Account-Based Cloud Storage Service Virtualization

Yen-Hung Kuo

Abstract—**Public cloud storage services offer convenient file sharing and backup services. However, many enterprises have been wary to employ these services and some even prohibit their employees from using such services for business data due to concerns over privacy, security, control of their assets, and vendor lock-in issues. To address these issues, this work proposes a new service utilization model (an architecture), which virtualizes public storage services among multiple user accounts to provide each user with a more secure, better HA (High Availability), and more fixable storage service.**

Index Terms—**Cloud storage, storage service virtualization, distributed wide-area storage systems.**

I. Introduction

In the last few years, public cloud storage services become the most popular cloud application. Typically usages of such services can be categorized into two types. Firstly services as Dropbox, Google Drive, and others offer convenient file sharing both to single users across multiple devices and to groups of users wishing to share data for collaboration. Secondly services like Amazon S3 and Windows Azure Storage as a remote backup storage or a distributed object store, which can incorporates with business logics to provide cloud based software (Software-as-a-Service). However, to services like Dropbox or Google Drive, many enterprises have been wary to employ these services and some even prohibit their employees from using such services for business data due to concerns over privacy, security, and control of their organizations' assets. In addition, recent research [1] also mentioned the "vendor lock-in" issue of using services like Amazon S3 or Windows Azure Storage.

To address these issues, recently many researches [2]-[6] tried to distribute objects into more than one public cloud storage services, which provide users with a better data availability (compared with single service) and also mitigates the vendor lock-in issue. In addition to aforementioned benefits, these works also have their own unique strengths and contributions. Abu-Libdeh et al. [2] introduced a Redundant Array of Cloud Storage (RACS), which uses a central proxy to stripe and distribute object into different storage services. RACS solves the vendor lock-in issue and also improves the data availability by adding redundancy among services. Agarwala et al. [3] presented a storage service called iCostale, in which an intelligent cloud storage service selection and placement engine was presented to distribute objects to cloud storages while reducing overall cost. Bermbach et al. [4] implemented a federated hashtable named MetaStorage, which replicates objects to cloud storages. A way of control consistency-latency trade-offs was also introduced in this work. Bowers et al. [5] developed a distributed cryptographic system called High-Availability and Integrity Layer (HAIL), which stores encrypted data to diverse cloud storage services. HAIL mitigates the data security issue and also improves the data availability. Hill and Humphrey [6] proposed an abstraction layer, called Cloud Storage Abstraction Layer (CSAL), which provides Blob, Table, and Queue abstractions across multiple providers and presents applications. The importance of providing applications/programs with abstracted common REST APIs is also mentioned in the study.

Previous researches dealt with vendor lock-in and security issue, they also generate two new cost problems. One is that adopting multiple public storage services requires a higher cost. The other is that a cloud storage service with better SLA (ex. 99.9% and 99.999% availability and higher bandwidth) is usually expensive, and selection of a proper service for an object by according to its purpose becomes another important issue. To reduce the cost, this work proposes a general architecture to virtualize multiple cloud storage services by sharing storage services among multiple users' account basis. Two archetypes of leveraging cloud storage services are also proposed in this paper. This study not only resolves the aforementioned issues but also provides developers with a clear guideline to plan/design their (hybrid) cloud storage services.

The rest of this paper is organized as following. A different level of storage virtualization is introduced in Section 2. Section 3 depicts a general architecture for storage service virtualization. Along with the framework, two archetypes of virtualization are proposed. Finally, conclusion remarks are shown in the last section.

II. Levels of Storage Virtualization

Generally speaking, different levels of storage virtualization are results of pooling of lower-level storages. As shown in Fig. 1, the SAN is a result of pooling of Hard Disks, and multiple SANs can support one Cluster/Distributed File System, and so on. An important characteristic among storage virtualization levels is that the higher level of storage virtualization has better scalability and usage flexibility, but also along with higher access latency. The other characteristic is that any object is able to migrate to other storage unit within the same virtualization level. The design purpose of each level of storage virtualization is to complete a certain lack from the lower-level storage.

