Computing and Informatics, Vol. 21, 2002, 547–561

THE ARCHITECTURE AND PERFORMANCE EVALUATION OF ISCSI-BASED UNITED STORAGE NETWORK MERGING NAS AND SAN

Xianglin Fu, Changsheng Xie, Ruifang Liu, Shenggang Pan

National Storage System Laboratory
Computer School of Huazhong University of Science and Technology
Wuhan, Hubei, P. R. China 430074
e-mail: `fu_xianglin@hotmail.com`

Manuscript received 5 April 2002; revised 10 January 2003

Communicated by Ivan Plander

Abstract. With the ever increasing volume of data in networks, the traditional storage architecture is greatly challenged; more and more people pay attention to network storage. Currently, the main technology of network storage is represented by NAS (Network Attached Storage) and SAN (Storage Area Network). They are different, but mutually complementary and used under different circumstances; however, both NAS and SAN may be needed in the same company. To reduce the TOC (total of cost), for easier implementation, etc., people hope to merge the two technologies. Additionally, the main internetworking technology of SAN is the Fibre Channel; however, the major obstacles are in its poor interoperability, lack of trained staff, high implementation costs, etc. To solve the above-mentioned issues, this paper creatively introduces a novel storage architecture called USN (United Storage Networks), which uses the iSCSI to build the storage network, and merges the NAS and SAN techniques supplying the virtues and overcoming the drawbacks of both, and provides both file I/O and block I/O service simultaneously.

Keywords: iSCSI, Network Attached Channel (NAC), United Storage Networks (USN), NAS (Network Attached Storage), SAN (Storage Area Network)

1 INTRODUCTION OF TRADITIONAL NAS AND SAN

The Storage Networking Industry Association (SNIA) Technical Dictionary defines NAS as follows: "A term used to refer to storage elements that connect to a network and provide file access services to computer systems". In common usage, a NAS system is a special-purpose device that is designed to serve files to clients over a LAN. The are many benefits in the architecture: heterogeneous file sharing; internal resource pooling; exploits the existing infrastructure; simple to implement; connectivity; improved manageability; reduced total cost of ownership, etc. However, there are still many drawbacks in practice:

1. The access speed: the client sends file I/O request using the NFS or CIFS; at the client, the file I/O request must be encapsulated by the TCP/IP protocol stack first, then the encapsulated request travels across the network and arrives at the NAS devices; at the destination, the encapsulated request must be stripped. After the storage subsystem performs the physical operations, similar encapsulation and stripping process must be performed again; thus, software overhead increases severely. Consequently, NAS is not suitable for situations where high access speed is needed.
2. Backup of data: backup consumes the bandwidth and other network resources of the LAN; even worse, it decreases the overall LAN performance greatly.
3. Consolidation and central management of the storage resource: disks of the same NAS device can be consolidated, but the disks of different NAS devices cannot be consolidated into a single storage pool.

SNIA defines SAN as "a network whose primary purpose is the transfer of data between computer systems and storage elements and among storage elements". Currently, the primary building technology of SAN is the Fibre Channel (FC-SAN). The architecture introduces a number of opportunities: high performance, high availability, high scalability, improved manageability and storage sharing, etc. Unfortunately, there are still several drawbacks which obstruct the development of FC-SAN. First, all vendors apply the Fiber Channel protocol, but the implementation is different with each vendor, and thus the compatibility of the storage devices from different vendors is decreased. Some vendors have made an alliance now to increase their compatibility, but the scope of the alliance is limited, and thus it is still inefficient. Second, to build SAN, skilled professionals with special training are necessary, and the management tool is insufficient; thus, the cost of building and maintaining is too high for many enterprises. Moreover, the spreading distance of FC-SAN is limited to 10 km.

NAS and SAN are different, but, in fact, they are complementary to each other in providing access to different types of data. They are used under different circumstances; SAN is optimized for high-volume block-oriented data transfers, while NAS is designed to provide data access at the file level. It is quite possible that both NAS and SAN are needed in the same company. To reduce the TOC (total of cost) and to overcome the drawbacks coming from the FC, the paper introduces a novel storage architecture called USN.

