

# 이기종 DBMS간 데이터 교환과 복제를 위한 XML 웹 에이전트 설계 및 구현

유선영<sup>†</sup>, 이춘근<sup>\*\*</sup>, 임재홍<sup>\*\*\*</sup>

## 요 약

오늘날 지식과 정보 교류의 기반이 인터넷으로 옮겨지면서 기업도 인터넷을 활용한 정보공유와 경제활동에 대한 요구가 증대되고 있다. 표현 위주의 하이퍼텍스트 마크업 언어(HTML : HyperText Markup Language)는 제한된 태그들을 사용하므로 문서를 구조화시키지 못하고 정보를 저장하거나 추출하는 방법이 비효율적이다. 반면에, 확장성 마크업 언어(XML : eXtensible Markup Language)는 사용자가 문서상에 사용될 태그를 정의할 수 있고 구조적인 데이터를 위해 만들어진 언어이므로 HTML에 비해 정보추출이 용이하다. 따라서 기업의 상업적인 데이터가 저장되어 있는 데이터베이스에서 필요한 데이터를 추출하여 웹에서 교환하기 위해서는 HTML보다는 XML이 더 적합하다. 본 논문에서는 이기종 데이터베이스 관리 시스템(DBMS : DataBase Management System)간의 데이터 관리와 XML 문서와 데이터베이스간의 정보 교환에 대하여 연구하고 XML을 매개물로 이용하여 이기종 DBMS간의 정보를 교환할 수 있는 웹 에이전트 시스템을 설계하고 구현하였다.

## Design and Implementation of XML Web Agent for Data Exchange and Replication between Heterogeneous DBMSs

Sun-Young Yu<sup>†</sup>, Chun-Keun Lee<sup>\*\*</sup>, Jae-Hong Yim<sup>\*\*\*</sup>

## ABSTRACT

HTML is unstructured document because of using restricted tag. HTML is difficult to extract data from HTML document. But XML is able to use user definition tag, that is easy to store information. Also XML is easy to extract data from XML document. This is the reason why XML is a standard for data exchange format on the Internet, so XML is fitted to exchange data between heterogeneous DBMSs(DataBase Management System). In this paper, we designed and implemented of XML web agent for data replication between heterogeneous DBMSs. A XML web agent system controls data of DBMS, and generates a XML document from data of DBMS. Also XML web agent is data exchange or replication between heterogeneous DBMS by the medium of XML.

**Key words:** XML, HTML, DBMS(데이터베이스관리시스템)

※ 교신저자(Corresponding Author): 유선영, 주소: 부산시 영도구 동삼동(606-791), 전화: 051)410-4910, E-mail: niccysy@hhu.ac.kr

접수일: 2003년 10월 30일, 완료일: 2003년 12월 31일  
<sup>†</sup>준회원, 한국해양대학교 대학원 전자통신공학과 박사과정

<sup>\*\*</sup>종신회원, 동의대학교 교양과정부  
(E-mail : cklee@hyomin.donguei.ac.kr)

<sup>\*\*\*</sup> 한국해양대학교 전자통신공학과 부교수  
(E-mail : jhyim@hanara.kmaritime.ac.kr)

## 1. 서 론

인터넷 보급이 확산되면서 웹 문서의 표준으로 텍스트 기반의 하이퍼텍스트 생성 언어(HTML : HyperText Markup Language)를 이용하였다. HTML은 단순하고 일반인이 사용하기 편리하다는 장점으로 초기의 인터넷을 전 세계적으로 확산시키는데 큰 역

할을 하였다[1,2].

현재 웹 정보 저장에 가장 많이 사용되고 있는 HTML은 정보를 어떻게 화면에 표시할 것인지를 정의하므로 간편하고 사용하기 쉽다. 그러나, 제한된 태그들로 인하여 구조화되지 못한 문서 형태를 가진 HTML은 정보추적의 한계와 문서를 분석하여 정보를 추출하거나 추출한 정보의 가공이 어렵다[3].

HTML의 단점을 보완하기 위해 월드 와이드 웹 컨소시엄(W3C : World Wide Web Consortium)은 1996년 웹 문서의 표준으로 다양한 기능과 구조적인 표현 능력을 가진 표준 범용 마크업 언어(SGML : Standard Generalized Markup Language)에 기반을 둔 확장성 마크업 언어(XML : eXtensible Markup Language)를 제안하였다[4].

XML은 SGML의 기능성과 구조적인 표현을 지원할 뿐 아니라 사용의 편리성을 강조하였다. 사용자가 문서상에 사용될 태그를 자유롭게 정의하고, 정의된 태그를 자유롭게 사용할 수 있어 문서의 확장이 용이하고 구조적인 표현이 가능하므로 정보추출 및 정보의 가공이 쉽다.

오늘날 지식과 정보 교류의 기반이 웹으로 옮겨지면서 기업들은 인터넷을 활용한 전자거래에 대한 요구가 증대되고 있다. 그러나, 기업이 가지고 있는 대부분의 상업적인 데이터는 관계형 데이터베이스 관리 시스템(RDBMS : Relational Database Management System)에 저장되어 있으므로 서로 다른 데이터베이스를 가지고 있는 기업과 거래하기 위해서는 서로 다른 데이터베이스에 저장되어 있는 정보를 공유하고 교환할 필요가 있다[5,6].

