데이터베이스 테이블과 무선 마크업 언어간 변환

조 승호*, 조 범준**
*강남대학교 컴퓨터미디어공학부, **조선대학교 컴퓨터공학부

Transformation between Database Tables and Wireless Markup Language

Seung Ho Cho*, Beom Jun Cho**
*School of Computer Eng. and Media Eng., Kangnam University
**School of Computer Eng., Chosun University
E-mail: shcho@kangnam.ac.kr , bjcho@mail.chosun.ac.kr

요 약

본 연구는 이동 전화나 PDA와 같은 모바일 단말기를 이용하는 무선 인터넷 환경에서 데이터베이스 테이블로부터 Java 객체를 생성하여 이들 중간 형태인 XML 문서로 출력하고, 이러한 XML 문서들을 다시 XML 파서와 XSLT 처리기를 통해 무선 단말기에 적합한 마크업 언어 파일을 생성하여 단말기로 전송한다. 이를 통해 기존의 유선 인터넷상의 HTML 컨텐츠뿐만 아니라 데이터베이스 등 다른 정보들도 실시간으로 접근할 수 있는 장점을 제공한다. 향후에는 편리한 GUI, 매핑 규칙 선택 도구, previewer 제공 등을 위한 연구가 진행되어야 할 것이다.

1. 서론

정보화 사회에서 정보가 차지하는 국가적, 사회적 비중이 증대됨에 따라 특정 시스템과 응용 소프트웨어에 의해 작성된 방대하고 다양한 정보들은 응용 소프트웨어의 버전업 및 이기종의 환경 및 시스템으로 변화될 경우 기존의 정보가 그대로 이용될 수 없어 재구성되거나 변환되어야 할 요구가 증가하고 있다. 이러한 요구에 부응하여 XML(eXtensible Markup Language)이 1996년에 W3C(World Wide Web Consortium)에 의해 제안되었다[13].

SGML은 관련 소프트웨어 개발이 쉽지 않으며, 인터넷상의 서비스를 위한 목적으로 만들어지지 않았기 때문에 웹상의 일반적인 기능을 수용하고 있지 않다는 단점이 있다. SGML의 장점은 문서 내용에 의한 논리 구조 정의, 자유로운 태그 생성을 통해 다양한 응용간에 구조화된 문서 데이터를 상호교환할 수 있어 문서구조를 기반으로 한 검색, 저장 등의 영역에 적용할 수 있다. 이에 반해 XML은 HTML이 인터넷상에서 손쉽게 하이퍼 미디어 문서를 작성할 수 있고 이식성이 뛰어난 점은 받아들여 웹 표준 문서 포맷으로 제안된 이래 빠르게 인터넷에서의 문서 양식의 표준으로 발전하고 있다. 이러한 XML의 장점으로는 우선 응용 프로그램의 개발에 있어 상호 운용성(interoperability) 확보 및 유지 보수의 편리성 등을 꼽을 수 있다[1,10,13,16].

XML은 플랫폼 독립적이기 때문에 새로운 응용 프로그램 개발시 기존 운영 체제나 데이터베이스 시스템을 고려할 필요가 없으며, 데이터 형태에 변화가 발생하는 경우에도 응용 프로그램에 미치는 영향을 최소화할 수 있다. 또한 기존의 데이터 시스템과의 호환성 확보가 용이하고, 웹에서 활용 가능하기 때문에 기존 시스템과 연동이 쉽게 지원된다. 또 다른 장점은 보다 지능적이고 정확한 검색이 가능하다는 점과 네트워크 대역폭의 효율적 활용을 들 수 있다. 웹 상에서 HTML 문서와 비교시 기존의 HTML 문서들은 페이지의 일부에 변경이 일어나면 전체 페이지를 재전송해야 하지만 XML 문서의 경우에는 변경된 일부분만을 재전송할 수 있게 설계되어 네트워크의 부하를 줄일 수 있다.

본 논문은 이러한 XML의 상호운용성을 적용하여 데이터베이스 테이블들을 무선 마크업 언어 문서로 상호 변환에 관하여 연구한 것이다. 예를 들면, 이동

전화나 PDA와 같은 모바일 단말기를 이용하는 무선 인터넷 환경에서 데이터베이스에 대한 조회 결과를 중간 형태인 XML로 변환한 후, 이 XML 파일을 각종 무선 단말기에 적합한 마크업 언어 파일로 생성하여 전송하는 것이다. 이를 통해 기존의 유선 인터넷상의 HTML 컨텐츠뿐만 아니라 데이터베이스 등 다른 정보들도 실시간으로 접근할 수 있게 제공하는 것이다.