2 INTRODUCTION OF USN (UNITED STORAGE NETWORK)

In this section the USN is introduced, which merges the NAS and SAN, and uses the iSCSI as the interconnecting technology. As shown in Figure 1, the servers (including the application server and the metadata server) and the storage subsystem are interconnected by the IP-based switch (not by the FC-based switch as in FC-SAN). Both servers and storage devices support the iSCSI functions, and are connected to switch by iSCSI HBA (Host Bus Adapter). Most importantly, there is a direct high-speed interconnection channel, the Network Attached Channel (NAC) between switch 1 and switch 2, so the storage devices can be connected to the LAN directly without any other servers.

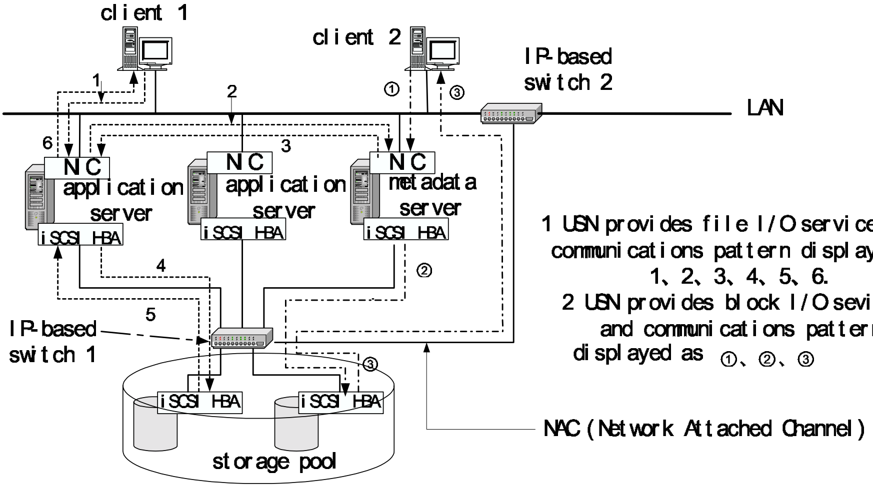


Fig. 1. USN architecture and communication pattern

2.1 The Character of USN

First, USN uses the iSCSI to build the storage network. iSCSI integrates the existing storage protocols (SCSI) with the IP protocol directly. With the integration, the storage and network can be merged, i.e. people can use IP networking devices (hub, switch, router and so on) to build their own IP-based storage network. Thus, the implementation of the storage network is simplified and total cost is greatly reduced. Moreover, the skills that people have developed to control and manage IP network will be applied to the USN. Currently, IP and Ethernet are used for all forms of networking. There are more management tools and trained staff available to install and operate networks based on these technologies. In USN, the existing software applications and tools can be used well. Especially with iSCSI, USN can spread across the MAN/WAN and break the distance limitation of FC-SAN (10 km).

Second, USN merges the NAS and SAN. USN attaches the LAN as the NAS, with benefits of the NAS: heterogeneous file sharing; internal resource pooling; exploiting the existing infrastructure; simple implementation; extensive connectivity; improved manageability, etc. Unlike the traditional NAS, it changes the tight coupling between the OS and the storage subsystem and separates the OS (file system) component from the storage system. In USN, there is a special server to perform the managing task concerning the metadata, it can also supply the block I/O services to the client directly. In addition, there is an application server to supply the file I/O services to the client directly. In USN, the storage subsystem adopts the SAN model, all storage devices can be connected to each other through network, but not through a bus. Thus, USN has the benefits of SAN, e.g. high scalability, etc. Especially, all storage devices can be virtualized and consolidated to a single storage pool by storage virtualization. NAS are file servers, SAN are network-based disks/RAID boxes, and the USN blurs the boundaries between the two systems. USN provides both the file I/O and block I/O services simultaneously; or all the storage space can be divided into the FILE and BLOCK spaces.

Most importantly, a high speed NAC (Network Attached Channel) is introduced creatively in USN, which connects the storage devices and LAN directly and moves data between the storage devices and the client without the server. When the USN provides block I/O services, and if the data block must be processed by the metadata server before its transfer to client, it is sent to the metadata server; if not, it can be transferred to the client via the NAC without the metadata server. This can decrease the metadata server workload, and shorten the response time of the client request. This approach enables data transfer between the client and storage subsystems directly.

