

# Oracle 8i를 이용한 통합 상품 표현 모델의 구현

하 상 호<sup>†</sup>

## 요 약

B2B의 전자상거래의 발달로 상품 정보를 효과적으로 표현하고, 이러한 정보를 회사들간에 교환하기 위한 디지털 카탈로그는 점차 그 중요성이 인식되고 있다. 본 논문은 상품들을 통합하여 효과적으로 표현할 수 있는 통합 상품 표현 모델을 참조하고, 이 모델을 Oracle 8i에서 제공하는 XDK(XML Development Kit)를 사용하여 구현한다. 먼저, 상품 표현 모델을 XML DTD로 표현하고, 이 DTD로 표현된 XML 상품 정보를 효과적으로 Oracle 8i에 데이터베이스에 저장하고 검색하는 시스템을 구축한다. 마지막으로, 구축된 시스템에 대한 실행 예제를 보여준다.

## Implementation of an Integrated Product Expression Model using Oracle 8i

Sangho Ha<sup>†</sup>

## ABSTRACT

With the rapid growth of the B2B electronic commerce, digital catalogs for expressing product information effectively and exchanging it easily between companies become increasingly important. In this paper, we refer a product expression model to effectively integrate product information, and implement the model using XDK(XML Development Kit), which is given by Oracle 8i. We will first derive an XML DTD reflecting the model. We will then develop a system that stores and retrieves product information on/from Oracle 8i databases, which is written in XML, complying the XML DTD structure. Finally, We will give an experiment on the system.

**Key words:** 전자카탈로그, 통합 상품 표현모델, XML, 전자상거래, Oracle 8i, Java

## 1. 서 론

최근 인터넷을 통해서 거래되고 있는 모든 상품에 대한 정보가 전자카탈로그 형태로 표현되고 있다. 전자카탈로그[1,2]는 주문, 지불 등과 같은 전자상거래에서 사용되는 다양한 비즈니스 프로세스와 통합이 가능하고, B2B 전자상거래의 경우에는 인터넷과 연결되기 때문에 이를 지원할 수 있는 언어로 표현하는 것이 필요하다. 또한, 전자상거래에 사용되는 시스템은 분산 환경이고, 다양한 기계가 사용될 수 있

기 때문에, 이질적 분산 환경이 고려되어야 하고, 전자카탈로그의 데이터는 문서(document)보다는 구조적 데이터의 특성을 갖는다는 것이 고려되어야 한다.

이러한 모든 조건들을 충족시킬 수 있는 언어가 최근에 웹 상에서 데이터 교환을 위해서 W3C에 의해서 채택된 새로운 웹 문서 표준 기술 언어인 XML[3]이다. XML은 HTML과는 달리, 출력 형태보다는 내용을 기술하도록 설계되었으며, 사용자가 태그를 새롭게 정의할 수 있고, 임의 수준까지 중첩된 구조를 표현할 수 있고, 데이터의 구조를 표현하는 문법을 포함할 수 있는 특징을 갖는다. XML에 기반한 전자상거래 프레임워크 ebXML[4], BizTalk[5]

본 논문은 정보통신부의 대학 IT 연구센터 육성 지원 사업에 의해 수행된 것임.

접수일 : 2002년 2월 27일, 완료일 : 2003년 4월 3일

<sup>†</sup> 정회원, 순천향대학교 정보기술공학부 조교수

등에서 수행되고 있으며, 특히 xCBL[6]과 Rosetta Net[7] 등에서는 전자상거래에서 사용되는 다양한 비즈니스 프로세스에 대해서 XML 문서 표준화를 시도하고 있다. 그러나 xCBL이나 RosettaNet에서 제안하고 있는 상품 카탈로그를 살펴보면, 상품 속성들이 분류되어 있지 않고, 고객의 상품 검색을 용이하게 해주는 유형 정보 등은 고려하지 않는다.

