

# XML 데이터베이스 시스템을 기반으로 한 B2B 통합 시스템 개발

이정수<sup>†</sup> · 정상혁<sup>\*\*</sup> · 주경수<sup>\*\*\*</sup>

## 요 약

전자상거래는 여러 가지 종류의 교류와 예측하기 힘든 양의 데이터 변화를 필요로 한다. 따라서 많은 플랫폼과 시스템은 서로 중립적인 데이터 교류에 필요한 표준을 필요로 하는데, 이 요구를 만족시켜 줄 수 있는 기술 중 하나가 바로 XML이며, 이는 W3C에 의해 표준으로 자리잡았다. 또한 XML은 표준화와 운영체제 중립적이라는 특성 때문에 전자상거래에서 중요한 위치를 차지하며, 이미 많은 전자상거래 시스템에서 이용하고 있다. 본 논문에서는, 기업 간 거래에서 사용될 XML 문서를 설계하였고, 이 문서를 주고받아 거래를 완료하는 B2B 통합 시스템을 XML 저장시스템을 기반으로 구현하였다. 또한 기업간의 이질적인 XML 문서의 교환을 위하여 XML 문서의 형식을 변환 시켜 줄 수 있는 XSLT를 사용하였다. 이로써 본 논문에서 구축한 시스템을 이용하여 기업간의 XML 문서 교환을 형식에 제한을 받지 않고 원활하게 할 수 있을 것이다.

## Developing a B2B Integration System based on XML Database System

Lee Young-Soo<sup>†</sup>, Jung Sang-Hyuk<sup>\*\*</sup> and Joo Kyung-Soo<sup>\*\*\*</sup>

## ABSTRACT

E-commerce requires many different types of communications and an unprecedented amount of data changes hands. The many different platforms and systems interacting require a platform neutral standard for data exchange. One of the technologies that can fill this niche is XML, the extensible markup language established as a standard by the W3C. By being standardized and platform neutral, XML to be factor in e-commerce application and using many systems. In this paper, we designed XML document that is used in transaction between corporations, and implement the B2B integration system based on XML database system. Also we use XSLT that make efficient transformation XML documents for exchanging heterogeneous XML data between corporations. So, he or she can more easily and efficiently exchange XML documents between corporations using this system.

**Key words:** XML, RDBMS, XSLT

## 1. 서 론

XML은 1996년 W3C의 XML 워킹 그룹에 의해 제안되었다[2]. XML은 기존의 HTML에서는 불가능했던 사용자 정의형 태그를 사용할 수 있는 것이

특징이며, SGML처럼 구조화된 문서를 효과적으로 작성하고 처리할 수 있는 장점이 있다. 또한, 복잡성 측면에서는 SGML보다 훨씬 쉽고 간편하게 사용할 수 있도록 만들어진 인터넷 표준 마크업이다. 초기 XML은 문서관리 및 정보검색의 관점에서 응용분야가 확대되었다. 그러나, 전자상거래가 활성화되면서 XML은 데이터 중심의 응용분야에도 많이 활용되는 추세이다. XML이 문서 중심에서 데이터 중심으로 그 범위가 확대되는 이유는 XML 장점인 사용자 태

접수일 : 2002년 4월 18일, 완료일 : 2002년 10월 24일

<sup>†</sup> 순천향대학교 일반대학원 전산학과 석사 졸업

<sup>\*\*</sup> 정회원, 순천향대학교 일반대학원 전산학과 석사 과정

<sup>\*\*\*</sup> 순천향대학교 정보기술공학부 교수

그를 정의할 수 능력, 플랫폼이 독립적으로 적용 가능, 강력한 표현 능력으로 인해 전자상거래와 같은 데이터 처리분야에 적극적으로 도입되고 있다. 그러나, XML이 강력한 의미 표현 능력을 가지고 있으며, 사용자가 태그를 정의할 수 있을지라도 태그의 의미에 대한 표준이 정의되지 않으면 서로 다른 조직사이에서 의미전달을 정확하게 할 수 없다[4-7].

특히, 전자상거래 분야에는 기업간 비즈니스에 대한 업무 규칙과 절차가 표준화되어야 한다. 기업이 기업을 상대로 하는 인터넷 비즈니스로는 원자재 및 부품의 상호조달, 운송 및 물류 시스템의 공유 등의 거래가 주류를 이루고 있다. 소비자가 인터넷 쇼핑몰에서 물품을 구입하는 B2C와 달리 B2B는 기업이 거래 주체이며 시장 규모나 성장성이 훨씬 방대하다고 하겠다. 이전에는 EDI 또는 CALS와 같은 형태로 주문, 공급자 관리, 재고관리 등이 이루어져 왔는데 정보 네트워크 기술의 발달과 인터넷 이용자가 증가되고 인터넷 접속 비용이 낮아짐에 따라 규모가 작은 기업들도 B2B 인터넷 비즈니스를 구축하는 것이 가능해져서 B2B는 현재 급속히 확산되고 있는 상황이다[4,15,18]. 그리고 이 형태의 비즈니스는 기업의 마케팅 활동뿐만 아니라 생산 관리, 재고 관리 시스템, 그리고 물류시스템 부분도 지원하는 형태로 나아가고 있다. 이러한 형태의 예로는 미국의 Ford, Cisco, Dell Online사 등이 인터넷을 통해 생산, 제조, 물류 시스템을 지원하고 있다[9]. 이러한 요구를 기반으로 많은 조직과 기업에서는 XML을 기반으로 하는 전자상거래 표준화 작업을 프레임워크 형태로 진행하고 있다.

본 논문에서는 구매자와 판매자간에 물품 구매 시스템을 XML 저장시스템을 이용하여 구현한다[17]. 2장에서는 관련연구로서 매핑기법, XML과 데이터베이스, XML 저장시스템에 대한 정의를 설명하고 3장에서는 XML 저장시스템의 구조를 설명한다. 4장에서는 XML 저장시스템을 응용한 B2B 통합에 대한 물품구매 시스템을 설계하며, 5장에서는 설계에 대한 구현을 다룬다. 마지막으로 6장에서 결론을 기술한다.

## 2. 관련연구

### 2.1 매핑기법

XML 문서와 관계형 데이터베이스 사이에 데이터

를 전송하기 위한 매핑기법이 필요하다. 매핑기법은 테이블-기반 매핑기법과 객체-기반 매핑기법이 있다[8].

테이블-기반 매핑기법은 XML 문서를 테이블로 직접 변환하는 기법으로 단순한 XML 문서일 때 사용한다. XML 문서에서 루트 엘리먼트가 테이블이 되고, XML 문서에 있는 루트 엘리먼트 내에 있는 각각의 엘리먼트들이 테이블의 컬럼이 되고 중복된 엘리먼트 내에 있는 각각의 엘리먼트에 속하는 내용이 테이블의 레코드가 된다고 생각하면 된다. 이런 기법은 단순한 XML 문서에 대하여 하나의 테이블로만 변환이 되므로 이해하기 쉽고 데이터 변환하는 코드가 간단히 이루어지므로 XML 문서와 관계형 데이터베이스 사이에 데이터가 쉽게 변환이 된다. 그러나 복잡한 XML 문서들에서 사용될 수 없으므로 단점으로 드러난다.

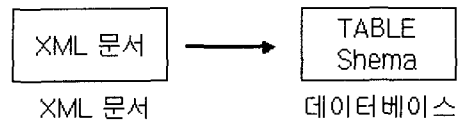


그림 1. 테이블-기반 매핑기법

객체-기반 매핑기법은 복잡한 XML 문서를 변환하기가 불가능한 테이블-기반 매핑기법에 대한 단점을 개선시키기 위한 매핑기법이다. 이 매핑기법은 중간에 객체를 이용하여 테이블 스키마를 추출하는 기법이다.