2.2 The Communication Pattern

In practice, two special software modules must be developed and installed on the application server (USN Client Module) and the metadata server (USN Server Module) accordingly.

USN can provide both file I/O and block I/O services. With file I/O, all the file I/O requests sent to the application server OS are intercepted by the USN Client Module, and transferred to metadata server via LAN. The communication patterns are shown in Figure 2 as follows:

1. Client 1 sends the file I/O request to the application server via the LAN.
2. When the file I/O request arrives at the application server, the application server has no knowledge about the metadata of the file; so the USN Client Module on

the application server intercepts and forwards the request to the metadata server via LAN.

3. After the request has arrived at the metadata server, the metadata server (USN Server Module) finished functions such as security check, and verifies whether the application server has the permission to access the responding related storage subsystems; provides the locking mechanism, avoids access to the current file by subsequent applications and destroying the consistency; returns the metadata information back to application server, etc.
4. The application server (USN Client Module) accepts the metadata and sends the corresponding block I/O request to the storage subsystems.
5. The storage subsystems finish the block I/O operations; when the request is a write operation, the data are written into the storage subsystem and the finishing termination message is sent to the application server; when the request is a read operation, the data blocks are sent to the application server.
6. If the request is a read operation, the blocks of data returned from the storage subsystem are transformed to file by the file system of the application server.
7. Application server returns the file to client 1.

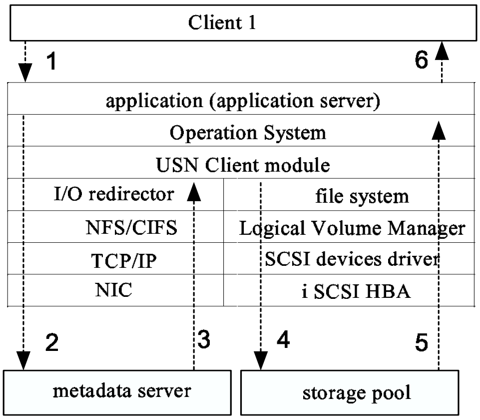


Fig. 2. The software layers of the application server

USN can also use the iSCSI protocol to provide the block I/O service. The communication patterns are shown in Figure 3.

- Block I/O requests (SCSI commands) are sent from the client 2, then encapsulated by the iSCSI HBA at client 2 and become the IP packets (iSCSI commands) which are transmitted in the TCP/IP network.
- Metadata server (USN Server Module) accepts the iSCSI commands, and completes functions such as authentications, authorizations, allocations, locks, etc.; then forwards the iSCSI commands to the corresponding storage subsystems.

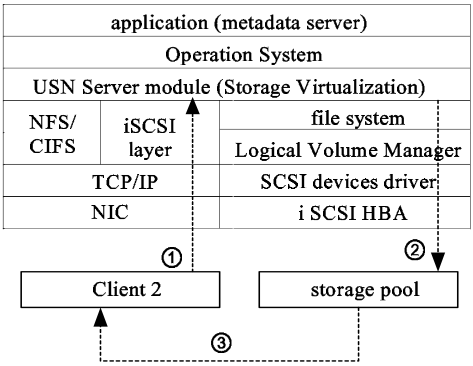


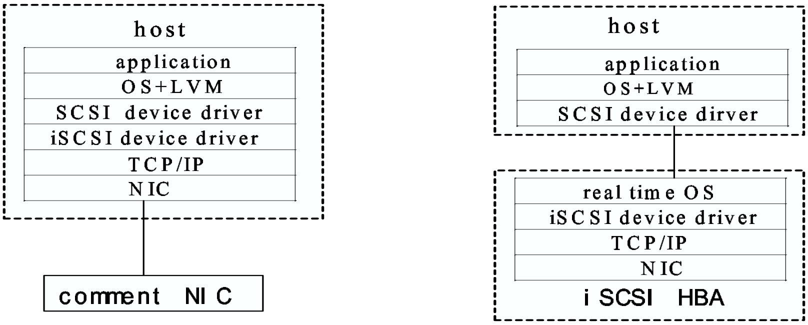
Fig. 3. The software layers of the metadata server