[8]에서는 상품정보 속성들을 유형화하는 기준을 설정하고, 이러한 기준에 의해서 속성들을 체계적으로 분류하고, 고객의 상품 검색을 위한 유형 정보를 포함함으로써 상품정보에 대한 하나의 표준 모델을 제안하고 있다. 본 논문에서는 이 모델을 XML DTD (Document Type Definition)[9]로 표현하고 구현한다. 구현은 모델로 표현된 XML 데이터를 Oracle 8i 데이터베이스[10]에 저장하고, 저장된 XML 데이터를 효과적으로 검색하는 시스템을 구축하는 것으로 이루어진다. 시스템 구축은 Oracle 8i에서 제공하는 XML 파서, XSLT 프로세서, XML SQL Utility 등으로 구성된 XDK(XML Development Kit)[11]를 사용한다.

2장에서는 논문에서 참조하고 있는 XML 기반 상품 카탈로그 모델에 대해서 간략히 기술하고, 도서와 컴퓨터의 상품에 대해서 모델을 적용한 결과를 보이고, 3장에서는 2장에서 기술된 XML DTD를 관계 데이터베이스 스키마로 사상하는 방법을 제안하고, 4장에서는 제안된 상품 표현 모델을 구현하고, 5장에서는 시스템에 대한 실행 예제를 보인다. 마지막으로 6장에서는 결론을 언급한다.

## 2. 통합 상품 표현 모델

[8]에서는 상품을 통합하여 효과적으로 표현할 수 있는 상품 카탈로그 모델을 제안하고 있다. 이 모델에서, 상품군은 대표상품을 표현하며, 상품은 대표상품에 속한 특정 개별상품을 표현한다. 개별 상품은 분류체계 정보와 자신에 대한 상세정보를 표현하는 본문으로 구성된다. 본문은 내용정보와 유형정보로 구분되며, 내용정보는 기본정보, 판매정보, 부가가치정보로 세분화된다. 이러한 정보와 유형정보는 공통정보와 고유정보로 또다시 세분화된다. 고유정보는 각 상품에 대해서 특정한 정보가 표현되는 것으로, 상품에 따라서 특정한 정보가 없으면 해당 상품의 고유정보는 존재하지 않는다. 또한, 특정한 정보를

갖는 상품은 기본, 판매, 부가가치, 유형 정보 상에 그 정보를 추가 표현할 수 있다. 이러한 점에서 이 상품 카탈로그 모델은 확장 가능하다. 또한, 모든 상품 정보가 동일한 모델로서 표현될 수 있다는 점에서 통합적이다.

그림 1은 [8]의 상품 카탈로그 모델에 따라 도서와 컴퓨터 상품을 효과적으로 통합하여 표현한다. 따라서 상품의 분류체계 정보는 도서나 컴퓨터이다. 본문은 내용정보와 유형정보로 구분되고, 내용정보는 기본정보, 판매정보, 부가가치정보로 세분된다. 기본정보는 두 상품에 대해서 공통적인 공통정보와 각 상품에 대해서 고유한 정보들을 표현하는 고유정보로 구분된다. 공통정보는 브랜드명, 모델명, 메이커, 가격, 제품외관 등으로 구성되는데, 이러한 정보는 도서와 컴퓨터에 대해서 동일한 목적으로 사용될 수 있다. 가령, 브랜드명은 책의 경우에 제목을 표현하고, 컴퓨터의 경우에 제품명을 표현하며, 모델명은 책의 경우에 ISBN 번호가 표현되며, 컴퓨터의 경우에는 모델 번호가 표현되고, 메이커의 경우에 책은 출판사에 해당되며, 컴퓨터는 제조사에 해당된다. 고유 정보는 다시 도서와 컴퓨터로 구분된다. 도서의 경우 도서에 특정한 정보가 여기에 기술된다. 가령, 저자, 역자, 출판일, 페이지, 목차, 요약, 판형, 첨부물 등의 정보는 도서에 특정한 정보이다. 컴퓨터의 경우 컴퓨터의 기본 사양과 같이 컴퓨터에 특정한 정보가 여기에 표현된다.

판매정보, 부가가치정보, 유형정보도 공통정보와 고유정보로 각각 세분된다. 판매정보에는 컴퓨터의 고유정보가 표현되고, 유형정보에는 도서의 고유정보가 표현된다. 그리고 부가가치정보에는 두 상품에 대해서 특정한 정보가 존재하지 않는다. 그러나 상품에 대해서 고유한 정보가 발생하면, 이러한 정보들이 추가로 모델상에 표현될 수 있다. 또한, 그림 1에 다른 상품도 추가할 수 있으며, 추가된 상품에 대해서 고유정보가 존재할 경우에 이러한 정보는 그 성격에 따라서 적절히 해당 고유정보란에 표현될 수 있다.

