

# 다중 스래드를 이용한 개인용 VOD 시스템의 설계와 구현

(A Design and Implementation of Personal VOD System using Multi-thread)

박 성 식<sup>†</sup> 이 호 석<sup>\*\*</sup>

(Sung Sik Park) (Ho Suk Lee)

**요 약** 비디오 정보를 사용자에게 서비스하기 위해서는 기존의 범용 파일 서버와는 다른 새로운 개념의 서버가 요구된다. 본 논문에서는 실시간 재생 서비스를 제공하는 개인용 VOD(Video-On-Demand) 서버를 설계하고 이를 Windows NT 환경에서 구현하였음을 보인다. 제안하는 개인용 VOD 시스템은 VCR(Video Cassette Recorder) 연산의 지연시간, 사용자 접속 승인 지연시간을 줄였으며 빠른 VCR 연산으로 VOD 시스템에 접속하는 사용자에게 좀더 빠른 데이터를 전송할 수 있도록 하였다. 또한 사용자들의 접속을 원활히 하기 위하여 대화 형태의 인터페이스를 제공한다.

**Abstract** In this paper, we designed and implemented a personal VOD server using multi-thread for the real-time services of video data to the users. The system consists of service manager, disk manager, and network manager. The service manager controls the user admission to the system using the VBR(Variable Bit Rate) characteristics of MPEG (Moving Picture Experts Group) video data. The disk manager provides the functions for VCR-like operations. The disk manager reads blocks from the disk by the size of GOP(Group of Pictures) and buffers the data to the user's thread. The network manager uses TCP/IP for the communication between the server and the client. The communication starts from the client and the VOD server must respond to this request. The implemented personal VOD system shows an acceptable performance.

## 1. 서 론

컴퓨터 및 통신 기술의 발전으로 고속 망에서 문자, 이미지, 오디오, 비디오 등 다양한 형태의 정보를 제공할 수 있게 되었다. 이러한 다양한 형태의 멀티미디어 정보를 다수의 고객에게 서비스하기 위해서는 기존의 범용 파일 서버와는 다른 새로운 개념의 서버가 요구되고 있다[1][2]. 이러한 서버는 광대역 통신망의 등장이나 데이터의 압축기법의 향상으로 대용량과 실시간 서비스를 요구하는 멀티미디어 데이터의 서비스를 가능하게 한다. 멀티미디어 데이터는 숫자나 문자 등의 정형

등의 비정형 데이터(unformatted data)가 통합된 데이터이다[3][4][5]. 특히 오디오나 비디오와 같이 시간에 따라 표현되는 매체를 연속매체(continuous media)라 한다[5]. 이러한 연속매체는 시간 연속적인 특성을 가지고 있고, 시간 연속성이 보장되지 못하면 정보를 가진 데이터로서 의미를 상실하게 된다. 또한 연속 매체는 대용량의 특징으로 인하여 데이터의 압축처리가 필수적이다. 비디오 데이터의 압축 기법이 표준화되고 고속 통신망의 확산에 따라 비디오 객체 저장 서버는 주문형 비디오(VOD:Video-On-Demand), 전자상거래, 홈쇼핑(home shopping), 그리고 대화형 TV(interactive TV) 등에 넓게 응용되고 있다[6][7]. 연속 매체는 기존의 데이터와 달리 검색을 위한 특징 표현이 어렵다. 그리고 연속 매체는 시간에 따라 표현되므로 탐색 방법이 기존 데이터의 방법과는 다르다. 비디오 객체의 경우, VCR(Video Cassette Recorder) 기능과 같이 사용자가

<sup>†</sup> 비 회 원 : 호주 타스마니아대학교 컴퓨터과학과

<sup>\*\*</sup> 정 회 원 : 호서대학교 컴퓨터공학부 교수  
hslee@dogsuri.hoseo.ac.kr

논문접수 : 1999년 7월 2일

심사완료 : 2000년 4월 3일

데이터(formatted data)와 문자, 이미지, 오디오, 비디오

직접 시스템과 대화를 통해 원하는 화면을 찾는 방법이 효율적이다[8]. 이 경우 멀티미디어 서버는 재생, 임의 속도 고속 재생, 그리고 정지 등 다양한 비디오 연산을 수행하고 지원해야 한다. 대용량 연속 매체의 처리를 위하여 디스크의 버퍼와 네트워크 대역폭 같은 자원의 효율적 관리가 필수적이다[6]. 연속 매체는 시간 연속적인 성질로 인하여 일단 서비스가 시작되면, 종료될 때까지 시스템은 사용자가 요구하는 품질을 보장하여야 한다. 멀티미디어 서버는 사용자의 대용량 정보의 요구와 다양한 연산의 요구 때문에 자원의 관리가 어렵다. 특히 상영시간이 100분 가량 되는 영화 한편 당 약 1GB 정도의 저장공간이 필요하며 VOD 서버의 경우 이러한 영화를 수십 ~ 수백 편을 저장하고 있어야 하므로 디스크의 저장능력이나 처리 능력의 향상을 위하여 디스크 배열을 이용하는 것이 일반적이다. 또한 많은 수의 사용자 접근에 의한 영화의 병행 접근이 요구되며 이를 위해서 디스크 스트라이프 기술이 사용된다. 멀티미디어 서버는 연속 매체의 유형에 따라 요구되는 저장 시스템의 입출력 대역폭이 다르다. 100분 분량의 비디오 데이터를 100:1로 압축할 경우에도 1GB 이상의 저장공간이 필요한데 수백, 수천 편의 비디오 데이터를 저장하기 위해서는 수백 기가 또는 수 테라(tera)바이트 이상의 대용량 디스크 배열이 필요하다. 또한 많은 사용자가 동시에 VOD 서비스를 받기 위해서는 입출력 대역폭이 수백에서 수천 Mbit/s가 요구된다[9][10].

본 논문에서는 연속 매체를 서비스하기 위한 비디오 객체를 병렬 디스크에 저장하고 이를 실시간으로 서비스하기 위한 개인용 VOD 시스템의 개발에 초점을 둔다. 사용자의 요구로 인한 VCR연산, 사용자 프로그램과 서버와의 프로토콜 개발, 그리고 서버의 자원을 실시간으로 파악할 수 있도록 시스템을 개발하였다[11][12].

## 2. VOD 시스템의 특성 및 구성도

### 2.1 VOD 시스템의 요구사항

주문형 비디오 시스템의 요구 사항은 다음과 같다. 첫째, 미디어 재생의 연속성을 보장해야 한다. 둘째, 많은 수의 사용자를 동시에 서비스 할 수 있어야 한다. 이는 위에서 언급한 연속성 보장 조건과 상충되는 면이 있다. 즉, 시스템의 가용능력을 넘어서서 새로운 사용자를 수용하면 기존의 사용자들은 서비스 질의 저하를 경험하게 된다. 사용자의 수와 연속성 조건간의 상충적 관계는 효과적인 새로운 사용자의 수용 여부 검사(feasibility test)를 요구한다. 셋째로, 대규모의 저장능력이 요구된다. 넷째, 서버의 구조가 충분한 수의 사용자를 지원할

수 있도록 입출력 대역폭을 지원해야 한다. 프로세서와 디스크의 속도는 계속 벌어지고 있는 추세이므로 디스크 배열과 같은 방법을 통해 이를 해소해야 한다. 다섯째, FF(Fast Forward) 및 FR(Fast Reverse)등의 서비스를 지원하기 위한 임의 접근 기능이 요구된다[5][11][13].

