

---

# 전자상거래를 위한 XML 데이터 처리 시스템 설계 및 구현

김철원\*

Implementation and Design of XML Data Processing System for EC

Cheol-Ueon Kim\*

## 요 약

현재 XML 응용 분야 가운데 XML 표준 데이터를 이용하여 정보의 교환 및 변환을 하여 시스템의 이식성과 확장성을 증대시키고자 하는 연구가 활발하게 이루어지고 있다. 본 논문은 클라이언트와 서버사이에서 비호환성과 확장에 따른 개발비용 문제를 해결하기 위해 클라이언트와 서버사이에 XML 중간계층을 이용한 시스템 설계를 제시한다. 이 방법은 클라이언트와 서버사이의 프로그램 개발시 중간 조정 역할을 하며, 이질적인 환경에서의 이식성과 시스템의 확장성을 증대시킨다. 본 논문은 XML 기반 DB를 이용하여 전자상거래를 위한 XML 데이터 처리 시스템을 설계 및 구현하였다. 기대효과로는 XML 기반의 전자상거래 구조를 활용함으로써 시스템 확장성과 시스템의 유지보수를 위한 비용절감이 기대된다.

## ABSTRACT

Many issues about portability and extension of system converting and exchanging information using XML standard data are being studied currently. This paper proposes design using XML processing layer for mediation between client and server to solve the development cost problem for compatibility and extension of client/server environment. This maintains mediation role for programming development between client and server and increases extension of system and portability of heterogeneous environment.

This paper implements and designs XML data processing system for EC using XML DB among heterogeneous databases. We expect cost-down by maintenance of system using this architecture for electronic commerce.

## 키워드

XML Data Processing, XML-Enabled Databases, XML Middleware, Client/server environment

## 1. 서 론

인터넷상의 정보를 표현하기 위한 HTML이 구조화된 문서를 표현하기에는 부족하다는 단점을 가지고 있음에 따라 SGML과 HTML의 단점을 보완하여

1998년 W3C(World Wide Web Consortium)에서 차세대 웹문서의 표준으로 XML(eXtensible Markup Language)을 제안하였다[1]. XML은 사용하는 플랫폼에 독립적이며, 문서정보의 전송과 교환이 쉽고, 문서의 풍부한 의미를 그대로 나타낼 수 있는 장점을 지

---

\*호남대학교 컴퓨터공학과

접수일자 : 2003. 5. 6

년 웹 표준 언어이다.

현재 XML에 대한 관심과 연구가 한창 진행되고 있으며, 점차적으로 핫 이슈가 될 것으로 예상된다. 따라서 사용자나 개발자가 쉽게 접근할 수 있도록 설계 기법과 단순 기법을 채택하고 있으며, 이에 대한 응용은 다양하다. 차세대 웹 문서포맷으로 부각하는 XML은 W3C에서 제안된 국제 표준의 전자문서 메타 언어이다[2,3].

이와 같이 전세계적으로 XML에 대한 관심이 고조되고 실제 많은 분야에서 활용되고 있기 때문에 향후 XML 응용분야의 진행 발전 단계에서 정보의 생성, 재사용성, 처리 및 지속성, 이식성 등과 같은 XML 사용 분야가 점차 증대될 것으로 예상된다.

현재 인터넷상의 전자상거래 사이트를 개발 및 운영하는 시스템들이 상당히 많이 나타나고 있으며, 앞으로도 계속적으로 증대될 것으로 예상된다. 그러나 현재의 시스템 모델은 대부분 클라이언트/서버 모델을 이용하고 있으나, 이에 대한 문제점들을 고려해 보아야 할 것이다.