## 3. 관계 테이블 스키마

그림 1에서 보여준 도서와 컴퓨터의 상품 카탈로그 모델은 XML DTD로 기술되고, 도서와 컴퓨터 상품들은 이 XML DTD의 구조를 갖는 XML 문서로

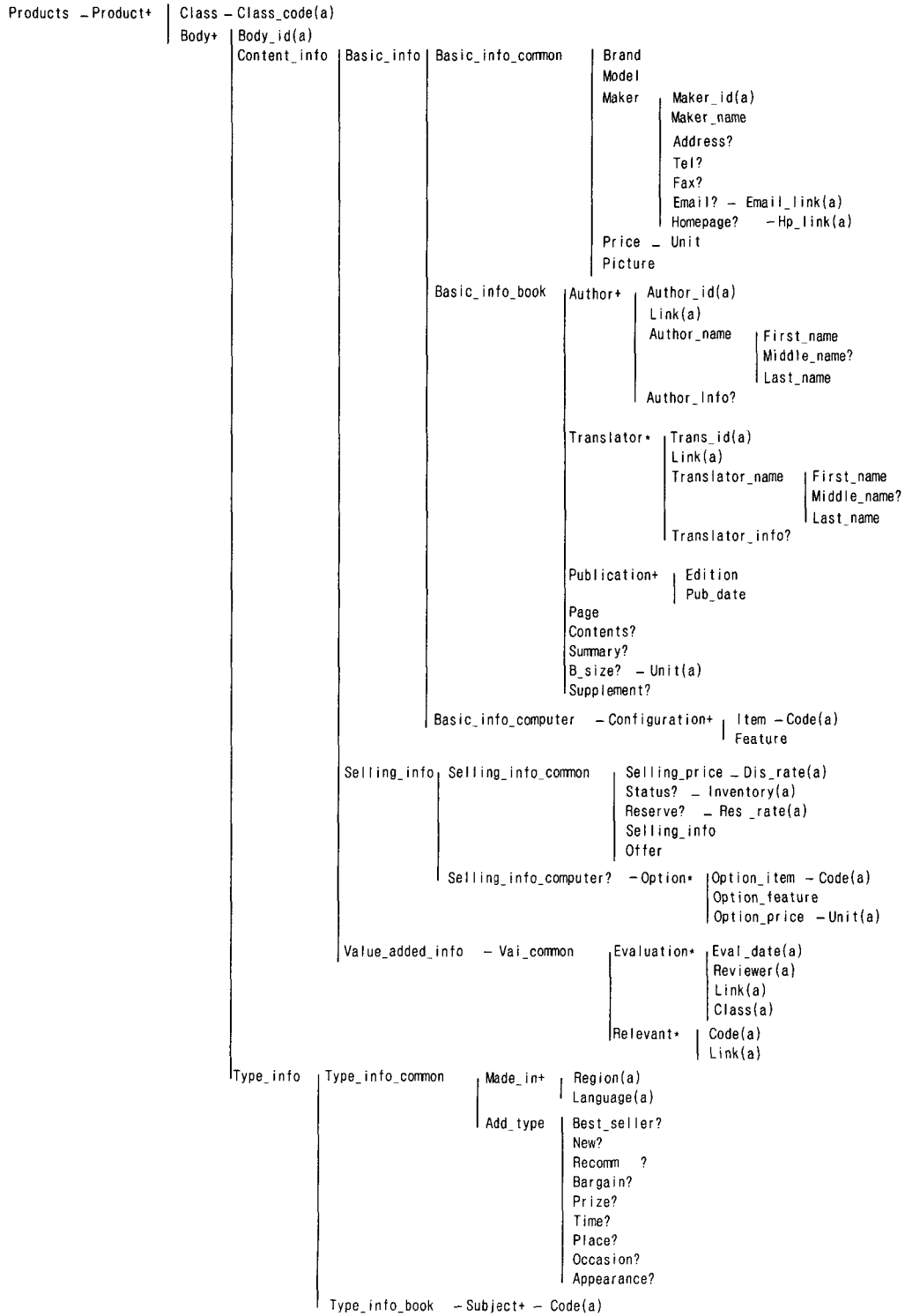


그림 1. 도서와 컴퓨터의 통합 상품 카탈로그 모델

작성된다. 본 논문에서는 이와 같이 작성된 XML 문서를 해당 데이터베이스에 저장하고 검색하는 방법을 다룬다. 논문에서는 관계 데이터베이스를 가정하고서, XML 문서를 관계 데이터베이스에 저장하기 위해서 XML 문서의 DTD를 관계 테이블 스키마로 사상하는 방법을 기술한다.

주어진 XML DTD로부터 관계 테이블 스키마에 사상하는 규칙을 다음과 같이 제안한다. 그림 1과 같이 DTD를 표현하는 트리 T를 가정한다. 첫째, T의 루트 노드나 중간 노드는 자신의 테이블로 생성된다. 이 테이블은 자신의 기본 키(primary key)와 이 노드의 부모 노드에 대해서 생성된 테이블의 기본 키를

외부 키(foreign key)로 갖는다. 기본 키 값은 ID의 속성으로부터, 외부 키 값은 IDREF의 속성으로부터 각각 가져온다. 또한, 집합 값(set-value)을 갖지 않는 자식 노드는 생성된 테이블의 필드가 된다. 둘째, 집합 값을 갖는 잎 노드는 별도의 테이블로 생성된다. 이 경우에 생성된 테이블은 그 부모 노드의 기본 키 값을 외부 키로 갖는다. 셋째, 엘리먼트(ELEMENT)의 각 속성 (ATTRIBUTE)은 그 엘리먼트가 속하는 테이블의 한 필드가 된다.

