

CORBA 기반 회의 시스템의 회의 관리자에 대한 부하분산 및 결함허용을 제공하는 모듈 구현

성재철*, 최항목**, 박세명**

*인제대학교 전산학과

**인제대학교 전산학과

e-mail:keepupic@orgio.net

The Implementation of Module to Provide Load Balancing and Fault Tolerance for Conference Manager of CORBA Based 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 연구와 애플리케이션의 개발이 활발히 진행되고 있다. 특히 원격 회의 시스템과 같은 실시간 시스템에 있어서는 서버 측의 부하 집중과 서비스 객체의 결함 발생에 의한 시스템에 발생할 수 있는 문제에 대해 분산 시스템을 이용함으로써 해결하고자 하는 연구가 수행되고 있다. 본 논문에서는 CORBA 기반의 회의 시스템을 대상으로 하여 개별 회의를 관리하는 회의 관리자 객체의 생성, 복사, 이동, 삭제 기능을 통해 동적 부하 분산과 동적 결함 허용 기능을 제공하는 Factory 객체와 객체 관리 모듈의 구현에 대하여 논한다.

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 원격 회의 시스템에 대한 연구가 수행되어 왔다[1][2][3][4].

원격지에서 회의에 네트워크를 통해 회의에 참여해 의견을 교환할 수 있는 시스템에서 고려해야 할 사항으로는 서버 측의 부하를 시스템 전체로 분산하는 방안과 네트워크 붕괴나 회의 관리자 객체의 결함에 의한 회의의 중단에 대한 복구 대책이라 할 수 있다. 이에 본 논문에서는 웹에서 클라이언트용 자바 애플릿을 다운로드 받아 장소, 소프트웨어 및 장비에 구애받지 않고, 웹 상에서 원격 회의를 수행할 수 있는 시스템에서 회의 관리자에 대하여 부하 분산 기능과 결함 허용을 제공하는 모듈을 구현하였다.

본 논문의 구성은 다음과 같다.

2장에서는 구현에 있어서 고려된 개발 환경에 대해서 살펴보고, 3장에서는 시스템의 각 구성 요소와 기능 및 구현에 대해 살펴본 후 4장에서 결론을 맺는다.

2. 개발 환경

원격 회의 시스템이나 메신저 프로그램들과 같은 실시간 의사 소통을 위한 프로그램에서는 장소나 컴퓨터의 사용 환경에 구애받지 않는 환경을 조성하는 것이 사용자의 편의의 측면에서 중요하다 할 수 있다. 본 연구에서는 클라이언트 측의 사용 상의 편의와 서버 측 회의 객체의 고장에 대한 결함 허용 기능의 구현을 위해 다음과 같은 환경을 사용하였다. 클라이언트 측은 플랫폼 독립적이고, 구현 코드의 배포가 용이한 Java Applet을 사용하고, 서버 측은 Java에 비해 속도 면에서 뛰어난 C++ 언어를 사용하여 구현하였다. 클라이언트와 서버 측을 모두 C++ 언어를 사용한다면 보다 빠른 속도를 기대할 수 있지만, Java 언어의 이식성과 유연성을 고려해 클라이언트 측의 구현에 Java Applet을 사용함으로써 특별한 장비나 별도의 소프트웨어 없이 웹브라우저의 사용으로 클라이언트용 코드 배포가 용이하고 서버 측의 부하를 클라이언트 측으로 분산시킴으로써 서버 측의 성능을 향상시킬 수 있다[5].

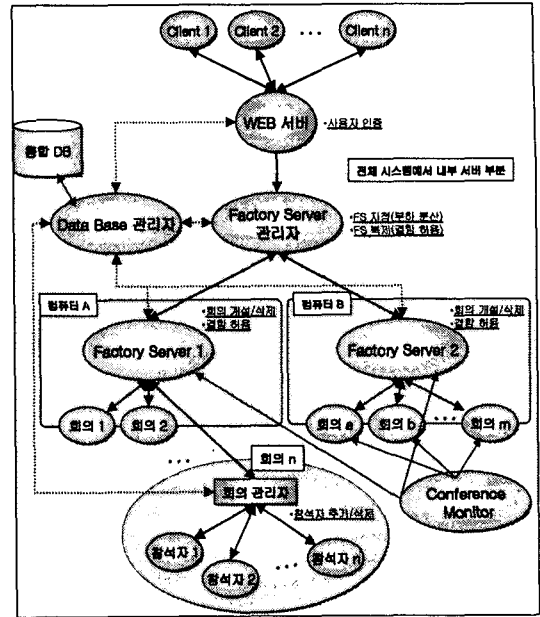
또한, 서로 다른 언어로 구현된 클라이언트와 서버 객체의 연결 및 서버 측의 분산된 객체들을 연결하기 위한 미들웨어로써 'Inprise'사의 'Visibroker 3.3'을 사용하였다[6][7].