첫째, 공통 특성은 미리 제한된 수의 사용자들을 가정하고 설계가 이루어진다는 점이다. 즉 동시에 서버에 접근해서 요청을 하는 최대 클라이언트의 수에 대한 예측이 가능하고, 이 예측에 맞추어 설계가 이루어진다. 그러나 보통 웹 환경처럼 사용자의 수가 급격하게 증가하는 환경에서는 이러한 설계가 곧 한계 용량을 초과함으로써, 다시 시스템 구조를 전면적으로 재설계해야 하는 문제가 있다. 또한 둘째 특성은 대부분의 클라이언트/서버 환경은 매우 안정적이고 조절 가능한 환경에서 수행이 이루어지며, 클라이언트/서버 양쪽환경에서 직접적으로 코드를 개발하는 것이 일반적이다. 이러한 제어와 개발 방법은 개발자들이 시스템 구조를 단순화하면서도 원하는 성과를 단시간에 이룰 수 있는 장점이 있지만, 다른 환경에 포팅하여야 하는 경우는 상당한 어려움을 갖게된다. 특히 웹 환경에서 클라이언트/서버간의 복잡한 제어를 해야하는 경우에 변경 및 재설계가 어려워진다.

본 논문에서는 기존의 클라이언트/서버간의 문제점을 보완하기 위해 클라이언트와 서버간의 중간 완충계층으로서 표준인 XML 처리 계층을 두고 설계하였다. 이것은 클라이언트/서버 사이에서 프로그램 개발의 중립성을 유지하며, 프로그램 개발시 시스템의 확

장과 다른 환경에 대한 이식성을 높인다. 또한 데이터베이스의 확장성과 호환성을 증대시키기 위해 XML 기반의 데이터베이스인 Excelon DB를 이용하여 XML 데이터를 관리하도록 전자상거래 시스템을 설계 및 구현하였다.

## II. 관련 연구

XML은 그 특성에 따라 데이터 중심의 XML과 문서 중심의 XML로 구분되어 진다. 데이터 중심의 XML은 애플리케이션과 데이터 저장소간의 데이터 교환을 위해 사용되기 때문에 문서의 구조 자체가 매우 정형적이며 엘리먼트와 내용이 서로 혼합되어 있지 않다. 또한 엘리먼트들간의 순서가 중요한 의미를 가지지 않는다. 반면에 문서 중심의 XML은 계약서나 도서, 그리고 광고 같은 문서를 생성하는데 사용되기 때문에 데이터의 구조가 불규칙적이며 엘리먼트와 내용이 혼합된 경우가 많다. 또한 구조적인 문서를 표현해야 하기 때문에 엘리먼트들간의 순서가 매우 중요한 의미를 갖는다. 그러나 두가지 방법 모두 중간 매핑언어 또는 중간언어를 이용하여 XML 데이터와 관계 스키마 사이의 매핑정보를 유지하는 공통적인 특징을 갖고 있다.

### 2.1 XML-가능 데이터베이스 (XML-Enabled Databases) 구조

XML-가능 데이터베이스는 XML 문서와의 데이터 전송을 위한 확장 모듈을 가진 데이터베이스를 말한다. XML-가능 데이터베이스들은 대부분 데이터 중심의 문서(data-centric document)를 저장하도록 설계되어 있다. 이것은 XML 문서를 저장할 수 있도록 특별히 설계된 테이블이 아닌 일반 사용자가 만든 테이블에 저장해야 하기 때문이다. 또한, 문서 중심의 문서(document-centric document)인 경우는 한 필드에 저장하는 경우도 있다. 이러한 구조를 가진 데이터베이스로는 DB2, Informix, Oracle8i, Microsoft SQL-Server 등이 존재한다[4].

#### - SQL 서버 2000

SQL 서버는 마이크로소프트사의 웹서버인 IIS의 ISAPI 필터를 사용하여 HTTP로 SQL 쿼리를 실행할 수 있는 기능을 제공한다. 즉 다음 예와 같이 URL에

직접 쿼리를 주어서 실행할 수 있다[5].

예) `http://server/database?sql=SELECT+last_name+FROM+Customer+FOR+XML+RAW`

또는 복잡한 SQL문을 사용하여 처리할 경우는 다음과 같이 템플릿을 만들어 저장한 후 실행할 수 있다.

