

A Novel Storage Virtualization Scheme for Network Storage Systems

Jui-Pin Yang

Department of Information Technology and Communication Shih Chien University, Kaohsiung City, Taiwan
juipin.yang@g2.usc.edu.tw

Abstract—The network storage systems are generally composed of clients, storage servers and metadata servers. In this paper, we proposed a novel storage virtualization (NSV) scheme which is capable of alleviating the heavy load of metadata server, guaranteeing the storage quality of service and dynamically adapting storage resources. The metadata server automatically constructs a dedicated storage cluster according to various requirements of storage quality of service. The storage cluster may consist of one to many storage servers which includes one master storage server and zero to many slave storage servers. In other words, a network storage system consists of at least one storage cluster. The requests of each client are forwarded to corresponding master storage server within a specific storage cluster. In addition, the master storage server determines the best storage server which handles the requests based on the conditions of storage servers. Next, the requests will be redirected to the selected storage server. Finally, the responses are directly transmitted to the client.

Keywords: Storage virtualization, network storage system, metadata, quality of service, storage cluster.

I. INTRODUCTION

In traditional storage architectures, the storage resources are individually managed. Therefore, they are difficult to extend and control by using a common fashion. A virtual architecture is relatively useful to improve the drawbacks of traditional storage architectures because it greatly decreases management complexity in downtime. In general, virtual resources are easier to manage and control while comparing with physical resources. The reason is that less hardware-specific details are involved. In addition, the utilization, scalability and flexibility of storage systems could be enhanced because of virtualization of storage resources. The virtualization techniques are beneficial not only to

realize quality of service in the storage systems but also to use a simple and uniform management way because the physical details are hidden.

In an agile data center, applications have less relationship with the physical resources and hence they can share their resources more easily. In addition, a nondestructive manner that can migrate applications among different resources is attractive. Moreover, agile cloud computing infrastructures enhance sharing and management for data centers. The import technique that corresponds to the demands is virtualization [1]. In general, data centers are equipped with enormous storage capacities in order to cope with a great deal of user's requirements. Time-sensitive

applications need certain access performance related to the storage systems such as on-line transactions, online gaming and so on. Otherwise, they fail. It is too complicated to guarantee quality of service because I/O workloads are extremely various depending on device behaviors. To ensure independent but competing workloads is still quite difficult [2]. Storage virtualization is a pool of physical storage from several network storage devices into and hence a single storage device is shown that is managed from a central console virtualization [3]. The general storage virtualization is built in a Storage Area Network (SAN). With storage virtualization, administrators can simplify the tasks inclusive of backup, archiving and recovery. In addition, it is a useful way to hide the actual complexity of the SAN [4].

Virtualization is generally used to construct virtual mapping resources such as servers, storage devices and operating system. For instance, virtualization divides a hard disk drive into several partitions. The reason is that one single hard disk drive is divided into partitions and hence we can efficiently and simply utilize partitions [5]-[7]. In this paper, we concentrate on the most important part of virtualizations, that is, storage virtualization. The novel storage virtualization scheme uses block virtualization for network storage systems. Internet Small Computer System Interface (iSCSI) is capable of building Storage Area Network (SAN). The storage virtualization in the SAN provides a layer of abstraction between the host that performs the IO operations and the storage devices that provides storage resources [8]. Besides, block virtualization is an efficient virtualization technique that combines available storage devices together which known as storage pools. According, one single individual storage device is demonstrated to the host. The

main functionality is the address re-mapping from physical addresses to logical addresses that redirects IO operations to corresponding physical devices. In addition, block virtualization provides non-disruptive data mobility and data migration. In a word, we develop the methods and system of a novel storage virtualization scheme which is suitable for network storage systems based on network and block virtualization.

II. STORAGE VIRTUALIZATION SCHEME

Figure 1 is the system architecture of network storage system which consists of three parts; a client, five storage servers and a metadata server. The client issues various file access operations, for example; read, write, create or lookup. A storage server represents a storage device with specific storage capabilities (such as disk array functions, such as; RAID 0, RAID 1 or RAID 5, etc.), storage capacity, memory size and CPU computing power. The metadata server is responsible for the storage resource allocation and planning of the entire network storage system, by obtaining the status of each storage server, for example; remaining storage capacity, available network bandwidth, system load and computing power. Also, it is responsible for network storage management that determines the number of storage servers in a storage cluster. In particular, the metadata server is only responsible for the overall storage resource allocation and management of the network storage system. The master storage server is responsible for the storage resource allocation and dynamic adjustment within a storage cluster. The hierarchical management not only effectively decreases the load of the metadata server but also dynamically adjusts resources in the storage cluster in time that ensures the storage quality for each client.

