

복합 다단계 주문형 비디오 서버의 설계 및 구현

서 덕 록[†] · 강 대 혁^{††} · 김 수 정^{†††} · 이 원 석[†] · 이 정 수^{††††}

요 약

주문형 비디오 서비스는 컴퓨터 및 데이터 통신에 관련한 기반 기술의 발전으로 가능해진 멀티미디어 서비스로써 비디오 서버가 저장·관리하고 있는 디지털 동영상 데이터를 사용자의 요구에 따라 온라인으로 전송하여 단말장치에서 실시간으로 재생하는 서비스이다. 특히 대용량의 데이터를 효율적으로 저장해야 하는 서버는 주문형 비디오 서비스를 위한 필수적인 요건이며 현재 활발한 연구가 진행되고 있는 분야이다. 기존의 비디오 서버에 대한 연구는 동영상 데이터를 하드디스크에 저장하는 방식에 집중되어 왔으나, 하드디스크는 대용량 저장 장치에 비하여 안정성이 낮고 고가의 저장비용이 든다. 복합 다단계 주문형 비디오 서버는 대용량 저장 장치인 광자기 디스크 주크박스를 비디오 데이터의 주저장 매체로 사용하며, 데이터의 저장 방법 및 전송 스케줄링을 최적화함으로써 대용량 저장 장치가 갖는 단점을 보완하도록 제안된 주문형 비디오 서버 시스템이다. 본 논문에서는 복합 다단계 서버의 프로토타입을 설계 및 구현에 있어서 필요한 다양한 소프트웨어 모듈 및 구성을 설명하고 주문형 비디오 서비스를 위한 서버와 클라이언트의 프로토콜을 제안한다.

Design and Implementation of a Multi-level VOD Server System

Duk-Rok Suh[†] · Dae-Hyuk Kang^{††} · Su-Jung Kim^{†††} ·
Won-Suk Lee[†] · Jeong-Su Lee^{††††}

ABSTRACT

A Video-on-Demand(VOD) service is a multimedia service that is realized by the rapid advance of computer and data communication technologies. Basically, a VOD system is composed of a server and a number of clients. The server stores and manages a large amount of digital moving picture data. Each client sends an on-line request to the server and receives data for real-time displaying. Many researches are performed on a VOD server using hard disks for the permanent video data storage. However, a hard disk is less reliable and requires more storage cost than a massive storage device. Due to these reasons, a multi-level VOD system is proposed for using the jukeboxes of optical disks as the permanent video storage device. In this paper, we propose the necessary software modules and protocols between the server and its clients for the implementation of the multi-level VOD server.

※이 논문은 한국통신 '96 정보통신 기초연구과제 연구비에 의하여 연구되었음.

† 정 회 원:연세대학교 컴퓨터과학과

†† 정 회 원:LG전자 멀티미디어 연구소

††† 정 회 원:삼성전자 멀티미디어 본부

†††† 정 회 원:한국통신 전송기술연구소 VOD 연구팀

논문접수:1996년 10월 14일, 심사완료:1997년 2월 11일

1. 서 론

컴퓨터 및 통신 관련 기술의 발전은 이전까지 아닐로그 데이터를 기반으로 제공되었던 영상 서비스의 단점을 극복하여 사용자의 요구에 따라 서버가 응답하여 단말기까지 원하는 데이터를 실시간(real-time)으로 전송하며 데이터의 흐름을 제어할 수 있는 주문형 비디오 서비스(Video-on-Demand Service)를 가능하게 하였다. 주문형 비디오 서비스는 스크린, VCR 및 공중파 등을 통하여 단방향, 수동적으로 이루어져 왔던 영상매체 서비스를 비디오 서버가 저장·관리하고 있는 영화, 뉴스, 쇼평 그리고 교육 등 다양한 정보의 디지털 동영상 데이터를 사용자의 요구에 따라 온라인(on-line)으로 제공하여 퍼스널 컴퓨터 또는 셋톱박스(set-top box) 등의 단말장치에서 재생하는 서비스이다. 이전까지의 영상 매체들이 서비스 제공자에 의하여 스케줄링된 데이터만을 동시에 일괄적으로 받아들이는 구조를 지녔다면, 주문형 비디오 시스템은 사용자의 의사에 따라 제공되어 데이터가 가변적일 수 있고 향후 멀티미디어 데이터의 주문형 서비스의 초석이 되기 때문에 차세대 매체로써 활발하게 연구 및 개발되고 있는 분야이다.

주문형 비디오 서비스를 위한 서버 시스템은 MPEG1 이상의 고해상도 동영상 데이터를 ATM(Asynchronous Transfer Mode), ADSL(Asymmetric Digital Subscriber Line) 또는 Cable Network 등의 네트워크를 통하여 실시간으로 사용자에게 전송할 수 있어야 한다. 비디오 서버는 사용자에게 서버에 저장된 비디오 데이터들 중에서 원하는 비디오를 선택하고 재생하는 기본적인 기능 이외에 빨리 감기, 되감기, 멈춤 등의 임시 접근 제어(temporal access control) 기능을 제공해야 하며, 운영 센터에서는 비디오 데이터의 추가 및 삭제, 사용자 요금 정산 그리고 서비스 허가(admission control) 등의 서비스의 운용에 관련된 포괄적인 기능을 제공해야 한다[1, 14]. 비디오 서버 시스템은 이러한 요구 조건을 만족시키기 위하여 우선적으로 대량의 비디오 데이터를 효과적으로 저장·관리하여 사용자가 요구하는 비디오 데이터를 실시간으로 제공해야 한다. 따라서 데이터의 저장 매체 및 이에 적합한 서버 시스템의 설계가 비디오 서비스 시스템 개발을 위한 핵심적인 과제가 된다.

비디오 데이터는 압축되어도 다른 데이터형에 비해 그 크기가 매우 크다. 예를 들어 MPEG1의 평균 소비율 1.5Mbps를 고려할 때 2시간 분량의 영화를 저장하기 위해서는 1.5GB의 저장 공간이 필요하다. 대부분의 서버 시스템은 사용자의 요구에 대한 빠른 반응시간을 보장하기 위하여 비디오 데이터 저장 매체로 하드디스크를 사용하지만[13], 많은 비디오 데이터의 저장에 요구되는 대용량 주문형 비디오 서버에서는 데이터 저장을 위해 사용되는 디스크의 수가 늘어나게 되어 이들의 관리 및 데이터에 대한 신뢰성 보장에 대한 처리가 복잡해진다. 이러한 환경에서 디스크 입출력 전송률을 향상시키기 위하여 디스크 '스트라이핑(disk striping)[2, 18]과 RAID(Redundant Arrays of Inexpensive Disks)[3, 4]를 이용한 기술이 적용되었고, 일정한 속도로 데이터를 전송하기 위해 디스크 탐색 연산을 최적화 시키기 위한 여러 가지 실시간 디스크 스케줄링 기법이 제안[6, 7, 8]이 제안되었다. 또한 디스크 배열 내의 신뢰성 보장을 위해 패리티(parity) 방식[3]과 미러링(mirroring) 방식[9]이 제안되었으나 이는 디스크 용량과 대역폭의 손실을 가져온다.