### 2.2 개발 시스템의 전체 구조

위에서 언급한 요구 사항들을 반영하는 동시에 효율적인 구현을 위하여 다음 사항들을 고려하였다. 첫째, 모든 멀티미디어 스트림에서 각 데이터 블록의 재생시간은 일정하다. 이는 정해진 서비스를 보장하며 주기적 입출력 요청을 유발한다. 둘째, 오디오, 비디오를 비롯한 여러 미디어는 동기화 형태로 저장되므로 별도의 동기화 기법은 불필요하다. 오디오 및 비디오를 별도의 미디어로 저장하는 경우 여러 가지 장점이 있으나 서버와 클라이언트간의 피드백 메커니즘이 필요하다. 셋째로 개발시스템은 그림 1과 같이 기능적으로 구분된 계층구조를 갖는다.

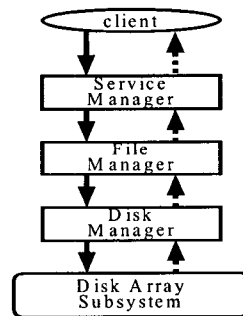


그림 1 개발시스템의 기능적 구성

서비스 관리자는 새로운 사용자의 수용 여부 검사, 서비스 목록에 등록된 사용자의 주기적 서비스, 스트림 관리 서비스 등을 수행한다. 서비스 관리자는 각 사용자에게 연속적인 재생 서비스를 제공해야 하며 따라서 전체 서버가 과부하가 되지 않도록 해야 한다. 파일 관리자와 디스크 관리자는 파일 관리자가 요구하는 디스크의 입출력을 수행한다. 파일관리자의 구현을 용이하게 하기 위해 디스크 서버 시스템의 구성에 관계없이 요구 순으로 제공되어 진다[14][15].

### 2.3 하드웨어사양과 구현 환경

하드웨어 환경은 그림 2와 같다. 서버는 Adaptec-AHA-2940 SCSI어댑터, 4개의 Quantum fireball tm2110s SCSI 디스크, 233 MMX 프로세서, 3Com509 Lan Card, ISA 버스, IDE 디스크 등으로 구성된다.

SCSI 디스크는 멀티미디어 데이터를 저장하며 Windows NT 4.0은 별도의 IDE 디스크에서 동작한다. 사용자 하드웨어는 MPEG viewer가 설치된 Pentium급이며 네트워크 카드로 서버와 접속되어 있다. 사용자의 운영체제는 MS-Windows이며 서버와의 통신 프로토콜은 TCP/IP를 이용하며, 시스템 제어와 VCR 프로토콜은 6장에서 기술하였다.

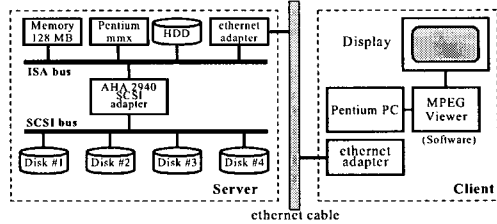


그림 2 하드웨어 사양

2.4 서버의 설계

본 연구에서는 서버의 자원을 공유하기 위하여 디스크 및 네트워크 관리를 하는 스레드 모듈을 각각 두었다. 그림 3은 서버의 구성도이다. 스레드 모듈은 디스크 관리자와 네트워크 관리자들의 실행 순서를 결정한다. 이때 자원의 사용량은 동적으로 자원 상태표(Resource Table)에 기록된다. 서비스 관리자 스레드는 자원 상태를 참조하여 사용자 계획확인 및 수용여부를 판단하고, 수용이 허가된 클라이언트의 자원정보를 GOP(Group of Picture)단위로 읽어 들이도록 디스크 관리자 스레드에게 요청을 한다. 네트워크 관리자 스레드는 단일 클라이언트를 담당하고 클라이언트의 요청에 따라 버퍼의 데이터를 전송한다. 전송시의 전송 패킷의 크기 또한 GOP단위이며 버퍼는 동적으로 할당된다. 서버에서 서비스 관리자와 사용자의 구조는 메인 스레드와 새로 생성되는 스레드이며 새로운 사용자가 서버에 서비스를 요하는 접속이 이루어지면 서버에서는 사용자 스레드가 생성이 된다. 서비스 관리자 스레드는 네트워크 관리자 스레드와 디스크 관리자 스레드를 제어하면서 사용자 스레드에게 서비스를 한다. 디스크 관리자는 담당 클라이언트로 데이터를 읽기 위한 스레드로 구성된다. 이 스레드는 전역 FAT(File Allocation Table)에서 해당 서비스의 GOP를 참조한 후 이를 이용하여 디스크로부터 접근 버퍼로 데이터를 입력한다. GOP의 순서와 크기는 FAT에 기록되어 있으며, FAT는 그림 3에서 Data Allocation Process에 의해 오프라인 작업으로 이루어진다. Sub FAT는 서비스 중인 클라이언트의 디스

크 접근 위치와 history를 기록한다. Sub FAT는 클라이언트가 서비스를 받은 내용과 현재 진행중인 상태를 기록하며 차후 클라이언트의 재접근 시 이용된다. 그림 4는 디스크 관리자 스레드와 네트워크 관리자 스레드의 실행 순서이다.

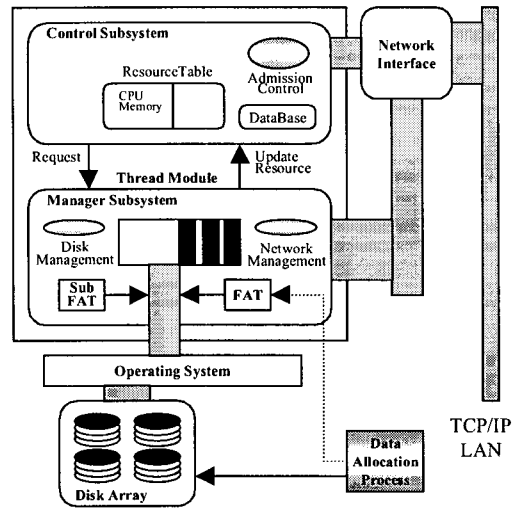


그림 3 서버 시스템의 구조

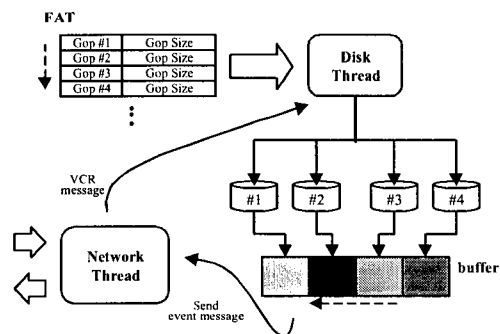


그림 4 디스크 관리자 스레드와 네트워크 관리자 스레드의 실행 순서

디스크 관리자와 네트워크 관리자는 버퍼를 공유한다. 두개의 스레드가 이러한 공유 버퍼에 접근을 위하여 세마포어 동기화가 필요하다. 디스크 관리자 스레드는 전송할 버퍼에 데이터를 채운 후 네트워크 관리자 스레드로 전송 메시지를 보낸다. 전송 메시지를 받은 네트워크 관