그림 2. 객체-기반 매핑기법

객체-기반 매핑기법에 대한 기본적인 사항은 다음과 같다. 첫 번째로 XML의 객체 그리고 테이블과 컬럼에 대한 기본적인 매핑기법이다. 이것은 복합 엘리먼트 타입부터 객체로 매핑 한 후, 그 다음 테이블로 매핑하고 단순 엘리먼트 타입으로부터의 속성 매핑을 통한 컬럼으로 매핑 한다. 두 번째로 중첩된 복합 엘리먼트들에 대응되는 테이블을 조인하기 위한 기본키와 외래키를 사용하여 매핑하는 기법이다. 세 번째로는 형제 엘리먼트들의 순서에 대한 정보를 저

장하는 순서 컬럼을 지정하여야 한다.

## 2.2 XML과 데이터베이스

### 2.2.1 XML 데이터 서버 시스템

XML 데이터 서버 시스템은 처음부터 XML 문서를 저장하는 것을 목적으로 설계된 시스템이다. XML 문서의 구조 정보와 내용 정보 등 XML 문서에 대한 특성을 고려하여 그러한 특성을 효율적으로 지원할 수 있도록 설계되어 있다. XML 데이터 서버 시스템의 대부분은 관계형 데이터베이스 시스템이나 객체 지향 데이터베이스 시스템을 사용하여 XML 문서의 구조 정보 및 내용 정보를 저장한다[12].

대표적으로 XML 데이터 서버 시스템으로는 dbXML, eXcelon, Tamino 등이 존재한다. dbXML은 XML 서버인데 XML 문서들에 대한 수집을 관리한다. dbXML은 C++와 같이 자바에서도 구현되어 있고 아파치와 연결되었다. 기본적으로 인-메모리와 플랫폼-파일 그리고 단순한 버킷(bucket) 인덱스 저장에 용이하게 구현되었다. eXcelon은 XML 파서와 마찬가지로 DOM과 XQL 인터페이스를 제공한다. 데이터 저장은 XML 문서, Java 확장, XQL 질의어 등을 보여줄 수 있는 제품으로서 Explorer 인터페이스를 통하여 접근할 수 있다. eXcelon은 ODBC를 통하거나 파일 시스템에서 데이터를 가져올 수 있다. 문서들은 Explorer를 통하여 완전히 검색할 수 있고 문서에 대한 내용도 보여준다[10]. Tamino은 XML 문서를 순수한 표준 XML 포맷으로 저장·검색하는 데이터베이스 서버로 XML 문서 외에 오디오, 비디오 등 현재 웹 환경에서 사용하는 다양한 데이터를 저장할 수 있다. 기존 관계형 데이터베이스 등 분산된 데이터 소스에 대한 통합된 접근 기능을 제공한다[11].

### 2.2.2 XML 문서 서버 시스템[12]

XML 문서 서버 시스템은 기존의 데이터베이스 시스템에 XML 문서와의 데이터 전송을 위한 확장 모듈을 추가한 데이터베이스 시스템을 말한다. XML 문서 서버 시스템은 대부분 데이터 중심의 문서를 저장하도록 설계되어 있다. 이것은 XML 문서를 저장할 수 있도록 특별히 설계된 테이블이 아닌 일반 사용자가 만든 테이블에 저장해야 하기 때문이다. 또한 문서 중심의 문서인 경우는 한 컬럼에 저장하는

경우도 있다.

대표적인 XML 문서 서버 시스템으로는 DB2, Informix, Oracle8i, Microsoft SQL-Server 등이 존재한다. 오라클은 XML에 대한 지원이 강화된 오라클 8i를 발표한 이후에 클러스터링 등의 기능을 보강하여 새롭게 떠오르는 ASP(Application Service Provider)와 B2B 시장을 공략하기 위한 오라클 9i를 발표하였다. 마이크로소프트는 B2B 시장 공략을 위해 트랜잭션 처리 기능과 OLAP 기능 강화에 초점에 맞추고 XML을 통한 타 플랫폼과 애플리케이션의 통합을 주요 기능으로 하는 SQL 서버 2000을 공개하였다. Mecally의 DymaniX은 분산 환경에서 XML 문서를 관리하고, 필터링, 캐쉬, 배치를 하기 위한 서버이다. DymaniX은 문서 채널을 통하여 XML 문서들을 제공하는 발행자/구독자 패러다임을 사용한다. 문서 채널은 XML 문서 캐쉬를 유지하는데 문서 채널에서 발행하는 각 XML 문서들에 대하여 많은 최신 데이터를 보유한다. 발행자는 캐쉬에 있는 데이터 발행을 변경시킬 수 있다. IXIASoft는 XML 서버를 제공하는데 저장소와 같은 인덱스된 XML 문서들에 대한 검색을 지원하고 인덱스 모듈을 형성한다. 각각의 인덱스는 각각 발생하는 XML 문서 위치를 포함하고 있다. Percussion은 XML 내용을 지원하는 도구들을 제공한다. Phythmyx 도구는 데이터 저장 및 검색을 XML로 이용할 수 있고, 사용자가 XML 내용들을 변경하기 위해 XSL를 사용할 수 있다. Phythmyx은 웹 서버, 서블릿, 플러그 인 등을 사용할 수 있다. Sequoia Software의 XPS은 개별화된 웹 기반 인터페이스를 통하여 XML을 이용하여 B2B 서비스를 하는 포털 소프트웨어이다.

### 2.2.3 XML 미들웨어(Middleware) 시스템

XML 미들웨어 시스템은 XML 문서와 데이터베이스간에 데이터를 전송하기 위해 사용하는 프로그램으로 기존에 데이터가 축적된 레거시 데이터베이스를 활용하기 위한 시스템이다[12]. 대부분 데이터베이스 시스템과의 접속은 ODBC, JDBC 또는 OLE DB를 사용하며, 특정 데이터베이스 시스템에 독립적인 구조를 가지고 있다. 대표적으로 XML 미들웨어 시스템으로는 XML-DBMS, XML-DB, XML Junction, XML Shark 등이 있다. XML-DBMS는 XML 문서와 관계형 데이터베이스 사이에 데이터를 전송하기 위한 자바 패키지(package)이다. XML-

DBMS는 문서 안의 문자 데이터와 속성 값뿐만 아니라 XML 문서의 계층 구조 안에서 노드의 순서도 유지한다. 그러나 XML-DBMS는 문서가 아닌 데이터를 전송하기 때문에, 엔티티(entity)와 CDATA 세션(section)과 같은 문서의 물리적 구조나 문서 타입 선언(DTD)에 대한 정보는 관리하지 않는다. XML-DBMS는 XML 문서를 객체의 트리로 간주하며, 이러한 객체와 관계형 데이터베이스를 연결하기 위해서 객체-관계 매핑기법을 사용한다[8].

### 2.3 XML 저장시스템

#### 2.3.1 XML 저장시스템 정의

XML 문서의 생성은 XML 문서 편집기나 각종 응용 프로그램에 의해서 생성될 수 있다. 이렇게 생성된 XML 문서는, 양이 적을 경우 간단한 파일시스템으로 저장 및 검색 등의 운용이 가능하다. 그러나 양이 많아지게 될 경우, 파일 시스템만을 가지고 필요한 문서를 수정, 삭제 및 검색해내는 것이 어렵게 된다. 기존의 저장시스템 중에는 효율적으로 다량의 텍스트 문서의 유지 관리를 위해 개발된 시스템들이 있다. 하지만 이러한 시스템들은 문서의 구조적인 정보를 중요하게 생각하지 않고, 단순히 문서내의 내용만을 유지 관리하기 때문에 XML과 같은 구조화된 문서의 저장 및 관리 시스템으로는 적합하지 않다. 따라서 XML을 위한 전용의 DBMS가 필요하게 된다[17].