Manuscript received April 30, 2012; revised June 10, 2012. This study is conducted under the "Cloud computing systems and software development projects" of the Institute for Information Industry which is subsidized by the Ministry of Economy Affairs of the Republic of China.

Yen-Hung Kuo is with the Cloud System Software Institute, Institute for Information Industry, Taipei, Taiwan (e-mail: keh@iii.org.tw).

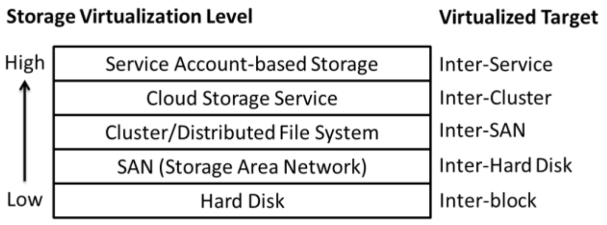

Fig. 1. Levels of storage virtualization.

Following introduces each level of storage virtualization.

1) **Hard Disk** [7]**:** In a Hard Disk, there are organized basic storage units called (logic) blocks which can be used to store data, and for different purposes the blocks could by organized by different fashions.
2) **SAN (Storage Area Network)** [8]**:** SAN is a shared storage network primarily uses hard disk drives to provide OS with block level virtual data storage (Logical Unit Number, LUN). SAN improves the storage capacity and the utilization since it pools hard disks as multiple LUNs for multiple systems. Note that an LUN can only be assigned to a system.
3) **Cluster/Distributed File System** [9]**:** A Cluster/Distributed File System usually consists of either single or multiple metadata node(s) and multiple data nodes to provide multiple systems with mountable network storage protocols by pooling of block-level storages. Differing from the SAN, a single file in a Cluster/Distributed File System is able to be accessed by multiple systems simultaneously, and it is able to add new underlying block-level storages to increase its storage capacity.
4) **Cloud Storage Service** [10], [11]**:** Nowadays, most cloud storage services are built on top of cluster/distributed file systems plus management functions. Compared with cluster/distributed file systems, it more focuses on high availability (HA), scalability, service level agreement (SLA), and disaster recovery (DR). To access a cloud storage service, a storage service access key is required, and an access key usually associated with a (paid) user account.
5) **Service Account-based Storage:** In this study, it bases on previous proxy-based researches [2]-[6] to propose a new Service Account-based Storage. It virtualizes multiple Cloud Storage Services and offers a platform, in which a group of trusted users is able to securely leverage each other's storage services. This approach resolves the vendor lock-in issue, data security issue, and the service cost issue. The most important contribution is that it provides a way to share/exchange cloud storage services by user account basis instead of just simply sharing/exchanging files among services. Details of the Service Account-based Storage are going to be shown in next section.

III. Account-Based Cloud Storage Service Virtualization

A. General Architecture

A general architecture for account-based cloud storage service virtualization is shown in Fig. 2, in which the key components in the proxy are introduced below.

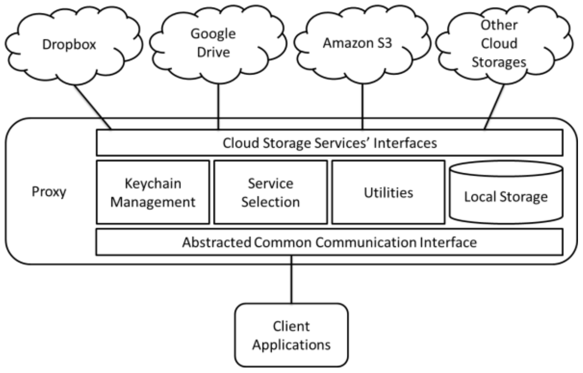

Fig. 2. A general architecture for account-based cloud storage service virtualization.