리자 쓰레드는 먼저 클라이언트로의 전송을 위하여 플래그를 set한다. 이 플래그가 reset되지 않으면 디스크 관리자는 전송 메시지를 보낼 수 없다. 이 과정 후 해당 클라이언트에 전송할 데이터의 size를 전송한 다음 실제 데이터를 전송한다. 전송이 완료하면 플래그를 reset한다. 데이터를 받은 클라이언트는 받은 데이터의 size와 수신한 내용을 비교하여 데이터 오류 체크를 한다. 만약 오류가 발생하면 재전송 요구를 하며 오류가 없으면 클라이언트는 MPEG viewer와 연결하여 사용자에게 서비스를 한다. 클라이언트가 서비스를 받는 중간에 VCR정보(Play, Stop, FF, FR)를 보내면 네트워크 관리자는 이것을 분석하여 디스크 관리자에게 메시지를 보내고 현재 진행중인 서비스를 중단한다. 디스크 관리자는 다시 이를 위한 버퍼링에 들어간다. 그림 5는 위에서 언급한 서비스 관리자와 디스크 관리자 그리고 네트워크 관리자의 관계를 간단히 나타낸 관계도이다.

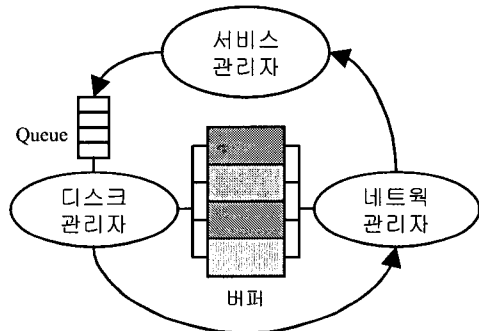


그림 5 관리자들의 제어 흐름도

### 3. 자원의 재배치

#### 3.1 자원 재배치의 필요성

멀티미디어 데이터는 대용량의 특성과 실시간 처리를 특성으로 한다. 그래서 일반적으로 VOD서버는 동영상 데이터의 저장을 위하여 여러 개의 디스크를 가진 디스크 스트라이프 구조나 RAID(Redundant Arrays of Inexpensive Disks) 구조를 한다. 이러한 여러 개의 디스크로 분리된 구조를 가진 VOD 시스템에서 디스크의 부하 균형을 위하여 동영상 데이터를 분리하여 재배치하는 방법은 필수적이다[16]. 이러한 이유에서 본 논문에서도 비디오 CD인 MPEG-1 데이터를 GOP단위로 재배치하였다. 비디오 CD는 순수 MPEG 데이터와는 조금 다른 pack과 packet layer로 묶여져 있으며 이는

각 네트워크로 전송을 목적으로 나누어져 있다. 이러한 구조를 가지고 있는 비디오 데이터의 형식에서 GOP 단위로 저장하는 것은 비효율적일 수 있으나 VOD 시스템에서는 원거리 서비스를 고려하지 않기 때문에 많은 시스템이 GOP 단위로 저장을 한다. 본 논문에서는 시퀀스 헤더와 GOP단위로 각 디스크에 저장하는 방법을 사용하였으며 저장된 데이터를 인덱싱할 수 있는 별도의 테이블을 작성하였다. 시퀀스 헤더단위로 분리하여 저장하는 것은 단일 GOP데이터도 재생이 가능하도록 하기 위한 것이다.

#### 3.2 비디오 CD의 구조

비디오 CD는 ISO 11172 스트림 형식으로 구성되어 있고 그 형식은 Pack layer와 Packet layer로 나누어진다. 비디오 CD 마지막에 32비트 코드 ISO\_11172\_end\_code(0x000001B9)를 포함한다. 그림 6은 비디오 CD 즉 MPEG-1 비디오 스트림의 구조이다[3][4][17][18].

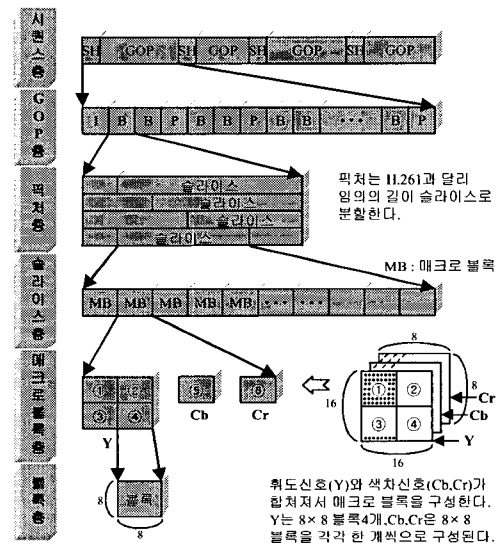


그림 6 MPEG-1비디오 스트림의 구조

#### 3.3 재배치 기법

그림 6과 같은 구조를 가진 MPEG-1 데이터는 시퀀스를 하나의 단위로 각 디스크로 분리하여 재배치한다. 시퀀스를 분류하기 위하여 시퀀스 헤더 코드(0x000001B3)를 검색해야 하며 많은 데이터를 검색하는 작업이 진행되는 동안 소요시간을 줄이기 위하여 임플리크 쓰레드를 두고 버퍼에 push하는 overlapped I/O방

식을 적용하였다. 다음 그림 7은 MPEG-1 비디오 파일을 시퀀스 헤더부분으로 분리하는 자원 재배치 프로그램의 순서도이다[19].

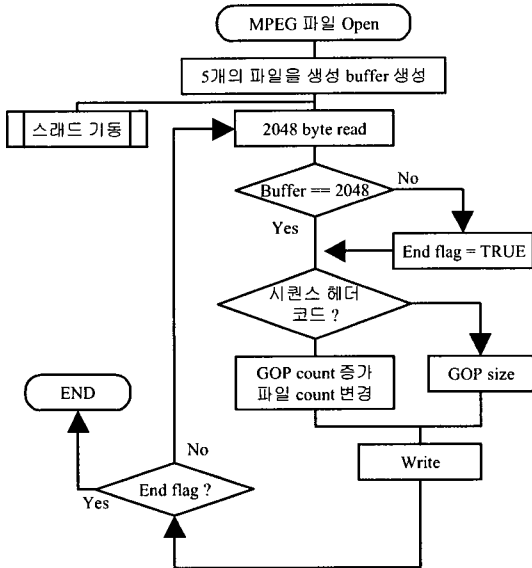


그림 7 자원 재배치의 순서도

MPEG-1 파일을 open한다. open된 파일의 검색을 위한 버퍼를 생성하고 MPEG-1 데이터를 버퍼로 push 하기 위한 파일 입출력 스래드를 기동하며 스래드는 검색이 완료될 때 같이 종료된다. 파일이 open되면 검색된 파일을 재배치할 5개의 파일을 생성하며 재배치할 GOP의 크기를 테이블에 기록한다. 이때 GOP의 크기와 번호를 기록할 파일은 별도의 디스크에 기록하고 각각의 생성된 4개의 파일은 디스크에 하나씩 생성이 된다. 입출력 스래드가 읽은 버퍼의 값을 검색하는 과정은 먼저 시퀀스 헤더를 찾으려면 파일 카운트는 다음 파일로 옮기고 생성된 GOP의 번호와 크기를 기록하는 것이다. 이러한 과정을 MPEG-1 데이터의 마지막 바이트를 읽을 때까지 반복하여 MPEG-1 파일을 재배치 한다. 분리된 시퀀스 헤더와 GOP를 그림 8과 같이 저장할 한다.