#### 2.3.2 XML 저장시스템 기능

XML 저장시스템이 가져야 할 기능은 크게 6가지로 나누어 볼 수 있다. 첫째로 DTD와 XML 문서 Loading 기능이다. 이는 XML 저장시스템이 문서의 생성을 담당하고 있지 않기 때문에 다른 애플리케이션에서 생성된 XML 문서 및 DTD를 XML 저장시스템으로 가져오는 역할을 담당한다. 둘째로 파싱과 유효성 기능이다. 이는 여러 엘리먼트로 구성된 XML 문서를 파싱하고 엘리먼트 단위로 문서를 처리해 주기 위해서 필요한 기능이며, 추가적으로 유효한 문서를 검증하기 위해서 필요하다. 파싱에는 문서의 파싱뿐만 아니라 DTD의 파싱 기능도 있어야 한다. 셋째로 저장 기능이다. 이는 XML 저장시스템의 기본적인 기능으로 검색 요청이나 문서의 변경 요청이 있을 경우, 이를 처리해 주기 위해 저장하는 기능이다. 문

서뿐만 아니라 해당 문서의 DTD를 저장할 수 있는 기능도 지원되어야 한다. 넷째로 검색기능이다. 검색 기능에는 엘리먼트, XML 문서의 내용(Content)부분 검색 기능, 부모, 자식, 형제, 조상, 자손 등의 구조 정보 검색 기능, 속성의 이름 및 값에 대한 검색 기능 등이 제공되어야 한다. 다섯째로 변경 기능이다. 저장된 DTD의 삭제 기능, XML 문서의 삭제 기능, 엘리먼트와 애트리뷰트의 추가, 삭제, 수정 기능 등이 필요하다. 여섯 번째로는 순회기능이다. 검색 기능과의 차이점은 검색기능이 정적인 자료의 추출인데 반해, 순회 기능은 동적으로 자료에 접근하는 기능이다. 순회 기능으로 현재 접근중인 엘리먼트 또는 애트리뷰트에 대한 기준노드 유지기능과, 기준노드의 이름, 값, 타입등의 기준노드 특성 추출 기능, 그리고 기준노드를 DTD에서 문서로, 문서에서 엘리먼트로 또는 이 반대로 바꾸어 주는 노드 레벨에 대한 순회 기능이 필요하다. 이러한 노드 레벨에 대한 순회뿐만 아니라, 엘리먼트의 경우 같은 레벨간에 존재하는 부모, 자식, 형제 노드들간의 순회 기능도 필요하다.

### 3. XML 저장시스템의 구조

XML 저장시스템의 구조는 그림 3과 같이 변환 모듈, 저장 모듈, 검색 모듈로 나누어진다[16]. 변환 모듈은 XML DTD를 입력받아 관계형 데이터베이스 스키마로 변환해 준다. 저장 모듈은 XML 문서를 관계형 데이터베이스에 저장하는 부분이고, 검색 모듈은 관계형 데이터베이스에서 키, 테이블, SQL을 이용하여 검색한 후, 검색한 데이터를 XML 문서화한다. XML 저장시스템 구조는 XMLDBMS에서 제

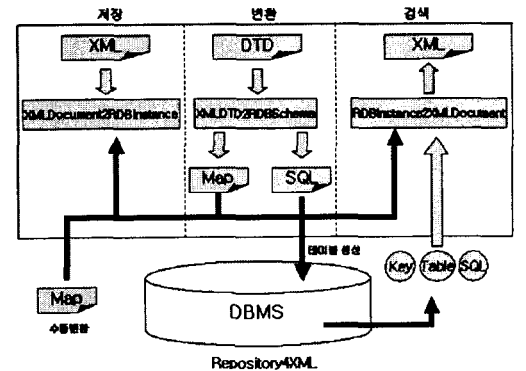


그림 3. XML 저장시스템 구조

공하는 기능들을 EJB 컴포넌트화하고 각각의 EJB 컴포넌트를 조립하여 시스템을 구현하는 방식으로 설계하였다.

### 3.1 변환모듈 컴포넌트

XML 어플리케이션 시스템과 관계형 데이터베이스 시스템 사이에 원활한 연계를 위해서는 XML DTD를 관계형 데이터베이스 스키마로의 변환이 필요하다. 변환 모듈은 자동 변환과 수동 변환으로 이루어지며, 자동 변환은 'XMLDTD2RDBSchema' EJB 컴포넌트에 의해 변환되며, 수동 변환은 사용자가 직접 Map 문서를 작성하여 XML 저장시스템에 적용해야 한다. 'XMLDTD2RDBSchema' 컴포넌트는 EJB 기반의 세션 빈으로 구현된 것으로, XML DTD를 관계형 데이터베이스 스키마로 변환하는 EJB 컴포넌트이다. 그림 4에서는 컴포넌트를 구현하기 위해 정적으로 표현한 클래스 다이어그램이다.

XMLDTD2RDBSchemaBean은 엔터프라이즈 빈 클래스로서 실제적인 비즈니스 로직을 기술한 클래스이고, XMLDTD2RDBSchemaHome은 Home 인터페이스로 XMLDTD2RDBSchema 객체를 생성하는 역할을 한다. 또한 XMLDTD2RDBSchema은 Remote 인터페이스로, 비즈니스 메소드들을 가지고 있다. 그림 5는 그림 4에 제시한 클래스들에 대하여 순차적으로 보여주는 순차 다이어그램이다.

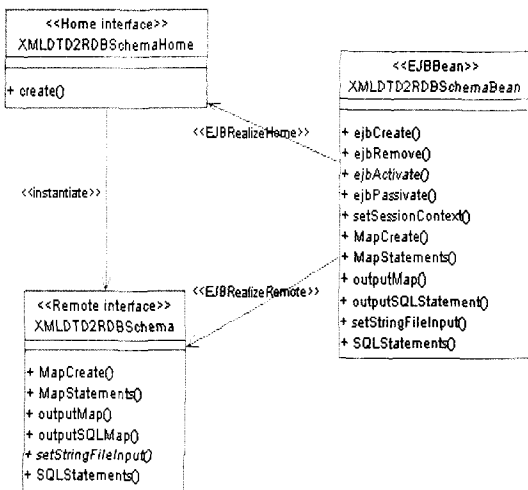


그림 4. XMLDTD2RDBSchema 컴포넌트 클래스 다이어그램

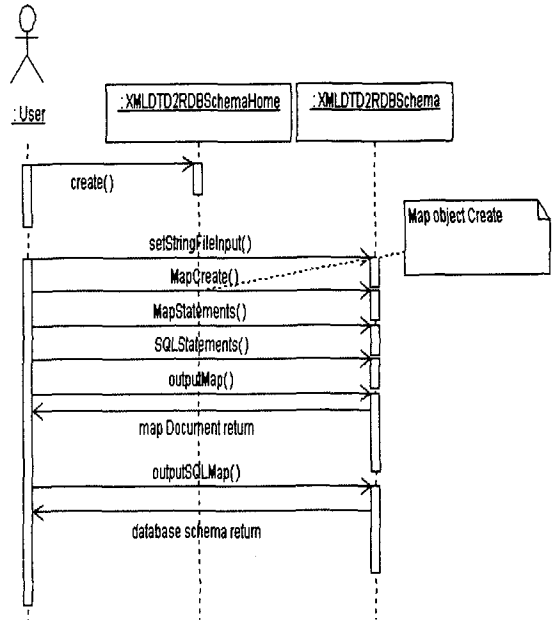


그림 5. XMLDTD2RDBSchema 컴포넌트 순차 다이어그램

Home 인터페이스에서는 상태 없는 세션 빈이기 때문에 create() 메소드만 제공되고, Remote 인터페이스에서는 XML DTD를 입력하는 setStringFileInput() 메소드와 Map 객체를 생성하는 mapCreate() 메소드, 생성된 Map객체를 이용하여 map 문서로 나열해주는 mapStatements() 메소드, 또는 Map 객체를 이용하여 sql 문서로 나열해 주는 SQLStatements() 메소드, 출력을 담당해주는 outputMap() 메소드와 outputSQLStatements() 메소드를 정의한다.