그림 2는 이러한 규칙에 따라서 그림 1의 모델로부터 유도되는 관계 테이블 스키마를 보여준다. 상품들간의 공유정보는 상품들간에 공유되는 테이블에

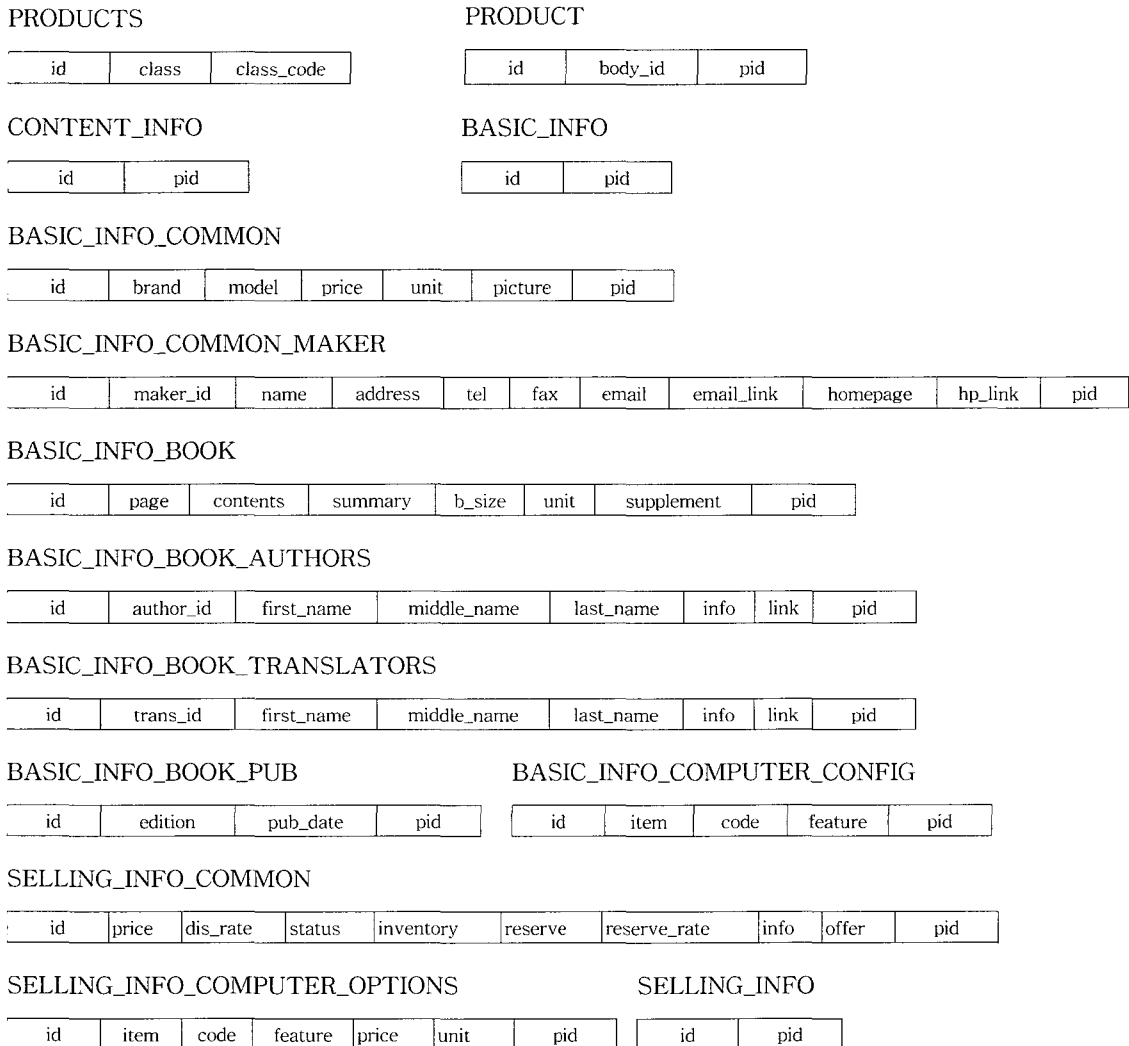


그림 2. 관계 테이블 스키마

VAI		TYPE_INFO								
id	pid	id	pid							
VAI_COMMON_EVAL										
id	evaluation	eval_date	reviewer	link	class	pid				
VAI_COMMON_RELEVANT										
id	erelevant	code	link	pid						
TYPE_INFO_COMMON_MADE_IN				TYPE_INFO_BOOK_SUBJECT						
id	made_in	region	language	pid	id	subject	code	pid		
TYPE_INFO_COMMON_ADD_TYPE										
id	best_seller	new	recomm	bargain	prize	time	place	appearance	add_date	pid

그림 2. 계속

표현되고, 특정 상품의 고유정보는 자신의 테이블에 표현되는 것을 알 수 있다. 가령, 도서와 컴퓨터의 공통된 기본정보는 BasicCommonInfo, BasicCommonInfo\_Maker 상에 표현되고, 도서의 고유 기본정보는 BookBasicInfo, BookBasicInfo\_Authors, BookBasicInfo\_Translators, BookBasicInfo\_Pub의 3개의 테이블에 표현되고, 컴퓨터의 고유 기본정보는 ComputerBasicInfo에 표현된다.

