

CORBA 기반 원격 회의 시스템의 부하분산 및 결함허용 모듈 설계

성재철*, 최항목**, 박세명**

*인제대학교 전산학과

**인제대학교 전산학과

e-mail:keepupjc@orgio.net

The Design of Module to Give Load Balancing and Fault Tolerance for CORBA Based Remote Conference System

Jae-Chul Sung*, Hang-Mook Choi**, Se-Myung Park*

*Dept of Computer Science, In-Je University

**Dept of Computer Science, In-Je University

요약

인터넷 및 통신 기술이 발달함에 따라 원격 회의 시스템, 원격 교육, 가상 대학과 같은 응용에 대한 연구가 활발히 진행되고 있다. 이와 같은 시스템에 기존의 중앙 집중 방식을 사용할 경우에는 서버 측의 부하가 모두 집중되어 전체 시스템에 영향을 미치게 될 수 있으므로 분산 시스템을 이용한 연구가 활발히 진행되고 있다. 본 논문에서는 원격 회의 시스템에서 동적 부하 분산과 동적 결함 허용을 제공하기 위해 개별 회의를 관리하는 회의 관리자 객체의 생성, 복사, 이동, 삭제 기능을 가진 Factory 객체 모듈을 제안한다.

1. 서론

인터넷을 비롯한 네트워크 기술의 급속한 발전과 정보화 사회로의 변화에 따라 보다 빠른 데이터의 획득과 그 데이터의 정보화가 쉬워졌다. 데이터로부터 정보를 추출하는 것은 컴퓨터를 이용한 처리에 의한 방법이나 전문가들과의 의견 교환에 의한 방법 등 다양하게 수행될 수 있다. 기존의 의견 교환을 위한 회의는 회의 참여자들이 모두 일정한 장소에 모여 수행되었으나, 하드웨어와 통신 시스템의 발전에 따라 음성, 영상, 데이터 등을 종합적으로 이용하는 멀티미디어 통신 서비스가 가능해지고, 기존의 회의가 급변하는 사회에서 시간적, 경제적인 손실을 초래할 여지를 갖고 있어 컴퓨터를 이용하여 회의 참여자들이 위치적으로 분산되어 회의에 참여할 수 있도록 하는 화상 회의 시스템에 대한 연구가 수행되고 있다[1][2][3][4][5].

기존의 중앙 집중 방식의 컴퓨팅 시스템은 중앙 서버에 대부분의 부하가 집중되고, 이종의 시스템간의

연결에 있어서 새로운 이종 시스템이 추가될 때마다 그에 맞는 인터페이스를 제공해 주어야 하는 등 많은 문제를 갖고 있다. 이의 해결을 위해 분산 시스템으로 발전되어 왔으며, 분산 시스템은 주로 클라이언트/서버(C/S) 기술로 구성되며, C/S 시스템은 크게 2단계 C/S 시스템과 다단계 C/S 시스템으로 분류할 수 있으며 2단계 C/S 시스템은 서버 부분과 클라이언트가 직접 연결되는데 비해서 다단계 C/S 시스템은 서버와 클라이언트 부분을 완전히 분리하고 이들의 연결 기능을 제공하는 RPC(Remote Procedure Call)나 CORBA(Common Object Request Broker Architecture) 등과 같은 미들웨어를 위치시킴으로써 이질적인 요소를 갖는 시스템들이 네트워크를 통해 자유롭게 데이터를 주고받을 수 있는 좀 더 유연하고 확장 가능한 시스템을 구현할 수 있다[6].

본 논문에서는 클라이언트가 특별한 장비나 소프트웨어 없이도 웹브라우저만으로 회의에 참여할 수 있

는 환경을 조성하기 위해 웹을 기반으로 하고, 플랫폼 독립성과 위치 투명성을 제공하는 이기종 분산 객체 환경의 CORBA를 기반으로 한 원격 회의 시스템에서 서버 측 회의 관리자의 동적 부하 분산과 동적 결합 허용을 위해 객체의 동적인 생성/삭제 등을 관리해 주는 Factory 객체를 설계하고 Unified Modeling Language(UML)[7]을 기반으로 Rational Rose 2000을 이용해 모델링하였다.

