

초고속망의 원격교육 서비스를 위한 가상클래스의 그룹 관리 시스템

박 판 우

대구교육대학교 전산교육과

요 약

본 연구에서는 초고속망의 원격교육을 지원하는 그룹 관리 시스템을 연구하였다. 효율적인 원격교육을 위해서는 가상클래스 상의 피관리 객체와 그 동작들을 관리, 제어하는 메카니즘이 필요하며, 교수자는 학습그룹 및 하나의 학습그룹을 형성하는 학습자의 다양한 상태변화를 직시할 수 있어야 한다. 그리고, 가상클래스 상에서 교수자와 학습자는 학습그룹에 따라 토론하고 상호 메시지 송수신이 가능해야 한다. 본 논문에서는 이러한 가상클래스 운용을 위한 그룹 관리 및 관리정보베이스 모델을 제안하고, 거기에 요구되는 그룹 관리 알고리즘을 연구하였다.

Group Management System for Virtual Class of Distance Education on the Information Superhighway

Phanwoo Park

Department of Computer Education, Taegu National University of Education

ABSTRACT

I studied a group management model of virtual class for distance education system on the information superhighway. There are many objects and actions that need to be managed and controlled in virtual classes of a distance education model. Educators should be able to manage students' learning group and immediately be aware of who is attending and who is quitting the learning group in a virtual class. Also, students and educators in virtual classes can communicate and discuss various topics. I proposed a group management model for distance education on a network, and studied management algorithm and MIB (Management Information Base), which are required to manage virtual classes in distance education systems.

1. 원격교육 관리의 필요성

최근 ATM과 같은 초고속망이 실현되면서 이를 위한 많은 응용서비스와 운용되는 서비스 및 망관리를 위한 다양한 연구가 이루어지고 있다. 특히, 초고

*이 논문은 한국과학재단의 지원으로 이루어졌음.

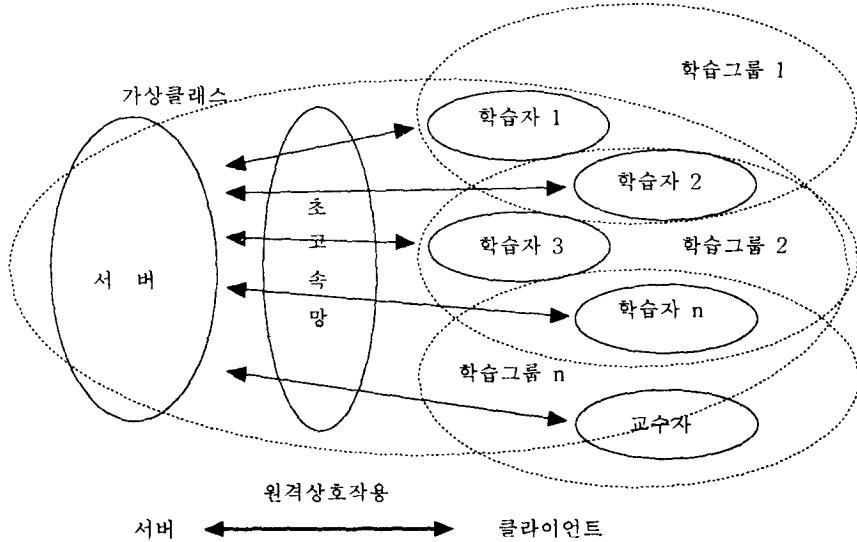
속망의 많은 응용 중에서 원격교육 서비스는 파급효과가 크고 국민생활을 직접적으로 변화시키는 분야이기 때문에 선진국에서도 별도의 많은 계획들을 추진하고 있다. 대다수의 원격교육 모델에서는 원격지의 학습자가 원하는 학습 프로그램을 선택하여 개별적으로 학습하고, 통신망을 통한 실시간 질문/응답을 실행하는 환경을 제공한다. 즉, 개별학습 ITS에 통신기능과 WWW 브라우저 접속 기능을 갖는 시스템 형태로 발전하고 있다. 멀티미디어 전송 환경의 초고속망을 이용한 가상교실 (Virtual Classroom) 등의 원격교육시스템은 우리나라와 같이 과밀학급, 산간벽지 및 도서 학교가 많고, 높은 교육열을 갖고 있는 실정에서는 국가적 중대 사업으로 완수해야 할 분야이다[1][5]. 초고속망을 활용한 원격교육 모델에서는 실시간/다자간 형식의 교육 모델을 통한 교수/학습방법이 일반화될 것이다. 이러한 교육 모델에서는 통신망을 이용한 다양한 교수/학습장(이하 가상클래스라고 정의한다)이 존재할 것이고, 보다 상위 수준의 원격교육에서는 이러한 클래스들의 구성요소도 더욱 복잡해질 것이다. 분산환경의 완전한 원격교육 모델에서는 BBS, 화상강의 등의 서비스뿐만 아니라, 참여자 상호간의 협조작업을 위한 그룹웨어 기능 등도 지원되어야 하는데, 이러한 분산환경 모델에서 중요한 문제가 참여자 및 작업 그룹 관리이다. 가상클래스에서는 교수/학습에 참여한 교수/학습자들의 상태 관리, 공유 객체그룹 관리, 참여자 사이의 메시지 전송 등이 효율적으로 관리되고 지원되어야 한다[3],[5]. 이러한 트래픽 관리가 효율적으로 이루어질 때, 결국 망의 부하를 줄일 수 있고 전체의 작업 효율을 높이면서 현장감 및 일체감을 높여서 학습효과를 더할 수 있다. 따라서 초고속망의 원격교육 응용 서비스에서는 이러한 가상클래스를 관리하는 새로운 모델 및 관리기술 문제가 중요한 핵심 요소가 될 것이다. 본 논문에서는 초고속망의 원격교육 응용서비스를 지원하기 위한 다양한 기술 중에서 실시간, 다자간 교육 모델에서 동적으로 생성되고 소멸되는 가상클래스를 관리하기 위한 관리 시스템에 관하여 연구하였다. 효율적이고 체계적인 클래스 관리는 원격교육 응용서비스의 다양성과 품질을 향상시킬 것이며, 멀티미디어로 구성되는 복잡한 분산 정보들의

관리를 보다 단순화시킬 수 있다.