본 연구의 2절에서는 본 연구와 관련된 기존의 연구 결과 및 동향에 대하여 기술하고, 3절에서는 본 연구에 의해 개발된 시스템 구성 및 부시스템들에 대하여 기술하며 4절에서는 실제 구현 및 개발환경 등에 대하여 설명하고 5절에서 결론을 맺는다.

2. 관련 연구

2.1 Hibernate/Castor

오픈 소스 진영에서 관계형 데이터베이스, 자바 객체, XML과의 매핑(Object-Relational Mapping, OR Mapping)을 지원하는 API들의 집합으로 컴포넌트로 제공된다. XML 문서를 자바 객체로 생성하거나 자바 객체에 저장된 내용을 XML 문서로 생성할 수 있고, OQL(Object Query Language)를 제공하여, 데이터베이스 테이블을 자바 객체로의 매핑을 제공한다. 이것은 EJB 컨테이너 영속성 제공자(persistence provider) 기능에 의해 지원된다. 따라서, OQL을 사용하여 데이터베이스의 테이블 데이터들을 객체에 저장하고, 이 객체를 XML 형태로 저장하거나, 반대로, XML 파일내 데이터들을 객체에 저장한 후 이를 데이터베이스 테이블로 저장할 수 있다[3,6].

2.2 Allora

이것은 관계형 데이터베이스와 XML 문서간에 데이터를 전달하기 위한 응용에서 사용되는 미들웨어(middleware)로서 세 가지 유형으로 사용된다. 첫번째는 XML 문서와 데이터베이스 간에 데이터를 직접 전달하기 위해 사용될 수 있다. 두번째는 XML 문서와 데이터베이스 간에 SOAP 접속를 통해서 데이터를 전달하기 위해서 사용될 수 있다. 세번째는 XML 데이터 바인딩(data binding)으로 사용되기도 한다. 이 경우, 사용자는 XML 문서나 데이터베이스 테이블상의 데이터에 적합한 클래스를 생성한다. 이러한 클래스들은 응용 프로그램에서 그대로 사용될 수 있다[9].

2.3 XML 통합기(Integrator)

이것은 IBM alphaWorks에 의해 연구된 것으로 관계형 테이블이나 LDAP 데이터와 같은 구조 데이터 형태와 XML 문서 사이에서 양방향 데이터 변환을 수행하는 Java 도구이다[14]. XRTL(XML Relational Transformation Scripting Language)는 XSL을 사용하여 관계형 테이블의 행과 열들을 XML 객체의 요소(element)나 속성(attribute)으로의 매핑을 규정하는 프레임워크를 제공한다. 트리 구조의 트랜잭션 프레임워크 기반 하에 insert, update, delete 질의들을 지원한다.

DTDSA(DTD with Source Annotations)는 DTD(Document Type Definition)의 확장된 표현으로 행과 열이 묶여진 형태의 DTD로 구성되며 이러한 바인딩을 통해 관계형 테이블의 행과 열을 XML 객체의 요소와 속성으로 매핑한다. XML 객체를 관계형 데이터베이스의 테이블에 저장하기 위해서 스크립트에서 지정된 트랜잭션 프레임워크내 insert, update, delete 질의가 지원된다.

3. 시스템 구조

본 절에서는 연구된 시스템의 전체 구조와 부시스템들의 주요 기능들에 대하여 기술한다.

3.1 시스템 구조

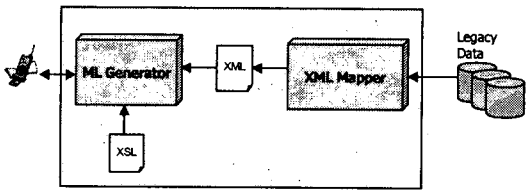


그림 1. 시스템 구조

본 시스템은 크게 2개의 부시스템인 XML Mapper와 ML Generator로 이루어지며, 그림 2는 이러한 시스템 구성을 보여준다[15]. XML Mapper는 데이터베이스 스키마로 표현된 테이블의 행과 열 정보들을 OQL를 사용하여 자바 객체로 변환하고, 이를 marshalling 연산에 따라 XML 문서를 생성한다. 이렇게 생성된 XML 문서는 XSLT 스타일시트와 결합하여 ML 생성기에 의해 무선용 마크업 언어 문서로 변환되어 단말기로 전달된다[4,5,10].

3.2 XML Mapper

이것은 관계형 데이터베이스 테이블들에 대한 질의 결과를 XML 문서로 생성한다.

