REMOVE MULTIMEDIA SERVER BOTTLENECK BY NETWORK ATTACHED DISK ARRAY WITH HETEROGENEOUS DUAL CHANNELS

Dan Feng, Fang Wang, Yuhui Deng, Jiangling Zhang
Key Laboratory of Data Storage System , Ministry of Education
Huazhong University of Science and Technology, Wuhan 430074, China

Keywords: Multimedia server, Disk array, Data transfer, Data redirection

Abstract: Multimedia service is pervasive on the Internet now and continues to grow rapidly. Most multimedia service provider systems have adopted a typical system architecture in which the storage devices are attached privately to the server. When a client browses some multimedia data from the server, data should be fetched from the storage devices and then forwarded to the client by the server. Unfortunately, with the steady growth of Internet subscribers, the multimedia server quickly becomes a system bottleneck. Network attached Disk Array is proposed to solve the bottleneck problem. There are two different channels in the disk array. One is a traditional peripheral bus to make the disk array work as a normal storage system. And the other is network interface to transfer data between clients and the disk array directly. The architecture avoids expensive store-and-forward data copying between the multimedia server and storage devices when clients download/upload data from/to the server. The latency is less than that with the traditional architecture and the average data transfer rate is higher. The system performance of the proposed architecture is evaluated through a prototype implementation based on the logical separation in the File Transfer Protocol. In multi-user environment, its data transfer rate is 2~3 times higher than that with a traditional disk array, and service time is about 3 times shorter. The most salient feature of the architecture is that it eliminates the server bottleneck, while dynamically increasing system bandwidth with the expansion of storage system capacity.

1 INTRODUCTION

Multimedia service is pervasive on the Internet now and continues to grow rapidly. All storage devices with a large volume of digitized media files are attached privately to the multimedia server.(H. Radha et Al., 1999), (J. Pieper et Al., 2001) In such systems, the multimedia server can become a bottleneck whenever bulk media data is requested, because the intensive memory-to-memory data copying between storage devices and the server occupy will quickly consume all resources of CPU and memory. Consequently, such systems cannot sustain the bandwidth requirements of a large number of media streams.

More recently, there have been some research efforts invested in solving the bottleneck problem of the multimedia servers. A distributed server architecture that places the streaming servers close to the user clusters has been proposed,)FA Tobagi et Al., 1995), (SA Barnett and GJ Anido, 1996) where the system is able to achieve scalable storage and streaming capacities by introducing more repository servers and local servers as the traffic increases. A scalable multimedia server based on a clustered architecture is discussed, (R. Tewari et Al., 1996) where a group of nodes are connected by a switch (interconnection network). Storage Area Network (SAN) based on Fibre Channel is built for many servers to share its huge logical capability while multiple servers can process multiple I/O requests synchronously. (Marc Farley, 2000) Massively_parallel And Real_time Storage (MARS) architecture, which connects storage devices to an ATM_based broadband network, is proposed for multimedia storage server. (M.M.Buddhikot et Al., 1994) These related works have made it progressive to enhance the aggregate bandwidth of the multimedia system.

In this paper, Network attached Disk Array (Net-DA) is proposed to solve the server bottleneck.

Feng D., Wang F., Deng Y. and Zhang J. (2004).
REMOVE MULTIMEDIA SERVER BOTTLENECK BY NETWORK ATTACHED DISK ARRAY WITH HETEROGENEOUS DUAL CHANNELS.
In *Proceedings of the First International Conference on E-Business and Telecommunication Networks*, pages 230-235
DOI: 10.5220/0001395202300235

There are two different channels in the disk array. One is SCSI (Small Computer System Interface) bus to make the disk array work as a normal storage system. And the other is network interface to transfer data between clients and the disk array directly. The architecture avoids expensive store-and-forward data copying between the multimedia server and storage devices when clients download/upload data from/to the server. For example, when a client requires data from the server, the read command is sent to the disk array through the SCSI bus by the server, but the data is directly transferred from the disk array to the client. So the latency is lower than that with traditional architecture. The system performance of the proposed architecture is evaluated through a prototype implementation based on the logical separation in the File Transfer Protocol. In multi-user environment, its data transfer rate is 2~3 times higher than that with a traditional disk array, and service time is about 3 times shorter.

