

# CORBA를 이용한 학교간 분산데이터베이스 프로토타입 시스템의 설계 및 구현

최현중<sup>†</sup> · 정상욱<sup>††</sup> · 김태영<sup>†††</sup>

## 요 약

분산객체 기술을 이용한 분산데이터베이스 시스템 개발은 현재의 컴퓨팅 환경을 그대로 사용하면서 적은 비용으로 시스템을 개발할 수 있기 때문에 많이 사용되고 있다. 따라서 현재 학교에서 운용중인 독립형 데이터베이스 시스템을 통합하는 분산데이터베이스 시스템을 개발하면 교사와 교육 공무원 모두에게 쉽게 자료를 통합하고 관리할 수 있는 시스템이 될 것이다. 따라서 본 연구는 CORBA, Java, JDBC를 이용하여 3-Tier 분산 데이터베이스 관리 시스템의 프로토타입을 설계하고 구현하고자 한다.

## Design and Implementaion of Distributed Database System Prototype of Schools using CORBA

Hyun-Jong Choi<sup>†</sup> · Sang-wook Chung<sup>††</sup> · Tae-Young Kim<sup>†††</sup>

## ABSTRACT

The development of distributed databases using a distributed object technology is more popular than ever before because the current computer system environment is adaptable for a use as well as the cost of the integration of legacy systems is cheaper than making a new application. In schools, development of an integrated system consisted of many stand-alone databases give a teacher and an administrator the convenience of managing the database in a school and the ease integration of data in it. Therefore, this research is to design and implement the 3-tier distributed database using CORBA, Java and JDBC

### 1. 서론

21세기로의 발전을 주도하고 있는 정보와 기술은 어느새 교육현장에까지 자신의 영역을 확고히 다져나가고 있다. 이를 뒷받침하기 위해 정부는 97

년부터 “교육정보화” 정책을 일관되게 추진하고 있고, 이제는 학교 현장에서 ICT 교육을 위한 인프라가 하드웨어·소프트웨어적으로 어느 정도 구축되고 있는 실정이다[15]. 하지만, 컴퓨팅 환경이 하루가 다르게 변화하고 있기 때문에 새로운 패러다임으로 인해 기존의 인프라가 이제는 장애물이 되는 경우도 있다. 특히, 네트워크의 발달은 인터넷의 대중화로 인해 기존의 독립형 컴퓨팅 환경에 많은 변화를 가져다 주었다. 학교 현장의 하드웨어·소프트웨어 컴퓨팅 환경도 독립형 시스템이기 때문에, 하드웨어적으로는 학교안의 모든 컴퓨터를

† 정 회 원: 한국교원대학교 컴퓨터교육과 석사과정

†† 종신회원: 한국교원대학교 컴퓨터교육과 박사과정

††† 종신회원: 한국교원대학교 컴퓨터교육과 부교수

논문접수: 2000년 12월 15일, 심사완료: 2001년 2월 20일

\* 본 논문은 2000년도 BK21 사업 핵심분야에 의하여 지원되었음

연결하는 학교망(school net)을 설치하는 학교망 사업이 한참 진행중이다. 하지만, 소프트웨어적 변화는 아직까지 그리 활발하지 못하고 있다. 특히 현재 학교에서 사용중인 데이터베이스는 모두 독립형 데이터베이스이기 때문에 앞으로 어떻게 이 자료를 네트워크를 이용한 데이터베이스로 통합하여 재사용할 수 있을까하는 문제가 생기게 된다 [6].

기존에 이미 구축되어져 있는 데이터베이스를 재사용할 수 있는 방법은 크게 두가지가 있다. 첫 번째 방법은 기존의 데이터베이스를 새로운 데이터베이스 관리 프로그램에서 사용할 수 있도록 새로운 데이터베이스로 변환하여 사용하는 것이고, 두 번째 방법은 기존의 데이터베이스를 새로운 데이터베이스 관리 프로그램이 사용할 수 있도록 미들웨어 객체를 통해 변환해 주는 것이다. 첫 번째 방법이 간단한 해결방법이긴 하지만, 이렇게 될 경우 응용 프로그램에 따라 데이터베이스를 계속 변환해 주어야하는 부담이 생기기 때문에 많은 시스템 개발 업체들은 두 번째 방법을 선택하고 있다. 두 번째 방법을 선택하게 되면 전체 시스템을 계층(layer)별로 나누어 관리 할 수 있기 때문에 프로그램의 개발 및 유지 보수가 쉽고, 기존의 시스템 환경을 그대로 사용할 수 있기 때문에 비용과 사용자 교육에도 많은 이점이 있다[9].