서로 다른 데이터베이스에 저장되어 있는 데이터를 기업이 B2B(Business to Business) 거래에 이용하고자 하는 기업의 요구를 충족시키기 위해서는 HTML보다는 XML이 적합하다. B2B 시대에는 XML 문서의 양도 증가하기 때문에 XML 문서를 저장, 관리할 XML 저장관리 기술과 데이터베이스 관련 아키텍처를 XML 문서로 변환하는 기술이 필요하다. 이러한 데이터베이스 관련 아키텍처를 데이터 형태로 저장하기 위해서 XML 문서의 구조적 정보를 토대로 데이터베이스에 그 형태로 저장하는 방식에 대한 연구가 진행되고 있다[7,8].

본 논문은 이기종 데이터베이스 관리 시스템(DBMS : Database Management System)간의 데이터 교환과 복제를 요구하는 기업의 요구를 충족시키

기 위해 XML을 이용한 웹 에이전트 시스템을 설계하고 구현하였다. 구현된 시스템은 이기종 DBMS간의 데이터 복제를 위한 매개물로 XML을 이용함으로써 복제하고자 하는 데이터를 웹 브라우저를 통하여 관리한다. 이는 데이터베이스에 저장된 데이터를 웹 상에서 관리하므로 데이터베이스를 관리하기 위한 전문적 지식이 없는 일반인도 쉽게 다룰 수 있을 뿐만 아니라 기업들간의 전자거래를 위해 새로운 데이터베이스를 구축하고 데이터를 다시 정리하는 번거로움을 줄여준다는 장점이 있다.

본 논문은 2장에서 XML의 개요와 구현하고자 하는 웹 에이전트 시스템에 대해 설명하고 3장에서는 실제로 구현한 내용을 소개하고 끝으로 4장에서는 결론과 향후 연구방향에 대하여 기술하였다.

## 2. XML 웹 에이전트 시스템 설계

### 2.1 확장성 생성 언어(XML)

1996년 4월 팀 브레이는 구조화된 문서를 위한 XML의 설계목표를 발표하고 1996년 11월 SGML '96 컨퍼런스에서 처음 XML 초안이 발표하였다. 1998년 2월 10일에는 W3C 권고에 따르는 XML 스펙 1.0이 발표되고 오늘날까지 꾸준히 연구되어 오고 있다[9,10].

웹에서 구조화된 문서를 전송 가능하도록 설계된 XML은 표준화된 텍스트 형식의 생성 언어로 SGML의 일부분이며 SGML보다 간결하고 인터넷에서 사용 가능한 문서를 표현하는 표준이다. 또한, XML은 구조화된 문서를 정의하고 자유롭게 태그를 정의할 수 있는 SGML과 인터넷에서 손쉽게 하이퍼미디어 문서를 제공할 수 있는 HTML의 장점을 취합하고 있다. HTML과 SGML의 필수적인 기능만을 취합하고 복잡하고 어렵거나 비효율적인 부분은 제외함으로써, XML은 두 언어의 핵심적인 장점을 가지고 있다.

### 2.2 관련연구

오늘날 대부분의 상업적인 데이터가 저장되어 있는 RDBMS와 기존의 DBMS와 연계하여 다양한 플랫폼과 이질의 데이터에 관계없이 저장과 관리가 용이한 시스템의 구축이 필요성이 대두되고 있다. RDBMS에서 DTD(Document Type Definition) 기

반의 XML 문서 저장 구조를 설계하기 위해서 DTD를 데이터베이스 테이블로 사상시키기 위해 XML 데이터를 저장하는 방법과 관계형 테이블로 사상시키는 방법에 대한 연구가 진행되고 있다[11-13].

객체지향 데이터베이스에서는 질의 결과를 XML 데이터로 변환하는 연구가 진행되고 있다. 객체지향 데이터베이스에 저장된 데이터의 XML로의 변환과 정은 데이터베이스 스키마의 DTD로의 변환과 실제 저장된 데이터의 XML 문서로의 변환 두 부분으로 나눈다[14,15].

### 2.3 기존 제품과 비교

데이터베이스에 저장되어 있는 데이터를 XML을 이용하여 데이터를 교환할 수 있는 대표적인 예는 오라클에서 제시하고 있는 XDK(XML Developer's Kit)와 펜타시스템의 타미노가 있다.

XML 문서를 XML 포맷으로 저장하고 관리하는 타미노는 클라이언트와 서버간의 연결을 위하여 별도의 소프트웨어와 라이선스를 구매해야 하고, 이를 통하여 데이터를 교환해야 하고, 오라클에서 제작한 XDK는 오라클 데이터베이스와 다른 데이터베이스간의 정보 교환만 가능하므로 원본 데이터가 저장되어 있는 데이터베이스는 반드시 오라클 제품이어야 하는 제약이 있다.

본 논문에서 구현한 웹 에이전트 시스템은 추가 소프트웨어가 없이 보편적인 웹 브라우저를 사용하므로 별도의 소프트웨어나 라이선스를 구매하지 않아도 되며 웹 서버가 가지고 있는 JDBC(Java DataBase Connectivity)의 드라이버에 의해 연결 가능한 데이터베이스가 결정된다. 즉, 웹 서버가 JDBC를 이용하여 연결할 수 있는 모든 데이터베이스간의 정보 교환이 가능하다. 현재 많은 데이터베이스 제조 회사들이 제품과 함께 JDBC 드라이버를 같이 제공하는 대표적인 예는 Oracle, MSSQL, mySQL, DB2, FoxPro, AS400 등이다.

