

# 분산 미들웨어 기반의 전자투표시스템

최진우\*, 김은숙\*, 김영만\*, 우종우\*, 한제일\*, 전성익\*\*  
 \*국민대학교 컴퓨터학부, \*\*한국전자통신연구원 실시간 OS 연구실

## Electronic Voting System based on Distributed Middleware

J. W. Chor\*, E. S. Kim\*, Y. M. Kim\*, J. W. Woo\*, J. I. Han\*, S. I. Jun\*\*  
 \*Kookmin Univ. Computer science, \*\*ETRI

### 요약

인터넷 사용의 폭발적 증가로 인해 다양하고 인터넷에서 제공되는 새로운 서비스들이 계속 증가하고 있다. 본 연구는 새로운 인터넷 서비스인 전자투표시스템에 대한 것으로 분산 미들웨어를 사용하여 H/W 나 S/W 의 제한 없는 클라이언트/서버 시스템을 목표로 하였다. 본 논문은 대표적인 분산 미들웨어인 CORBA 와 DCOM 을 사용하여 전자투표시스템의 프로토타입을 구현하고 그 결과를 비교한다. 이는 앞으로 개발될 실제 전자투표시스템의 개발에 요구되는 분산 미들웨어 선택에 유용한 지침이 될 것이다

### 1. 서론

네트워크와 컴퓨팅 환경의 발전은 PC 통신과 인터넷 사용자를 기하급수적으로 증가시켰으며, 사회제도 역시 이러한 컴퓨팅 환경 및 제공 서비스에 따라 변화하고 있다. 대표적인 사회제도의 하나인 선거제도에서도 인터넷과 PC 통신을 이용하는 것이 시도되고 있으며, 실제로 브라질에서는 전자투표가 실시된 바도 있다[1]. 현재 국내에서는 선거에서 여론 수렴과 토론을 위해 통신이 사용되고 있으나, 앞으로는 투표도 네트워크를 통하여 이루어질 수 있게 될 것이다. 인터넷상에서 수행되는 전자투표시스템은 실제 선거에 사용될 수도 있으며, 여론조사 따위에 응용이 가능하다.

전자투표시스템 구축은 네트워크를 통한 분산 환경에서 이루어지며, 이를 보다 쉽게 구현하기 위해 분산 객체 미들웨어의 사용이 필요하다. 또한 웹을 이용하여 선거 결과를 투표와 동시에 인터넷으로 생중계 하고, 네트워크를 통해 투표 프로그램을 배포하여, 통신이 가능한 어느 곳이나 투표가 가능하게 한다. 이 시스템은 어떠한 플랫폼에서도 수행이 가능하도록 지바를 사용하여 작성한다. 이를 위해 대표적인 분산 객체 미들웨어인 CORBA 와 DCOM 을 사용하여 간단한 투표 프로그램을 작성한 후 두 분산 객체 미들웨어의 성능과 구조를 비교 분석하여, 실제 투표시스템을 위한 분산 객체 미들웨어의 장 단점을 비교해 본다.

본 논문은 2장에서 분산 객체 미들웨어인 CORBA 와 DCOM 의 기본적인 구조를 살펴보고, 3장은 CORBA 와 DCOM 을 기반으로 한 간단한 투표시스템을 설계 제작하여 그 성능과 효율을 비교한다. 4장은 본 논문의 결론과 향후 연구과제에 대하여 기술한다.

### 2. 분산 객체 미들웨어의 구조

객체가 컴퓨터와 네트워크를 타고 생성, 소멸되는 분산 객체 미들웨어에서는 복잡한 클라이언트/서버 시스템을 계획할 수 있는 소프트웨어 컴포넌트를 이용해서

짧은 시간에 구축 할 수 있다. 또한 객체를 다른 컴포넌트에 영향을 주지않고 수정할 수 있으며, 네트워크를 통한 '플러그 앤 플레이'가 가능하게 된다. 다음은 대표적인 분산 객체 미들웨어인 CORBA 와 DCOM 에 대하여 기술하고 있다.

#### 2.1 CORBA

CORBA 는 800 여 개의 컨소시엄 구성업체(SUN, IBM, Netscape, Iona 등)가 제안한 분산 객체 미들웨어로 다양한 하드웨어와 운영체제 플랫폼 상에서 지원 되고있다. 또한 WWW, CORBA, Java 를 통합한 오브젝트 웹을 제공하여 웹 기반의 컴퓨팅환경을 효율적으로 운영할 수 있게 하고있다.