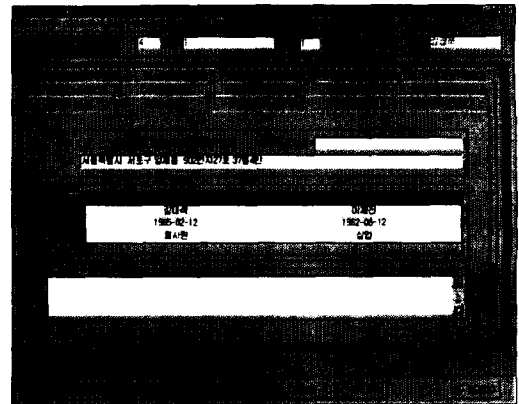
따라서, 본 연구는 학교 환경에서 현재 사용중인 독립형 데이터베이스 관리 시스템을 네트워크를 기반인 데이터베이스 관리 시스템으로 변환할 수 있는 가장 적절한 모형을 선택하여 설계해 보고, 프로토타입 구현 시스템을 통해 앞으로의 설계 및 구현 방법을 모색해보고자 하는 것이다.

## 2. 학교형 데이터베이스 시스템의 현재

현재 학교에서 사용되고 있는 데이터베이스 관리 프로그램은 그 수가 많지 않다. 교육행정에서 쓰이는 데이터베이스, 학교생활기록부 데이터베이스, 건강기록부 데이터베이스가 있다. 이 중에서 전국적으로 동일하게 사용되고 있는 것이 학교생활기록부 데이터베이스이다. 건강기록부 자료도 현재 사용중이기는 하지만, 최근에 구축되기 시작했기 때문에 자료가 많이 누적되어 있지 않다. 하지

만 학교생활기록부는 3년전부터 체계적으로 자료가 구축되어 왔기 때문에 자료가 많이 누적되어 있고 초등학교에서 고등학교까지 체계적으로 관리되고 이용되는 자료이다.

학교생활기록부 프로그램은 마이크로소프트(Microsoft)사의 액세스(ACCESS) 데이터베이스 엔진을 사용한 윈도우용 프로그램으로 기존의 종이 장부로 관리되던 학생들의 생활기록부 자료를 데이터베이스화하는 프로그램이다. 1998년에 프로그램이 처음 보급되어 초등학교 학생들의 자료부터 데이터베이스화하기 시작했다. 데이터베이스화되는 자료는 모두 15개 항목으로 각 학생의 인적사항, 학적사항, 출결사항, 신체발달사항, 수상경력, 자격증 및 인증 취득 상황, 진로지도 상황, 재량활동 상황, 특별활동 상황, 체험활동 상황, 행동발달 상황, 종합 의견, 교과학습발달 상황, 기타 사항이다. (그림 1)은 이 프로그램의 한 장면이다.

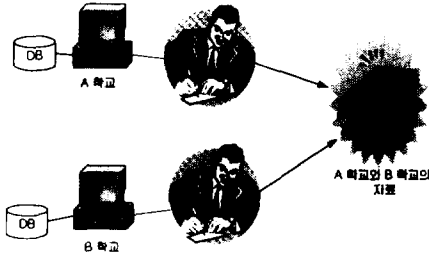


(그림 1) 학교생활기록부 프로그램의 한 장면

학교에 설치되고 있는 학내망의 보급은 학교의 컴퓨팅 환경을 많은 부분에서 발전시켰는데, 특히 데이터베이스 관리 프로그램도 네트워크를 이용한 모델이 선보이게 되었다. 그 첫번째 작품이 바로 종합정보관리 시스템인데, 이 시스템은 네트워크를 이용한 데이터베이스 관리 모델이기는 하지만 시스템의 개발이 기존의 시스템을 활용하는 것이 아니라 새로운 시스템을 도입하는 방법이고, 기존의 데이터베이스를 새로운 데이터베이스로 변환하여 사용하기 때문에 시스템을 구축하는데 많은 비용과 시간이 필요하다.