### 2.4 시스템 구성

웹 에이전트에서 사용자는 자바를 지원하고, XML 문서를 볼 수 있는 익스플로러 4.0 이상, 넷스케이프 4.0 이상의 브라우저를 사용한다. 웹 서버는 웹 프로그램의 솔루션으로 자바와 연동이 되고 이식성이 좋은 JSP(Java Server Page)를 이용한다.

JSP도 자바와 같이 데이터베이스와 프로그램을 연동시키기 위해 JDBC를 사용한다. JDBC는 자바나 JSP 등으로 작성된 프로그램을 데이터베이스 안의 데이터들과 연결하기 위한 응용프로그램 인터페이스이다. JDBC는 데이터베이스와 자바 프로그램을 연결시켜주는 연결자 역할을 한다. 본 논문의 웹 서버에는 데이터베이스마다 전용 드라이버를 설치하여 속도나 성능을 향상시키고, 환경에 구애받지 않고 폭넓게 사용할 수 있는 순수 자바 드라이버를 이용한다.

본 논문에서 구현한 이기종 DBMS간 데이터 복제를 위한 웹 에이전트 시스템은 기존의 데이터베이스에서 이루어지던 데이터의 저장 및 삭제, 수정 등의 관리를 웹에서 할 수 있게 하였다. XML을 매개물로 이용하여 서로 다른 DBMS간의 데이터 하도록 한 것이 본 논문에서 구현한 웹 에이전트 시스템의 주요한 기능이다.

웹 에이전트의 전체 시스템 구성은 그림 1과 같이 웹에서 데이터베이스를 관리하는 과정과 XML 문서와 데이터베이스간의 정보를 교환하는 과정, 그리고 데이터베이스간의 데이터 복제과정으로 나누어진다.

웹에서 데이터베이스를 관리해야 하는 경우 사용자는 컴퓨터 브라우저를 통해 데이터베이스에 접근할 수 있는 아이디와 패스워드에 관한 정보를 웹 서버로 전송하며 웹 서버는 전송된 정보를 토대로 데이터베이스와 연결을 시도한다. 웹 서버와 데이터베이스의 연결이 이루어지면 사용자로부터 수정하고자 하는 데이터에 대한 추가 입력된 정보를 이용하여 데이터베이스에 저장된 데이터를 추가, 삭제 및 수정한다.

사용자가 새로운 정보를 데이터베이스에 저장하

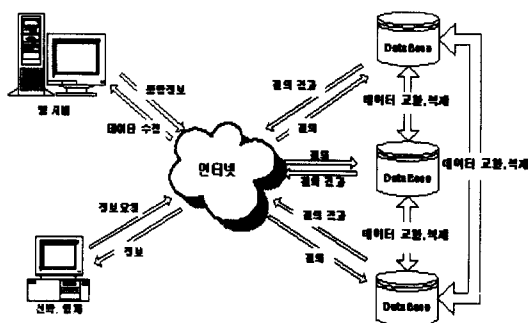


그림 1. 웹 에이전트 시스템의 구성

고자 할 경우, 사용자는 정보를 XML 문서로 저장한 뒤 XML 문서의 위치에 대한 정보를 입력하여 웹 서버로 전송하면 웹 서버는 입력된 정보를 토대로 XML 문서에 접근하여 XML 문서의 내용을 웹 브라우저를 이용하여 보여준다. 사용자는 웹 브라우저를 통해 보여지는 내용을 확인하거나 수정을 한 다음 XML 문서의 내용이 저장될 데이터베이스에 대한 정보를 입력하면 웹 서버는 XML 문서의 내용을 데이터베이스에 저장한다. 서로 다른 데이터베이스간 데이터를 복제할 경우, 사용자가 해당 데이터베이스에 대한 정보를 입력하면 웹 서버는 입력된 정보를 이용하여 해당 데이터베이스와 연결되어 사용자는 복제하고자 하는 테이블을 선택한다. 웹 서버는 선택된 테이블의 내용을 데이터베이스로부터 전달받아 XML 문서로 사용자가 지정한 위치에 저장한다.

본 논문에서 구현한 시스템은 인터넷에 연결되어 있는 컴퓨터에 사용자가 브라우저를 통해 웹 서버에 접속하게 되면 언제 어디서나 데이터베이스를 관리할 수 있고, 웹 환경에 익숙한 사용자라면 누구라도 관리할 수 있다.

### 3. 웹 에이전트 시스템 구현

#### 3.1 데이터베이스 연결

2장에서 언급한 것과 같이, 본 논문에서 구현한 웹 에이전트 시스템은 사용자가 웹 서버에 접속하여 데이터베이스의 종류를 선택하면 해당 드라이버의 이름이 저장된 웹 페이지로 이동하여 데이터베이스의 종류에 상관없이 데이터를 관리할 수 있도록 하였다. 예를 들어, 데이터베이스의 종류를 오라클 데이터베이스로 설정하고, 연결하고자 하는 데이터베이스의 주소와 데이터베이스와 연결할 때 필요로 하는 사용자 아이디와 패스워드를 입력하는 화면은 그림 2와 같다.

사용자가 입력하는 아이디와 패스워드는 연결하고자 하는 데이터베이스에 대한 접근권한을 가지고 있어야 한다. 그림 2와 같이 연결하고자 하는 데이터베이스의 주소는 "203.230.253.8"이고, 연결 포트번호는 "1521", 데이터베이스 이름은 "nice"이다.