예) 템플릿 파일 이름 : lastname.xml

```
<root>
  <sql:query
    xmlns:sql="urn:schema-microsoft-com:xml-sql">
    SELECT last_name FROM Customer FOR
    XML RAW
  </sql:query>
</root>
```

위와 같이 작성하였을 경우 이를 실행하기 위해 다음과 같이 URL에 입력한다. 여기서 "FOR XML RAW"은 SQL 서버에게 XML 규칙으로 반환해 주는 형식이다.

예) `http://server/database/lastname.xml`

#### - Oracle

XML Developer's Kit(XDK)에서 XML을 사용할 수 있는 도구들을 제공한다. 여기서 XML 파서는 DOM의 확장 기능을 제공하며, DOM 인터페이스, SAX 인터페이스, 적합성 검사지원, 네임스페이스와 XSLT의 완벽한 호환에 대한 기능을 갖는다. 또한 XDK에서는 자바용 XML SQL 유틸리티를 제공하며, 이를 이용하여 SQL 쿼리로부터 XML 문서를 텍스트나 DOM 형태로 생성할 수 있다[6].

## 2.2 XML 미들웨어(Middleware) 구조

XML 미들웨어는 XML 문서와 데이터베이스간에 데이터를 전송하기 위해 사용하는 프로그램이다. 이것은 매우 여러종류의 언어로 작성될 수 있으나, 대부분은 ODBC, JDBC 또는 OLE DB를 사용한다. XML 미들웨어 구조를 가지는 XML 저장 시스템에는 다음의 것들이 존재한다.

#### - XML-DBMS[7]

XML-DBMS는 XML 문서와 관계형 데이터베이스 사이에 데이터를 전송하기 위한 자바 패키지

(package)이다. 이 패키지는 사용자가 작성하는 응용 프로그램에서 사용될 수도 있다.

XML 문서 구조와 관계형 스키마와의 매핑(mapping)을 기술하기 위해서 XML-DBMS 매핑 언어를 사용한다. 이 언어는 XML 기반의 언어로서 XML 문서의 태그와 테이블의 필드의 관계를 기술한다. XML-DBMS는 이 매핑 언어를 XML 문서를 데이터베이스에 저장하는데 사용하기도 하고, 데이터베이스의 데이터를 추출하여 XML 문서를 구성하는데 사용하기도 한다.

#### - DB2XML[8]

DB2XML은 관계형 데이터베이스로부터 XML 문서로 데이터를 전송하기 위한 자바 클래스이다. XML-DBMS는 데이터베이스와 XML 문서사이에 데이터 전송이 양방향 모두 가능하지만, DB2XML은 데이터베이스에서 XML 문서로의 한방향만 가능하다. SQL 문장을 입력받아서 XML 문서를 출력하는데, 옵션에서 태그 이름을 조정할 수 있다.

#### - SilkRoute[9]

관계형 데이터를 XML 형태로 뷰를 생성하고 질의하는 시스템으로 XML을 위한 일반적이고 동적이며, 효율적인 시스템이다. 사용자가 XML-QL로 질의를 하면 사용자와 RXL(Relational to XML transformation Language)로 이루어진 뷰 질의를 바탕으로 새로운 RXL을 생성하고 이는 다시 템플릿과 데이터 로그 형태로 구성되는 뷰 트리를 생성한다.

## III. XML 중간 계층의 필요성.

관계형 데이터베이스에는 이를 관리하는 DBMS가 있고, 프로그램적으로 이 DBMS와 정보를 교류하면서, 인출, 검색, 수정 등의 작업을 하기 위해서는 이러한 데이터베이스들이 소프트웨어 적인 접근을 가능하게 하는 드라이버를 제공해 주어야 한다. 윈도우 운영 체제에서는 이러한 드라이버들은 DLL 형태로 되어 있고, 이들 DLL에 대한 사용법은 각 회사마다 차이가 있어서, 이를 구현하기 위해서는 해당 DLL의 사용법에 맞게 ASP나 JSP 같은 웹서버 프로그램들이 이를 사용해야 한다[10].

