

# 원격 교육 시스템을 위한 멀티캐스트 미들웨어

## (A Multicast Middleware for the Remote Educational Systems)

변상선<sup>†</sup>    진현욱<sup>\*\*</sup>    유혁<sup>\*\*\*</sup>  
(Sang-Seon Byun) (Hyun-Wook Jin) (Hyuck Yoo)

**요약** 원격 교육 시스템은 멀티캐스트를 사용함으로써 교육 내용을 전송하는 서버의 부하를 줄일 수 있을 뿐만 아니라 네트워크 대역폭을 효율적으로 사용할 수 있다. 본 논문은 원격 교육 시스템을 지원하기 위한 멀티캐스트 미들웨어를 설계, 구현한다. 본 논문에서 제시하는 멀티캐스트 미들웨어는 다음과 같은 특성을 갖는다: 1) 피동적 멤버를 위한 중앙 집중적인 멀티캐스트 그룹 관리를 통해 특정 수신자 그룹을 송신자가 지정하여 멀티캐스트 전송 2) 그룹 관리 서버와 멤버의 비정상적인 종료로 발생하는 예외 처리를 위한 멀티캐스트 그룹 정보 유지/복구 3) 대용량 멀티미디어 데이터의 멀티캐스트 전송을 위한 단편화 및 역단편화 지원과 버퍼 공유를 통한 추가적인 데이터 복사의 제거. 개발된 미들웨어는 30대의 PC가 Fast Ethernet으로 연결되어 있는 원격 교육 시스템에 실제로 적용하여 초당 18프레임의 320 X 120 픽셀의 해상도를 갖는 동영상과 128kbps의 음성 데이터, 기타 텍스트 메시지를 멀티캐스팅을 통해 요구 조건을 만족하는 송수신 성능을 보였다.

**키워드** : 멀티캐스트, 중앙 집중적 그룹 관리, 수동적 멤버

**Abstract** By choosing Multicast for transmission of educational contents in the Remote Educational System, we can reduce the server load and increase network bandwidth utilization. We design and implement Multicast Middleware for the Remote Educational System in this paper. There are three characteristics in this Multicast Middleware: 1) Through Centralized Multicast Group Management for passive members, it allows a host to make multicast group, which is composed of receivers, called Group Member and who are chosen by the host, called Group Maker. Because, all groups are created by the Group Maker in Centralized Group Management, Group Member's join action will be passive 2) Maintenance and recovery of multicast group information in order to restore from exception and crash; the maintenance and recovery mechanism of Group Maker is distinct from that of Group Member. 3) The mechanism which enables to transmit large size multimedia data through multicasting and remove additional copy operation through shared buffer. Fragmentation/de-fragmentation for large data delivery results in additional copy operation in user level. But by using user level shared buffer, it can be done without user level copy operation. By applying to Remote Educational environment which consists of 30 PCs and Fast Ethernet, we can examine the efficiency of this middleware, which can transmit 18frames/sec movie which resolution 320 X 120 pixels, 128Kbps encoded sound data and some text data.

**Key words** : Multicast, Centralized Group Management, Passive Member

<sup>†</sup> 비회원 : 고려대학교 컴퓨터학과  
ssbyun@os.korea.ac.kr

<sup>\*\*</sup> 학생회원 : 고려대학교 컴퓨터학과  
hwjin@os.korea.ac.kr

<sup>\*\*\*</sup> 종신회원 : 고려대학교 컴퓨터학과 교수  
hxy@os.korea.ac.kr

논문접수 : 2002년 2월 22일  
심사완료 : 2002년 10월 18일

### 1. 서론

원격 교육 시스템은 교육 내용을 전송하는 서버(교사 컴퓨터)와 그 내용을 받는 클라이언트(학생 컴퓨터)로 이루어진다. 이러한 원격 교육 시스템에서 멀티캐스트를 사용하면, 교사 컴퓨터의 부하를 줄일 수 있을 뿐만

아니라, 네트워크의 대역폭을 효율적으로 사용할 수 있다.

또한 멀티캐스트 그룹을 각각의 채널로 활용하여 데이터 타입별로 송수신할 수 있는 장점이 있다[1]. 따라서 원격 교육 시스템을 위한 멀티캐스트 미들웨어는 시스템의 효율성 및 응용프로그램 구현의 용이성을 제공한다[2][3].

원격 교육 시스템을 위한 멀티캐스트 미들웨어에게 가장 요구되어지는 특성은 바로 피동적 멤버를 위한 멀티캐스트 그룹 관리이다. 교육 시스템에서 교사 컴퓨터는 능동적으로 멤버를 정하여 교육 정보를 제공하고 학생 컴퓨터들은 피동적 멤버가 된다. 따라서 교사 컴퓨터가 그룹 및 멤버십 관리를 관장해야 한다. 하지만, 기존의 IGMP(Internet Group Management Protocol)를 기반으로 한 멀티캐스트는 그룹의 멤버가 주체가 되어 가입(join) 또는 탈퇴(leave)를 수행한다[4].

교육용 시스템의 또 다른 특성 중 하나는 학생 컴퓨터의 비정상적인 종료에 있다. 이와 같은 비정상적인 종료 후 재시동 시에도 형성되었던 멀티캐스트 그룹 멤버십에 관련된 정보를 복구해줘야 한다. 이와 같은 예외 처리는 교육 중에 일어날 수 있는 예기치 못한 상태를 자동 복구해줄 수 있다.

또한 네트워크 대역폭이 증가함에 따라, 원격 교육 자료는 단순한 텍스트뿐만 아니라 음성, 화상과 같은 멀티미디어 데이터로 이루어진다. 따라서 대용량 데이터 전송에 적합한 미들웨어가 요구된다[1].

본 논문은 이와 같은 원격 교육용 시스템의 요구에 적합한 멀티캐스트 미들웨어를 설계, 구현한다. 멀티캐스트 미들웨어는 마이크로소프트사의 윈도우 운영체제 제품군에 구현하고 실제 원격 교육용 소프트웨어에 적용하여 그 유용성을 검증한다[5].