2. 관련 연구

2.1 CORBA의 생명 주기 서비스[8][9]

객체들의 생성, 복사, 이동, 삭제에 대한 연산을 제공하는 서비스로 구현 가능한 명세(implementable specification)가 아니라 추천들(recommendations)의 집합으로 이 서비스를 이용하여 원격 주소 영역에 CORBA 객체를 생성할 수 있다.

CORBA의 생명 주기 서비스는 일반 객체에 대한 생명 주기 서비스와 서로 연관된 집합적 객체를 위한 생명 주기 서비스 인터페이스를 제공하며 이를 <그림 1>과 <그림 2>에 나타내었다.

2.2 CORBA의 부하 분산[10]

같은 서버가 여러 개 실행되어 있을 경우 클라이언트의 요청이 오면 라운드 로빈 방식을 사용하여 순서대로 돌아가면서 서버에 연결시켜 준다.

2.3 CORBA의 결합 허용[10]

CORBA의 OSAgent는 다음과 같은 2가지 결합 허용 기능을 제공한다.

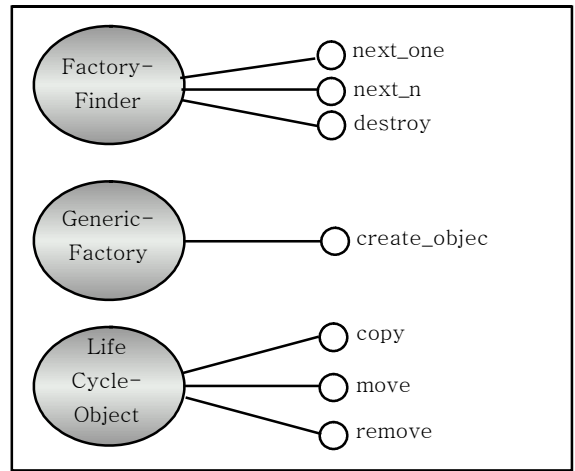
- 1) 같은 구현 객체가 포함된 여러 개의 서버가 실행되고 있는 경우, 서버가 실행이 종료되거나 네트워크 연결이 안 될 경우 같은 기능을 하는 다른 서버로 연결해 주기 때문에 클라이언트는 내부적인 메커니즘을 느끼지 못한 채 서비스를 계속 이용 가능하다.
- 2) OSAgent의 프로세스가 종료되거나 네트워크 연결이 안되었을 경우, 모든 서버들은 자신의 위치를 관리해줄 OSAgent를 찾기 위해 브로드캐스트 메시지를 보내 응답하는 OSAgent가 있을 경우 자신의 위치 정보를 등록한다.