만약 관계형 데이터베이스에 들어갈 레코드가 중복

이 생겨서 불가피하게 기존에 있는 테이블의 내부 구조와 테이블의 분할 등에 대한 복잡한 변동이 생길 경우에는 데이터베이스와 연동하고 있는 부분을 수시로 수정을 하기가 어렵다. 이와 같이 스키마의 수정과 변동이 동적으로 이루어 질 경우, 관계형 데이터베이스에 바로 웹 프로그램을 연동시키기 보다는, XML로 중간 형태의 데이터를 만들어 두고, 이것을 웹 프로그램과 연동시키는 것이 바람직하다. 따라서 관계형 데이터베이스가 내부적인 문제 때문에 스키마 변동이 발생하더라도, 웹 프로그램들은 표준화된 XML을 이용하고 있으므로 웹 프로그램들은 변경할 필요 없고, 단지 관계형 데이터베이스에서 XML 문서를 만드는 부분만을 수정하면 문제가 간단해 진다. 이 경우 변환 비용이 아무리 비싸도 DOM을 가지고 하는 처리라서 코딩이 용이하며, 모듈간의 상호작용을 변경하는 것이 쉽다[그림 1].

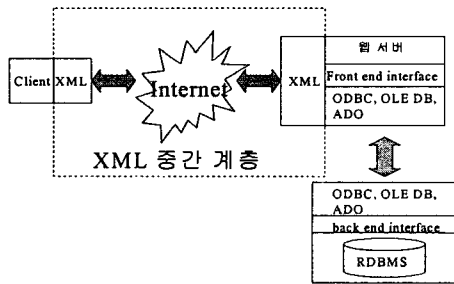


그림 1. 클라이언트와 서버사이 XML 변환방법

따라서 관계형 데이터베이스에 직접적으로 의존하는 부분이 적어질수록, 수정 및 변동에 대한 부담이 적어지고, 웹 응용을 유지 보수하는 비용이 절감된다. 또한 XML과 연동할 데이터베이스가 변경이 되더라도 쉽게 연동할 수 있다.

#### IV. XML 데이터 처리 시스템 설계 및 구현

##### 4.1 시스템 설계기술

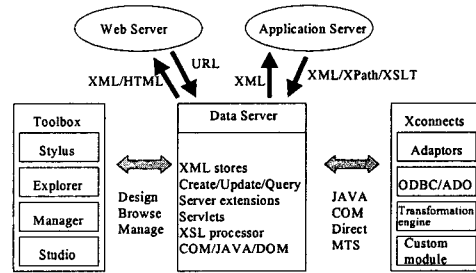


그림 2. Excelon DB의 시스템 구조

본 시스템을 구현하기 위해 Excelon DB를 이용하여 설계 및 구현하였다[그림 2]. 이 Excelon DB는 XML 문서를 전문적으로 취급할 수 있도록 고안된 DB이다. 일반적인 XML문서는 Excelon Explorer 같은 시각적인 도구를 이용해서 문서를 작성할 수 있고, 질의, 수정 등의 DB 작업을 문서 편집기를 사용하는 방식처럼 쉽게 처리할 수 있다. Excelon DB는 캐쉬 기능이 우수하여 사용량이 많은 데이터를 여러 캐쉬에 담아서 DB의 부하를 효율적으로 분산시킬 수 있는 기능이 있다. 또한 Excelon DB는 웹서버 확장 기능이 뛰어나서, 바로 URL 방식의 GET과 POST 호출만으로도 웹과 데이터베이스의 기능을 연결시킬 수 있는 기능이 있어서 웹 프로그램의 설계와 시간을 단축시킨다. 이러한 기능을 보완하기 위해 DOM과 함께 작성할 수 있는 API 함수들이 Java용과 COM용, 두 가지 라이브러리로 제공된다. 또한 기존의 데이터베이스와의 연동과 호환을 위해 관계형 데이터베이스나 비정형 데이터 등을 XML로 변환하는 비주얼 방식의 도구와 역으로 Excelon DB의 내용을 관계형 데이터베이스에 변환하여 저장할 수 있는 도구를 제공함으로써 다른 데이터베이스와의 연동이 쉽다.