#### 4. 서비스 관리자

##### 4.1 수용 제어

VOD 서버와 같이 연속적인 미디어 데이터를 서비스 하는 시스템은 서버의 능력을 고려하여 사용자의 서비

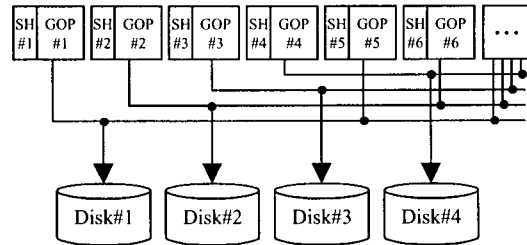


그림 8 재배치 분리도

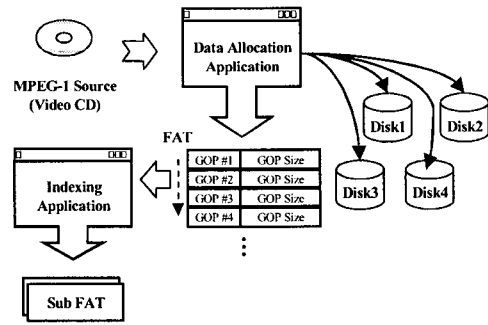


그림 9 서비스 전 작업과정

스 승인을 행하는 수용제어가 반드시 필요하다. 서버는 일단 서비스 요청을 수락하면 일정량의 자원을 수락된 클라이언트를 위해 할당하고 서비스 종료시까지 서비스 품질을 유지하여야 한다. 최근 MPEG 데이터를 서비스 하는 비디오 서버는 VBR(Variable Bit Rate)인 데이터의 특성을 반영하여 수용제어를 함으로써 다른 자원에 비해 상대적으로 병목인 디스크 시스템의 대역폭을 효과적으로 사용할 수 있다. 서비스 데이터의 특성을 이러한 연구는 서버 자원의 효율적 사용에 집중되어 통계적이고 경험적인 수치를 알고리즘에 반영하는 추세이다. 서비스를 받는 클라이언트는 서비스 수락 지연 시간보다 수락후의 QoS(Quality of Service)를 엄격히 요구하는 특성이 있다. 이것은 서버에게 실제 자원보다 초과하는 자원을 예약하여 수락시 지연을 고려하지 않고 수락후의 지연시간을 최대한 줄이기 위함이다.

##### 4.2 수용 제어 알고리즘

수용제어에서 고려할 시스템 자원은 버퍼공간과 디스크 대역폭이라고 말할 수 있다. 이는 서비스를 위한 시스템의 대역폭이 프로세스와 네트워크에 비해 상대적으로 병목인 디스크 대역폭에 의해 결정된다는 것을 의미한다. 수용제어에 필요한 서버의 능력은 이와 같이 디스크

대역폭으로 결정할 수 있으며 네트워크 서브시스템에 전송될 데이터를 저장할 버퍼 공간도 수용제어의 기준이 된다.

다음은 본 논문에서의 수용제어 알고리즘이다

$$\begin{aligned}
 &(\text{현재 서비스 대역폭}(i) + \text{새로운 요구 서비스 대역폭}(i) \leq \\
 &\quad \text{총 디스크 대역폭}) \\
 &\quad \&\& \\
 &(\text{현재 할당된 버퍼}(i) + \text{새로운 요구 버퍼량}(i) \\
 &\quad \leq \text{총버퍼량}) \quad (1)
 \end{aligned}$$

즉 모든 주기  $i$ 에서 총 디스크 대역폭과 버퍼량을 넘지 않는 새로운 서비스 요청 시에 새 사용자가 승인된다. VBR의 데이터 특성으로 인해 위 수식에서 현재 서비스 대역폭과 새로운 요구 대역폭을 결정하는 문제가 발생하게 된다. 본 논문에서는 이러한 문제를 해결하기 위하여 기존의 FAT 테이블에서 한 사용자에게 서비스를 하기 위한 새로운 요구 서비스를 GOP size로 결정하여 이 문제를 해결 하였다. 또한 VOD시스템 전체의 서비스 대역폭은 자원 상태표에 이를 기록하게 하여서 전체 사용자의 현재 서비스를 합산한 결과가 현재 서비스중의 대역폭이 된다.

$$N_{max} + \sum P_{max}^i \leq D_{max} \quad (2)$$

알고리즘은 위 식 2와 같이 표현할 수 있는데 이는 새로운 사용자의 버퍼 최대값( $N_{max}$ )과 현재 서비스 중인 사용자들  $i$ 버퍼 값들의 합( $\sum P_{max}^i$ )이 전체 디스크 대역폭( $D_{max}$ )을 넘지 않을 때 서비스가 수락됨을 나타낸다.

### 4.3 VOD 클라이언트 소프트웨어

VOD 클라이언트는 Microsoft사의 Window 95를 OS로 하는 개인용 PC에서 동작하도록 작성하였다. 클라이언트 소프트웨어는 그림 10과 같이 크게 사용자 인터페이스 모듈, 데이터베이스 관리 모듈, 네트워크 서비스와 버퍼 관리 모듈 그리고 MPEG재생 모듈 등 4가지로 나누어진다.

사용자 인터페이스 모듈은 영화선택과 VCR기능을 사용자에게 제공해 주며 프로그램의 최상위 모듈이 된다. 사용자가 영화를 선택하면 데이터베이스 모듈을 호출하여 영화 번호를 얻고 네트워크 서비스 모듈에 메시지를 보내 서버에 영화를 개방할 것을 알린다. 재생이나 빠른 전진 혹은 빠른 후진 등의 기능을 선택하면 네트워크 서비스 모듈에 메시지를 전송하여 서버에 이 기능이 수행되어야 함을 알리고, MPEG재생 모듈은 수신된 MPEG

데이터를 재생한다.