#### 4. 구현

여기서는 [8]에서 제안된 XML 기반 상품 카탈로그 모델을 위한 시스템을 설계하고 구현한다. 그림 3은 시스템의 전체 구조를 보여준다. 시스템은 크게 사용자, 미들웨어, 그리고 데이터베이스의 3부분으로 구성된다. 사용자와 미들웨어간의 인터페이스는 Java Servlet을 사용하여 구현하고, 시스템과 데이터베이스와의 인터페이스는 JDBC를 사용하여 구현한다. 데이터베이스에는 Oracle 8i가 사용된다. 시스템과 인터페이스의 구현은 모두 Java를 사용하여 구현되었으며, 따라서 개발된 시스템은 특정 플랫폼에 독립되어 사용될 수 있다.

미들웨어는 크게 저장 모듈과 검색 모듈로 구성된다. 저장 모듈은 사용자로부터 XML 문서를 입력받고, 이것을 파싱하여 의미 있는 데이터를 추출하여 데이터베이스 테이블에 저장하는 역할을 수행하고, 검색 모듈은 요청된 데이터를 데이터베이스로부터

가져와서 XML 문서로 구성하고, 이것을 HTML 문서로 변환하여 사용자에게 전달하는 역할을 수행한다. 이 두 모듈을 용이하게 구현하기 위해서 Oracle에서 제공하는 XML 개발 도구, XDK(XML Developer's Kit)를 사용한다. 다음은 저장 모듈과 검색 모듈에 대해서 설명한다.