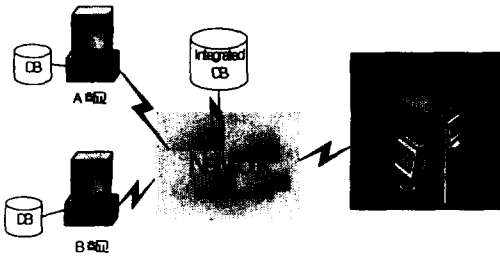
### 3. 학교형 데이터베이스 시스템의 미래

현재 학교에서 운영중인 데이터베이스의 자료를 통합하여 정책 운영의 자료로 사용하기 위해서는 (그림 2)와 같이 수작업을 또 다시 반복해야 한다.



(그림 2) 수작업을 이용한 자료의 통합

하지만, 네트워크의 각 사이트(site)에 분산되어 있는 데이터베이스를 통합하는 분산 데이터베이스 시스템을 적용하면, 실시간(real-time)으로 각 학교의 자료를 필요할 때 통합하여 관리·이용할 수 있다. (그림 3)은 분산 데이터베이스 시스템을 적용했을 때의 사용 모습이다.



(그림 3) 분산 데이터베이스 시스템의 활용

분산 데이터베이스 관리 시스템을 현재의 학교 네트워크 환경에 적용하기 위해서는 분산 객체 시스템을 적용하여 관리 시스템을 개발하는 방법이 대표적이다. 분산객체 시스템 기술로는 OMG(Object Management Group)의 CORBA(Common Object Request Broker Architecture)와 마이크로소프트의 DCOM이 대표적이다. 그 중에서 CORBA는 800여개의 각종 벤더들이 모여 결성한 컨소시엄으로 활발한 활동을 보이고 있으며, 현재 많은 소프트웨어 업체들이 이

표준안에 따른 분산객체 모델을 개발하여 시스템 개발에 이용하고 있다[4][5][17][18].

따라서 본 연구는 대표적인 분산 객체 모델인 CORBA를 적용하여 학교간에 자료를 쉽게 통합하여 관리·이용할 수 있는 데이터베이스 관리 시스템을 개발하고자 한다.

## 4. 연구의 배경

### 4.1 분산 데이터베이스 시스템

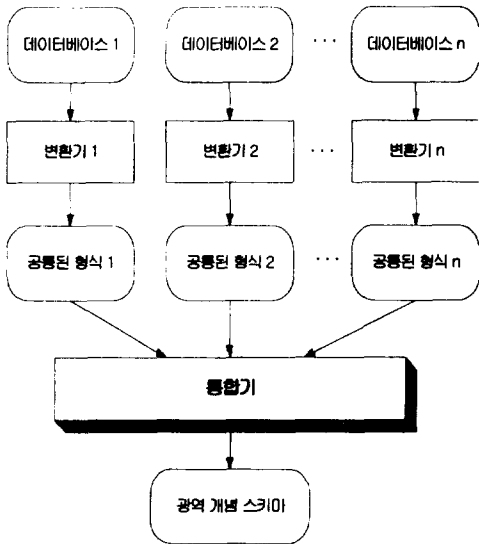
#### 4.1.1 분산 데이터베이스 시스템의 개념

분산데이터베이스란 네트워크에 분포된 다양한 지역 데이터베이스들의 집합을 말한다. 즉 컴퓨터 네트워크상의 각 사이트에 물리적으로 분산되어 있는 공유된 데이터들을 논리적으로 통합한 데이터베이스이다[8][9][10]. Date는 분산데이터베이스 시스템이 가져야할 성격으로 자치성(local autonomous), 균등성(no reliance on a central site), 기능성(continuous operation), 지역 투명성(location independence), 분할 독립성(fragmentation independence), 복제 독립성(replication independence), 분산 쿼리 실행성(distributed query processing), 분산트랜잭션관리(distributed transaction management), 하드웨어 독립성(hardware independence), 운영체제 독립성(operation system independence), 네트워크 독립성(network independence), DBMS 독립성(DBMS independence)을 들고 있는데[8], 이는 분산데이터베이스 시스템의 편리함과 복잡함을 동시에 잘 나타내고 있다.

