

음반 B2B를 위한 ebXML 등록기 및 저장소의 설계 및 구현⁺

김 주 성* 김 유 성**

*인하대학교 정보통신대학원

**인하대학교 정보통신공학부

c2031004@inhavision.inha.ac.kr yskim@inha.ac.kr

Design and Implementation of ebXML Registry & Repository for B2B e-Commerce of Music Records

Joo-Sung Kim* Yoo-Sung Kim**

*Graduate School of Information Technology and Telecommunication, Inha University

**School of Information and Communication Engineering, Inha University

요 약

음악 상품의 검색, 주문(계약), 대금결제, 배송 등에 있어서 기업들은 자신의 독자적인 비즈니스 방식과 거래 시스템을 구축하여 운영하기 때문에 기업과 기업간(B2B) 음악 상품의 전자 상거래에는 많은 어려움이 따른다. e비즈니스 표준 프레임워크인 ebXML은 기업의 전자 상거래를 위한 비즈니스 정보를 교환할 때 확장성 표기 언어를 적용해 기업간 시스템간의 상호 운용을 가능하게 하지만, 음반 산업 분야의 적용은 미흡한 실정이다. 본 논문에서는 음악 B2B를 위해 ebXML 등록기 및 저장소를 설계 및 구현하였다. 본 논문에서 설계, 구현한 등록기는 거래 당사자인 기업간에 음악 상품 및 기업의 거래 관련 정보를 공유하는 서비스를 제공하며, 저장소는 실세계의 기업간 음악 거래 정보 및 음악 거래에 사용되는 개체간의 연관성 정보를 저장하고 있다.

1. 서론

전자상거래는 초기 실물 중심의 거래에서 게임, 음악, 영화와 같은 디지털 상품 분야로 그 영역을 넓혀가며 성장을 지속하고 있다. 기존의 음악 상품 거래에서는 기업들이 독자적인 비즈니스 방식과 각각의 거래 시스템을 구축, 운영하였기 때문에 전자상거래를 위해 기업간 협업 및 상호 운영성에 있어서 많은 어려움이 따랐다. 이를 해결하기 위해 e비즈니스 표준 프레임워크인 ebXML이 UN/CEFACT 및 OASIS를 중심으로 제정되었다[1].

ebXML의 등록기 및 저장소(Registry / Repository)는 기업 거래를 위해서 기업서비스와 관련된 정보를 저장 관리하는 필수적인 요소로 제안되었지만, 국내 음반 산업에서 증가되고 있는 음악 B2B를 위한 ebXML 등록기/저장소의 설계 및 구현은 전무하다. 본 논문에서는 음반 B2B 전자상거래를 위한 ebXML 등록기 및 저장소를 설계하고 프로토타입 시스템을 구현하였다. 등록기는 거래 당사자

인 기업간에 음악 상품 및 기업의 거래 관련 정보를 공유하는 서비스를 제공하며, 저장소는 실세계의 기업간 음악 거래 정보 및 음악 거래에 사용되는 개체간의 연관성 정보를 저장하고 있다. 개발된 음반 B2B를 위한 ebXML 등록기 및 저장소를 활용하면, 서로 다른 기업들의 시스템들이 상호 운용되어 기업간 음악 상품의 전자 상거래가 효율적으로 이루어질 수 있다.

본 논문의 구성은 다음과 같다. 제 2장에서는 B2B를 위한 ebXML 등록기 및 저장소가 가져야 할 요구 사항 및 구조를 알아본다. 제 3장에서는 본 논문이 제안한 음반 B2B 전자상거래를 위한 ebXML 기반 등록기 및 저장소를 설계하고, 해당 시스템의 구현에 대해 기술한다. 마지막으로, 제 4장에서는 결론 및 향후 연구 과제를 제시한다.

2. 음반 B2B를 위한 ebXML 기반의 등록기 및 저장소

국내 음반 산업의 유통구조는 제작사-도매상-소매상-소비자등의 다단계 유통구조이고 도매상의 입지

⁺ 본 연구는 한국과학재단 지정 인천대학교 동북아 전자물류연구센터의 지원에 의한 것임.

가 강하다는 것이다. 이는 도매상이 음반시장에 막대한 영향을 미친다는 것이다. 제작사-도매상 간 도매상-소매상 간의 음반B2B 상거래가 이루어지는데 팩스와 전화 등으로 거래가 되고 있어 자료가 전산화 되어 있지 않고 대형 도매상들의 자체 DB에 구축되기 때문에 정확한 집계가 이루어지기 힘든 구조로 되어있다[2]. 따라서 ebXML 등록기/저장소와 같은 시스템을 구축하여 제작사-도매상-소매상 또는 제작사-소매상과 같이 투명한 유통구조를 보장할수 있어야 한다.