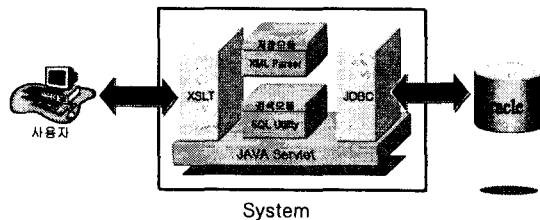


그림 3. 전체 시스템 구조도

#### 저장 모듈

XML 문서의 저장은 웹 브라우저를 통해서 이루어질 수 있다. 시스템과 웹 브라우저와의 인터페이스는 Java Servlet을 사용하여 구현한다. 사용자는 브라우저상에서 해당 XML 문서의 URL을 입력하거나 XML 문서 자체를 편집기를 통해서 작성한 후, 이 문서의 저장을 요청할 수 있다. 그림 4는 XML 문서의 저장 과정을 보여준다. 먼저, XML 문서는 Servlet을 통해서 미들웨어의 XML Parser에 전달된다. 이 파서는 전달된 XML 문서를 파싱하고, 그 유효성 검사를 수행한 후에 DOM 트리를 생성하고, Document를 추출한다. 이 Document로부터 Element와

Attribute에 포함된 데이터를 추출하고, 이를 해당 관계 테이블에 저장하기 위한 SQL 문을 생성한다. 이 SQL 문을 JDBC를 통해서 Oracle 8i에 전달한다. Oracle8i는 전달된 SQL 문을 수행함으로써 데이터를 명세된 관계 테이블에 저장한다.

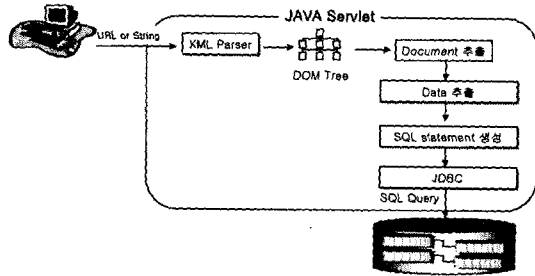


그림 4. XML 문서 저장 모듈

**검색 모듈**

데이터베이스에 저장된 XML 데이터에 대한 검색은 XSQL Servlet에 의해서 수행된다. 그림 5는 XML 데이터에 대한 검색 과정을 보여준다. 사용자가 XML 데이터에 대한 검색을 요청하면, 시스템은 이러한 사용자 요청을 토대로 XSQL Page를 생성하고, XML Parser에 전달한다. XML Parser는 전달된 XSQL Page가 올바르게 작성되었는지를 검증하고, XSQL Page 프로세서에 전달한다. XSQL Page 프로세서는 XSQL Page를 처리하면서 페이지에 포함된 질의어를 추출하여 XML SQL Utility에 전달한다. 또한, Page에 명세된 XSL 스타일시트 문서를 XSLT에 전달한다. XML SQL Utility는 전달받은 질의어를 Oracle 8i에 보내어 수행하도록 하고, 그 결과를 XML 문서로 생성한다. 마지막으로, XSLT 프로세

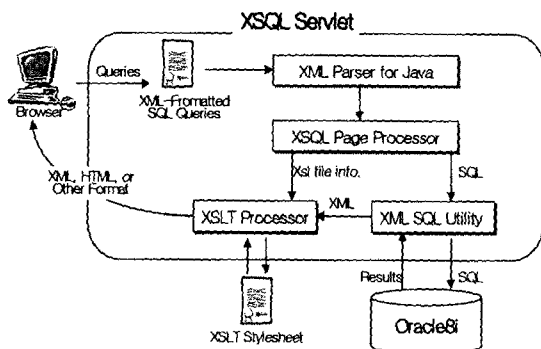


그림 5. Oracle8i에서의 XML 문서의 질의

서는 XQL Page 프로세서로부터 전달된 XSL 문서를 사용하여 결과 XML 문서를 HTML 문서로 변환하여 사용자에게 전달한다.