[그림 1] ORB 의 구조

CORBA 는 서비스 객체의 인터페이스와 구현부는 별개로 작성되고 운영된다. CORBA 는 인터페이스를 기술하기 위한 언어인 IDL (Interface Definition Language)을 별도로 제공하여 IDL 로 인터페이스를 작성하여 IDL 컴파일러를 통해 스텝(stub)과 스킴러톤(skeleton)을 만든다. 클라이언트는 스텝을 사용함으로써 자신의 주소영역에 서비스 객체가 있는 것 같이 느끼게 된다. CORBA 에서는 스텝 내부에 의사 객체(proxy object)를 만들어 이 객체를 통해 실제 서비스 객체의 메소드를 호출하고 호출이 끝난 뒤 리턴하기 위한 통신 역할을 담당하도록 한다.

객체 어댑터(Object Adapter)는 클라이언트에서 보내진 바이너리 스트림(binary stream)을 중간에서 올바른 서비스 객체에게 보내주는 역할을 하는 것으로 BOA(Basic

본 논문은 한국전자통신연구원의 '98 학연공동연구과제('1998-47)로 수행되었음

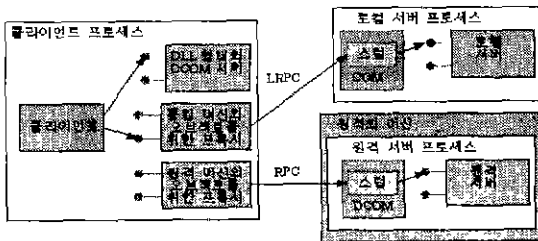
Object Adapter)가 있다 그러나 BOA는 CORBA를 구현한 업체마다 나름대로의 기능을 확장하고 있기 때문에 서로 다른 CORBA 제품과의 상호 연동성이 떨어지게 된다. 이러한 점을 해결하기 위해 CORBA 2.2에서는 POA(Portable Object Adapter)를 제안하고 있다.

실제 동작을 살펴보면, [그림 1]에서 하나의 컴포넌트(클라이언트)가 다른 컴포넌트를 호출하게 되면, 클라이언트는 IDL 스텝이나 동적 호출 인터페이스를 통해서 메소드 호출을 ORB로 보낸다. 여기서 스텝은 객체 서비스에 정적 인터페이스를 제공한다. 그러면 ORB는 호출하려는 컴포넌트의 위치를 파악해 자료를 전달하게 된다.

ORB는 분산된 객체들을 클라이언트/서버 관계로 만들어주는 핵심적인 컴포넌트이며 실시간으로 객체들을 찾아내고 서비스를 호출할 수 있도록 하는 역할을 한다. 이러한 ORB는 정적호출(static method invocation)과 동적호출(dynamic method invocation)을 모두 지원하므로 메소드 호출을 컴파일 시에도 할 수 있고, 실행 중에도 할 수 있다.

### 2.2 DCOM

COM/DCOM은 Microsoft사의 모든 제품의 근간이 되는 기술로, 윈도우라는 운영체제의 대중성으로 인해 가능성이 높게 평가되고 있다. DCOM은 COM에 분산 기능을 추가한 것으로 COM의 기본개념은 CORBA와 많은 면에서 유사하다. 각 서비스는 COM 서버에 의해 제공되며 COM 서버는 자신의 서비스를 하나 이상의 인터페이스를 사용하여 외부에 공개한다. CORBA의 IDL과 유사한 MIDL(Microsoft Interface Definition Language)를 제공, 인터페이스를 작성하고, MIDL 컴파일러를 사용 프록시와 스텝을 만든다.

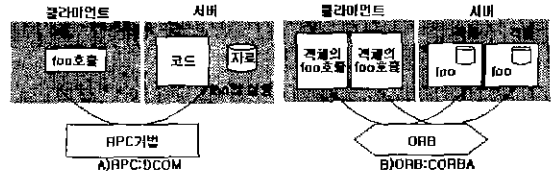


[그림 2] DCOM의 구성

[그림 2]에서 클라이언트가 동일주소영역에서 동작하는 DLL 형태의 COM 서버가 제공하는 인터페이스를 사용하는 것과 다른 주소영역의 COM 서버와 원격 머신의 COM 서버에 접근하기 위해 프록시와 스텝을 사용하는 것을 보여주고 있다.

DCOM에서는 원격 객체를 지원하기 위해서 DCE-RPC 위에 ORPC(Object Remote Procedure Call)계층을 만든다. ORPC는 표준 RPC 패킷에 DCOM의 인터페이스 포인터를 구분하는 ID(Interface Identifier, IID), 버전 정보 등을 담아 전달해 함수를 호출한다.

