The Journal of Korean
Institute of CALS/EC
Vol 6, No. 2, August 2001

XML 기반의 인터넷 정보 중재자의 설계 및 구현*

이강찬**, 이규철***

A Design and Implementation of Internet Information Mediator based on XML

Kangchan Lee, Kyuchul Lee

Abstract

As the proliferation of the Internet, especially World Wide Web, numerous information resources have been constructed. The characteristics of information resources on the Internet are that the information resources are distributed, autonomous, and heterogeneous. Moreover each information resource has its own query method, data representation, and schema structure. The integration of information resources is one of the most important research issues in the Internet data management. The task of information resources integration system is to answer queries that require extracting and combining data from multiple information sources. In this paper, we propose an XML-based Mediation Framework(XMF) for integrating the Internet information resources.

Keyword XMF, information integration, mediator, XML

---

* 본 연구는 소프트웨어연구센터(SOREC)의 핵심응용기술과제인 "XML 저장/검색 및 분산문서 시스템의 설계 및 구현"의 일부로 수행된 결과임.
** 한국전자통신연구원 정보화기술연구본부 표준연구센터
*** 충남대학교 공과대학 컴퓨터공학과 데이터베이스실

1. 서론

수십년간 컴퓨터 시스템의 발달, 즉 컴퓨터 하드웨어와 소프트웨어의 급속한 발전에 따라 각 조직의 컴퓨팅 환경은 매우 다양한 형태를 가지게 되었다. 특히 정보 시스템(information system)의 확산에 따라 각기 다른 데이터 모델(data model)을 지원하는 수많은 데이터베이스 관리 소프트웨어가 사용되어 왔다. 정보시스템의 초창기에는 파일 시스템을 사용하여 정보 시스템을 개발하였으며, 시간이 지남에 따라 사용자는 네트워크 및 계층형 DBMS, 관계형 DBMS, 객체 지향형 DBMS, 그리고 객체 관계형 DBMS 와 같은 다양한 모델을 지원하는 DBMS 를 이용하였다.

정보 시스템의 다양성 뿐만 아니라 정보 자원의 다양성은 인터넷의 확산을 통하여 더욱 가속화 되었다. 1990 년대의 인터넷의 확산은 WWW(World Wide Web) 기술의 보편화를 통하여 이루어졌으며, 데이터베이스와 웹의 결합을 통하여 파일 기반으로 정적인 문서만을 제공하는 웹 서비스가 아닌 데이터베이스를 이용한 다양한 컨텐츠를 제공하는 웹 서비스로 발전하였다. 즉, 기존의 중앙 집중식 컴퓨팅 환경이 아닌 보편화된 분산 환경에서 웹을 통한 손쉬운 정보의 접근은 정보의 생성을 용이하게 하였으며, 이로 인한 정보 자원의 증대 및 다양한 형태의 인터넷 서비스를 꾀하였다.

이러한 현대 컴퓨터 시스템의 발달, 즉 수많은 정보 시스템의 출현과 다양한 정보 서비스로 인한 거대한 인터넷 정보 자원으로 인하여 분산되어 있는 정보 자원에 대한 통합의 요구가 발생하게 되었으며, 더 나아가 분산에 대한 해결뿐만 아니라 서로 이질적인 정보 자원에 대한 통합의 요구가 발생하게 되었다. 기존에 구축된 정보 시스템에 대하여 통합을 시도함으로써 각각의 지역 시스템의 데이터 저장소에는 전혀 영향을 끼치지 않으며, 그들 정보 자원을 통합함으로써 새로운 정보를 창출하는 시도들이 증가하게 되었다. 이러한 시도들은 CALS 의 IDE(Integrated Data Environment) 및 PDM(Product Data Management), 메타 검색 엔진[9], 디지털 도서관 등의 초기 정보 통합 서비스 뿐만 아니라, 현재의 가상 문서(virtual document), ERP(Enterprise Resource Planning), 가상 기업(virtual enterprise), EAI(Enterprise Application Integration), 그리고 최근에 들어서 많은 사람들이 관심을 가지고 있는 EP (Enterprise Portal) 또는 EIP(Enterprise Information Portal) 등의 비즈니스 관점에서의 정보 통합에서 쉽게 찾을 수 있다.