#### 4.1.2 분산 데이터베이스 시스템의 설계

분산데이터베이스 설계는 Ceri가 제시한 하향식 접근방법(top-down approach)과 상향식 접근방법(bottom-up approach)이 있다[14]. 하향식 접근방법의 설계는 요구사항 분석, 개념 설계와 뷰(view) 설계, 분산 설계, 물리 설계의 순으로 진행되는데 분산데이터베이스를 목적으로 분산되는 지역데이터베이스를 설계하고 구현하는 방법으로 처음부터

시스템을 구축하는 경우에 적합한 설계 방법이다. 상향식 접근방법의 설계는 이미 구축되어 운용되고 있는 데이터베이스를 통합하는 목적으로 분산 데이터베이스를 설계할 때 적용하는 방법으로 (그림 4)와 같이 각 지역데이터베이스를 공통 모델에 의해 통합될 수 있는 데이터베이스로 변환하여 시스템을 구축한다. 이 때 가장 중요한 것은 각 지역 데이터베이스를 광역개념 스키마(global concept schema)로 변환하는 하나의 공통된 형식 모델의 선택이다. (그림 4)에서는 각 데이터베이스별로 각기 다른 형식 모델을 그림으로 나타냈지만, 이상적인 것은 하나의 형식 모델로도 모든 지역 데이터베이스를 변환할 수 있는 모델이다.



(그림 4) 데이터베이스의 통합과정

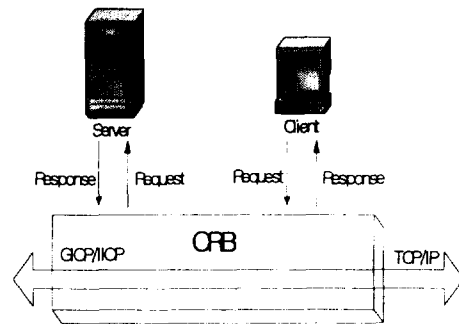
설계 방법과 관련해 분산데이터베이스를 설계할 때 중요하게 생각해야 될 것 중의 하나가 바로 데이터의 분산이다. 분산데이터베이스에서 관리되는 데이터들이 각 노드에 분산되어 관리되는 것이 보통인데, 이 분산된 데이터를 어떻게 설계하여 운영하느냐는 분산데이터베이스 시스템의 성능에 큰 영향을 미치는 요소이다.

데이터가 각 노드에 나누어지는 방법으로 수평 분할(horizontal fragmentation), 수직 분할(vertical fragmentation), 혼합 분할(hybrid fragmentation)이 있다.

## 4.2 CORBA

### 4.2.1 CORBA의 개념

CORBA는 (그림 5)와 같이 클라이언트와 서버가 ORB(Object Request Broker)라는 미들웨어 버스(bus)를 통해 서로 자유롭게 통신할 수 있는 분산객체 미들웨어 모델이다[1][3].

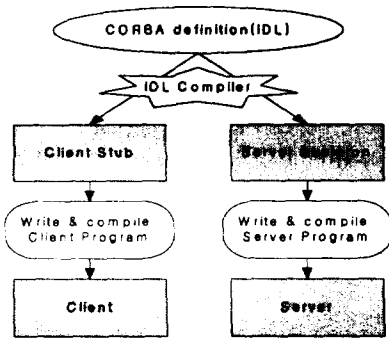


(그림 5) CORBA의 구조

CORBA가 분산객체 업체에서 성공할 수 있었던 것은 특정 프로그래밍언어와 운영체제, 네트워크에 종속적이지 않는 독립적인 인터페이스(IDL, Interface Definition Language) 명세어이고, 개방 구조를 통해 여러 ORB 구현 제품들간의 상호 운영이 가능하기 때문이다. 특히, 특정 프로그래밍 언어에 종속적이지 않아 C/C++, Java, COBOL 등의 여러 언어로 매핑(mapping)하여 개발할 수 있기 때문에 기존 개발자들이 쉽게 제품 개발에 적용할 수 있도록 해 준 것이 성공의 큰 요인이라 볼 수 있다[2][7].

### 4.2.2 CORBA 프로그래밍

일반적인 CORBA 프로그래밍의 과정은 (그림 6)과 같다. 우선 객체에 대한 IDL 스펙을 작성하고, IDL 컴파일러를 이용하여 서버 스펙레톤(skeleton)과 클라이언트 스텝(stub)을 생성한다. 생성된 스펙레톤과 스텝을 이용하여 서버와 클라이언트 프로그램을 코딩한다. 코딩한 프로그램을 컴파일하고, 서버와 클라이언트 프로그램을 실행한다.



