2011

P2CP: a new cloud storage model to enhance performance of cloud services

Zhe Sun
zs789@uow.edu.au

Jun Shen
University of Wollongong, jshen@uow.edu.au

Ghassan Beydoun
University of Wollongong, beydoun@uow.edu.au

P2CP: a new cloud storage model to enhance performance of cloud services

Abstract

This paper presents a storage model named Peer to Cloud and Peer (P2CP). Assuming that the P2CP model follows the Poisson process or Little's law, we prove that the speed and availability of P2CP is generally better than that of the pure Peer to Peer (P2P) model, the Peer to Server, Peer (P2SP) model or the cloud model. A key feature of our P2CP is that it has three data transmission tunnels: the cloud-user data transmission tunnel, the clients' data transmission tunnel, and the common data transmission tunnel. P2CP uses the cloud storage system as a common storage system. When data transmission occurs, the data nodes, cloud user, and the non-cloud user are all together involved to complete the transaction.

Disciplines

Physical Sciences and Mathematics

Publication Details

Sun, Z., Shen, J. & Beydoun, G. (2011). P2CP: a new cloud storage model to enhance performance of cloud services. CONF-IRM 2011 Proceedings (pp. 1-12). USA: AIS Electronic Library (AISeL).

P2CP: A New Cloud Storage Model to Enhance Performance of Cloud Services

Zhe SUN, Jun SHEN, Ghassan BEYDOUN
University of Wollongong
zs789@uowmail.edu.au, jshen@uow.edu.au, beydoun@uow.edu.au

Abstract

This paper presents a storage model named Peer to Cloud and Peer (P2CP). Assuming that the P2CP model follows the Poisson process or Little's law, we prove that the speed and availability of P2CP is generally better than that of the pure Peer to Peer (P2P) model, the Peer to Server, Peer (P2SP) model or the cloud model. A key feature of our P2CP is that it has three data transmission tunnels: the cloud-user data transmission tunnel, the clients' data transmission tunnel, and the common data transmission tunnel. P2CP uses the cloud storage system as a common storage system. When data transmission occurs, the data nodes, cloud user, and the non-cloud user are all together involved to complete the transaction.

Keywords

Cloud, Storage, P2CP, and P2P.

1. Introduction

Cloud computing is an area of information systems that is undergoing rapid development. Many large corporations, e.g. Google, Amazon and Microsoft have recently been focusing on developing and releasing a number of related storage products such as Google file system (GFS), Amazon elastic compute cloud (EC2), Azure, etc. All of these use cloud distributed storage models based. These models are appealing as they can lead to a significant decrease in the utilization rate of bandwidth. During download session, the current alternative file sharing models based on Peer to Peer (P2P) communication all suffer high utilization rates of bandwidth and lower availability. In this paper, we have analysed several existing distribution storage models and designed a hybrid model, P2CP, which exploits the P2P protocol to enhance the data transmission performance and at the same time it uses a cloud storage system to provide continuous availability. For our purpose, we assume that the P2CP model follows the Poisson process or Little's law and mathematically prove that the speed and availability of P2CP is generally superior to that of the pure P2P model, the Peer to Server and Peer (P2SP) model or the pure Cloud model. The paper is organized as follows: Section 2 introduces related works and background on distributed storage models; and introduces a study of the existing distribution storage models; Section 3 proposes a new storage model design; Section 4 details comparison and evaluation; and Section 5 concludes with a discussion and summary of the analysis.

2. Related Work

This section overviews the three main models for distributed storage, P2P, P2SP and cloud storage models. It highlights key applications, strengths and weaknesses of each of the three models.