The rest of the paper is organized as follows. The network attached disk array architecture is introduced in Section 2. Section 3 describes a prototype implementation of the FTP server. Performance of the proposed architecture is evaluated and analyzed through the prototype in Section 4. Section 5 concludes the paper with remarks on main contributions of the paper and future research directions.

2 ARCHITECTURE DESCRIPTIONS

2.1 Network attached Disk Array Architecture

Figure1 shows the architecture of the network attached disk array. There are two interfaces in the system, one is SCSI (Small Computer System Interface) and the other is NIC (Network Interface Card). One SCSI adapter that receives I/O command from the server is used as target to the server. Other three SCSI adapters are used as initiator (which we call a string controller) to the SCSI disks. One NIC is used to connect the Net-DA to Internet to transfer data directly.

The control software controls all components. All disks are organized as one large logical disk and configured as RAID style. The server manages the Net-DA through SCSI channel. All data requested by Internet clients are directly sent out through NIC to the clients to avoid memory-to-memory data copying between the server and Net-DA.

2.2 Multimedia Server System with Net-DA

A broadcast server for playing TV programs is designed with the network attached Disk array. It is shown in Figure 2. All Net-DAs are centrally controlled by the server through the SCSI channel for the convenience of management just like a normal storage system, while all network interfaces of Net-DAs are allowed parallel data transmission. By keeping the SCSI channel of Net-DA connected to the multimedia file server to exert central control, it strikes a good balance between a centralized file management and a distributed data storage.

When TV programs, which are stored as MPEG-2 files, are uncompressed and played by the server, the multimedia data is accessed through SCSI bus. When data needn't be processed by the server, such as TV program files are uploaded/downloaded by the clients in the editor network, they are transferred directly between the disk array and the clients through the network interfaces. A FTP server in the prototype system is implemented to support Net-DA transfer mode and it is discussed in detail in the following sections.

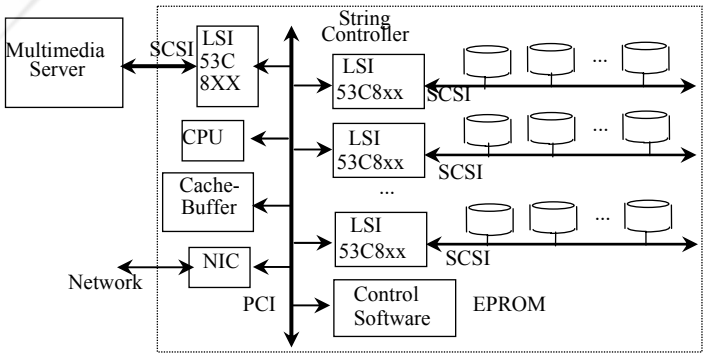

Figure 1: Architecture of Network attached Disk Array

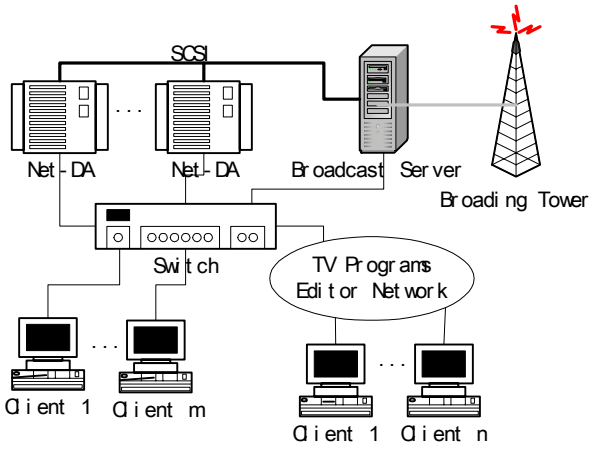

Figure 2: The architecture of the broadcast server system

Storage system capacity must keep pace with the continuous growth of multimedia data. The system in Figure 2 achieves this capacity scalability by expanding the system storage capacity incrementally with additional Net-DAs along with associated network interfaces that expand data transmission rate proportionally.