Excelon DB는 크게 세부분으로 구성되며, Excelon Toolbox, Excelon Data Server, Excelon Xconnects 부분으로 구성되어 있다[11].

Excelon Toolbox의 기능은 데이터베이스 관리자나 데이터베이스의 설계, 자료검색, 수정 등의 작업을 하

고자 하는 사용자에게 손쉽게 데이터베이스를 관리할 수 있도록 비주얼 방식의 도구들을 포함한다.

Excelon Xconnects의 기능은 기존의 RDB나 다른 ODBC/ADO방식의 데이터베이스로부터 데이터를 변환해서 Excelon DB에 저장할 수 있도록 지원하는 도구이다. 또한 역으로 Excelon DB에서 다른 데이터베이스로 저장을 위해 변환을 하는 기능도 있다. 이때 다른 데이터베이스의 데이터에 접근하기 위해서는 해당 데이터베이스 드라이버를 제공해야 하는데, 이러한 드라이버가 약 70여가지 이상을 포함하고 있다.

Excelon Data Server의 기능은 XML을 파싱된 형태로 저장하고 있어서, 노드단위의 정보 처리에도 신속하게 대응할 수 있고, 웹 서버와 연동하여 데이터베이스의 기능을 활용할 수 있는 웹서버 익스텐션 기능이 있다. 이는 URL 방식으로 오는 요청을 웹서버로부터 바로 Excelon 서버로 연결하여 데이터베이스가 제공하는 모든기능을 사용할 수 있도록 하며, 웹 프로그램을 손쉽게 제공한다. 또한 데이터베이스 기능을 바로 프로그램과 연결할 수 있도록 하는 서버 익스텐션 API를 JAVA방식과 COM 방식으로 지원한다.

#### 4.2 DOM 트리구성

엑셀론 데이터 서버는 DOM 빌더를 사용하여 데이터베이스로부터 XML 문서의 DOM 트리를 생성하고, 이를 운영 관리한다. DOM 트리의 노드는 DOM API 명령어를 이용하여 삽입, 삭제, 수정, 추가 등의 다양한 액션을 처리한다. 이러한 DOM 트리는 수정 즉시 자동으로 내부에서 관리하는 데이터인 XML 문서로 변환 수정시킨다. 이 XML 문서에 대한 처리 동작은 마치 데이터베이스의 데이터를 관리 운영하는 것과 유사하며, 이러한 XML 문서는 항상 최근의 문서로 실시간으로 유지 관리한다. 다음 그림은 XML 문서를 요소와 속성, 콘텐츠로 분리 파싱하여 DOM 트리를 구성한 그림이며, 이러한 각 노드들은 DOM API를 이용하여 접근 가능하다. 외부 사용자로부터 각 노드에 대한 수정 및 추가에 대한 정보를 입력받아 이를 즉시 트리 노드에 반영한 후 자동으로 XML 문서를 업데이트 시킨다[그림 3].