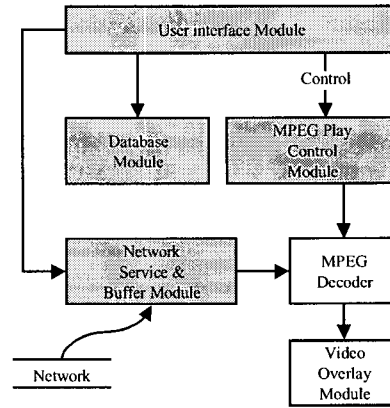


그림 10 사용자 프로그램의 모듈 구성

### 4.4 사용자 클래스

서비스 관리자는 새로운 사용자가 접속될 때 새로운 사용자 클래스를 생성한다. 이 클래스는 현재 사용자의 진행상태나 개인 정보를 참조할 수 있는 인덱스를 가지고 있다. 사용자 클래스는 다른 사용자 클래스와 같이 FAT 테이블을 공유하므로 절대적인 세마포가 필요하다. 이것은 서비스 관리자가 제어를 하여 사용자 클래스가 공유 테이블을 참조할 때 지연시간을 최대한 줄여준다. 본 논문에서 서비스 관리자는 MFC ver 6.0에서 지원하는 동기화 객체인 CSyncObject의 파생 클래스인

```

class User {
private:
    unsigned int index; // 새로운 사용자의 접속번호
    char *userID; // 사용자 account
    char *userName; // 사용자의 이름
    int State; // 사용자의 현 진행 상태
    long blocknum; // 블록 번호
    int disk; // 액세스 디스크 번호
    unsigned int port; // 사용자 접속 포트
    char *address; // 사용자 접속 address
    char *datetime; // 사용자 접속 날짜, 시간
    unsigned int indexDB; // 사용자 Database 레코드 번호
public:
    char * GetUserport();
    int GetCurrentState();
    int readLogFile();
    long GetblockNum();
}; // End of User Class
    
```

그림 11 사용자 클래스의 구조

CEvent, CMutex, CCriticalSection, CSemaphore를 이용하여 구현하였다. 이 동기화 클래스들은 CSingleLock 클래스를 이용하여 시그널 상태와 락 시그널 상태로 상태 변경을 통하여 스레드를 제어한다. 그림 11은 사용자 클래스이다.

5. 디스크 관리자

5.1 디스크 관리자의 역할

VOD 서비스가 갖는 가장 큰 특징은 대화형 서비스가 가능하다는 것이다. VOD 서비스는 사용자에게 대화성을 제공하기 위해서 VCR 기능을 제공하며 본 논문에서는 디스크 관리자가 VCR 기능을 제공하기 위한 연산을 한다. 디스크 관리자는 하나의 스레드로 구성되어 있으며 여러사용자의 디스크에 대한 접근을 관리한다. 디스크 관리자는 사용자 스레드가 의뢰한 특정 디스크의 블록을 GOP 크기만큼 읽어 달라는 요청을 받은 다음 디스크에 접근하여 읽은 데이터를 사용자 스레드의 버퍼에 채운다. 사용자 스레드가 버퍼에 데이터를 채운 다음에는 사용자 스레드의 네트워크 관리자 스레드에 전송하라는 이벤트 메시지를 전송한다.

5.2 VCR 연산 기능

VOD 서비스에서 기본적으로 제공하는 VCR 기능으로는 재생, 정지, FF(Fast Forward), FR(Fast Reverse)기능 등이 있다.

표 1 VCR연산과 시스템 서비스

연산분류	사용자 연산	시스템 연산
검색	고속전진(fast forward)	Read, randomaccess
	고속후진(fast reverse)	Read, randomaccess
	후진(reverse)	Read, sequential access
	재생(play)	Read, sequential access
제어	정지(stop)	Exit
	0점(jump)	Locationseek
	열기(open)	Fileopen
	닫기(close)	Fileclose

5.2.1 재생 모드

재생 모드는 사용자가 서비스를 요청하면 시작하는 가장 기본적인 VCR 연산으로 디스크에서 블록을 읽기 위하여 접근하는 디스크의 연산 수행 지연시간이 가장 적고 접근 알고리즘이 단순해야 한다. 그림 12는 사용자의 서비스 요구로 재생시에 디스크의 순차적인 접근이 되며 순차적으로 사용자에게 서비스를 하게 된다.

$$D_n = G_i \% N + 1 \quad (3)$$

◆ 재생 모드

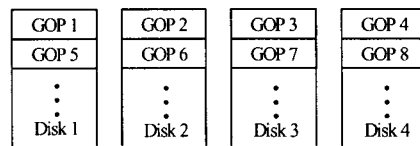
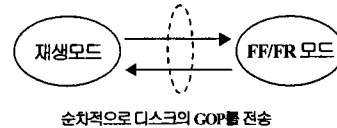


그림 12 재생 모드 VCR 연산

식 3은 해당 GOP번호가 어느 디스크에 저장되어 있는지 알수 있는 식이다. 서비스될 GOP 번호( $G_i$ )에서 디스크의 총 개수( $N$ )로 mod연산을 한 다음 1을 더하여 디스크의 번호( $D_n$ )을 알아낸다. GOP 번호로 FAT 테이블에서 GOP가 저장된 블록 size를 알수 있으며, 디스크 번호와 블록크기로 곱했을 더하여 해당 GOP의 위치를 얻을 수 있다[20].

◆ FF/FR 모드

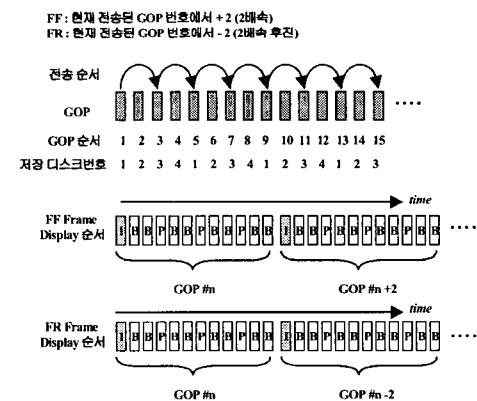


그림 13 2배속 FF/FR시 GOP 전송순서

5.2.2 FF/FR 모드

랜덤 액세스를 가능하게 하는 단위인 MPEG의 GOP를 이용함으로써 고속 전후진 기능을 제공할 수 있다. 이 방법은 임의 개수의 GOP를 건너 뛰면서 재생하는

방법으로, FF요구 시에도 네트워크 대역폭을 MPEG-1 시스템 스트림이 요구하는 1.5Mbit/sec와 동일하게 사용할 수 있고, 여분의 저장공간을 필요로 하지 않는 장점이 있다. 하지만 FR 요구 시에는 많은 문제점이 발생한다. 그러나 일반적으로 VOD 서비스를 받는 사용자는 후진보다는 Play, FF, Jump를 많이 사용한다. 그림 13은 2배속으로 전진과 후진을 서비스 할 때 접근되는 GOP의 순서와 디스크 번호이다. 만약 3배속 전 후진을 하게 될 경우에는 GOP가 3개씩 건너뛰면서 전송을 하게 된다. 그림 14는 2배속으로 전송 받은 클라이언트의 재생 순서이다. 정상적인 2배속 FF는 각 GOP에서 프레임 하나씩 건너뛴 서비스를 받아서 재생하여야 한다.