- In the storage subsystem, the iSCSI commands are stripped and become the SCSI commands which, like the SCSI commands, are sent by the application running at client 2. If the SCSI command is a “write” operation, the storage subsystem performs the operation, and the metadata server sends relevant metadata to the client; If the SCSI command is a “read” operation, the storage subsystems finish the operation, and the retrieved data blocks are encapsulated by the iSCSI HBA and converted to the IP packets again; the IP packets are transmitted to client 2 via the high-speed NAC directly, then striped by its iSCSI HBA of client 2 and become the block data needed by client 2. In the model, the functions of the file system are finished by client 2 or metadata server; client 2 finished mapping the files and blocks, and the directory operations must be finished by the metadata servers to maintain a single and global view of all the BLOCK spaces; thus, the metadata of all the data in the BLOCK space must be stored in the clients and metadata server simultaneously.

2.3 iSCSI Implementation

There are several approaches to implement iSCSI. Two approaches are introduced in the paper. The first is traditional NIC as shown in Figure 4a. The traditional NIC is designed to transfer packetized file level data, but not block level data. For a NIC, to process block level data they need to be placed into a TCP/IP packet before being sent over the IP network. Through the use of iSCSI driver on the host or the server, a NIC can transmit packets of block level data over an IP network. When using a NIC, the server handles the packet creation of block level data and performs all of the TCP/IP processing. This is extremely CPU demanding on the CPU and lowers the overall server performance. The advantage of such approach is that it can work on existing Ethernet NICs. The main disadvantage is that heavy CPU utilization is required for TCP/IP processing.

The second approach is the iSCSI HBA shown in Figure 4b. It combines the functions of NIC with the function of storage HBA; all the tasks of iSCSI driver and TCP/IP processing are performed by special adapter. The adapter takes the data in blocks, handles the segmentation and processing on the adapter card with TCP/IP processing engines, and then sends the IP packets via an IP network. The implementation of these functions enables users to create an iSCSI based storage network without reducing the performance of the server.



a traditional NIC

b iSCSI HBA

Fig. 4. iSCSI implementation

3 THE ANALYSIS OF ISCSI

3.1 Overview

iSCSI is the emerging Internet Engineering Task Force (IETF) specification that defines how SCSI and Ethernet work together to perform SCSI data transfers across TCP/IP networks. It allows the SCSI block storage SCSI commands to be carried by the standard TCP/IP protocols over the Ethernet wire. iSCSI is the convergence of SCSI, the dominant protocol for block storage I/O, with IP, the dominant protocol for computer internetworking, and leverages the existing TCP/IP infrastructure. With the development of the TCP/IP, the IP-based network more and more becomes the main infrastructure of the internet and intranet. Building the storage network with the iSCSI protocol fits in the development trend. iSCSI employs the TCP/IP flow control, congestion control, segmentation and assembly mechanisms, IP addressing and discovery mechanisms, etc. To evaluate whether iSCSI protocol is suitable to build the storage network, we have analyzed the protocol and compared it with the Fibre Channel protocol.

3.2 The Overhead on the I/O Channel

In TCP/IP, the data are copied from the NIC into the kernel memory first, subsequently to the user application memory. In the case of zero copy, the data are copied directly from the NIC into the user application memory and not via the kernel memory. The benefits of zero copy are obvious — it shortens the data flow path length and decreases the host CPU utilization. In FC, the FC Host Bus Adapter performs the zero copy processing with specialized tag approaches. The TCP protocol is a stream-based protocol, and it is difficult to implement the zero copy semantics because the TCP segments could be spread over multiple Ethernet frames. Therefore, the iSCSI header and data could be spread over multiple Ethernet frames. Data belonging to multiple unrelated iSCSI requests could arrive as part of a single TCP segment as well. Therefore, if the Ethernet frame containing the iSCSI header information is dropped or arrives out of sequence, then the NIC needs to buffer the trailing Ethernet frames until the frame containing the iSCSI header information is received; if iSCSI protocol is operating at link speeds, then the NIC would require a lot of memory in order to buffer the trailing Ethernet frames until the frame containing the iSCSI header arrives. Thus, in order to reduce the memory requirements at the network cards and to efficiently implement the zero copy protocol, it is necessary for iSCSI to adopt a framing mechanism similar to the one present in FC and to implement it on top of the TCP. In most cases, iSCSI will most likely be implemented using customized host adapter with built-in zero-copy support.

