

# 객체 모델을 기반으로 한, XML 문서 검색을 위한 EJB 컴포넌트 개발

김용수, 주경수  
순천향대학교 전산학과  
e-mail:admin@kimys.pe.kr, gsoojoo@sch.ac.kr

## Developing EJB Component for Retrieval XML Documents based on Object Model

Kim Yong-Soo, Joo Kyung-Soo  
Dept of Computer Science Soonchunhyang University

### 요약

최근 소프트웨어 개발비용의 절감과 신뢰성 있는 소프트웨어의 개발을 위하여 컴포넌트 기반의 소프트웨어 개발에 대한 연구가 활발하게 진행되고 있다. 컴포넌트를 기반으로 한 소프트웨어 개발의 핵심은 컴포넌트의 품질이며, 컴포넌트의 품질을 결정짓는 가장 중요한 특성 중의 하나가 컴포넌트의 맞춤 용이성이다.

본 논문에서는 객체 모델을 기반으로 하여 관계형 데이터베이스의 데이터를 XML 문서로 변환을 위한 EJB 컴포넌트를 구현하였다. 이에 따라 사용자들은 관계형 데이터베이스 기반의 XML 응용 시스템을 개발하고자 할 때 본 검색 컴포넌트를 단순·조립함으로써, 시간단축 및 비용절감을 기대할 수 있다.

### 1. 서론

XML은 웹에서 구조화된 정보나 반-구조화된 정보를 교환하기 위해 표준 마크업 언어로 채택되어 가고 있다. 이러한 XML 문서의 광범위한 확산은 필연적으로 XML 정보의 효과적인 저장 관리 및 검색 문제가 발생하게 되었다. 따라서 현재 데이터베이스 학계 및 업계에서는 XML 정보를 데이터베이스에서 어떻게 원활하게 관리할 수 있는가에 대해 활발한 연구와 개발 작업이 진행 중에 있으며 이러한 결과로 다양한 표준들이 개발되고 있다. 본 논문에서는 관계형 데이터베이스에서 데이터를 검색하여 XML 문서로 변환할 수 있도록 하였다.

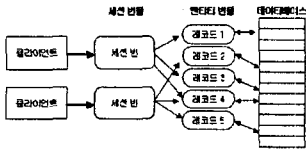
본 논문의 제2절에서는 EJB(Enterprise Java Bean)에 대하여 기술한다. 3절에서는 관계형 데이터베이스로부터 검색 방법을 설명한다. 4절에서는 EJB 컴포넌트 구현을 보도록 한다. 마지막으로 결론을 기술한다.

### 2. EJB 아키텍처

J2EE(The Java 2 Platform, Enterprise Edition)는 다계층 엔터프라이즈 애플리케이션을 개발하기 위한

표준을 말한다. 즉 J2EE는 표준화되고 모듈화 된 컴포넌트상에 다계층 엔터프라이즈 애플리케이션의 기본이 되고, 해당 컴포넌트에 완벽한 서비스를 제공한다. 또한 복잡한 프로그래밍 없이 자동으로 애플리케이션을 처리함으로써, 엔터프라이즈 애플리케이션을 단순하게 만든다[5]. J2EE는 단순성, 이동성, 확장성, 통합성 등의 특징을 지원하는 엔터프라이즈 솔루션을 위한 플랫폼이다. 또한 J2EE는 JSP, Servlet, Java Bean, EJB등의 집합이라고 말하며, 핵심적인 기술로는 EJB가 있다.

EJB는 컴포넌트 기반 분산 객체 기술로서 엔터프라이즈급 애플리케이션 개발에 있어 추상 데이터와 비즈니스 로직에 대한 부분을 담당하는 매우 중요한 핵심 기술을 가지고 있다. 또한 개발자를 도와주는 EJB 컨테이너는 자동으로 엔터프라이즈 빈(Enterprise Bean)의 생명주기 관리, 상태 정보관리, 보안, 트랜잭션 처리, 영속성 처리 등을 포함한 수많은 내재된 서비스를 제공해 주기 때문에 개발자는 비즈니스 로직만 담당하면 된다.

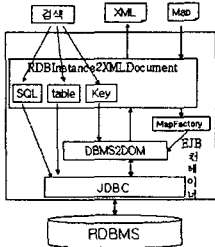


(그림 1) 세션 빈과 엔티티 빈의 관계

엔터프라이즈 빈이란 클라이언트가 호출하여 사용할 수 있는 EJB 컴포넌트를 의미한다. 이런 엔터프라이즈 빈은 세션 빈(Session Bean)과 엔티티 빈(Entity Bean)으로 구분되어 개발할 수 있다. 그림 1은 세션 빈과 엔티티 빈의 관계를 보여준다.