그림 2와 같이 데이터베이스에 대한 정보를 사용자가 입력하여 웹 서버로 전송하면, 웹 서버는 입력된 정보를 이용하여 데이터베이스에 연결할 수 있도

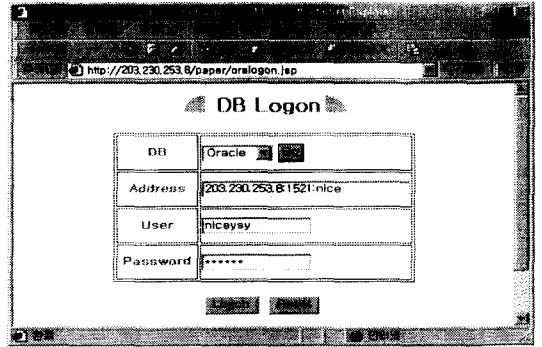


그림 2. DB 로그인 웹 인터페이스

록 질의 문장을 완성한다. 사용자가 선택한 데이터베이스의 종류에 따라서 바뀌는 JDBC 드라이버의 이름은 "Class.forName" 부분을 이용하여 원하는 드라이버를 로딩할 수 있다. 데이터베이스 회사별로 제공하고 있는 드라이버를 사용하기 위해서는 각 데이터베이스 회사로부터 파일을 다운받아 JDK가 설치된 디렉토리 밑에 저장하고 배치 파일을 수정하여야 한다.

본 논문에서 구현한 웹 에이전트 시스템에서는 경로를 설정하기 위해서 배치 파일을 그림 3과 같이 설정하고, 드라이버를 사용하기 위한 파일은 그림 4와 같이 "c:\jdk1.3\jre\lib\ext"에 저장하였다.

데이터베이스의 종류에 따라 드라이버의 이름은 다르게 정의되는데 데이터베이스가 오라클일 때 드라이버 이름은 "oracle.jdbc.driver.OracleDriver"이고, SQL 서버를 사용할 경우는 "com.inet.tds.TdsDriver"이다. 그림 2에서 선택한 데이터베이스가 오라클인 경우 소스는 그림 5와 같다.

그림 5에서 "url1" 변수는 접속하고자 하는 데이터베이스 종류에 따라 주소영역에 붙은 접두어으로써 오라클 데이터베이스인 경우에는 "jdbc:oracle:thin:@"가 되고 "list"라는 변수는 데이터베이스에 저장되어

```
set path=c:\jdk1.3\bin;
set classpath=c:\orant\jdbc\lib\classes12.zip;
c:\jdk1.3\jre\lib\ext\Sprinta2000.jar;
c:\jdk1.3\jre\lib\ext\classes12.zip;
c:\jdk1.3\jre\lib\ext\parser.jar;
c:\jdk1.3\jre\lib\ext\jaxp.jar;
c:\jdk1.3\jre\lib\ext\xerces.jar;
c:\jdk1.3\jre\lib\ext\xercesSamples.jar;
```

그림 3. 자바와 JDBC 드라이버의 경로 설정

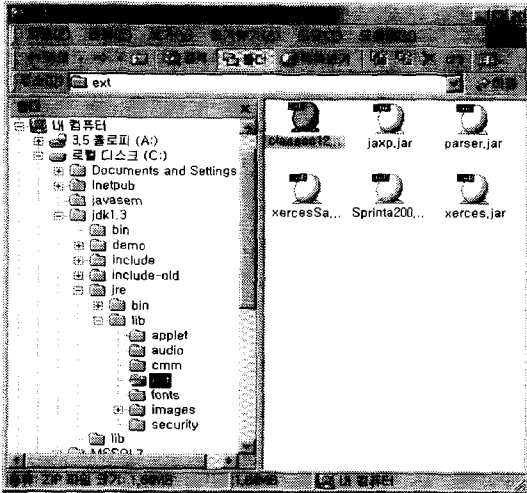


그림 4. JDBC 드라이버 설정 화면

```

Connection conn=null;
Statement stmt=null;
ResultSet rs=null;

try{
    Class.forName("oracle.jdbc.driver.OracleDriver");
    conn
    =java.sql.DriverManager.getConnection(url1+url2, user, pw);
    stmt=conn.createStatement();
    rs=stmt.executeQuery(list);
} catch (java.sql.SQLException e){
    out.println(e);
}
    
```

그림 5. 로그인 프로그램 소스

있는 테이블의 목록을 요청하는 “select \* from tab” 이라는 질의문을 의미한다.

그림 2에서 사용자가 로그인 버튼을 선택하면 텍스트 박스에서 입력된 정보를 이용하여 웹 서버는 그림 5의 try문을 수행하여 웹 서버와 데이터베이스를 연결하고, 데이터베이스에 저장된 테이블 목록을 가져오는 질의문인 변수 “list”를 수행한 결과를 웹 서버에게 전송한다.

데이터베이스와 연결이 되면 웹 서버는 데이터베이스로부터 테이블 목록에 대한 정보를 얻게 된다. 웹 서버는 데이터베이스에 저장되어 있는 테이블 목록에 대한 정보를 테이블의 형태로 웹 브라우저 화면을 통해 사용자에게 보여준다. 실제 오라클 데이터베이스와 연결하여 테이블 목록을 알아본 그림 6의 내용과 웹 서버가 보여주는 화면인 그림 7의 내용이 같다는 것을 확인하였다.

