From the Proceedings of the Fourth Workshop on Workstation Operating Systems, Napa, CA, October 1993.

Providing Network Video Service to Mobile Clients[1]

Bruce A. Mah, Srinivasan Seshan, Kimberly Keeton, Randy H. Katz, Domenico Ferrari
{bmah,ss,kkeeton,randy,ferrari}@CS.Berkeley.EDU
Computer Science Division
University of California at Berkeley

Abstract

Mobile computing and multimedia are two emerging trends in computer systems. One foreseeable application suggested by these two trends is the playback of stored video on both mobile devices and conventional workstations. A system supporting such an application must provide performance-guaranteed delivery of video data to different types of clients, some of which may be mobile. In this paper, we address some of the issues involved in supporting such an application, namely the efficient layout of multiple representations of video data on a file server and network support for host mobility.

1.0 Introduction

Two recent trends in computer systems are the emergence of small, mobile computers using wireless networks and the increasing popularity of multimedia applications. These can be combined in new and interesting ways to produce a new class of personal, multimedia applications. One such application is the playback of stored video to mobile devices; one can imagine a “video server” which retrieves video sequences on demand for playback on the displays of hand-held, portable computing devices as well as conventional workstations.

Such a playback application has implications for both the storage and transport of data. The data storage and network facilities must provide for the timely delivery of large amounts of information to a variety of client machines, whose locations may change over time. Current hardware and software technology has addressed at least some of these points. Storage and retrieval of large amounts of data can be efficiently performed using Redundant Arrays of Inexpensive Disks (RAIDs), such as the RAID-II prototype storage server [Lee92]. RAIDs use parallel transfers from an array of drives to support high data transfer rates and high request rates. Real-time (performance-guaranteed) network services to support multimedia applications can be performed using (for example) the Tenet Real-Time Protocol Suite [Ferrari92]. This approach uses connection-oriented resource allocation to provide guarantees on performance parameters, such as delay, bandwidth, delay jitter, and buffering.

1. This research was supported by the Department of Energy under Grant DE-FG03-92ER25135/A000, by the National Science Foundation and the Defense Advanced Research Projects Agency (DARPA) under Cooperative Agreement NCR-8919038 and by the Corporation for National Research Initiatives, AT&T Bell Laboratories, Hitachi Ltd., Hitachi America Ltd., Pacific Bell, the University of California under a MICRO grant, and the International Computer Science Institute. Bruce A. Mah was supported by an NSF Graduate Fellowship. Kimberly Keeton was supported by an AT&T Graduate Fellowship.

In this paper, we discuss ongoing work concerning some of the remaining operating system and network issues that must be addressed in order to provide a video playback service to mobile clients. The first is the efficient layout of video data, in this case on a RAID-style disk array. We introduce a multiple representation file system tailored to the requirements of storing video data for consumption by a wide variety of display devices. The second issue is that of host mobility. The network must be able to quickly and efficiently reroute network connections when a user moves, for instance between the cells of a microcellular network. To address this need, we present several schemes for supporting connections in mobile environments.

2.0 The Environment

We envision a future computing environment that contains both high-end workstations, connected to a traditional wired local-area internetwork, and *mobile hosts* (MHs), which communicate using a wireless networking technology such as radio or infrared. The two types of networks comprise a heterogeneous internetwork, with *base stations* (BSs) acting as gateways between the wired and wireless networks. This environment is illustrated in Figure 1. The wireless network uses a microcellular architecture, with each base station controlling one or more room-sized microcells. When an MH moves between cells, the network must compensate by routing future data through the BS controlling the new cell.