2.1 Peer to peer storage model

In a pure P2P storage model (Figure 1), each peer is equal. Peers can act as both clients and servers. In the P2P storage model, users get data from each other, but sometimes, the server or seed that contains the particular resource does not exist in the network, so the file sharing process has to stop. In the P2P storage model, there is no master server to manage the network, metadata, and data. Thus, it is hard to offer persistent availability. Well-known applications are Gnutella before version 0.4 (Kirk, 2003), Freenet (Clarke, 2000), Sorrento (Hong et al. 2004), etc. The Eliot file system (Stein et al. 2002) is a reliable mutable file system based on P2P block storage. The system exploits a metadata service in an auxiliary replicated database separated and generalized to isolate all mutation and client states. It consists of the following four components: an un-trusted, immutable, reliable P2P block storage substrate known as the Charles block service; a trusted, replicated database, known as the metadata service (MS), storing mutable nodes, directories, symlinks, and superblocks; a set of file system clients; and zero, one, or more cache servers intended to improve performance, but are not necessary for correctness. FS2You (Sun et al. 2009) is another large-scale online storage system. It also has four main components: directory server, tracking server, replication servers and peers. With the peers' assistance, it makes semi-persistent files available and reduces the server bandwidth cost.

Serverless Network Storage (SNS) is a persistent P2P network storage application. It has four layers, which are operation logic; a file information protocol (FIP) that exploits XML-formatted messages to maintain files and disk information; a proposed security layer; and a serverless layer, which is responsible for routine network state information (Ye et al. 2003).

2.2 Peer to Server and Peer

To solve the problem of persistent availability in pure P2P storage model (Figure 2), a hybrid P2P model emerged that is Peer to Server and Peer (P2SP). In this storage model, peers are distributed into the client group or the server group. The client group responses handle the data transmission, and the server group acts as a master server to coordinate the P2P structure. However, the workload of the master servers is very heavy, and furthermore, without the server group, the P2P network does not work. Well-known P2SP applications are eMule (Merkur, 2002), BitTorrent (Cohen, 2001) and FS2You. This latter is a large-scale online storage system. When the clients are going to download data, firstly, they download data from the server, and then, they exchange data with each other. If the other peers are not available, the client will download all the data from the server. In (Fang et al. 2009), differences between the pure P2P network and the P2SP network are analysed. The work assumes is that the peer arrival rate and departure rates follow the Poisson process or Little's law. Finally, they proved that P2SP has higher performance than P2P based on two assumptions

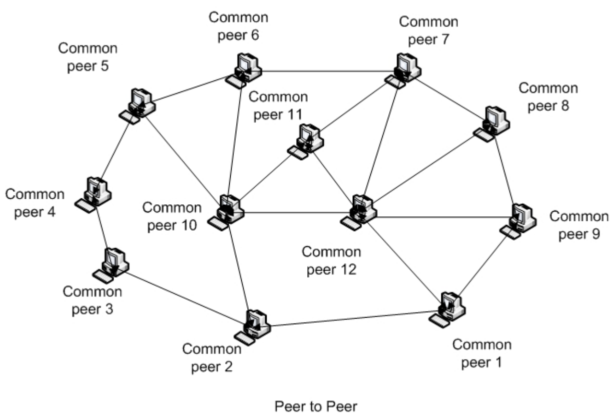

Figure 1: P2P Network model.

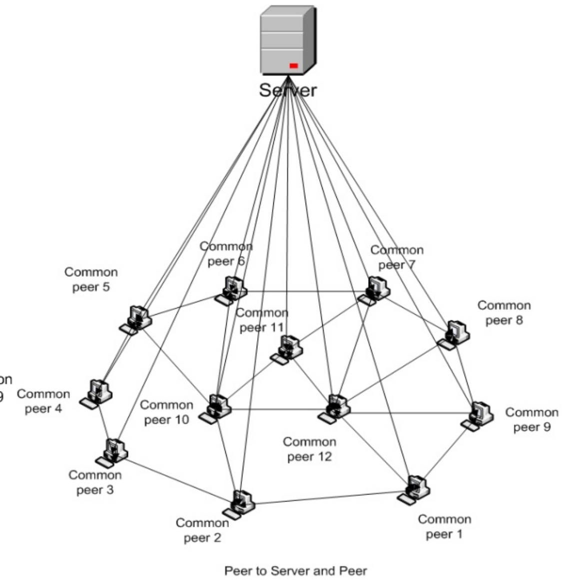

Figure 2: P2SP Network model.