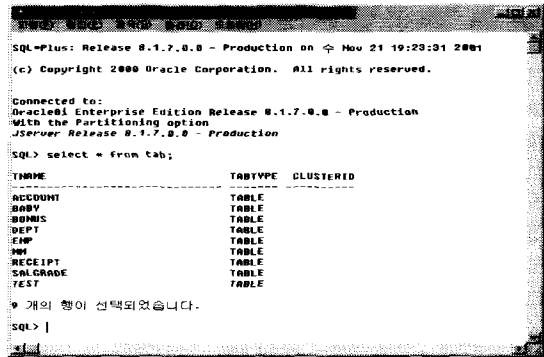


그림 6. 오라클 DB의 테이블 리스트

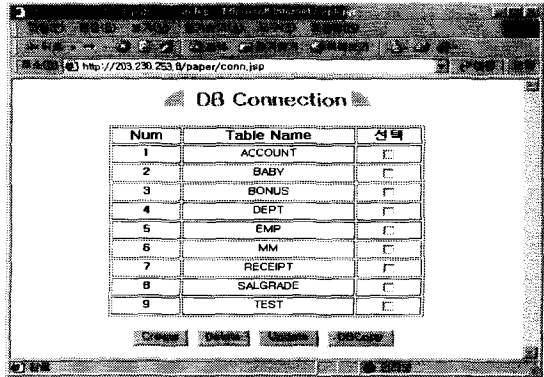


그림 7. DB 연결 후 웹 인터페이스

### 3.2 데이터베이스 관리

그림 7에서 보는 것과 같이 사용자는 테이블의 생성과 삭제 및 수정하는 데이터 관리 기능과 사용자가 선택한 특정 테이블을 다른 데이터베이스로 복제할 수 있다. 그림 7에서 사용자가 선택한 작업에 따라 웹 서버는 사용자에게 작업을 수행하기 위해 필요한 정보를 요구하고, 사용자는 필요한 추가 정보를 입력한다. 웹 서버는 입력된 정보를 이용하여 새로운 질의문을 완성하여 데이터베이스로 전송하고 데이터베이스로부터 전송된 질의 결과 내용을 브라우저를 통해 사용자에게 보여준다. 따라서 사용자는 데이터베이스 관리를 위한 전문적인 지식이 없더라도 웹 서버가 요구한 추가 정보를 입력하면 웹 서버에서 질의문을 생성하여 사용자가 원하는 작업을 할 수 있다.

그림 7에서 테이블을 생성하는 Create 버튼을 선택할 경우 사용자는 생성하고자 하는 테이블 이름, 컬럼 이름과 같은 추가 정보를 입력해야 하고 웹 서

버는 입력된 정보를 이용하여 테이블을 생성하는 질의문을 데이터베이스로 전송하여 연결된 데이터베이스에 테이블을 추가한다. 테이블이 생성된 다음 웹 서버는 사용자로부터 추가하고자 하는 컬럼 수와 컬럼 내용을 입력받아 생성된 테이블에 데이터를 추가시킨다. 그림 7에서 테이블을 삭제하는 Delete 버튼을 선택하면 웹 서버는 사용자가 선택한 테이블을 삭제하는 질의문을 완성하여 데이터베이스로 전송하여 데이터베이스에서 선택된 테이블을 삭제시킨다. 그림 7에서 Update 버튼을 선택할 경우, 웹 서버는 데이터베이스에게 사용자가 선택한 테이블의 내용을 알아내기 위한 질의문을 완성하여 데이터베이스로 전송한다. 데이터베이스가 테이블에 저장되어 있는 컬럼과 레코드의 내용을 웹 서버에게 전달하면, 웹 서버는 이를 테이블의 형태로 바꾸어 사용자에게 보여준다.

예를 들어, "SALGRADE"라는 테이블을 체크하여 Update 버튼을 선택할 경우, 웹 서버는 그림 8과 같이 테이블에 저장되어 있는 데이터의 내용을 보여준다.

그림 8에서 추가, 삭제, 수정 버튼의 기능은 그림 7에 있는 버튼들과 같은 비슷한 기능을 수행하지만, 관리하는 대상이 테이블 자체가 아닌 테이블에 저장되어 있는 컬럼과 레코드의 내용이라는 점에서 차이가 있다.

그림 7과 달리 그림 8에는 테이블의 내용을 XML 문서로 저장하는 XML 버튼이 있다. XML 문서의 루트 태그는 테이블 이름으로 하고, 루트 태그의 자식 엘리먼트로 <record> 태그를 두고 데이터베이스에 저장된 컬럼의 이름은 XML 문서에서 <record>의 자식 엘리먼트인 태그 이름으로 매핑시킨다.

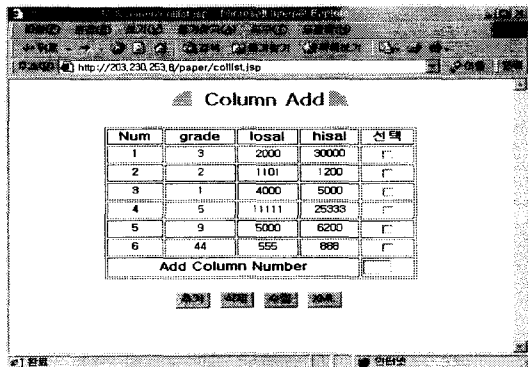


그림 8. 데이터 수정을 위한 웹 인터페이스

본 논문에서 구현한 웹 에이전트 시스템은 루트 디렉토리에 "테이블이름.xml"이라는 이름으로 저장된다. 예를 들어, "SALGRADE"라는 테이블의 내용을 XML 문서로 저장하는 소스는 그림 9와 같고, 웹 브라우저 화면으로 확인해 보면 그림 10과 같다.