음악 B2B를 위해 UN/CEFACT 및 OASIS를 중심으로 제정하고 있는 XML기반의 국제 표준 프레임워크인 ebXML은 활용하여 효과적인 정보 교환 및 처리가 가능하다. 특히 ebXML의 등록기 및 저장소(Registry / Repository)는 기업 거래를 위해서 기업 서비스와 관련된 정보를 저장 관리하는 필수적인 요소이다. 현재 ebXML 등록기 및 저장소에 관한 구현이 버전 2.0에 대해서 KNET의 XENI Registry V2.0, 이노디지탈의 SOLME 등에서 빠른 속도로 개발되고 있으며, 특히 한국전자거래진흥원(KIEC)에서 운영하는 REMKO(Regisry & Repository of ebXML in KOREA)라는 한국 중앙등록저장소 2.1버전으로 업그레이드 하였고 2003년 12월에는 ebXML Spec 2.5 버전의 Federation 기능을 구축하여 시범 운영하고 있다[3].

그러나 아직까지 특정 산업분야에 적용한 등록기 및 저장소의 개발은 미흡한 실정이다. ebXML 기반 등록기 및 저장소는 실제 기업간 거래 환경에서 표준이 되는 조직에 의해 중심으로 관리되는 단일의 중앙 등록기 및 저장소의 형식이 아니며, 국가나 산업체 등의 조직단위의 분산된 등록기/저장소의 형식을 가지고 등록기/저장소 상호간에 연동을 하여야 한다. 따라서 특정 산업 분야에 적용한 등록저장소의 연구를 통하여 ebXML 등록기/저장소의 검증이 필요하다.

음반 B2B 전자상거래를 위한 B2B기반 등록기 및 저장소는 기업간의 ebXML 기반 음반거래에 있어서 발생하는 모든 정보의 공유를 가능하게 하는 Registry(등록기) 서비스를 제공하며, 이 서비스를 위한 모든 객체의 정보들을 Repository(저장소)에 저장하고 관리하는 시스템으로 정의한다.

ebXML 기반 등록기/저장소는 Registry, Repository, Registry Client Interface, Registry Web Application 으로 구성되어 있다. Registry는 거래 당사자들이 등록한 정보에 대한 메타 정보를 저장하고 이에 대한 인터페이스를 제공하는 구성 요소이다. Repository는 CPP(Collaboration Protocol Profile), CPA(Collaboration Protocol Agreement), BP(Business Process), CC(Core Component) 및 기타 RegistryObject를 저장하고 관리하는 구성요소이다. 위의 두 요소 Registry와 Repository를 이용하여 기업간 거래에 필요한 모든 정보들을 등록하고, 검색할 수 있는 기능을 제공하여 거래당사자간에 상호작용을 돕는다. Registry / Repository를 사용하는

클라이언트를 위해 웹 기반의 Registry Web Application, JAVA 프로그램에서 레지스트리를 사용하기 위한 Java API(JAXP, JAXR)인 클라이언트 인터페이스가 있다.

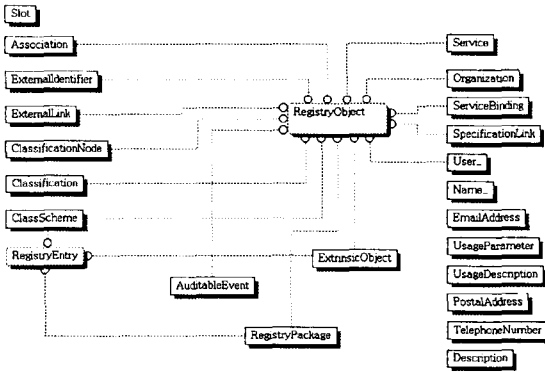
Registry는 등록기 정보 모델(RIM: Registry Information Model)과 등록기 서비스(RS: Registry Service)로 나눌 수가 있다. 등록기 서비스는 다시 LCM(Life Cycle Management) 서비스와 QM(Query Management)서비스를 제공해야 한다. LCM 서비스는 등록기에 객체를 저장, 삭제하는 연산과 같이 객체들의 생명주기를 관리하고, QM 서비스는 등록기에 저장된 정보들에 대해서 단순한 형태의 Filter Query와 복잡한 질의를 제공하는 SQL Query와 같은 질의 방법을 제공한다. 저장소에 저장된 객체의 생명주기를 관리하는 LCM은 7가지 서비스로 이루어진다. SubmitObjects 서비스는 저장소에 정보를 저장하고, UpdateObjects 서비스는 저장소에 저장된 정보들을 업데이트하며, RemoveObjects 서비스는 저장소에 저장된 정보들을 삭제하는 역할을 한다. ApproveObjects 서비스와 DeprecateObjects 서비스는 저장소에 저장된 정보들을 승인하거나 deprecate한다. Addslots 서비스는 객체의 표준에서 정의되지 않는 정보들을 추가하게 되고, Removeslot 서비스는 추가한 Slot을 삭제한다.

