

분산 객체를 이용한 웹기반 실시간 경매 시스템

김완수*, 김식**

*세명대학교 일반대학원 전산정보학과 정보통신전공

** 세명대학교 정보통신학과

e-mail:wskim@telcom.semyung.ac.kr

Web based Real time Auction System using Distributed Object

Wan-Soo Kim*, Shik Kim**

*Dept. of Computer & Information Science, Semyung Univ.

**Dept. of, Information & Communications, Semyung Univ.

요약

기업이나 일반 사용자의 인터넷 이용이 급증하면서 인터넷을 통한 전자 상거래가 증가되고 있으며, 일반적인 생활형태의 한 방법으로 정착되고 있다. 현재 인터넷을 통해 제공되고 있는 다양한 형태의 서비스 중 경매, 예약, 주식거래 시스템은 클라이언트 역할을 담당하는 Web Browser상에서의 실시간 처리 및 공유 데이터의 일관성 유지가 필수적이다. 본 논문에서는, 웹 상에서 실시간 처리를 위한 하나의 실험 모델로서 경매시스템을 제안하였고, N-tier 모델 하에 RMI 메커니즘을 채택한 객체 지향 분산 프로그래밍 환경의 일종인 HORB[1]를 적용하여 구현하였으며, 클라이언트 수에 따른 실행시간을 비교하는 실험을 통해 웹에서 적용 가능성을 확인하였다.

1. 서론

인터넷의 보급 확대에 의한 전자상거래가 급격히 증가하고 있다. 국내 전자상거래의 규모는 99년 21억7000만 달러였고, 2003년이면 96억1000만 달러로 4.5배 성장하리라 예상되며 사이버무역의 비중도 99년 4.6%에서 30.4%로 6.5배로 성장할 것으로 전망된다[2]. 앞으로 21세기는 전자상거래가 보편화되어 일상생활과 불가분의 관계가 될 것이다. 그러나 현재 전자상거래는 폭발적인 수요에 비해 시스템 상의 여러 가지 문제점을 내포하고 있다. 본 논문에서는 전자상거래 중 웹 상에서 실시간 처리에 대한 제반 문제를 살펴보고 그 해결방안을 모색하고자 한다. 인터넷상에서 제공되는 다양한 형태의 서비스 중 경매 시스템, 좌석예약 시스템, 주식정보, 뉴스속보, 화상회의, 항공 항로 시스템들은 실시간 서비스 제공을 필요로 한다. 현재 웹 기반 경매시스템의 경우 입찰

자가 입찰 후 경매 상황을 실시간으로 확인하기 위해서는 입찰자에 의한 재전송 요구가 필수적이다. 따라서, 경매시스템과 입찰자 사이에 공유 데이터의 일관성(consistency) 유지 문제가 발생하게 된다.

2월 29일 서울->부산행 조회사항입니다.

확인	입찰번호	출발시간	도착시간	잔여량		
				일반석	특급	사무석
<input type="radio"/>	17	12:00	16:24	2	0	0
<input type="radio"/>	19	13:00	17:25	0	0	11
<input type="radio"/>	45	13:25	17:55	116	35	0
<input type="radio"/>	21	14:00	18:27	72	94	49
<input type="radio"/>	23	15:00	19:25	37	90	45
<input type="radio"/>	3	15:00	20:10	27	-	35
<input type="radio"/>	25	17:00	21:31	189	-	49
<input type="radio"/>	27	18:00	22:24	87	97	43
<input type="radio"/>	29	19:00	23:29	0	43	41

신용카드로 예약할 절차를 확인후 선택하세요.