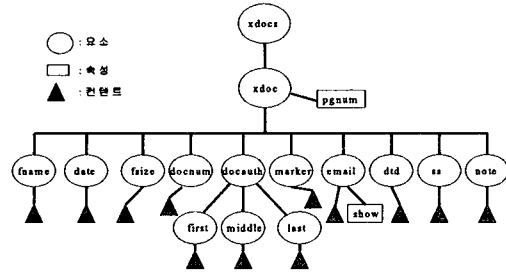


그림 3. DOM 트리구조 예

#### 4.3 시스템 설계 및 구현

본 논문의 전자상거래를 위한 XML 데이터처리 시스템에 적용할 정보의 예로 자동차에 대한 경매 관리를 하는 시스템을 구성하였다. 본 시스템에서는 Excelon DB를 이용하여 사용한 설계 기술은 다음과 같다. 첫 번째로는 서버 사이트 XSL 프로그래밍 기술로써 내용과 구조를 담고 있는 XML과 이를 사용자에게 보여주는 것을 정의해 놓은 XSL을 Excelon에 내장되어 있는 XSLT 프로세서를 통해 HTML을 생성해서 클라이언트의 브라우저로 내보내는 프로그래밍 스타일이다. 이 방식은 데이터 계층과 표현계층을 구분하도록 하는 프로그래밍 방식이다. 두 번째로 XPATH 기술로서 RDBMS에서는 SQL이라는 것을 이용해서 원하는 데이터를 추출한다. 마찬가지로 XML에 저장되어 있는 데이터를 추출할 경우 XPATH라는 방법을 사용하였다. 세 번째로 서버 익스텐션 기술로서 Excelon의 확장된 특성으로 JAVA나 COM으로 구현이 가능하고, Excelon의 캐쉬내에서 관리된다. JAVA나 COM으로 만들어서 Excelon에 등록한 다음, 서버 프로그램(JAVA, JSP, ASP 등)에서 해당 컴포넌트의 메소드를 호출한다.

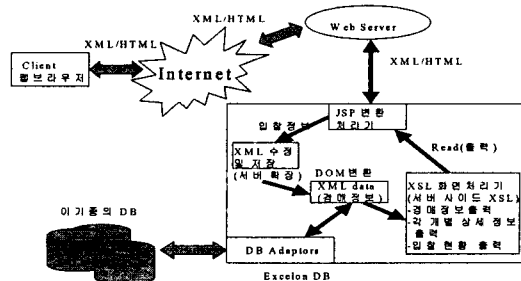


그림 4. 전자상거래를 위한 시스템 구현구조

구현 환경으로는 JDK1.3, 서버컴 JSP 엔진인 Jakarta-tomcat3.2.1, 화면출력을 위한 XSL 변환기술, 전자상거래 정보를 XML 문서로 작성, DB로는 Excelon DB를 이용하여 설계하였다.

위 [그림 4]에서 XML 데이터는 마치 DB처럼 관리해야 할 대상이며, 여기서는 데이터의 예로서 자동차에 대한 경매정보를 입력하였다. 또한 JSP 변환 처리기는 XSL파일과 XML 데이터를 분석하여 웹서버에 화면 출력 정보인 XML/HTML을 생성하는 역할을 하며, 이러한 정보가 클라이언트에 전송된다. XSL 파일들은 경매정보들의 전체 정보를 출력하거나, 개별 항목들에 대한 경매 현황 등을 출력할 수 있도록 구성한다. XML 데이터는 실제 데이터베이스의 내용물인 경매정보이며, XML 수정 및 저장 처리기에 의해 관리된다. [그림 4]에서 본 시스템에서 설계된 각 모듈을 설명하면 다음과 같다.

(1) XSL 화면처리기 설계

제조사와 스타일에 따라 XML 데이터를 검색하여 각각의 개별 상세 정보 출력을 지원하며, 현재의 입찰 현황과 입찰참여하기에 대한 HTML 변환을 설계하였다. 화면 설계는 표준화된 XSL을 이용함으로써 프로그램 개발과 유지보수에 따른 부담을 덜도록 XSL 파일을 설계하였다. 이것은 클라이언트/서버 사이에서 프로그램 개발의 중립성을 유지하며, 프로그램 개발시 시스템의 확장과 다른 환경에 대한 이식성을 높인다.