(그림 6) CORBA 프로그래밍 과정

## 5. 프로토타입 시스템의 설계 및 구현

### 5.1 시스템의 설계

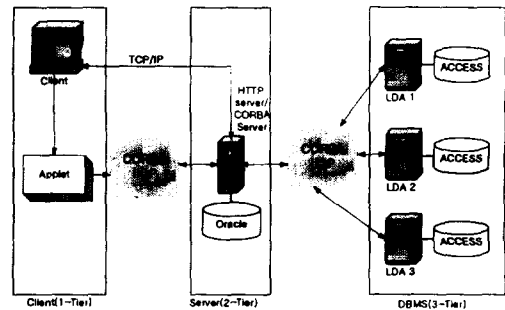
분산데이터베이스 시스템을 구현하는 것은 독립형 데이터베이스 시스템을 구축할 때 해결해야 할 문제점을 더욱 복잡한 형태로 가지고 있기 때문에 많은 시간과 노력이 필요하다. 그렇기 때문에 본 연구에서는 시스템의 프로토타입을 구현하고자 한다. 이미 구축되어 운영되고 있는 데이터베이스를 대상으로, 이를 통합하여 사용자와 관리자에게 편리한 관리시스템을 제공하는 것이 본 프로토타입의 목표이다. 대상 데이터베이스는 현재 전국의 모든 초·중등학교에서 운영되고 있는 학생생활 기록부 자료이고, 이 데이터베이스를 분산객체 기법을 사용하여 사용자들이 하나의 사용자 인터페이스를 통해 여러 학교의 자료를 동시에 접근하여 자료를 검색, 수정할 수 있는 분산데이터베이스 시스템을 구현하고자 한다. 이미 구축되어져 있는 자료를 대상으로 하기 때문에 상향식 접근방법의 설계를 하고, 각 학교별로 분할되어져 있는 자료는 수평 분할되어져 있다. 학생생활 기록부 자료가 동일한 데이터베이스 스키마(schema)를 가지고, 동일한 데이터베이스 엔진(ACCESS 엔진)을 사용하고 있기 때문에 각 학교의 데이터베이스를 통합하기 위해 데이터베이스 변환 과정은 필요하지 않다[6][7]. 다만, 각 학교의 고유 아이디(identity)를 통해 자료에 접근하여 이를 통합하는 과정의 알고리즘만 필요하기 때문에 비교적 쉽게 3-Tier 형식의 분산데

이터베이스 시스템을 구현할 수 있다. 본 시스템의 개발 환경은 <표 1>과 같다.

<표 1> 구현 시스템의 개발 환경

Hardware	CORBA Server	SUN Ultra Sparc
	CORBA Client	Pentium PC
	Local DB	IBM Netfinity
Software	IDL 컴파일러	Java IDL 1.3
	Java 컴파일러	JDK 1.3
	Database	ACCESS Oracle 7
	JDBC	JDBC-ODBC Oracle JDBC
	Tool	JBuilder 3.5

구현 시스템의 전체 구조는 (그림 7)와 같다. 시스템은 사용자의 CORBA 클라이언트 애플릿(applet), 사용자의 요구에 서비스하는 CORBA 서버, CORBA 서버에 데이터베이스 자료를 넘겨주는 지역 데이터베이스 어댑터(Local Database Adapter, LDA)로 구성되어 있다.