**5. 실행 예제**

여기서는 4장에서 기술한 XML 기반 상품 카탈로그 모델을 구현한 시스템을 테스트하고, 도서와 컴퓨터 상품에 대해서 시스템의 실행 예제를 보인다. 이를 위해서 먼저, 도서와 컴퓨터의 상품 정보를 그림 1의 상품 카탈로그 모델에 따른 XML DTD를 작성하고, 이 DTD를 갖는 XML 문서로 작성하고, 다음에 이 문서를 시스템의 XML 저장 모듈을 통해서 Oracle 8i에 저장한다. 다음에, 시스템의 XML 검색 모듈을 통해서 저장된 XML 데이터를 다양한 형태로 검색한다.

그림 6은 XML 문서를 웹 브라우저 상에서 개발된 편집기를 통해서 XML 문서가 작성된 화면을 보여준다. 화면 아래의 보내기 단추를 누르면, 작성된 XML 문서가 Java Servlet을 통해서 저장 모듈에 전달된다. 다음은 XML 문서가 저장 모듈을 통해서 데이터베이스에 저장되는 과정을 설명한다. XML 문서는 books.xml로 명명되었다고 가정한다. 저장 모듈은 일련의 XSQL 명령어를 사용하여 XML 문서를 그림 3의 테이블에 사상한다. 가령, XML 문서를 BASIC\_INFO\_COMMON 테이블에 사상하는 경우를 생각해보자. 이를 위한 XSQL 명령어는 다음과 같이 작성된다.

```
/xsql/bin/xsql insert_basic_info_common.xsql
posted-xml = books.xml
```

위 XSQL 명령어는 XSQL 페이지 프로세서가 posted-xml 매개변수로 전달되는 books.xml을 처리하게 한다. 위 명령어에 명세된 insert\_basic\_info\_common.xsql의 XSQL 페이지는 다음과 같이 작성된다:

```
<!--insert_basic_info_common.xsql-->
<xsql:insert-request xmlns:xsql="urn:oracle-xsql"
table="basic_info_common"
transform="insert_basic_info_common.xsl"
connection="demo"/>
```

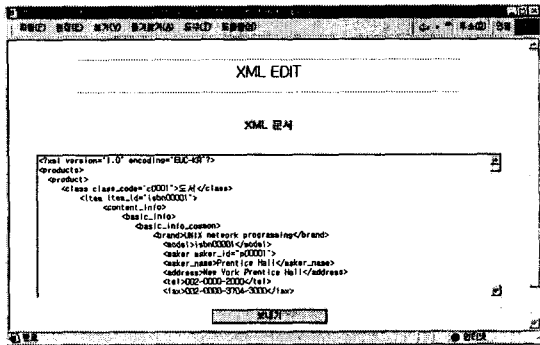


그림 6. XML 문서의 저장 화면

위의 XSQL 페이지는 다음과 같이 두 가지 행동을 취한다. 먼저, XSLT 프로세서를 사용하여 HTTP를 통해서 전달된 books.xml을 transform 매개변수에 명세된 insert\_basic\_info\_common.xml의 XSLT 문서를 사용하여 변환한다. 여기서 insert\_basic\_info\_common.xml은 books.xml 문서로부터 그림 3의 BASIC\_INFO\_COMMON의 테이블에 포함될 내용만을 발췌하도록 작성된다. 다음에 XML SQL Utility를 사용하여 변환된 XML 문서를 table 매개변수에 명세된 basic\_info\_common의 테이블에 저장한다.

그림 7은 Oracle 8i에 저장된 XML 도서 데이터에 대한 검색 메뉴를 보여준다. 검색은 도서명/키워드 검색, ISBN 검색, 저자/역자 검색, 발행기간별 검색 등으로 구성된 기본 검색과 고급 검색으로 구성된다. 그림 8의 화면은 고급 검색을 선택했을 경우의 화면을 보여준다. 고급 검색은 도서명/키워드, ISBN, 저자/역자명, 출판일, 출판사, 분야 등을 조합하여 검색 조건을 명세하고, 검색 결과도 선택적으로 명세하는