본 논문은 다음과 같이 구성되어 있다. 본 서론에 이어 2장에서는 본 논문에서 개발하는 멀티캐스트 미들웨어의 구조를 설명한다. 3장은 피동적 멤버를 위한 멀티캐스트 그룹 관리에 대해서 논한다. 4장은 예외 처리를 위한 멀티캐스트 그룹 유지 기법을 설명하고, 5장은 대용량 멀티미디어 데이터 전송 방법에 대해서 기술한다. 제시한 미들웨어에 대한 구현 및 실험은 6장에서 설명한다. 7장에서는 관련연구를 서술하고 마지막으로 8장에서 본 논문의 결론을 맺는다.

## 2. 멀티캐스트 미들웨어

### 2.1 어플리케이션 요구 사항

이 멀티캐스트 미들웨어가 적용될 교육용 어플리케이션이 요구하는 사항은 다음과 같다.

(1) 중앙 집중적 멀티캐스트 그룹 관리 : Group Maker(교사 PC)에 의해 모든 멀티캐스트 그룹이 생성/소멸/수정 될 수 있어야 한다. Group Member(학생 PC)는 피동적 멤버로서 Group Maker로부터 그룹 가입/탈퇴의 제어를 받게 된다. 이 때, 이러한 그룹 생성/소멸/수정과 관련된 제어 메시지 전달에 있어 최소한의 지연 시간을 갖도록 하고, 그 결과를 어플리케이션에게 전달한다.

(2) 교사 PC와 학생 PC의 비정상적인 종료에 따른 예외 상황 처리 : 교사 PC 또는 학생 PC가 비정상적인 종료 후 재시작을 하더라도 기존의 멀티캐스트 그룹 정보를 유지할 수 있어야 한다.

(3) 대용량 데이터 전송 : 64KB이상의 대용량 데이터의 멀티캐스팅을 위한 단편화와 역단편화 기능을 수행할 수 있어야 한다. 단편화와 역단편화 수행에 따른 별도의 복사과정으로 인한 성능저하를 제거한다.

(4) 다양한 데이터의 동시 송수신 가능 : 멀티캐스트를 통해 다양한 데이터의 송수신이 동시에 이루어져야 한다. 즉, 여러 종류의 데이터를 각기 다른 주소를 사용하여 동시에 송수신 할 수 있어야 한다.

### 2.2 미들웨어 전체구조

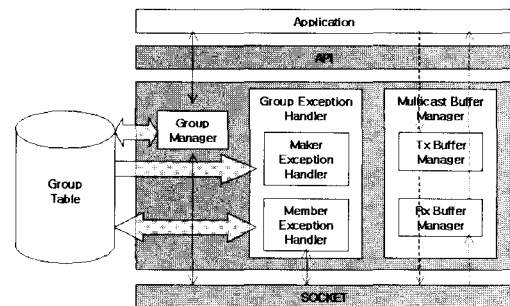


그림 1 Group Maker의 미들웨어 구성도

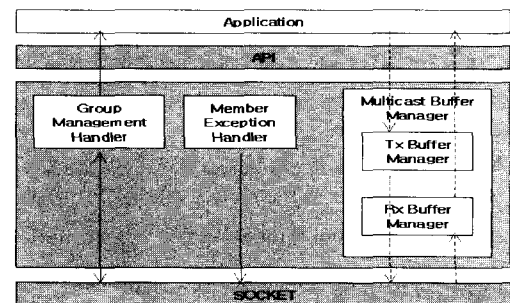


그림 2 Group Member의 미들웨어 구성도

본 미들웨어를 사용하는 원격 교육 시스템은 교사의 컴퓨터에서 모든 멀티캐스트 그룹의 관리가 이루어지고, 학생의 컴퓨터는 피동적으로 교사 컴퓨터의 그룹 관리에 따라 가입 및 탈퇴를 한다. 따라서, 본 논문에서는 교사의 컴퓨터를 Group Maker, 학생의 컴퓨터를 Group Member 라는 용어를 사용하여 설명한다.

멀티캐스트 미들웨어는 그림 1, 2와 같이 Group Maker와 Group Member를 위한 미들웨어가 각각 존재한다. 각 모듈별 역할은 간략하게 다음과 같다.

(1) API: 응용프로그램에게 미들웨어의 기능을 사용하기 위한 프로그램 인터페이스를 제공한다. 제공되는 대표적인 API들은 표 1과 같다.

표 1 멀티캐스트 미들웨어 API

API	Description
Make_McastGroup()	멀티캐스트 그룹 생성
Group_Modify_Send()	멀티캐스트 그룹에 하나의 멤버를 추가/삭제
Close_Group()	멀티캐스트 그룹 소멸
Mcast_Send()	멀티캐스트 데이터 송신
Mcast_SendLarge()	대용량의 멀티캐스트 데이터 송신
Mcast_Recv()	멀티캐스트 데이터 수신

(2) Group Manager, Group Management Handler: Group Maker의 Group Manager가 멀티캐스트 그룹을 생성/수정/삭제하는 메시지를 작성하여 해당 Group Member에게 전달하고 Group Member의 Group Management Handler는 그룹에 가입 또는 그룹으로부터 탈퇴를 수행한다.

(3) Group Exception Handler, Member Exception Handler: 비정상적인 종료로부터 재 시작하였을 경우 자신이 유지하고 있던 멀티캐스트 그룹 관련 정보를 복구하기 위해 필요하다. Group Maker는 그룹 관리 연산이 이루어질 때마다 이를 파일로 저장을 하게 되고, Maker Exception Handler를 통해 파일로부터 그룹 정보를 복구한다. Group Member는 Member Exception Handler를 통해 Group Maker에게 그룹 정보를 요청하고, 이를 전달 받은 Group Maker로부터 해당 그룹으로의 가입 메시지를 전달 받아서 그룹 정보를 복구 한다.