1) **Data Communication Interfaces:** There are two types of data communication interfaces in the Proxy. One is the Cloud Storage Services' Interfaces, which provide Proxy with the ability to communication with diverse cloud storage services, and the other is the Abstracted Common Communication Interface, which offers Client Applications a single interface to communicate with Proxy and to indirectly interact with cloud storage services. Most Cloud Storage Services' Interfaces are different and they are formed in either Web Services or REST (Representational state transfer) [12] APIs which can be found in each storage service's developer forum and easily integrated within Proxy.
2) **Keychain Management:** The Keychain Management is responsible for managing users' keys which can be used to access users' cloud storages. A user can have multiple cloud storage services' keys in the proxy, and each distributed object is associated with a quartet (storage service endpoint, storage service access key, object ID on storage service, object owner on proxy), which can be used to retrieve the object from a specific cloud storage service.
3) **Service Selection:** The Service Selection can decide an object's distribution destination (cloud storage service) by according to given criteria, policies and rules. Implementation of the Service Selection is subject to the Multiple Criteria Decision Making (MCDM) [13] research. Common decision factors are organized in Table I.
4) **Utilities:** To enhance data security and this services' flexibility, three kinds of utilities can be adopted to this architecture. Firstly, a file striping/merging tool [2] can stripe a file into several pieces and merge striped pieces to the original file. Striped pieces can then be distributed and replicated to different cloud storage services. It enhances data security and availability since no single cloud storage service has all pieces and single piece can be replicated to multiple storage services. Secondly, a data encryption/decryption tool [5] can further protect (striped) objects before sending to cloud storage services. The third useful utility is the diff tool [14], which can generate differential parts of two files (delta information) by comparing them. To keep file version history, it can simply send the delta information to public cloud storage

services. Moreover, the size of delta information is much smaller than the original file, therefore stores it to cloud storage services is secure and economic.

5) **Local Storage:** Depends on purposes of the proxy, the Local Storage can provide system with two different functionalities: the data transmission and processing local buffer and the local storage of hybrid cloud storage service. In case of the local buffer usage, object transferring and Utilities' tasks can be formed in queues, where object GET and PUT requests are temporally stored, and applications do not need to wait until completed, therefore a better response time can be obtained. With regarding to the hybrid cloud storage case, the Local Storage becomes an alternative object distribution destination, which can be used to store mission critical data and sensitive business files.

TABLE I: COMMON DECISION FACTORS FOR MCDM

Decision Factor	Values
Max. file size	• No limit • *X* GB (*X* depends on service policy)
File extension	• No limit • A list of allowed file extensions
Available storage	*Y* MB (*Y* depends on storage service usage)
Data availability	• Very good (> 99.9%) • Good (~99.9%) • Normal (< 99.9%)
Network bandwidth (Data available region)	• Good (in the same region) • Normal (not in the same region) Speed also depends on the connection between the proxy and storage services.
Pricing	**Storage pricing:** • Expensive (>= $0.1 per GB/month) • Inexpensive (< $0.1 per GB/month) / Free **Request pricing:** • Requests are charged • Free **Data transfer pricing:** • Data transfer is charged • Free
Data protection	• Supports extra data security and protection • Not supported
Auto scale up	• Supports storage auto scale up • Not supported

B. Archetypes of Cloud Storage Service Leveraging

Based on the proposed architecture, two archetypes of cloud storage service leveraging among users' accounts are then going to be shown in this section.

In Fig. 3 and Fig. 4, two archetypes of cloud storage service leveraging are drawn: Cloud Storage Borrowing and Exchanging. The end user 1 and 2 are two trust users who can access the proxy and share their own cloud storage service access keys to each other (cloud storage service 1 belongs to end user 1 and the other storage service belongs to end user 2). Each cloud storage service has its own usage rules (max. file size, allowed file extension, and so on), characteristic (SLA, network bandwidth, and pricing model), and metadata (service endpoint, storage quota, and available storage). All these information is stored in proxy for the Service Selection to do judgment. Note that the cloud storage services are not necessarily to be provided by different providers. Instead two accounts from a cloud storage service are treated as two different services. Additionally, aforementioned Utilities can also apply to the two archetypes to obtain a more flexible storage usage model. Advantages of the two archetypes are introduced in following.

