

ebXML등록 및 저장소에서 XQuery 지원

곽기욱*, 이원근*, 이상돈*, 최한석*
*목포대학교 멀티미디어공학과

e-mail : ja-33@hosanna.net, wglee2000@hotmail.com

XQuery Support for ebXML Registry and Repository

Ki Wook Kwag*, Won Gun Lee*, Sang Don Lee*, Han Suk Choi*
*Dept. of Multimedia Engineering, Mokpo University

요 약

ebXML 등록 및 저장소는 ebXML 프레임워크의 핵심부로 거래에 필요한 콘텐츠를 등록, 저장, 검색하는 기능을 제공한다. 거래 파트너간 상호 협업을 위해서는 등록된 기업 정보를 검색하는 것이 필수적이다. 현재 표준인 ebXML Registry Service 2.1 명세에서는 Filter Query와 SQL Query에 기반한 검색을 지원하고 있다. 본 논문에서는 ebXML 등록 및 저장소에서 XQuery에 기반한 검색지원 시스템을 제안한다. 이를 위해서 주요 검색 질의를 XQuery로 지원하는 과정을 기술하고, 이를 지원하기 위한 시스템을 설계한다.

1. 서론

전자 상거래는 기업, 정부기관과 같은 독립된 조직 혹은 조직과 개인간에 다양한 전자적인 매체를 이용하여 상품이나 서비스를 교환하는 방식의 의미로 사용되고 있다. 전자상거래의 시초로 EDI(Electronic Data Interchange)가 있다[5]. EDI는 비즈니스에 있어서 문서와 정보를 교환하는데 매우 효율적인 해결책을 제시해 줄 것으로 예상되었으나, 거래하는 업체가 늘어날 수록 거래업체마다 구축 및 통신에 따른 비용, 확장의 한계에 부딪히게 되었다.

XML[4]은 이러한 문제를 간단하게 풀어준다. XML은 인터넷 상에서 언어를 통일시키고 모든 기업이 간단하고 저렴하게 문서를 교환할 수 있다. XML이 가지는 확장성은 유연성을 제공하는데, 이러한 유연함은 전자상거래에 있어 상호연동을 극대화하기 위한 표준화와는 배치되는 것이다. 수십 개의 표준화 작업이 진행되고, 각각의 나름대로 서로 다른 프레임워크를 정의한다면, 전자상거래가 활성화 되기는 커녕 퇴화될 수도 있다.

이에 XML기반 단일 표준화를 추진할 필요성을 인식하고 1999년 11월 국제 EDI 추진기구인 UN/CEFACT가 IT 민간 컨소시엄인 OASIS(Organization for the Advancement of Structured Information Standards)와 공동으로 ebXML(e-business XML)이라는 차세대 인터넷 전자상거래 표준 프레임워크를 제정하기에 이르렀다[5].

ebXML에서 핵심요소중 하나는 ebXML Registry and Repository[1,2]이다. 이는 B2B 거래에 필요한 콘텐츠를 등록, 저장, 검색할 수 있도록 하며 이를 이용해 기업간 거래를 성사시킬 수 있는 핵심적인 역할을 한다.[5]

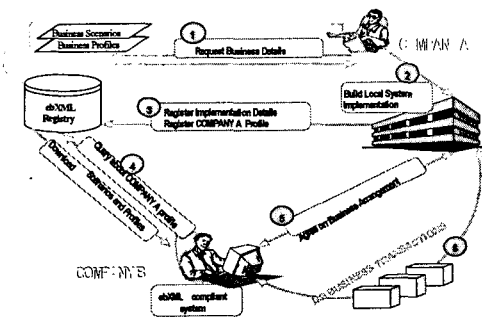
본 논문에서는 ebXML Registry and Repository에서 검색을 위해 지원되는 기존 질의 방식을 확장하여 XQuery[3]를 이용한 새로운 검색 기능을 제공하는 방안을 제시하고자 한다.

2장에서 ebXML 프레임워크와 ebXML Registry and Repository의 역할 및 XQuery의 주요 기능에 대하여 기술한다. 3장에서는 검색을 위해 XQuery를 도입할 때 고려사항 및 기존의 검색질의 방식과 비교하고 4장에서는 XQuery를 적용하기 위한 시스템의 설계 및 구현에 대하여 설명한다. 그리고 5장에서는 결론을 맺는다.