예약 예약요금조회 운행시간

(그림 1.1) 인터넷 상의 실시간 서비스 사례1

이러한 공유 데이터 일관성 유지 문제는 좌석 예약 시스템에서도 동일하게 발생한다. 그림 1.1의 철도 좌석 예약 시스템을 보면 사용자가 화면을 보

고 있는 시점에서 예약 가능한 좌석 수를 쉽게 파악할 수 있다. 그러나 보는 시점에서 열차번호 17의 일반실 2는 5초 전에 0으로 표시되어 있다가 사용자가 재전송 요구를 하고 나서 2로 증가되었다. 만약 사용자가 데이터 재전송 요구를 하지 않았다면 변경된 좌석정보는 알 수 없었을 것이다.

이러한 문제를 해결하기 위한 방안으로서, 그림 1.2의 인터넷 실시간 뉴스 서비스 시스템은 클라이언트가 일정 간격으로 서버에 공유 데이터의 재전송 요구를 하는 방법을 사용하고 있다. 그러나 이 방법은 웹 브라우저의 새로고침 버튼을 사용자가 누르는 것과 동일한 방법으로서 변경되지 않은 불필요한 데이터까지 전부 재전송 받게되고, 사용자에게 꼭 필요한 데이터의 변화가 지연되며, 뉴스속보와 같은 경우 서버가 클라이언트에게 즉각적인 서비스를 제공할 수가 없다는 단점이 있다.



(그림 1.2) 인터넷상의 실시간 서비스 사례2

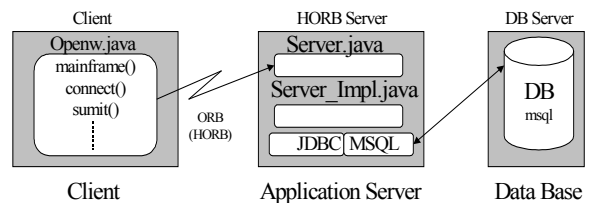
본 논문에서는 HORB에서 제공되는 Callback 메커니즘을 이용하여 공유 데이터의 실시간 처리와 불일치성의 문제를 해결하고자 한다. HORB는 RMI 메커니즘에 기반한 분산 미들웨어의 일종이며 Java RMI보다 두 세배 빠른 성능을 자랑한다[3]. 서버는 현재 접속중인 클라이언트들의 리스트를 유지하며, 특정 클라이언트에 의한 서버 측 공유 데이터의 변경이 발생하면, 서버는 변경된 데이터만을 접속중인 클라이언트들로 전송하므로 전달되는 데이터의 양을 최소화하였다. 시스템 작동 중 네트워크 연결 문제나 클라이언트의 결함이 발생할 경우, 서버 측에 유지되고 있는 클라이언트들의 접속 리스트 중 결함이 발생한 클라이언트 정보가 자동적으로 삭제되도록 하였다. 또한, N-tier 모델이 갖는 장점을 토대로 JDBC를 이용, 이 기종의 DBMS들을 다른 물리적 위치 상에 배치시켜 주 DBMS의 손상이 발생하더라도 보조 DBMS를 사용하여 데이터의 사본을 유지할 수 있도록 하였다. 이러한 N-tier 모델은 기존에 사

용하고 있는 DBMS들의 정보통합을 클라이언트들에게 투명하게 유지시켜 줄 수 있다[4]. 분산 객체를 이용한 일관성 있는 실시간 경매 시스템 모델의 설계 및 구현을 통해, 제시한 모델이 효율성과 적용가능성을 가지고 있는지 실험을 통해 확인하고, 웹 기반에서 실시간 시스템의 구성 모델을 제시하고자 한다.