2.3 Cloud Storage model

Cloud computing consists delivers applications and hardware to users as services via the Internet. With the rapid development of cloud computing, cloud services have emerged, such as SaaS (software as a service), PaaS (platform as a service) and IaaS (infrastructure as a service). Most prominently, cloud computing led to the concept of cloud storage which refers to a storage device accessed over the Internet via Web service application program interfaces (API). A traditional storage cloud system is a high performance, scalable, reliable, and available file distribution system (a typical architecture is shown in Figure 3). There are many existing cloud storage systems, for example, Amazon S3 (Amazon, 2006), the Google file system (Sanjay et al. 2003), HDFS (Borthakur, 2007), etc. These systems consist of master nodes and multiple chunk servers. Data is accessed by multiple clients and all files in the system are divided into fixed–size chunks. The master node maintains all file system metadata. At start-up or whenever a chunk server joins the cluster, the master node registers each chunk server with their chunks of information. Clients never read and write file-data through the master, but request from the master a chunk server to contact. A key problem in this model is that clients get data from the individual data nodes, but clients do not have any communication amongst themselves. The work in (Feng et al. 2010) analyzes several existing cloud storage platforms such as Simple Storage Service, Secure Data Connector, and Azure Storage Service, with their focus on the problem of security. The work identifies the problem of repudiation and proposes a non-repudiation protocol suitable for cloud computing environments by using third authorities certified (TAC) and secret key sharing (SKS) techniques.

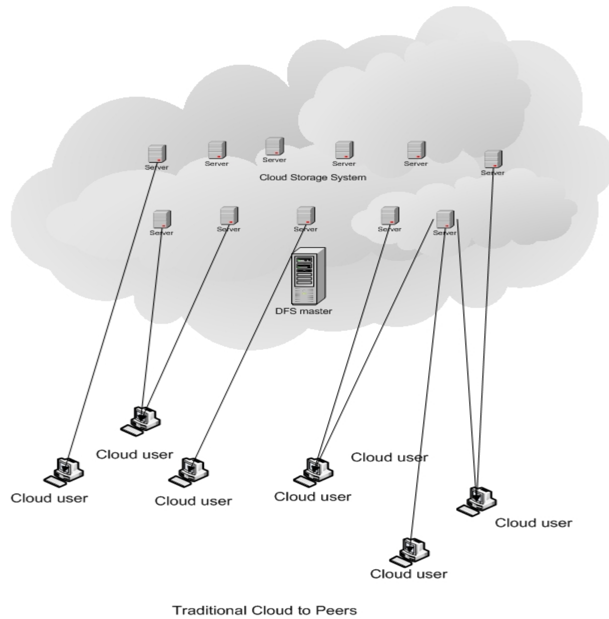

Figure 3: Traditional cloud network model.

3. A cloud model: P2CP

We propose a storage cloud model, which is the peer to cloud and peer (P2CP) model. This means that cloud users can download data from the storage cloud and exchange data with other peers at the same time, regardless of whether the other peers are cloud users or not. There are three data transmission tunnels in this storage model. The first is the cloud-user data transmission tunnel. The cloud-user data transmission tunnel is responsible for data transactions between the cloud storage system and the cloud users. The second is the clients' data transmission tunnel. The clients' data transmission tunnel is responsible for data transactions between individual cloud users. The third is the common data transmission tunnel. The common data transmission tunnel is responsible for data transactions between cloud users and non-cloud users. Figure 4 is an example to show how a P2CP cloud model works. In Figure 4, we can see that cloud user2 is downloading data from data node 1, which is in the cloud, and at the same time, cloud user2 is exchanging data with cloud user1, cloud user3, and common peers 2, 5, and 6. By exploiting multi data transmission tunnels, cloud users can achieve a high download speed. On the other hand, P2CP model avoids extremely high workload for cloud servers as the number of cloud users increases. When the resources are committed to other transmitting activities, non-cloud users may still get access to resources in the cloud which are not in common with the P2P networks.