본 논문의 가상클래스의 그룹 관리 모델에서는 클래스 상에 동적으로 발생하는 교수자/학습자의 다양한 주제별, 등급별에 따른 참여자 그룹 관리를 중점적으로 연구하였다. 가상클래스란 원격교육시에 교수자와 학습자들간의 연결을 추상화한 객체로서 통신 프로토콜을 기반으로 하여 호스트들간의 메시지 송수신 채널 기능을 수행하는 공간이다. 이러한 클래스를 구성하는 참여자 그룹의 관리 기법은 다양한 측면과 요소를 고려하여야 한다. 클래스는 교수자와 학습자의 연결설정에 따라 유동적으로 객체가 생성 혹은 소멸될 수 있다. 물리적인 하나의 교실에서는 교수자와 하나의 학습자 집단으로 구성되는 유일한 학습그룹만이 존재한다. 그러나 통신망을 이용한 원격교육 모델에서는 하나의 가상클래스 공간에 여러 개의 학습그룹 (Learning Group)을 생성할 수 있도록 모델링이 가능하다. 그 기준은 학습 주제가 될 수도 있고, 같은 학습 주제 내에서 수준별에 따른 그룹화일 수도 있다. 이러한 가상클래스 내의 다중 그룹들을 효율적으로 관리하는 것이 학습효과를 높이고, 통신망의 부하를 줄일 수 있는 방법이 된다. N 개의 다중 목적지로 구성된 그룹에 동일 정보를 전송하는 통신유형을 멀티캐스트 통신이라고 하는데, 이것은 최근의 각 시스템에서 많이 활용되는 서비스이다[6],[7],[11],[12],[13]. 원격교육에서 참여자의 학습그룹은 학습정보 메시지의 송수신 주체가 되는 일련의 프로세스 그룹이다. 즉, 이것은 원격교육 서비스에서 멀티캐스트 전송을 필요로 하면서 가상클래스를 구성하는 기본 학습단위로 볼 수 있다. 물론, 하나의 참여자 그룹은 네트워크 상의 한 호스트에 존재할 수도 있고, 여러 호스트 상으로 분산될 수도 있다. 그리고, 특정 참여자가 동시에 한 개 이상의 학습그룹의 구성원이 될 수도 있다. 여기서 실시간/다자간 환경에서 학습활동 및 학습정보 송수신의 효율성을 위하여 가상클래스 상의 참여자 집합(학습그룹)을 동적으로 관리할 필요가 있다[8],[12],[16]. 이를테면, 같은 주제의 학습그룹 내에서 등급별로 세분된 또 다른 하위 그룹들로 구분할 수 있을 때, 저급의 학습 메시지를 고급 등급에 속한 학습자들에게도 멀티캐스팅하는 것은 네트워크 효율상, 학습자 능력상 모두

비효율적이다. 이러한 원격교육 모델의 효율성을 위해서는 구성원들의 학습그룹 관리가 체계적으로 이루어져야 한다. 즉, 가상클래스의 학습그룹을 구성

이 있을 수 있다. 그룹 관리 기능은 이러한 참여자 행위에 따른 각 상태 변화를 관리할 수 있어야 한다. 이와 같이 분산환경의 학습자, 교수자들이 시공간적



<그림 1> 가상클래스 환경

하는 참여자들 상호간의 메시지 배달, 관리, 장애 감지, 새로운 참여자의 가입, 탈퇴, 그룹의 생성 및 삭제 등이 효과적으로 지원되어야 한다[8],[9]. 참여자 학습그룹이란 하나 혹은 분산된 시스템 상에서 존재하는 하나 이상의 참여자로 구성된 집단을 말한다. 따라서 한 참여자 그룹은 하나의 논리적 엔티티로서 정의될 수 있으며, 이 엔티티 상에서 구성원 상호간의 그룹통신이 제공되어야 한다[1],[6]. 그룹 관리 시스템은 원격교육 참여자와의 인터페이스를 담당하는 기능, 참여자 그룹의 상태 변화를 관리하는 그룹 관리 기능 등으로 구분할 수 있다. 이러한 그룹 관리 시스템은 다양한 관리 오퍼레이션을 요구한다. 그룹 관리 기능 오퍼레이션에는 그룹 구성원들의 가입, 탈퇴 등에 따른 관계 변화를 관리하여 그룹의 참여자 상태를 저장하고 구성원에게 통지하는 것 등이 있다. 참여자의 구성원 관계, 상태 등을 변화시키는 오퍼레이션에는 새로운 학습그룹의 생성, 학습그룹의 구성원 가입 및 탈퇴, 그룹변경, 그룹삭제, 장애 감지 등

개념을 초월하여 실시간/다자간 형태의 효율적인 가상클래스를 유지하기 위해서는 이들 구성원들뿐만 아니라, 그들 사이에서 형성되는 다양한 가상공간들을 효율적이고 체계적으로 관리하는 문제가 매우 중요한 설계요소이며 쟁점이다. 본 연구에서는 초고속망의 원격교육 모델에서 필수적으로 동반되는 동적이면서 다양한 가상클래스 상의 학습그룹들을 효율적으로 관리하기 위한 모델 정립과 관리 알고리즘, 관리데이터베이스의 정보 구조 등을 설계하였다.

2. 원격교육 모델

본 장에서는 가상클래스에서 요구되는 교수자 및 학습자의 교육, 학습 활동을 기본으로 한 모델을 설정하고 기본 기능들을 정의한다. <그림 1>은 본 논문에서 제안하는 가상클래스의 운영 환경으로서 고속망 상에서 분산되어 있는 여러 명의 학습자와 교수자 사이의 학습모델이다.

2.1 클라이언트

본 논문의 원격교육 모델에서 클라이언트 시스템은 크게 학습자용 시스템과 교수자(관리자) 시스템으로 구분하였다. 학습자용 시스템은 원격서버 시스템이 제공하는 기본적 학습 메뉴를 이용하여 학습을 진행하고, 임의의 학습그룹에 가입하여 특정 학습자와 토론할 수 있으며, 교수자와도 대화할 수 있는 인터페이스를 제공한다. 교수자용 클라이언트 시스템은 서버 시스템에 접속하여 학습 데이터베이스를 유지 관리할 수 있는 인터페이스를 제공한다. 그리고, 관리자는 관리자용의 클라이언트 시스템을 통하여 원격교육 상의 가상클래스에 발생하는 다양한 관리 객체에 접근할 수 있다. 임의의 가상클래스에서 교수자(관리자)는 필요에 따라 원격지의 학습자에게 임의의 학습 정보와 관리정보를 전송할 수 있다. 이 때 교수자는 지정된 한 학생에게만 임의의 정보를 보낼 수도 있고, 여러 명의 학습자로 구성된 임의의 그룹으로 동시에 동일한 정보를 멀티캐스팅할 수도 있다. 그리고 특정 학습그룹의 전체 학습자에게도 동시에 지정된 메시지를 브로드캐스팅할 수도 있다.