The block I/O request issued by the application normally varies in size between 4 K and 64 K; thus, the size of Ethernet (iSCSI) frame and FC frame is 1.5 K and 2 K, respectively. If the storage application has issued a block I/O request for a 16 K block, this request must be performed by fragmentation processing. There is more overhead with fragmenting and assembling the smaller frame size than with the larger frame size. On the other hand, in FC, the FC host adapter performs the fragmentation and assembly processing; thus, they offload these operations from the host CPU. In TCP/IP, the same operations are performed by the host; this increases the overhead and CPU utilization of the host machine. Therefore, as to the overhead on the I/O channel, the iSCSI is less suitable to build the storage network than the FC. In most cases, the above issues can be solved by using the specialized host adapter to implement the iSCSI, which has the built-in zero-copy supported, and performs the fragmentation and assembly processing, etc.

3.3 Flow and Congestion Control Mechanism

FC uses a credit based flow control mechanism. The receiver of the data allots the credits to the sender; when it has the necessary buffer space to store the sender's data, it allots the credits on the base of the sender's request. Only when the sender has not used up the credits, it can send data. The receiver returns the credits back to the sender when it sends an acknowledgement to the sender. The credit based flow control mechanism ensures that packets are never dropped due to data congestion.

In iSCSI, the End to End flow control mechanism is employed. That is, the two end points of a connection negotiate a window size based upon the buffer space available at their respective ends. The window represents the number of messages that can be sent without receiving an acknowledgment form the receiver. In iSCSI, the congestion will occur at the networking devices and at the end points; iSCSI reacts to congestion by dropping the packets.

The End to End flow control mechanism of iSCSI is more scalable than the credit-based flow control mechanism of FC. In the WAN/MAN, the sender has to wait quite long to get credit from the receiver for injecting new data into the network; thus the credit-based flow control mechanism decreases the utilization of the network and it is only adequate when the network delay is small. However, the End to End flow control mechanism of iSCSI is more suitable than the FC in the wide-area network, because the senders can dynamically increase or decrease their data transfer rates at the expense of packet drops at the network nodes during periods of high congestion. Currently the distance of SAN spread over is extending constantly; thus, the advantages of iSCSI are obvious.

3.4 The Discovery Mechanism

In FC, when a new device comes on-line, it contacts its plant manager. The plant manager, in turn, informs all the devices that have registered with the plant manager and which want to be informed about this event. Furthermore, in FC, when a device comes on-line, it performs a login with all the other devices that are present in the same zone, the switch to which this device connects informs all the other switches in the plant about its event. Therefore this mechanism can result in sending of many messages in environments with thousands of switches and devices, and in reducing the performance of the network greatly. Therefore the FC is only suitable for smaller scale networks. In iSCSI, an iSCSI initiator can discover an iSCSI target in the following different ways:

1. by configuring the target address on the initiator,
2. by configuring a default target address at the initiator, while the initiator connects to the target and requests a list of iSCSI names via a separate SendTargets command,
3. by issuing Service Location Protocol (SLP) multicast requests, to which the targets may respond,
4. by querying a storage name server for a list of targets that it can access.

In large network with thousands of devices, the storage node will use the mechanism of querying a storage name server rather than the multicast approach. Once the initiator receives the IP address and TCP port number of the target from the storage name server, the initiator establishes a connection. In the addressing, FC employs the 24 bit address, and iSCSI employs the 128 bit address. Thus the iSCSI is more scalable than the FC. Therefore, from the point of discovery mechanism and addressing, the iSCSI is more suitable than the FC in the wide-area networks.

3.5 Timeout and Retranslation Mechanism

iSCSI uses an adaptive timeout and retranslation mechanism of the TCP/IP protocol stack. In TCP/IP protocol, TCP monitors the performance of each connect and deduces the reasonable values of timeout. As the performance of a connection changes, TCP revises its timeout value. Every time, TCP records the time at which the segment is sent, and the time at which an acknowledgement arrives for the data in the segment. From the two times, TCP computes the elapsed time known as a *sample round trip time* or *round trip sample.* Whenever a new round trip sample is obtained, TCP adjusts its notions of the average round trip time for the connection. Usually, TCP software stores the estimated round trip time, RTT, as a weighted average, and uses new round trip samples to change the average slowly. Whereas the FC has a static timeout and retranslation mechanism, which does not adjust itself dynamically according to he network conditions, the sender may be in a timeout and retranslate the message either too soon or too late, and this can negatively impact the overall performance.