1) **Cloud Storage Borrowing:** The Cloud Storage Borrowing treats all cloud storage services (by account basis) as a large storage pool. It achieves a sort of storage service thin provision [11], which allows a cloud storage provider over commits storage quota to its users. In other words, if a user has 10 GB cloud storage and only 2 GB is using, the cloud storage service provider can allocate the available 8GB to other meaningful usage instead of allocating total 10 GB to that user. When the idea is implemented on multiple cloud storage services, in addition to thin provision, it also resolves the Economic Failure [2] of cloud storage services. Even a piece of data cannot be retrieved from one cloud storage service (account), it still can be obtained from other available destinations.

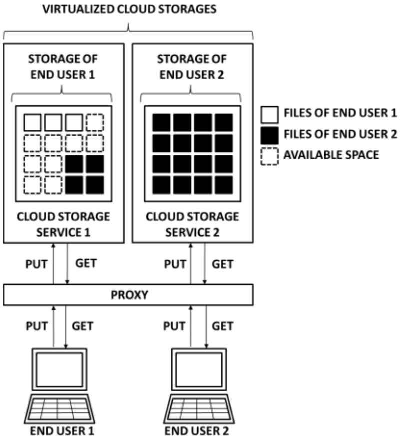

Fig. 3. Cloud storage service leveraging: Cloud storage borrowing.

2) **Cloud Storage Exchanging:** Characteristics of cloud storage services are diverse. Sensitive data should be put to cloud storage services, which have stronger security protection. Storage services with better availability are more suitable for mission critical data. Large files such as backup images require high speed network bandwidth. Product flyers can be served by free hosts. Generally speaking, cloud storage services with more advanced features are usually more expensive. Therefore it is difficult to let a single user have multiple diverse storage service accounts. To address this issue, the Cloud Storage Exchanging allows a user to use a portion of cloud storage service to exchange with other users' services. All users in a exchanging relationship will have a virtualized cloud storage service with better data availability and other advantages accordingly.

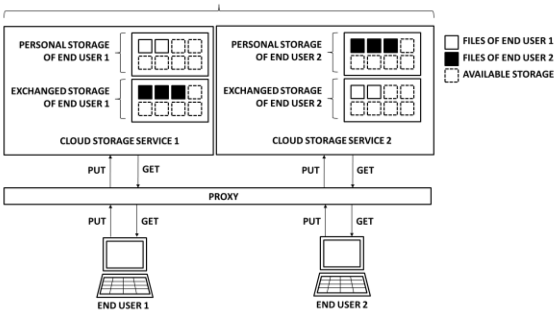

Fig. 4. Cloud storage service leveraging: Cloud storage exchanging.

IV. Conclusions

In this paper, it presented a general architecture for pooling of cloud storage services by users' accounts basis. From taxonomy, it belongs to storage service virtualization, which allows one user's data to migrate to/from other users' cloud storage services. Additionally, the proposed architecture also resolves the cloud storage vendor lock-in, the data security, and the cost issues. Finally, along with the architecture, two archetypes of leveraging cloud storage services and their advantages were introduced. This study gave a direction of leveraging cloud services belongs to other users, and wishes it could inspire follow-up researches in this area.

References

[1] R. Ananthanarayanan, K. Gupta, P. Pandey, H. Pucha, P. Sarkar, M. Shah, and R. Tewari, "Cloud analytics: Do we really need to reinvent the storage stack?," in *Proc. of the 2009 Conf. on Hot Topics in Cloud Computing (HotCloud'09)*, San Diego, CA, 2009.