(2) JSP 변환처리기 설계

화면출력 XSL 파일, 외부 사용자로부터의 입력 파라미터(제조사, 스타일 등)와 XML 데이터 파일을 입력으로 받아서 HTML로 변환시킨다. 여기서 엑셀론 DB의 API를 사용하여 변환 처리를 설계하였으며, JSP로 구현하였다.

(3) XML 데이터를 이용한 수정 및 저장 설계

DOM 변환을 한 후 경매정보의 삽입과 수정을 수행한다. 여기에서 XML 데이터의 수정 및 저장을 관리하기 위해 JAVA를 이용하여 구현하였다.

다음 [그림 5]는 본 시스템에서 설계한 자동차 입찰에 대한 DTD와 XML 데이터 예를 보여준다. 또한 입찰현황은 사용자의 입찰에 따라 XML 데이터가 수시로 업데이트된다.

```
<?xml version="1.0" encoding="euc-kr"?>
<!DOCTYPE 자동차종류 [
<!ELEMENT 자동차종류 (자동차)*>
<!ELEMENT 자동차 (제조사, 모델, 제조년, 사진,
스타일, 도어, 가격, 소유자, 입찰현황*)>
<!ELEMENT 소유자 (이름, 이메일, 전화)>
<!ELEMENT 입찰현황 (입찰자, 수량, 입찰금액)>
<!ELEMENT 제조사 (#PCDATA)>
<!ELEMENT 모델 (#PCDATA)>
<!ELEMENT 제조년 (#PCDATA)>
<!ELEMENT 사진 (#PCDATA)>
<!ELEMENT 스타일 (#PCDATA)>
<!ELEMENT 도어 (#PCDATA)>
<!ELEMENT 가격 (#PCDATA)>
<!ELEMENT 이름 (#PCDATA)>
<!ELEMENT 이메일 (#PCDATA)>
<!ELEMENT 전화 (#PCDATA)>
<!ELEMENT 입찰자 (#PCDATA)>
<!ELEMENT 수량 (#PCDATA)>
<!ELEMENT 입찰금액 (#PCDATA)>
]>
<자동차종류>
<자동차>
    <제조사>현대</제조사>
    <모델>레오</모델>
    <제조년>1996</제조년>
    <사진>아반떼.jpg</사진>
    <스타일>세단</스타일>
    <도어>4</도어>
    <가격>7000</가격>
    <소유자>
        <이름>홍길동 </이름>
    <이메일>Hong@Domain.Com</이메일>
    <전화>781781782</전화>
    <소유자>
    <입찰현황>
    <입찰자>박아무개 </입찰자>
    <수량> 1 </수량>
    <입찰금액>4000 </입찰금액>
    <입찰현황>
</자동차>
</자동차종류>
```

그림 5. 본 시스템에 적용한 dtd와 XML 데이터 예

자동차 경매에 대한 시스템 구현은 다음과 같다.

사용자가 제조사와 스타일을 입력하면 그에 따른 XML 데이터를 검색하여 해당 항목을 추출하여 XSLT 프로세서를 거쳐 웹브라우저에 HTML로 출력한다. 사용자가 해당 스타일을 클릭하면 경매에 대한 정보를 보여준다. 여기에는 입찰자, 수량, 금액을 차례대로 출력시킨다. 사용자가 입찰참여하기를 클릭하면 입찰자명, 입찰수량, 입찰단가를 입력하여 경매에 입찰할 수 있으며 다음 [그림 6]과 같다. 입찰 후 입찰 현황보기를 클릭하면 전체 입찰 현황을 볼 수 있으며 [그림 7]과 같다.

[그림 7]과 같다.