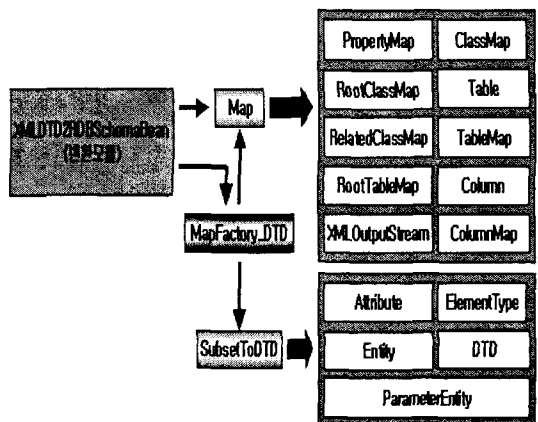


그림 6. XMLDTD2RDBSchema 컴포넌트의 자바 빈

XMLDTD2RDBSchema' 컴포넌트는 EJB 구성 요소인 홈 인터페이스와 리모트 인터페이스, 실제 구현된 빈 클래스 이외에 또 다른 자바 클래스들이 존재한다. 주요 핵심적인 자바 클래스로서 Map 클래스와 MapFactory\_DTD 클래스, 그리고 SubsetToDTD 클래스이다. Map 클래스는 Map 문서와 관계형 데이터베이스간의 매핑을 도와주는 클래스이고, 변환 모듈, 저장 모듈, 검색 모듈에 사용하는 중요한 클래스이다. MapFactory\_MapDTD 클래스는 DTD 객체에서 기술한 엘리먼트 타입과 어트리뷰트들을 관계형 데이터베이스의 테이블과 컬럼에 저장할 수 있도록 도와주는 클래스이고 SubsetToDTD 클래스는 DTD를 총괄적으로 관리 및 제어할 수 있는 클래스이다. 그림 6에서 오른쪽 상단에 있는 클래스들을 관계형 데이터베이스 쪽을 담당하는 클래스들이고 오른쪽 하단에 있는 클래스들을 DTD 쪽을 담당하는 클래스이다.

3.2 저장모듈 컴포넌트

유효한 XML 문서와 Map 문서를 입력받고, 이를 Map 문서 형태에 맞게 XML 문서의 내용(Content)을 관계형 데이터베이스에 저장한다. 저장 모듈에서 주의할 점은 XML 문서의 데이터 중복인데, 이는 데이터베이스에 존재하는 데이터와 입력받은 XML 문서가 동일한 내용일 경우 중복이 발생하므로 이를 해결하여야 한다.

본 저장 모듈은 'XMLDocument2RDBInstance' EJB 컴포넌트로 구현되었으며, XML 문서와 Map 문서를 입력받는다. 'XMLDocument2RDBInstance' 컴포넌트는 EJB 기반의 한 세션 빈으로 구현된 것으로, XML 문서를 관계형 데이터베이스에 저장하는 EJB 컴포넌트이다. XML 문서를 관계형 데이터베이스에 저장하기 위해서는 Map 문서가 추가된다. Map 문서는 XML 문서와 관계형 데이터베이스간에 데이터를 원활하게 전송할 수 있도록 도와주는 문서이다. 그림 7에서는 컴포넌트를 구현하기 위해 정적으로 표현한 클래스 다이어그램이다.

XMLDocument2RDBInstanceBean은 엔터프라이즈 빈 클래스로서 실제적인 비즈니스 로직을 기술한 클래스이고, XMLDocument2RDBInstanceHome은 Home 인터페이스로 XMLDocument2RDBInstance 객체를 생성하는 역할을 한다. 또한 XMLDocument

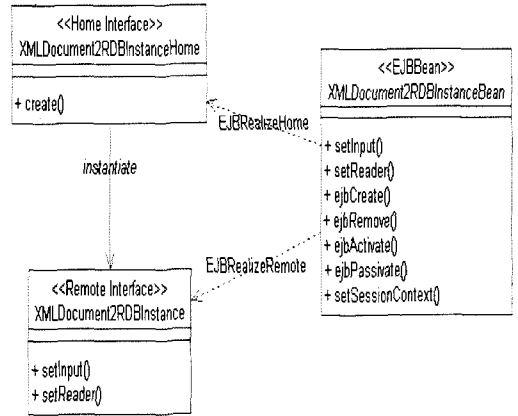


그림 7. XMLDocument2RDBInstance 컴포넌트 클래스 다이어그램

2RDBInstance은 Remote 인터페이스로, 비즈니스 메소드들을 가지고 있다. XMLDocument2RDBInstance 인터페이스를 통해 XML 문서와 Map 문서가 입력한다. 그림 8은 그림 7에 제시한 클래스들에 대하여 순차적으로 보여주는 순차 다이어그램이다.

Home 인터페이스에서는 상태 없는 세션 빈이기 때문에 create() 메소드만 제공되고, Remote 인터페이스에서는 외부에서 읽은 XML 문서와 Map 문서를 String값으로 받아 임시 파일을 생성하는 setInput() 메소드, 데이터베이스 접속 정보들을 설정하는 setReader() 메소드를 정의한다.

'XMLDocument2RDBInstance' 컴포넌트는 EJB 구성 요소인 홈 인터페이스과 리모트 인터페이스, 실제 구현된 빈 클래스 이외에 또 다른 자바 클래스들

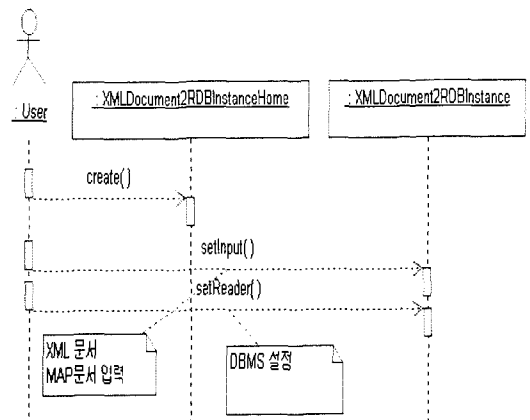


그림 8. XMLDocument2RDBInstance 컴포넌트 순차 다이어그램

이 존재한다. 그림 9에서 제시한 클래스들 중에서 핵심이 되는 클래스가 DOMToDBMS 클래스이다. 이 클래스는 XML 문서를 파싱하여 DOM 객체로 변환한 후 DOM 객체를 관계형 데이터베이스에 데이터를 전송할 때 사용되는 클래스이다. Map 클래스는 그림 6에서 제시한 것과 동일하고 MapFactory\_MapDocument 클래스는 Map 문서가 매핑이 잘 되어있는지를 점검해 주는 클래스이다. PropertyMap 클래스는 엘리먼트 타입과 프로퍼티의 관계를 표시해 주는 클래스이다.

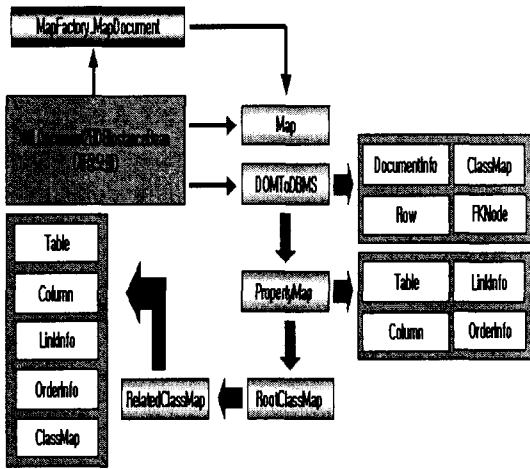


그림 9. XMLDocument2RDBInstance 컴포넌트의 자바빈

### 3.3 검색모듈 컴포넌트

‘RDBInstance2XMLDocument’ 컴포넌트는 관계형 데이터베이스를 검색하여 XML 문서 생성하는 EJB 컴포넌트이다. 이 EJB 컴포넌트를 세 가지 기능으로 분류한다면 첫째는 키를 이용하여 검색하는 방법, 둘째는 테이블을 이용하여 검색하는 방법, 셋째는 사용자는 SELECT문을 작성하여 검색하는 방법이 있다. 그림 10은 컴포넌트를 구현하기 위해 정적으로 표현한 클래스 다이어그램이다.

RDBInstance2XMLDocumentBean은 엔터프라이즈 빈 클래스로서 실제적인 비즈니스 로직을 기술한 클래스이고, RDBInstance2XMLDocumentHome은 Home 인터페이스로 RDBInstance2XMLDocument 객체를 생성하는 역할을 한다.