복합 다단계 주문형 비디오 서버 시스템은 하드디스크에 비해 저렴한 비용으로 대용량의 비디오 데이터를 저장할 수 있는 광자기 디스크 주크박스(jukebox)를 사용한 비디오 서버 시스템으로 비디오 데이터의 주저장 장치로 광자기 디스크 주크박스를 이용하고, 현재 서비스되고 있는 비디오 데이터의 임시 저장 장치로 하드디스크를 사용한다. 광자기 디스크와 같은 대용량 저장 매체는 하드디스크 시스템에 비하여 데이터의 안정성과 신뢰성이 우수하나 상대적으로 느린 접근 시간(access time)때문에 실시간 서비스를 지원하기에는 부적합하다. 복합 다단계 주문형 비디오 서버는 데이터의 저장 구조 및 전송 스케줄링을 최적화함으로써 대용량 저장 장치가 갖는 단점을 보완하였다[5].

본 논문에서는 복합 다단계 주문형 비디오 서버 시스템의 구성에 요구되는 하드웨어 사양과 설계 방안에 대하여 고찰하고, 비디오 서버 시스템을 이용한 서비스에서 요구되는 소프트웨어 모듈 및 서버와 클라이언트의 프로토콜을 살펴보고 이의 설계와 프로토타입(prototype) 시스템의 구현에 대하여 논한다.

본 논문의 구성은 다음과 같다. 2장에서는 하드 디

스크를 사용하는 비디오 서버와 광자기 디스크 줍크박스를 주저장 매체로 사용하는 복합 다단계 비디오 서버 시스템에 대해서 개괄적으로 설명하며, 3장에서는 서버의 프로토타입 시스템의 하드웨어적인 구성에 대해서 기술한다. 4장에서는 서비스에 필요한 소프트웨어 모듈들의 특성과 서비스 스케줄링에 대하여 논하며, 5장에서는 이와 같은 구성에 따라 통신에 필요한 프로토콜 및 동작에 대해서 설명하고, 6장에서는 프로토타입 시스템에 대한 성능 평가를 하며 결론을 맺는다.

2. 복합 다단계 주문형 비디오 서버 시스템의 구조

현재 연구되고 있는 대부분의 비디오 서버들은 스케줄러, 데이터베이스 시스템, 그리고 화일 시스템으로 이루어진다[10, 16]. 스케줄러는 실시간으로 사용자가 요구한 자료를 최소한의 반응시간으로 전송할 수 있도록 데이터 전송을 스케줄링하며, 특히 입출력에 관한 우선권(priority)를 중심으로 연구가 되어왔다. 운영 센터의 역할을 담당하는 데이터베이스 시스템은 사용자에 대한 정보 및 비디오 화일 시스템에 저장된 비디오 데이터에 대한 메타데이터(metadata)를 저장·관리하는 기능을 갖는다. 화일 시스템은 데이터베이스 시스템에서 관리하기 어려운 비디오·오디오 등의 스트림(stream)형 데이터를 저장한다. 화일 시스템의 연구는 효율적으로 자료를 접근하여 제공할 수 있는 새로운 구조의 제안[4, 17]이나 디스크 스케줄링[6, 7, 13] 등에 관한 것이다.

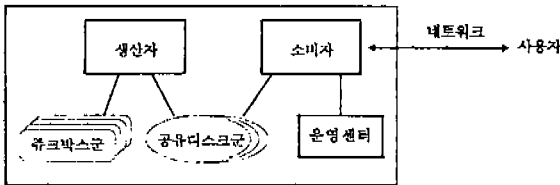
하드 디스크를 이용한 비디오 서버는 모든 비디오 데이터가 하드 디스크에 저장되므로 대용량 주문형 비디오 시스템을 구성하기 위해서는 많은 하드 디스크를 필요하므로 비교적 높은 구성 가격을 갖는다. 또한 하드 디스크의 잦은 에러에 대한 대비와 높은 입출력 처리량을 얻기 위하여 RAID기법을 이용할 경우 에러에 대한 복구 시간이 비교적 길며, 복구 처리시 시스템의 프로세싱 능력에 많은 영향을 줄 수 있어 실시간 처리를 완벽하게 보장할 수 없다. 그러므로 실시간 처리를 위해서는 시스템의 처리 능력 증진을 위한 능력을 항상 예약해 두어야 한다. 하드 디스크 시스템의 이러한 단점을 보완하기 위하여

Yurdar의 연구[10]에서는 화일 시스템에 하드 디스크와 테잎 라이브러리의 계층적 구조를 가지고 있어 자주 접근되는 데이터는 하드 디스크에 저장하고 자주 접근되지 않는 데이터는 테잎 라이브러리에 저장하는 방법을 제시하였으며, 데이터가 저장되는 매체의 구분은 데이터의 접근 빈도에 따라 이를 수학적 모델링에 근거하여 밝혔다. 그러나 이러한 구조에서는 자료의 접근 빈도가 확률적으로 정의되어 설계된 시스템이기 때문에 실제의 데이터 접근 양식이 가정된 패턴을 벗어나게 될 경우 사용자에게 대한 반응시간이 실시간으로 보장되지 못하며, 테잎 라이브러리의 전송 속도가 하드 디스크와 같은 임의 접근 장치(random access device)에 비하여 매우 느리다는 단점이 있다.

복합 다단계 주문형 비디오 서버 시스템에서는 방대한 비디오 데이터를 저장하는 주저장 매체로서 하드 디스크 대신 대용량의 데이터를 안정적으로 저장 가능한 광자기 디스크를 이용하고, 여러 개의 광자기 디스크를 광자기 디스크 드라이브에 기계적인 팔(robotic arm)을 이용하여 자동으로 갈아 끼워 주는 줍크박스를 사용한다. 이러한 구성에서는 줍크박스에 있는 광자기 디스크 드라이브의 수가 동시에 서비스 가능한 사용자의 수만큼 있지 않다는 가정 하에 줍크박스의 긴 디스크 교환 시간(disk swap time)때문에 실시간으로 여러 사용자에게 비디오 서비스를 수행하는 것이 불가능하다. 줍크박스의 긴 디스크 교환 시간을 보완하기 위해서 줍크박스에서 버퍼로 많은 데이터를 전송한 후 사용자에게 데이터를 전송하는 시간동안 다른 사용자를 위한 비디오 데이터가 저장된 광자기 디스크로부터 데이터를 로딩하게 된다. 또한 줍크박스에 접근하기 위한 디스크 입출력과 사용자에게 데이터를 전송하기 위한 네트워크 입출력은 실제 서비스에 있어서 시스템 전체의 과부하를 초래할 우려가 있기 때문에 각각의 역할을 담당하는 호스트를 독립적으로 구성하고 호스트들간의 데이터 교환은 스테이징 디스크(staging disk)[10]를 사용하며, 이 스테이징 디스크는 서비스되고 있는 데이터를 임시적으로 저장하는 역할을 동시에 수행하게 된다.

서버 시스템이 현재 서비스 중인 각 비디오 스트림에 대해 줍크박스로부터 일정 크기의 데이터를 로딩하여 스테이징 디스크로 버퍼링하는 단계를 데이터 생산과정으로 볼 수 있으며 이를 디스크 로딩이라고

정의한다. 또한 스테이징 디스크로부터 일정 크기의 데이터를 읽어 들여 사용자에게 전송하는 단계를 데이터 소비과정으로 볼 수 있으며 이 과정은 메모리 로딩과 네트워크 로딩 단계로 세분할 수 있다. 생산 과정을 담당하는 모듈을 생산자라고 명명하고 소비 과정을 담당하는 모듈을 소비자라고 명명한다. 스테이징 디스크는 두 모듈들에 의해 공유되어야만 생산된 데이터가 소비될 수 있으므로 공유 디스크라 명명한다. 따라서 전체 시스템의 대략적인 구성도는 (그림 1)과 같다.