그림 9에서 FileOutputStream(nameOfTextfile)은 웹 브라우저가 보여주는 파일 내용을 텍스트문서로 저장하는 기능이고, "pw"는 그림 8에서 보여주는 테이블의 내용을 XML 문서로 저장하는 역할을 한다.

변수 "title"은 XML 문서의 선언 "<?xml version='1.0' encoding='euc-kr'?>"을 표시하고 변수 "root"는 테이블의 이름을 나타내는 것이다. 그림 9에서 두 번째 for문의 내용은 그림 8에서 보여주는 각각의 컬럼인 grade, losal, hisal의 내용을 XML 문서의 <record> 엘리먼트의 자식 엘리먼트로 저장한다.

그림 9와 같은 소스에 의해서 그림 8의 내용을 저장한 XML 문서의 내용은 그림 10과 같다.

### 3.3 데이터베이스 복제

데이터베이스간의 특정 데이터 복제를 위해 DBCopy 버튼을 선택하면 사용자가 선택한 테이블을 복제해서 저장할 목적 데이터베이스에 연결하기

```
try{
    PrintWriter pw
    = new PrintWriter(new FileOutputStream(nameOfTextfile));
    pw.println(title);
    pw.println(root);...
    for(int j=1; j<=num; j++){
        String colnmk = "colnm" + j;
        String colvalue = request.getParameter(colnmk);
        colname[j] = colvalue;
    }
    for(int i=1; i<=n; i++){
        String starttag = "<" + record + ">";
        pw.println(starttag);
        for(int k=1; k<=num; k++){
            String itemik = "item" + i + k;
            String item = request.getParameter(itemik);
            String tag = "<" + colname[k] + ">";
            .....
            String endtag = "</" + colname[k] + ">";
            pw.println(endtag);
        }
        String lasttag = "</" + record + ">";
        pw.println(lasttag);
    }
}
```

그림 9. XML 저장 프로그램 소스

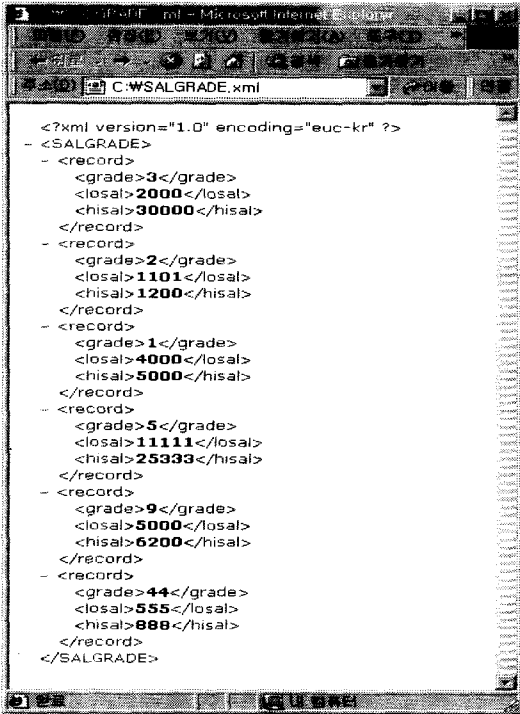


그림 10. SALGRADE.xml 문서

위한 정보와 복제해서 저장하는 새로운 테이블의 이름을 입력할 수 있는 그림 11과 같은 화면이 나타난다. 그림 11은 그림 2와 달리 복제한 테이블의 이름을 디폴트로 지정한 원본 테이블 이름이나 임의로 지정하는 부분이 추가되어 있다.

사용자는 단순히 그림 11과 같은 화면에 정보를 입력하고 DBConn 버튼을 선택하면 목적 데이터베이스에 복제하고자 테이블이 포함되어 있는 테이블 목록을 확인할 수 있다. 이 때, 웹 서버는 사용자가 선택한 테이블로부터 테이블의 컬럼 이름과 컬럼의

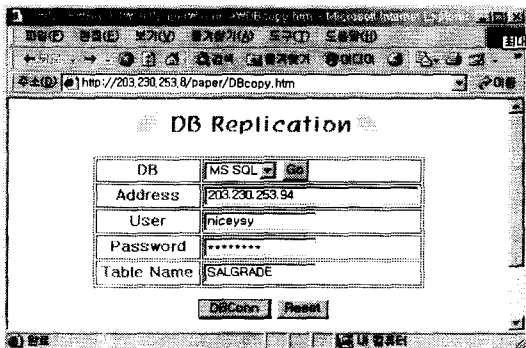


그림 11. 목적 DB 로그인 웹 인터페이스

데이터 타입을 알아내는 질의문을 완성하여 데이터베이스로부터 저장되어 있는 테이블의 정보를 이용하여 복제하고자 하는 테이블에 저장되어 있는 데이터를 XML 문서로 저장하고, 이 XML 문서를 이용하여 목적 데이터베이스에 복제한 테이블을 저장한다.