등록기에 저장된 정보들에 대한 검색을 관리하는 QM은 3가지의 질의 서비스를 제공한다. 가장 기본이 되는 서비스로서 등록기 정보 모델에서 정의된 구조 정보를 기반으로 XML 구문을 이용해서 메타 데이터를 검색할 수 있는 Filter Query 서비스, ebRIM에서 제공하는 관계 스키마를 기반으로 SQL 을 사용하여 복잡한 질의가 가능하도록 하는 선택적 서비스인 SQL Query 서비스, 등록기에 저장되어있는 객체들을 가져올 수 있는 질의 서비스로서 모든 질의의 검색 결과 내에서 특정 원하는 세부정보를 검색할 수 있는 Content Retrieval 서비스로 구성되어 있다

3. 음반 B2B를 위한 ebXML 기반의 등록기 및 저장소의 설계 및 구현

음반B2B를 위한 등록기/저장소는 버전 2.1을 기반으로 시스템을 설계하였으며 RIM과 RS 두 부분으로 나누어서 설계하였다.

ebXML RIM([5])은 어떠한 형태의 정보들(CPP, CPA, XML Schema, Business Process 등)이 어떤 형식으로 등록기에 저장되고 관리되는지 정의한다. 이러한 정보들이 등록기에 어떤 클래스들을 포함시킬 것인지, 선택한 클래스들이 어떤 속성과 어떤 메소드를 가질 것인지 결정하기 위하여 [그림 1]과 같이 RIM을 설계하였다.



[그림 1] RIM 2.1을 RDMS로 설계한 ER-Diagram

테이블(또는 뷰)의 기능을 다음과 같이 정의하였다.

- **RegistryObject**: RIM에서 다른 클래스에 대한 추상클래스로 가장 기본이 되는 공통적인 부분들이 정의. 이 클래스에는 등록기 객체들에게 공통적으로 사용되는 최소한의 메타 정보를 가지고 있다. 또한, 등록기 객체에 대하여 추가적인 메타 정보를 제공하는 관련된 객체들에 대하여 접근할 수 있는 메소드들을 제공한다.
- **Slot**: 등록기 정보 모델에 확장성을 제공하는 수단으로서 등록기가 어느 정도는 자유롭게 고정된 정보 모델을 확장할 수 있는 동적인 기능을 제공하는 중요한 클래스. 콘텐츠에 대한 메타 정보를 저장하는 "RegistryObject"에서 이미 정의해둔 속성들 이외에 환경에 따라 추가적인 정보인 메타 정보를 저장하고 관리하고자 할 경우에 활용할 수 있는 클래스이다. 즉, "Slot"을 활용하여 필요한 속성들을 각각의 RegistryObject"에 추가할 수 있다.
- **Association**: "Association"을 통하여 등록기 정보 모델 안의 각 정보간의 유기적인 다대다 (Many-to-Many) 연관 관계들을 형성하고 이에 대한 정보들을 관리할 수 있는 "RegistryObject"이다.
- **ExternalIdentifier**: "RegistryObject"가 가지고 있는 고유 식별자인 UUID 이외에 추가적인 식별자를 활용하기 위하여 사용된다. "RegistryObject"에 부여되는 UUID는 시스템이 생성하는 고유 식별자로서 그 자체에 어떤 의미가 포함되어 있지는 않다. 다만 메타 정보들에 고유성을 부여할 뿐이다. 그러므로 어떤 의미를 가진 식별자를 부여하여 메타 정보를 검색할 수 있는 기능을 제공하기 위해서는 "ExternalIdentifier"를 활용하여 그러한 기능들을 지원할 수 있다.
- **ExternalLink**: 등록기 외부에 저장되어 있는 콘텐츠의 접근 경로와 식별자(URI)를 표현하기 위하여 사용되는 "RegistryObject"이다. 즉, 실제 콘텐츠는 자신의 홈페이지의 특정 위치에 저장해두고 그에 대한 메타 정보만을 등록기에 등록하여 다른 콘텐츠와 마찬가지로 등록기를 통하여 공유될 수 있도록 하기 위하여 사용된다.