CORBA는 각기 다른 언어로 만들어진 객체간의 연결 통로로써 매우 뛰어난 미들웨어이며, 플랫폼에 독립적이라서 호환성이 뛰어나고 개발 및 유지 보수가 쉬우며 좀 더 유연하고 확장 가능한 시스템을 구현할 수 있다[8][9][10].

3. 시스템의 구성

본 시스템은 주제가 다른 다수의 회의가 독립된 컴퓨터에 개설되어 진행되고 있을 때 발생할 수 있는 회의 참여 인원과 개설 회의의 증가에 따른 부하를 다른 컴퓨터의 Factory 객체에 회의 객체를 생성하도록 함으로써 부하를 시스템 전체로 분산하는 부하 분산 기능과 진행 중인 회의 객체의 결함 발생 시 회의 객체와 참여자를 동적으로 이동시킴으로써 회의 객체의 결함 허용 기능을 제공한다.

시스템의 전체 구성은 <그림1>과 같다.



< 그림 1 > 시스템의 전체 구조

3.1 회의 객체

한 대의 컴퓨터에 서로 다른 주제의 복수개의 회의가 개설되어 진행되며, 새로운 회의의 개설 요청에 따라 Factory Server 객체에 의해 생성되며, 회의 참여자 관리와 참여자 간 메시지 전송, 회의 자료 전송 기능을 갖는다. 콜백 메커니즘을 사용함으로써 클라이언트의 객체 참조를 관리하고, 클라이언트를 호출한다.

3.2 Factory Server 객체

진행 중인 회의의 관리를 위해 독립적인 컴퓨터에 하나의 Factory Server 객체가 존재한다. 참여 인원 증가에 따른 부하 증가와 개설된 회의의 결함 발생에 따라 회의 객체와 그 참여자의 이동에 의해 연속적으로 회의를 진행할 수 있도록 하는 회의 객체의 동적 부하 분산과 동적 결함 허용을 제공한다. 이를 위해 Factory Server 객체에는 자신이 관리하는 회의 객체들의 목록과 전체 참여 인원내 대한 정보를 유지하고 있다.

또한, 회의 객체들의 결함 발생에 의해 기존 회의의 진행이 불가능한 경우에 대비하여 회의 객체가 생성된 후 회의 이름과 회의 객체의 레퍼런스 및 회의별 참여자 이름과 참여자 레퍼런스의 정보를 유지하는 회의 정보 목록 구조체를 Factory Server 객체

내에 두었다.

이는 같은 서버가 여러 개 실행되어 있을 경우 클라이언트의 요청이 오면 라운드 로빈 방식을 사용하여 순서대로 돌아가면서 서버에 연결시켜 주는 형태인 CORBA의 부하 분산 기능[11]과 서버의 실행 종료나 네트워크 연결이 안 될 경우, 같은 구현 객체가 포함된 여러 개의 서버가 실행되고 있을 때 다른 서버로 연결해 주거나 서버 객체 자신을 관리해 줄 CORBA의 OSAgent를 찾아 서버 객체 자신을 등록하는 CORBA의 결합 허용 기능[11]을 강화하고, 보다 동적인 부하분산과 결합 허용 기능을 제공한다.

회의 객체는 서버의 물리적 주소, 서버와 통신에 사용되는 버전을 포함한 프로토콜 그리고 목적 객체의 객체 ID와 같은 정보를 제공함으로써 다른 서버 내에 객체 참조를 생성할 수 있으나, 이는 CORBA 객체 모델의 위치 투명성이라 특성과는 동떨어진 모델이 될 수 있으므로[12] Factory Server 객체가 있는 위치에 생성되도록 하였다.

3.3 Factory Server 관리자

전체 시스템에 하나만 존재하며, 분산되어 실행되고 있는 Factory Server 객체들의 목록을 관리하며, 생성될 회의를 각 Factory Server 객체의 부하를 고려하여 분배하고, 결합이 발생한 회의 객체와 참여자를 수용할 수 있는 Factory Server 객체를 선택하여 할당하는 역할을 수행할 수 있도록 Factory Server 관리자는 각 Factory Server 객체들의 참조 정보와 각 Factory Server 객체 당 총 참여 인원 및 전체 참여 예상 인원에 대한 정보를 유지하고 있다.

3.4 Conference Monitor 객체

각 Factory Server 객체에 포함되어 있는 개별 회의의 결합 발생 여부를 감시하기 위한 객체로 주기적으로 각 회의 객체를 호출하여 응답이 없는 경우 결합이 발생한 것으로 판정한다. 회의 객체에 결합이 발생하면, 그 회의를 수용할 수 있는 Factory Server 객체를 선택해 새로운 회의를 생성하고 참여자를 이동하여 등록할 수 있도록 결합이 발생한 회의를 포함하고 있던 Factory Server 객체에 결합 발생을 알린다.