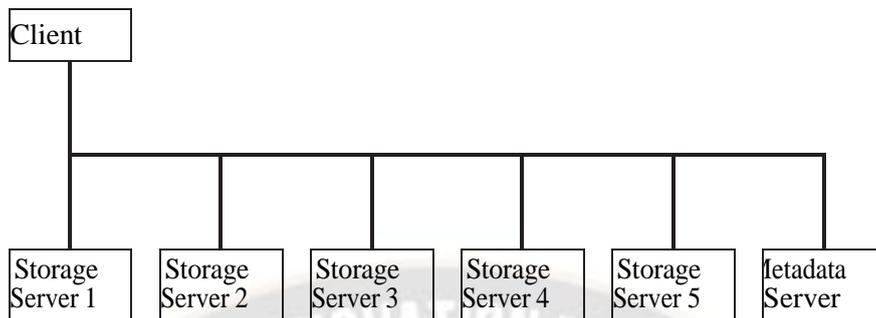


Figure 1. System architecture of a network storage system

A flowchart of the create request is depicted in Fig. 2. In the network storage system, the metadata server will construct a corresponding storage cluster according to the client's storage service quality requirements. In this example, the storage cluster is composed of a master storage server and two slave storage server. The storage server 1 is the master storage server. In addition, the storage server 2 and storage server 3 they both belong to slave server. Next, the metadata server will send the configuration of the storage cluster to the related storage servers referring to A01, A02 and A03. When the metadata server completes the configuration settings of the storage cluster, the client will obtain the network information of the master storage server from the metadata server, for example; IP address, transmission protocol (TCP or UDP), communication port and so on. Next, the client establishes a network connection and then sends a create request to the master

storage server referring to A04. When the master storage server receives the create request, it will determine which storage server in the storage cluster is the best one for processing the request. The conditions are evaluated such as the storage quality of storage servers, the importance of data and the number of serving as master or slave storage servers. Once the responsible storage server was determined, the master storage server will forward the create request to the slave storage server referring to A05. Next, the slave storage server will execute the create request, and then send a response message to the client referring to A06 and also update the access statistics to local repository such as; file access time, file hotspot, access statistics. Finally, the slave storage server updates the relevant information (directory name, file name) to the master storage server referring to A07.

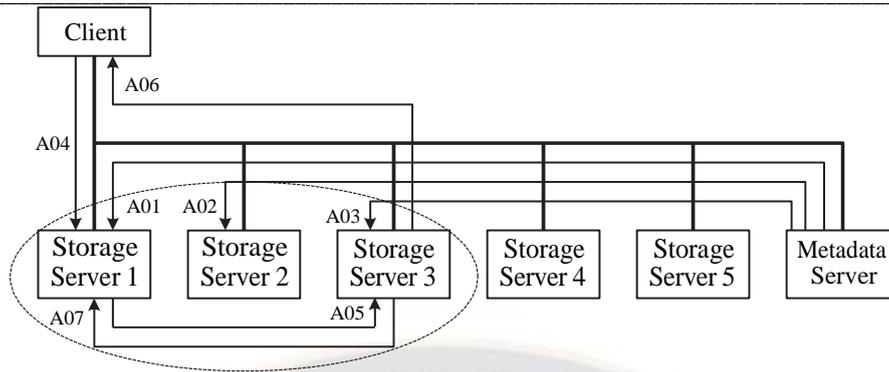


Figure 2. A flowchart of create request

A flowchart of the lookup request is depicted in Fig. 3. The client directly sends a lookup request to the master storage server via the network connection referring to A01 and then the master storage server will sequentially compare the lookup request with the cache and the directory name and file name in the remote repository. Next, the master storage server finds out the slave storage server which is responsible for dealing with the lookup request. Finally, it forwards the lookup request to the slave storage server for subsequent

processing referring to A02. In this example, a slave storage server will receive the lookup request forwarded by the master storage server and then it further queries the file with related attributes of the directory. Finally, the access statistics of the directory and file will be updated and stored in the local repository for assessment of storage service quality. Finally, the results of lookup request will be forwarded to the client referring to A03.