- DB Wrapper

다양한 DBMS에 공통된 형태로 접근하기 위한 Wrapper이다.

- 질의 처리기(Query Processor)

OQL를 사용하여 데이터베이스에 대한 질의를 수행하며 수행 결과를 돌려받아 처리한다.

- OR 스키마 사상기(Object-Relational Schema Mapper)

관계형 데이터베이스 스키마와 객체(object)를 표현하는 XML Template 파일을 대응시킨다.

- 데이터 바인더(Data Binder)

생성된 객체와 XML과의 (un)marshalling하기 위한 바인더로서 사전에 객체를 생성하는 정적 바인딩과 실행시 객체를 생성하는 동적 바인딩을 지원한다.

- XML 생성기

생성된 객체에 대응하는 XML 문서를 생성한다.

3.3 ML Generator

데이터가 저장되어 있는 XML 문서에 해당 무선 마크업 문서 생성에 적합하도록 고안된 XSL을 적용하여 최종적으로 무선 마크업 문서를 출력한다[1,2].

- XML 파서

생성된 XML의 파싱을 수행한다.

- XSL 변환기(Transformer)

XML 사상기로부터 생성된 XML 문서에 저장소에 저장되어 있는 XSL을 선택하여 모바일 단말기에 적합한 마크업 언어 페이지들을 최종 생성하여 단말기로 전송한다.

3.4 저장소 및 캐쉬 관리자

본 시스템의 융통성 및 성능 향상을 위해서 여러 종류의 캐쉬(Cache)들과 저장소(Repository)들이 존재한다. 캐쉬는 성능 향상을 꾀하기 위해 도입되었으며 저장소는 이미 사용되었거나 사용될 질의어, 규칙, 사상, 틀(Template), XSL 등을 관리하고 필요에 따라 재사용을 하거나 선택적으로 재활용함으로써 반복 작업을 줄임과 함께 확장성을 높여준다. 저장소 관리자들로는 질의 저장소(Query Repository), 사상 저장소(Mapping Repository), 규칙 저장소(Rule Repository) 등이 있으며, 캐쉬 관리자들로는 XML 캐쉬, 페이지 캐쉬(Page Cache)가 있다.

4. 구현

본 절에서는 시스템의 순서도, 사용된 개발 환경, 간단한 구현 예를 보여준다.

4.1 순서도

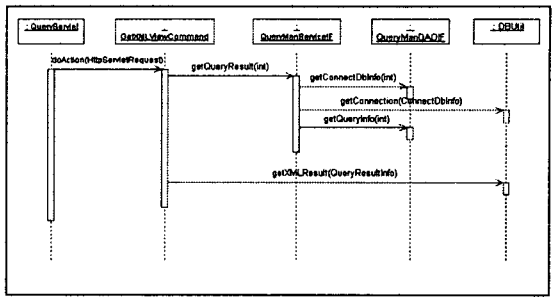


그림 2. 순서도(Sequence Diagram)

그림 2의 순서도는 XML 사상기 부시스템에서 수행되는 주요 메소드들의 순서를 보여준다.

- doAction(HttpServletRequest)

사용자 요청을 받아들인다.

- getQueryResult(int)

사용자 요청을 분석하여 조회할 내용을 추출한다.

- getConnectDbInfo(int)

접속 DB에 관한 정보를 조회한다.

- getConnection(ConnectDbInfo)

조회할 DB와 연결을 설정한다.

- getQueryInfo(int)

사용자 요청에 적합한 SQL 질의에 대해 수행 결과를 조회한다.

- getXMLResult(QueryResultInfo)

DB에 대한 조회 결과를 객체로 대응시킨 다음 XML 문서를 생성한다.

4.2 개발 환경

표 1. 주요 사양

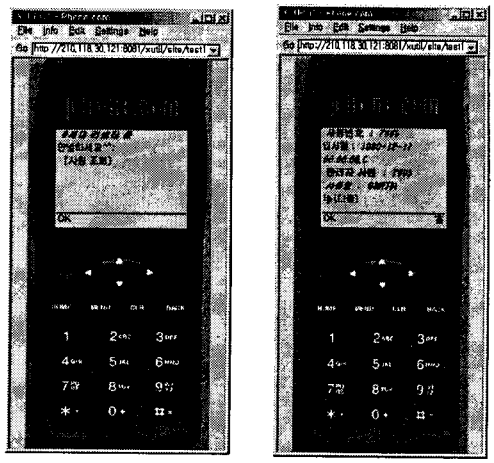
구분	시스템 요구사항
운영체계	Windows-2000/Unix/Linux 등, JRE1.4 필요
데이터베이스	ORACLE, MS-SQL, SYSBASE, INFORMIX, OS2등 JDBC/ODBC 지원 상용 데이터베이스
하드웨어	서버급 PC 이상 (메모리 128M 이상, HDD 100MB 이상의 공간 필요)
개발환경	Java Development Kit 1.4/J2EE 1.3
Library	Xalan, Xerces, JDOM, Hibernate, Castor