To conclude, in the circumstances of traditional LAN-based SAN, FC is more suitable for internetworking protocol than the iSCSI because of its zero-copy and the fragmentation and assembly mechanism. With the development of the application, the storage network can be deployed across a longer distance. In the case of the wide-area network, we can conclude from the above analysis that the iSCSI is more suitable because of its flow and congestion control mechanism, discovery and addressing mechanism, timeout and retranslation mechanism, etc.

4 PERFORMANCE EVALUATION

4.1 Methodology

For the case of using storage networks, the most important performance measures are average response time and network throughput. In this paper the two parameters are employed to evaluate the iSCSI-based USN. Average response time is the average time elapsed since the I/O request is sent by the initial device, until the I/O operations are finished by the destination device and the termination finishing message was sent back to the initial device by the destination device. Throughput is the maximum amount of I/O request processed by the storage subsystem per time unit. Average response time and throughput can be measured from different views, for example that of the user, OS, and disk controller, etc. For simplicity, in our experiment the two parameters are measured from the OS of the server.

To confirm that whether the iSCSI technology is well suitable to implement the USN, we have carefully designed two groups of experiments, one for the traditional

	CPU	RAM	OS	Disk	NIC/HBA
Host 1	Intel Pentium4 1500	256MB	Linux 7.1	Maxtor 91020D6 (IDE)	AGE-1000SX (NIC)
Host 2	PIII 450 (double)	256MB	Linux 7.1	ST318437LW (SCSI)	AGE-1000SX (NIC)
Host 3	PIII 450 (double)	256MB	Windows NT 4.0	ST318437LW (SCSI)	Qlogic SANbladTM 2300 (HBA)

Table 1. The machine configuration

FC-SAN (group 1), the other for own iSCSI-based USN (group 2) introduced in this paper. The iSCSI HBA is simulated with the traditional NIC as shown in Figure 4a. The Linux iSCSI packets developed by Intel Corporation are compiled and run (`https://sourceforge.net/projects/intel-iscsi`). Figure 5 gives a highly simplified view of the experiments. In the experiment, there are three host machines; as shown in Figure 5a, host 1 acts as both application server and metadata server, which connect the IP-based switch with the traditional NIC. Host 2 simulates the iSCSI-supported storage device, which connects the switch with the traditional NIC and connects the local disk drive with SCSI controller. As shown in Figure 5b, host 3 acts as application server, which connects the FC switch (HP-FC16) with the FC HBA. Unlike the FC16, it is the FC-based disk array (HP-FC60 disk array).

In the experiments, we have tried try our best to create a similar experimental environment; in fact, some configurations are different, for example the OS, the measurement tools, etc. The main purpose of the experiment is to conclude that whether the iSCSI is suitable to implement the USN, not to compare the performance of the iSCSI and FC, so these differences do not influence the feasibility of our experiment.

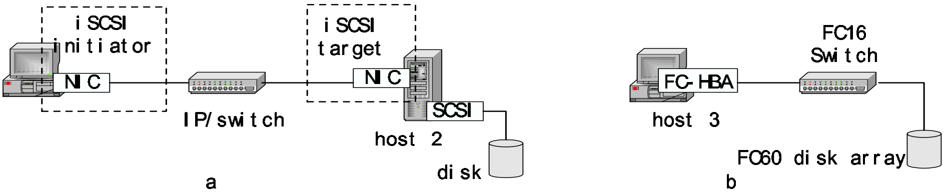


Fig. 5. The experiment configuration

4.2 Results and Discussion

In the experiment of group 1, the I/O meter is used to measure the throughput and average response time in different block sizes of the I/O request. In the experiment of group 2, the measurement tool comes from Intel (`https://sourceforge.net/projects/intel-iscsi`). The values of the experiment are shown in Table 2, and the corresponding curves of throughput and average response time dependence on the block size are shown in Figure 6.