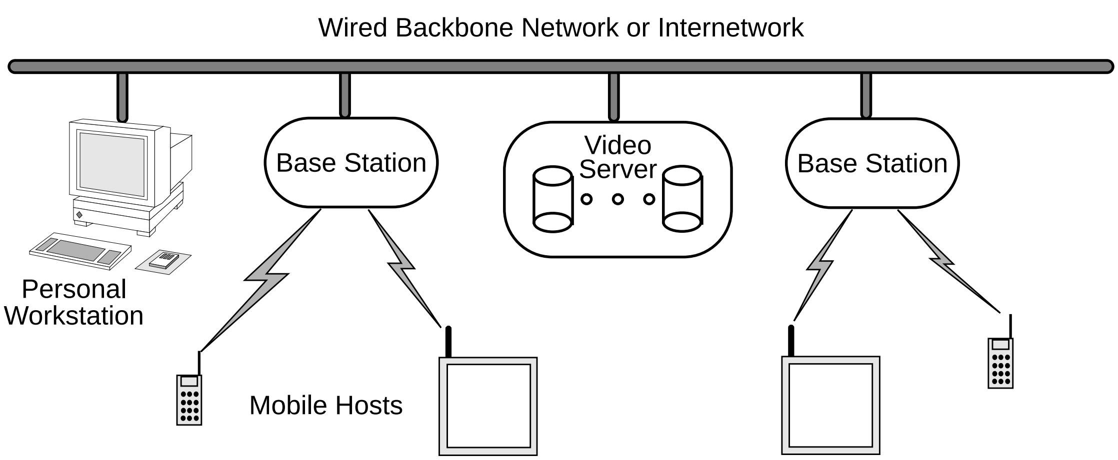

FIGURE 1. A future multimedia computing environment.

Many applications, such as personal communications services, conferencing services, or information access applications, will involve multimedia data. In this paper, we consider a video service in which data will be stored on a video server, such as a RAID or other large storage device, and transmitted over the network in response to user requests.

2.1 Requirements of a Video Service

A video service places several demands on its computing, storage, and communication environments. Computing requirements primarily involve processing of video data during playback. First, the server needs to be able to support the multitude of Qualities of Service (QoS) and format requirements of the different display devices, ranging from simple monochrome flat-panel displays to 24-bit color workstation monitors. Existing video servers generally store one representation of each video sequence, which is then converted using resizing or dithering algorithms as necessary.

This transformation is usually performed on the client machines to avoid overloading the server. In a mobile environment, the computing facilities available to perform such processing on the clients will be limited by the power budget. One solution to this problem is to store multiple representations of the video data on the server, using an encoding such as scalable compression [Chang93]. The storage of multiple representations allows the server to more closely satisfy client format requirements; clients then only need to perform decompression.

The storage system must ensure that video data arrive at the clients with low delay variance (jitter) and perhaps with low latency. The file system used to store the data must efficiently support periodically scheduled, long, sequential reads under strict delay requirements. Conventional file systems, such as the BSD Fast File System [McKusick83] and the Sprite Log Structured File System [Rosenblum91], store data in retrieval blocks that may be scattered across the disk. They provide no real-time guarantee on the retrieval time of video data. Several file systems have been designed and optimized specifically for continuous media data [Rangan91, Lougher93]. Although they address the needs of multimedia applications and can guarantee performance on file retrieval through the use of admission control, they do not support multiple representations of data. To solve both these problems, we propose a file system which is specifically designed for storing multiple representations of continuous media data.

The communication requirements concern the network services. As mentioned previously, the video data must arrive at the clients with low jitter and perhaps low latency. In conventional, wired networks, real-time performance guarantees can be provided using solutions such as the Tenet Real-Time Protocol Suite. These protocols allocate network resources to individual connections in order to provide bounds on throughput, delay, delay jitter, and congestion loss. The use of a connection-oriented network layer is necessary to furnish guarantees to each video stream and to protect those guarantees from malfunctioning or misbehaving hosts.

Host mobility requires that the connections used to deliver the video data be rerouted as a host's location changes. To minimize the disruptions in the video playback, the time required to complete this rerouting (also known as a *handoff*) must be minimized. Although there has been considerable work in supporting mobility in computer networks, much of it, such as [Ioannidis91, Teraoka91], has focused on IP, a connectionless network layer that cannot support hard performance guarantees. To support the network needs of a video service, we present several schemes for performing handoffs in a connection-oriented, mobile environment, which may eventually be used to provide Tenet-style performance guarantees to mobile hosts.

