

분산 객체 미들웨어를 이용한 NMS HelpDesk 시스템 구축 사례

김정수*, 이윤재*, 서상구**
*한국통신기술 초고속사업팀
**광운대학교 경영정보학과

e-mail : {projskim, yoonjae}@kti.co.kr, skseo@daisy.kwangwoon.ac.kr

A Case Study on NMS HelpDesk System using Distributed Object Middleware

Jeong-Su Kim*, Yoon-Jae Lee*, Sang-Koo Seo**

*Dept. of Information Super Highway Business Team, KTI(Korea Telecom International)

**Dept. of Management Information System, Kwangwoon University

요 약

현재 국내에서 가장 많이 쓰이는 분산 미들웨어 기술은 크게 세가지로 구분할 수 있다. OMG(Object Management Group)의 CORBA(Common Object Request Broker Architecture), 마이크로 소프트사의 DCOM/COM+ 그리고 썬(SUN)사의 EJB(Enterprise Java Beans)이다. 이 분산 미들웨어 기술들은 분산환경에서 시스템들간의 운영체제나 프로그래밍 언어에 관계없이 객체 서비스를 제공하여 독립적이고 표준화된 환경을 제공한다. 본 논문에서는 한국통신 초고속사업 중 NMS HelpDesk System(고장관리시스템)의 구축사례를 소개한다. 개발된 시스템은 분산 미들웨어 기술을 이용하여 DCOM/COM 응용시스템, CORBA 서버, 데이터베이스 응용시스템, 그리고 최종 운영자용 웹 인터페이스 시스템 등으로 구성된 대규모 분산시스템이다.

1. 서론

고성능 PC와 충분한 대역폭을 제공하는 네트워크에 의해 분산 컴퓨팅이 가능해짐에 따라 기업 정보 시스템은 다양한 이기종 컴퓨터 환경에서 데이터와 응용 프로그램을 분산 운용하면서 프로그램의 상호 운영성과 사용자에 분산의 투명성을 제공할 수 있게 되었다. 규모가 큰 통신업체들이 소유하고 있는 중앙시스템은 원거리상에서 원격으로 각 지역의 전화국이나 기지국을 운영, 진단 조치 및 처리하는 전형적인 분산 시스템을 포함한 NMS(Network Management System) 시스템의 예라고 할 수 있다. 이러한 분산 시스템상의 NMS 시스템은 원격 시스템간 자원 공유, 개방성, 시스템 사이의 병렬성, 확장성, 고장 감내성, 위치 투명성 등을 제공해야 한다. 객체지향 기술이 제공하는 추상화, 캡슐화, 상속성, 다형성 등의 성질은 분산 객체 프로그래밍을 할 수 있게 했으며 소프트웨어 생산성을 보장하는 소프트웨어 개발 방법론으로 성장했다. 또한 객체는 데이터와 그 데이터를 조작할 수 있는 메소드(method)로 구성돼 있으며, 프로그램은 메시지에 의한 객체간 상호 작용을 구현함으로써 여러 문제를 해결할 수 있게 되었다. 분산 객체 기술은 이러한 두 기술의 장점을 효과적으로 통합하는 기술이며, 개발자에게 애플리케이션 개발의 생산성을 향

상시켜주고, 사용자에게는 투명하게 통합된 정보를 제공한다.^[1]

이를 위해 OMG에서 CORBA라는 이기종의 분산된 환경을 통합하는데 필요한 기능으로서 분산 시스템들간의 운영체제나 프로그래밍 언어에 관계없이 객체 서비스를 제공하여 독립적이고 표준화된 환경을 제공한다. CORBA에서는 IDL(Interface Definition Language)이라는 표준 언어를 제공하며 클라이언트 객체와 구현 객체 사이의 인터페이스를 정의한다. ORB(Object Request Broker)는 클라이언트와 구현 객체사이의 중계 역할을 수행하며 여러 개의 컴포넌트와 구현 객체사이의 중계 역할을 수행하는 여러 개의 컴포넌트와 인터페이스로 구성되어 있으며 분산된 객체들을 클라이언트/서버 관계로 만들어 주는 객체 미들웨어이다. ORB는 클라이언트 객체가 구현 객체의 함수를 호출할 때 ORB는 클라이언트의 호출을 받아 구현 객체의 정보를 참조하여 클라이언트가 원하는 구현 객체를 찾고 클라이언트에서 보내준 서비스 요청 정보를 해당 구현 객체에게 전달한다. 구현 객체의 함수가 완료되면 ORB는 결과 값을 받아 클라이언트에게 전달하는 방식이다.^{[2][3]}