### 3. XML 문서 검색

XML 문서의 검색은 3가지 조건을 이용하여 검색할 수 있다. 각각은 Key 값에 의한 검색, SQL 문에 의한 검색, 데이터베이스 테이블 전체 검색으로 나누어진다. 그림 2는 XML 문서 검색 구조를 보여준다.



(그림 2) XML 문서 검색 구조

### 3.1 검색 방법

#### 3.1.1 Key 값에 의한 검색

그림 3의 Map 파일에서 “<ToRootTable>”으로 선언된 부분에서 테이블 이름을 book으로, 컬럼을 title로 선언하였다. 이것은 Key 값 검색의 테이블과 컬럼을 의미한다. DBMS2DOM에서 검색을 위한 SQL 문이 만들어지고 검색된 결과를 받아 Map 객체의 풀에 따라 XML 문서를 만든다.

```
<ToRootTable>
  <Table Name="book"/>
  <CandidateKey Generate="Yes">
    <Column Name="title"/>
  </CandidateKey>
  <OrderColumn Name="booksOrder" Generate="Yes"/>
</ToRootTable>
```

(그림 3) book.map 파일일부

#### 3.1.2 SQL 문과 테이블 검색

SQL 문에 의한 검색은 SQL 문에 의해 검색된 데이터를 DBMS2DOM에서 Map 객체의 풀에 따라 XML 문서를 만든다. 테이블 검색은 테이블의 모든 내용을 불러오는 select \* from tablename과 같은 결과를 가져와 테이블에 있는 모든 데이터를 XML 문서로 만든다.

### 3.2 MapFactory

MapFactory는 Map 객체를 만드는 곳으로 Map 안에 기술된 레지스트리 정보와 스키마 정보, 매핑을 등의 정보를 관리하면서, 매핑 풀에 대한 매핑 테이블을 생성하고, DBMS2DOM이 검색 값에 대한 SQL 문을 생성하는데 필요로 하는 여러 가지 정보를 제공한다.

```
<ClassMap>
  <ElementType Name="book"/>
  <ToClassTable>
    <Table Name="book"/> </ToClassTable>
  <PropertyMap>
    <Attribute Name="id"/>
    <ToColumn>
      <Column Name="id"/> </ToColumn>
    </PropertyMap>
  <PropertyMap>
    <ElementType Name="publication"/>
    <ToColumn>
      <Column Name="publication"/> </ToColumn>
    <OrderColumn Name="publicationOrder" Generate="Yes"/>
  </PropertyMap>
  <PropertyMap>
    <ElementType Name="ISBN"/>
    <ToColumn>
      <Column Name="ISBN"/> </ToColumn>
    <OrderColumn Name="ISBNOrder" Generate="Yes"/>
  </PropertyMap>
  <PropertyMap>
    <ElementType Name="title"/>
    <ToColumn>
      <Column Name="title"/> </ToColumn>
    <OrderColumn Name="titleOrder" Generate="Yes"/>
  </PropertyMap>
  <RelatedClass KeyInParentTable="Candidate">
    <ElementType Name="authors"/>
  </CandidateKey Generate="Yes"><Column Name="bookPK"/>
  </CandidateKey>
  <ForeignKey><Column Name="bookFK"/></ForeignKey>
  <OrderColumn Name="authorsOrder" Generate="Yes"/>
  </RelatedClass>
</ClassMap>
```

(그림 4) books.map 파일일부

그림 4의 Map 문서는 시스템이 자동으로 생성하기도 하며, 사용자 인터페이스를 통해서 나타나는 XML 문서의 구조를 표현한 DTD 트리와 테이블의 메타데이터 정보를 표현한 메타데이터 트리를 사용자가 보고 판단하여 관계 있는 것끼리 매핑을 시켜주는 방법으로 생성하게 된다[3][4].

### 3.3 DBMS2DOM

검색하여 얻어진 데이터는 MapFactory로부터 각 노드와 테이블간의 매핑 정보를 제공받아 XML 문서를 만든다.