	Throughput (IO/s)		Throughput (MB/s)		Average Response Time (ms)	
block size	FC	iSCSI	FC	iSCSI	FC	iSCSI
1 K	253.3162	1539.891	0.247379	1.50381	1.34857	3.94328
2 K	253.5375	1385.011	0.49519	2.70523	1.6384	3.93962
4 K	229.799	1202.381	0.897652	4.69681	2.27465	4.34659
8 K	206.6714	941.4272	1.61462	7.35429	3.36874	4.83426
16 K	179.6503	583.0336	2.807036	9.14099	4.02834	5.56148
32 K	151.3856	380.1472	4.730799	11.87946	4.97374	6.60105
64 K	124.0166	254.0096	7.751036	15.87556	6.03846	8.05808
128 K	97.82123	150.2704	12.22765	18.78378	8.43294	10.21478
256 K	76.77741	101.1032	19.19435	25.27588	12.56235	13.05359
512 K	50.22503	57.7428	25.11251	28.87154	17.347623	19.90252
1024 K	30.00799	31.8124	30.00799	31.81239	30.34289	33.31466

Table 2. The values of the experiment

Figures 6a, b show the effect of block size on throughput. As seen from Figure 6b, when block size is increased, the throughput (MB/s) increases (but the I/O/s decreases). Most importantly, the throughput of iSCSI is bigger than that of FC. As can be seen in Figure 6a, when the block size is small (smaller than 64 K), the throughput of the iSCSI is obviously larger than that of the FC. In most cases, the write/read request initiated by the application is decomposed into small I/O requests with block size of 4 K–64 K. Thus, from the point of throughput, the iSCSI is fit for implementing the storage networks. When the block size is larger than 64 K, the difference of throughput decreases. Currently the performance of the storage subsystem is becoming more and more depending on the disk controller. When the block size is 1024 K, the throughput of the iSCSI and the FC is similar. It can be understood easily that, in the experiment, to create similar experiment environment, the maximum throughput of the two disk subsystems is similar; and that when the block size is 1024 K, the throughput of the storage subsystem has reached the maximum throughput of the disk subsystems. In fact, after the throughput reached the maximum, increasing block size does not guarantee increased throughput but decreases the throughput instead.

Figure 6c shows the effect of block size on average response time. Note that average response time increases with block size. When the block size is more than 1024 K, performance of the storage subsystem decreases intensively. Thus, large block size is not fit for the I/O request. Most importantly, when the block size is smaller than 1024 K, the average response time of iSCSI is lower than that of the

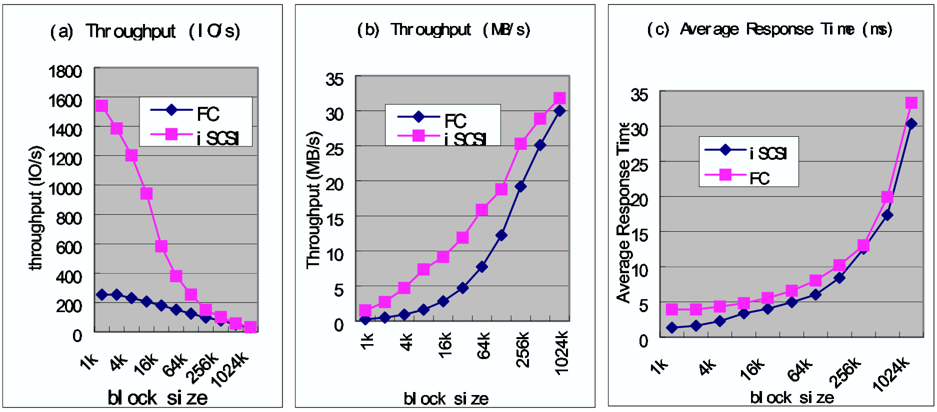


Fig. 6. The throughput and average response time curves

FC. This indicates that the performance of iSCSI is better than that of FC as to the average response time.