2.2 서버

서버 시스템은 가상클래스의 생성 및 소멸, 속성의 관리, 원격지 클라이언트로부터의 연결요구 처리, 접속 허가된 학습자의 정보 관리, 학습이력 관리, 새로운 학습자의 인적 및 관련된 학습정보 관리, 가상클래스 상의 메시지 전송 및 배달기능, 통지 기능 등을 갖는다. 이러한 서버 시스템은 멀티미디어 학습자료를 제공하는 웹서버와는 분리하여 구성하였는데, 따라서 관리자로부터 관리 요구를 전달받아서 가상클래스 상의 다양한 이벤트 및 객체를 관리하는 관리 에이전트 시스템 역할을 하는 것이다. 그리고 서버는 학습자 및 교수자 클라이언트 상호간의 학습 메시지 전송을 위한 중개 채널을 생성시키는 일을 한다. 서버와 클라이언트들은 지정된 액세스 포인트를 통하여 서버와 클라이언트, 클라이언트와 클라이언트 상호간의 통신채널을 형성한다. 이러한 통신 프

로토콜의 실험 모델은 TCP/IP를 기반으로 하였으며, 소켓(Socket) 인터페이스를 이용하여 액세스 포인트를 생성한다.

3. 가상클래스의 관리정보

3.1 관리자 모델

분산환경의 가상클래스에는 관리되어야 하는 다양한 피관리 객체(MO : Managed Object)들이 존재한다. 본 절에서는 이러한 피관리 객체들의 구성요소와 관리자들에 관하여 기술한다. ATM과 같은 스위칭 모델을 채택한 초고속망은 시스템의 방대함과 복잡성으로 인하여, 현재 TMN(Telecommunications Management Network) 등의 망관리 연구가 매우 활발하게 진행되고 있다. 이러한 망관리 시스템은 대부분이 관리자(Manager)와 관리 대행자(Managing Agent)로 구성되는 분산시스템 구조로서, 망과 관련된 다양한 피관리 객체 정보를 수집하여 관리자로부터 전달된 관리 정보를 처리하는 시스템으로 발전하고 있다. 본 논문에서는 이러한 망관리 시스템과는 별도로 초고속망의 응용 서비스인 가상클래스 관리를 연구하는데, 먼저 가상클래스 상에서 발생하는 피관리 객체들의 구성 요소와 학습그룹 및 그들의 관리자에 관하여 설명한다.

- 작업(Task) 관리자 : 일반적으로 망 관리 환경에서의 작업 관리자는 실행되는 여러 작업 프로세스들을 관리하는 임무를 갖고 있다. 본 가상클래스 상의 작업 관리자는 가상클래스의 생성과 등록, 운용, 소멸을 관리하는 관리자이다. 작업 관리자는 감독자로부터 명령을 전달받아 가상클래스의 생성 및 등록, 그리고 실행과 소멸 등을 담당한다.

- 트렌드(Trend) 관리자 : 트렌드 관리자는 분산환경의 가상클래스 상에서 발생하는 다양한 이벤트 및 정보량을 체크, 보고하는 관리자이다. 가상클래스 상에서 임출력되는 데이터들을 실시간으로 수집하여 관리자에게 보고하여 가상클래스 상의 트렌드를 감

시할 수 있게 한다. 트랜드 관리자는 관리자의 지시에 따라 정보량을 적절히 조절할 수 있다. 따라서 가상클래스 상의 학습그룹을 구성하는 구성원의 임계치를 갖고 접속 거부 등의 적절한 관리 메시지를 발생시킬 수 있다.

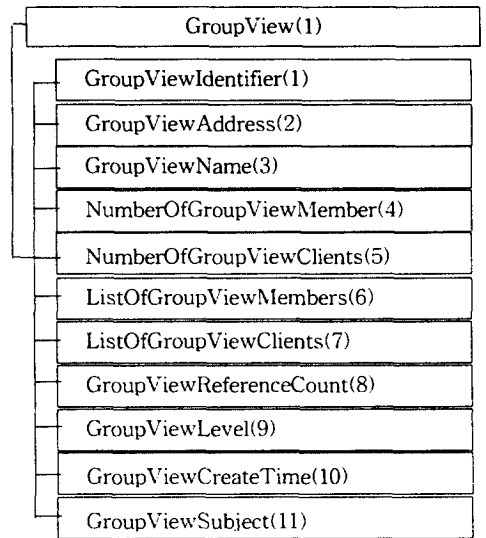
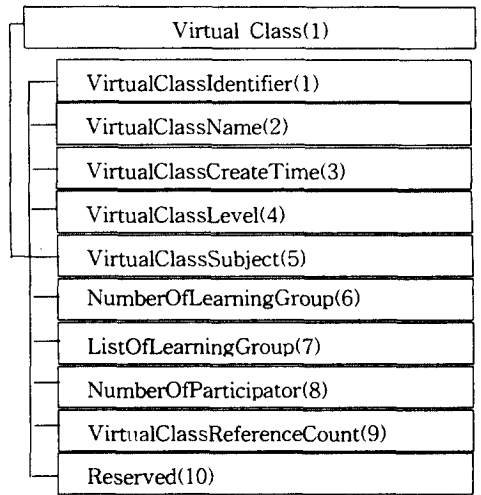
- 로그 (Log) 관리자 : 로그 관리자는 가상클래스 환경에서 발생하는 참가자들의 다양한 이력 정보 (Historical Information)를 저장하고 관리하는 역할을 수행한다. 로그 관리자는 축적된 이력 데이터를 관리자의 요구 사항에 따라 가공하여 아스키 파일 형태로 보고한다. 이력 데이터에는 가상클래스의 접속 시간, 종료 시간, 전체 학습시간, 접속자의 신분, 접속자의 위치, 최근 학습 데이터 정보, 접속자의 전자우편 주소 등이 기록된다.