#### 3.3.1 Key 값에 의한 SQL문 생성

검색 값으로 title에 있는 JAVA를 검색 할 경우 DBMS2DOM은 모든 테이블을 조인하여 데이터베이스에서 데이터를 검색한다. 그림 5는 DBMS2DOM에 의해 생성되는 SQL 문장이다.

```
select book.id, book.ISBN, book.title, book.publication,
       book.bookPK, authors.bookFK, authors.authorPK,
       author.author, author.authorFK
from book, authors, author
where book.bookPk = authors.bookFK and
       authors.authorPK = author.authorFK and title =
       'JAVA'
```

(그림 5) 검색 SQL 문장

#### 3.3.2 XML 문서 생성과정

- ① 검색한 데이터와 Map 객체의 매핑이 이루어진다.
- ② Map 객체를 통해 book 이라는 ElementType 이 book 테이블과 매핑 되고, id 라는 컬럼은 Attribute id, 그리고 나머지 컬럼들이 ElementType으로 된다는 것을 알 수 있다.
- ③ <RelatedClass KeyInParentTable="Candidate">에 의해 bookPK 는 자동 생성된 숫자를 가지고 authors 테이블과 연결이 된다는 것을 알 수 있다. 그러므로 authors는 DOM에서 자식을 가지는 노드가 된다. 그림 6은 테이블 구조를 보여 주고 있으면 그림 7은 검색한 결과를 XML 문서로 변환된 것이다.

book 테이블				
id	ISBN	title	publication	bookPK
11314	89-7467-8113-6	JAVA	삼각지	2

authors 테이블	
bookFK	authorPK
2	3

author 테이블	
author	authorPK
민계철	3

(그림 6) 저장된 테이블 형태

```
<book id="11314" >
  <ISBN> 89-7467-8113-6</ISBN>
  <title>JAVA를 이용한 3D 만들기</title>
  <authors>
    <author>민계철,</author>
  </authors>
  <publication> 삼각지</publication>
</book>
```

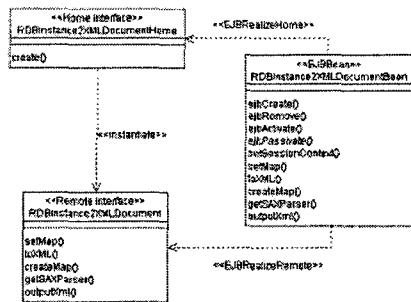
(그림 7) 검색하여 생성된 XML 문서

### 4. 구현

본 논문에서 구현한 컴포넌트명은 'RDBInstance2XMLDocument'라고 명명하였다. 본 컴포넌트는 XML 문서로 검색하기 위해, EJB1.0 기반으로 하여 세션 빈을 구현하였다. JBuilder4.0을 이용하여 컴포넌트를 구현하였고, 실행 환경으로서 EJB 서버인 IAS(Inprise Application Server)4.1에서 'RDBInstance2XMLDocument' 컴포넌트를 실행하였다

#### 4.1 컴포넌트 클래스 다이어그램

관계형 데이터베이스에서 검색을 위한 컴포넌트 클래스 다이어그램은 그림 8과 같다. 클라이언트 측에서 검색 쿼리와, 검색 조건에 해당하는 Map 문서를 EJB 컴포넌트에 보낸다. 컴포넌트는 클라이언트 쪽에서 넘겨준 값을 String으로 받게 되고 3절에서 설명한 방법을 이용하여 데이터베이스 테이블에서 검색을 하고 Map 객체의 규칙에 따라 XML 문서를 생성한다.



(그림 8) RDBInstance2XMLDocument 컴퍼넌트 클래스 다이어그램

#### 4.2 컴포넌트 인터페이스

##### 4.2.1 Home 인터페이스

EJB 세션 빈에 접근하려는 모든 클라이언트는 홈 인터페이스를 통한다. 즉, JNDI 이름을 이용하여 세션 빈을 찾아낼 때 홈 인터페이스를 이용한다.

(표 1) 홈 인터페이스

메소드	설명
create()	RDBInstance2XMLDocument 객체 생성

##### 4.2.2 Remote 인터페이스

클라이언트가 세션 빈의 비즈니스 로직을 사용하

기 위해서 리모트 인터페이스를 사용한다. 즉, 클라이언트가 홈 인터페이스의 create()를 호출하면 EJB 컨테이너는 리모트 인터페이스를 구현한 객체인 EJB 객체를 생성한다. 다음으로 EJB 객체의 비즈니스 메소드를 호출하면 동일한 이름을 가지는 메소드를 빈 인스턴스로부터 호출하여 서비스를 제공한다. 아래 표 2 에서는 리모트 인터페이스와 각각의 동작을 설명한다.