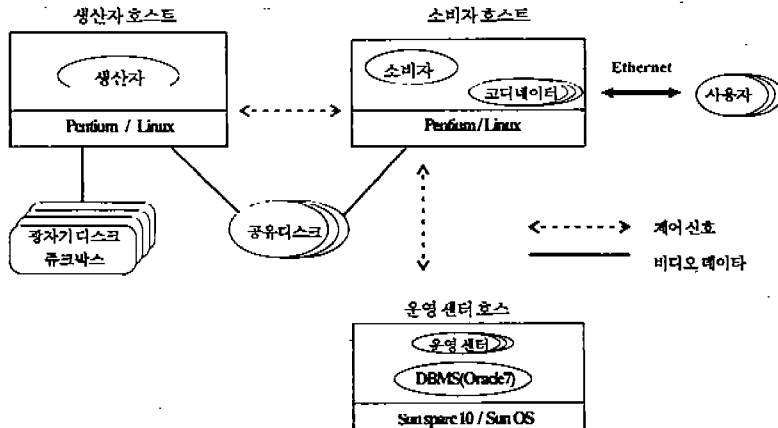
(그림 1) 복합 다단계 주문형 비디오 서버 시스템의 구조
(Fig. 1) Architecture of multi-level VOD server system

복합 다단계 주문형 비디오 서버는 데이터의 저장, 로딩 및 스케줄링에 있어서 튜브박스의 상대적으로 긴 디스크 교환 시간을 감추기 위한 기법이 사용되고 있다. 특히 서비스 초기 대기시간과 임시 접근 제어에 있어서 광자기 디스크 교환 시간 이상으로 사용자

가 데이터를 기다려야 하는데 이를 보완하기 위하여 공유 디스크에 비디오 데이터에 대한 초기 데이터와 중간 데이터를 프리패칭(pre-patching)시켜 저장함으로써 사용자의 서비스 요구 시간에 최단 시간에 서버가 반응하여 새로운 데이터를 전송할 수 있도록 서버를 구성하는 방안이 연구되었다.

3. 복합 다단계 주문형 비디오 시스템의 프로토타입의 설계

생산자와 소비자의 실질적인 하드웨어의 구성요건은 구축하고자 하는 서버 시스템의 특성과 동시에 서비스 가능한 사용자의 최대 수에 따라 퍼스널 컴퓨터, 중·대형, 병렬 컴퓨터 등의 다양한 형태로 구성이 가능하다. 튜브박스로부터 서비스되어 질 비디오 데이터를 읽어 공유 디스크에 쓰는 생산자와 공유 디스크로부터 사용자에게로 데이터를 전송하는 소비자가 각각의 독자적인 작업 수행에 간섭을 받지 않으려면 각각 독립적인 연산장치, 메모리, 시스템 버스 및 입출력 버스를 가지고 있는 것이 바람직하며, 이를 그림으로 나타내면 (그림 2)와 같다. 입출력 버스 방식의 하나인 스카시(SCSI)버스는 다수의 디바이스와 하나 이상의 호스트가 동시에 연결될 수 있어 공유 디스크를 사용하는 본 시스템에 대한 프로토타입의 구성에 적합한 입출력 장치로 사용 될 수 있다.



(그림 2) 복합 다단계 시스템의 프로토타입
(Fig. 2) Prototype of multi-level VOD system

서버의 프로토타입 시스템의 생산자 호스트와 소비자 호스트는 각각 펜티엄 프로세서를 가지는 퍼스널 컴퓨터 환경을 기반으로 구성하였다. 생산자 호스트에는 주크박스의 데이터를 정해진 프로토콜과 사용자의 요구에 동적으로 대응하는 스케줄링에 따라 공유 디스크에 쓰는 생산자 모듈과 서버 시스템 내의 다른 모듈과 제어 신호를 교환하기 위한 통신 모듈이 동작한다. 소비자 호스트에는 사용자의 서비스 엔트리를 기록하고 정해진 순서에 따라 공유 디스크의 데이터를 로딩하여 네트워크를 통하여 비디오 데이터를 전송하는 소비자 모듈과 통신 모듈이 동작한다.

생산자 호스트에서 관리하는 광자기 디스크 주크박스는 매우 고가이고 프로토타입 설계의 비용을 고려할 때, 여러 대의 주크박스를 사용하는 것이 비효율적이므로 하드 디스크에 테스트를 위한 비디오 데이터를 저장한 후, 사용자의 요구에 대하여 비디오 데이터를 검색·제공할 때 주크박스의 특징인 긴 디스크 접근 시간과 광자기 디스크 교환 시간을 모듈 내에서 시뮬레이션하는 방법을 취한다.

한편 생산자와 소비자 사이의 공유 디스크군은 양측에서 읽고 쓰기가 가능하도록 두 대의 컴퓨터에 스카시버스로써 하드 디스크를 연결하여 사용한다. 공유 디스크는 동일 입출력버스에서 생산자와 소비자의 작업이 이루어질 수 있도록 두 개 이상 존재할 수 있으며, 생산자와 소비자가 데이터 교환을 사용할 수 있는 대역폭은 공유 디스크군을 구성하는 디스크의 수에 비례하여 커진다. 공유 디스크에 데이터를 임시 저장하는 방법은 그 기법에 따라 오프라인 머징, 온라인 머징, 하이브리드 머징의 세가지 방식과 비머징 방식[11, 15]이 있다. 생산자와 소비자는 각각 최대 서비스 가능 사용자에 대해 연속적인 데이터 로딩 작업을 수행할 수 있도록 각 사용자마다 한번에 로딩되는 데이터 블록의 크기에 해당하는 메모리 버퍼를 가져야 한다.

소비자는 사용자와 네트워크로 연결되어 비디오 데이터를 전송한다. 네트워크는 다양한 형태를 사용할 수 있으며 프로토타입의 설계에는 비디오 데이터를 전송하기에는 대역폭이 적지만 전송 가격이 저렴하고 구성이 편리한 10Base-T 이더넷(ethernet)을 사용한다. 또한 운영 센터 호스트에는 비디오 서비스를 위하여 필요한 비디오 정보에 대한 관리, 사용자의

등록, 서비스 승인, 요금 산정 등의 기능을 갖는 운영 센터 모듈이 수행된다. 데이터베이스 서버로는 관계형 데이터베이스 관리 시스템인 Oracle 7을 사용한다.