3.0 Continuous Media File Service in Heterogeneous Client Environments

Many existing disk file systems are unsuitable for the storage and retrieval of continuous media data, in that they are not optimized for long, periodic transfers and do not support multiple representations of data. We propose the use of a multiple representation file system to store video or other continuous media data, which uses knowledge of the disk system parameters, as well as the video data structure, to store multiple representations of video sequences for efficient playback. Our design is influenced by the architecture of RAID-II, a prototype disk array which provides high-bandwidth, network file service [Lee92]. We have evaluated several schemes for laying out video data on this disk array using a trace driven simulator.

3.1 A Multiple Representation File System

We propose several strategies for the layout of multi-representation video data. Each of these strategies explores different aspects of the *parallelism* and *concurrency* offered by striping data across disks in a disk array. Parallelism describes the number of disks that service a single user request for data. The higher the degree of parallelism (e.g., the more disks used to service a request), the higher the transfer rate that the request sees. However, as more disks cooperate to satisfy a single request, fewer independent requests can be serviced simultaneously. The degree of concurrency in the system is defined as the average number of outstanding user requests in the system at one time. The disks in a RAID system can be arranged into a two-dimensional array. Data is transferred in parallel across a *row* of disks. The disks in a single *column* perform independent, concurrent transfers. The general arrangement of a RAID system is shown in Figure 2. Our work uses the configuration of RAID-II, which currently has eight disks in each row and three disks in each column.

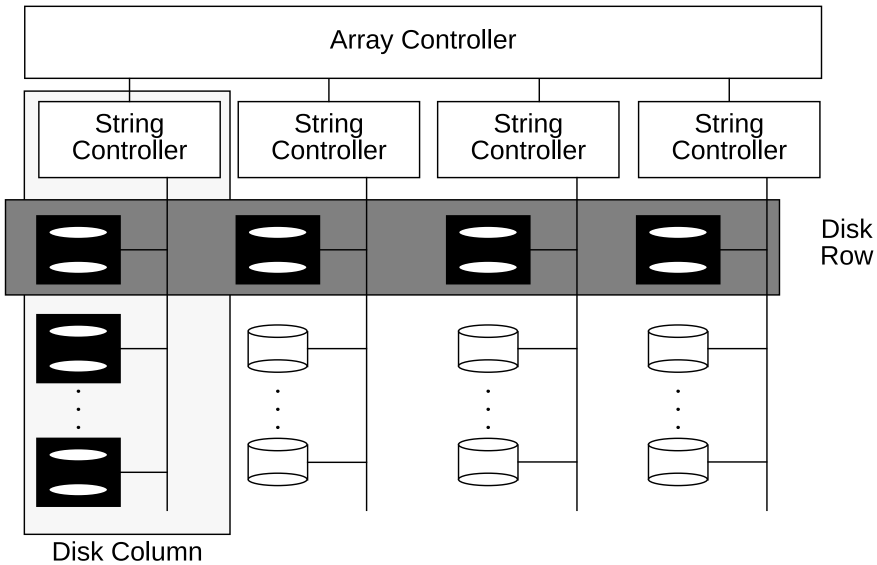

FIGURE 2. Architecture of a RAID.

Each of the layout schemes stores video data in retrieval blocks, whose sizes are defined based on user access patterns. In our system, the data for a retrieval block corresponds to one second's worth of playback data, as is done in [Lougher93]. While the data for a single retrieval block is stored contiguously on the disk array, logically contiguous retrieval blocks are not necessarily stored in a physically contiguous manner. Each retrieval block's data is striped across rows of the disk array. The *striping unit* describes the amount of logically contiguous data that is stored on a single disk.

We propose several data layout strategies, which differ in the degrees of parallelism and concurrency provided by varying the size of the data striping unit. In the first layout scheme, 1DISK, each retrieval block is stored on only a single disk. This scheme has a low degree of parallelism, because only a single disk from a single row is used to satisfy each request. The scheme has a high degree of concurrency, however, because each column can simultaneously satisfy an independent request. Here, the striping unit size is equal to the size of a retrieval block.