또한 RDBInstance2XMLDocument은 Remote 인

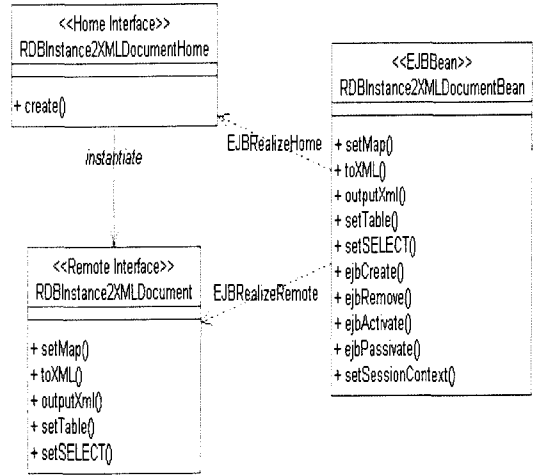


그림 10. RDBInstance2XMLDocument 컴포넌트 클래스 다이어그램

터페이스로, 비즈니스 메소드들을 가지고 있다. RDBInstance2XMLDocument 인터페이스를 통해 Map 문서와 각 기능에 맞는 검색 키들을 입력한다. 그림 11은 그림 10에 제시한 클래스들에 대하여 순차적으로 보여주는 순차 다이어그램이다.

Home 인터페이스에서는 상태 없는 세션 빈이기 때문에 create() 메소드만 제공되고, Remote 인터페

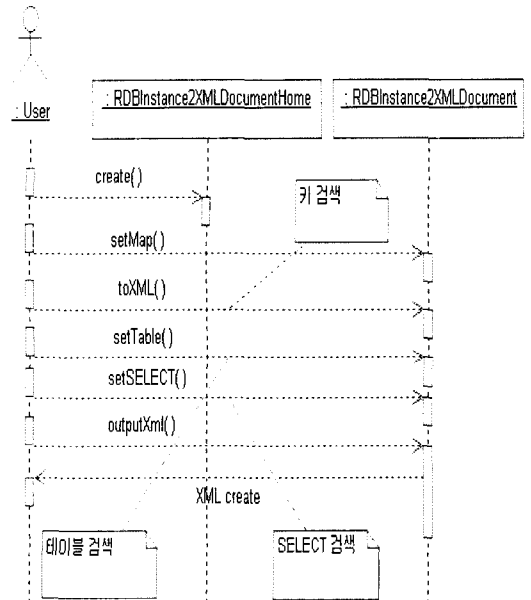


그림 11. RDBInstance2XMLDocument 컴포넌트 순차 다이어그램

이스에서는 외부에서 읽은 XML 문서와 Map 문서를 String값으로 받아 임시 파일을 생성하는 setMap() 메소드, 루트 테이블에 있는 데이터를 이용하여 전체 테이블들을 검색한 후 Map 문서에 따라 XML 문서를 생성해 주는 toXML() 메소드, 하나의 테이블에 있는 데이터를 모두 검색한 후, Map 문서에 따라 XML 문서를 생성해 주는 setTable() 메소드, 직접 사용자가 SELECT문을 이용하여 데이터를 검색한 후 Map 문서에 따라 XML 문서를 생성해 주는 setSELECT() 메소드, 생성된 XML 문서 출력을 담당 해 주는 outputXML() 메소드를 정의한다.

'RDBInstance2XMLDocument' 컴포넌트는 EJB 구성 요소인 홈 인터페이스와 리모트 인터페이스, 실제 구현된 빈 클래스 이외에 또 다른 자바 클래스들이 존재한다. 그림 12는 저장모듈인 그림 9와 동일하나, 이 그림에서 중요한 클래스는 DBMSToDOM 클래스이다. 이 클래스는 관계형 데이터베이스로부터 데이터를 검색하여 DOM으로 데이터를 전송하는 클래스이다. DocumentInfo 클래스는 문서 검색에 필요한 정보들을 가지고 있는 클래스이고, Row 클래스는 관계형 데이터베이스 테이블에서 하나의 행을 검색하여 데이터를 일시적으로 저장하는 클래스이다. 즉 임시보관 클래스라고 말할 수 있다.

4. 설 계

4.1 시스템 구조

실제로 기업 사이의 거래는 주고받는 메시지를 통

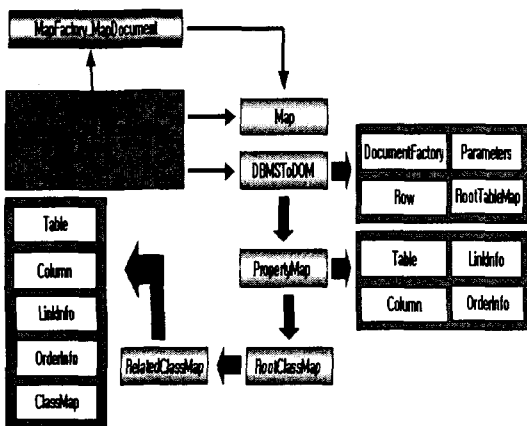


그림 12. RDBInstance2XMLDocument 컴포넌트의 자바 빈

해 이루어진다. 메시지는 카탈로그, 견적서, 주문요청서, 송장, 영수증 등이 각 기업별로 개별화되어 다양하게 존재한다. 무엇보다도 기업간 거래는 쇼핑몰처럼 소액이 아니라 건당 거래 규모가 매우 크고, 기업의 입장에서 기밀이 유지되어야 하는 메시지가 많다. 따라서 기업간 거래의 활성화는 기업이 요구하는 거래 무결성, 연결 안전성, 인증, 자료 저장 등을 만족하여야 한다. 대표적인 예로서, 구매자와 판매자간의 B2B 통합이다. 본 논문에서는 구매자와 판매자간의 B2B 통합을 위해 XML 저장시스템을 이용하여 전달된 메시지를 저장 및 검색하는데 목적이 있다[17,18]. XML 저장시스템은 이미 구현된 EJB 컴포넌트들을 조립한 XML 응용 시스템이다.

4.1.1 구매자시스템

구매자시스템은 구매요청서를 만들어 판매자시스템에서 보낸다. 보내는 내용은 구매요청서와 구매자시스템 URL 주소이다. 구매자시스템 URL 주소는 판매자시스템에서 송장을 받을 URL 주소이다. 또한 구매자시스템은 판매자에서 보내온 송장을 받아 구매자시스템 데이터베이스에 XML 저장시스템을 이용하여 저장하고, XSL을 적용하여 웹 브라우저에 출력해준다. 구매자시스템 데이터베이스에 저장방식은 XML을 하나의 객체로 저장하는 방법이 아니고, 기존 데이터베이스 시스템과 통합하기 위해 XML 문서를 여러 개의 테이블로 분할하여 XML 문서 내용을 저장하는 방법이다. 구매자시스템 구조는 그림 13과 같다.

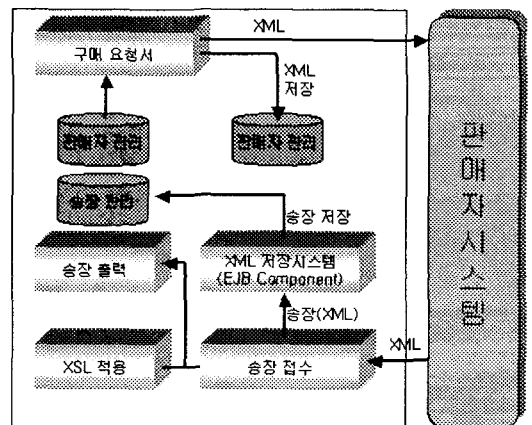


그림 13. 구매자시스템



#### 4.1.2 판매자시스템

판매자시스템은 구매자시스템에서 보내 온 구매요청서를 접수하고 구매요청서와 구매자시스템에 대한 URL 주소를 판매자시스템 데이터베이스에 저장한다. 그 다음으로는 구매요청서는 주문번호가 없기 때문에 판매자시스템에서 다시 XML 문서에 주문번호를 추가하여 XML 저장시스템을 이용하여 데이터베이스에 저장한다. 또한, 판매자시스템에서 납품처리 및 송장 발송을 처리하기 위해 구매요청서 주문번호를 검색하여 해당 주문번호에 대한 상세 정보를 XML로 변환하여 구매자시스템에게 보낸다. 판매자시스템의 구조는 그림 14와 같다.