2. 관련연구

2.1 ebXML 시스템

ebXML은 기업간 거래를 위해 XML에 기반하여 표준 프레임워크를 제공한다. (그림 1)은 ebXML을 기반으로 전자상거래를 하고자 하는 두 거래 파트너들 간의 비즈니스 처리 과정을 보여 준다.

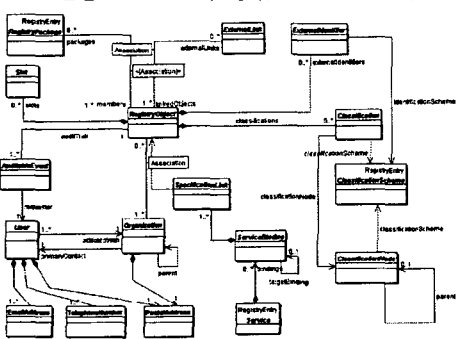


(그림 1) ebXML 수행과정

1~3단계까지는 회사 A가 거래를 하기 위해 필요한 사전 준비 과정이다. ebXML을 사용하기 위해 관련 명세정보를 검색하고 ebXML을 따르는 시스템을 개발하거나 구입하여 자신의 시스템에 연계하여 설치하고 회사 A는 자신의 회사와 시스템 정보를 등록한다. 4~5단계는 회사 B가 ebXML을 통해 회사 A의 정보를 검색하여 확인한 후, 회사 A를 파트너로 결정하여 거래를 요청하는 과정이다. 마지막으로, 6단계는 두 회사가 거래를 수행해 거래 데이터를 교환하는 과정이다.

2.2 ebXML Registry and Repository

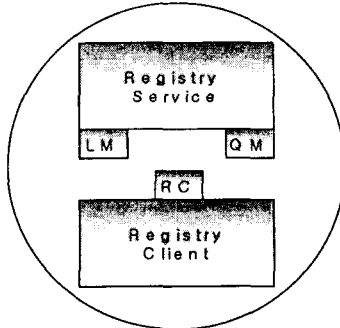
ebXML Registry and Repository의 목표는 전자상거래에 필요한 문서나 정보들을 등록하고 사용할 수 있게 서비스하는 것이다. ebXML에서는 등록 저장소의 데이터 구조와 등록/조회/수정/삭제 기능의 표준을 제시하고 있다. ebXML Registry and Repository의 표준 데이터 구조인 등록기 정보 모델(Registry Information Model)[1]의 핵심은 RegistryObject로서 등록 저장소에 제출되는 모든 등록 객체에 대한 최소한의 메타 정보를 갖고 있다[1,5]. (그림 2)는 ebXML Registry and Repository의 정보 모델을 UML 표기 방법으로 나타낸 것이다[1].



(그림 2) RIM2.1(Registry Information Model)

ebXML Registry 구조는 Registry Service와 Registry Client로 구성되어 있다. Registry Service는 Repository를 관리하기 위한 메소드를 제공한다.

Registry Service는 Life Cycle Management 인터페이스와 Query Management 인터페이스로 구성된다. Registry Client는 레지스트리에 접근하기 위한 응용이다. (그림 3)에서는 ebXML Registry 구조를 나타내고 있다.



(그림 3) ebXML Registry Service 구조

2.3 XQuery

XQuery는 XML 문서를 위한 질의 언어이다. 그동안 XML에 대한 많은 질의 방식이 제안되었으나 XQuery로 통합되어 가고 있는 추세이며, 현재 워킹 드래프트 XQuery 1.0 이 발표되었다.

XQuery의 주요 특징은 구조화된 문서, 반구조화된 문서, 관계형 데이터베이스, 객체 저장소등 여러 형식에 적용할 수 있으며, 스펙이 간단하고 적용이 용이하다. 또한 XPath, SQL과 같은 표준 기술을 적용하여 Syntax를 쉽게 이해할 수 있으며, XML 기반의 질의 형식을 갖는다[10].