(4) Multicast Buffer Manager: 대용량 데이터에 대한 멀티캐스트 송수신을 수행한다. 멀티캐스트는 전송계층으로 UDP를 사용하기 때문에 64KB 이상의 데이터 송수신을 위해서는 별도의 송수신 버퍼관리가 필요하게 된다. 이를 위해서 Multicast Buffer Manager는 64KB보다 큰 데이터에 대해서 단편화와 역단편화를 수행한다.

### 3. 그룹 관리

원격 교육 시스템은 Group Maker가 특정 Group Member들을 선택하여 교육 및 제어 데이터를 송신해야 효율적이다. 즉, 멀티캐스트 그룹의 멤버들은 기존의 멀티캐스트 시스템과 같이 그룹 가입과 탈퇴 동작이 허용되는 능동적 멤버이어서는 안되고, 오직 Group Maker에 의해서만 그룹 가입/탈퇴 동작을 수행되는 피동적 멤버이어야 한다. 또한, 교육 중간에 빈번하게 그룹이 생성/수정/소멸되기 때문에 효율적이고 정확한 그룹 정보 관리가 요구된다. 본 장에서는 이러한 피동적 멤버를 위한 관리 기법을 제시한다.

#### 3.1 그룹 생성

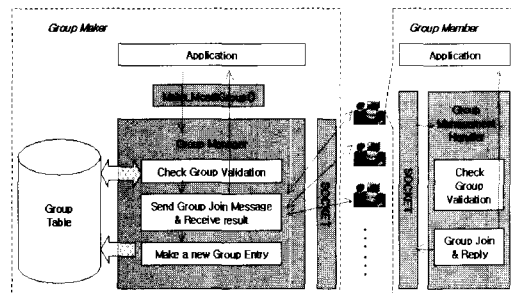


그림 3 그룹 생성 과정

그룹 생성은 하나의 멀티캐스트 그룹을 생성하여 Group Maker 자신도 멤버로 가입을 한 후, 지정된 Group Member들을 가입시키는 것이다.

그림 3과 같이 Group Maker의 응용프로그램이 그룹의 생성을 미들웨어에게 Make\_McastGroup()을 사용하여 요청하면, 같은 멀티캐스트 주소를 사용한 그룹이 이미 만들어졌는가를 검사한다. 생성, 수정의 경우 Group Member 측에서도 정당성 확인이 필요하다. 그 이유는 Group Maker가 비정상적인 종료 후 재시작 하였을 때 예외처리과정(4. 예외처리 참고)을 통해 재차 그룹을 생성하려 할 것이고, 이 때, 이미 같은 멀티캐스트 주소를 갖는 그룹의 멤버인 Group Member들은 이 요청에 응답을 해서는 안되기 때문이다. 만약, 응답을 하게 되면 같은 그룹에 재차 가입하는 것이 되어 소켓 및 자료구조, 송수신 쓰레드, 수신 버퍼를 중복 운용하게 되므로 자원의 낭비를 초래하게 된다. 정당성 확인 절차에서 Group Maker는 그룹 테이블을 통해 자신이 이미 만든 그룹 중에 같은 멀티캐스트 주소를 갖는 레코드 항목의 존재를 확인한다. Group Member는 자신이 가입한 그

그룹 주소 리스트를 통해 확인한다.

이미 존재하고 있는 동일한 그룹이 없으면 다음의 과정을 통해서 그룹을 생성한다.

(1) 그룹 가입 요청 메시지를 Group Member들에게 전달한다(그룹 가입 요청 메시지와 같은 그룹 제어 메시지들의 전송 프로토콜은 3.4절에서 설명한다).

(2) 그룹 가입 요청 메시지를 전달 받은 Group Member는 이미 해당 그룹의 멤버인지 그 정당성을 확인한다. 그룹 정당성에 이상이 없으면, 미들웨어는 요청 받은 그룹에 가입을 한다.

(3) 정상적으로 가입이 완료되면 미들웨어는 응용프로그램에게 해당 멀티캐스트 그룹으로 가입이 되었음을 알리고, Group Maker의 미들웨어에게 가입이 성공적으로 이루어졌음을 알린다.

### 3.2 그룹 수정

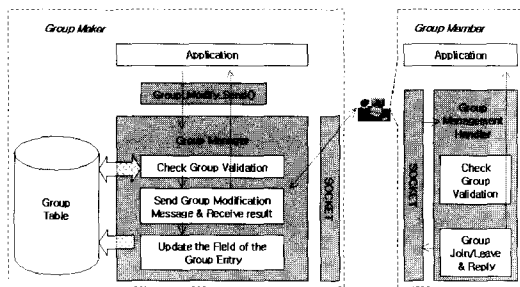


그림 4 그룹 수정 과정

그룹 수정은 그림 4와 같이 이미 생성되어진 그룹 내에서 하나의 멤버를 가입 또는 탈퇴하는 것을 말한다.

Group Maker는 Group Table을 통해서 가입일 경우는 그 Group Member가 해당 멀티캐스트 그룹의 멤버가 아닌지, 탈퇴일 경우는 멤버인지 확인한다. 확인후 이상이 없으면, 다음의 과정을 수행한다.

(1) Group Maker의 미들웨어는 응용프로그램으로부터 하나의 Group Member를 가입 또는 탈퇴할 것을 요청 받으면 그 Group Member에게 메시지를 전달한다.

(2) 그룹 수정 메시지를 수신 받은 Group Member의 미들웨어는 정당성을 확인하고, 정당성에 위배 될 경우 그룹 수정 거부를 Group Maker에게 응답한다.

거부를 응답하는 조건은 다음과 같다.

- i. 가입 요청에 대해 해당 멤버가 이미 해당 그룹의 멤버이다.
- ii. 탈퇴 요청에 대해 해당 멤버는 해당 그룹의 멤버가 아니다.

(3) Group Member는 정당성이 확인되면, 요청 받은 그룹에 가입 또는 탈퇴를 한다.