2. 웹 기반 실시간 경매 시스템의 구조

현재 많은 전자상거래 모델들은 Client/Server 모델과 같은 2-Tier 모델을 갖고 있다. 이러한 2-Tier 모델은 클라이언트 유지보수의 문제점, 무거운 클라이언트(heavy client) 구조, DB 서버의 과부하 초래, 객체 재사용의 문제, dirty-write 문제 등을 수반하고 있다[5]. 이 문제에 대한 해결책으로 3-Tier 모델이 제안되고 있으며, 3-Tier 모델은 전형적인 2-Tier 모델의 한계를 극복하기 위해서 DBMS 서버와 클라이언트 사이에 비즈니스 프로세스 역할을 담당하는 중간계층을 추가한 형태이다. 그러나 3-tier 모델의 문제점은 시스템을 구축할 때 비즈니스 프로세스 층의 구현이 어렵다는 것이다. 이러한 문제 해결을 위해 분산 미들웨어를 사용하면, 클라이언트와 중간층(비즈니스 프로세스층) 사이의 프로그램이 간단하게 되고, 초기 개발비용 삭감과 보수성, 확장성이 뛰어난 시스템을 구축 할 수 있다[6].

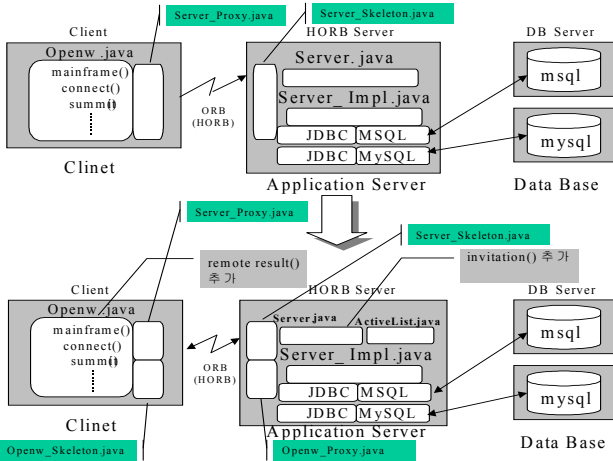
그림 2.1은 분산 미들웨어를 사용한 3-Tier 구조로서 클라이언트는 Java Applet 형태로 서버로부터 다운로드 된다. Application 서버는 비즈니스 프로세스 층으로서 Web Server와 분산 middle-ware platform인 HORB가 위치하며, DBMS는 독립적인 사이트에 위치하여 비즈니스 층과 통신한다.



(그림 2.1) HORB를 사용한 3-Tier 구조

비즈니스 프로세스 층의 시스템 구축 시 구현의 난점을 극복한 이러한 3-tier 모델에 공유 데이터 보호를 위해 또 다른 물리적 DBMS를 위치시킨 모습을 그림 2.2의 상단에서 볼 수 있다. JDBC를 사용하

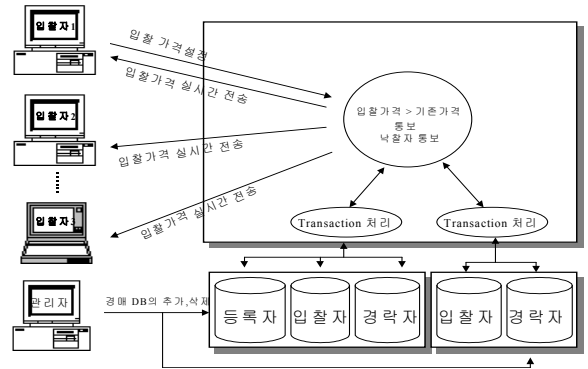
므로서 데이터베이스에 대해 각각의 독립적인 데이터베이스 API 이식성에 대해 고려 할 필요가 없다 [7]. 데이터 베이스 연동에 관한 부분은 서버의 객체 안으로 숨어버리게 되어, 클라이언트 측은 DBMS의 위치 및 종류에 무관하게 된다[8].