3. 시스템의 개요 및 구성

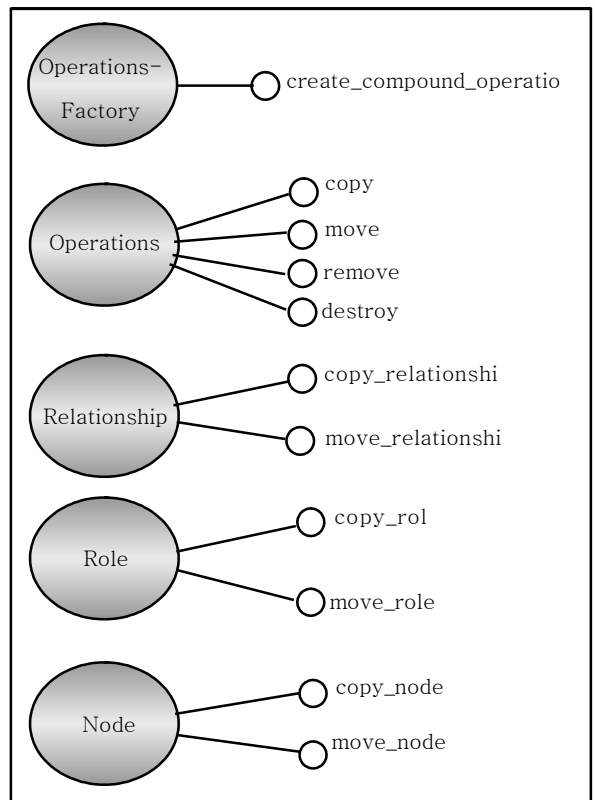
시스템은 이미지, 텍스트 문서 또는 프리젠테이션

파일과 같은 회의 자료를 필요로 하는 회의가 가능한 시스템으로써 독립된 컴퓨터에 회의 주제가 다른 다수의 회의가 개설된다. 또한 회의 참여 인원 증가에 따른 전체 시스템의 부하 증가와 개설된 회의의 결합 발생 시 회의 객체와 참여자의 이동에 의해 연속적으로 회의를 진행할 수 있는 환경을 제공하기 위해 Factory 객체를 사용한다.

회의 참여자는 서버로부터 다운로드받아 클라이언트 측에서 실행될 수 있는 자바 애플릿을 사용함으로써 서버에 집중될 수 있는 부하의 일부를 참여자의 몫



<그림 1> 일반 객체를 위한 인터페이스



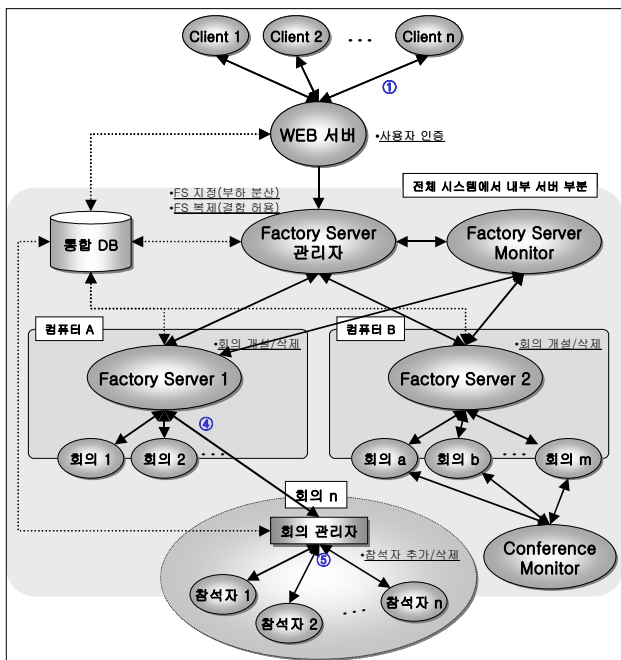
<그림 2> 집합적 객체를 위한 인터페이스

으로 돌리고, 웹브라우저만 있으면 어느 장소에서도 회의에 참여할 수 있고, 클라이언트용 프로그램을 쉽게 배포할 수 있는 환경을 제공한다. 클라이언트와 회의 서버인 CORBA 구현 객체 사이에 웹 페이지를 위치시킴으로서 클라이언트의 회의 참여 요청에 대해 애플릿을 전송하고, CORBA 구현 객체에게 클라이언트의 레퍼런스를 전달한다. 클라이언트와 웹 페이지 사이에서는 TCP/IP 프로토콜이 사용되고, 참여자가 회의에 참여된 후에는 CORBA의 IIOP 프로토콜을 사용하여 상호 통신을 수행한다.