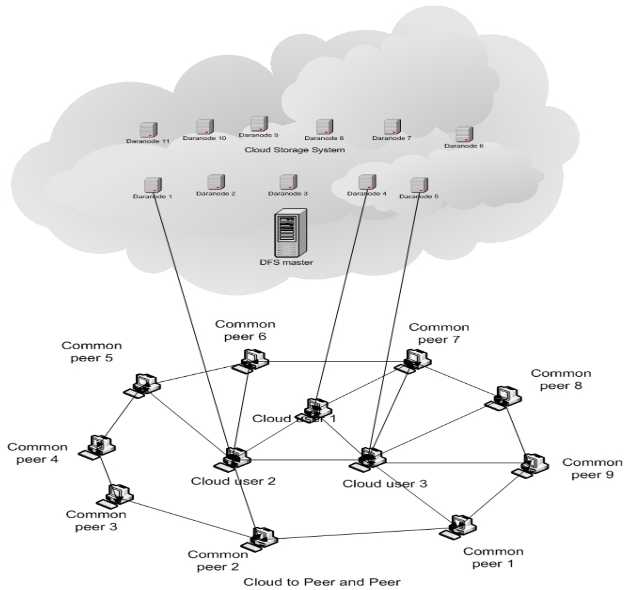

Figure 4: P2CP Network model.

In the pure P2P storage model, peers are divided into seeds, which are denoted by S, and leeches, which are denoted by L. Initially, seeds have the whole file, and leeches do not have any block of the file, but as time passes, leeches obtain blocks and exchange blocks with other peers. When the leeches get the whole file, they may leave the network, or stay in the network as seeds. In the P2SP network storage model, the difference is that it has a server group. Normally, in the Cloud storage model, there are three replicas of the file existing in different data nodes, and each data node keeps different amounts of blocks of the file. In the P2CP storage model, the storage cloud replaces the role of the server in the P2SP model. Compared to work mentioned in Section 2, our model addresses load balance issues via separating peers and cloud servers. Other existing models such as Groove (Ozzie, 2005), as known as comparable to Microsoft SharePoint (Chou, 2006), and Tahoe (O'Hearn and Warner 2008) tended to balance loads between peers and cloud serves in different ways. However, in our P2CP model, peers may communicate directly and flexibly between each other without tight dependence on servers, though some advanced features such as backing up, caching, security and versioning of data may still be elevated or mitigated to servers because peers' storage and computing capacities are supposedly inferior to those cloud servers.

4. Comparison and Evaluation

In this section, we evaluate our P2CP storage model against the three storage models described in Section 2: the pure P2P model, the cloud model and the P2SP model. For the network storage models, the two most important parameters for the performance are average downloading time and usability. In this part, we compare all these storage models in terms of these two parameters and evaluate our new P2CP model. We assume the following parameters:

- Upload bandwidth of each seed is U_s; the number of seeds is N_s.
- Upload bandwidth of each peer is U_p; the number of seed is N_p.

- Average upload bandwidth for each server is U_{SE}; the number of servers is N_{se}. The average number of threads for each server is N.
- Upload bandwidth of each node in the cloud is U_c; the number of data nodes is N_c.
- *F* is the size of the file.
- *T* is the average downloading time.
- *O* is usability.
- λ: The arrival rate of peers.
- μ: The departure rate of peers.