- 학습그룹 (Learning Group) 관리자 : 가상클래스 상의 여러 학습그룹들의 생성과 소멸, 학습자 및 교수자의 학습그룹 참여 및 탈퇴, 임의의 학습자 프로세스의 제거, 학습정보의 송수신 관리 등을 담당하는 관리자이다.

Process Group)은 관리대상이 되는 일련의 학습자 집단으로서, 하나의 학습그룹은 네트워크 상의 한 호스트에 존재할 수도 있고, 여러 호스트 상으로 분산될 수도 있다. 그리고, 특정 학습자 프로세스가 동시에 한 개 이상의 학습그룹의 구성원이 될 수도 있다.

3.2 관리정보베이스(MIB)의 구성

본 절에서는 가상클래스의 많은 피관리 객체 중에서 학습그룹을 형성하는 참여자들을 중심으로 한 관리정보베이스 (MIB:Management Information Base)의 구성 요소들을 정의한다. <그림 2>는 가상클래스의 관리정보베이스 구조를 나타내고 있다. 가상클래스 객체에는 가상클래스 식별자, 가상클래스 이름, 생성 시간, 클래스의 레벨과 주제, 클래스내의 학습그룹 수, 학습그룹 리스트, 참여자 수, 참조 수 등의 정보가 저장된다. 그리고, 하나의 학습그룹에 관계된 정보를 갖는 그룹 뷰 객체는 그룹 뷰 식별자, 그룹 뷰 이름 등을 비롯하여 학습 구성원의 수, 클라이언트 정보 등을 보유하고 있다.



<그림 2> 관리정보베이스의 구성 예

4. 그룹 관리 시스템

가상클래스에서 학습 프로세스 그룹 (Learning

가상클래스를 효율적으로 관리하기 위해서는 피관리 객체인 학습그룹을 구성하는 구성원 요소들의 관리가 체계적으로 이루어져야 한다[6],[8]. 즉, 학습그

를 구성하는 프로세스들 상호간의 메시지 배달, 관리, 장애 감지, 새로운 학습자 프로세스의 가입, 탈퇴, 학습그룹의 생성 및 삭제 등이 효과적으로 지원되어야 한다. 또한 학습그룹의 구성 요소들이 실존하는 호스트들의 집합으로 이루어진 호스트 그룹이 잘 관리되어야 한다.

한편, ISO에서는 OSI 관리 표준화 작업을 진행하고 있다. OSI 관리는 5 부분으로 나누어 표준화 작업이 진행되는데, 망 구성 자원들의 속성과 특성 값을 관리하는 구성 관리, 통신 자원 사용에 관한 요금 계산 기능을 갖는 계정 관리, 망의 장애 처리 및 회복 기능을 수행하는 장애 관리, 망의 성능을 측정 감시하는 성능 관리, 망 접근에 관한 보안 관리 등으로 구성되어 있다[17]. 응용 층의 OSI 관리 서비스를 위한 특정 관리기능 영역 중, 객체 관리 (Object Management) 등의 분야에서 넓은 의미의 모든 망 자원들의 변화, 속성 관리 등에 관하여 표준화 작업을 진행 중이다. 본 논문에서의 그룹 관리는 제안된 가상클래스 모델 내에서 학습그룹을 지원하기 위한 그룹 관리로 한정하며, 학습그룹과 관련된 동작들은 ISO / IEC의 그룹 통신 표준화 및 TCP/IP의 RFC (Request for Comments) 권고 문서를 참조하여 정의하였다[2],[17].

4.1 학습그룹 관리 시스템(LGMS) 구조

학습그룹이란 하나 혹은 분산된 호스트의 가상 공간에 존재하는 하나 이상의 프로세스로 구성된 집단을 말한다. 따라서 한 학습그룹은 하나의 논리적 엔티티로서 정의될 수 있으며, 이 엔티티 상에서 구성된 상호간의 그룹 통신이 가능하다. 학습그룹 계층은 프로세스 레벨의 통신을 담당하는 계층으로서 프로세스 상호간의 통신, 장애 감지, 변경 등을 효율적으로 처리할 수 있는 기능이 있어야 한다.

본 장에서는 프로세스 상호간에 발생하는 다양한 오퍼레이션을 수행, 관리할 수 있는 학습그룹 관리 시스템 (LGMS : Learning Group Management System)을 설계하고 각 기능을 담당하는 함수를 정의한다. LGMS의 중요 기능은 응용 프로세스와의 인터페이스를 담당하는 기능, 가상클래스내의 학습그룹

의 상태 변화를 관리하는 그룹 관리 기능 등으로 구분할 수 있다.

그리고 학습그룹 관리 기능은 학습그룹을 호스트 그룹별로 재구성하여 호스트 그룹 테이블이 효율적으로 유지될 수 있도록 지원한다. 즉, 학습그룹들이 단일 식별자들로 구분되어 학습그룹 테이블에 저장되는 반면, 각 프로세스들이 실제로 존재하는 네트워크 상의 호스트 그룹을 별도로 구성하여 호스트 그룹 계층으로 전송한다. 또한 학습그룹 관리 기능은 특정 호스트로 전송되어 온 그룹별 멀티캐스트 메시지를 학습그룹 테이블을 이용하여 그 호스트 상에 존재하는 여러 프로세스에게 분배하는 것이다.

4.1.1 응용 프로세스 인터페이스

먼저 가상클래스 모델에서 클래스 관리를 위한 메시지 구조 및 기본적 함수를 정의한다. 가상클래스에서 하나의 학습 환경을 구성하는 요소 설정이 필요한데, 뷰(View)는 한 학습그룹을 형성하는 구성원의 정보 리스트를 말한다. 그룹 g 에 대한 뷰 시퀀스는 다음과 같은 조건을 만족하는 리스트 즉, $VIEW_0(g), VIEW_1(g), \dots, VIEW_n(g)$ 이다.

- (1) $VIEW_0(g) = \emptyset$
- (2) $\forall i : VIEW_i(g) \subseteq P, P$ 는 모든 프로세스의 집합
- (3) 단순화를 위하여 $VIEW_i(g)$ 와 $VIEW_{i+1}(g)$ 는 단 하나의 학습자 프로세스를 감하거나 더하는 것으로 가정한다.