또한 서버 측에는 Factory 객체를 사용하여 새로운 회의 객체의 생성, 이동, 삭제 등을 통해 부하 분산과 결함 허용을 제공하는데, 이 Factory 객체는 각 회의 객체들을 관리하는 Factory Server(FS)와 각 FS들을 관리하는 Factory Server Manager(FSM)로 구성된다. FS의 하위에는 회의 주제에 따른 회의 객체인 회의 관리자(Conference Manager : CM)를 위치시켜 각 회의에 참여하는 클라이언트의 레퍼런스의 동적 관리와 메시지 및 회의 자료 전송 기능을 담당하게 된다.

시스템의 FSM와 FS에서의 부하 분산과 결함 허용을 위해 모든 회의와 참여자들의 정보를 관계형 통합 데이터 베이스에 저장해 두고 모든 객체에서 이를 접근할 수 있다.

<그림 3>은 시스템의 전체 구조를 보여 준다.



<그림 3> 시스템의 전체 구조

4. Factory 객체의 동작

회의 개설자가 회의 개설 요청을 하면 FSM는 통합 DB에 저장되어 있는 FS별 총 참여 인원을 참조해 가장 부하가 적은 FS 중에서 개설될 회의를 수용할 수 있는 FS를 찾거나 새로운 FS를 생성하고 FS의 레퍼런스를 DB에 등록하고, FS에게 회의 개설 정보를 전달한다. FS는 회의 관리자를 생성한다. 이로써 회의 개설 시에 동적인 부하 분산이 이루어질 수 있다.

또한 동적인 결함 허용 기능을 제공하기 위하여 Conference Monitor는 진행 중인 CM의 상태를 관찰해 결함이 발생한 CM의 레퍼런스를 FS에게 전달한다. FS는 새로운 회의를 생성하고 회의 참여자를 이동시킴으로써 연속적으로 회의를 진행시킨다.

또한 FS의 결함으로 포함된 회의들의 관리가 불가능해지는 경우에 대비하여 Factory Server Monitor를 두고 FS들의 상태를 관찰해 결함이 발생한 FS에 포함된 회의들을 새로운 FS를 생성하고 이동시킨다.

5. 클래스 다이어그램

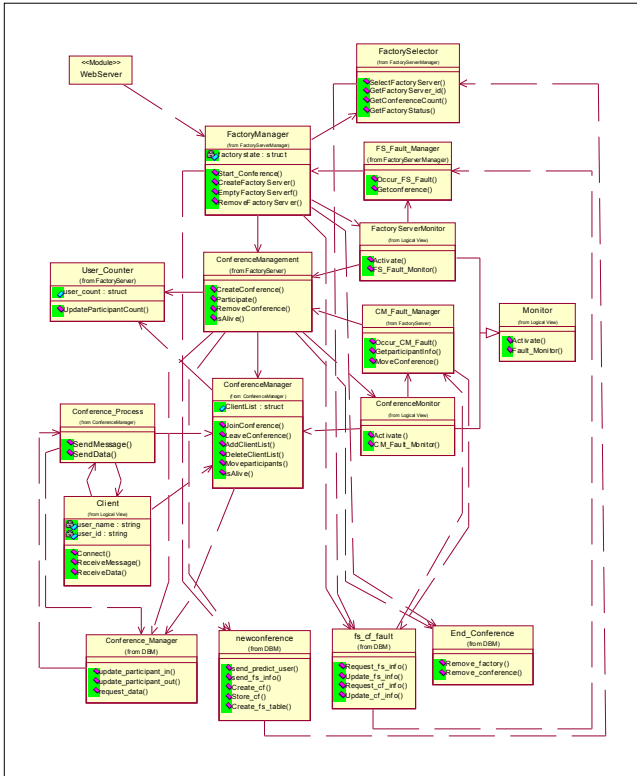
UML의 클래스 다이어그램은 클래스, 인터페이스, 협력들과 그들간의 관계를 보여 주는 도해로 시스템의 정적인 설계부를 표현하며, 시스템의 기능 요구 사항을 주로 지원한다[7].

<그림 4>는 시스템을 구성하는 각 객체들이 수행해야 할 오퍼레이션들을 구분하여 클래스로 구분한 클래스 다이어그램을 나타낸다.