4.1 Comparison based on Poisson Process

The Poisson distribution is very useful for modelling purposes in many practical applications. It is empirically found to well approximate many circumstances arising in stochastic processes(Adan and Resing 2001). For our purpose, we assume that peers arrive and leave according to a Poisson process. The numbers of peers and seeds existing in the pure P2P network modeled on M/G/∞ queue. We can get the number of peers and seeds that exist in the pure P2P network:

$$N = (\lambda - \mu)t \tag{1}$$

If a peer takes T time to download a file with size F in the P2P network, we get:

$$\sum_{k=1}^{N_s} \int_0^T U_s dt = F \tag{2}$$

$$\sum_{k=1}^{N_s} [(\lambda - \mu)U_s] \frac{1}{2} T^2 = F \tag{3}$$

If it takes a peer time T to download a file with size F in the P2SP network, we get:

$$\sum_{k=1}^{N_s} \int_0^T U_s dt + T \sum_{k=1}^{N_{se}} U_{se} = F \tag{4}$$

$$\sum_{k=1}^{N_s} [(\lambda - \mu)U_s] \frac{1}{2} T^2 + T \sum_{k=1}^{N_{se}} U_{se} = F \tag{5}$$

If it takes a peer time T to download a file with size F in the P2CP network, we get:

$$\sum_{k=1}^{N_s} \int_0^T U_s dt + T \sum_{k=1}^{N_c} U_c = F \tag{6}$$

$$\sum_{k=1}^{N_s} [(\lambda - \mu)U_s] \frac{1}{2} T^2 + T \sum_{k=1}^{N_c} U_c = F \tag{7}$$

If it takes a peer time T to download a file with size F in the cloud system, we get:

$$N_c * U_c * T = F \tag{8}$$

According the normal cloud storage system configuration, we get:

$$N_c \geq 3N_{se} \tag{9}$$

We assume that:

$$A = \sum_{k=1}^{N_{se}} (\lambda - \mu)U \tag{10}$$

$$B = \sum_{k=1}^{N_{se}} U_{se} \tag{11}$$

$$C = \sum_{k=1}^{N_c} U_c = 3B = 3\sum_{k=1}^{N_{se}} U_{se} \tag{12}$$

According the (3), (5) and (10), (11) we get:

$$\frac{A}{2}T^2 - F = 0 \tag{13}$$

$$\frac{A}{2}T^2 + BT - F = 0 \tag{14}$$

According the (12) and (13) we get:

$$\frac{A}{2}T^2 + CT - F = \frac{A}{2}T^2 + 3BT - F = 0 \tag{15}$$

Then, according the equations (13), (14) and (15), we get:

$$T_c = \frac{2F}{6U_{se} * N_{se}} \tag{16}$$

$$T_{p2p} = \frac{2F}{\sqrt{2FA}} \tag{17}$$

$$T_{p2sp} = \frac{2F}{\sqrt{B^2 + 2FA} + B} \tag{18}$$

$$T_{p2cp} = \frac{2F}{\sqrt{C^2 + 2FA} + C} = \frac{2F}{\sqrt{9B^2 + 2FA} + 3B} \tag{19}$$

For our comparative purposes, we assume that the size of the file is 100,000 KB, the upload bandwidth of the peers and seeds are 20 KB/s, the upload bandwidth of a server is 100 KB/s, the arrival rate of peers is 2 peers/s and the departure rate is 1 peer/s. Figure 5 clearly shows that the alternative with minimal cost of download time is P2CP. The maximum download time is found with P2P. P2SP falls in the middle when there are not too many peers. The difference in download time is quite obvious. When more peers join the network, download time decreases. With the growth of upload bandwidth for the peers, we have another test. Assume that the size of a file is 100000 KB, the upload bandwidths for peers and seeds are 20 KB/s, 40 KB/s, or 60 KB/s, while the upload bandwidth for the server is 100 KB/s. The arrival rate of peers is 2 peers/s, and the leaving rate is 1 peer/s. Figure 6 shows that when there is increase of upload bandwidth for the peers, the download time inversely decreases. At the same time, differences in download time between P2P, P2SP, and P2CP are also reduced. Pure cloud storage model performance is not shown in Figure 6, because the result changes significantly. In some instances it outperforms the P2P and the P2SP models depending on the chunk distribution in the cloud storage system, but it never outperforms our P2CP storage model.