예를 들어, "203.230.253.8"에 위치한 오라클 데이터베이스의 "SALGRADE" 테이블을 "203.230.253.94"에 위치한 MSSQL에 "SALGRADE"라는 테이블로 복제할 경우 오라클 데이터베이스에 저장된 테이블의 정보를 기억하고 있는 "user\_tab\_column"를 이용하여 테이블 정보를 얻는 소스는 다음 그림 12와 같다.

웹에 존재하고 있는 XML 문서를 사용자는 데이터베이스에 저장하기 위해서 XML 문서에 대한 위치 정보, 데이터베이스에 저장할 때 테이블 이름이 될 루트 태그 이름, 엘리먼트의 이름을 입력한다. 이 같은 수작업은 데이터베이스에 저장하고자 하는 엘리먼트의 수가 많을수록 번거로운 작업이 되지만, 필요한 정보만 입력하여 XML 문서 전체의 저장이나 필요한 부분만 추출하여 저장할 수 있다는 장점을 가지게 된다.

데이터베이스에 저장된 데이터와 웹에 존재하는 XML 문서의 내용을 추가, 삭제, 수정할 수 있다. 또한, 수정 과정을 거치지 않고 XML 문서의 내용을 그대로 데이터베이스에 저장할 수 있다. 이는 웹에 존재하는 XML 형식의 웹 페이지를 데이터베이스에 저장하여 관리함으로써 정보의 재사용과 정보 공유를 더욱 원활하게 하는 것이다.

예를 들어, "C:\paper\SALGRADE.xml"을 열기 위해 그림 13에 필요한 정보를 입력하면 그림 14와

```

String colup
= "select column_name, data_type from user_tab_columns
where table_name = SALGRADE"
try{
  Class.forName("oracle.jdbc.driver.OracleDriver");
  conn6=java.sql.DriverManager.getConnection(url, user, pw);
  stmt6=conn6.createStatement();
  stmt6.executeUpdate(colup);
}
catch(java.sql.SQLException e){
  out.println(e);
}
    
```

그림 12. 테이블 내용 질의 프로그램 소스

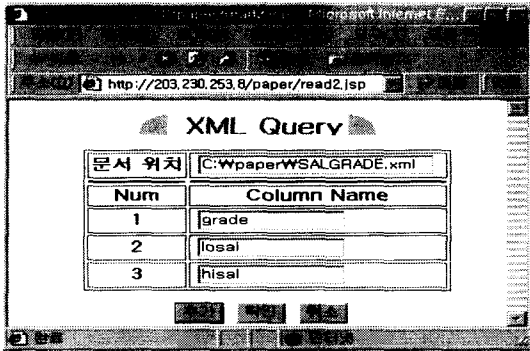


그림 13. XML 파일 Open 웹 인터페이스



그림 14. XML 파일 Update 웹 인터페이스

같은 XML 문서의 내용을 수정할 수 있는 화면에서 사용자가 요구하는 작업을 한 뒤 저장할 수 있다. XML 문서를 저장하기 위한 소스는 그림 9와 같이 사용자가 지정한 위치에 저장할 수 있다.

위에서 언급한 것처럼 본 논문에서 구현한 웹 에이전트 시스템은 별도의 소프트웨어 없이 서로 다른 데이터베이스간에 데이터를 교환하거나 데이터 복제를 할 수 있고 웹에서 정보검색 능력만 가진 초보자라 할지라도 데이터베이스와 정보교환과 데이터 관리를 할 수 있을 만큼 쉽게 구성된 것이 주된 장점이다.

#### 4. 결 론

오늘날 웹이 지식과 정보 교류의 기반이 되면서 기업들은 정보 공유와 웹을 이용한 경제활동에 대한 관심이 증가하였다. 기업들은 데이터베이스에 저장된 상업적인 정보를 공유하고 관리하기 위해 문서에 대한 구조 정보를 내포하고 있는 XML을 정보 저장

소로 이용한다.

XML은 문서에 대한 구조정보를 제공하고, 데이터를 해석할 수 있다. 이러한 XML을 정보 저장소로 이용한다면 웹을 이용한 정보 공유와 정보 검색 및 정보 관리까지 손쉽게 할 수 있다.

본 논문에서는 웹에서 이루어지는 이기종 DBMS 간의 데이터 교환과 복제, XML 문서와 데이터베이스간의 데이터 교환에 대한 연구를 통해 XML을 이용하여 이기종 DBMS간의 정보를 교환할 수 있는 웹 에이전트 시스템을 설계하고 구현하였다.

웹 에이전트 시스템의 주된 기능인 이기종 데이터베이스간의 데이터 복제를 위해 데이터베이스에 저장된 데이터를 XML 문서로 변환한다. 사용자의 입장에서는 복제하고자 하는 데이터베이스의 데이터를 직접 목적지 데이터베이스로 데이터를 보내는 것처럼 보이지만 실제로는 복제할 데이터의 내용을 XML 문서로 변환하고 변환된 XML의 내용을 목적지 데이터베이스에 저장하는 것이다. 이와 같이 웹 에이전트 시스템은 이기종 DBMS간의 데이터 복제와 XML 문서의 내용을 데이터베이스에 저장한다.