하지만 그림 14에서 나타난 2배속 서비스는 첫번째 GOP에서는 2배속 재생이 이루어지지 않고 있다. 두번째 GOP에서도 마찬가지이다. 결국 2배속 전진으로 재생이 될 경우 전체 재생 시간은 절반으로 줄어들지 몰라도 서비스의 질은 그렇지 않다.

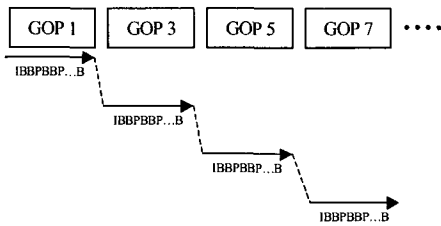


그림 14 2배속 FF

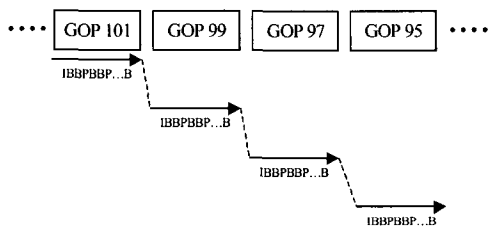


그림 15 2배속 FR

2배속 후진에 대한 그림 15에서 첫번째 서비스를 받는 GOP101은 정상적인 재생이다. 이것으로 GOP 내에서는 2배속 FF/FR재생이 불가능하며 2배속 FF/FR서비스의 문제점이 발생한다. 그러나 일반적으로 VOD 서버의 가장 중심적인 연산인 play가 대부분을 차지 하므로 본 논문에서는 다음과 같은 요건들을 만족하도록 구현하였

다.

- 요건1: 서버의 저장 공간을 최소화 해야한다.
- 요건2: FF/FR 요구 시에도 사용되는 네트워크 대역폭은 정상적인 스트림이 사용하는 대역폭과 같아야 한다.
- 요건3: 클라이언트의 부담을 최소화한다.

위에서 기술된 VCR연산으로 단점은 GOP 내에서 FF/FR 서비스를 하지 못하는 것이 있다. 장점으로는 2배속 또는 3배속 혹은 그 이상 FF/FR 연산 수행을 하더라도 VOD 서버에는 전혀 부하가 걸리지 않는다. 또한 클라이언트에서도 특별한 연산을 수행하지 않아도 서비스를 받을 수 있다. 다음 표 2는 FF/FR 연산 될 때 디스크에 접근되는 횟수를 나타낸다.

표 2 2배속/3배속 FF 접근시 디스크 참조 예

D <sub>1</sub>	D <sub>2</sub>	D <sub>3</sub>	D <sub>4</sub>	D <sub>5</sub>
1 <sup>2,3</sup>	2	3 <sup>2</sup>	4 <sup>3</sup>	5 <sup>2</sup>
6	7 <sup>2,3</sup>	8	9 <sup>2</sup>	10 <sup>3</sup>
11 <sup>2</sup>	12	13 <sup>2,3</sup>	14	15 <sup>2</sup>
16 <sup>3</sup>	17 <sup>2</sup>	18	19 <sup>2,3</sup>	20
21 <sup>2</sup>	22 <sup>3</sup>	23 <sup>2</sup>	24	25 <sup>2,3</sup>
26	27 <sup>2</sup>	28 <sup>3</sup>	29 <sup>2</sup>	30
31 <sup>2,3</sup>	32	33 <sup>2</sup>	34 <sup>3</sup>	35 <sup>2</sup>
36	37 <sup>2,3</sup>	38	39 <sup>2</sup>	40 <sup>3</sup>

표 2는 디스크가 5개인 시스템에서 2배속, 3배속 FF 재생시 접근하는 디스크의 참조를 나타낸 표이다. 이 표에서 2배속,3배속 FF재생시는 디스크 부하가 균형인 것을 알 수 있다. 그러나 디스크의 개수가 짝수개의 시스템일 때는 2배속 FF재생에서 디스크1과, 디스크3번에 부하 편중이 발생한다[21][22].

## 6. 네트워크 관리자

### 6.1 네트워크 관리자의 역할

VOD 서버와 클라이언트는 Winsock을 기반으로 하는 TCP/IP프로토콜을 사용한다. 모든 통신의 시작은 클라이언트에서 시작되며 서버는 이에 대해 응답하여야 한다. 네트워크 관리자는 하나의 쓰레드로 생성이 되며 전송할 버퍼에 데이터를 채운 후 디스크 관리자 쓰레드로부터 전송 메시지를 받은 다음 클라이언트에 전송한다. 네트워크 관리자 쓰레드는 사용자 쓰레드에 포함되어 있으며 각 사용자 쓰레드는 최소한의 네트워크 관리자 쓰레드를 포함한다. 네트워크 관리자 쓰레드는 사용자가 서비



스를 받는 중간에 VCR정보(Play, Stop, FF, FR)를 보내면 네트워크 관리자는 이를 분석하여 서비스 관리자에게 정보를 알려주고 디스크 관리와 네트워크 관리자에게 그에 따른 대응 처리를 한다. 네트워크 관리자는 현재 서비스중인 클라이언트의 네트워크 사용량을 자원 상태표에 기록을 한다. 이렇게 함으로써 현재 VOD 시스템이 사용하는 네트워크 패킷량을 알아낼 수 있다. 네트워크 관리자는 많은 양의 데이터를 짧은 시간에 전송하여야 하므로 큰 대역폭을 가져야 한다. 본 연구를 위하여 100MHz 대역폭을 갖는 이더넷 카드를 사용했다.

6.2 네트워크 관리자의 클래스 구조

그림 16은 네트워크 관리자의 클래스 구조이다. 네트워크 관리자는 사용자의 접속으로 인하여 동적으로 클래스가 생성된다. 그림 16에서와 같이 CFirstSocket이 특정 포트를 감시하고 있다. 새로운 사용자로 CPssSocket에 의한 새로운 사용자 클래스 CSecondSocket이 생성이 되며 이 클래스는 CMsg, CClientInfor클래스를 동반하여 생성한다. CMsg 클래스는 클라이언트에서 보내온 패킷 블록을 분석하여 서비스 관리자에게 정보를 보내며, CClientInfor 클래스는 사용자의 개인정보와 사용자의 행위를 기록한다.

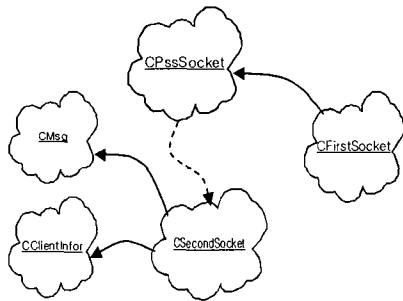


그림 16 네트워크 관리자의 클래스 관계

6.3 프로토콜

6.3.1 접속 시나리오

클라이언트와 서버 사이의 데이터 이동은 명령과 파라미터들로 구성된 데이터 헤더를 클라이언트에서 서버로 전송하면, 서버는 이에 대한 응답과 함께 데이터 길이를 담은 헤더를 전송하고 그 다음 실제 데이터를 전송함으로써 이루어진다. 그림 17은 서버/클라이언트간의 접속 및 서비스 시나리오이다.