(4) 정상적으로 수정이 이루어지면, Group Member의 미들웨어는 응용 프로그램에게 그룹 수정이 되었음을 알리고, 그룹 수정 메시지를 전달한 Group Maker의 미들웨어에게 그룹 수정이 성공적으로 이루어 졌다는 메시지를 전송한다.

### 3.3 그룹 소멸

그룹 소멸은 그림 5와 같이 존재하고 있는 그룹의 모든 멤버를 탈퇴(pruning)시키고 Group Maker 자신도 탈퇴하는 것을 말한다.

그룹 소멸을 수행하기 전에 Group Maker는 정당성을 확인한다. Group Maker에서 그룹 소멸시 확인하는 정당성은 Close\_Group()이 해당 그룹의 Group Maker에 의해 호출되었는지를 확인하는 것이다. Group Maker의 정당성이 확인되면 다음의 과정을 통해 그룹을 소멸한다.

(1) Group Maker는 그룹 탈퇴 요청 메시지를 Group Member들에게 전달한다.

(2) Group Member의 미들웨어는 해당 그룹으로부터 탈퇴를 한다.

(3) 탈퇴가 정상적으로 이루어지면 Group Member의 미들웨어는 응용 프로그램에게 해당 그룹으로부터 탈퇴하였음을 알리고, Group Maker의 미들웨어에게 그룹 탈퇴가 성공적으로 이루어졌음을 알린다.

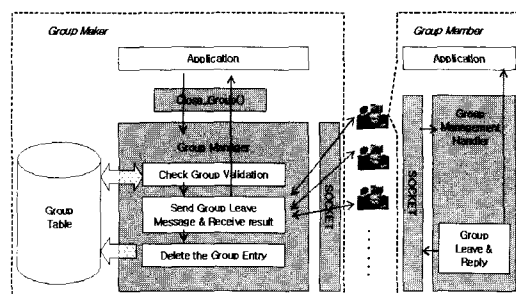


그림 5 그룹 소멸 과정

### 3.4 그룹 관리 메시지 전송 프로토콜

Group Maker가 Group Member들에게 그룹 생성/수정/소멸과 관련한 그룹 제어 메시지를 전송하는 프로토콜에 따라 그룹 제어 성능이 크게 달라질 수 있다. 본 논문에서는 Broadcasting, Broadcasting with ACK, TCP, UDP with ACK의 네 가지 다른 전송 프로토콜을 비교하여 그룹 제어 메시지 전송에 적합한 프로토콜을 선택한다. UDP나 Broadcasting이 TCP에 비해 나은

성능을 보일 것은 자명하나, 이 가운데 Broadcasting with ACK과 UDP with ACK의 성능이 본 실험의 주 비교 대상이다. Broadcasting은 근거리 교육 시스템에서 그룹 제어 메시지를 모든 Group Member에게 한번에 전송할 수 있는 장점이 있다. 하지만, Broadcasting을 이용할 경우, 제어 메시지에 대한 신뢰성을 보장해주지 못한다. 또한 그룹에 가입 또는 탈퇴할 Group Member들이 명시되어 있어야 하기 때문에 메시지의 길이가 길어진다. 그리고 Broadcasting된 메시지에 반응하지 않아야 하는 Group Member들까지 메시지를 수신하는 단점이 있고, Group Maker와 Group Member들이 근거리 내에 위치하지 않을 경우에는 적용하지 못한다.

Broadcasting의 비신뢰성을 보완하기 위해서 Broadcasting with ACK을 사용할 수 있다. Group Member들이 그룹 제어 메시지를 보낸 Group Maker에게 ACK을 보냄으로써 신뢰성을 보장한다. 이 때, 고려되어야 하는 것은 ACK가 서버 측으로 일시에 몰리는 현상(ACK Implosion)이다. 이러한 현상을 제거하기 위해서 Group Member는 랜덤 대기시간 후 ACK를 송신할 수 있다. 하지만 효율적인 랜덤 대기시간을 정하는 것은 어려운 일이다.

TCP는 신뢰성이 보장되는 전송 프로토콜이므로 그룹 제어 메시지를 전송하기에 적합하다. TCP를 사용하여 그룹 제어 메시지를 전송할 경우, Group Maker는 해당 Group Member를 각각에게 TCP 연결을 설정하고 제어 메시지를 전송한 후 TCP 연결을 닫는 작업을 차례로 한다. 시스템 수행동안 제어 메시지를 위한 TCP 연결을 모든 Group Member와 설정하고 지속적으로 유지할 수도 있지만, Group Member의 수가 많을 경우에는 연결 설정 유지에 필요한 메모리의 낭비를 초래할 수 있다.

UDP with ACK은 전송 계층으로 TCP보다는 오버헤드가 적은 UDP를 적용하고, UDP의 비신뢰성을 보완하기 위해서 ACK기법을 사용한다. TCP를 사용하는 경우와 마찬가지로, Group Maker는 해당 Group Member들에게 그룹 제어 메시지를 전송하고 ACK을 받는 작업을 순차적으로 수행한다. 재전송을 위한 타이머는 Group Maker와 Group Member간의 Round-Trip-Time(RTT)를 이용하여 설정할 수 있다.

설정된 네 프로토콜에 대한 성능 분석은 그림 6과 같다. 그림 6은 그룹 생성 동작의 성능을 측정하는 것이다. 그룹 생성시 모든 프로토콜은 Group Maker 자신의 그룹 가입, 그룹 가입 요청 메시지 작성, 메시지 전송의 절차를 거치게 되며, Broadcasting without ACK을 제외한 모든 프로토콜은 Group Member로부터 처리 결과

를 수신한다. 정상적인 상황에서의 비교를 위해 네트워크 상의 물리적 오류, 운영체제 상의 오류로 인한 패킷 손실은 없다고 가정하였다.