3 THE PROTOTYPE IMPLEMENTATION

3.1 The Redirection of FTP Procedure

A FTP session consists of two connections (see Figure 3). One is control connection for a client to connect with the server and it is kept in the whole session. Another is data connection for the server to transfer data with a client and it is established when data should be downloaded/ uploaded to the server. Because FTP uses different logical channels to transport control and data packet, we move the logical channel of data connection from the server to the physical network channel of Net-DA. Figure 4 shows the redirection procedure of the system, step by step, as explained below.

(1) After a client submits his request for a file service through FTP, a control connection is established for basic file service, such as displaying directory, deleting file and so on.
(2) When the client downloads a file, the server parses the data information (start address and data length) of the requested file over SCSI channel. Afterwards, the server sends the data information and client information to Net-DA over the network.
(3) A data connection is established between the Net-DA and the client.
(4) Net-DA gets the requested data from SCSI disks in terms of the data information, and transfers the data to the client according to the client information. The client begins to receive the TCP packets.
(5) When the file transfer ends, the Net-DA returns the finish status to the server.

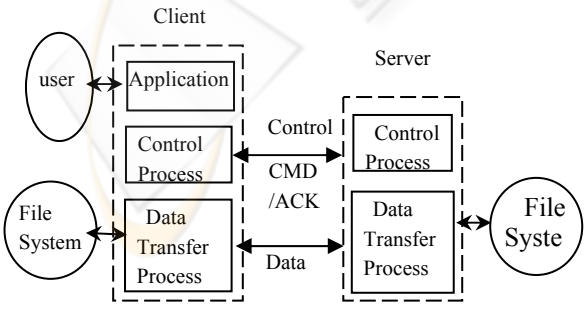

Figure 3: FTP connection

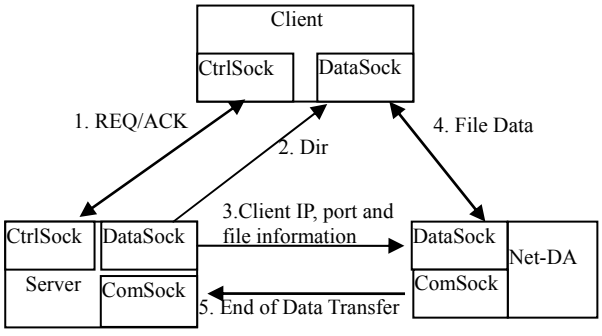

Figure 4: Redirection of FTP procedure based on the Net-DA

3.2 The Software Implementation

3.2.1 Software on the Net-DA Side

The control software modules of Net-DA (see Figure 5) are based on our earlier research prototype HUST-RAID (Steve Hotz et Al., 1998), with several noticeable due to because the fact that Net-DA has two heterogeneous interfaces (network interface and SCSI interface). The main differences are found in and described in terms of the following two modules.

(1) The network interface module

We use embedded real-time Linux with the priority-scheduling model as the development platform of Net-DA. Thus simple FS (file system) is retained, and the network interface module of Net-DA is implemented in application layer.

(2) The data redirection module

Because different and independent I/O tasks (coming from the server and network clients) are in the command queue, we use the data redirection module to distinguish these I/O tasks and return appropriate responses.

3.2.2 Software on the Server Side

A simple storage management is developed and FTP server in Linux is modified to support Net-DA transfer mode. In order to support a maximum number of simultaneous requests, the process of content retrieval is often optimized via two major approaches, namely, read scheduling and data placement. We use round-robin read scheduling method in the system. Data placement will be a future research direction in our work.

3.2.3 Client Side

Net-DA transfer mode is transparent to the clients and normal FTP Client software, such as CuteFTP 4.0, can run on the client.

A client application is developed to evaluate the efficiency of the Net-DA mode. The client application is multithread. A thread is dedicated to receiving packets from the network and placing them in application buffers. Another thread is invoked with interrupts, takes data from the buffer and save it. Experimentally, the parallel execution of I/O and CPU was shown to eliminate packet loss.