마이크로소프트사의 DCOM은 COM 컴포넌트를 다른 컴퓨터에서 수행 가능하게 만드는 환경을 제공한다. 클라이언트의 메소드 호출은 네트워크를 통해 다른

시스템에서 수행되는 컴포넌트로 전달하는 방식으로 이루어진다. 또한 COM은 클라이언트 프로그램과 객체 서버 프로그램간의 상호 통신방식을 정의한 모델이라 할 수 있는데 이런 방식을 이용한 기술로 OLE와 ActiveX 기술의 기초가 되는 개념이며, 인터페이스는 미리 정의된 루틴의 세트를 통해 각 객체들간의 상호운용을 가능하게 해주는 객체기반의 프로그래밍 스펙이다.

센사의 EJB는 멀티 티어(multi-tier)의 분산형 객체지향 자바 애플리케이션을 개발하고, 보급하기 위한 컴포넌트 아키텍처이며 서버 파트에서 확장이 가능하고 트랜잭션과 엔터프라이즈 애플리케이션에 적합한 형태로 제작된 스펙이다. 이는 트랜잭션 처리 모니터(TP Monitor), 웹 서버, 데이터베이스 서버, 애플리케이션 서버 등과 같은 트랜잭션 처리 시스템에서 운영된다. 이 세가지 분산미들웨어 기술의 특성과 관련 제품들을 보면 아래 표 1과 같다.

특성	비즈니스	제공업체	
CORBA	<ul style="list-style-type: none"> 상속성과 다형성이 좋다. 네트워킹 프로그램에서 발생하는 수고를 없게 만들어 준다. (프로그램의 양이 줄어들어 개발 시간을 줄일 수 있다.) 	<ul style="list-style-type: none"> ObjectDirector Orbix ORBPPlus PowerBroker Solaris NEO SOMobjects VisiBroker 	<ul style="list-style-type: none"> Fujitsu IONA Hewlett-Packard Expersoft SunSoft IBM Visigenic
DCOM /COM+	<ul style="list-style-type: none"> 사용하기 쉽다. 컴포넌트 방식이므로 쉽게 확장할 수 있다. 유지보수가 쉽다. 프로그램 개발자들은 비즈니스 로직만 구축 하면 된다. 	<ul style="list-style-type: none"> Microsoft 제품군 EntireX 기타 3rd party 제품 	<ul style="list-style-type: none"> Microsoft 벤타컴퓨터
EJB	<ul style="list-style-type: none"> 애플리케이션 개발 기간이 단축된다. 폭 넓은 분야에 적용 가능하다. 애플리케이션 이식성이 우수하다. 벤더 비선택 서버간 이식성이 우수하다. 서버간 확장성이 우수하다. 	<ul style="list-style-type: none"> JRun Server 3.0 WebLogic Application Server Application Server WebSphere iAS 	<ul style="list-style-type: none"> Allaire BEA Inprise iPlanet IBM Oracle

표 1. 분산 미들웨어 기술의 특성 및 관련제품

한국통신 인터넷운영국 NMS HelpDesk은 보다 빠른 양질의 서비스를 고객에게 제공해 주기 위한 시스템이다. 이 사업은 '98년 하반기부터 추진하였으며 현재, 내부적으로 여러 개의 CORBA 서버(UNIX), DCOM/COM 및 PL/SQL을 사용하여 개발한 후 NMS HelpDesk가 운용되고 있으며, 지속적인 유지보수 및 하부 프로젝트 개발이 추진중이다. 본 논문에서는 이 개발 내용을 간략히 소개하고자 한다. 2절에서 NMS HelpDesk 시스템의 개요, 3절에서 NMS HelpDesk 시스템 구조, 4절에서 NMS HelpDesk 시스템 세부 동작 및 CORBA와 데이터베이스 핸들링, 마지막 5절 결론에서 향후 유사 시스템 개발시 조언 및 연구개발방향을 제시하였다.

2. NMS HelpDesk 시스템 개요

초고속 NMS HelpDesk 시스템은 가입자나 통합망관리 시스템으로부터의 정보 요청이나 불만사항, 장애신고 사항 등을 접수하여 필요한 정보를 안내하거나 문제점을 해결하는 전과정을 체계화한 시스템이다.