본 시스템은 J2EE 환경에서 개발되었으며 표 1과 같이 여러 운영체제상에서 동작할 수 있으나, 실행되

기 위해서는 반드시 JRE 1.4가 요구된다. 아울러 본 시스템은 데이터베이스 테이블들에 대한 XML 문서로의 변환을 위하여 JDOM[8], Hibernate[6], Castor[3]와 같은 컴포넌트들을 활용하였고, XML 문서의 무선 마크업 언어 문서로의 변환을 위하여 DOM 파서인 Xeces[12]와 XSLT 처리기인 Xalan[11]을 활용하였다.

4.3 구현

그림 3은 원격 서버 상의 데이터베이스에 대해 검색을 요청하는 질의를 보낸 후 그에 대한 결과를 화면에 보여준다. 그림 3 (가)의 단말기 화면에서 "사원조회"라는 항목에 대하여 OK 버튼을 누르게 되면 단말기는 "사원 조회"를 위한 요청을 서버로 전송한다. 본 시스템의 처리 결과를 보여주는 그림 3 (나)에서 조회 결과인 "사원번호", "입사일", "관리자 사번", "사용자명" 등으로 이루어진 처리 결과의 일부를 볼 수 있다. 단말기 화면 크기가 제약됨에 따라 다음 버튼을 통해 서버가 처리한 결과들을 계속 볼 수 있다.



(가)요청 화면 (나)결과 화면

그림 3. DB 검색 구현 예

5. 결론 및 향후 연구과제

본 연구는 그동안 많은 시간 소모와 별도의 프로그래밍 작업이 필요했던 기존 DB-to-XML 혹은 XML-to-DB 변환 작업을 쉽고 빠르게 생성할 수 있는 시스템을 제공한다. 이 시스템은 기업이 새로운 관계형 데이터베이스를 채택하거나 기존 데이터베이스를 다른 종류의 데이터베이스로 변경하는 경우에도 쉽게 적용되며, 적용되었던 매핑 규칙들은 프로젝트별로 저장/관리되어 재사용할 수 있기 때문에 실행시간 시 새로 수정되는 데이터베이스에 같은 매핑 규칙을 자동 적용할 수 있어 번거로운 반복 작업을 제거할 수 있다. 또한 데이터베이스 질의를 단순화시키고, 데이터베이스 테이블간의 관계 및 구조를 XML 문서에 간결하게 반영할 수 있는 기능 등을 제공한다.

향후에는 테이블과 XML 문서간의 변환 작업을 시각적으로 수행할 수 있도록 편리한 GUI 제공, 쉽게 검색할 수 있는 매핑 규칙 선택 도구 제공, 최종 변환 결과를 미리 볼 수 있는 viewer 제공, 로그 데이터 처리 등의 기능들을 향상시킬 수 있는 연구가 진행되어야 할 것이다.

[참고문헌]

[1]Richard Anderson, *Professional XML*, WROX Press, 2001.

[2]Neil Bradley, James Jaworski, *The XSL Companion*, SYBEX, 2000.

[3]Castor, http://www.exolab.org /

[4]Nocholas Chase, *XML and Java from Scratch*, QUE, 2001.

[5]B. Frank, *XML Applications*, WROX Press, 1999.

[6]Hibernate, http://hibernate.bluemars.net

[7]JBoss, http://www.jboss.org/

[8]JDOM. http://www.jdom.org /

[9]HiT Software, http://www.hitsw.com / products_services/xmlplatform.html

[10]Hiroshi Maruyama, Kent Tamura, Naohiko Uramoto, *XML and Java*, Addison Wesley, 1999.

[11]Xalan, http://xml.apache.org/

[12]Xeces, http://xml.apache.org/

[13]XML, http://www.w3.org/XML /

[14]XML Integrator, http://www.alphaworks.ibm.com/aw.nsf/techs/XI

[15]조 승호 외 8인, *차세대 지능형 무선 컨텐츠 퍼블리싱 및 자동변환 솔류션*, 우수신기술 최종보고서, 정보통신부, 2002.

[16]"차세대 웹 문서 표준 XML", 한국정보과학회 제6권 제3호 pp.25~35, 1999.