4 PERFORMANCE EVALUATION

4.1 Analysis of I/O Performance

4.1.1 Shorten I/O Path

In a traditional system where data is transferred through a peripheral bus, the I/O path consists of the following parts:

1. Data is read from disks to the memory in the Net-DA and the time is T_1.

2. Data is transferred from Net-DA to the kernel memory of the server through SCSI channel and the time is T_2.

3. Data is copied from the kernel memory to FTP application in the server and the time is T_3.

4. Data is transferred from the server to the client through network and the time is T_4.

5. Data is saved by the client and the time is T5.

So the total time of data download in a traditional system is $T=T_1+T_2+T_3+T_4+T_5$.

For Net-DA mode in the prototype system in Figure 2, the I/O path consists of only three parts which are:

1. Data is read from Disks to the memory in the Net-DA and the time is T_1.

2. Data is transferred from the Net-DA to the client through network and the time is T_4.

3. Data is saved by the client and the time is T_5.

Compared with the traditional system, the I/O path of the prototype system cut two parts. So the total time of data download in the prototype system is $T'=T_1+T_4+T_5$ and it is less than that of the

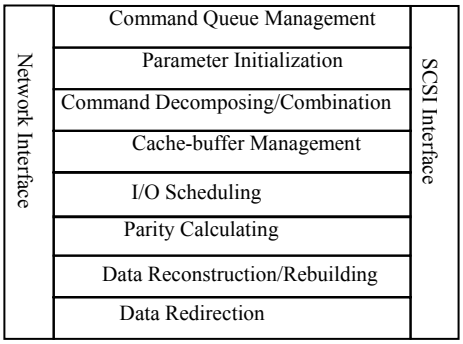

Figure 5: Control software modules of Net-DA

traditional system.

When only one client download/upload multimedia data from/to the server and the data size is L, the average rate of data transfer is

$$R_{Net-DA\text{mode}} = \frac{L}{T_1 + T_4 + T_5},$$

while the rate of the traditional architecture is

$$R_{Tradittional-\text{mode}} = \frac{L}{T_1 + T_2 + T_3 + T_4 + T_5}.$$

4.1.2 Enlarge Bandwidth

Assume that the number of the Net-DAs in the system is n. When multiple clients access the server at the same time, the aggregate bandwidth can be gotten from all the network interface of the Net-DAs.

Assume that the data size for client i (i=1,2,…,m) is Li and Pj (j=1,2,…,n) is the probability that the data is on the Net-DA j. Pj is affected by the I/O request distribution and the data location policy on the server system. When the I/O request follows a uniform distribution and the location policy balances data on the Net-DAs dynamically, for an instance, all Net-DAs are organized as RAID 0 style, the data is located on any Net-DA equally likely, resulting in Pj =1/n. Total size of requested data on a Net-DA is

$$l_i = \sum_{i=1}^{m} L_i \cdot p_i = \frac{1}{n} \sum_{i=1}^{m} L_i$$

When all requested data $\sum_{i=1}^{m} L_i$ is on a Net-DA and assume that the access time is t, the access time for size l_i on a Net-DA will be t/n.

When data is balanced on all Net-DAs in the system, aggregate bandwidth is nearly n times of one Net-DA because Net-DAs can work in parallel.

4.2 Performance Measurement

Table 1 shows the configuration of the prototype with Net-DA. In order to get a performance comparison between the prototype and the traditional system where the disk array is only attached to the server, we configure a HUST-RAID that has the same hardware platform as Net-DA, except for NIC. The HUST-RAID is directly attached to the multimedia server through the SCSI

Table 1: Configurations of the prototype system

	Client	Multimedia server	Net-DA
CPU	Pentium 450M	AMD Athlon MP 1600	Pentium 450M
Memory	128MB	512MB	128MB
NIC	RealTek100M	RealTek100M	RealTek100M
Target SCSI Adapter			LSI 53C895
Initiator SCSI Adapter		LSI 53C895	LSI 53C875
OS	RedHat6.0	RedHat6.0	RT Linux
SCSI Disks			ST173404LC