NMS HelpDesk는 HelpDesk Call 관리시스템, HelpDesk 장애관리시스템 및 HelpDesk 정보검색시스템의 3개 시스템으로 구성된다. HelpDesk Call 관리시스템은 고객과의 점점 제공을 위한 호 접수 및 분배 기능(호 관리기능), 서비스개통현황 및 가입약관 안내 기능(가입응대기능), 과금내역에 대한 문의 및 불만사항을 처리 기능(과금응대기능), 서비스장애나 설비고장신고 접수 기능(고장응대기능)을 대상영역으로 하며, HelpDesk 장애관리시스템은 장애위치진단 및 장애복구절차관리기능(장애관리기능)을 담당하고 HelpDesk 정보검색시스템은 운용지침서 및 장비매뉴얼을 대상으로 한 정보 DB 구축 및 검색서비스(정보검색기능)를 대상 영역으로 한다.

가입응대기능과 과금응대기능은 전화나 E-mail, Fax 등 다양한 매체를 통해 고객의 문의사항을 접수하고, 사안에 따라 관련 정보를 안내하거나 적절한 관리시스템에 중계하는 업무를 지원한다. 고장응대기능은 가입자로부터 서비스장애나 설비고장신고를 접수하여 장애진단 및 복구절차를 가동시키고, 그 결과를 가입자에게 안내하는 작업을 담당한다. 직접 처리가 가능한 사항인 경우에는 처리 작업을 실시하고 그 결과를 안내하며, 불가능한 경우에는 가입자장애티켓(Customer Trouble Ticket)을 발행하여 장애관리기능으로 장애복구절차를 이관할 수 있도록 지원한다. 장애관리기능은 고장응대기능에서 발행한 가입자장애티켓과 통합망관리시스템에서 발행한 공급자장애티켓(Provider Trouble Ticket)을 접수하여 장애리포트(Trouble Report)를 생성한 후, 장애진단에서 장애복구종료에 이르기까지의 장애복구절차 관련 전 과정을 지원하는 기능을 담당한다.

NMS HelpDesk는 초고속망 운영과 관련된 각종 정보들을 세분화하여 관리하는 외부시스템들과 상호 유기적으로 연동하여 타 시스템에서 생성/관리하는 정보를 수집하고 반대로 외부시스템에 필요한 정보를 제공한다.

3. NMS HelpDesk 시스템 구조

한국통신 초고속망 내에 NMS HelpDesk 시스템 구성도 및 주요시스템 S/W 구성은 아래의 그림 1과 같다. NMS SDMS는 가입자에 대한 정보를 관리하는 시스템이며, NMS PS는 장비에 대한 정보, 그리고 MADB는 이중화 서버이다. 이런 시스템들 중 NMS HelpDesk 시스템의 구성에 있어 크게 세가지 요소(그림 1-①, ②, ③:::Faxsvr1, m1570, HelpDesk)로 나눌 수 있으며 투입된 개발 기술은 먼저 Faxsvr1에서는 ActiveX(VC++6.0) 및 통지서버개발(VB6.0), Iona사의 Orbix2.3C, OrbixCOMet1.0C(COM::VBScript), DCOM 서버, MTS로 개발하였다. m1570에서는 Orbix2.3C를 이용한 CORBA Toss GateWay(VC++6.0) 방식으로 개발되었고, HelpDesk에서는 Orbix2.3C0.3을 기반으로 데이터베이스를 핸들링하기 위해서 ProC*를 이용하였다. 운영자에게 최종으로 서비스를 해주기 위해서 PL/SQL과 JAVA(AWT1.2::Bar/Pie Graph)로 개발 및 구성되었

다. 그림 1을 기본으로 동작원리를 간략히 설명을 하면 10개의 국사(서울, 부산, 경기, 강원, 전북, 전남, 충남, 충북, 대구, 제주본부)에 있는 담당 전화국 운영자들이 KTS 망을 통해 통신망 관리단의 시스템으로 들어온다. 제일 처음(그림 1-①)으로 접속되는 시스템은 Faxsvr1이다. 이곳에는 DCOM 서버, 통지서버, COM 서버가 개발되어 설치되어 있다. DCOM 서버는 운영자가 가입자의 정보를 알고자 할 때 각 해당하는 모듈 서버에서 처리하여 가입자의 정보를 알려주는 역할을 한다. 통지서버에서는 각 10개의 국사에 있는 해당국사 전화국 운영자의 id와 password의 정보를 모니터링 할 수 있다. 예를 들면 전라북도 서군산전 전화국 운영자의 id/password가 kunsan/atm01이라고 하면 이 운영자가 접속할 때 통지서버에 기록이 된다. 운영자들이 몇 명 접속했는지와 중복 접속했는지의 체크 여부, 운영자가 아닌 사람들이 침투함을 방지하는 역할을 한다.