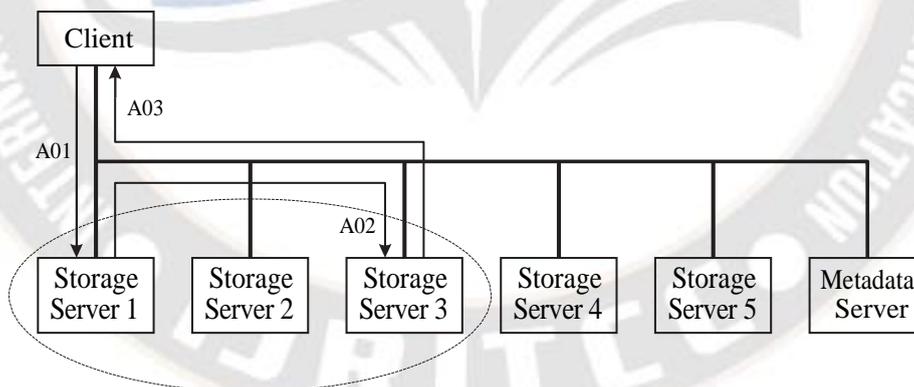


Figure 3. A flowchart of lookup request

A flowchart of the read/write request is depicted in Fig. 4. Repeatedly, the client directly sends a read/write request to the master storage server referring to A01. The master storage server will sequentially compare read/write request with the cache, remote repository and file system. If the master storage server could not confirm the responsible storage server because the cache was not updated in time, the remote repository data was incomplete or the file wasn't found in the file system. Therefore, the master storage server will forward the read/write request to all slave storage

servers in the cluster referring to A02 and A03. When the slave storage servers receive a read/write request, they just need to compare with their own caches and file systems without querying the remote repository. Once the file was found, the file system executes read and write operations and then returns a response message to the client referring to A04. Finally, the corresponding slave storage server will update the local repository and also return a message to the master storage server to update the related storage information of the file in the remote repository referring to A05.

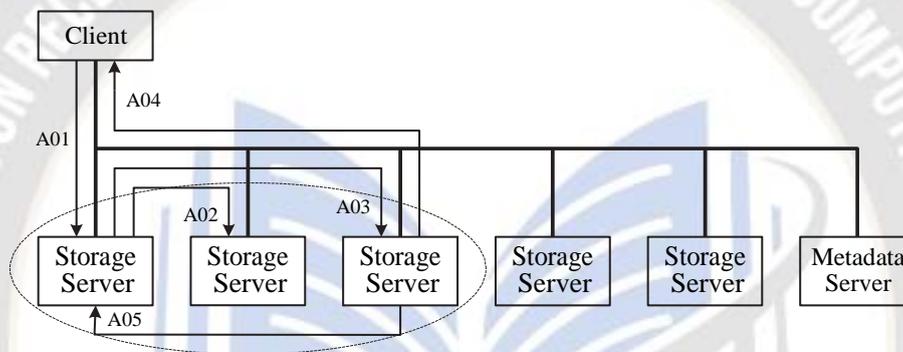


Figure 4. A flowchart of read/write request

III. CONCLUSION

The paper proposes an efficient storage virtualization scheme, namely NSV. The NSV can effectively decrease the load of metadata server with hierarchal management. The NSV not only ensures the storage quality of each client but also is capable of dynamical adjustment on storage resources. The metadata server is responsible for the overall management of storage resources within the storage cluster and the master storage server is responsible for detailed management of storage servers. The master storage server automatically adjusts the data based on status of storage servers or completes the storage capacity expansion that ensures storage quality of each client. Due to hieratical management, the NSV avoids the metadata server becoming a performance bottleneck. As a result, the entire

storage resources can be efficiently managed and dynamically configured so that the efficiency of storage resource utilization can be greatly improved.

References

1. Aameek S, Madhukar K & Dushmanta M, Server-Storage Virtualization: Integration and Load Balancing in Data Centers, *In Proc IEEE Int Conf on SuperComp*, (2008)53:1-53:12.
2. Adrian De Luca and Mandar Bhide, "Storage Virtualization For Dummies, Hitachi Data Systems Edition", Wiley Publishing, Inc. 111 River Street Hoboken, NJ 07030-5774
3. A New Method of SAN Storage Virtualization, *Raidtec Cooperation White Paper*, (2010) 1-5.

4. Storage Virtualization, SNIA Technical Tutorial (2013) 1-88.
5. Jun-wei Ge, Yong-long Deng, Yi-qiu Fang, "Research on Storage Virtualization Structure in Cloud Storage Environment", Multimedia Technology (ICMT), International Conference, IEEE, (2010) 1-4.
6. Sandeep Kumar and Syam Kumar P, "Secure and efficient design and implementation of out-of-band storage virtualization", IEEE, (2015) 1021-1025.
7. S. D. Joshi, Naveenkumar Jayakumar, Farid Zeimfar, "Workload Characteristics Impacts on File System Benchmarking", Int. J. Adv. Res. Computer Sci. Software Eng., 4(2) (2014) 39-44.
8. R. Salunkhe, A. D. Kadam, N. Jayakumar, and D. Thakore, "In Search of a Scalable File System State-of-the-art File Systems Review and Map view of new Scalable File system.," International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT) (2016) 1-8.
9. Jui-Pin Yang* and Ying-Chin Chen, A Novel Information Lifecycle Management Scheme for a Web-Based Database System, Journal of Scientific & Industrial Research, 76(2) (2017) 85-89.