[2] H. Abu-Libdeh, L. Princehouse, and H. Weatherspoon, "RACS: A case for cloud storage diversity," in *Proc. of the 1st ACM Symposium on Cloud Computing (SoCC'10)*, Indianapolis, IN, 2010, pp. 229-240.

[3] S. Agarwala, D. Jadav, and L. A. Bathen, "iCostale: Adaptive cost optimization for storage clouds," in *Proc. of the 2011 IEEE 4th International Conf. on Cloud Computing (CLOUD'11)*, Washington DC, MD, 2011, pp. 436-443.

[4] D. Bermbach, M. Klems, S. Tai, and M. Menzel, "MetaStorage: A federated cloud storage system to manage consistency-latency tradeoff," in *Proc. of the 2011 IEEE 4th International Conf. on Cloud Computing (CLOUD'11)*, Washington DC, MD, 2011, pp. 452-459.

[5] K. D. Bowers, A. Juels, and A. Oprea, "HAIL: A high-availability and integrity layer for cloud storage," in *Proc. of the 16th ACM Conf. on Computer and Communications Security (CCS'09)*, Chicago, IL, 2009, pp. 187-198.

[6] Z. Hill and M. Humphrey, "CSAL: A cloud storage abstraction layer to enable portable cloud applications," in *Proc. of 2010 IEEE Second International Conf. on Cloud Computing Technology and Science (CloudCom)*, Indianapolis, IN, 2010, pp. 504-511.

[7] Wikipedia. Hard disk drive. Retrieved on Apr. 2012. [Online]. Available: http://en.wikipedia.org/wiki/Hard_disk_drive.

[8] Wikipedia. Storage area network. Retrieved on Apr. 2012. [Online]. Available: http://en.wikipedia.org/wiki/Storage_area_network.

[9] K. Kamisaka, "A survey of distributed storage and parallel I/O technique for security incident analysis," *Journal of the National Institute of Information and Communication Technology*, vol. 55, no. 2/3, pp. 73-78, June/September 2008.

[10] H. Dewan and R.C. Hansdah, "A survey of cloud facilities," in *Proc. of the 2011 IEEE World Congress on Services (SERVICES'11)*, Washington DC, MD, 2011, pp. 224-231.

[11] W. Zeng, Y. Zhao, K. Ou, and W. Song, "Research on cloud storage architecture and key technologies," in *Proc. of the 2nd International Conference on Interaction Sciences: Information Technology, Culture and Human (ICIS'09)*, Seoul, Korea, 2009, pp. 1044-1048.

[12] Wikipedia. Representational state transfer. Retrieved on Apr. 2012. [Online]. Available: http://en.wikipedia.org/wiki/Representational_state_transfer.

[13] T. J. Stewart, "A critical survey on the status of multiple criteria decision making theory and practice," *OMEGA International Journal of Management Science*, vol. 20, no. 5/6, November 1992.

[14] Wikipedia. Diff. Retrieved on Apr. 2012. [Online]. Available: http://en.wikipedia.org/wiki/Diff.

Yen-Hung Kuo was born and raised in Kaohsiung, Taiwan, in 1981. He received his B.S. degree in Information Management from National Kaohsiung First University of Science and Technology (NKFUST), Kaohsiung, Taiwan, in 2004. He earned his Ph. D. degree in the Department of Engineering Science at the National Cheng Kung University (NCKU), Tainan, Taiwan, in 2008. He was a Senior Engineer of Cloud Service Technology Center, Institute for Information Industry, Taiwan. Presently he is a Section Manager of Cloud System Software Institute, Institute for Information Industry, Taiwan. He is serving as a Managing Editor at International Journal on Smart Sensing and Intelligent Systems. He is as an Editor of ISRN Education. Hi is an honor member of the Phi Tau Phi Scholastic Honor Society of R.O.C. His research interests include cloud computing, data mining, distributed data management, information retrieval, intelligent systems, distance education, and software engineering.