이러한 뷰 조건아래 가상클래스의 분산 환경에서 응용 프로그램이 호출하는 학습그룹 관리 오퍼레이션은 표 4-1과 같다. 그리고, 학습그룹 관리 시스템의 학습그룹을 구분하는 식별자(Process Group Identifier, Group_Id)는 다음과 같은 구조체로 정의한다.

```
struct identifier
{
    short address_process; /*(UNIX) process
```

```

        identifier */
short address_port; /*(UDP) port number*/
short address_host; /* host identifier */
u_char address_entry; /* entry point to
        invoke */
u_char address_res; /* reserved */
}
    
```

위의 Identifier 구조체에서 address_host 값은 각 호스트를 구분하기 위한 식별자이고, UNIX Process Identifier 값은 getpid() 함수를 이용한다. 한편, address_port는 UDP 포트 번호로 구성하였다.

<표 1> 학습그룹 관리 함수

그룹 기능	관리 함수
그룹 생성	Group_Id = Pg_create (gname)
그룹 탐색	Group_Id = Pg_lookup (gname)
정보 추출	groupview=Pg_getview(Group_Id)
그룹 가입	Group_Id = Pg_Join (gname)
그룹 탈퇴	Ret_code = Pg_leave (Group_Id)
그룹 제거	Ret_code = Pg_delete (Group_Id)

한편, 학습그룹 테이블을 구성하는 엔트리는 다음의 그룹 뷰 구조와 같고, 특정 학습그룹에 관한 정보는 학습그룹 테이블의 그룹 뷰를 호출하여 알 수 있다.

```

/* groupview */
#define pg_len number
typedef struct
{
    int gv_viewid; /*group view number*/
    identifier gv_gaddress; /* group address*/
    char gv_gname[pg_len]; /*group name*/
    short gv_nmem;
        /*number of group members*/
    short gv_nclient; /* number of clients */
    identifier gv_members[pg_len];
        /* list of group memebbers */
    
```

```

        identifier gv_clients[pg_len];
        /* list of clients */
    int gv_reference; /*group reference count*/
    int gv_level; /* group level */
} groupview;
    
```

위에서 정의된 학습그룹 테이블의 그룹 뷰 엔트리 정보, 관리 함수 등을 이용하여 관리 함수는 외부로부터 관리 데이터를 입력받아서 여러 가지 동작을 행하게 된다. 관리 시스템에게 특정 오퍼레이션 요구가 발생할 때마다, 거기에 해당되는 관리 함수에 의하여 그룹 테이블의 엔트리가 변경, 수정된다.

한편, 상위 층에서의 학습그룹에 대한 전송 단위는 메시지이다. 따라서 이러한 전송 메시지의 처리를 위한 함수들이 제공되어야 한다. 다음은 멀티캐스트 전송을 위한 메시지 구조이다.

```

type struct
{
    message *mesg_next; /* 메시지 리스트를
        관리하기 위한 포인터 */
    int mesg_length; /* 메시지 길이 */
    int mesg_refers; /* 해당 메시지의 참조 수*/
    int head_mesgid; /* 메시지 식별자 */
    int head_length; /* 메시지 헤더 길이 */
    identifier head_dest; /*목적지 그룹 주소값*/
    int head_timest[]; /* 메시지 타임스탬프 */
    identifier head_sender; /* 메시지 송신자 */
    u_char *mesg; /* data message */
} message;
    
```

여기서 주어진 메시지 구조에 대한 별도의 메시지 처리 함수와 그룹 관리 함수가 정의되어야 하는데, 먼저 Pg_create(g)는 단일 학습그룹 g의 생성을 실행하는 함수이며, mesg_create(m)는 메시지 m의 생성을 실행하는 함수이다. 구체적으로 위에서 정의한 메시지 구조체를 이용하여 메시지 처리를 행하는 관련 함수는 다음과 같다.

```

message *mesg_create(); /*새 메시지의 생성*/
    
```

```

msg_delete(mp)
    message *mp; /* 해당 메시지를 제거 */
identifier *msg_getsender(mp)
    message *mp; /* 메시지의 전송자의 주소 값을 구한다 */
identifier *msg_getdests(mp)
    message *mp /* 메시지의 전송 목적지 리스트를 구한다 */
msg_put (mp, form)
    message *mp, char *form; /* 빈 메시지의 해당 필드에 메시지 값을 넣는다 */
msg_get(mp, form)
    message *mp, char *form; /* 메시지의 포인터 mp로 부터 해당 문자열을 찾는다 */
    
```

그리고, 학습 및 관리 메시지 송수신에 관련된 학습그룹, 메시지 등의 여러 가지 정보를 저장하기 위한 방법으로 큐 구조를 정의한다. 큐 노드 구조는 다음과 같으며 각 함수들의 실행 시에 큐 노드의 값을 참조하고 변경하여 데이터 처리를 실행한다.

```

typedef struct
{
    qnode *qu_next; /* next pointer */
    qnode *qu_last; /* previous pointer */
    int qu_flag; /* flag 비트 */
    int qu_name; /* 큐 노드 이름 */
    identifier *qu_pname; /* 프로세스 그룹 주소 */
    groupview *qu_groupv; /* 그룹 뷰 정보를 저장하기 위한 구조체 */
    message *qu_mesg; /* 메시지 정보를 저장하기 위한 구조체 */
    short *qu_hid; /* 호스트 식별자 */
} qnode;
    
```