우선 사용자는 VOD 서비스를 받기 위하여 사용자 프로그램을 동작한다. 클라이언트 측에서 서버에 접속을 시도하면 사용자의 고유 계정과 패스워드를 서버 쪽에

전송한다. 전송된 정보와 서버의 승인제어 알고리즘을 거쳐 수락이 되면 클라이언트는 ACK신호를 받는다. ACK 신호를 받은 클라이언트는 서버에게 서비스 목록을 요청한다. 서버는 클라이언트에게 서비스 목록을 전송한 후 서비스 대기 상태로 들어간다. 사용자는 서비스 목록 중에서 한가지를 선택한 후 서비스 요청을 한다. 서버는 서비스를 시작하기 위하여 자원을 할당받고 서비스를 시작한다. 클라이언트가 서버에 접속을 하여 사용자가 많을 경우 서버는 인접한 서버로 접속을 할 수 있도록 프로토콜을 확장하였다.

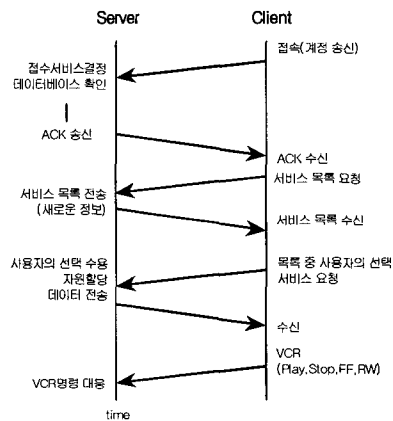


그림 17 서버/클라이언트 접속 서비스 요청 시나리오

6.3.2 프로토콜 형식

네트워크의 프로토콜 헤더는 총6바이트로 구성되고, 실제 동영상 데이터나 초기화 파일의 전송은 헤더 정보를 포함하므로 가변적이다. 그림 18은 프로토콜의 헤더의 블록형식을 나타낸다.

▶ 프로토콜의 블록 형식

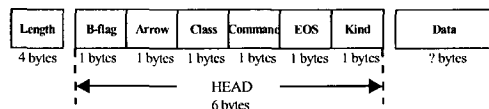


그림 18 프로토콜 블록 형식

첫 번째 바이트인 B-flag는 단일 블록 메시지인지 다중 블록 메시지인지를 나타낸다. 두 번째 바이트인 Arrow는 방향을 나타낸다. Arrow가 0x01이면 서버→

클라이언트, 0x80은 클라이언트→서버를 의미한다. Class는 모든 서비스의 종류를 5가지로 구분한다. Cmd는 각각의 서비스에 따라 달라질 수 있다. 가령 VCR 서비스는 Cmd에 play, stop, FF(Fast Forward), FR(Fast Rewind) 의 값을 가진다. Kind는 부가 서비스 정보를 나타내어 준다. VCR 헤더 정보는 사용자에게 의해서 결정이 되며, 서버는 이 정보를 통하여 클라이언트에 서비스를 한다.

7. 구현 및 실험

본 논문에서는 100Mbps에서 이더넷으로 구성된 LAN상에서의 구현을 통해 제안하는 방법이 VOD 서비스에 직접 적용이 가능하다는 것을 보였다. VOD서버는 Windows NT 4.0에서 MS Visual C++ 6.0[23][24]을 이용하여 구현하였고, 서비스 대상이 되는 MPEG 시스템 데이터와 VCR 데이터를 포함한다. 클라이언트는 Windows 95/98에서 구현하였으며, 서버로부터 전송된 MPEG 시스템 데이터를 실시간으로 재생한다. 그림 19는 서버의 실행 모습이다.

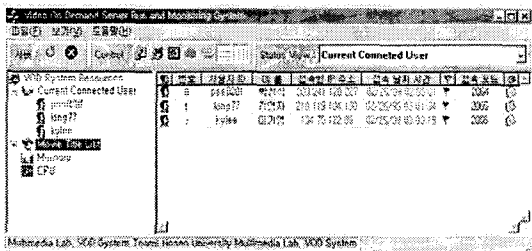


그림 19 서버의 실행 모습

7.1 시뮬레이션 환경

본 논문에서는 제안한 방법에 대한 시뮬레이션을 통해 성능 평가를 수행하였다. 시뮬레이션을 위해 사용된 영화는 모두 다섯개(M1..M5)이며, 각 영화는 모두 같은 수직, 수평크기 그리고 같은 프레임 및 같은 프레임 전송순서를 갖는다. 우선 구현된 서버에 접속한 클라이언트의 수에 따른 응답 시간을 측정하여 성능을 평가한다. 클라이언트는 가상으로 제작하여 5에서 35대까지 접속하면서 클라이언트의 수를 변화 시키면서 성능을 평가하였다. 가상 클라이언트는 실제 MPEG 데이터를 재생하는 기능은 제외하고 작성하였다.

7.2 성능 평가

작성된 가상 클라이언트에는 1초 주기로 서버에 약 1초 정도 분량의 데이터를 요구하여 서버에서 데이터가

도착한 시간을 측정하여 기록한다. 그림 20는 서버에 가상 클라이언트를 접속시켜 반응시간을 기록한 것으로 클라이언트의 수에 따라 평균 반응시간과 최대 반응시간을 나타낸 것이다. 35대의 클라이언트가 접속했을 경우에도 대부분의 경우 0.4초를 넘어가지 않은 것을 알 수 있다. 그러나 클라이언트 수가 20대 정도가 되면 지연시간이 급속도로 길어지는 것을 볼 수가 있다. 이로 인하여 다른 클라이언트의 QoS를 떨어뜨리게 된다. 본 논문의 시스템에서 지연시간이 길어지는 이유는 많은 클라이언트의 접속으로 인하여 디스크의 요청횟수가 많아지게 되기 때문이다. 이러한 문제점을 해결하기 위한 방법으로 디스크의 요청횟수를 줄여야 한다. 본 시스템에서는 사용자의 승인 제어서 현재의 접속자 수를 최대 25로 정하였다.

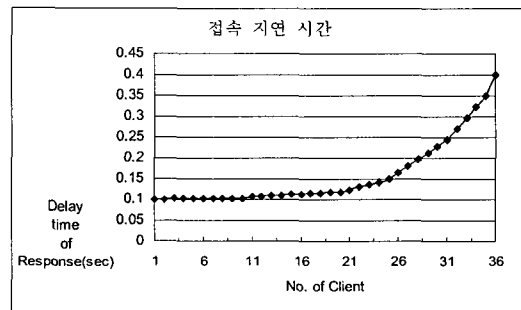


그림 20 클라이언트 접속 지연 시간

7.3 다른 시스템과의 성능 비교

VOD는 대용량의 저장장치와 버퍼공간을 필요로 하고 우수한 하드웨어 자원을 이용하여 구축하는 것이 일반적이며 분산 환경에서 서비스를 제공하기 위한 노력이 활발히 진행되고 있다. 현재 상용화된 VOD 시스템 또한 분산환경에서 개발된 고성능 시스템에서 Web을 통한 서비스를 하고 있으며, 본 논문에서 소개한 개인용 VOD 시스템과는 사양 및 성능면에서 객관적인 평가가 어렵다. 그리고 개발한 개인용 VOD 시스템의 특징으로는 독자적인 클라이언트 재생 프로그램과 VOD 서버, 대화형태로 서비스를 할 수 있도록 설계된 프로토콜, 그리고 성능대비 규모면에서 사용자에게 대한 서비스의 질적 향상등을 생각할 수있다.