- **ClassificationScheme**: "RegistryObject" 객체들을 체계적으로 분류하여 활용하기 위하여 정의된 "RegistryEntry"이다. 활용하고자 하는 분류체계가 등록기 내부에서 정의될 수도 있고 외부에서 정의된 것을 활용할 수도 있다. 등록기에서 분류체계를 정의하는 최상위 노드로서 실제생활에서 활용하고 있는 다양한 분류체계들을 정의할 수 있다.
- **ClassificationNode**: "ClassificationScheme" 아래의 트리구조를 정의하기 위하여 활용되는 "RegistryObject"이다. 이렇게 구성된 트리는 각각의 노드가 "ClassificationNode"이고 루트노드가 "ClassificationScheme"이다. "ClassificationNode"들로 구성된 트리 구조의 분류 트리들은 산업별, 지역별, 표준분류 체계 등을 정의하기 위하여 활용된다. 즉, 트리 구조를 구성하는 노드를 대표하는 "ClassificationNode"는 등록기의 메타 정보들을 더욱 효율적이고 계층적으로 분류하기 위한 구조를 정의하기 위하여 사용된다.
- **Classification**: 특정 기준에 의해 체계적으로 분류해 놓은 분류 체계의 트리구조에 따라 콘텐츠의 메타 정보에 연관 관계를 설정해 주는 역할을 하는 "RegistryObject"이다. 즉, "ClassificationNode"와 콘텐츠의 메타 정보를 담고 있는 "RegistryObject" 사이의 연관 관계를 설정해준다. 분류체계가 등록기 내부에 정의되어 있는지 외부에 정의된 것을 사용하는지에 따라서 "Classification"도 내부의 객체와 설정된 관계인지 외부의 객체에 의존적인지가 결정된다.
- **RegistryPackage**: 논리적 연관성이 있는 다양한 "RegistryObject"들을 하나의 패키지로 만들어 그룹화하는 역할을 하는 "RegistryEntry"이다.
- **AuditableEvent**: 등록기에 저장된 객체인 "RegistryObject"들에 대한 감사 추적을 제공하기 위하여 사용되는 "RegistryObject"이다.
- **User**: 정식으로 등록된 등록기의 사용자들에 대한 정보를 제공하기 위하여 활용되는 "RegistryObject"이다.
- **PostalAddress**: 우편배달이 가능한 주소에 대한 내용을 정의하는데 사용하는 재사용 가능한 간단한 엔티티 클래스이다.
- **EmailAddress**: email 주소에 대한 내용을 정의하는데 사용하는 재사용 가능한 간단한 엔티티 클래스이다.
- **Organization**: 등록기의 사용자로서 실제 콘텐츠를 등록하는 제출 기관과 같은 조직에 대한 정보를 제공하기 위하여 활용되는 "RegistryObject"이다.
- **Service**: 서비스들(예, 웹서비스)에 대한 정보를 제공하기 위하여 활용되는 "RegistryEntry"이다.
- **ServiceBinding**: "Service"에 대하여 제공되는 기술적인 접근경로에 대한 정보를 제공하기 위하여 활용되는 "RegistryObject"이다.
- **SpecificationLink**: "ServiceBinding"과 이를 어떻게 사용하는지에 대한 상세한 정보를 제공하는 기술 지침서들을 연결하기 위하여 매개체로서 사용된다.

ebXML RS([4])은 객체를 기본적으로 관리하기 위하여 설계된 인터페이스(LM)와 등록기에 관련된 질의를 처리하기 위한 인터페이스(QM)로 구성되어 있다.

• LifecycleManager 인터페이스

객체 생명주기 관리 기능을 구현한 등록기 인터페이스이다. 각 메소드들은 등록기 클라이언트에 의해 호출된다. 예를 들어 클라이언트는 객체를 제출(submit), 분류(classify), 연관관계 생성(associate), 사용 중지(deprecate), 삭제(remove)하기 위해 이 인터페이스를 사용할 수 있다. 제출, 분류, 연관관계 생성, 사용 중지, 삭제의 의미는 등록기 정보 모델에 정의되어 있다. [표 1]에 이러한 메소드를 구현하였다.