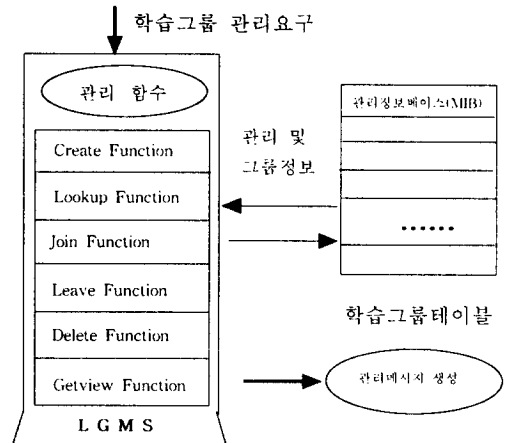
한편, 위의 큐 노드 구조를 생성, 삭제, 관리하는 별도의 함수들이 필요하며, 이들은 이중 연결 리스트 관리 알고리즘과 같다. 다음은 큐 노드를 관리하기 위한 함수들이다.

```

qu_alloc(qnode *qp); /* 새로운 큐 노드를 위한 메모리 확보한다 */
qu_append(qnode *qp, *np); /* 큐 포인트 qp에 새로운 큐 node np를 추가한다 */
qu_free(qnode *qp); /* 큐 노드 qp에 대한 메모리를 해제하며 큐를 제거한다 */
qu_add(qnode *qp, int name, qnode *data); /* 지정된 큐 노드에 data를 기록한다 */
qu_freeall(qnode *qp); /* 큐 포인트 qp의 모든 메모리를 해제한다 */
qu_find(qnode *qp, int item); /* 큐 포인트에서 해당되는 큐 노드의 위치를 찾는다 */
qu_sort(qnode *qp) /* 큐 리스트의 재배열 */
pg_find(qnode *qp, identifier *pname); /* 지정된 큐 포인트에서 주어진 프로세스 이름과 비교하여 해당 위치를 찾는다 */
    
```

4.1.2 학습그룹 관리 모델

그룹 관리 기능은 학습그룹 구성원들의 가입, 탈퇴 등에 따른 관계 변화를 관리하여 그룹의 학습자 상태를 저장하고 다른 구성원에게 통지한다. 이러한 개념의 학습그룹 관리 기능 모델을 <그림 3>에 나



<그림 3> 학습그룹 관리 기능 모델

타내었다. 컴퓨터 네트워크 환경의 각 호스트 상에

존재하는 학습그룹은 수시로 동적으로 변화하기 때문에 특별한 학습그룹 관리 기능을 수행할 필요가 있다. 학습그룹 관리 기능은 그룹의 상태나 구성원 관계를 변화시키는 모든 오퍼레이션의 영향을 받는다.

프로세스 구성원 관계, 상태 등을 변화시키는 오퍼레이션은 새로운 그룹의 생성, 구성원 가입, 탈퇴, 그룹 삭제, 장애 감지 등이 있다. 따라서 이러한 오퍼레이션 요구가 발생할 때마다 그룹 관리 기능은 학습그룹 테이블을 수정해야 한다. 학습그룹 테이블을 구성하는 각 멀티캐스트 학습그룹은 단일 식별자로 구분되는 여러 가지 정보들을 포함하고 있다. LGMS는 프로세스 레벨의 그룹들을 생성, 유지, 관리하는 시스템으로 학습그룹 테이블과 이것을 관리하는 관리 함수들로 구성된다.

4.2 그룹 관리 함수

본 절에서는 학습그룹 관리 시스템을 구성하는 관리 함수의 기능을 정의한다.

(1) 그룹 탐색 (Pg_lookup(gname))

Pg_lookup()은 가상클래스 내에 존재하는 학습그룹 이름 gname의 주소 값을 결정하는 함수이다. 그룹 이름은 이미 생성된 학습그룹이다. 다음은 pg_lookup(char *gname)의 실행 과정이다.

```

identifier *Pg_lookup(char *gname)
{
    qnode *qp; /* 큐 노드 포인터를 선언 */
    message *mesg;
    groupview *result;
    while( ) /* 요구 발생 */
    { if ( strcmp
        (qp->qu_groupv.gv_gname, gname))
        /* 학습그룹 정보를 저장하고 있는 큐
        노드에서 해당 그룹을 찾는다 */
        return(qp -> qu_pname);
    }
}
    
```

```

mesg = mesg_create(gname); /* 그룹 이름
    gname을 데이터로 하는 메시지 생성*/
if (broadcast (mesg, result, LOOKUP) < 0)
/*다른 호스트와 통신하여 gname을 찾는다*/
return(NULLADDRESS);
else qu_add( ) /* 큐 구조에 정보 저장*/
return(result -> gv_gaddress);
}
    
```

(2) 학습그룹 정보 추출 (Pg_getview(Group_Id))

Pg_getview() 함수는 지정된 학습그룹에 관한 뷰 구조를 제공하는 함수이다. 학습그룹 뷰는 해당되는 그룹의 여러 가지 정보를 갖는 구조로서 뷰 번호, 그룹 이름, 학습자 수, 학습 구성원 리스트 등을 제공한다.

```

groupview *Pg_getview(Identifier *Group_Id)
{
    groupview *gv, *result;
    message *mesg;
    gv = pg_getlocalview(&Group_Id)
    /* 현 호스트 상에서 주어진 그룹을 찾아
    그룹 뷰 정보를 구한다*/
    if (gv > 0) return(gv);
    mesg = mesg_create(Group_Id) /*그룹 식별자
    Group_Id를 데이터로 하는 메시지 생성*/
    if (broadcast(mesg, result, GETVIEW)>0)
    /* 다른 호스트 상으로 통신하여 해당되는
    그룹 뷰 정보를 넘겨받는다 */
    return(&result);
    else
    return(NULL);
}
    
```

(3) 그룹 생성 (Pg_create(gname))

Pg_create()는 새로운 학습그룹의 생성을 실행한다. gname은 생성하는 학습그룹 이름이며 Group_Id는 프로세스 그룹의 식별자이다. 그룹 식별자는 학습

그룹 시스템에서 유일한 값을 갖는다. 즉, 그룹 생성 함수는 주어진 그룹 이름으로 새로운 학습그룹을 생성하는 관리함수이다.

```

identifier *Pg_create(char *gname, identifier
*mlist)
{
    identifier *gid;
    groupview *pg;
    pg=pg_alloc( ); /*그룹을 생성하기 위한 메모리
공간을 확보한다.*/
    gid=make_gid( ); /*그룹 식별자를 생성한다.*/
    make_groupview( ); /*구성원 리스트 mlist,
주소 값 등을 이용하여 그룹 뷰를 완성한다.*/
    gid->gv_viewwid = 1; /* 그룹 식별자의 뷰
번호를 최초로 한다.*/
    qu_add( ); /* 생성한 그룹 정보를 큐 노드에
저장시킨다. */
    return (gid);
}
    
```

(4) 그룹 가입 (Pg_join(Group_Id))

