008.
4. J. Dean and S. Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. *Communications of the ACM*, 51(1):107–113, 2008.
5. Amazon Elastic Compute Cloud (EC2). http://aws.amazon.com/ec2/.
6. Emulab. http://www.emulab.net/.
7. Google App Engine. http://code.google.com/appengine/.
8. S. L. Garfinkel. An Evaluation of Amazon's Grid Computing Services: EC2, S3 and SQS. Technical Report TR-08-07, Harvard University, August 2007.
9. A. Greenberg, J. Hamilton, D. A. Maltz, and P. Patel. The cost of a cloud: Research problems in data center networks. *ACM SIGCOMM Computer Communication Review*, 39(1):68–73, 2009.
10. Hadoop. http://lucene.apache.org/hadoop.
11. N. Hu, L. Li, Z. M. Mao, P. Steenkiste, and J. Wang. A Measurement Study of Internet Bottlenecks. In *Proceedings of IEEE INFOCOM*, 2005.
12. M. L. Massie, B. N. Chun, and D. E. Culler. The Ganglia Distributed Monitoring System: Design, Implementation, and Experience. *Parallel Computing*, 30(7), July 2004.
13. Parallel NFS. http://www.pnfs.com/.
14. V. Paxson. End-to-end routing behavior in the Internet. *IEEE/ACM Transactions on Networking*, 5(5):601–615, 1997.
15. PlanetLab. http://www.planet-lab.org/.
16. Amazon Simple Storage Service (S3). http://aws.amazon.com/s3/.
17. Amazon Web Services. http://aws.amazon.com/.
18. J. Strauss, D. Katabi, and F. Kaashoek. A Measurement Study of Available Bandwidth Estimation Tools. In *Proceedings of ACM SIGCOMM Internet Measurement Conference*, October 2003.
19. HP/Intel/Yahoo! Open Cloud Computing Research Testbed. http://opencirrus.org/.
20. Illinois Cloud Computing Testbed. https://agora.cs.illinois.edu/display/ccttech/home.
21. Zenoss. http://www.zenoss.com/.