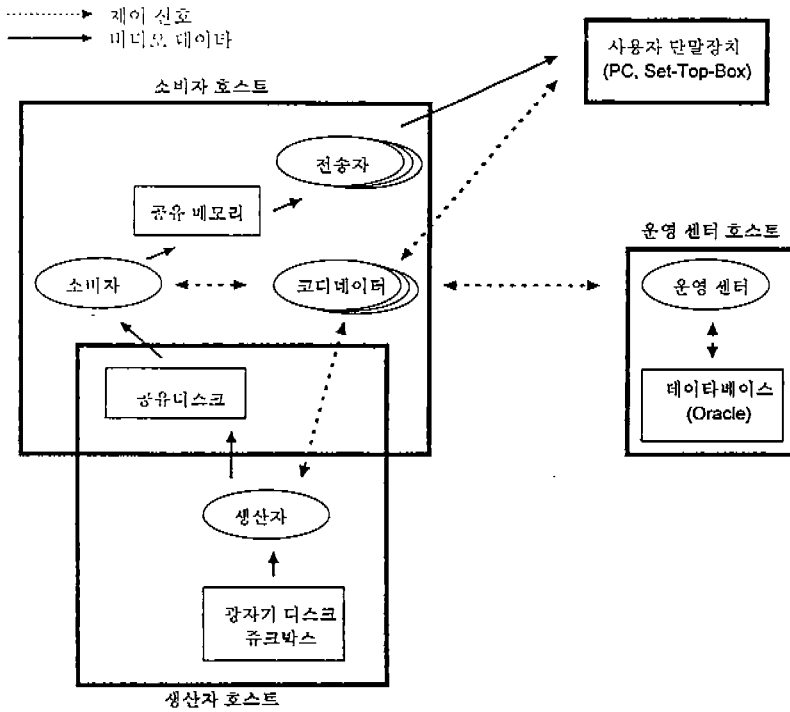
4. 복합 다단계 주문형 비디오 서버 시스템의 소프트웨어 구성

서버 시스템은 (그림 2)에서 살펴본 것과 같이 생산자, 소비자 그리고 운영 센터에 해당하는 세 개의 독립적인 호스트가 공유 디스크군과 네트워크로 연결된 하드웨어 구성을 갖는다. 본 장에서는 이와 같은 하드웨어 구성을 기반으로 복합 다단계 서버 시스템의 비디오 서비스에 필요한 소프트웨어 모듈들을 설계하고 모듈간의 상호 작동에 대해 자세히 설명한다.

서버 시스템의 소프트웨어 모듈은 주크박스 안에 있는 비디오 데이터를 공유 디스크에 임시 저장하는 생산자와 공유 디스크에 저장된 데이터를 실시간으로 사용자에게 전송하는 소비자, 소비자의 관리하여 비디오 데이터의 전송을 담당하는 전송자, 비디오 데이터 및 사용자 정보 등의 비디오 서비스를 위한 정보를 데이터베이스에 저장·관리하고 있는 운영 센터, 그리고 사용자로부터 전달되는 다양한 제어 신호를 해석하여 서버 시스템의 운영이 사용자의 요구에 따라 적절히 수행될 수 있도록 생산자, 소비자 및 운영 센터에 분배하는 코디네이터(coordinator)로 구성된다. (그림 3)은 복합 다단계 주문형 비디오 서버 시스템의 하드웨어 및 소프트웨어 구성에 따른 전체 시스템의 구성을 보여준다.

4.1 코디네이터

코디네이터는 서버와 클라이언트간의 초기 접속 및 제어 신호를 관리하며, 서버 시스템 내에서 각 모듈간의 통신을 총괄하는 모듈이다. 사용자가 정해진 포트를 통하여 서버에게 비디오 서비스를 요구하면 새로운 코디네이터 모듈이 이에 응답하여 생성되며 서비스 초기화를 위한 일련의 작업을 수행한다. 즉 사용자 식별자를 받아들여 서버에의 로그인 작업을 수행하며, 서버가 해당 사용자에게 비디오 서비스를 제공할 수 있는 정상적인 상태인지 시스템 내의 각 모듈과 통신하여 확인한다. 최종적으로 코디네이터는 서버에서 제공할 수 있는 비디오 데이터의 목록



(그림 3) 복합 다단계 주문형 비디오 시스템의 소프트웨어 모듈

(Fig. 3) Software module for multi-level VOD system

및 이에 대한 상세 데이터를 사용자에게 전송한다. 사용자가 원하는 비디오를 선택하면 선택된 비디오 데이터에 대한 정보를 코디네이터에게 보내고 코디네이터는 이를 생산자, 소비자 그리고 운영 센터에 전달하여 서비스 시작을 알린다.

사용자의 정지 및 재생 등의 조작은 비디오 데이터

전송채널과는 다른 채널을 통하여 코디네이터에 전달되며 코디네이터는 제어 신호의 처리를 위하여 생산자 프로세스와 소비자 프로세스에게 알리는 역할을 수행한다. 이상에서 기술한 코디네이터의 구조는 다음과 같이 기술된다.

AcceptConnection()에서는 UDP 소켓(socket)을 통하여 사용자의 접속을 처리한다. 사용자로부터의 접속은 현재 서버에 접속되어 있는 사용자수와 생산자의 서비스 스케줄링 등의 서버의 상태에 따라 새로운 서비스의 허가 여부가 결정이 되며, 접속이 허가 되면 해당 사용자의 서비스를 담당하는 프로세서를 생성시켜 복수의 사용자가 동시에 접속하는 것을 허락한다. 서비스를 위한 초기화 작업을 거친 후 LoginCenter()에서는 사용자에게 부여된 고유 식별자로 운영 센터의 데이터베이스에 서비스 승인을 요구하며 서버 내의 비디오 데이터에 대한 정보를 얻어 사용자에게 전송

```

while (1) {
    AcceptConnection(); // 사용자로부터의 접속
    if ( fork == 0 ) {
        Initialization();
        LoginCenter(); // 서비스 시작을 위한 운영 센터 로그인
        while (1) {
            RequestFromUser(); // 사용자로부터의 제어 요구
            DoService(); // 요구 처리
        }
    } else
        continue;
}
    
```

(그림 4) 코디네이터의 접속처리

(Fig. 4) Connection Processing of Coordinator

한다. 이후 사용자의 서비스 시작 제어 신호를 받아 비디오 서비스가 동작한다. 이러한 서비스 허가 여부의 결정 기능은 운영 센터에서 수행되도록 설계할 수도 있지만, 제어 신호와 비디오 데이터 전송 채널 관리의 일원화를 위하여 코디네이터에서 전담하여 처리하도록 구성하였다.

4.2 생산자

생산자는 현재 서비스 중인 비디오 스트림 각각에 대해 일정한 크기의 데이터 블록을 큐크박스에서 순차적으로 디스크 로딩하고 이러한 디스크 로딩 작업이 반복되어 수행됨으로써 각 사용자에게 연속적인 비디오 데이터를 전송하게 된다. 이 과정은 크게 두 단계로 구성되는데 큐크박스로부터 생산자의 메모리로 로딩하는 단계(큐크박스-메모리 로딩)와 생산자 메모리의 데이터를 공유 디스크에 쓰는 단계(메모리-공유 디스크 로딩)이다.

모든 사용자에게 하나의 일정 크기의 데이터 블록을 로딩하는 주기를 로딩 주기 T_L 로 정의한다. 현재 로딩 주기에서 생산자가 로딩한 데이터는 다음 로딩 주기에서 소비자에 의해 접근되어야 하므로 한 로딩 주기가 끝나면 생산자와 소비자간에 연결된 공유 디스크를 교환하는 작업이 수행된다. (그림 4)는 한 로딩 주기에서 수행되는 디스크 로딩 작업들을 보여준다.

연속된 디스크 블록이 전송되어야 하므로 다음과 같이 표현할 수 있다.

$$T_L < D_b / S_t$$

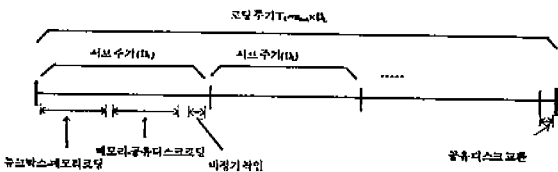
위의 식에서 S_t 는 데이터 소비율로써 MPEG1의 경우 평균 1.5Mbps이다.