[표 1] LifecycleManager의 메소드 리스트

메소드	설명
approveObjects (ApproveObjectsRequest req)	하나 또는 그 이상의 제출된 객체들을 승인한다.
deprecateObjects (DeprecateObjectsRequest req)	하나 또는 그 이상의 제출된 객체들을 사용 중지한다.
removeObjects (RemoveObjectsRequest req)	하나 또는 그 이상의 제출된 객체들을 등록기에서 삭제한다.
submitObjects (SubmitObjectsRequest req)	하나 또는 그 이상의 제출된 객체들과 Association이나 Classification과 같은 연관된 메타데이터를 제출한다.
updateObjects (UpdateObjectsRequest req)	하나 또는 그 이상의 제출된 객체들을 갱신한다.
addSlots (AddSlotsRequest req)	하나 또는 그 이상의 등록기 엔트리에 슬롯(Slot)을 추가한다.
removeSlots (RemoveSlotsRequest req)	하나 또는 그 이상이 등록기 엔트리에서 슬롯(Slot)을 삭제한다.

• QueryManager 인터페이스

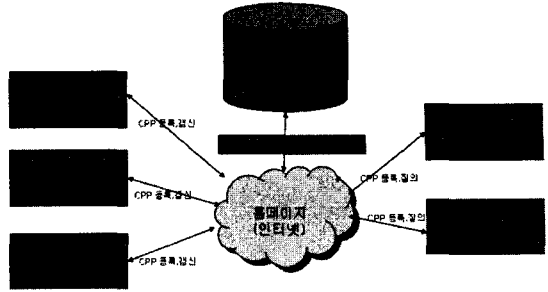
객체질의 관리 서비스를 구현한 등록기 인터페이스이다. 각 메소드들은 등록기 클라이언트에 의해 호출된다. 예를 들어 클라이언트는 등록기 콘텐츠에 대해 브라우즈 드릴 다운 질의나 에드혹 질의를 하기 위해 이 인터페이스를 사용한다.

[표 2] QueryManager의 메소드 리스트

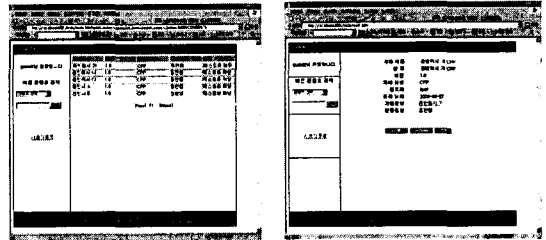
결과값 형식	메소드	설명
RegistryReponse	submitAdhocQuery (AdhocQueryRequest req)	에드혹 질의 요청메시지를 제출

위의 설계를 바탕으로 자바 API인 JAXM, JAXR 을 사용하여 구현하였고, 웹 인터페이스는 tomcat4.1 을 이용하여 soap 바탕으로 하여 제작하였다.

음악 B2B의 전자 상거래시 이용되는 클라이언트 형태는 모든 사용자가 인터넷환경이면 거래 가능한 Thin 클라이언트 서비스 형태로 개발하였다. 이 방식은 모든 사용자가 특별한 어플리케이션을 설치하지 않아도 되는 방식이므로 영세한 음반사도 이용할 수 있다. [그림 2]는 thin 클라이언트 구현한 화면이다.



[그림 (a)] 음반B2B의 thin 클라이언트 시나리오



[그림 (b)] 개체 선택 이전 [그림 (c)] 개체 선택 이후

[그림 2] 음악 B2B의 thin 클라이언트 구현화면

4. 결론 및 향후 연구

본 논문은 음반 B2B 전자상거래 시스템을 위해 ebXML기반 Registry & Repository 프로토 타입을 구축하였다. 이로 인해 거래 당사자 간에 개방적이고 유연한 형태의 전자 상거래를 가능하게 할 수 있다. 따라서 음반 거래를 위한 온라인 시장의 활성화 를 주도할 수 있다.

향후 연구로는 등록기 / 저장소 프로토타입의 안정화 및 유지보수를 할 것이고, 검색을 위해 내용이 반 음악 정보 검색 인터페이스를 추가할 할 예정이다. 또한 음반 B2C전자상거래 시장의 화두인 MP3 과 같은 디지털음원을 거래하기 위한 DRM 시스템 도입도 연구할 예정이다.

참고문헌

[1] ebXML, <http://www.ebXML.org>
 [2] 김종휴, "한국 음반 산업 연구", 삼성경제연구소, 1997
 [3] REMKO, <http://www.remko.or.kr>
 [4] ebXML Registry Project Team, "Registry Information Model v2.1, http://www.oasis-open.org/committees/regrep/documents/2.1/specs/ebxml_v2.1.pdf, 2002
 [5] ebXML Registry Project Team, "Registry Service v2.1, <http://www.oasis-open.org/committees/regrep/documents/2.1/specs/ebrs.pdf>, 2002