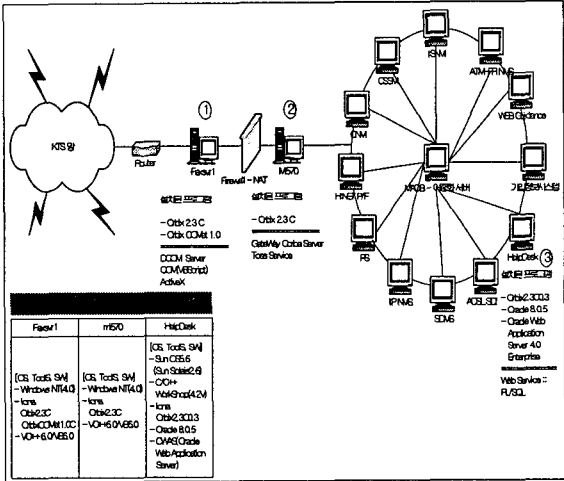


그림 1. 한국통신 초고속망 NMS 시스템 구성도

두번째(그림 1-②)에 접속되어진 시스템은 m1570이다. m1570의 시스템에 들어오기 전에 Firewall(NAT)이 있다. NAT는 IP를 바꾸는 기능을 하며, 이때 IP는 사실 IP 인터넷이 아닌 독립된 망에서 사용할 수 있는 IP이다. 단, CORBA에서는 NAT를 통과하지 못하는 단점이 있다.^[8] 그래서, 한번 더 거치는 CORBA Toss GateWay 프로그램의 모듈을 개발함으로써 Faxsvr1과 m1570 사이의 메시지를 주고 받는 데 성공하였다. 단, 한번 더 CORBA 프로그램을 거치기 때문에 속도는 약간 느려진다는 단점이 있다.

세번째(그림 1-③)로 접속되어진 시스템은 HelpDesk 서버이다. m1570에서 받은 CORBA의 인터페이스명을 처리하며 데이터베이스 핸들링을 한다. 물론 다른 NMS 시스템의 핸들링 및 데이터베이스 핸들링은 내부적으로 처리를 한다. 그리고 OWAS(Oracle Web Application Server)가 설치되어 있어 최종 운영자에게 웹으로 서비스를 한다. 구축되어진 NMS HelpDesk 시스템의 동작을 다음 절에서 세부적으로 분류했으며 간략한 설명을 할 것이다.

4. NMS HelpDesk 시스템 세부 동작 및 CORBA와 데이터베이스 핸들링

그림 2는 NMS HelpDesk의 전체적인 시스템 연동을 보여준다. 사내/외망 접속자(운영자 PC)는 외부에서 들어오는 운영자 PC나 사내에서 담당하는 운영자 PC를 뜻하며, 중계서버 1은 Faxsvr1과 m1570, 그리고 중계서버 2는 NMS HelpDesk와 다른 나머지 NMS 시스템을 말한다. 마지막으로 중계서버 2가 장애가 날 경우 이중화서버 madb로 Take over 하여 수행한다. 4항목의 내용은 DCOM/COM 서버명 및 CORBA 인터페이스명이다.

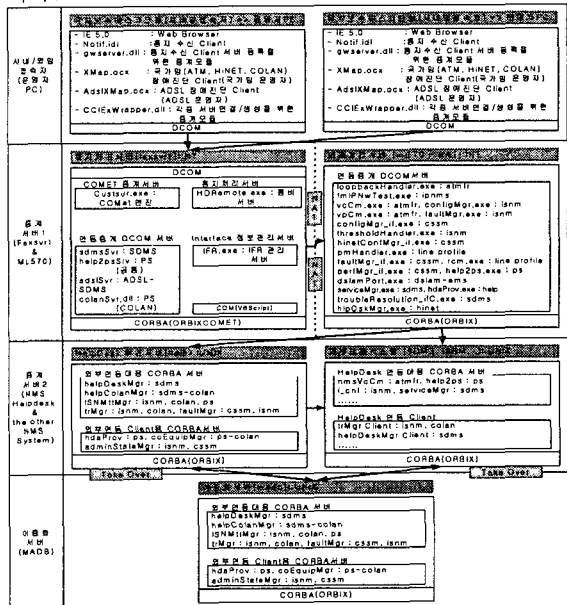


그림 2. 통지/연동시스템의 구성

구축되어진 NMS HelpDesk 시스템의 동작을 세부적으로 분류하면 9가지 형태로 나눌 수 있다.