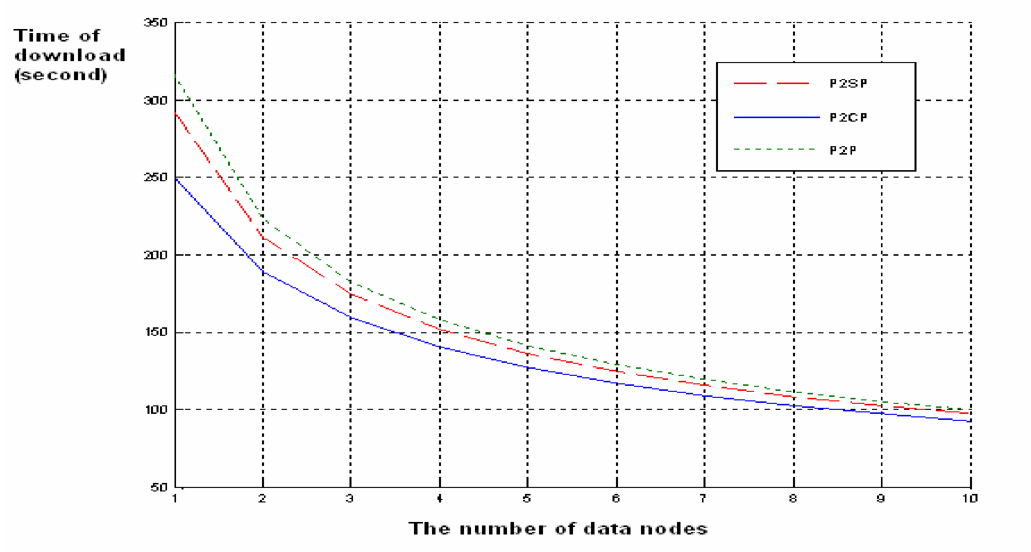

Figure 5: Time of download.

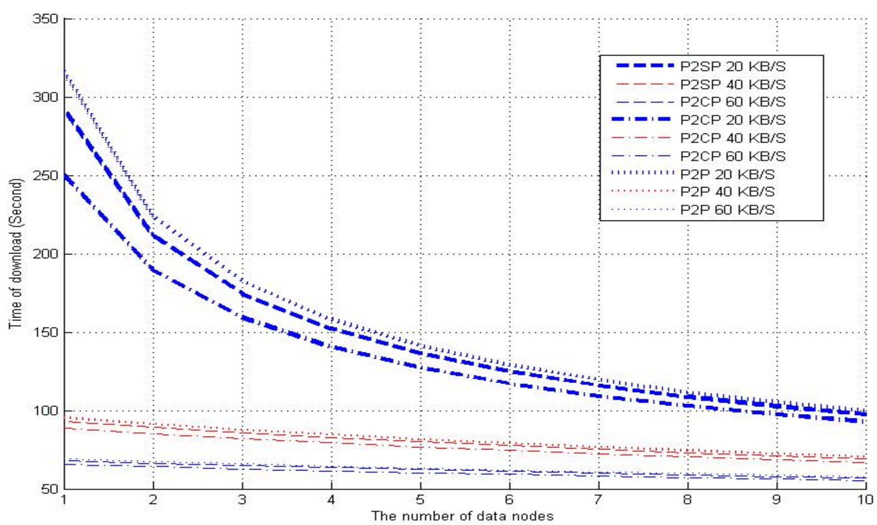

Figure 6: Comparing download time.