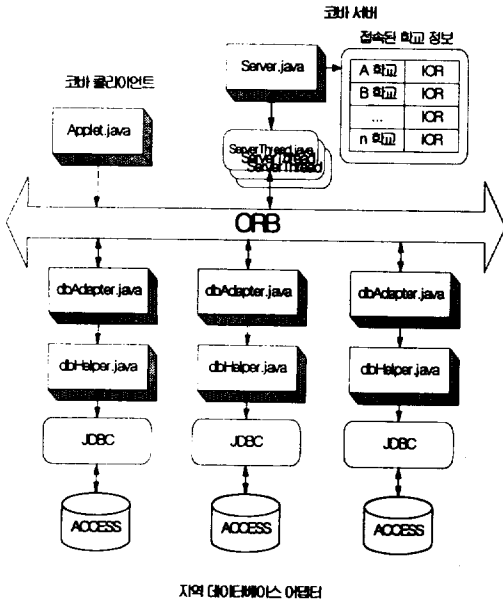


(그림 7) 구현 시스템의 구조

사용자가 웹 서버(CORBA 서버)에 접속하여 CORBA 클라이언트 애플릿을 다운 받아 이 프로그램을 이용하여 CORBA 서버의 서비스를 이용하게 된다. CORBA 서버는 사용자의 요구에 따라 학교를 선택하여, 그 학교에게 다시 서비스를 요청하여 그 답을 받아 사용자에게 돌려주는데 CORBA의 콜백(callback)을 이용하여 구현하였다. 사용자가 사용하는 데이터베이스가 학생의 인적 자료를 대상으로 하기 때문에 보안을 위하여 이 서비스를 이용하는 인가자 등급을 위해 웹 서버에 오라클 데이터베이스를 설치하여 사용자 등급에 따라 서비스의 이용을 제한하였다.

5.2 프로토타입 시스템의 구현

본 구현 시스템의 전체적인 프로그램 모듈은 (그림 8)과 같다.



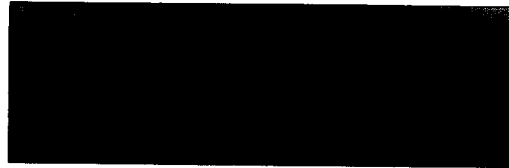
(그림 8) 구현 프로그램의 프로그램 모듈

5.2.1 CORBA 서버 프로그램

CORBA 서버에는 서버에 접속하여 서비스를 이용할 수 있는 사용자의 정보가 담겨져 있는 데이터베이스가 설치되어 운영된다.

서버에 접속한 사용자가 특정 자료를 원하면 서버는 자신이 가지고 있는 LDA 객체의 정보를 이용하여 실시간으로 LDA에 접근하여 자료를 가져와 사용자에게 넘겨주는 역할을 한다. 이 때 서로 다른 LDA에 접근할 경우, 서버 프로그램은 각 LDA에 하나의 새로운 쓰레드(thread)를 생성하여 클라이언트의 요구를 처리한다. 각 LDA가 자신의 데이터베이스에 접근하여 자료를 가져오는 시간이 길기 때문에, 여러 개의 LDA에 접근하여 자료를 가져오는 경우 쓰레드를 생성하여 병렬 처리(parallel processing)하는 것이 전체 시스템의 처리율을 높일 수 있다.

(그림 9)는 CORBA 서버가 실행되어 실시간으로 접속되는 LDA의 상태를 표시해 주는 상태를 나타내주는 그림이다.

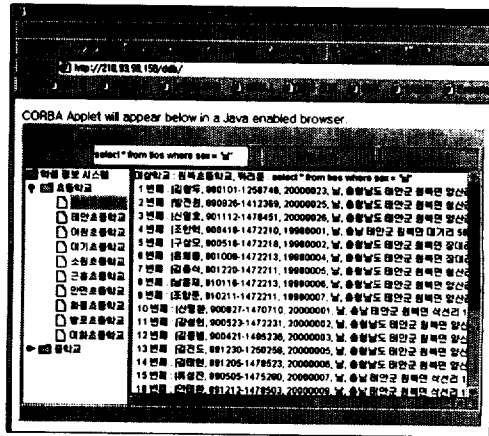


(그림 9) COBRA 서버의 실행 모습

5.2.2 CORBA 클라이언트 프로그램

CORBA 클라이언트 프로그램은 자바의 애플릿으로 구성되어 있다. 자바 애플릿은 웹 환경에서 가장 많이 사용되고 있는 플러그인(plug-in)으로 사용자에게 직관적이고 사용하기 쉬운 인터페이스를 제공해 주고 있기 때문에 클라이언트 프로그램으로 선택했다.

사용자가 웹 서버에 접속하여 클라이언트 애플릿을 호출하면 애플릿은 우선 사용자의 아이디(ID)와 비밀번호를 묻는다. 이 때 서버는 자신이 가지고 있는 사용자 데이터베이스와 비교하여 접속한 사용자에게 맞는 서비스를 제공하는 애플릿 기능을 호출할 수 있게 해 준다.