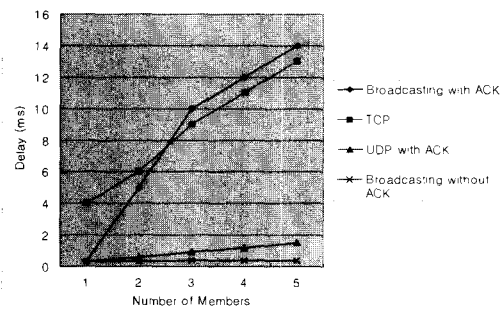


그림 6 그룹생성시 각 전송용 각 프로토콜의 처리 시간

각 프로토콜의 성능 분석 방법은 다음과 같다. Broadcast without ACK을 제외한 TCP, UDP with ACK, Broadcast with ACK 프로토콜은 그룹 관리 메시지를 전송하고 난 다음부터 모든 Group Member들로부터 응답을 받기까지 걸리는 시간을 측정하였다. TCP의 경우는 Group Maker로부터 모든 Group Member들과의 1:1 TCP 세션을 유지한 상태에서, UDP의 경우는 1:1 연결 소켓만 생성한 상태에서 순차적으로 Group Member들에게 일일이 그룹 관리 메시지를 전송하였다. 응답은 수신 쓰레드를 따로 두어서 처리를 하였다. Broadcast with ACK의 경우는 한번의 브로드캐스팅으로 그룹 관리 메시지를 모든 멤버에게 전달하고, 역시 응답은 수신 쓰레드를 따로 두어 처리를 하였다.

그림 6에서 Broadcasting without ACK이 가장 적은 지연시간을 보이고 있다. 또한 쉽게 예상할 수 있듯이 UDP with ACK이 TCP에 비해 훨씬 좋은 성능을 보이고 있다. Broadcasting without ACK의 경우 한번에 모든 Group Member들에게 제어 메시지를 전송할 수 있기 때문이다. 하지만, Broadcasting without ACK은 성능 비교를 위한 기반을 제시해줄 뿐, 신뢰성을 제공해주지 못하기 때문에 실제로 응용하기에는 문제점이 있다.

Broadcasting with ACK은 그룹의 멤버 수가 증가함에 따라 그 지연시간이 급격하게 증가하는 것을 알 수 있다. 이것은 ACK Implosion을 막기 위해서 Group Member들이 랜덤 대기시간 후 ACK을 전송하기 때문이다. 랜덤 대기시간을 조정하면 지연시간을 줄일 수는 있으나, 최상의 랜덤 대기시간을 찾는 것은 어려운 문제

이다.

TCP와 UDP with ACK은 그림 6에서 보듯이 선형적으로 처리 시간이 증가하는 것을 알 수 있다. 하지만 TCP와 UDP with ACK은 지연시간에서 큰 차이를 보이고 있다. 이것은 TCP가 연결을 설정하고 제거할 때마다 각각 three-way handshake와 active/passive close를 수행하기 위해서 TCP 제어 패킷들을 교환하기 때문이다[6]. 이러한 실험 결과는 하나의 제어 메시지를 보내기 위해서 TCP 연결을 생성, 제거하는 것은 비효율적이라는 것을 보여준다. 따라서 본 논문에서는 그룹 제어 메시지 전송을 위한 프로토콜로서 UDP with ACK을 선택하고, 비교된 나머지 전송 프로토콜들은 옵션으로 구현한다.

#### 4. 예외 처리

원격 교육시스템이 동작하는 동안 시스템 또는 사용자의 오류로 인해서 Group Maker 또는 특정 Group Member가 비정상적으로 종료 후 재시작할 수 있다. 이때, 재시작 된 Group Maker 또는 Group Member는 가입된 그룹의 멤버십을 모두 잃게 되어 수신해야 할 데이터를 수신하지 못하게 된다. 본 미들웨어는 예외처리 모듈을 통해서 Group Maker와 Group Member가 이러한 예외 상황을 처리할 수 있도록 한다.

##### 4.1 Group Maker 예외 처리

생성된 그룹과 그 그룹별로 멤버리스트를 관리하고 있는 Group Maker가 오류를 일으켜서 동작이 중지된다면 Group Maker가 정상적인 상태에서 유지하고 있는 group table은 자료구조이기 때문에 모든 멤버십 정보를 잃어버리게 된다. 이러한 경우를 위해서 Group Maker는 생성한 멀티캐스트 그룹 주소와 그 그룹의 멤버 주소를 자료구조와 동시에 파일의 형태로도 유지한다. 즉, 그룹 생성/수정/소멸 작업이 이루어 질 때마다 파일을 갱신하여 최신정보를 유지한다. 정상적으로 종료했을 경우에는 파일을 삭제한다. 그리고, 비정상적으로 종료 후 재시작 하였을 경우 다음의 과정을 수행한다.

- (1) Group Maker의 멀티캐스트 미들웨어가 시작될 때 마다, 그룹 정보가 저장되어 있는 파일의 존재유무를 확인한다.
- (2) 만약 비정상적인 종료 후, 재가동 된 것이면 파일로부터 정보를 받아 자동적으로 종료 직전의 상태로 되돌리기 위해 그룹 생성 연산을 수행한다.
- (3) 그룹 정보를 저장하는 파일이 존재하지 않으면 정상적인 종료가 된 것으로 간주한다.

##### 4.2 Group Member 예외 처리

Group Member가 하나 이상의 멀티캐스트 그룹에 가입된 상태에서 비정상적으로 종료될 경우, Group Member는 멀티캐스트 그룹에 대한 정보를 잃어버리게 된다. Group Member의 이러한 예외 상황에 대처하기 위해서 Group Member는 다음을 수행한다.