8. 결론

VOD는 대용량의 저장장치와 버퍼공간, 네트워크 대역폭을 필요로 하고 우수한 사양의 하드웨어 자원을 이용

하여 구축하는 것이 일반적이다. 개인용 VOD 서버는 대용량 데이터를 취급하는 멀티미디어 서버의 한 종류이다. VOD 서버를 제작하기 위하여 우선적으로 고려해야 할 것은 사용자의 QoS를 만족시키는 것이다. 본 논문에서는 30대 정도의 클라이언트가 접속하여 VCR 서비스를 할 수 있는 개인용 VOD 서버를 구현하였다. 구현된 서버는 Windows NT 4.0 Server상에서 동작하며 NT의 기능을 최대한 이용할 수 있도록 하였으며 서버는 근거리 소수 사용자에게 다양한 서비스를 제공하기 위한 목적으로 설계 및 구현되었다.

본 시스템은 추후 사용자의 인터랙티브한 VCR을 고려한 수용제어 알고리즘이 필요하며 사용자의 요구 부하를 확률적으로 이용하여 알고리즘을 더 효율적으로 개선할 필요성이 있다. 또한 비디오 서버에 데이터를 저장 시에 구성할 데이터가 GOP 단위가 아닌 프레임별 세그먼트 기법으로 저장되어 GOP내에서도 고속 전 후진 재생이 가능하도록 하는 것이 필요하다. 또한 본 논문의 개인용 VOD서버는 저장시스템의 고장으로 인한 데이터의 복구를 고려하지 않으며 DSM-CC(Digital Storage Media Command and Control)로 구현된 다른 VOD 시스템과의 연동을 고려하지 않았다. 앞으로 이러한 문제점은 연구되고 개선되어야 할 것이다.

참 고 문 헌

[1] 이승용, 이호석, 홍성수, "VOD서버에서 연속매체의 고속재생을 위한 디스크 부하균형 정책", 한국정보처리학회 논문지, 제5권 제5호 1997년 5월.  
 [2] 재길, 김은환, 전문석, 이철희, "MPEG의 특성을 고려하여 VCR 기능을 제공하는 I(Interactive)VOD 서비스에 관한 연구", 한국정보과학회 추계 학술 발표대회 논문집, pp.299-300, 1997.  
 [3] Didier J.LeGall, "MPEG: A Video Compression Standard for Multimedia Application," Comm. of the ACM, vol.34, no.4, pp.47-58, April 1991.  
 [4] Joan L. Mitchell, William B. Pennebaker, Chad E. Fogg, Didier J.LeGall, "MPEG Video Compression Standard," CHAPMAN & HALL, 1997.  
 [5] Ralf Steinmetz, Klara Nahrstedt. "Multimedia: computing communication, applications," Prentice Hall, pp.113~165, 1995.  
 [6] 박한규, 차호정, "실시간 자원 관리 정책을 제공하는 VOD서버의 설계 및 구현", 한국정보과학회 추계학술 발표대회 논문집, pp.199-222, 1997.  
 [7] T.D.C Little and D. Venkatesh, "Prospects for Interactive Video-on-Demand," IEEE Multimedia Magazine, vol. 1, no.3, pp.14~24, 1994.  
 [8] S. V. Raghavan, Satish K. Tripathi, "Networked

Multimedia Systems," Prentice Hall, 1998.  
 [9] R.O.Onvural, "Asynchronous Transfer Mode Networks:Performance Issues," AII, pp.77~138, 1995.  
 [10] Yun,L.C and Messershmitt, D.G., "Architectures for Multi-Source Multi-User Compositing," ACM Siggraph Proceeding, 1993.  
 [11] Banu Ozden, Rajeev Rastogi, Avi Silberschatz, "On the Design of a Low-Cost Video-on-Demand Storage Server," Multimedia System, 4:pp.40~50, 1996.  
 [12] Guojun Lu, "Communication and Computing for Distributed Multimedia Systems," pp.59~304.  
 [13] Jason Nieh and Monica S.Lam, "The Design of SMART : A Scheduler for Multimedia Application," Technical Report CSL-TR-96-697, Computer System Laboratory, Stanford University, June, 1996.  
 [14] 박성식, 이호석, "멀티스트래드를 이용한 Personal VOD 서버의 설계", 한국정보과학회 춘계학술 발표회(A), pp.146-148, 1998.  
 [15] 박성식, 이호석, "멀티스트래드를 이용한 Personal VOD 서버의 구현", 한국정보과학회 춘계학술 발표회(A), pp. 184-186, 1999.  
 [16] 권택근, "연속 매체 저장 시스템에서 디스크 입출력 성능향상 기법", 박사학위논문, 서울대학교 컴퓨터공학과, 1996.  
 [17] 정세창, "그림으로 보는 최신 MPEG," 교보문고, 1995.  
 [18] 정세창, "그림으로 보는 응용 MPEG," 교보문고, 1995.  
 [19] 박성식, 이호석, "VOD시스템에서 디스크 부하균형을 위한 동영상 데이터의 재배치에 관한 연구," 한국정보과학회, HCI98 학술대회, pp. 135-138, 1998  
 [20] Nitin, "Authoring and Navigating Video in Space and Time," M.S Project Paper, May 29, 1996.  
 [21] O. Rose, "Statistical properties of MPEG video traffic and their impact on traffic modeling in ATM system," Technical Report, University of Wuerzburg. 1995.  
 [22] E.A. Hyden, "Operating System Support for Qos," PhD thesis, Wolfson College, University of Cambridge, February 1994.  
 [23] Herbert Schildt, "Teach Yourself C++," Osborne McGraw-Hill, 1994.  
 [24] Charles Petzold, "Programming Windows 95," Microsoft Press, 1996.



박 성 식  
 1970년 4월 10일생. 1997년 8월 호서대학교 컴퓨터공학과 학사 졸업. 1999년 8월 호서대학교 컴퓨터공학과 석사 졸업. 2000년 3월 ~ 현재 University of Tasmania in Australia Computer Science 대학원재학중. 관심분야는

Multimedia, VOD, Expert System



이 호 석

1979년 3월 ~ 1983년 2월 서울대학교  
전자계산기공학과 학사 졸업. 1983년 3  
월 ~ 1985년 2월 서울대학교 컴퓨터공  
학과 석사 졸업. 1989년 3월 ~ 1993년 8  
월 서울대학교 컴퓨터공학과 박사 졸업.  
1994년 3월 ~ 현재 호서대학교 컴퓨터  
공학부 교수. 관심분야는 멀티미디어, 영상압축, 영상처리,  
프로그래밍 언어