① Faxsvr1에서 DCOM 서버로 전송 처리 과정

10개의 국사에 소속한 전화국의 운영자가 제일 처음 접속한 후 처리하는 부분이다.

② Faxsvr1에서 통지서버로 전송 처리 과정

운영자 id와 password를 확인하는 부분으로 모든 접속된 운영자들의 리스트를 볼 수 있고 운영자가 아닌 사람들은 접속을 할 수 없다.

③ Faxsvr1에서 OrbixComet1.0C으로 전송 처리 과정

COM과 COBRA를 처리하는 방식을 채택하려면, Orbix2.3C와 OrbixComet1.0C을 기반으로 하여 개발 처리하는 부분이다.

④ m1570에서 CORBA Toss GateWay 방식으로 전송 처리 과정

Faxsvr1과 m1570 사이에 NAT(그림 1. 참조)가 있는데 CORBA가 통과하지 못하므로 한번 더 전달해주는 CORBA Toss Gateway 프로그램을 VC+6.0로 개발 처리하는 부분이다.

● HelpDesk 서버상에서 CORBA 로 전송 처리 과정

m1570 에서 받은 코바 인터페이스명을 HelpDesk 서버에서 내부적으로 처리하는 부분이며, 다른 NMS 시스템과 연동 및 데이터베이스를 핸들링하여 개발 처리하는 부분이다.

◎ CORBA 와 데이터베이스 핸들링 처리 과정

CORBA 와 데이터베이스의 연동이다. 방식은 ProC*로 개발하여 핸들링 하였다. ProC*로 개발하게 된 이유는 ProC*가 native language 이므로 데이터베이스의 쿼리문을 처리할 때나 원하는 결과값을 가지고 올 때 상당히 빠른 성능을 보여주기 때문이다.

☞ HelpDesk 서버상에서 멀티 쓰레딩으로 전송 처리 과정

HelpDesk 서버상에서 데이터베이스를 핸들링하는 ProC*내에서 다른 여러 NMS 시스템으로 동시에 메시지를 전달 처리하는 부분이다.

◎ 최종 운영자에게 전송 처리 과정(PL/SQL 과 COM(VBScript))

운영자 PC ⇨ Faxsvr1 ⇨ m1570 ⇨ HelpDesk 로 처리한 후 다시 HelpDesk ⇨ m1570 ⇨ Faxsvr1 ⇨ 운영자 PC 상에 웹으로 서비스를 해준다. HelpDesk 는 운영자의 웹상에서 서비스 해주기 위해서 오라클 웹서버에서 지원하는 PL/SQL 로 개발을 하였다. PL/SQL 은 데이터베이스와 항상 바인딩이 되어 있는 상태라 데이터 값을 조회, 삽입, 삭제, 갱신에 있어서 상당히 빠른 속도를 내어준다. 또한, VBScript 내에서 CORBA/COM 방식으로 통신하는 방법이며, 간단한 프로그램 모듈 처리 방식은 DCOM ⇨ 통지서버 ⇨ OrbixCOMet1.0C(COM::VBScript) ⇨ Orbix2.3C(CORBA) ⇨ ORACLE ⇨ OWAS(PL/SQL)로 처리한다.^[7] 그리고 자바를 이용하여 고장관리현황 및 진행상황을 Bar/Pie graph 로 운영자에게 서비스를 해준다.

◎ madb 이중화에 대한 설명

madb 는 이중화 서버이며 madb 서버의 역할은 HelpDesk 서버의 시스템 장애가 발생할 경우, 자동적으로 madb 가 감지하여 자동적으로 시스템을 Take-Over 하는 역할을 한다. 물론 Orbix2.3C0.3, Oracle, OWAS 등의 모든 소프트웨어나 응용프로그램들이 자동적으로 올라오는 스크립트를 만들어서 동작을 한다. 처음 madb 서버에 Orbix2.3C 으로 운용하였지만, 후에 기능상의 문제로 Orbix2.3C0.3 로 패치 해야만 했다.