Pg_join() 함수는 그룹 이름 gname으로 주어지는 학습그룹으로의 가입을 실행한다. 함수의 성공 시에는 학습그룹의 식별자 값이 구해진다. 가상클래스에서 학습자는 해당 학습그룹에 먼저 가입하여야 한다. 그러나, 가입을 원하는 프로세스가 최초의 구성원이면 신규 학습그룹이 생성된다.

```

identifier * Pg_join( )
{
    identifier Group_Id ;
    char *gname ;
    groupview *gv ;
    message *mesg ;
    Group_Id = Pg_lookup (gname) ; /*그룹 이름
gname 을 추출하여 그룹 식별자를 구함*/
    if (Group_Id=NULL)
        (Group_Id=Pg_create( ) ;
    
```

```

return(Group_Id); /*그룹 식별자가 없으면
신규 학습그룹을 생성한다.*/
}
gv=Pg_getview(Group_Id); /* 그룹 식별자를
이용하여 학습그룹 뷰 정보를 취한다.*/
Modify_groupview( ); /*그룹 정보의 변경*/
return( Group_Id) ;
}
    
```

(5) Ret_Code = Pg_leave(Group_Id)

Pg_leave()는 특정 학습자 프로세스를 Group_Id 식별자를 갖는 학습그룹으로부터 탈퇴시키는 함수이다. 학습그룹의 구성원인 임의의 한 프로세스가 정상적으로 실행을 끝마쳤거나 혹은, 장애 발생 등으로 실행 종료되었을 때, Pg_leave() 함수가 호출되는데 이는 종료된 학습자 프로세스가 구성원인 모든 학습그룹에서 일어난다.

(6) Ret_Code = Pg_delete(Group_Id)

Pg_delete()는 지정된 학습그룹의 제거를 실행한다. Pg_delete()함수가 호출되고 나면 해당되는 Group_Id의 모든 자료는 소멸된다. 다음에 다시 생성되더라도 다른 학습그룹 식별자를 갖게된다.

(7) Ret_Code =Pg_multicast(Group_Id, message)

Pg_multicast()는 지정된 학습관리 메시지를 가상클래스 상에서 형성된 학습그룹으로의 전송을 실행하는 함수이다.

4.3 관리 알고리즘

앞의 4.2절에서 LGMS가 갖고 있는 프로세스 레벨의 학습그룹 생성, 유지, 관리 함수들을 정의하였다. 주어진 그룹 관리 시스템에서는 임의의 한 프로세스가 Pg_join() 오퍼레이션을 실행할 때마다 학습그룹 테이블이 탐색되어 해당되는 학습그룹의 구성원으로서 추가된다. 만일 해당되는 그룹 엔트리가 존

재하지 않으면 다시 Pg_create() 오퍼레이션이 실행되고 그 프로세스는 생성된 그룹의 최초 구성원이 된다. 같은 방법으로 Pg_leave() 오퍼레이션이 실행되면 학습그룹 테이블의 해당되는 프로세스가 제거되고, 만일 Pg_leave() 실행으로 제거되는 프로세스가 해당 그룹의 마지막 프로세스일 때는 Pg_delete() 오퍼레이션이 호출되어 그룹 엔트리가 삭제된다. 학습그룹 관리 시스템이 행하는 관리 절차는 다음과 같다.

< 관리 알고리즘 >

```
while (request_event) /*관리요구 데이터 수신*/
{
    switch ( type of request_event)
    {
        case join : Pg_join();
            /* 특정 학습그룹으로의 가입 */
            if (Pg_join() == -1)
                Pg_create(); /*지정그룹이 존재하지
                않으면 새로운 그룹 생성 */
            break;

        case getview : Pg_getview() /*지정된
            학습그룹의 정보 추출*/
            break;

        case leave : Pg_leave(); /*호출 프로세스를
            해당 그룹으로부터 제거 */
            if (groupview.gv_nmem == 0)
                Pg_delete(); /*구성원이 더 이상 존재
                하지 않으면 해당 학습그룹을 테이블
                에서 삭제*/
            break;

        case lookup : Pg_lookup();
            /*특정 학습그룹 어드레스 탐색*/
            break;

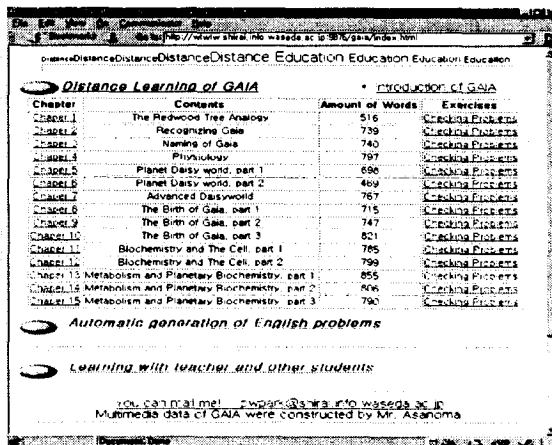
        case multicast : Pg_multicast(); /*지정된
```

```
학습그룹에 멀티캐스트 전송*/
break;
```

```
default : break;
    }
}
```

5. 적용 예

본 논문에서 제안된 가상클래스의 그룹 관리 시스템을 사례연구의 일환으로 인터넷 상에서 원격 영어 학습을 지원할 목적으로 개발한 하나의 원격교육 시스템에 적용하였다[14]. 구현된 응용 서비스는 인터넷 상에서 원격 영어학습을 지원해 주는 시스템으로 <그림 4>에서 시스템의 초기 접속 화면을 보여주고 있다. 구현된 원격 영어학습 시스템은 "GAIA (The Practical Science of Planetary Medicine, by Jim LovelockJim Lovelock)"라는 하나의 과학서 내용을 WWW 상에서 멀티미디어 자료화하여 교수/학습 기능을 구현하였는데[14], 여기에 가상클래스의 관리 기능을 추가하였다. 주교재 GAIA는 지구의 진화에 관련된 과학의 서로 다른 부분들을 통합한 하나의 이론이다. 시스템의 멀티미디어 데이터는 전체가 모두 15장으로 구성되어 있으며, 원어민의 음성 파일을 포함한 하이퍼미디어 구조로 되어 있다.