본 논문의 원격 회의 시스템은 크게 회의의 개설 예약, 취소, 회의 정보 수정, 회원 관리 등을 주로 담당하는 Web Page와 회의 참여자의 관리, 자료 및 메시지 전송을 담당하는 ConferenceProcess, 회의 관리자의 동적 생성/삭제 등 회의 관리자를 관리하는 ConferenceManager, FS의 동적 생성/삭제 및 회의의 동적 배치 등 FS를 관리하는 FactoryManager 그리고, FS와 CM 객체의 결함을 감시하여 동적인 결함 허용 기능을 제공하는 Monitor와 이를 상속받는 FactoryServerMonitor, ConferenceMonitor 등과 같은 주요 클래스들로 구성된다.

6. 결론 및 향후 연구 방향

본 시스템은 회의 개설 시 예상 참여 인원을 모두 수용하기에 무리가 없는 최적의 팩토리 서버를 찾아 회의를 개설할 수 있으므로 회의 개설 시 부하를 전



<그림 4> 클래스 다이어그램

체 시스템 내에 고르게 분산할 수 있을 뿐 아니라 팩토리 서버와 회의 객체가 생성됨과 동시에 모니터 객체에 의해 항상 점검을 받기 때문에 객체에 결함이 발생해 응답하지 못하게 되면 즉시 회의 객체를 타 컴퓨터나 동일 컴퓨터의 다른 팩토리 서버로 이동시킬 수 있어 연속적인 회의의 진행을 보장할 수 있을 것으로 기대된다. 회의 객체와 그에 따른 회의 참여자들의 레퍼런스가 이동되는 동안의 약간의 시간적인 공백이 있을 것으로 예상되므로 이에 대한 원활한 처리 방안의 고안이 요구된다. 본 시스템은 원격지의 계열 업체들이나 동일 프로젝트를 수행하는 원격지의 단체들이 시간과 장소에 구애받지 않고 회의의 자료를 이용해 원활한 의견 교환을 할 수 있으며, 더 나아가 원격 교육과 같이 부교재를 필요로 하는 시스템에 응용이 가능할 것이다.

참고문헌

[1] 김두현, 이경희, 정찬근, 임영환, “스트림 연결 모델에 기반한 영상 회의 시스템의 설계 및 연결 특성 분석”, 정보과학회 논문지(C), 2, 182-196, 1996

[2] 김정현, 강재효, 성병우, 박세명, 최항목, 김상균, 최홍국, “CORBA 기반의 의료 영상 회의 시스템

을 위한 Server 설계”, 한국 정보과학회 가을 학술 발표 논문집, 25, 391-393, 1998

[3] 최규상, 김동희, 조성빈, 진성일, “Video for Windows을 이용한 인터넷 화상 회의 시스템 개발”, '97년 제10회 산학연 멀티미디어산업기술 학술대회, 202-205, 1997

[4] 정진호, 임혜영, 양현승, “DeskShare: LAN상에서 그룹 작업을 위한 멀티미디어 탁상 회의 시스템”, 정보과학회 논문지(C), 1, 1995, pp12-21

[5] 강재효, 김정현, 김상균, 박세명, 최항목, 최홍국, “CORBA 기반의 의료 영상 회의 시스템 구현”, 한국정보과학회 봄 학술 발표 논문집, 26, 1999, pp340-342

[6] 박재현, 코아 코바, 영한 출판사, 1998

[7] Booch, Rumbaugh, Jacobson, “The Unified Modeling Language User Guide”, Addison-Wesley, 1999

[8] 김형주, “일주일만에 배우는 CORBA”, 마이트 Press, 1999

[9] Michi Henning, Steve Vinoski, “Advanced CORBA Programming with C++”, Addison-Wesley, 1999

[10] “코바 프로그램 이것만은 알고 하자”, 프로그램 세계 1999년 10월호