4.2 Comparison based on Little's Law

It is difficult to prove that the peers' and seeds' arrival and departure rates are accurate according to the Poisson process. Therefore we instead use Little's law to relate L (number of peers), W (sojourn time), and λ (average number of users) (Adan and Resing, 2001) as $L = \lambda W$ we can get:

$$N = (\lambda - \mu)T \quad (20)$$

According to (10) and (21), we get:

$$A = \sum_{k=1}^{N} \frac{N}{T} U \tag{21}$$

Then, according the equation, we get:

$$T_{p2sp} = \frac{2F}{NU + 2B} \tag{22}$$

$$T_{p2p} = \frac{2F}{NU} \tag{23}$$

$$T_{p2cp} = \frac{2F}{NU + 6B} \tag{24}$$

$$T_{c} = \frac{2F}{6B} \tag{25}$$

According the (22), (23) and (24), we get:

$$T_{p2p} > T_{p2sp} > T_{p2cp}$$

Thus, minimum download time is possible with P2CP, then P2SP, and lastly with P2P.

5. Discussion and Conclusion

For a storage service, availability and speed are high priority considerations. In the previous section, we proved that the speed of P2CP is superior. In this section, we compare and discuss the availability of P2CP in comparison with other models from the point of view of the whole network and of shared resources. According to (Sands and Tseng 2009), we know that common hardware failures are often clustered e.g. servers are expected to fail sometimes in a networked environment. In our comparative evaluation, we assume the failure rate of each peer is 1%, and the failure rate of each server is 0.1%. We assume that two peers constitute the smallest pure P2P network; the smallest P2SP network includes one server and one peer; the smallest cloud includes one master node and one data node; and the smallest P2CP network includes one smallest cloud and one peer.

From the point of view of whole network availability, based on the above, we observe the following: In the P2P network, even if only 1 peer existed in the P2P network, when the user connects to the peer, the P2P network can still be set up. Thus the maximum failure rate of the pure P2P network is 1%. To the failure rate of the P2SP network, failure for one machine will not lead to a whole breakdown of the P2SP network. If the server is shutdown, the network becomes a P2P network; if the peer is offline, the network becomes a client and server based. Only when both, the server and peer, break down at the same time, the whole network will shutdown. Thus the maximum failure rate of the P2SP network is 1%*0.1% =0.001%. To the failure rate of the cloud network, according the features of cloud, we know that no level of master or data nodes shutdown will lead to the whole cloud network being fully disabled; except until both master node and data node are broken at the same time. So, the maximum failure rate of the cloud network 1%*0.1% = 0.001%. P2CP network could run without peers, even the failures happened to master node or data node; until all peers is gone and both master node and all data nodes are broken, the whole P2CP network will shutdown. So, the failure rate of P2CP is

0.1%*1%*1%=0.00001%. Thus, in the worst network situation, the most stable network storage model is P2CP.

From the point of view of a particular shared resource, we know that the storage services follow the long tail law (Anderson, 2006). This means that the particular resource may be very popular at the beginning but demand eventually falls significantly for a long time. In the P2P storage model, initially the particular resource is frequently downloaded and uploaded in the network, so users can access the particular resource easily. However, when the particular resource is no longer popular, and the peers who hold the information for the particular resource leave, the P2P network is still there, but the resource is not available. Both the cloud storage model and the P2SP storage model solve this problem. They use a series of servers or a single server to record the particular resource to guarantee the availability, but with different transmission efficiency. The transmission efficiency of the P2SP storage model is improved, when the particular resource is popular and the transmission efficiency is high, but when the particular resource is unpopular, the transmission efficiency is low. The cloud storage model gets the opposite result. Only the P2CP storage model achieves the best result. Regardless of whether the particular resource is in fashion, the availability and speed are very good. From the evaluation results of Section 4.1, we can clearly see that in whatever the situation, the cost in time for P2CP is the lowest, and the usability is highest.

In summary, we have introduced a cloud storage system model to enhance data transmission performance and provide persistent availability, which has been named P2CP. The conclusion of our comparative studies presented in this paper, based on statistical modeling, P2CP not only enhances the utilization rate of bandwidth that exists in cloud storage systems, but also may be a solution of the problem of persistent availability in the P2P network model. We prove using a mathematical model that the utilization rate of bandwidth and the persistent availability of the P2CP model should be better than for the pure P2P model, the P2SP model, or the cloud model. In future work, we will pursue an empirical evaluation based on building the prototype P2CP storage system and test the performance.