한 주기에서 반복되는 비디오 데이터 로딩 작업의 새로운 사용자의 요구를 받거나 중단하기 위한 처리 등의 비정기적(aperiodic) 작업은 큐크박스-메모리 로딩과 메모리-공유 디스크 로딩 이후에 독립적으로 처리되며 로딩 주기가 끝나면 소비자와 공유 디스크를 교환하는 작업이 뒤따른다. 생산자의 하드웨어적인 구성이 다수의 프로세서를 갖고 더블 버퍼링(double buffering)을 지원한다면 앞서 기술한 두 단계가 서로 중복되어 수행될 수 있지만 퍼스널 컴퓨터를 기반으로 구현된 프로토타입에서는 두 로딩 단계가 순차적으로 수행된다. 이러한 생산자의 스케줄링 알고리즘은 다음과 같이 기술된다.

```

while (1) {
    for (i=0; i < n_max; i++) {
        jk_m_loading(); // 큐크박스-메모리 로딩
        m_sd_loading(); // 메모리-공유 디스크 로딩
        aperiodic_jobst(); // 비정기적 작업
    }
    swap_shreddisk(); // 공유 디스크 교환
}
    
```

(그림 6) 생산자의 스케줄링 알고리즘
(Fig. 6) Scheduling Algorithm of Producer



(그림 5) 생산자의 스케줄링 모형
(Fig. 5) Scheduling Model of Producer

서버의 환경으로 정해진 최대 사용자수 n_{max} 각각의 비디오 데이터에서 한번에 로딩되는 데이터의 크기를 디스크 블록 D_b 라고 정의하면 로딩 주기는 n_{max} 개의 D_b 을 로딩하는 시간 D_L 의 슬롯(slot)으로 구성된다. 로딩 주기의 실시간 제약 조건은 각 스트림마다 디스크 블록이 모두 소비되기 전에 해당 스트림의

비디오 데이터의 저장 방법은 큐크박스 내의 광자기 디스크에 비디오 데이터를 기록하는 방식에 따라 분산 저장 방식과 비분산 저장 방식으로 나누어진다. 전자는 비디오 데이터가 광자기 디스크에 연속적으로 기록되는 방식으로써 저장 방식이 간단 명료하며, 비디오 데이터의 삽입·삭제 등의 관리가 용이하다. 후자는 하나의 비디오 데이터를 디스크 블록 크기로 나누어 서로 다른 광자기 디스크에 분산시켜 저장시키는 방법으로 비디오 데이터의 유지·관리가 비분산 저장 방식에 비하여 복잡한 반면 사용자의 새로운 서비스 요구에 대한 서비스 허가 처리가 단순하고, 큐크박스의 입출력 대역폭을 최대한 활용할 수 있다는

장점을 있다.

4.3 소비자

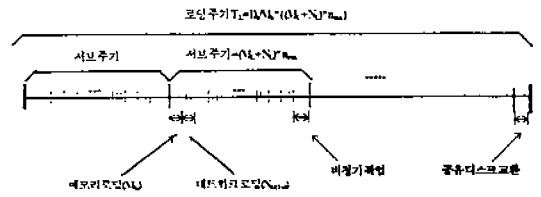
소비자 모듈에서는 공유 디스크로부터 비디오 데이터를 로딩하여 네트워크를 통하여 사용자에게 전송하는 작업을 수행한다. 사용자가 서버에 접속하여 비디오 서비스가 시작되면 코디네이터는 소비자에게 사용자의 네트워크 정보를 보내며, 소비자는 이 정보에 따라 사용자와 비디오 데이터 전송을 위한 TCP 소켓 채널을 확보한다. 이 과정에 채널 확보에 실패하면 코디네이터에 접속에 실패하였음을 알리고 서비스가 정상적으로 종료되기 위한 절차를 밟도록 한다.

소비자 모듈은 서비스 초기에는 공유 디스크에 프리패칭되어 있는 초기 비디오 데이터를 전송하며 이후 생산자와의 통신을 통하여 유크박스에 저장된 데이터가 공유 디스크에 로딩된 것을 확인한 후 공유 디스크의 데이터를 로딩하여 사용자에게 전송한다.

소비자 모듈의 알고리즘에 대한 설계에 있어서 가장 큰 제약 조건은 각 사용자에게 실시간으로 필요로 하는 데이터를 전송하여야 한다는 것이다. 이를 위하여 소비자 모듈은 각 사용자에 대한 단위 시간 동안에 데이터 전송 요구량과 실제 데이터 전송률을 알아야 한다. 따라서 단위 시간을 최대 사용자 수 만큼으로 배분하고 현재 사용자 수가 최대 사용자 수보다 적을 때에는 주기적으로 유휴(idle)상태에 있게 된다. 소비자의 메모리 로딩 작업은 각 사용자의 입장에서 보았을 때 많은 양의 데이터를 순간적으로 보내고 많은 시간을 다른 일에 사용하는 것보다는 적은 양의 데이터를 보내더라도 자주 보내는 것이 네트워크의 측면에서나 사용자의 버퍼 관리 측면에서 유리하다.

소비자에서도 생산자에서의 디스크 로딩 작업과 같이 현재 서비스하고 있는 사용자들에게 일정 크기의 데이터를 순차적으로 전송하여 모든 사용자에 대해서 비디오 재생을 위한 실시간 조건을 만족하도록 스케줄링 되어 진다. 소비자가 각 비디오 스트림에 대해 공유 디스크로부터 한번에 메모리로 읽는 크기를 메모리 블럭이라고 정의하고 이 작업을 메모리 로딩이라고 하며, 이를 보다 작은 크기의 네트워크 블럭으로 나누어 네트워크 로딩 작업을 수행함으로써 사용자에게 전송하게 된다.

하나의 메모리 블럭을 로딩하는데 걸리는 시간을 M_L , 모든 사용자에 대한 네트워크 블럭들을 네트워크 로딩하는데 걸리는 시간을 N_L , 공유 디스크 교환 주기 T_L 에서 i 번째 일어나는 메모리 로딩을 M_{Li} , i 번째 일어나는 모든 사용자를 위한 네트워크 로딩을 N_{Li} , 각 N_{Li} 중에서 j 번째 일어나는 한 사용자를 위한 네트워크 로딩을 N_{Lij} 로 정의하면 소비자의 전체 로딩 주기 T_L 에 대한 스케줄링 모형은 (그림 5)로 나타내어 진다. 단위 로딩 주기에 로딩되는 디스크 블럭의 수는 n_{max} 이며 각 디스크-블럭은 D_b/M_b 번 메모리 로딩되어 M_b/N_b 번 네트워크 로딩되어 사용자에게 전송되므로 한 로딩 주기에서 메모리 로딩 작업이 수행되는 슬롯은 $(D_b/M_b) * n_{max}$ 개로 나타내어지며, 네트워크 로딩 작업이 수행되는 슬롯수는 $(D_b/N_b) * n_{max}$ 개로 표현된다.



(그림 7) 소비자의 스케줄링 모형 (Fig. 7) Scheduling Model of Consumer

이에 대한 작업을 의사 코드(pseudo code)로 나타내면 다음과 같이 기술된다.

```

while (1)
{
    for( r=0; r < D_b/N_b; r++)
    {
        for( j=0; j < n_max; j++)
        {
            // j번째 사용자의 i번째 슬롯의 메모리 로딩
            memory_loading(j);
            for( k=0; k < M_b/N_b; k++)
            {
                // i번째 사용자의 네트워크 로딩
                network_loading(k);
            }
            // 비디오 데이터 전송 이외의 작업
            aperiodic_job(r);
        }
        swap_shareddisk(r); // 공유 디스크 교환
    }
}
    
```

(그림 8) 소비자의 스케줄링 알고리즘 (Fig. 8) Scheduling Algorithm of Consumer

4.4 운영 센터

운영 센터는 사용자의 등록과 승인, 비디오 정보 검색과 현황 유지, 서비스 통계 그리고 요금 청구 등의 기능을 수행하며, 대용량의 주문형 비디오 서비스 환경에서는 많은 사용자와 비디오 데이터에 대한 정보를 저장하면서 빠른 반응 시간을 지원하려면 주기억장치 데이터베이스 시스템(Main Memory Database System)[12]의 활용이 필요하지만, 본 연구에서 구현하는 프로토타입 시스템에서는 관계형 데이터베이스 관리 시스템을 사용하여 관리하도록 구현하였다. 운영 센터 모듈은 데이터베이스에 저장된 고객 정보와 서버의 상태를 참조하여 서비스를 승인하며, 사용자의 비디오 정보 검색을 지원하고, 수행된 서비스에 대한 정보를 갱신하는 역할을 수행한다. 이러한 사용자 데이터 갱신 작업은 사용자와의 통신 채널을 고려하여 모두 코디네이터를 통하여 이루어지며 사용자와 직접 통신하지 않도록 설계하였다.