기능어인 XQuery의 FLWR 표현식은 데이터를 가공하여 이들 데이터 변수를 이용하여 새로운 구조의 문서를 생성한다[3,10].

FLWR 표현식의 각 항목은 다음과 같다[3,10].

- * FOR : 조건에 맞는 여러 값을 얻어내어 여러 변수에 대입한다.
- * LET : 하나의 값을 추출하여 변수에 대입한다. FOR 문은 조건에 맞는 여러 값을 얻어 내는 반면, LET은 하나의 값을 얻어 낸다.
- * WHERE : FOR나 LET의 대상이 되는 값들 중에서 세부 조건을 제시한다.
- * RETURN : 얻어낸 정보를 이용하여 결과트리를 생성한다.

3. XQuery를 이용한 ebXML 저장소의 검색

3.1 기존 검색 방식

ebXML에서 제공되는 기존 질의 방식으로는 크게 Drill Down방식과 Ad Hoc방식이 있다. Drill Down방식은 각 분류 방법에 따라 저장되어 있는 객체를 윈도우 탐색 창에서 디렉토리를 검색하는 것과 같은 방법으로 객체를 찾아나가는 방식이다[5]. Ad Hoc 방식에는 Filter Query와 SQL Query가 있다.

Filter Query는 Registry를 구현할 때 반드시 지원되어야 한다. SQL Query는 추가적인 형태로 지원이 된다. XQuery1.0은 아직 표준화가 되지 않은 상태로서 ebXML 저장소에서 지원되지 않고 있으나, XQuery의 표준화의 진행에 따라 추가적인 검색 기능으로서 지원이 논의되고 있다[2,5].

3.2 XQuery에 기반한 검색 지원

ebXML 저장소 검색을 위해 XQuery1.0[3]을 적용하는데 있어 다음과 같은 조건이 요구된다. 첫째로, XQuery1.0은 RIM2.1을 기반으로 각 클래스가 바인딩이 되어야 한다. 기존 질의 방식인 Filter Query나 SQL Query도 RIM2.1[1]의 데이터 구조를 기반으로 이루어져 있다. 둘째로, RIM2.1이 확장된다면 XQuery1.0로 질의하는 대상 또한 확장 지원되어야 한다.

(1) 검색방법 및 제공 질의 유형

RIM2.1에서 Filter Query의 질의 대상은 RegistryObject를 비롯한 11가지이며, SQL 질의는 10가지이다. XQuery은 RIM2.1에 존재하는 모든 객체에 대하여 질의할 수 있도록 본 논문에서는 XQuery 질의 대상을 12가지로 설정하였다. 각 질의 방법에 따른 질의 대상을 정리하여 <표 1>에 나타내었다.

<표 1> 각 질의별 질의 대상 비교

RIM Class	Filter Query	SQL	XQuery
RegistryObject	○	○	○
RegistryEntry	○	○	○
Slot			
ExtrinsicObject	○	○	○
RegistryPackage	○	○	○
ExternalIdentifier			
ExternalLink		○	○
AuditableEvent(RegistryAuditTrail)	○	○	○
User(RegistryAuditTrail)			
Organization(RegistryAuditTrail)	○		○
PostalAddress(RegistryAuditTrail)			
TelephoneNumber(RegistryAuditTrail)			
EmailAddress(RegistryAuditTrail)			
PersonName(RegistryAuditTrail)			
Service(RegistryAuditTrail)	○		○
ServiceBinding(RegistryAuditTrail)			
SpecificationLink(RegistryAuditTrail)			
Association	○	○	○
Classification	○	○	○
ClassificationScheme	○	○	○
ClassificationNode	○	○	○
질의 대상 수	11	10	12

그 중에서 RIM2.1에서 상위 클래스에 해당하며 실제 데이터에 대해 필수적으로 보여주는 메타데이터인 Registry Object에 대한 질의 예제를 (그림 4)에