(그림 2.2) 일관성 유지와 실시간 처리를 위한 3-Tier 구조

분산 객체를 이용한 일관성 있는 실시간 경매 시스템의 구조는 그림 2.2의 하단과 같다. HORBB 서버에 비즈니스 논리를 가지는 중간층 객체에 Callback method를 추가하였고, Activelist.java를 추가하여 클라이언트들의 정보를 Vector에 자동으로 등록하게 하였다. Callback처리를 위해 서버 측에는 Openw_proxy와 Server_Skeleton 클래스, 클라이언트 측에는 Openw_Skeleton과 Server_Proxy 클래스를 Applet의 다운로드를 통해 자동적으로 위치시켰다. Transaction 처리는 JDBC 벤더들의 Transaction[6] 모델에 의존하였고, 한 클라이언트의 요구는 두 DBMS에 자동 등록되어 주 DBMS가 결함이 발생되면, 보조 DBMS의 데이터를 이용 수동 복구하도록 하여 데이터를 유지 할 수 있게 하였다. 클라이언트들은 인터넷에 연결된 곳이면 위치에 구애받지 않아도 되고, 등록된 클라이언트들의 공유 데이터 변경은 다른 모든 클라이언트들에게 자동적으로 전송된다. 서버 쪽에서 모든 클라이언트에게 Callback 기능을 통해 변경된 데이터를 전송 할 때 네트워크 에러나 데이터를 전송할 수 없는 경우가 발생되면 Exception 처리를 통해 문제점을 자동 감지하고 Activelist에서 Exception 처리된 클라이언트 정보를 삭제하여 서버 쪽의 Callback 부하를 절감하

게 된다.



(그림 2.3) 경매 시스템의 동작

경매 시스템의 동작은 그림 2.3처럼, 특정 클라이언트가 입찰 가격을 제시하게 되면 비즈니스 층에 통보된다. 비즈니스 서버는 다른 물리적 위치상의 주 DBMS에서 가격을 로드하고, 통보된 가격과 비교하여, 비교된 가격이 현재 가격 보다 낮다면 처리하지 않는다. 사용자의 입장에서 변동되는 입찰 가격은 항상 자신의 화면에 보이므로 낮은 가격은 대부분 설정하지 않게 된다. 입찰가격이 현재가격보다 높다면 회원 이름과 key-code와 가격이 주 DBMS와 보조 DBMS에 저장된다. 관리자가 새로운 경매물품을 주 DBMS와 보조 DBMS에 동일하게 등록 및 삭제하는 것이 불편하므로 웹 상에서 자동적으로 변경할 수 있도록 하였다.

3. 웹 기반 실시간 경매 시스템의 구현

3.1 개발 환경

<표 3.1> 개발 환경

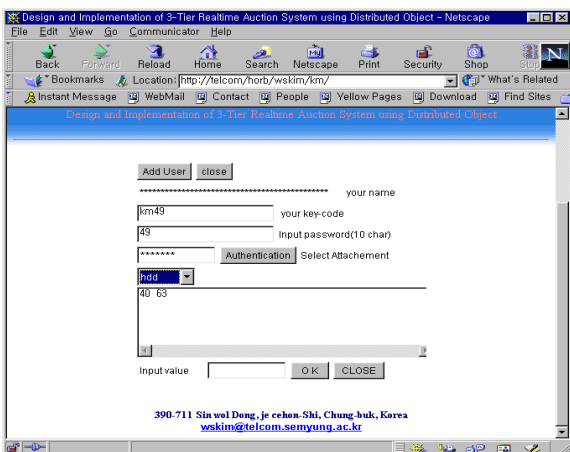
소프트웨어	제품	버전	용도
언어	JDK	1.1.7	Java Applet 작성
분산 미들웨어	HORBB	1.3b1	ORB
DBMS	mSQL	2.0.3	DBMS
DBMS	mysql	3.23.0	DBMS
DB인터페이스	JDBC		Java Enterprise API
JDBC Driver	mSQL driver	0.9b	mSQL driver
JDBC Driver	mysql driver	1.2b	mysql driver