5. 클라이언트와 서버, 서버 모듈 내에서의 통신

주문형 비디오 서버와 클라이언트는 비디오 데이터를 안정적으로 서비스 받기 위하여 쌍방의 신뢰성을 보장하는 프로토콜의 정의가 필요하다. 본 시스템의 구현에 사용한 프로토콜은 인터넷 표준 프로토콜인 TCP/IP를 기반으로 하여 응용 계층에서 정의하였으며 특히 다음과 같은 원칙하에 설계되었다. 첫째, 하나의 클라이언트와 서버간의 신호 전달을 위한 채널은 제어 신호를 위한 채널과 데이터 전송을 위한 채널이 각각 최소 하나씩 필요하다. 이것이 보장되지 않으면 사용자가 데이터를 전송받는 도중 서버에게 제어 신호를 전송할 수 없기 때문이다. 둘째, 신호의 유실을 막기 위하여 신호를 받는 측의 준비가 되어 있지 않을 경우에는 신호를 보내어서는 안된다. 이러한 원칙에 따라 본 절에서는 서비스의 단계를 서비스 초기화 단계, 비디오 데이터 전송 단계, 비디오 데이터 전송 중단 단계, 임시 접근 제어 단계 및 서비스 종료 단계로 나누어 제어 신호의 흐름을 중심으로 설계된 프로토콜을 설명한다.

5.1 제어 신호의 구조

서버와 사용자 단말간의 제어 신호 교환과 서버 시

스템 내의 각 모듈간의 통신을 위한 제어 신호는 서비스 초기화를 위한 패킷(packet), 비디오 정보를 위한 패킷, 제어 신호 패킷, 에러 신호 패킷, 그리고 비디오 데이터 패킷으로 나누어져 사용된다. 이 중에서 가장 중요한 제어 신호 패킷은 다음과 같이 이루어진다.

client_id	control_id	control_mode	offset
4 bytes	4 bytes	4bytes	4bytes

사용자 식별자(client_id)는 단말장치마다 부여된 고유한 사용자 번호를 나타낸다. 서버는 이 식별자를 통하여 데이터베이스에 있는 사용자 정보를 검색하고 서비스 정보를 유지·관리한다. 제어 신호 식별자(control_id)는 사용자가 발생시킬 수 있는 각 제어 신호마다 수치 코드를 부여하여 서버와 사용자간 뿐만 아니라, 서버 내부 모듈간의 통신에서 사용자의 요구를 사용자 식별자와 제어 신호 식별자의 조합으로 표현하여 서버에서 수행 중인 작업의 식별자로 사용된다. 제어 모드(control_mode)는 비디오 재생에 있어서 사용자의 임시 접근 제어 요구를 표현하기 위한 필드로 {Start, Stop, FF_begin, FF_end, REW_begin, REW_end, Pause, Continue} 중에 하나의 값이 된다. 오프셋(offset)은 초기화 단계의 비디오 식별자와 임시 접근 제어 단계의 시작과 종료 사이의 시간을 전송하는 필드로 사용된다. 이는 사용자가 버튼을 누르고 있는 시간에 따라 빨리 감기와 되감기의 최종적인 위치를 서버에게 알리기 위하여 사용된다.

5.2 서비스 초기화 단계

사용자가 고유한 식별자로 지정된 서버의 포트에 접속을 시도하면 코디네이터는 이를 받아 들어 주크박스, 광자기 디스크, 공유 디스크 상태, 프로세서의 프로세싱 능력의 정적인 상태를 고려하여 정적 서비스 허가를 수행하여 새로운 사용자에게 서비스 가능할 경우 접속을 허락한다. 접속된 사용자는 서버가 가지고 있는 비디오 정보에 대한 질의를 보내고 코디네이터는 이를 운영 센터에 보낸다. 운영 센터는 자신의 데이터베이스로부터 질의에 맞는 비디오 정보를 추출, 코디네이터를 통해 사용자에게 전달한다. 비디오 정보에 대한 질의와 응답은 필요에 따라 여러

번 반복할 수 있으며, 사용자가 얻은 비디오 정보로 원하는 비디오를 선택하면 스케줄링 알고리즘에 의한 동적 서비스 허가를 수행하여 승인이 되면 비디오 데이터 전송 단계로 이어진다. 동적 서비스 허가에 있어서는 서버가 최대 사용자수보다 적은 사용자를 서비스를 수행하고 있더라도 특정 데이터를 집중적으로 접근하는 서비스 패턴을 갖고 있다면 새로운 사용자에게 서비스를 할 수 없도록 한다.

5.3 비디오 데이터 전송 단계

비디오 데이터 전송 단계에서 제어 신호의 흐름은 (그림6)과 같다. 사용자가 사용자 단말을 통해 사용자 식별자와 함께 시작 신호를 서버에 보내면(1) 코디네이터가 이 신호를 받아 제어 신호 식별자를 덧붙여 소비자에게 전달하며(2) 소비자는 이를 바탕으로 사용자에게 데이터를 전송하기 위한 채널을 확보한 후 결과를 코디네이터에게 알린다(3). 생산자, 소비자 및 사용자 단말까지의 데이터 전송 경로가 생성되었으므로 이를 사용자 단말에게 알리고(4), 사용자 단말은 비디오 데이터의 수신이 가능함을 서버에 알린다(5). 코디네이터는 운영 센터에 서비스가 시작되었음을 알리는 동시에 생산자에게 서비스 시작을 알린다(6). 생산자는 디스크 로딩을 스케줄링하고 소비자에게 공유 디스크에 저장된 선택된 비디오 데이터의 시작 데이터를 전송할 것을 알린다(7). 시작 데이터에 대한 서비스 시작 신호를 받은 소비자는 확보한 네트워크

채널을 통해 데이터의 전송을 시작하고(8) 생산자는 해당 비디오 스트림의 시작 블록 이후의 데이터를 디스크에 로딩한다(9).

5.3 비디오 데이터 전송 중단 단계

사용자가 서비스 중지 신호를 서버에게 보내면 코디네이터는 소비자 및 운영 센터에게 중지 신호를 분배한다. 소비자는 해당 사용자에 대한 네트워크 로딩과 공유 디스크로부터의 메모리 로딩 작업을 중단하고 자신의 서비스 목록에서 해당 사용자를 삭제하며, 생산자에게 이를 알린다. 이 신호를 받은 생산자는 자신의 서비스 목록에서 사용자를 삭제한 후 코디네이터를 통하여 서비스가 중단되었음을 사용자에게 알린다. 데이터 전송 중단 단계에서 이러한 복잡한 과정을 거치는 이유는 사용자 단말이 받을 준비가 되어 있지 않은 데이터에 대한 불필요한 전송을 막고, 서버 시스템내의 데이터 전송 중단에 대한 신뢰성을 보장하기 위함이다.