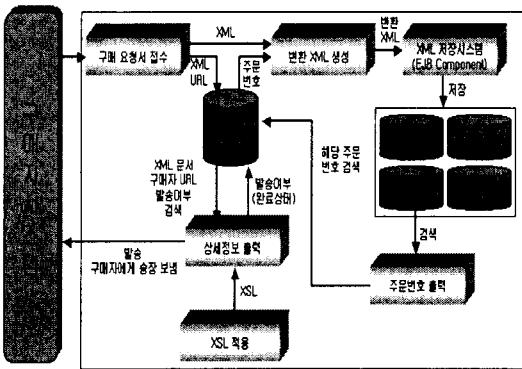


그림 14. 판매자시스템

#### 4.2 XML DTD 설계

구매자시스템에서 판매자시스템으로 구매요청서를 보내야 한다. 구매요청서는 구매자 정보, 판매자 정보, 구매(주문)내역 등이 있다. 구매자 정보과 판매자 정보는 구매자와 판매자라는 역할만이 다를 뿐 포함될 내용인 회사명, 주소 등은 동일하고, 구매내역은 구매물품, 구매물품명, 수량, 가격이 포함한다. 구매자시스템에서 보내온 구매요청서에서는 주문번호 없기 때문에 판매자시스템에서 주문번호를 추가하는 구매요청서로 변환하여야 한다. 그림 15는 주문번호가 없는 구매요청서 XML DTD와 주문번호가 있는 구매요청서 XML DTD를 제시한다.

#### 4.3 데이터베이스 설계

XML 문서의 구조를 정의한 DTD를 객체로 변환한 후, 변환된 객체를 다시 관계형 데이터베이스 스

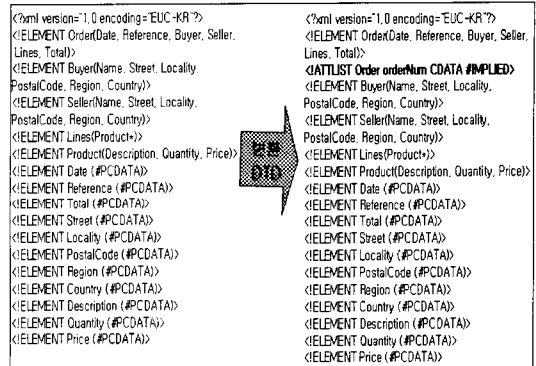


그림 15. 구매요청서 DTD

키마로 변환한다[6,13-16].

#### 4.3.1 XML DTD를 객체로 변환

XML에서 핵심인 엘리먼트와 애트리뷰트가 어떻게 객체로 변환하는지를 설명한다. 엘리먼트는 XML 문서의 기본이 되는 논리적 단위이므로 XML 문서의 모든 내용물은 반드시 요소 안에 포함되어야 하고, HTML과 마찬가지로 시작 태그(begin-tag)부터 그에 상응하는 끝 태그(end-tag)까지의 영역으로 되어 있다. 영역을 가지고 있는 엘리먼트들은 객체로 변환할 때 클래스 이름으로 변환이 되고, 영역을 가지고 있지 않은 엘리먼트들은 클래스 속성으로 변환된다. 즉, PCDATA형으로 선언된 엘리먼트들이 클래스 속성으로 변환된다. 또한 서브 엘리먼트는 별도의 클래스 이름과 그 서브 엘리먼트 영역에 속한 엘리먼트들은 클래스 속성으로 변환한다. 엘리먼트와 서브 엘리먼트의 관계는 클래스의 참조형으로 지정된다.

애트리뷰트는 엘리먼트와 관련된 이름과 값의 쌍으로 이루어지고, 애트리뷰트 값은 엘리먼트의 시작 태그에서 결정될 수 있으며 디폴트값은 DTD로부터 상속받을 수 있다. 애트리뷰트는 엘리먼트와 마찬가지로 단일 값을 가진 애트리뷰트와 다중 값을 가진 애트리뷰트로 분류한다. 단일 값을 가진 애트리뷰트는 문자열(CDATA), 토큰 애트리뷰트(ID, IDREF, NMTOKEN, ENTITY, NOTATION), 열거형 애트리뷰트로 선언되며, 객체로 변환될 때 클래스 속성으로 변환된다. 다중 값을 가진 애트리뷰트들은 IDREFS, NMTOKENS, ENTITIES으로 선언되어 있어, 애트리뷰트 값이 다중으로 존재하므로 객체로 만들 때 다중 값을 가지고 있는 엘리먼트는 클래스

이름으로 변환하고, 다중 값을 가진 애트리뷰트들은 클래스 속성으로 변환된다. 그림 16은 DTD를 객체로 변환한 클래스 다이어그램이다.

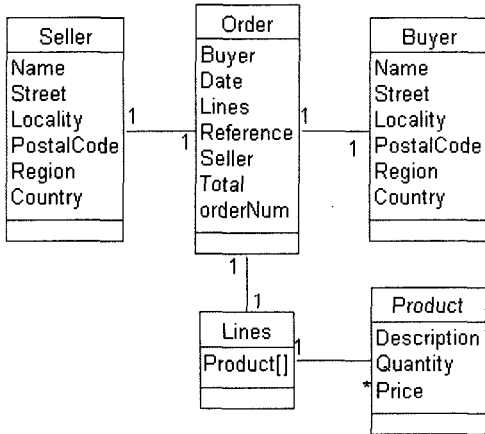


그림 16. 클래스 다이어그램

4.3.2 객체를 관계형 데이터베이스로 변환

엘리먼트들을 객체로 변환하고, 변환된 객체를 관계형 데이터베이스 스키마로 변환한다. 일반적으로 클래스 이름은 테이블 이름으로 변환이 되고 클래스의 속성은 테이블 컬럼으로 변환할 수 있다. 일반적으로 객체 지향 언어에서 변수의 데이터 유형은 정수형, 실수형, 문자형, 논리형 등과 같은 데이터 유형들이 있으며, 관계형 데이터베이스에서 데이터 유형은 데이터를 저장할 컬럼을 그 데이터 유형에 맞게 설정해 주어야 하는데, 객체 지향 언어와 관계형 데이터베이스간의 데이터 유형이 불일치하게 이루어지게 된다. 따라서 본 논문에서는 객체 지향 언어와 관계형 데이터베이스간에 데이터 유형은 사용자가 직접 설정을 해 주어야 한다.

관계형 데이터베이스 스키마로 변환할 때 제약 조건을 설정해 주어야 하는데, 컬럼에 값이 반드시 입력되어야 할 때 NOT NULL이라고 지정하여야 하고, 그렇지 못할 때는 NULL로 지정하여야 한다. 제약조건을 지정할 때는 DTD에 선언된 엘리먼트 연산자를 보고 그 엘리먼트가 아무런 연산자(+,\*)가 없으면 NOT NULL로 지정해 주고 연산자 기호가 있으면 NULL로 지정해 주면 된다.

DTD에서 엘리먼트에 속하는 서브 엘리먼트가 존재할 경우 부모 클래스와 하나 이상의 자식 클래스가

추출된다. 이렇게 추출된 부모 클래스와 자식 클래스간의 관계는 참조형을 이용하여 부모 클래스와 자식 클래스를 연결할 수 있다. 부모 클래스와 자식 클래스간의 관계를 관계형 데이터베이스에서는 기본키와 외래키를 이용하여 관계를 맺는다. 그림 17은 그림 16에 있는 클래스 다이어그램을 이용하여 관계형 데이터베이스 테이블로 변환한다.