3.2 실행화면

클라이언트가 웹 기반 경매시스템에 접속할 때 사용자 인터페이스 부분은 Java Applet으로 되어있

어 자동으로 클라이언트 측으로 전송된다. 전송된 인터페이스를 통해 사용자 명, key code, 암호를 입력한 후 Authentication 버튼을 클릭하면, 비즈니스 층에 입력 값이 전달되고, 비즈니스 층은 주 DBMS의 회원명단과 패스워드를 비교하여 연결을 결정한다. 연결 결정 확인 후 클라이언트 정보를 Vector List에 추가시키고, 비즈니스 층의 분산객체 클래스와 Callback 연결 설정 관련 클래스들을 다운로드하게 된다. 하나의 클라이언트에 의해 입찰 가격이 설정되면 현재 가격과 비교하여 기존가격보다 높다면 Server는 주 DBMS와 보조 DBMS에 값을 등록 또는 갱신하고 HORB Server의 Callback 기능을 통해 각 클라이언트들에게 가격과 코드번호를 통보하게 된다. 입찰 가격이 경매 상한가에 도달하거나 더 이상의 입찰 가격이 등록되지 않으면 현재 최고 가격을 당락가로 결정하여 주 DBMS와 보조 DBMS에 등록하게 된다. 본 논문에서는 mSQL[9]과 mysql[10]을 사용하여 실험하였지만, mSQL이나 mysql이 아닌 기타 JDBC를 지원하는 여러 DBMS를 사용할 수 있으며, 다수 DBMS의 동시사용도 가능하다.

인증된 사용자는 자신의 경매 품목을 선택하여 값을 등록하게 되고 이때 등록된 값은 그림 3.1처럼 자신의 해당영역 뿐만 아니라 다른 모든 클라이언트들의 해당영역에 함께 보여지게 된다. 이 때 Applet의 해당영역(Text Area)만을 변경하므로 통신량을 줄이고, 브라우저의 응답성을 좋게 하였다.



(그림 3.1) 웹 기반 실시간 경매시스템 접속 화면

새로운 사용자 등록은 Applet 상에서 user add 버튼을 클릭하면 그림3.2와 같이 open window을 생성

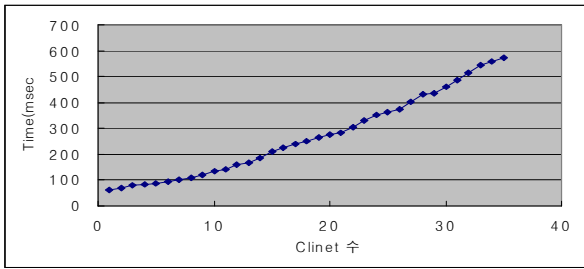
시키고, 모든 입력 처리 후 최종 입력 버튼을 클릭하면 기존 회원과 중복확인 및 비즈니스층과 연결설정이 이루어지고 DBMS와 연결을 하게 된다. DBMS에 등록 된 직후의 사용자가 경매 참석을 위해 Authentication 버튼 클릭시 다시 한번 연결시도를 하기 때문에 static variable을 이용하여 반복 연결하는 문제를 해결하였다.



(그림 3.2) 사용자 등록 화면

4. 실험 및 고찰

본 논문에서 제안한 웹 기반 실시간 경매 시스템의 성능 분석을 위해 특정 클라이언트에서 값을 변경한 시점부터, 변경된 값이 비즈니스 층에서 주 DBMS의 값과 비교한 다음 주 DBMS와 보조 DBMS에 정상적으로 처리되고, 비즈니스 층에서 모든 클라이언트에게 통보되기까지의 시간을 측정하는 실험을 하였다. 클라이언트 수에 따른 응답속도를 실험한 결과 그림 4.1에서 보는 것과 같이 클라이언트 수에 따라 응답속도가 비례적으로 증가됨을 확인하였으며, 실험을 통해 적정수의 클라이언트 수를 유지한다면 분산 객체를 이용한 실시간 시스템 모델이 인터넷에 적용 가능하다는 것을 확인하였다. 문제점으로는 DBMS사용시 각 벤더의 DBMS를 이용하므로 transaction[11] 부분은 DBMS에 종속적일 수밖에 없고, DBMS의 성능 및 기능이 전체 시스템에 영향을 주기 때문에 DBMS의 선택이 중요하며, Callback 처리의 overhead 때문에 적정수로 클라이언트 접속을 제한해야 한다.