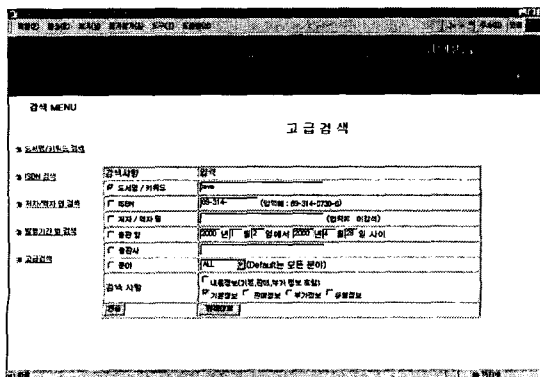


그림 7. XML 데이터 검색 요청 화면

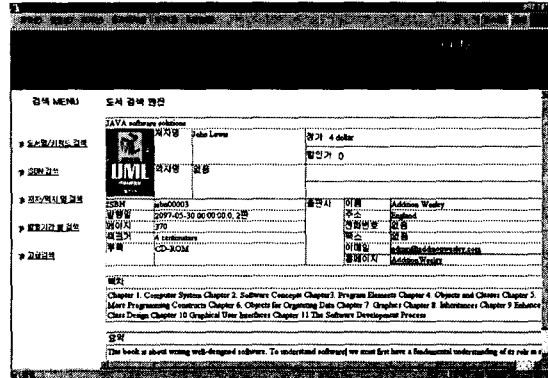


그림 8. XML 데이터 검색 결과 화면

것을 가능하게 한다. 가령, 검색 결과는 기본정보, 판매정보, 부가가치정보, 유형정보 등을 선택적으로 조합하여 명세하는 것이 가능하다.

## 6. 결론

본 논문에서는 다양한 상품들을 효과적으로 통합하여 표현할 수 있는 상품 카탈로그 모델[8]을 참조하고, 이 모델을 Oracle 8i에서 제공하는 XML 개발 도구를 사용하여 구현하였다. 구현된 시스템은 웹 브라우저를 통한 XML 문서의 저장과 저장된 XML 데이터에 대한 다양한 분류별 검색을 지원한다. 향후 연구로, 시스템의 구성 모듈을 최적화할 계획이다. 또한, XML 기반 데이터 모델의 구현 결과를 컴포넌트로 구성하여, 전자상거래 시스템 구축이나 각종 웹 기반 응용 프로그램 구축시 데이터베이스 관련 컴포넌트로 효과적으로 활용할 수 있게 할 계획이다.

## 참고 문헌

- [1] CommerceNet. "Catalogs for the Digital Marketplace," Research Report, www.commerce.net, march 1997.
- [2] Segev, A., Wan, D. and Beam, C., "Designing Electronic Catalogs for Business Value: Results of the CommerceNet Pilot," CITM working paper, CITM-WP-1005, Haas School of Business, University of California, Berkeley, October 1995.
- [3] 2000 "Extensible Markup Language (XML)",

<http://www.w3.org/XML/>

[4] <http://www.ebXML.org/>

[5] <http://www.biztalk.org/>

[6] Commerce One, "xCBL 3.0 Element and Data Type Structures(DTD Version)," Document Release 1.0, <http://www.xcbl.org>.

[7] <http://www.rosettanel.org/>

[8] Sangho Ha and Kunsu Suh, "A XML Based Product Description Model", Int'l Conf. on Software Engineering, Artificial Intelligence, Networking & Parallel/Distributed Computing, 2001.

[9] St. Laurent and Cerami, Building XML Applications, Reading, McGrawHill, 1999.

[10] Noel Yuhanna, Oracle 8i: Database Administration, Reading, Manning, 2000.

[11] Oracle Inc. 1999 Oracle XML Developer's Kit, <http://technet.oracle.com/tech/xml/info/htdocs/xdk.htm/>



하 상 호

1988년 서울대학교 계산통계학과(학사)  
 1991년 서울대학교 계산통계학과(석사)  
 1995년 서울대학교 전산학과(박사)  
 1995년~1996년 한국전자통신연구원 박사후 연구원  
 1996년~1997년 MIT LCS 박사후 연구원  
 1997년~현재 순천향대학교 정보기술공학부 조교수  
 관심분야: 프로그래밍언어, e-commerce, ubiquitous computing.

교 신 저 자

하 상 호 336-745 충남 아산시 신창면 읍내리 산 53-1  
 순천향대학교 정보기술공학부