3.5 객체 정보 관리

Factory Server 객체가 실행이 되면 Factory Server 관리자에 자신의 참조 정보를 저장하고, 회의 개설자가 웹 상에서 회의 개설 예약 시 미리 입력해 둔 회의 참여 인원에 대하여 Factory Server의 부하를 고려하여 회의 객체의 할당을 위해 Factory Server 관리자에 의해 관리된다.

Factory Server 객체들의 결합 발생에 대한 결합 허용 기능에 대해서는 고려하지 않았다.

Factory Server와 Factory Server 관리자가 실행되어 준비되어 있는 상태에서 클라이언트로부터의 회의 개설 요청이 Factory Server 관리자에 전송되면 Factory Server 관리자는 회의를 수용할 수 있는 Factory Server를 선택해 회의 객체 생성을 요청하여 회의가 생성되면 Factory Server 객체의 회의 정보 리스트에 생성된 회의 객체의 참조와 회의 id를 저장하고, 참여자의 정보를 전달하여 회의 객체의 결합 발생 시 새로운 회의 객체를 생성하여 진행 중인 회의의 참여자들을 이동시키기 위해 Factory Server 객체의 회의 정보 리스트에 저장하고, 또한 회의 객체와 클라이언트간의 호출을 위한 콜백 리스트로써 회의 객체 내에 참여자의 정보를 유지하도록 하였다.

4. 결론 및 향후 연구 방향

본 시스템은 클라이언트 측에 애플릿을 사용함으로써 사용자에게 프로그램의 설치, 유지 보수에 대한 부담을 주지 않고, 위치 투명성을 제공하여 어디에서나 웹 브라우저를 통하여 회의에 참석할 수 있는 편의성을 제공한다.

또한, 회의 개설 시 예상 참여 인원을 모두 수용하기에 무리가 없는 최적의 Factory Server를 찾아 회의를 개설할 수 있으므로 회의 개설 시 부하를 전체 시스템 내에 동적으로 고르게 분산할 수 있을 뿐 아니라 회의 객체가 생성되면 Conference Monitor 객체에 의해 주기적으로 점검을 받으므로 객체에 결합이 발생해 응답하지 못하게 되면 즉시 새로운 회의 객체를 생성해 참여자의 참조들을 이동시킬 수 있어 연속적인 회의의 진행을 보장할 수 있다.

좀 더 개선되어야 할 사항으로는 강화된 사용자 인터페이스를 제공과 현재 구현에서 보류되어 있는 데이터 베이스와의 연동을 통해 각 객체마다 중복 유지되고 있는 Factory Server 객체, 회의 객체 및

클라이언트에 대한 정보들의 중복을 최소화할 수 있도록 개선하여 시스템 구성 객체들 사이의 정보의 참조의 효율성을 증대시키고, 좀 더 다양하고 고품질의 회의 자료를 사용할 수 있는 환경을 제공하여야 할 것이다.

특히 본 시스템의 정확한 성능 분석을 통하여 최고의 성능을 위한 객체 배치에 대한 연구와 부하를 판단할 수 있는 보다 객관적인 기준에 대한 연구가 지속적으로 수행되어야 하겠다.

참고문헌

- [1] 김정현, 강재효, 성병우, 박세명, 최항목, 김상균, 최홍국, "CORBA 기반의 의료 영상 회의 시스템을 위한 Server 설계", 한국 정보과학회 가을 학술 발표 논문집, 25, 391-393, 1998
- [2] 최규상, 김동희, 조성빈, 진성일, "Video for Windows을 이용한 인터넷 화상 회의 시스템 개발", '97년 제10회 산학연 멀티미디어산업기술 학술대회, 202-205, 1997
- [3] 정진호, 임혜영, 양현승, "DeskShare: LAN상에서 그룹 작업을 위한 멀티미디어 탁상 회의 시스템", 정보과학회 논문지(C), 1, 1995, pp12-21
- [4] 강재효, 김정현, 김상균, 박세명, 최항목, 최홍국, "CORBA 기반의 의료 영상 회의 시스템 구현", 한국정보과학회 봄 학술 발표 논문집, 26, 1999, pp340-342
- [5] 김복순, 강혜민, 이은희, 윤영주, 박우창, "CORBA 환경에서 데이터베이스 클라이언트 성능 평가", 한국 정보처리학회 추계 학술 발표 논문집, 제6권, 제2호, 1999
- [6] Visigenic, "Visibroker for C++ Programmers's Guide 3.3" <http://www.visigenic.com> 1999
- [7] Visigenic, "Visibroker for C++ Reference Guide 3.3" <http://www.visigenic.com>, 1999
- [8] Roert Orfali, Dan Harkey, "Client/Server Programming With JAVA and CORBA", Wiley, 1997
- [9] John Siegel, "CORBA Fundamentals and Programming", Wiley, 1996.
- [10] 박재현, 코아 코바, 영한 출판사, 1998
- [11] 문왕식, "코바 프로그래밍", 대림, 2000
- [12] Michi Henning, Steve Vinoski, "Advanced CORBA Programming with C++", Addison-Wesley, 1999