- (1) Group Member의 미들웨어가 초기화 될 때, 현재 자신이 특정 그룹의 멤버였는가를 요구하는 메시지(멤버십 쿼리)를 Group Maker에게 전송하고, 응답을 기다리게 된다. 이 때, 명시적인 응답이 없으면 정상적인 종료(그룹 수정의 탈퇴연산 후 종료)를 수행한 것으로 간주한다.
- (2) 이 멤버십 쿼리를 수신 받은 Group Maker는 Group Table을 참조하여 해당 Group Member가 자신이 만든 그룹의 멤버인지 확인한다. 멤버십이 확인되면, 그 Group Member에게 그룹 수정의 가입 요청 메시지를 송신한다.
- (3) 멤버십 쿼리를 전송한 Group Member는 Group Maker로부터 그룹 가입 요청 메시지를 수신 받으면 해당 그룹에 가입을 하여 가장 최근 상태의 멤버십 정보를 유지하게 된다.

#### 5. 대용량 데이터의 멀티캐스팅

멀티캐스트의 전송 계층인 UDP는 최대 64KB의 데이터만을 전송할 수 있다. 하지만, 원격 교육 컨텐츠는 대용량의 멀티미디어 데이터로 이루어져 있다. 따라서 응용프로그램은 전송할 대용량의 데이터에 대한 단편화 작업을 수행해야 한다.

본 논문의 미들웨어는 이와 같은 대용량 데이터의 단편화에 대한 추상을 제공하기 위해서 그림 7과 같이 단편화를 수행한다. 각 단편의 헤더에 포함되는 정보는 다음과 같다.

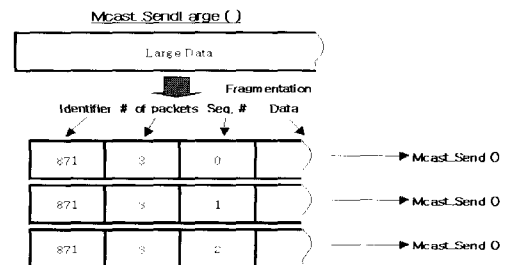


그림 7 대용량 데이터의 단편화

- (1) Identifier : 다른 대용량 데이터로부터 단편화된

일련의 패킷과 구별하기 위해 랜덤하게 생성된다. 이 식별자는 연속해서 다른 대용량의 데이터를 구성하는 단편이 들어왔을 경우 구별하기 위해서 필요하다.

(2) Number of Packets : 단편화를 수행한 후, 몇 개로 나누어졌는가를 나타낸다.

(3) Sequence Number : 단편화가 되었을 경우 0부터 순서대로 다음 단편으로 진행될 때 마다 1씩 증가된다. 이 sequence number를 이용하여 수신 측은 수신 버퍼의 시작 지점으로부터 (단편의 크기 \* sequence\_number) bytes 지점에 저장을 하게 된다.

위와 같이 송신되어진 멀티캐스트 데이터를 수신할 때, 응용프로그램이 수신 버퍼를 폴링하여 들어온 데이터가 있으면 응용프로그램이 가져가거나, 아니면 데이터가 들어올 때 미들웨어가 응용프로그램에게 윈도우 메시지를 전송하여 수신하게 할 수 있다. 이 때, 미들웨어가 수신한 데이터가 단편화가 발생한 데이터의 일부일 경우, 원래 데이터를 구성하는 모든 단편이 수신된 후에 응용프로그램이 데이터를 읽어가도록 한다.

단편화가 발생한 데이터를 수신할 경우, 단편화가 발생한 데이터의 마지막 부분이 손실되면 교착상태에 빠지게 된다. 이를 막기 위해 단편화가 발생한 데이터 중에서 서로 인접한 두 단편의 도착시간 차이를 수신자는 측정한다. 측정한 이 시간의 두 배를 수신자는 다음 단편이 도착하기까지 대기하는 시간으로 설정한다. 패킷 손실로 인해 연속적인 두 단편의 도착시간 차이 측정이 불가능 할 경우 대기시간을 일정한 시간으로 설정한다. 이 대기 시간이 지나도 다음 단편이 수신되지 않으면, 미들웨어는 이제까지 수신한 데이터의 버퍼 포인터를 응용프로그램에게 전달하게 된다.

미들웨어가 데이터의 송수신을 관찰할 때, 성능을 좌우하는 중요한 요소는 데이터 복사이다. 응용프로그램과 미들웨어가 별도의 통신 버퍼를 사용하게 되면, 응용프로그램의 버퍼와 미들웨어의 버퍼간 데이터 이동을 위해서 데이터 복사가 발생한다. 이 오버헤드는 멀티미디어 데이터일 경우에 그 대용량성에 의해서 더 치명적이다. 본 미들웨어는 버퍼에 대한 포인터만을 응용프로그램과 교환하고 버퍼 자체는 공유하도록 한다.

수신 버퍼 할당은 미들웨어가 수행하며, 이 버퍼의 포인터를 전달 받은 응용프로그램이 해지한다. 미들웨어는 버퍼를 할당할 때 단편화 없이 하나의 패킷으로 전달 가능한 데이터의 최대 사이즈로 할당한다. 소켓을 통해 데이터가 도착하면 버퍼의 포인터를 어플리케이션에게 전달하고 미들웨어는 다음에 수신될 데이터를 위해 새로운 버퍼를 할당한다. 단편화가 발생한 패킷이 수신되

었을 경우에는 모든 단편이 수신 될 때까지 이전 단편이 차지하고 있는 버퍼에 새로 수신된 단편을 계속 붙여나간다. 모든 단편이 수신되면 첫 번째 단편의 버퍼 포인터를 어플리케이션에게 전달한다. 어플리케이션은 수신 받은 데이터에 대한 처리가 모두 끝나면 버퍼를 해지한다.