(표 2) 리모트 인터페이스

메소드	인자	설명
setMap(String)	Map	이 메소드는 컴포넌트 외부에서 읽은 Map 문서를 String 값으로 받아 임시 파일을 생성하는 메소드
toXMLK(String, String, String, String, Object[])	user, passwd, url, tableName, key	데이터베이스 접속정보, 테이블 네임과 검색 값을 인자로 받는다.
toXMLS(String, String, String, String)	user, passwd, url, select	데이터베이스 접속 정보, 검색 SQL 문장이 전달된다.
outputXml()		toXML에서 만들어진 문서 형태를 스트링 처리하여 만들어진 문서의 결과를 불수 있게 한다.

#### 4.2.3 EJB 빈 클래스

빈 클래스는 리모트 인터페이스에 선언된 메소드가 실제로 구현되어 있는 클래스이다. 또한 내부적으로 쓰이는 자바 빈을 호출한다. 표 3은 리모트 인터페이스에서 선언한 메소드 이외의 빈 클래스의 메소드를 나타내고 있다.

(표 3) 빈 클래스

메소드	인자	설명
createMap(InputSource, Connection)	src, conn	toXMLK()와 toXMLS()에서 호출되며, getSAXParser()을 호출 한다
getSAXParser()		createMap에서 호출되며, Map 문서를 파싱한다.

#### 4.2.4 세부 구현

'RDBInstance2XMLDocument' EJB 세션 빈은, 관계형 데이터베이스에서 검색된 값을 XML 문서로 변환하는 컴포넌트이다. 그림 2는 'RDBInstance2XMLDocument' EJB 컴포넌트의 구조를 보여준다. 클라이언트에서 직접 EJB 서버에 파일 전송이 불가능하므로 Map 문서를 String으로 전송한다. 검색방법에 따라 Key값, 테이블명, SQL 문장이 EJB 컴포넌트에 전달된다. 전달된 Map 문서와 검색 값의 처리 과정은 3절의 절차에 의해 이루어지며 사용된 핵심 자바 빈은 표 4와 같다. 본 컴포넌트에서 사용한 자바 빈은 XMLDBMS[2]에서 제공

해 주는 자바 클래스들을 이용하여 구현하였다. XMLDBMS 1.01은 XML 문서와 관계 데이터베이스 사이에 데이터를 전송하기 위한 자바 패키지이다.

(표 4)자바 빈 기능

자바 빈	기능
MapFactory	EJB 빈 클래스의 createMap()에서 호출되어 실질적인 Map 객체를 만드는 자바 빈이다.
DBMS2DOM	검색된 데이터를 Map 객체의 틀에 따라 XML 문서로 변환 하는 자바 빈이다.

#### 5. 결론

XML은 웹에서 구조화된 정보나 반-구조화된 정보를 교환하기 위한 표준 마크업 언어로 채택되어 가고 있다. 이러한 현실 속에서 많은 데이터베이스의 데이터를 검색하여 XML 문서로 변환이 필요하다. EJB는 J2EE 스펙의 핵심으로 엔터프라이즈 어플리케이션 개발에 가장 많이 사용되고 있다. 본 논문에서는 XML 문서로 검색을 위한 EJB 컴포넌트를 구현하였다. 구현한 XML 문서 검색 컴포넌트를 다른 컴포넌트들과 조합함으로써, 사용자들은 관계형 데이터베이스 기반의 XML 응용 시스템을 개발하고자 할 때 시간단축 및 비용절감을 기대할 수 있다.

#### 참고문헌

- [1] "Extensible Markup Language(XML)", <http://www.w3.org/TR/PR-XML-971208>
- [2] Ronald Bourret, XML-DBMS, <http://www.informatik.tu-darmstadt.de/DVS1/staff/bourret/xmldbms/readme.html>, 2000
- [3] 이상태, 이정수, 주경수, "객체모델을 기반으로 한 XML DTD의 RDB 스키마로의 변환 방법", 대한전자공학회, 하계종합논문대회, 제24권 제1호, pp.113-116
- [4] 이상태, 이정수 주경수, "XML DTD를 기반으로 한, RDB 스키마 설계를 위한 Component 구현", 한국정보처리학회 지식 및 데이터공학연구회 제8회 학술발표대회 논문집, pp.309-316, 2001
- [5] 박지훈, 자바 개발자를 위한 EJB 최신 입문서 Enterprise JavaBeans, 대칭미디어, PP20-28