Table 2: Performance comparison between the prototype and the traditional system

operation	Number of clients	Traditional System			Prototype system with Net-DA		
		Data transfer rate(MB/s)	Average rate (MB/s)	Aggregate bandwidth (MB/s)	Data transfer rate(MB/s)	Average rate (MB/s)	Aggregate bandwidth (MB/s)
Download File	1	6.75	6.75	6.75	7.36	7.36	7.36
	2	2.83	2.62	5	4.82	4.67	9.33
		2.41			4.51		
	3	1.28	1.25	3.76	3.28	3.34	10.01
		1.12			4.01		
		1.36			2.72		
	4	0.88	0.82	3.29	2.62	2.48	9.93
		0.93			2.35		
		0.76			3.11		
		0.72			1.85		

channel. Peak read and write performances of HUST-RAID are 46MB/s and 33MB/s, respectively. The server is configured as a FTP server and it is connected to the 100Mbps Ethernet campus LAN of our university.

The performance of the system is measured by the aggregate bandwidth when a number of clients download/upload files from/to the server simultaneously. Table 2 shows the performance comparison between the prototype and the traditional system. The aggregate bandwidth of the prototype is larger than that of the traditional one and it approaches the network bandwidth. In multi-user environment, its data transfer rate is 2~3 times higher than that with a traditional disk array.

When we add another Net-DA to the prototype system, the aggregate bandwidth is nearly 20MB/s. It shows that the performance of the system increases almost linearly with the increase of the number of Net-DAs, and the system bottleneck has been removed from the server to network.

5 CONCLUSIONS

In this paper, we proposed and implemented an innovative network attached Disk array architecture, called Net-DA, which adds a network channel to the RAID and data can be transferred between the Net-DA and clients directly. A broadcast server with Net-DA is implemented to avoid the server bottleneck and has been applied in a TV station. We described the system architecture and software implementations in detail. The architecture removes the server bottleneck and dynamically increases system bandwidth with the expansion of storage system capacity. Experimental results provide useful insights into the performance behavior of the system based on the Net-DA with heterogeneous dual channels. The architecture can also be adopted to transfer massive data in other different servers, such as database server, HTTP server and so on.

Possible directions for future work include the development of parallel I/O scheduling algorithm, data placement methods and storage virtualization when many Net-DAs are attached to the server.

ACKNOWLEDGEMENTS

This research is supported by National Nature Science Foundation of China (No. 60273074, 60303032) and Huo YingDong education Foundation.

REFERENCES

H. Radha, Y. Chen, K. Parthasarathy, R. Cohen, 1999. Scalable Internet Video Using MPEG-4, *Image Communications*.

J. Pieper, S .Srinivasan, B. Dom, 2001. Streaming-Media Knowledge Discovery, *IEEE Computer*, IEEE Press.

FA Tobagi, 1995. Distance learning with digital video, *IEEE Multimedia*, IEEE Press.

SA Barnett and GJ Anido, 1996. A cost comparison of distributed and centralized approaches to video-on-demand, *IEEE J. Select. Areas Commun.*, IEEE Press.

M. M. Buddhikot, G. M. Parulkar, and J. R. Cox, 1994. Design of a Large Scale Multimedia Stoarge Server, *Computer Networks and ISDN Systems*, *Elsevier (North Holland)*.

R. Tewari, D. Dias, R. Mukherjee, H. Vin, 1996. High Availability for Clustered Multimedia Servers, In *Proc. Int. Conf. Multimedia Computing and Systems.*

Marc Farley, 2000. *Building Storage Networks*, Osborne/McGraw-Hill, USA.

Peng Chen, 1999. Design of High Performance RAID in Real-Time system, *ACM, Computer Architecture News,* ACM Press.

Steve Hotz etc, 1998. Internet Protocols for Network-Attached Peripherals, In *Proc. of 6th NASA Goddard Conference on Mass Storage System and Technologies in conjunction with 15th IEEE Symposium on Mass Storage System*, IEEE Press.

J. Mache, J. Bower-Cooley, J. Guchereau, P. Thomas, and M. Wilkinson, 2001. How to achieve 1 GByte/sec I/O throughput with commodity IDE disks, In *Proceedings of SC2001 - 14th ACM/ IEEE Conference on High-Performance Networking and Computing*, ACM/IEEE Press.