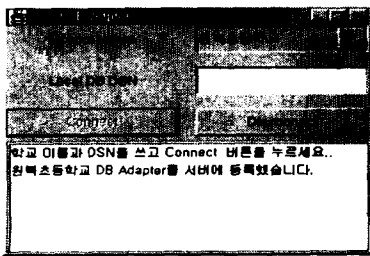
(그림 10) COBRA 클라이언트의 실행 모습

(그림 10)은 관리자로 접속하여 쿼리(query)를 통해 학교의 데이터베이스에 접근하여 학생 자료를 조건을 가지고 검색한 모습이다.

5.2.3 지역 데이터베이스 어댑터 프로그램

지역데이터베이스 어댑터 프로그램(LDA)은 CORBA 서버의 요청을 받아 자신의 데이터베이스에 있는 자료를 넘겨주는 콜백(callback) 객체이다.

네트워크에 LDA가 많이 있기 때문에 각 LDA를 구별할 수 있는 아이디가 필요한데, 사용자는 학교의 이름으로 구별하고 CORBA 서버는 IOR(Interoperable Object Reference)로 LDA를 구별한다. LDA가 CORBA 서버에 접속하면서 자신의 학교 이름과 IOR를 서버에게 넘겨주면 서버는 각 LDA의 정보를 저장하고 있다가 사용자가 자료를 요청하면 이 정보를 이용하여 필요한 자료를 찾게 된다. 그림 11은 LDA의 실행 모습이다.

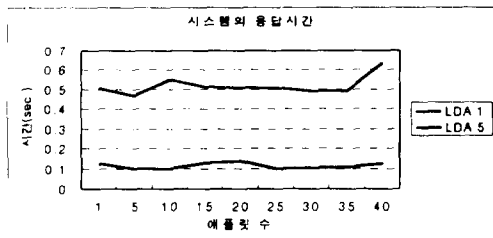


(그림 11) LDA의 실행 모습

### 5.3 구현 시스템의 성능

구현된 시스템의 클라이언트의 응답 시간(response time)성능을 측정해 본 결과 다음과 (그림 12)과 같은 결과를 얻을 수 있었다. 동시에 LDA의 자료를 요구하는 클라이언트를 40개까지 늘여 측정한 결과이다. 단일 네트워크 계층에서 측정하였기 때문에 실제 학교를 대상으로 한다면 다른 측정 결과가 나올 수 있다.

5개의 LDA에게 동시에 자료를 요청하는 병렬 처리 절차를 확인해 본 결과 전체 시스템의 성능을 좌우하는 것은 각 LDA의 자료 취득 성능임을 알게 되었다.

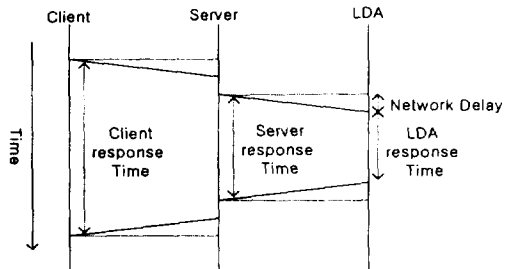


(그림 12) 구현 시스템의 응답 시간

## 6. 결론

분산객체 기술이 현재의 컴퓨팅 환경 개선에 매력적이기는 하지만 아직까지 표준안에 따른 기술이기 때문에 서비스 구현의 어려움, 서버측의 확장 문제, 로드 밸런싱(load balancing)과 고장 감지 문제, 컴퍼넌트화의 미비로 인해 DCOM에 비해 현장에 활용되는 빈도가 적은 것이 사실이다. 하지만, 국내와 외국의 많은 업체들이 꾸준히 기술과 툴(tool)을 개발하여 시간이 갈수록 CORBA를 이용해 어플리케이션을 개발하는 경우가 많아지고 있다[2].

따라서 현재 초·중등 학교에서 운영되고 있는 독립형 데이터베이스 시스템을 통합하는 방법중에서 분산 객체 모델인 CORBA를 이용하여 분산데이터베이스 시스템이 구현한다면 비용과 시간측면에서 적당할 선택이 될 것이다.