(그림 4.1) 클라이언트 증가에 따른 실행시간

본 논문을 통해 구현한 시스템의 장점은 다음과 같이 요약될 수 있다.

- HORB의 Callback 메커니즘을 이용하여 변경된 데이터를 클라이언트에게 실시간으로 제공하였다.
- 보조 DBMS를 사용하여 데이터의 손실에 대비하였다.
- Java 언어를 이용하여 하드웨어 구조에 무관하게 작동한다.
- N-Tier 모델이 갖는 장점을 통해 이기종의 DBMS를 사용할 수 있으며, 사용자에게 DBMS에 대한 투명성을 제공한다.
- HORB의 프로그램 방식은 많은 노력 없이도 자바프로그램 자체의 개발과 동일하게 작성할 수 있어 개발이 용이하다[12].
- Java에 의해 단기간, 저비용으로 개발이 가능하며, 설계시 메모리 관리를 고려하지 않는 장점이 있다.
- 실시간 처리시 클라이언트측의 브라우저 응답성이 나빠질 수 있으나, Callback 기능을 사용하여 브라우저의 표시 영역 중 변경될 일부분만을 갱신함으로써 이 문제를 해결하였다.

5. 결론 및 향후 연구방안

본 논문에서는 분산객체를 이용한 웹기반 응용프로그램들의 실시간 처리방안을 제시하였고, 하나의 실험모델로서 경매시스템을 구현하였다. 실험을 통해 적정수의 서비스를 하는 웹 응용프로그램에 적용가능함을 알았다. 그러나 사용자에게 즉각적이고 일관성 있는 데이터를 서비스하기 위해선 적정 사용자의 유지가 선행되어야 한다. 향후 연구 과제로는 본 논문에서 제시한 분산객체를 이용한 실시간 웹기반 경매시스템의 DBMS 손상시의 자동 복구 기능, 전체 시스템의 성능 향상 및 신뢰성 향상을 위한 분산

객체 기술의 결함허용(Fault-tolerance) 추가 및 DBMS의 자동 백업기능, 이기종 DBMS을 위한 transaction 등의 문제를 연구 하고자 한다.

참 고 문 헌

- [1] HIRANO, S., HORB, "Distributed Execution of Java Programs", Worldwide Computing and Its Applications, Lecture Notes in Computer Science 1274, pp.29-42, 1997
- [2] 전자신문 <URL: <http://www.etnews.co.kr>> 2000. 2 .16 전자상거래 경제와 산업의 중심축으로 부각
- [3] HIRANO, H., Yasu, Y. and Igarashi, H., "Performance Evaluation of Popular Distributed Object Technologies for Java", ACM Workshop on High-Performance Network Computing for Java, 1998.
- [4] Robert Orfali, Dan Harkey, "Client/Server Programming with JAVA and CORBA", John Wiley & sons, 1997
- [5] George Reese, "Database Programming with JDBC and JAVA", O'Reilly, 1996
- [6] Copyright d-tec Distributed Technologies GmbH, 1998. <URL:<http://www.corba.ch/e/>>
- [7] Jeri Edwards with Deborah DeVoe, "3-Tier Client/Server At Work", John Willy & SONS, 1997
- [8] AMJAD UMAR, "Client/Server Internet Environments", Prentice Hall, 1997
- [9] J. Hughes, Mini SQL <URL:<http://www.hughes.com.au>>
- [10] MySQL <URL:<http://www.mysql.com>>
- [11] Graham Hanilton, Rick Cattell, Maydence Fisher, " JDBC Database Access with Java", ADDISON-WESELY, 1997
- [12] Yamanaka A., Nakajina S., and Tomono M., "A HORB-Based Network Management System", ICODP'97, pp.220-224, 1997.