## 6. 구현 및 실험

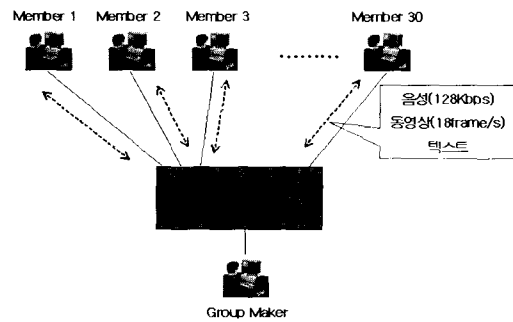


그림 8 실험 환경

제시한 멀티캐스트 미들웨어는 마이크로소프트사의 윈도우 제품군(Windows NT/98/ME/2000)에 구현되었으며, 응용프로그램으로는 (주)훈소프트의 Schoolcap2를 적용시켰다. Schoolcap2는 근거리망에서 PC실습 교육을 위한 원격 교육 시스템으로서 음성, 화상, 텍스트 데이터의 송수신이 이루어진다. 실험은 30대의 PC가 Fast Ethernet으로 연결되어 있는 환경에서 수행하였고, 2.1 절에서 제시한 이 교육용 소프트웨어가 미들웨어에게 요구하는 기능들을 대상으로 다음과 같이 실험하였다.

(1) 중앙 집중적 멀티캐스트 그룹 관리: 교사 PC를 통해 30대의 학생 PC를 대상으로 멀티캐스트 그룹을 생성/삭제/수정 연산을 수행하였다. 그룹 연산은 UDP with ACK을 사용하였다. 실험과정에서 교사 PC로부터 처음 송신된 생성/삭제/수정 메시지를 받지 못한 학생이 발생하기도 하였다. 그러나 교사 PC에서 메시지를 재전송함으로써 모든 학생이 메시지를 수신 받아서 메시지에 따른 그룹 연산을 수행하는 것을 확인할 수 있었다.

(2) 교사 PC와 학생 PC의 비정상적인 종료에 따른 예외 상황 처리: 윈도우 운영체제 상에서 강제적인 프로세스 종료를 통해, 교사 PC와 학생 PC에 예외상황을 발생 시켰다. 예외상황을 발생시킨 후, 교사 PC와 학생 PC의 어플리케이션을 재시작 하였을 때, 예외상황 발생 직전의 상태로 모든 그룹의 정보가 복구되었다. 교사 PC만을 강제 종료 후 복구하는 과정, 학생 PC만을 강

제 종료 후 복구하는 과정, 교사 PC와 학생 PC를 모두 강제 종료 후 복구하는 과정에 대해 모두 실험하였다.

(3) 대용량 데이터 전송: 대용량 멀티미디어 데이터의 전송지원을 확인하기 위해 초당 18프레임의 320 X 120의 MPEG 동영상 파일과 캠코더와 MPEG 보드를 통해 실시간 인코딩된 동영상 데이터의 전송을 실험하였다. 그리고, 교사 PC에서 동영상을 재생하면서 학생 PC에서도 교사 PC로부터 전송 받은 동영상 파일을 실시간으로 재생하였다. 실험 결과, 교사 PC에서 재생되는 프레임과 학생 PC에서 재생되는 프레임이 거의 일치하는 것을 확인할 수 있었다. 구현된 미들웨어는 멀티캐스트 데이터 전송의 신뢰성을 보장하지 못하기 때문에, 패킷 손실에 의한 학생 PC 화면의 일시적인 품질저하 현상이 관찰되었다. 다른 대용량 데이터의 전송을 실험하기 위해 1024 X 768의 윈도우 데스크 탑 화면을 실시간으로 인코딩하여 전송하였다. 인코딩은 RLE(Run Length Encoding)만을 사용하였고, 동영상과 같이 빈번하게 프레임이 변하지 않는다는 점을 감안하여, 손실된 매크로 블록에 대해서 재전송을 해주는 에러 제어 모듈을 어플리케이션에서 구현하였다. 또한, 동영상 전송과 같이 일정한 프레임으로 전송하지 않고, 윈도우 데스크 탑 화면의 변화를 감지하여, 변화가 있을 경우, 변화가 생긴 매크로 블록만을 전송해주는 방식을 사용하였다. 실험 결과, 변화가 발생한 화면 영역이 커지면 커질수록, 전송해야 하는 데이터의 용량이 커지고 또 그에 따른 에러 제어 모듈의 동작으로 인해 품질 저하가 발생함을 확인하였다. 이는 전송의 MPEG 보드를 사용한 동영상 전송과 달리 어플리케이션에서 직접 RLE 인코딩과 디코딩 작업을 행함에 따라, 성능이 저하되는 것을 확인하였다.

(4) 다양한 데이터의 동시 송수신 : 아래와 같은 데이터 종류 별로 각각 하나의 멀티캐스트 그룹을 할당하여 동시에 전송하는 과정을 실험하였다.

- (i) 128Kbps로 인코딩된 음성 파일
- (ii) 채팅 메시지
- (iii) 윈도우 데스크탑 화면(1024 X 768)
- (iv) 동영상 파일(320 X 120, 18frame/s)
- (v) 어플리케이션이 사용하는 각종 제어 메시지

실험 결과, (iii)의 윈도우 데스크탑 화면의 변화로 데스크탑 화면의 인코딩과 전송, 수신 과정이 발생하게 되면 (i)의 음성 파일과 (iv)의 동영상 파일의 일시적인 품질저하(끊김 현상)가 발생하였다.