In the second strategy, 8DISKS, each retrieval block is striped across all eight of the disks in a row. This scheme gives a much higher degree of parallelism, because it transfers data from many disks for a given retrieval block. Due to this fact, the concurrency for this strategy is limited. Because data for a single retrieval block is spread across all of the disks in a row, the striping unit for this scheme is one-eighth that for the 1DISK scheme.

To evaluate the performance of these data layout strategies, we have created a simulator of a RAID-style disk array. It models the lowest-level components of the array (for example, the disks) in detail and uses approximations for the higher-level components (such as software delays on the file server). We are currently studying the performance of these and other layout strategies. Preliminary results are presented in [Keeton93b].

3.2 Additional Factors in Providing Continuous Media File Service

The storage of multiple representations of the same file allows some optimization of the system admission control policy. If new requests arrive during a period of system saturation, it may be possible to dynamically reduce the QoS provided to current requests (without violating client specifications) in order to free system resources that can then be used to satisfy the new client requests. Thus, the number of clients supported may be improved over that of other continuous media file services.

While an intelligent file layout policy aids in providing continuous media file service for heterogeneous users, it is only one component of a complete solution. To ensure that the real-time requirements of video (continuous media) transmission are met, we must ensure that the file server's operating system (in particular, the scheduler) can provide real-time guarantees. In addition, intelligent disk scheduling algorithms can further optimize the allocation of the disk subsystem's resources.

4.0 Network Services and Mobility

When a mobile host (MH) moves in the network, existing connections must be rerouted to maintain current conversations and various routing and location databases must be updated to allow new connections to be established. We present several algorithms for performing handoffs in connection-oriented networks.

4.1 Connection Rerouting

When an MH travels between wireless cells, the task of forwarding data between the wired network and the mobile host must be transferred to the new cell's base station (BS). This process, known as a handoff, must maintain end-to-end connectivity in the dynamically reconfigured network topology. Since our model of network communication is connection-oriented, each of the MH's connections (or channels) and the associated connection state must somehow be transferred to the BS in the new cell. The effectiveness of the rerouting performed by the handoff algorithm is determined by several criteria. In particular, it is desirable to minimize the service disruptions and overheads such as latency, necessary buffering, and network complexity.

Two straightforward solutions to the challenge of connection-oriented handoffs are forwarding data from the original BS to the new BS and establishing a new connection between the video server and the new BS. Because these approaches incur considerable overheads, we propose several alternate algorithms that modify an existing connection at a point called the *crossover point*. We define the crossover point to be the network node where the old and new connections diverge. Each algorithm chooses the crossover point in a different manner and performs different modifications to the existing connections.

The first scheme, *Incremental Re-establishment* (IR), capitalizes on the logical locality of geographically adjacent cells. This algorithm attempts to re-use as much of the existing connections as possible. The crossover point is chosen to be the point at which paths from the server to the two BSs diverge. New connections are established from their crossover points to the new BS. The effects of this procedure are shown in Figure 3.

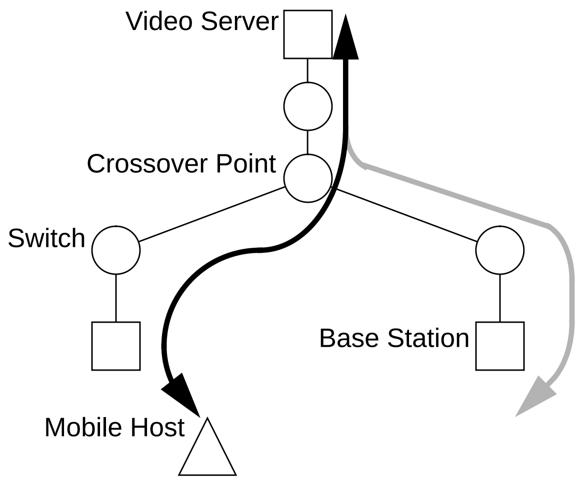

FIGURE 3. The MH has just moved from the right BS to the left BS. The existing network connection has been incrementally re-established from the crossover point.