이 시스템 구축시 가장 큰 장점은 운영자가 원격으로 망관리 시스템을 제어할 수 있으며, 장비에 대한 정보 및 장비가 고장이 발생할 경우 원격으로 치료가 가능하고, 망간, 구간 및 장비간 장애진단을 할 수 있다는 점이다. 이를 통하여 가입자들에게 보다 빠르고 보다 좋은 서비스를 할 수 있게 되었다. 분산 미들웨어를 이용하여 시스템 구축시 몇가지 어려운 점이 있었다. 예를들면, ip 가 전체적으로 바뀌었을 경우 셋팅 및 소스상에 수정을 일일이 다 해주어야 한다는 번거로움이 있으며, Firewall(NAT) 구축시 CORBA 의 IIOP 를 통과하지 못하는 단점과 이중화서버에서 Take-Over 상에서 CORBA 가 시스템 회복에 어려운 점이 있었다.

5. 결론

분산 미들웨어 기술을 기반으로 한 웹서비스 시스템이 점점 더 증가 할 것이다.^{[4][6]} 본 논문에서는 분산 미들웨어 기술 중 CORBA 와 DCOM/COM 방식으로 구축된 한국통신 초고속망 NMS 시스템 구축 사례를 살펴 보았다. 이기간의 통합 및 포팅작업에 있어 객체지향개념의 적용에 CORBA 및 COM/DCOM 기술이 아주 적합하지만 너무 많은 시간과 인력 그리고 시행착오 및 초기 비용이 든다는 어려운 점이 있었다. 분산 시스템의 개발방법으로서 처음 설계할 때 너무 복잡하게 설계하지 않는 것이 좋을 것이며, 간단하고 작은 시스템을 개발한 후 서서히 모듈을 확장하는 것이 바람직할 것으로 판단된다.

현재 고장관리시스템은 각 서버에서 하나의 데이터베이스를 핸들링 하였으나, 추후에는 각 서버상의 서로 다른 데이터베이스(Sybase, Informix, -)를 연동하여 처리하는 방식, 각각의 Schema Integration Service(SI), Multidatabase Administrator, Global Database Agent(GDA), Local Database Agent(LDA), Query Processor 를 두어 처리하는 방식^[5]등으로 확장될 수 있을 것으로 보이며 EJB 와 CORBA 를 혼용한 시스템 구축도 가능할 것이다. 또 일반적인 관계형 데이터베이스(Relational DataBase)방식이 아닌 객체지향데이터베이스(OODB:Object Oriented DataBase)의 특성을 이용한 시스템 구축도 시스템성능, 확장성, 유지관리 측면에서 시도할 가치가 있을 것으로 보인다.

참고문헌

[1] 정지훈, 신현목, 김경현, 유형열, "분산객체기술 어디까지 왔나?," 마이크로소프트웨어, 2000년 11월, pp.224-297.
 [2] 나길성, 이상호, "CORBA 이벤트 서비스의 연동방법 및 성능평가," 정보과학회논문지, 제 27 권 제 1 호 2000년 3월, pp.33-41.
 [3] 나길성, 이상호, "CORBA 이벤트 서비스의 구현 기법," 정보과학회논문지, 제 27 권 제 9 호 2000년 9월, pp.530-538.
 [4] A. Bouguettaya, et al, "World Wide Database - Integrating the Web, CORBA and Databases," '99 ACM SIGMOD Conference, May 1999, pp594-596, Philadelphia PA.
 [5] Asuman Dogac, Cevdet Dengi, and M. Tamer Oszu, "Distributed Object Computing Platforms, CORBA offers a useful methodology and middleware for building interoperable databases," Communications of the ACM, Vol. 41, No. 9, September 1998, pp95-103.
 [6] Ralf Kramer, "Databases on the Web : Technologies for Federation Architectures and Case Studies," '97 ACM SIGMOD Conference, 1997, pp503-506, AZ, USA.
 [7] 김동진, 김완석, 이상구, 손덕주, CORBA & JAVA 분산객체기술, ㈜교학사, 1999.07.
 [8] 경영과컴퓨터, "비교 분석 국산 방화벽," 경영과컴퓨터호, 1998.04, pp176-180.