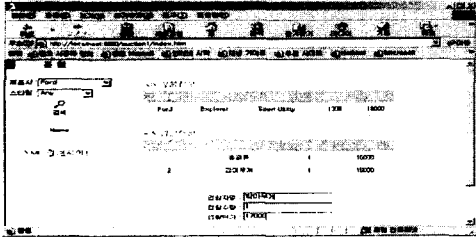


그림 6. 경매 입찰 입력 화면

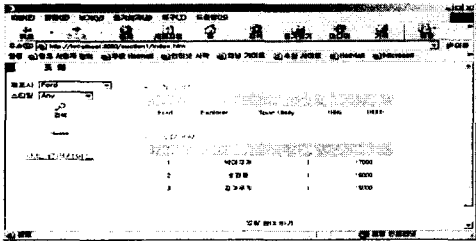


그림 7. DB에 입찰정보 저장 후 출력

#### 4.4 고찰

현재 기존의 프로그램 언어를 이용하여 서버 프로그램 설계를 대부분 하였으나, 본 시스템에서는 표준 기술인 XSL 변환과 XML 기술을 이용하여 자동차 경매에 관련된 서버용 프로그램 설계를 설계 및 구현하였다. 이것은 표준 기술인 XSL과 XML 기술을 적용하여 구현함으로써 향후 유지보수와 호환성에 유리한 장점을 갖는다고 생각된다. 따라서 서버 프로그램 개발시 중간 표준 언어로 XML 개발 환경 계층을 적용할 경우 다른 이질의 데이터베이스와의 연동과 스키마 수정에서도 매우 효과적으로 대처할 수 있다.

### V. 결 론

기존의 클라이언트/서버간의 문제점을 보완하기 위해 본 논문에서는 클라이언트와 서버간의 중간 완충 계층으로서 표준인 XML 처리 계층을 두고 설계하였다. 이것은 클라이언트/서버 사이에서 프로그램 개발의 독립성을 유지하며, 프로그램 개발시 시스템의 확장과 다른 환경에 대한 이식성을 높인다. 또한 데이터

베이스의 확장성과 호환성을 증대시키기 위해 XML 기반의 데이터베이스인 Excelon DB를 이용하여 XML 데이터를 관리하도록 전자상거래 시스템을 설계 및 구현하였다.

이러한 XML 기반의 전자상거래 구조를 활용함으로써 시스템 확장성과 시스템의 유지보수를 위한 비용절감이 기대된다. 또한 XML 기반의 전자상거래 제품 설계 및 구현시에 기본 모델로서 활용될 것으로 예상된다.

### 참고 문헌

- [1] Extensible Markup Language(XML)1.0, W3C Recommendation, <http://www.w3.org/TR/1998/REC-xml-19980210>, 1998.
- [2] Elliotte Rusty Harold, XML Bible, IDG BOOKS, 1999.
- [3] Steven Holzner, XML complete, McGraw-Hill, 1999.
- [4] Ronald Bourret, XML Database Products, <http://www.informatik.tu-darmstadt.de/DVS1/staff/bourret/xml/XMLDatabaseProds.htm>, 2000
- [5] <http://msdn.microsoft.com/workshop/xml/articles/xmlsql/default.asp>
- [6] [http://otn.oracle.com/tech/xml/xdk\\_java/content.html](http://otn.oracle.com/tech/xml/xdk_java/content.html)
- [7] Ronald Bourret, XML-DBMS, <http://www.informatik.tu-darmstadt.de/DVS1/staff/bourret/xmldbms/readme.html>, 2000
- [8] Volker Turau, DB2XML 1.3, <http://www.informatik.fh-wiesbaden.de/~turau/DB2XML/index.html>, 2000
- [9] Mary Fernandez WangChiew Tan Dan Suci, "SilkRoute: Trading between Relations and XML", WWWg, 2000
- [10] Homer, et al., Professional Active Server Pages 3.0, WROX, 2000.
- [11] <http://www.excelon.com>

저자 소개



**김철원(Cheol-Ueon Kim)**

1997년 광운대학교 컴퓨터공학과  
(공학박사)

1988년 ~ 현재 호남대학교 컴퓨터  
공학과 교수

※ 관심분야 : XML 응용, 멀티미디어 정보