## 7. 관련연구

원격 교육 시스템과 같은 중앙 집중적인 그룹 관리는

CGM(Contractual Group Membership)[7] 메커니즘을 통해서 그 예를 찾아 볼 수 있다. CGM은 멤버를 대역폭, 전송 지연, 패킷 손실률에 따라 비슷한 성능을 갖는 멤버끼리 서브 그룹을 형성하여 멀티캐스팅에 있어서 공정성을 이루는 그룹 관리 메커니즘이다. 이 연구는 중앙 집중적인 그룹 관리를 통해 서브 그룹 형성이 있을 경우가 전반적인 성능의 향상을 가져온다는 것을 보이고 있다. 그러나, 그룹관리와 관련한 구체적인 구조 제시와 구현은 이루어지지 않았다. STORM(Structure-Oriented Resilient Multicast)[8]과 LVMR(Layerd Video Multicast Recovery)[9]에서는 멀티미디어 스트리밍을 전송할 때 있어서 각 멤버들과 링크의 성능에 따라 서브 그룹을 형성한다. 그런 후에, STORM은 각 그룹 별로 그 그룹의 성능에 적합한 bit rate로 스트리밍을 인코딩하는 것이며, LVMR은 패킷 손실을 복구할 때, 각 그룹 별로 그 그룹의 성능에 부합하는 레이어에 해당하는 인코딩 데이터를 복구하는 것이다. LMR(Layered Multicast Recovery)[10]은 각 멤버들의 패킷 손실률을 통계적으로 예측하여 비슷한 손실률을 갖는 멤버끼리 서브 그룹을 형성한다. 송신자는 이들 그룹 별로 해당 손실률에 따른 에러를 충분히 복구할 수 있는 FEC(Forward Error Correction)데이터를 여분으로 첨가하여 전송하는 것이다. 이와 같은 STORM, LVMR, LMR의 서브 그룹 형성에 본 멀티캐스트 미들웨어에서 사용하는 중앙 집중적인 그룹 관리를 통해 더욱 효율적으로 이루어질 수 있다.

## 8. 결론 및 향후과제

본 논문은 특정 호스트(Group Maker)가 멀티캐스트 그룹을 생성/수정/소멸할 수 있는 중앙 집중적인 멀티캐스트 그룹 관리, Group Maker와 Group Member 각각을 위한 예외 처리, 단편화와 역단편화, 그리고 공유버퍼 관리를 통해 대용량 멀티미디어 데이터의 송수신이 가능한 멀티캐스트 미들웨어를 설계, 구현하였다. 그리고, 다양한 형태의 데이터를 데이터 별로 각각 하나의 멀티캐스트 그룹을 할당 동시에 송수신할 수 있도록 하였다. 이 미들웨어를 교육용 어플리케이션에 적용, 위와 같은 기능들이 원활히 수행되어 짐을 확인하였다.

향후 계획으로는 원거리 멀티캐스팅으로의 확장을 위해 신뢰성을 갖는 멀티캐스트 프로토콜과, 이질적(heterogeneous) 수신자를 위한 세션 및 수신자간의 공정성(fairness) 확립, 실시간 멀티미디어 데이터 전송에 적합한 에러 복구, 흐름 제어 기법의 개발 및 도입이 이루어져야 할 것이다. 또한, 본 미들웨어의 예외처리는 전적으로 로그 파일에 의존을 하고 있기 때문에, 파일에 그



룹 정보의 변경 사항을 기록하는 도중 발생하는 예외상황에 대해서는 대처를 하지 못하고 있다. 향후, 파일에 의존하지 않는 예외 처리가 요구된다.

그리고, Group Maker와 Group Member간의 멤버십 정보의 동기화를 위한 정형 검증 등이 필요하다.

### 참 고 문 헌

- [1] X. Li, "Scalable and Adaptive Video Multicast over the Internet," PhD thesis, Georgia Institute of Technology, 1998.
- [2] C. Diot, B.N. Levine, B. Lyles, H. Kassem, and D. Balensifen, "Deployment issues for the IP multicast service and architecture," IEEE Network, Volume: 14, Issue: 1, Jan.-Feb. 2000.
- [3] R. Wittmann and M. Zitterbart, *Multicast Communication: Protocols and Applications*, Morgan Kaufmann Publishers, 2001.
- [4] W. Fenner, "Internet Group Management Protocol, version 2," IETF RFC 2236, Nov. 1997.
- [5] A. Jones and J. Ohlund, *Network Programming for Microsoft Windows: Clear, Practical guide to Microsoft's networking APIs*, Microsoft Press, 1999.
- [6] R. Stevens, *TCP/IP Illustrated: The Protocols*, Addison-Wesley Publishing Company, 1994.
- [7] T. Asfour, S. Block, A. Serhrouchni and S.Tohme, "Contractual Group Membership CGM: a new mechanism for multicast group management," Computers and Communications, 2000, Proceedings, ISCC 2000, Fifth IEEE Symposium on, 2000.
- [8] X. R. Xu, A. Myers, H. Zhang, and R. Yavatkar, "Resilient multicast support for continuous-media applications," in Proceedings of NOSSDAV '97, May 1997.
- [9] I. Rhee, S. R. Joshi, M. Lee, S. Muthukrishnan and V. Ozdemir, "Layered Multicast Recovery," in Proceedings of IEEE INFOCOM 2000, 2000.
- [10] X. Li, S. Paul, P. Pancha, and M. H. Ammar, "Layered Video Multicast with Retransmission (LVMR): Evaluation of error recovery," in Proceedings of NOSSDAV '97, May, 1997.



변 상 선

1996년 고려대학교 전산학과 졸업. 2002년 고려대학교 통신시스템 기술학과 석사 졸업. 2002년~현재 고려대학교 컴퓨터학과 박사과정. 관심분야는 멀티캐스트 전송 프로토콜, 그리드 컴퓨팅, 무선 네트워크 프로토콜



진 현 욱

1997년 고려대학교 전산학 학사. 1999년 고려대학교 전산학 석사. 1999년 ~ 현재 고려대학교 컴퓨터학과 박사과정. 관심분야는 기가비트 네트워크 프로토콜, 무선 네트워크 프로토콜, 운영체제



유 혁

1982년 서울대학교 전자공학 학사. 1984년 서울대학교 전자공학 석사. 1986년 University of Michigan 전산학 석사. 1990년 University of Michigan 전산학 박사. 1986년 Center for Information Technology Integration 연구원. 1986년 ~ 1988년 CAEN 연구원. 1990년 ~ 1995년 Sun Microsystems, Inc. 연구원. 1995년 ~ 2002년 고려대학교 컴퓨터학과 부교수. 2002년 ~ 현재 고려대학교 컴퓨터학과 교수. 관심분야는 운영체제, 네트워크 프로토콜, 멀티미디어