A second scheme, known as *Multicast-Based Re-establishment* (MB), uses multicast operations (in networks that support multicasting) to execute handoffs. An MH's new base station is first added as a destination of a multicast channel; that MH's old base station is then deleted from the channel's set of destinations. Because data destined for the MH is transmitted simultaneously to multiple base stations during the interim, the actual switchover can be relatively quick compared to the IR scheme. This scheme requires only a small layer of functionality to be added on top of the multicast facility in order to support mobility.

A more complete description of these algorithms, as well as a preliminary analysis, can be found in [Keeton93a]. We are constructing a trace-driven simulator that we will use to quantitatively evaluate the performance of these approaches under various workloads and network configurations.

4.2 Connection Establishment

In current wired network environments, host addresses provide important routing information. However, with the introduction of mobile hosts using fixed network addresses, this correspondence is no longer valid. Therefore, it necessary to use new host location algorithms. Examples of solutions to this problem can be found, for example, by adapting algorithms used by mobile IP solutions such as in [Ioannidis91, Teraoka91].

5.0 Conclusion

The advent of small, mobile computers using wireless networks and the increasing popularity of multimedia applications imply the emergence of a number of new applications. One such application is a video playback service to the displays of mobile devices. Current technology addresses some of the requirements of such an application, namely the storage and retrieval of large amounts of data and the ability of a network to provide real-time performance guarantees to applications.

Several such application requirements remain to be addressed. To satisfy the widely varying requests of heterogeneous users, we have proposed the use of a multiple resolution file system tailored to the requirements of video data. This approach may also allow us to improve a video file server's ability to accept many client requests. To provide connection-oriented network services which support host mobility, we have proposed several strategies for performing cell-to-cell handoffs. We are in the process of analyzing and evaluating these ideas through simulation.

6.0 References

[Chang93] E. Chang and A. Zakhor. "Scalable Video Coding Using 3-D Subband Velocity Coding and Multirate Quantization," *Proceedings ICASSP 1993* (to appear).

[Chen93] P. Chen, et al. "Performance and Design Evaluation of the Raid-II Storage Server," *Proceedings of International Parallel Processing Symposium 1993 Workshop on I/O* (to appear).

[Ferrari92] D. Ferrari, A. Banerjea, and H. Zhang. "Network Support for Multimedia," Technical Report TR-92-072, International Computer Science Institute, 1992.

[Ioannidis91] J. Ioannidis, D. Duchamp, and G. Maguire. "IP-based Protocols for Mobile Internetworking," *Proceedings of SIGCOMM '91*, pp. 235-245.

[Keeton93a] K. Keeton, B. Mah, S. Seshan, R. Katz, and D. Ferrari. "Providing Connection-Oriented Network Services to Mobile Hosts," *Proceedings of the 1993 USENIX Symposium on Mobile and Location-Independent Computing* (to appear).

[Keeton93b] K. Keeton and R. Katz. "The Evaluation of Video Layout Strategies on a High-Bandwidth File Server," submitted to the Fourth International Workshop on Operating Systems and Network Support for Digital Audio and Video, 1993.

[Lee92] E. Lee, et al. "RAID-II: A Scalable Storage Architecture for High-Bandwidth Network File Service," Technical Report UCB/CSD 92/672, University of California at Berkeley, 1992.

[Lougher93] P. Lougher and D. Shepherd. "The Design of a Storage Server for Continuous Media," *The Computer Journal*, Vol. 36, No. 1, 1993, pp. 32 - 42.

[McKusick83] M. McKusick, W. Joy, S. Leffler, and R. Fabry. "A Fast File System for UNIX," *ACM Trans. on Computer Systems*, Vol. 2, No. 3, August 1984, pp. 181 - 197.

[Rangan91] P. Rangan and H. Vin. "Designing File Systems for Digital Video and Audio," *Proceedings of the 13th SOSP*, pp. 81 - 94.

[Rosenblum91] M. Rosenblum and J. Ousterhout. "The Design and Implementation of a Log-Structured File System," *Proceedings of the 13th SOSP*, pp. 1 - 15.

[Teraoka91] F. Teraoka, Y. Yokote, and M. Tokoro. "A Network Architecture Providing Host Migration Transparency," *Proceedings of SIGCOMM '91*, pp. 209-220.