서는 Filter Query 예제, (그림 5)에서는 SQL Query 예제, (그림 6)에서는 XQuery 예제를 보여준다.

```

<AdhocQuery/Request>
  <ResponseOption returnType = "RegistryEntry"/>
  <FilterQuery>
    <RegistryObjectQuery>
      <ClassifiedByBranch>
        <ClassificationFilter>
          <Clause>
            <SimpleClause leftArgument = "path">
              <StringClause stringPredicate =
                "Equal">//Automotive/</StringClause>
            </SimpleClause>
          </Clause>
        </ClassificationFilter>
        <ClassificationSchemeQuery>
          <NameBranch>
            <LocalizedStringFilter>
              <Clause>
                <SimpleClause leftArgument = "value">
                  <StringClause stringPredicate =
                    "Equal">urn:ebxml:cs:industry</StringClause>
                </SimpleClause>
              </Clause>
            </LocalizedStringFilter>
          </NameBranch>
        </ClassificationSchemeQuery>
      </ClassifiedByBranch>
    </RegistryObjectQuery>
  </FilterQuery>
</AdhocQuery/Request>
    
```

(그림 4) FilterQuery 예제

```

SELECT r.id from RegistryObject r, Name nm, Description d where nm.value
LIKE '%Acme%' AND d.value LIKE '%bicycle%' AND r.id = nm.parent AND
raid = d.parent;
    
```

(그림 5) SQL Query 예제

```

For $ RO IN RegistryObject
WHERE $RO/Classification/ClassificationScheme="Industry" AND
$RO/Classification/ClassificationNode="Automotive" AND
$RO/Classification/ClassificationScheme="Geography" AND
$RO/Classification/ClassificationNode="Asia/Korea"
Return $RO
    
```

(그림 6) XQuery 예제

(2) 각 검색방법의 비교 분석

Filter 질의는 XML로 기술되며, 결과값을 ObjectRef, RegistryObject, RegistryEntry, LeafClass, LeafClassWithRepositoryItem과 같이 다양한 형태로 반환할 수 있으나, Syntax 작성이 복잡하다. 또한 필터질의는 Repository의 성격에 따라 해당 Repository에서 사용 가능한 질의 언어로 변환하는 과정이 필요하다. 예로, Repository가 RDBMS 라면 필터 질의를 SQL 질의로 변환하는 과정이 필요하다.

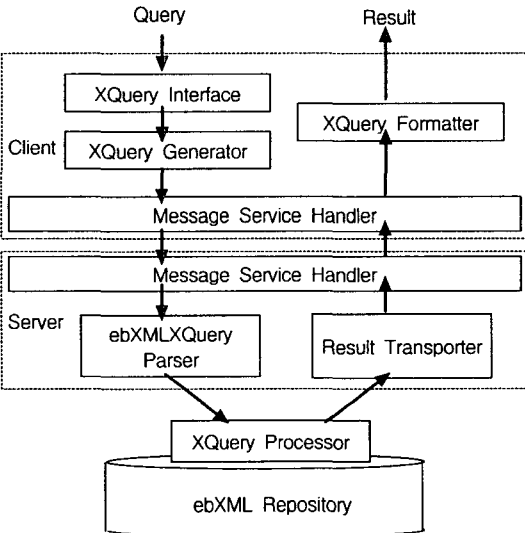
SQL 질의는 기존의 RDBMS에서 표준 질의 언어로 사용하고 있는 SQL의 문법을 그대로 이용하여, 사용하기 편리하고 기존 시스템에서 활용할 수 있다는 장점을 가지고 있지만 결과 값을 ObjectRef 형식으로만 반환할 수 있고 다른 객체 타입으로 결

과 값을 표현할 수 없다. 또한 Repository가 RDBMS인 경우에는 별도의 변환 과정이 필요 없고 Repository가 갖는 질의 언어 처리 기능을 충분히 활용하여 만족할 만한 성능을 기대할 수 있으나, Repository가 RDBMS가 아닌 경우에는 SQL질의를 사용 할 수 없거나, 사용하기 위해서는 해당 Repository의 질의 언어로 SQL 질의문을 변환하여야 한다.