### 2.3 CORBA와 DCOM의 비교

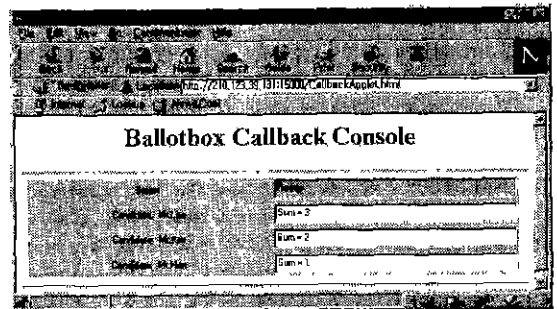


[그림 3] ORB 대 RPC

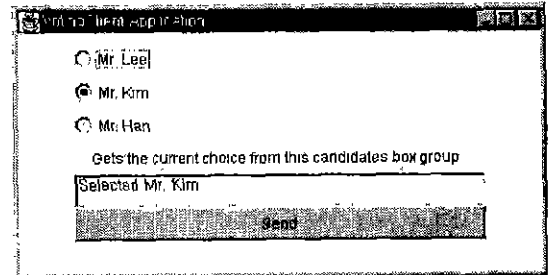
DCOM은 CORBA와 많은 면에서 유사하나 몇 가지 제약이 있다. 예를 들어, CORBA는 다양한 플랫폼에서 지원되지만 DCOM은 마이크로소프트(Microsoft, 이하 MS)로 특화되어 있고, DCOM은 자바 플랫폼 중 MS와 Visual J++만을 지원한다. 네이밍 서비스에 있어서도 비연속적이다.

그리고 OMG(Object Management Group)에선 CORBA 객체와 DCOM의 객체 사이에 통신이 가능하도록 CORBA를 위한 소프트웨어 브릿지를 개발 중에 있다.

### 3. 전자투표시스템



[그림 4] 홈페이지



[그림 5] 클라이언트 투표 프로그램

전자투표시스템은 다음의 투표방식의 기본 사항을 만족하여야 한다[4].

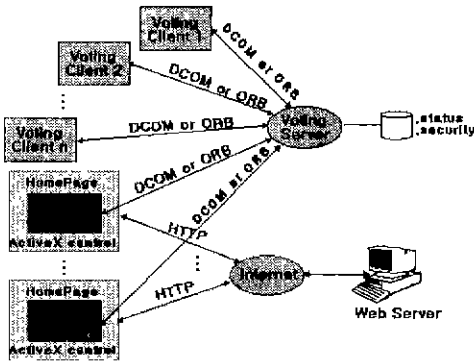
- 1 정확성: 투표내용을 수정 제거하는 시도 및 잘못된 투표 입력을 방지
- 2 민주성: 모든 유권자는 1표에 한해 투표할 수 있음
- 3 프라이버시: 유권자의 투표내용을 다른 사람들이 파악할 수 없음
- 4 증명성: 누구라도 투표집계의 정확성을 증명할 수 있음
- 5 유연성: 여러 형태의 투표 형식 규모 내용에 쉽게 전환
- 6 편리성: 간단하고 신속하며 최소한의 장치로 이루어짐
- 7 이동성: 투표자, 투표장소, 관리자 등의 위치에 제한 없음

는 구조

- 투명성. 투표시스템의 H/W, S/W, 운영 상황이 공개되어 누구나 감시가능

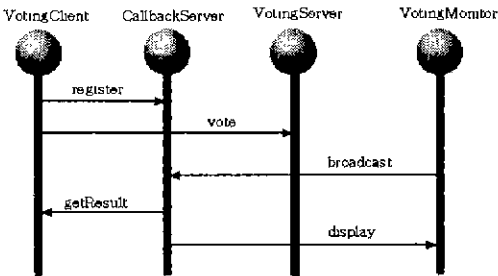
본 논문에서 이러한 특징을 갖는 투표시스템의 본격적인 구현에 앞서서 우선 기본적인 기능을 갖는 프로토타입을 작성하였다. 이 프로토타입은 이동성과 유연성 그리고 투명성(이를 위해 CORBA 로 작성된 시스템에서는 POA, 네이밍 서비스를 사용하였다.)을 제공한다. 그리고 테스트를 위해 3인에 대한 인기투표를 분산 방식으로 수행하며, 투표 프로그램은 따로 유권자에게 배포하는 것을 가정한다[그림 5]. 그리고 투표가 이루어질 때마다, 웹의 홈페이지는 투표결과가 실시간으로 업데이트가 되는 프로그램을 작성하였다[그림 4].