References

Adan, I. and J. Resing (2001) *Queueing Theory*, pp. 111-113 Eindhoven, The Netherlands.

Amazon (2006) "Amazon S3," http://aws.amazon.com/s3/ (07/12, 2010).

Anderson, C. (2006) The Long Tail, in *Why the Future of Business Is Selling Less of More*, 2 edition, vol. 24, pp. 170–177: Hyperion.

Borthakur, D. (2007) "The Hadoop Distributed File System: Architecture and Design", *The Apache Software*

Foundation.

Chou, Y. (2006) "Get into the Groove: Solutions for Secure and Dynamic Collaboration," TechNet Magazine., http://technet.microsoft.com/en-us/magazine/2006.10.intothegroove.aspx (22.March, 2011).

Clarke., I. (2000) "Freenet," http://freenetproject.org/ (07/12, 2010).

Cohen, B. (2001) "BitTorrent," http://www.bittorrent.com/btusers/what-is-bittorrent (07/12, 2010).

Fang, L., L. Peng, Y. Jie, and L. Zhenming. (2009) "Contrastive Analysis of P2SP Network and P2P Network. Wireless Communications", *Networking and Mobile Computing, 2009. WiCom '09. 5th International Conference on, 2009*, pp. 1-5.

Feng, J., Y. Chen, W.-S. Ku, and P. Liu. (2010) "Analysis of Integrity Vulnerabilities and a Non-repudiation Protocol for Cloud Data Storage Platforms", *Parallel Processing Workshops (ICPPW), 2010 39th International Conference on, 2010*, pp. 251-258.

Hong, T., A. Gulbeden, Z. Jingyu, W. Strathearn et al. (2004) "A Self-Organizing Storage Cluster for Parallel Data-Intensive Applications". *Supercomputing, 2004. Proceedings of the ACM/IEEE SC2004 Conference, 2004*, pp. 52.

Kirk, P. (2003) "Gnutella," http://rfc-gnutella.sourceforge.net/).

Merkur (2002) "eMule," http://www.emule-project.net/home/perl/general.cgi?l=1 (07/12, 2010).

O'Hearn, Z. W. and B. Warner (2008) "Tahoe: the least-authority filesystem", *Proceedings of the 4th ACM international workshop on Storage security and survivability,* pp. 21-26. Alexandria, Virginia, USA: ACM.

Ozzie, R. (2005) "Microsoft, Groove Networks to Combine Forces to Create Anytime, Anywhere Collaboration," http://www.microsoft.com/presspass/features/2005/mar05/03-10GrooveQA.mspx (22 March, 2011).

Sands, A. and V. Tseng (2009) "An analysis of reported laptop failures from malfunctions and accidental damage", *SquareTrade.*

Sanjay, G., G. Howard, and L. Shun-Tak (2003) "The Google file system", in *Proceedings of the nineteenth ACM symposium on Operating systems principles*, pp. 29-43. Bolton Landing, NY, USA: ACM.

Stein, C. A., M. J. Tucker, and M. I. Seltzer. (2002) "Building a reliable mutable file system on peer-to-peer storage. Reliable Distributed Systems", *2002. Proceedings. 21st IEEE Symposium on, 2002*, pp. 324-329.

Sun, Y., F. Liu, B. Li, and B. Li. (2009) "FS2You: Peer-Assisted Semi-Persistent Online Storage at a Large Scale", in *INFOCOM 2009, IEEE*, pp. 8. Rio de Janeiro

Ye, W., A. I. Khan, and E. A. Kendall. (2003) "Distributed network file storage for a serverless (P2P) network". *Networks, 2003. ICON2003. The 11th IEEE International Conference on, 2003*, pp. 343-347.