(그림 13) 시스템의 모듈별 응답시간 구조

3-Tier 방식의 데이터베이스 어플리케이션 모듈의 응답 시간의 구조는 (그림 13)와 같기 때문에 전체 시스템의 성능 향상을 위해서는 LDA의 응답 시간(response time)을 줄이는 것이 가장 중요하다.

또한 클라이언트의 요구에 대한 처리율을 높이기 위해서는 멀티 쓰레드 ORB 모델을 채택하여 클라이언트의 요구를 병렬 처리할 필요가 있으며 [12][13][15], 한 클라이언트의 요구가 여러 학교의 자료를 동시에 요구할 경우 (그림 8)의 "ServerThread.java" 모듈과 같이 서버 구현 프로그램의 LDA별 쓰레드를 생성하여 서비스하는 데이터베이스 서버쪽 멀티 쓰레드 처리 또한 필요하다.

**참 고 문 헌**

[ 1 ] 김형주(1999). 일주일만에 배우는 CORBA, 마이트Press,  
 [ 2 ] 교육부. <http://www.moe.go.kr/>  
 [ 3 ] 송우일(2000). "CORBA", 마이크로소프트 11월호, 소프트뱅크미디어  
 [ 4 ] 왕창중·이세훈(1999). Inside CORBA3 프로그래밍, 대림  
 [ 5 ] 정의천(1999). CORBA와 관계형DB의 객체지향적통합을 위한 게이트웨어 설계 및 구현, 한국외국어대학교  
 [ 6 ] 조재성(1999). COBRA환경을 기반으로하는분산데이터베이스 통합에 관한 연구, 전북대학교  
 [ 7 ] 최현중(2000). CORBA를 이용한 학교간 분산데이터베이스 시스템 프로토타입 설계, 스쿨넷 2000  
 [ 8 ] 최현중(2000). CORBA를 이용한 학교간 분산데이터베이스 시스템 프로토타입 설계 및 구현, 10·11월호, 소프트뱅크미디어  
 [ 9 ] C.J.Date(1994). An Introduction To Database Systems, Addison-Wesley  
 [10] Daniel Serain(1999). Middleware, Springer  
 [11] David Bell, et al.(1992), Distributed Database Systems, Addison-Wesley  
 [12] Douglas C. Schmidt et al. (1995). Introduction to Distributed Object Computing, SIGS, Vol. 7, No. 1  
 [13] Douglas C. Schmidt. Evaluating Architectures for Multi-threaded CORBA Object Request Broker, ACM.  
 [14] Douglas C. Schmidt et al.(1997). Evaluating CORBA Latency and Scalability Over High-Speed ATM Networks, IEEE ICDCS 97.  
 [15] Douglas C. Schmidt. Evaluating Architectures for Multi-threaded CORBA Object Request Brokers, <http://www.cs.wustl.edu/~schmidt/corba-research-performance.html>.  
 [16] IONA. <http://www.iona.com>.  
 [17] OMG. <http://www.omg.org>.  
 [18] M. Tamer Özsu(1999). Principles of Distributed Database Systems, Prentice-Hall.

**최 현 중**

1993 공주교육대학교 교육학과 (학사)

1999~현재 한국교원대학교 컴퓨터교육과 석사과정

관심분야: 컴퓨터교육, 분산 객체, 분산 DB  
 E-Mail: [blueland@comedu.knue.ac.kr](mailto:blueland@comedu.knue.ac.kr)

**정 상 욱**

1992 대구교육대학교 초등교육과 (교육학학사)

2000 한국교원대학교 컴퓨터교육과 (교육학석사)

2000~현재 한국교원대학교컴퓨터교육과 박사과정  
 관심분야: 컴퓨터교육, 원격교육 등

E-Mail: [ccomo@cc.knue.ac.kr](mailto:ccomo@cc.knue.ac.kr)



**김 태 영**

1985 한양대학교 산업공학과 (공학학사)

1990 Texas A&M University 컴퓨터과학과(공학석사)

1994 Texas A&M University 컴퓨터과학과 (공학박사)

1994.4~1994.8 삼성데이터시스템즈(주) 정보기술연구소 선임연구원.

1994.9~현재 한국교원대학교 컴퓨터교육과 부교수  
 관심분야: 데이터베이스, 컴퓨터교육, 네트워크 등

E-Mail: [tykim@cc.knue.ac.kr](mailto:tykim@cc.knue.ac.kr)