5.5 임시 접근 제어 단계

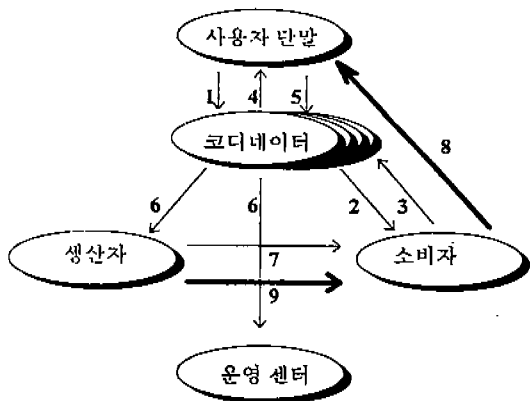
임시 접근 제어에서 일어나는 제어 신호는 빨리 감기, 되감기, 일시 정지 및 계속이 있으며 빨리 감기와 되감기는 사용자가 단말을 이용하여 해당 버튼을 눌렀을 때와 떼었을 때를 각각 시작과 끝의 쌍이 되며, 일시 정지와 계속이 하나의 쌍이 된다. 즉 다음 <표 1>과 같이 임시 접근 제어 신호를 분류할 수 있다.

<표 1> 임시 접근 제어 신호

<Table 1> Control Signal for Temporal Access Control

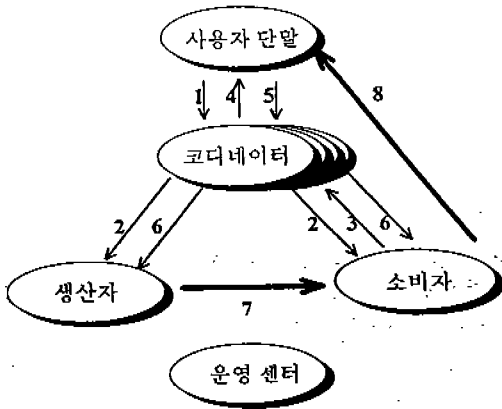
	시작	끝
빨리 감기	FF_begin	FF_end
되감기	REW_begin	REW_end
일시 정지와 계속	Pause	Continue

각 신호의 시작은 현재 데이터를 로딩하는 것을 중지하는 것을 의미하며 끝은 새로운 데이터의 로딩 작업이 시작되는 것을 의미한다. 이와 같은 임시 접근 제어 시의 사용자 단말과 서버간의 제어 신호의 흐름은 (그림 7)로 표시될 수 있다. 사용자가 임시 접근 제어의 시작 신호를 보내면(1) 코디네이터는 생산자와



(그림 9) 비디오 데이터 전송 단계의 제어 신호 흐름
(Fig. 9) Flow of Control Signal on Data Transmission Phase

소비자에게 데이터 로딩을 중지할 것을 알린다(2). 소비자가 데이터 전송을 중지하고 이를 코디네이터를 통해 사용자 단말에게 알리게 되며(3,4), 이 신호를 받을 때까지 사용자 단말은 데이터 채널을 통해 전송되어 오는 데이터를 모두 원하지 않는 데이터로 간주하여 폐기시킨다. 사용자 단말에서 데이터의 읍셋과 함께 임시 접근 제어 끝 신호를 서버에게 보내면(5) 코디네이터는 이를 생산자와 소비자에게 보내고(6), 생산자는 사용자가 원하는 데이터를 튜브박스로부터 디스크 로딩하며 소비자는 공유 디스크에 미리 프리패칭된 중간 데이터 블록들 중에 사용자 단말에서 보내온 읍셋과 가장 가까운 중간 데이터 블록을 메모리 로딩하여 사용자에게 전송한다. (7,8).



(그림 10) 임시 접근 제어 시의 제어 신호 흐름
(Fig. 10) Flow of Control Signal on Temporal Access Control phase

5.6 비디오 데이터 전송 종료 단계

서버로부터 보낼 데이터가 모두 전송되었다고 해서 서버가 독자적으로 서비스 정보를 삭제시킨다면 사용자는 이를 인식하지 못하고 임시 접근 제어를 시도할 수 있다. 이와 같은 상황은 서비스 정보가 삭제된 이후이므로 서버가 해당 임시 접근 제어에 대한 적절한 처리를 수행할 수 없는 상태가 된다. 따라서 생산자와 소비자는 비디오 데이터가 모두 전송되더라도 서비스 정보를 삭제시키지 않고 서비스 종료 제어권을 사용자에게 일임하는 것이 타당하다. 즉 데이

타 전송 채널에 전송 종료를 나타내는 신호를 보내 이를 인식한 사용자 단말이 접속 종료 신호를 보내어 서버와의 접속을 끊거나 비디오 데이터 전송 준비 단계로 되돌아갈 수 있도록 구성하는 것이 바람직하다.

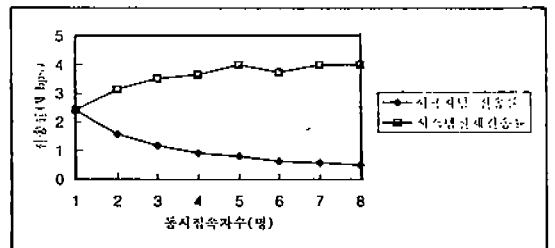
6. 성능 평가

본 논문에서 제안한 복합 다단계 주문형 비디오 서버 시스템의 프로토타입 시스템의 성능 평가를 위하여 설정된 주요 시스템 변수는 다음과 같다.

〈표 2〉 성능 평가를 위한 시스템 변수
(Table 2) System Parameter for Performance Evaluation

광자기 디스크 교환 시간	12 sec
디스크 블록의 크기	3M byte
메모리 블록의 크기	64K byte
네트워크 블록의 크기	4K byte
실험용 데이터	MPEG1 스트림 54M byte

서버의 성능 평가를 위한 실험은 접속자 수에 따라 각5회 반복 수행하여 이의 평균치를 산출하였으며, 사용자 단말에서 MPEG스트림을 복호, 재생하는데 필요한 시간은 무시하였다. (그림11)에서 보듯이 사용자가 3명 또는 4명을 넘어서는 경우 해당 사용자에 대한 평균 전송률이 1.5Mbps를 만족시키지 못하게 되는데, 이 경우 서버는 정적 서비스 허가 단계에서 접속을 허락하지 않게 되나 실험에서는 서버의 부하에 관계없이 서비스를 허가하여 전송률을 산출하였다.



(그림 11) 접속 증가에 따른 서버 시스템의 성능
(Fig. 11) Performance of Server System as Increase of User Connection

서버 시스템 전체의 전송률은 동시 접속자 수가 증가함에 따라 지속적으로 증가하였으며, 동시 접속자가 증가함에 따라 각 사용자에 대한 전송률의 평균값이 MPEG1 데이터의 평균 전송률을 만족시키지 못하지만, 이는 프로토타입 시스템의 구현 환경이 Pentium-60 PC로써 비디오 데이터를 처리하기 위한 프로세싱 파워가 부족하고, 성능 평가를 위한 네트워크 환경이 Ethernet인 관계로 다수의 사용자에 대하여 충분한 네트워크 대역폭을 확보하지 못함에 기인한 것이다.