XQuery 질의는 질의문에 반환할 결과 값의 형태를 정의할 수 있기 때문에 질의의 결과를 사용하게 될 응용이 질의의 결과를 재 가공하는 번거러움을 피할 수 있다. 또한 XQuery는 향후 XML 표준 질의 언어로 개발되고 있기 때문에 대부분이 Repository에서는 XQuery를 자체 지원하게 될 것이다. 따라서 XQuery를 이용하게 되면 Repository에 따른 질의 언어로 변환하는 변환 과정이 필요 없게 된다. <표 1>은 기존의 질의 방식과 XQuery에 기반한 질의 방식을 비교한 내용을 정리한 것이다.

4. XQuery 검색 시스템

ebXML 저장소에서 XQuery 검색을 지원하기 위한 전체적인 시스템의 구성도는 (그림 7)과 같다.



(그림 7) XQuery 검색 시스템 구성도

사용자가 정보 검색을 하면 Registry Client에는 클라이언트로부터 질의를 받아들이는 인터페이스가 있다. 입력받은 질의는 XQuery로 변환되어 메시징 서비스를 통하여 등록저장소의 서버로 전달된다. 이를 ebXML XQuery Parser가 ebXML RIM2.1과 ebXML RS2.1에 적합하게 작성되었는지 분석한다. 분석된 질의는 저장소에서 지원하는 XQuery Processor를 통하여 직접적으로 저장소에서 값을 추출해 결과 값이 만들어지고 Result Transporter가 클라이언트에 전달해 주고 사용자가 원하는 형태로 Result Formatter가 만들어 준 다음 결과 값을 보여 주게 된다.

현재 XQuery 검색 시스템 프로토 타입을 구현 중에 있으며, 시스템 구현을 위한 환경은 다음과 같다.

- * 하드웨어 : SunSparc Server 4
- * 운영체제 : SunSolaris2.6 Linux
- * 개발언어 : Java
- * 기반 DBMS : Oracle9i release 2
- * ebXMLrr Client
- * ebXMLrr Server

5. 결론

본 논문에서는 ebXML Registry and Repository에서 표준 XML 질의 언어인 XQuery를 이용하여 검색을 제공하는 방안을 제시하였다.

XQuery를 기반으로 제공되는 질의 유형을 식별하고 제공되는 각 검색 방법들의 특성을 비교 분석하였다. 또한 XQuery 검색 시스템의 구조를 설계하였다.

XQuery는 SQL 질의와 마찬가지로 Syntax작성이 단순하고, Filter질의 보다 더욱 다양한 결과 값들을 반환해 줄 수 있고, 저장소에서 XQuery를 지원하기 때문에 XQuery 질의를 처리하기 위한 별도의 질의 변환기를 고려할 필요가 없다.

앞으로 XQuery 검색 시스템 프로토 타입을 완성하고 분산환경을 지원하는 등록기에 대해 질의를 처리하고 결과를 통합할 수 있는 XQuery 검색 시스템을 구현해 나갈 예정이다.

참고문헌

- [1] OASIS/ebXML Registry Technical Committee "OASIS/ebXML Registry Information Model v2.1" June 2002
- [2] OASIS/ebXML Registry Technical Committee "OASIS/ebXML Registry Services Specification v2.0" June 2002
- [3] W3C Working Draft "XQuery1.0:An XML Query Language" 16 August 2002
- [4] Extensible Markup Language(XML)1.0, "http://www.w3.org/TR/REC-xml/20001006".
- [5] 한국전자거래진흥원 "2002 ebXML 백서" 2002.2
- [6] 김채미, 최학열 "글로벌 e비즈니스 리더를 위한 ebXML" 대청미디어, 2001.12.
- [7] 전희영, 김계용, 유정연, 이규철 "ebXML 등록기/저장소에서의 객체 질의의 관리 시스템" 정보처리학회 논문집 p37-39, 2001.10
- [8] 조강녕, 박재홍, 김상균, 이규철, 조현규 "ebXML Registry에 대한 Filter Query의 SQL 변환기법" 정보처리학회 논문집 p40-42, 2001.10
- [9] 이명철 "효율적인 구조검색을 지원하는 XQL 질의 처리기의 설계 및 구현" 충남대학교대학원 컴퓨터공학과 학위논문 2000.12
- [10] 김종민, 이민우, 허준희, 최한석 "Power XML" 정보게이트 2002.1
- [11] W3C Working Draft "XML Query Use Cases" 16 August 2002