CORBA 를 사용하여 작성한 것은 Visibroker for Java 를 사용하였다. DCOM 을 사용하여 작성한 것은 Visual C++ 과 액티브 X 기술을 사용하였다. 그 구조는 [그림 6]와 같다.



[그림 6] 전체 프로그램 구성도

### 3.1 CORBA 로 구현한 전자투표시스템



[그림 7] 전자투표시스템의 시나리오

CORBA 를 이용한 전자투표시스템의 시나리오 [그림 7]는 투표에 참여하는 모든 어플리케이션에게 투표가능하다는 메시지를 브로드캐스트하게 되고 해당 어플리케이션이 투표 후엔 자신이 선택한 후보자의 표가 증가되었다는 메시지를 받게 된다. 이때 모든 투표 상황을 지켜 볼 수 있는 애플릿을 두어 웹브라우저 상에서 그때 그때의 변화하는 득표 상황을 볼 수가 있다.

본 시스템의 특성은 다음과 같이 크게 두 가지로 말할 수 있다. 첫 번째로는 네이밍서비스를 사용함으로써 분산 객체들을 액세스하기 쉽도록 객체들을 한 곳으로 등록시켜 이용하기 편하게 하였으며 두 번째로는 이벤트 서비스의 확장인 콜백을 사용함으로써 클라이언트의 인터페이스를 분산 객체로 구성한 뒤 클라이언트의 객체 참조자를 서버에 등록시켜 놓음으로써 클라이언트가 서버 측의 객체를 호출하지 않아도 서버에 등록되어 있는 클라이언트의 객체 참조자를 통해 비동기적으로 호출을 수행하는 방법을 사용하였다

이러한 기술을 사용하여 객체가 어떠한 분산환경에 존재하더라도 이름만으로도 그 객체를 액세스 할 수 있도록 구현하였으며 또한 서버로 접속된 모든 어플리케이션이 레코드들을 변경하였을 때 즉시 웹상의 애플릿으로 통보되어 질 수 있게 하였다.

### 3.2 DCOM 으로 구현한 전자투표시스템

DCOM 으로 구현한 전자투표시스템도 그 구성에 있어 CORBA 와 유사하다. 단 웹 브라우저 상에서 애플릿을 사용하지 않고 액티브 X 컨트롤을 사용한다. 새로이 업데이트된 데이터가 발생하면 서버프로그램에 바인딩된 모든 웹 브라우저상의 액티브 X 컨트롤내의 실시간 투표집계 수치를 변경한다.

## 4 결론

본 논문의 전자투표시스템은 다양한 H/W 와 운영체제에서 수행되어야 한다. CORBA 는 전자투표시스템의 이러한 요구를 만족하는 데 반하여 DCOM 은 MS 사의 윈도우 시스템에 최적화되어 있고, 코딩에 있어서도 CORBA 보다 복잡하다[2][3]. 전자투표시스템은 CORBA 를 사용함으로써 인터넷에서 위치 투명성을 제공하고, 애플릿 프로그램으로 웹에 실시간으로 개표 상황의 전달이 가능하게 되었다. 앞으로의 연구 방향은 CORBA 를 기반으로 전자 투표시스템의 모든 사양을 충족하는 전자투표시스템을 개발하는 것이다.

## 5 참고문헌

- [1] 이용석, "전자민주주의의 대두와 발전방향", <http://srd.nca.or.kr/BBS/info/tp/1997/4-19/nou4-19.html>
- [2] R. Orfali, D. Harkey, "Client/server Programming with JAVA™ and CORBA", 1998
- [3] P. E. Chung, Y. Huang, S. Yanjnik, D. Liang, J. C. Shih, C. Y. Wang, and Y. M. Wang, "DCOM and CORBA Side by Side, Step by Step, and Layer by Layer", Available [http://www.bell-labs.com/~emerald/dcom\\_corba/Paper.html](http://www.bell-labs.com/~emerald/dcom_corba/Paper.html)
- [4] Lorrie Cranor, "Electronic Voting Hot List", Available from <http://www.cerc.wustl.edu/~lorracks/sensus/hotlist.html>
- [5] "Electroic Voting", Available from <http://info.acm.org/crossroads/xrds2-4/voting.html>
- [6] Object Management Group, RealTime Technologies Request for Information, OMG PTC Document ORBOS/96-09-02