7. 결 론

본 논문에서는 구성된 복합 다단계 주문형 비디오 서버 시스템은 하드디스크로 구성된 기존의 비디오 서버에 비하여 광자기 디스크를 데이터 저장 매체로 사용함으로써 상대적으로 저가격으로 대용량의 비디오 데이터를 안정적으로 저장할 수 있는 시스템이다. 생산자는 데이터 신뢰성이 높은 광자기 디스크를 대용량 저장 장치인 주크박스에 저장·관리하고 실시간 서비스가 가능하도록 비디오 데이터를 효율적으로 스케줄링한다. 소비자는 생산자에서 제공하는 비디오 데이터를 공유 디스크를 통하여 메모리에 로딩하여 이를 사용자에게 전송하며 임시 접근 제어를 위한 시작 데이터 및 중간 데이터를 관리하여 서버가 사용자의 요구에 빠른 반응 시간을 유지하도록 하며, 운영 센터는 비디오에 대한 정보, 사용자 등록, 서비스 승인, 요금 산정 등의 기능을 수행하여 서버 시스템에 대한 운용을 담당한다. 또한 코디네이터는 사용자 단말과 서버간의 제어 신호의 교환 및 해석을 전담하며 서버가 사용자의 요구에 따라 최적화하여 동작될 수 있도록 한다.

본 논문에서는 복합 다단계 주문형 비디오 서버 시스템에 있어서 필요한 소프트웨어 모듈들을 정의하고 각 모듈간의 제어 신호 및 비디오 데이터의 전송 방법과 전송 프로토콜을 설계·구현하였다. 이러한 소프트웨어 모듈의 구성요소는 복합 다단계 구조가 아닌 일반적인 주문형 비디오 서버 시스템에서도 필요한 사항이며 구축하고자 하는 서버의 환경에 알맞게 재구성되어야 한다. 현재 서비스 시스템은 본 논문에서 제안한 프로토타입과 프로토콜에 준하여 리눅스

(Linux) 운영체제를 기반으로 응용 프로그램 레벨에서 구현하여 동작하고 있으며, 향후 ISDN 등의 광대역 네트워크에서의 작동 및 적용 실험이 필요하다.

참 고 문 헌

- [1] Daniel Deodere, Willem Verbiest and Henri Verhille "Interactive Video On Demand", *IEEE Communications Magazine*, pp82-88, May 1994.
- [2] K. Salem and H. Garcia-Molina, "Disk Striping", *Proceeding of the Second International Conference On Data Engineering*, 1986.
- [3] David A. Patterson, Garth Gibson and Randy H. Katz, "A Case for Redundant Arrays of Inexpensive Disks(RAID)", *ACM SIGMOD Conference*, pp109-116, 1988.
- [4] Fouad A. Tobagi, Joseph Pang, Randall Baird and Mark Gang, "Streaming RAID™-A Disk Array Management system For Video Files", *ACM Multimedia '93 Proceedings*, pp393-400, 1993.
- [5] 김수정, "복합 다단계 VOD서버 시스템에서의 효율적 저장 방법 연구", *연세대학교 컴퓨터과학과 석사학위논문*, 1995.
- [6] A.L. Narasimha Reddy and J. Wyllie, "Disk Scheduling in a Multimedia I/O System", *ACM Multimedia '93 Proceedings*, pp225-234, 1993.
- [7] Wei Kuan Shih, Jane W. S. Liu and C. L. Liu, "Modified Rate-Monotonic Algorithm for Scheduling Periodic Jobs with Deferred Deadlines", *IEEE Trans. on Software Engineering*, vol. 19, no. 12, pp1171-1179, 1993.
- [8] A.L. Narasimha Reddy and James C. Wyllie, "I/O Issues in a Multimedia System", *IEEE Computer Magazine*, pp69-74, March 1994.
- [9] D. Bitton and J. Gray. "Disk Shadowing", *VLDB*, pp331-338, 1988.
- [10] Yurdaer N. Doganata and Asser N. Tantawi, "Making a Cost-Effective Video Server", *IEEE Multimedia*, pp22-30, Winter 1994.
- [11] 김수정의, "복합 다단계 VOD 서버의 효율적 자

료 저장 기법”, 한국정보과학회 가을 학술발표논문집, pp201-204, 1995.

- [12] Hector Garcia-Molina and Kenneth Salem, “Main Memory Database System: An Overview”, *IEEE Trans. On Knowledge and Data Engineering*, VOL.4. No.6 pp509-516, 1992.
- [13] Andrew Lausen, Jeffrey Olkin and Mark Porter, “Oracle Media Server: Providing Consumer Based Interactive Access to Multimedia Data”, *Document for the DAVIC Opening Forum*, Jun. 1994.
- [14] T.D.C. Little and D. Venkatesh, “Prospects for Interactive Video-on-Demand”, *IEEE Multimedia*, pp14-24, Fall 1994.
- [15] P.V. Rangan and H.M. Vin, “Efficient Storage Techniques for Digital Continuous Multimedia”, *IEEE Trans. on Knowledge and Data Engineering* Vol.5, No4, pp564-573, 1993.
- [16] Reza Rooholamini and Vladimir Cherkassaky, “ATM-Base Multimedia Servers”, *IEEE Multimedia*, pp39-52, Spring 1995.
- [17] S. Ghandeharizadeh and L. Ramos, “Continuous Retrieval of Multimedia Data Using Parallelism”, *IEEE Trans. of Knowledge and Data Engineering* Vol.5, No.4, pp658-669, 1993.
- [18] Tat Seung Chua and Jiandong Li, “Disk Striping Strategies for Large Video-on-Demand Servers”, *ACM Multimedia '96 Proceedings*, pp297-306, 1996.

관심분야: 컴퓨터 네트워크, 무선 통신, 데이터베이스



김수정

1993년 이화여자대학교 전산과 학과 졸업(학사)
 1996년 연세대학교 컴퓨터과학과 졸업(이학석사)
 1996년~현재 삼성전자 멀티미디어본부 컴퓨터사업부 객체소프트사업팀 주임연구원

관심분야: 객체지향 데이터베이스, VOD

이원석

1985년 Boston Univ. 컴퓨터공학과 학사
 1987년 Purdue Univ. 컴퓨터공학 석사
 1990년 Purdue Univ. 컴퓨터공학 박사
 1992년 3월~1992년 12월 삼성전자 선임연구원
 1993년~현재 연세대학교 컴퓨터과학과 조교수
 관심분야: 비디오 데이터 모델링, 연역 데이터베이스, 멀티미디어 데이터베이스, 객체지향 데이터베이스



이정수

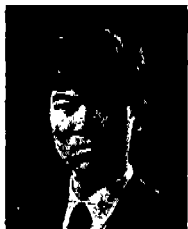
1980년, 한양대학교 전자공학과 졸업(학사)
 1982년 한국과학기술원 전기 및 전자공학과 졸업(공학석사)
 1989년 한국과학기술원 전기 및 전자공학과 졸업(공학박사)

1980년~1991년 금성통신연구소 선임연구원
 1992년~1993년 일본 NTT Human Interface연구소 객원연구원
 1991년~현재 한국통신 전송기술연구소 VOD 연구팀장
 관심분야: 영상통신 서비스, 영상처리, 컴퓨터비전



서덕익

1995년 연세대학교 컴퓨터과학과 졸업(학사)
 1995년~현재 연세대학교 컴퓨터과학과 석사과정
 관심분야: VOD, 멀티미디어 시스템



강대혁

1994년 연세대학교 전산과학과 졸업(학사)
 1996년 연세대학교 컴퓨터과학과 졸업(이학석사)
 1996년~현재 LG전자 멀티미디어 연구소 연구원