또한, 웹 에이전트 시스템은 사용자에게 웹 브라우저를 통해 데이터베이스의 데이터와 XML 문서의 내용을 보여준다. 사용자는 데이터베이스를 관리하기 위한 질의문에 대한 사전지식 없이 단순히 원하는 기능에 해당하는 버튼을 선택하고 필요한 추가 정보를 입력하면 원하는 작업을 수행할 수 있다. 즉, 사용자가 웹 서버에 접속한다면, 언제 어디서든지 데이터베이스 관리가 이루어진다.

향후 연구 방향으로서는 웹에서 이루어지는 정보교환 과정에 보안을 고려하고, XML 질의언어인 XQL을 이용하여 XML 문서에 대한 효율적인 정보검색 기능을 가진 웹 에이전트 시스템에 대하여 연구할 계획이다.

#### 참 고 문 헌

[1] W3C HTML, <http://www.w3.org/MarkUp/#historical>, 2004. 1. 18.  
 [2] 이호섭, 홍충선, "분산환경에서의 CORBA와 XML의 연동구조", 한국정보과학회 봄학술 발표논문집 Vol. 28, No. 1, pp. 424-426, 2001.  
 [3] 이강찬, 손홍, 박기식, "XML 표준화 동향", 정



보과학회지 제 19 권, 제 1 호, pp. 6-14, 2001.

[4] W3C XML 1.1 <http://www.w3.org/TR/xml11/>, 2004. 1. 18.

[5] 이제민, 민경섭, 박상원, 김형주, "관계형 데이터베이스 시스템에서 XML-뷰를 통한 XML 데이터의 지원", 정보과학회논문지: 컴퓨팅의 실제, 제 7 권, 제 3호, 2001. 6.

[6] Daniela Florescu, Donald Kossmann, "Storing and Querying XML Data using an RDBMS", Data Engineering Bulletin, 22(3), September, 1999.

[7] 장우혁, 김홍식, "XML기반의 효율적인 데이터 저장관리를 위한 DB2XML 변환 Wrapper의 설계", 한국정보과학회 학술발표논문집, Vol. 28, No. 1, pp. 106-108, 2001.

[8] 박철현외 정재현, 심대익, 이상구, "구조화된 문서에 대한 DBMS와 IRS의 성능 비교", '99 한국 데이터베이스 학술대회 논문집 15권 1호, pp. 218-225, 1999.

[9] W3C Extensible Markup Language(XML) 1.0, <http://www.w3.org/TR/2000/REC-xml-20001006>, 2004. 1. 18.

[10] Martin Bryan, An Introduction to the Extensible Markup Language, <http://www.personal.u-net.com/~sgml/xmlintro.htm>, 2004. 1. 18.

[11] 박은경, 정채영, 김현주, 배종민, "XML DTD로부터 관계형 테이블로의 사상구조 설계", 한국정보과학회 학술논문집, Vol. 28, No. 1, pp. 133-135, 2001.

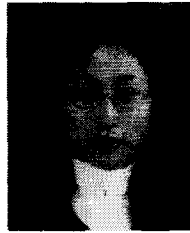
[12] Jayavel Shanmugasundaram, Kristin Tufte, Chun Zhang, Gang He, David J. Dewitt, and Jeffrey F. Naughton, "Relational Databases for Querying XML Documents : Limitations and Opportunities", VLDB, 1999.

[13] 강형일, 최영길, 이종설, 유재수, 조기형, "RDBMS와 IRS를 이용한 XML 저장관리 시스템 설계 및 구현", 정보과학회논문지: 컴퓨팅의 실제, 제 7 권, 제 1 호, pp. 1-10, 2001. 2.

[14] Takeyuki Shimura, Masatoshi Yoshikawa,

and Shunsuke Uemura, "Storage and Retrieval of XML Documents Using Object-Relational Databases", DEXA, 1999.

[15] 윤정희, 박창원, 정진완, "객체 식별자를 이용한 객체지향 데이터베이스의 XML 문서로의 변환", 정보과학회논문지: 데이터베이스, 제 28 권, 제 2 호, pp. 131-139, 2001.



**유 선 영**

1999년 2월 한국해양대학교 전자통신공학과 졸업(공학사)  
 2002년 2월 한국해양대학교 대학원 전자통신공학과(공학석사)  
 2003년 2월~현재 한국해양대학교 대학원 전자통신공학과 박사과정

관심분야: XML, EDI, Semantic Web, Intelligent Agent

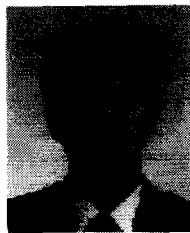


**이 춘 근**

부경대학교 공과대학 건축공학과 학사  
 동의대학교 공과대학 건축공학과 석사  
 동의대학교 공과대학 산업공학과 박사

현, 동의대학교 교양과정부 교수, 멀티미디어 기술사, ISO 9000 국제심사원(IRCA)

관심분야: Multimedia Authoring Tool 분야



**임 재 홍**

1986년 2월 서강대학교 전자공학과 졸업(공학사)  
 1988년 8월 한양대학교 대학원 전자공학과(공학석사)  
 1995년 2월 한양대학교 대학원 전자공학과(공학박사)

1995년 3월~1997년 2월 한국해양대학교 공과대학 전자통신공학과 전임강사  
 1997년 4월~2001년 3월 한국해양대학교 전자통신공학과 조교수  
 2001년 4월~현재 한국해양대학교 전파·정보통신공학부 부교수

관심분야: 분산 네트워크 시스템, 임베디드 시스템