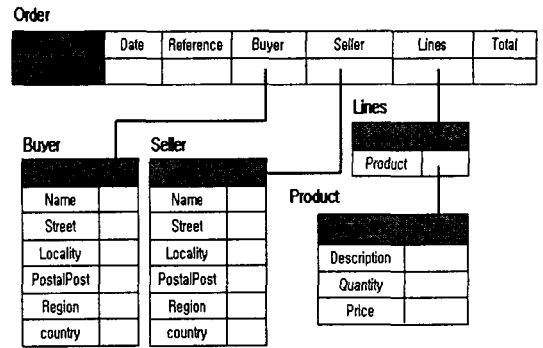


그림 17. 판매자 데이터베이스

4.4 유스케이스 다이어그램

본 논문에서 구현하고자 하는 시스템에 대한 유스케이스 다이어그램을 보여준다. 구매자는 구매요청서를 만들어 판매자에게 보내고, 판매자는 구매요청을 처리한 후 송장을 만들고 구매자에게 보낸다. 그림 18에서는 구매자가 구매요청서를 바로 보내거나

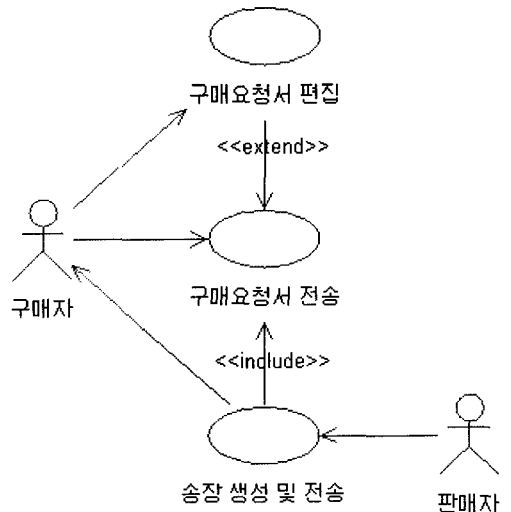


그림 18. 유스케이스 다이어그램

편집하여 보낼 수도 있기 때문에 <<extend>>를 사용해 선택 관계임을 표시하고, 송장은 구매요청서가 있어야만 보낼 수 있기 때문에 <<include>>로 필수 관계임을 표시한다.

#### 4.5 순차 다이어그램

##### 4.5.1 구매자요청서 전송

구매자는 구매자 데이터베이스를 이용하여 구매 요청서를 만들고, 만든 구매요청서를 판매자의 구매 요청서 접수 시스템에 전송한다. 판매자는 구매요청서를 받아 XML 저장시스템을 통해 판매자 데이터베이스에 저장한다.

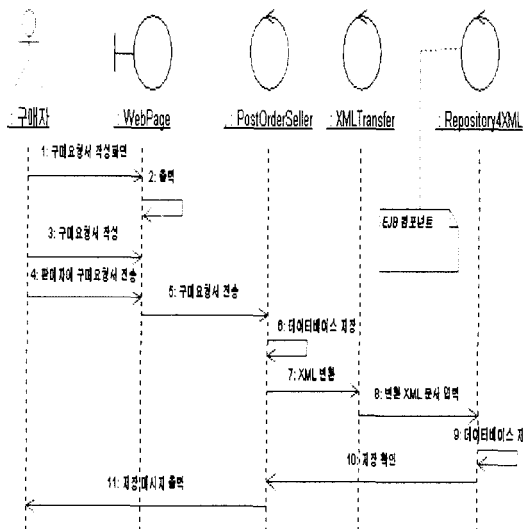


그림 19. 구매 요청서 처리 과정

##### 4.5.2 송장 발송

판매자는 저장된 구매 요청서를 검색하여 관리 화면에 리스트로 보여준다. 판매자는 주문번호를 선택하여 해당 구매자의 상세 정보들을 검색한 후 송장을 작성하고, 작성된 송장은 해당 구매자에게 발송한다. 구매자는 송장을 받아 XML 저장시스템을 통해 구매자 데이터베이스에 저장하고, XSL 적용하여 웹 브라우저로 송장을 보여준다.

### 5. 구 현

#### 5.1 환경

실행환경은 운영체제인 Windows 2000 Server에

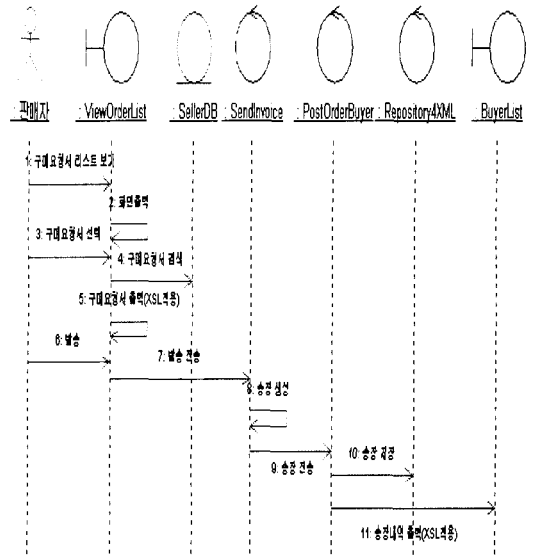


그림 20. 송장 발송 처리 과정

서 구매자 시스템을 실행하였다. 구매자 시스템, 즉 구매요청서를 작성하는 시스템은 자바의 스윙기법을 사용하여 사용자인터페이스를 구성하였고, 구매 요청서를 받는 판매자 시스템은 서블릿을 이용하여 구매요청서를 받는다. 판매자 시스템은 아파치 서버와 서블릿 엔진인 톰캣을 사용하였다. XML 저장시스템에서 사용한 EJB 컴포넌트들은 EJB 서버인 IAS(Inprise Application Server) 4.1 환경하에서 실행하고 DBMS는 오라클 8i를 사용한다.

#### 5.2 구매요청서 사용자 인터페이스

구매자는 구매요청서를 XML 형식으로 작성한다. XML 내용에는 구매자 정보와 판매자정보, 그리고 주문내역들과 구매자시스템 URL을 작성한다. 구매자시스템 URL은 송장을 받기 위해 구매자시스템 위치를 작성하여야 한다.

그림 21은 구매자가 구매요청서를 작성하는 사용자 인터페이스이다. 분류에서는 구매자와 판매자를 구분한다. 고객부분에서는 고객정보를 검색하여 고객명만 화면에 출력해준다. 구매자/판매자입력은 분류에 따라 고객이 구매자인지 판매자인지를 결정하여 화면에 출력해준다. 출력된 결과는 그림 21에서 구매자 정보, 판매자 정보에 나타내고 있다. 주문입력부분은 구매자가 주문항목을 입력하여 주문내역에 포함시킨다, 마지막으로 구매자가 구매요청서를

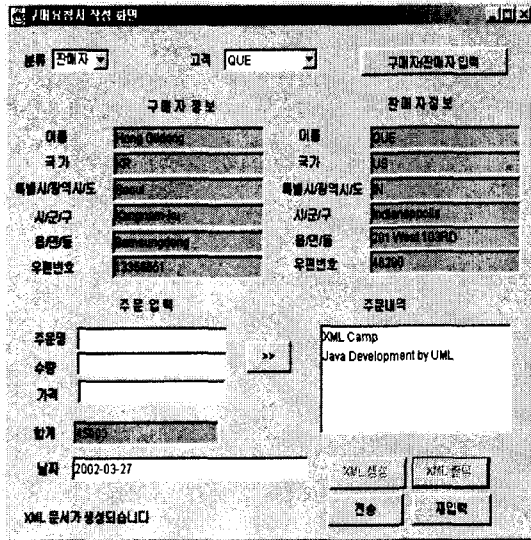


그림 21. 구매요청서 사용자인터페이스

모두 작성하면 XML 문서화로 생성되어 판매자시스템으로 전송한다.

### 5.3 송장 발송 사용자 인터페이스

구매자가 구매요청서를 판매자에게 요청하면 판매자는 구매요청서를 처리하여 구매자에게 송장을 보내준다. 전송은 XML 형태로 전송한다.

판매자는 구매요청서를 받은 날짜를 검색하고 그 날짜에 해당한 구매요청서 주문번호가 리스트로 출력된다. 출력된 구매요청서의 주문번호를 선택하면 해당한 주문번호, 구매자이름, 주문일자, 주문내역 등을 판매자에게 보여준다. 마지막으로 판매자가 납품처리를 한 후 전송을 하며 해당 구매자에게 송장이 XML 형태로 전송되며, 전송된 송장은 구매자 측에서 받아 자사 데이터베이스에 저장한다.