<그림 4> GAIA 시스템의 실행 화면

그리고, <그림 5>는 가상클래스 상에서 교수자가 하나의 학습 그룹 (GAIA_Learning)을 탐색, 선택하여 그룹의 구성원들에게 특정 학습 메시지를 멀티캐스트 전송하는 부분이다. 한편, <그림 6>은 GAIA 시스템 상에서 교수자가 특정 메시지를 통지하기를 원하는 하나의 그룹을 형성하여 해당 메시지를 전송하는 예이다.

```

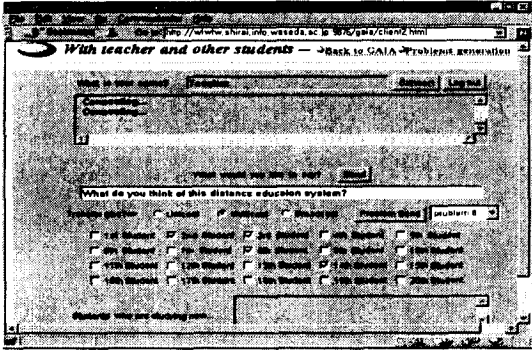
.....
learning_init(1702);
identifier *Group_Id;
.....
message_sending ( )
{
    message *mesg_p;
    Group_Id= Pg_lookup ("GAIA_Learning");
    mesg_p = mesg_create(m);
    Pg_multicast (Group_Id, mesg_p);
    .....
}
    
```

<그림 5> 학습 그룹의 탐색과 메시지 전송

관리 메카니즘에 관하여 연구하였는데, 원격교육 환경에서 생성되는 가상클래스의 그룹 관리 시스템을 정의하고 관련된 프리미티브 및 알고리즘을 연구하였다. 그리고, 가상클래스를 관리하고자 할 때 발생하는 다양한 피관리 객체에 관한 관리정보베이스 구조를 정의하였다.

그리고, 본 논문에서는 가상클래스의 통신 모델을 크게 학습그룹 계층, 전송 계층, 호스트 그룹 계층으로 구분하였는데, 제안된 가상클래스 모델의 학습그룹 계층에서 생성되는 그룹 관리 시스템을 연구하였다. 가상클래스의 학습그룹 계층에서 학습그룹의 생성, 제거, 학습 프로세스의 가입, 탈퇴 등의 오퍼레이션을 수행하면서, 학습그룹 관리를 담당하는 학습그룹 관리 시스템을 설계하고, 기본 함수를 구현하였다. 사례연구의 일환으로 가상클래스의 그룹 관리 시스템을 인터넷 상에서 원격 영어 학습을 지원할 목적으로 개발한 응용 서비스인 "GAIA" 시스템에 적용하여 실용성을 보였다.

앞으로의 연구 과제로 초고속망의 가상클래스 상에서 학습그룹을 구성하는 프로세스 수의 급진적 증가에 따른 복잡성 문제 및 사용자 인터페이스를 비롯하여, 보안관리, 성능관리 등의 일반적인 망의 관리 기능에도 연구가 계속되어야 한다. 그리고, 최근에 국내외에서 많은 연구가 진행되어 이와 관련 시스템이 출현되고 있는 지식 관리 시스템 (KMS : Knowledge Management System)과 연계된 연구가 필요하다[4]. 원격교육 시스템에 이러한 지식 관리 시스템을 적용하게 되면 방대한 학습 데이터의 관리, 지식 베이스 문서 관리, 그룹웨어, 검색 및 인식 시스템 등의 기술 적용을 통하여 효율적이고 체계적인 교육 시스템의 구축이 가능할 것이다.



<그림 6> 그룹전송의 예

6. 결 론

본 논문에서는 초고속망의 응용서비스인 원격교육

참고 문헌

[1] 김태영, 김영식(1995), "초고속정보통신망에 기반한 원격교육시스템 기술," 정보과학회지, 제13권 제6호, pp 5-22.
 [2] 박관우, 한순희, 조국현(1996), "정보통신 기술에 관한 국제 표준화 동향 분석 연구", 정보통신학술 연구과제(95-18), 정보통신부.

- [3] 박현제, 전길남(1990), "분산시스템에서의 그룹상호 작용", 한국과학기술원, 박사학위 논문
- [4] 안성철(1998), "지식 관리 시스템과 에이전트 시스템의 산업 기술 동향", 한국정보처리학회지, 제5권 제6호, pp. 91-95.
- [5] 왕창중(1996), "초고속 정보 통신망에서 원격교육을 위한 개방형 시스템", 한국정보과학회 학술발표논문집 Vol. 22, No. 2.
- [6] Aleta Ricciardi and P.Birman(1991), "Using Process Groups to Implement Failure Detection in Asynchronous Environments", Technical Report, Cornell University Computer Science Department.
- [7] Bala Rajagopalan(1991), "Reliability and Scaling Issues in Multicast Communication", SIGCOMM'92, ACM, Aug. 1992.[6]. C.A. Eills, S.J. Gibbs, G.L. Rein, "Groupware : Some Issues and experiences", Communications of the ACM, Vol. 34, No. 1.
- [8] C. Ellis, S. Gibbs and G.Rein(1991), "Groupware Some Issues and Experiences," Communication of ACM, Vol.34, No. 1.
- [9] C. Ellis, S. Gibbs(1991), "Concurrency Control in Groupware Systems," Communication of ACM, pp 399-407.
- [10] Hiroshi Ishii, M. Kobayashi, and K. Arita(1994), "Iterative Design of Seamless Collaboration Media", Commu. of the ACM, Vol. 37, No.8.
- [11] J. Crowcroft, K. Paliwoda(1991), "A Multicast Transport Protocol", Communications of The ACM, Vol. 34, No. 1.
- [12] Kenneth P.Birman, Robert Cooper, and Barry Gleeson(1992)," Programming with process groups : Group and Multicast Semantics", Technical report, Cornell University Computer Science Department.
- [13] L. H. Ngoh(1992), "Multicast Support for Group Communication", Computer Networks and ISDN Systems, Vol. 22, pp. 165-178.
- [14] Phanwoo Park, Katsuhiko Shirai(1998), "Distance Education Systems for English Learning on the Internet", Proceedings of FIE'98, IEEE, Arizona USA.
- [15] Richard M. Adler(1995), "Distributed Coordination Models for Clients/Server Computing", IEEE COMPUTER.
- [16] Walter R., Jean S., Gerd V., Michael W.(1994), " CSCW Tools: Concepts and Architectures." IEEE Computer, Vol 27, No. 5, pp. 28-36.
- [17] ISO / IEC(1990), Information Processing System, Open System Interconnection, Management Information Services : Structure of Management Information- Part 1:Management Information Model.