5 SUMMARY

In this paper, we have introduced a new storage architecture — the United Storage Network, which merges the NAS and SAN, and provides the file I/O services as NAS devices and the block I/O services as SAN. To overcome the drawbacks of the FC, we employ the iSCSI to implement the USN. To evaluate whether iSCSI is more suitable to implement the USN, we analyze the iSCSI protocol and compare it with FC protocol with respect to several components of a network protocol which impacts the performance of the network. From the analysis and comparison we can conclude that the iSCSI is more suitable to implement the storage network than the FC in the wide-area network environment. Finally, we have designed two groups of experiments carefully. It can be concluded from Table 2 and Figure 6 that when iSCSI and FC are employed to implement the storage network, the performance of iSCSI is better than that of the FC as to throughput and average response time (with the conditions of the two experiments being different). Most importantly, in most cases the read/write request initiated by the application was transmitted into the I/O request with 64 K block size. Figure 6 reveals that iSCSI throughput is larger and average response time is smaller than with the FC when block size is 64 K; thus, it can be concluded that iSCSI is more suitable for the storage network implementation.

Acknowledgements

This paper is supported by the National Natural Science Foundation of the P. R. China under the grant NO 60173043. It is also supported by National 973 Great Research Project of P. R. China under the grant NO G1999033006. We thank sincerely Cao Qiang, Liu Zhaobin, Han Dezhi, Luo Dongjian and some other members of the USN project group. They have creatively done a lot of works in modeling and implementing the USN.

REFERENCES

[1] Intel iSCSI protect. `https://sourceforge.net/projects/intel-iscsi`.

[2] Satran, J. et al.: iSCSI. `http://search.ietf.org/internet-drafts/ draft-ietf-ips-iscsi-09.txt`.

[3] Farley, M.: Building storage area networks. McGraw-Hill, 2000.

[4] Fibre Channel Industry Association. Fibre Channel Standard. `http://www.fibrechannel.com`.

[5] Xianglin, F.—Changsheng, X.—Zhaobin, L.: The IP based Storage of Image Information. Proceedings of the 2nd International Symposium on Multispectral Image Processing and Pattern Recognition, 22–24 Oct. 2001, China.

[6] Molero, X.—Silla, F.—Santonja, V.: Modeling and Simulation of Storage Area Networks. Accepted at the 8th International symposium on modeling, analysis, and simulation of computer and telecommunication systems, August 2000.

[7] Molero, X.—Silla, F.—Santonja, V.: Performance Analysis of Storage Area Networks Using High-Speed LAN Interconnects. Proceedings of the 8th ISPAN Conference, IEEE Computer Society, September 2000.

[8] Ruemmler, C.—Willkes, F.: Unix Disk Access Patterns. Processing of the 1993USENIX winter conference, pp. 228–235.

[9] Ruemmler, C.—Willkes, F.: An Introduction to Disk Drive Modeling. IEEE Computer, Vol. 27, 1993, No. 3, pp. 17–28.

[10] Brinkmann, A.—Salzwedel, K.—Scheideler, C.: Efficient, Distributed Data Placement for Storage Area Networks. In Procesing of the 12th ACM Symposium on Parallel Algorithms and Architectures (SPAA'2000), 2000.

[11] Wilson, S.: Managing a Fibre Channel Storage Area Network. Storage Networking Industry Association (SNIA) 1999.

Xianglin Fu received the BS degree in mechanics and the MS degree in mechantronics from the Wuhan University of Science and Technology, Wuhan, P. R. China in 1994 and 1999, respectively. Currently he is a PhD graduate of the Huazhong University of Science and Technology, and his research interests include network storage system, network attached storage, IP-based storage area network, etc.

Changsheng Xie received the BS degree in computer science and technology and the MS degree in computer architecture from the HUST (Huazhong University of Science and Technology), Wuhan, P. R. China in 1980 and 1985, respectively. Currently, he is a professor and supervisor of PhD of the HUST, and his research interests include network storage, and super high speed I/O channel.

Ruifang Liu received the BS degree in computer science and technology and the MS degree in computer science and technology from HUST (Huazhong University of Science and Technology), Wuhan, P. R. China in 1996 and 2000, respectively. Currently he is a PhD graduate of the HUST. His research interests include network storage systems, storage area networks, etc.

Shenggang Pan received the BS degree in computer science and technology from HUST (Huazhong University of Science and Technology), Wuhan, P. R. China in 1999. Currently he is a graduate of the HUST. His research interests include: iSCSI, networked storage systems, etc.