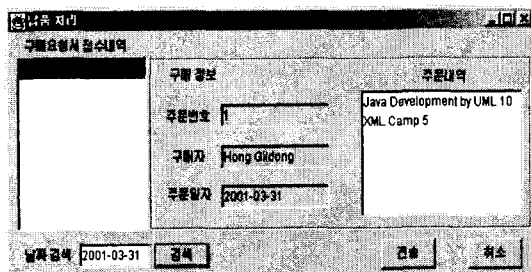


그림 22. 납품처리 사용자인터페이스

## 6. 결론

XML은 웹 상에서 데이터 교환을 위해 제안된 표준언어이다. 웹에서의 자유로운 데이터 교환을 위해, XML은 DTD를 통하여 문서 자체에 문서의 구조를 기술하고 있다. 이와 같이 문서의 구조를 사용자가 원하는 대로 정의할 수 있는 구조적 유동성은 모든 형태의 데이터가 XML로 기술될 수 있도록 해준다. 이것은 웹에서 운용되는 모든 데이터가 동일한 형태로 통합, 저장, 처리될 수 있는 기반을 제공한다[18].

본 논문에서는 B2B 거래의 가장 대표적인 비즈니스 프로세스인 물품 구매 과정에서 XML이 어떻게 사용되는지를 설명하기 위하여 기업 간 거래에서 사용될 XML 문서를 설계하고, 이 문서를 주고받아 거래를 완료하는 시스템을 구현하였다. 또한, 서로 다른 개별 기업의 문서 양식을 지원하기 위하여 기업 간 문서 교환을 할 수 있는 XML과 스타일시트인 XSL를 이용하여 구현하였다. 실제 B2B 거래를 완전히 처리하기 위해서는 본 논문에서 생략한 협상, 배송, 결제 등과 같은 또 다른 기업과의 문서 교환을 추가로 개발해야 한다. 이러한 작업이 완성된다면 이것이 바로 XML 기반의 B2B 솔루션의 예가 된다.

향후 기업간 전자상거래는 모든 기업에 있어서 피할 수 없는 현실이며, 이에 효과적인 대응을 위해서는 표준의 진행상황을 주시해야만 한다. 또한 대부분의 전자상거래 프레임워크에서 공통적으로 적용되는 XML 기술, 레지스트리 관련기술, 메시지/시스템 보안 기술, 기업 내부의 정보시스템과 전자상거래 시스템을 통합하는 EAI(Enterprise Application Integration) 기술분야에 많은 투자와 노력을 기울여야 효율적 B2B 통합을 위한 필수적 요소인 XML 기반의 어플리케이션을 개발할 수 있을 것이다.

## 참고 문헌

[1] Carey, M., Florescu, D., Ives, Z., Lu, Y., Shanmugasundaram, J., Shekita, E., Subramanian, S.,: XPERANTO: Publishing Object Relational Data as XML. In Suciu, D., Vossen(eds.), G., Proceedings of the Third International Workshop on the Web and Databases, WebDB 2000, Dallas, Texas, USA,

- May 18-19, pp. 105-110, 2000.
- [2] "Extensible Markup Language(XML)", <http://www.w3.org/TR/PR-XML-971208>
- [3] Florescu, D., Kossmann, D.: Storing and Querying XML Data using an RDBMS. *Data Engineering* 22:3(1999), pp. 27-34.
- [4] Joo Kyung soo, "A Design of Middleware Components for the Connection between XML and RDB", 2001 IEEE International Symposium on Industrial Electronics Proceedings, Pusan, Korea, June 2001.
- [5] Joo Kyung Soo, "Constraint-Preserved Managing XML documents in Object-relational database", International Conference on EALPIIT 2002, 2002.
- [6] Lee, D., Chu, W. W.: Constraints-Preserving Transformation from XML DTD to Relational Schema. In Laender, A., Liddle, S., Storey (eds.), V., *Conceptual Modeling*, Salt Lake City, Utah, USA, October, 9-12, 2000, LNCS 1920, Springer-Verlag, Berlin, pp. 323-338, 2000.
- [7] Lee Sang Tae, Joo Kyung Soo, "Transforming XML DTD to SQL Schema based on JDBC", International Conference on EALPIIT 2002, 2002.
- [8] Ronald Bourret, XML-DBMS, <http://www.rpbouret.com/xmldbms/index.htm>.
- [9] David.Linthicum, *B2B Application Integration*, Wesley, 2000.
- [10] eXcelon, <http://www.datec.co.kr/>.
- [11] Tamino, <http://www.tamino.co.kr/taminoserver/jsp/index.jsp>.
- [12] GRAVES, *DESIGNING XML Databases*, PH PTR, 2001.
- [13] 이상태, 이정수, 주경수, "객체모델을 기반으로 한 XML DTD의 RDB 스키마로의 변환 방법", 대한전자공학회, 하계종합논문대회, 제24권 제1호, pp. 113-116, 2001.
- [14] 이상태, 이정수, 주경수, "XML DTD를 기반으로 한, RDB 스키마 설계를 위한 Component 구현", 한국정보처리학회 지식 및 데이터공학연구회 제8회 학술발표대회 논문집, pp. 309-316, 2001.
- [15] 이정수, 방승윤, 주경수, "전자상거래 응용을 위한 XML DTD의 RDB 스키마로의 변환 방법", 순천향 산업기술연구소 논문집 2001, 제 7권 제 1호, pp. 5-12, 2001.
- [16] 이정수, 방승윤, 주경수, "XML DTD의 관계형 데이터베이스 스키마로의 자동변환을 위한 컴포넌트 모델링", 한국인터넷정보학회논문지, 제2권 제5호, 2001년 12월, pp. 81-91, 2001.
- [17] 정상혁, 이정수, 주경수, "EJB 컴포넌트 기반의 XML 저장관리시스템 설계 및 구현", 한국정보처리학회 춘계학술발표논문집, 제9권 제1호, pp. 15-18, 2002.
- [18] 김채미, 최학열, 김심석 공저, 전문가와 함께 가는 XML Camp, 마이트 Press, 2001.
- [19] 정강용, 허영남, "인터넷 마켓의 B2C에서 B2B로의 변화 동향 분석", 한국인터넷정보학회, 인터넷정보학회지, 2000년 6월, 제1권 1호, pp. 44-50, 2000.
- [20] 조현성, 박찬규, 송병열, 오수영, 김록원, 김경일, 조현규, 함호성, "XML 기반 전자상거래 프레임워크 기술", 한국정보과학회, 정보과학회지, 2001년 1월, 제19권 1호, pp 38-50, 2001.



### 이 정 수

2000년 청운대학교 컴퓨터과학  
학과 학사 졸업  
2002년 순천향대학교 일반대학  
원 전산학과 석사 졸업

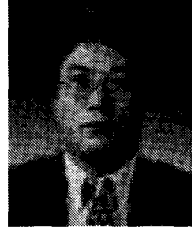
관심분야 : Database Systems, EJB, CBD  
E-mail : jungsoo6@hitel.net



정 상 혁

2002년 순천향대학교 컴퓨터공  
학과 학사 졸업  
2002년 ~ 현재 순천향대학교 일반  
대학원 전산학과 석사과  
정 재학

관심분야 : Database Systems, 분산처리 시스템, B2Bi  
E-mail : grotest@hanmail.net



주 경 수

1980년 고려대학교 이과대학 수  
학과 학사 졸업  
1985년 고려대학교 일반대학원  
전산학과 석사 졸업  
1993년 고려대학교 일반대학원  
전산학과 박사 졸업  
1986년 ~ 현재 순천향대학교 정보

기술공학부 교수

관심분야 : Database Systems, System Integration,  
Object-oriented Systems.  
E-mail : gsoojoo@sch.ac.kr

교신저자

이 정 수 (336-745) 충남 아산시 신창면 읍내리 646 순천향  
대학교 멀티미디어관 M602