그러나, 정보 시스템마다 사용하는 플랫폼과 프로토콜이 다르고, 질의 언어가 모두 다르며, 그러한 환경적인 요소의 문제가 아니라 할지라도 정보 자원의 데이터 모델에서 기인하는 다양한 이질성, 그리고 모든 요소들이 같다고 할지라도 정보 자원을 위한 스키마가 또 다를 수 있다[Hector, 1999; Kim, 1995b]. 또한, 스키마까지 같다 하더라도 데이터 값 체에 대한 이질성을 고려해 볼 수 있는 것이다. 이러한 다양한 문제에 대한 해결 방법으로 기존의 데이터베이스 분야에서는 여러 가지 방법을 통하여 해결하려고 시도하였으나, 단일 시스템의 정보

자원의 통합이 아닌 이질적인 인터넷 정보 자원을 통합하는 일은 그리 쉬운 일은 아니다. 더욱이, 데이터베이스 내의 정보 통합 뿐만 아니라 다양한 정보 시스템의 동적인 인터넷 정보 자원을 통합하는 일은 더욱 어려운 일이다.

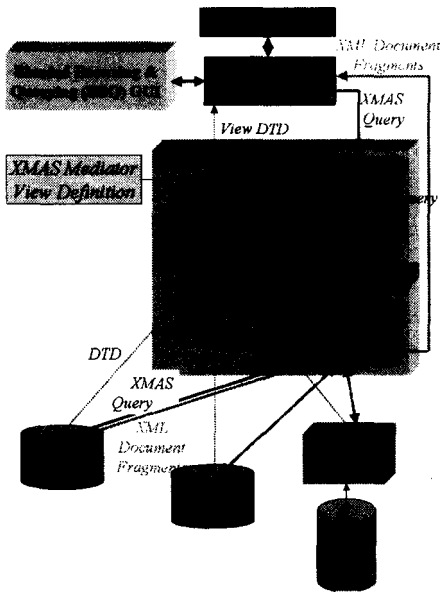
본 논문의 목적은 인터넷 정보 자원을 통합하기 위하여 모델링, 인터넷 정보 자원의 특징인 분산성, 자치성, 그리고 이질성을 해결할 수 있으면서, 유연한 통합 방법의 요소 기술을 제시하고, 이러한 요소 기술을 통하여 인터넷 상의 정보 자원을 통합하는 데 있다. 본 논문의 결과는 인터넷 상의 정보 자원의 통합을 요구하는 다양한 응용 분야의 기반 시스템으로 활용할 수 있을 것이다.

본 논문의 구성은 다음과 같다. 2 장에서는 기존의 통합 시스템과 XMF 를 비교 하였으며 3 장에서는 XMF 의 주요 구성 요소에 대하여 설명하고, 4 장에서는 XMF 에서 제공하는 중재 규칙에 대하여 설명하였다. 마지막으로 5 장에서 결론 및 향후 연구에 대하여 언급하였다.

2. 통합 시스템 비교

2.1 MIX

MIX(Mediation of Information using XML)[MIX, 1999a; MIX, 1999b]는 XML 자원들에 대한 통합을 수행하는 미디에이터 방식의 통합 시스템이다.



<그림 1> MIX 의 구조

이 시스템은 미디에이터 뷰 정의(Mediator View Definition)에 XMAS(XML Matching And Structuring Query Language)를 이용하여 XML 자원이 제공하는 데이터에 대한 뷰를 정의하고, 이 통합된 뷰에 대하여 질의를 수행한다. 결과 데이터 통합 시에 DTD 추론 모듈을 통하여 결과에 대한 DTD 를 생성해낸다.

MIX 시스템은 BBQ 라 불리는 사용자 인터페이스를 통해, 통합될 XML 문서에 대한 DTD 를 보여주며 이 구조에 따른 질의를 생성해 낼 수 있도록 한다. BBQ 로부터 생성된 질의는 VXD API 를 통하여 MIX 미디에이터에 전달되며, 이는 미리 정의되어 있는 미디에이터 뷰 정의를 참조하여, 서브 질의로 분해된 다음, 각 정보 자원들에게 전송된다.

XIX 에서 사용하는 XMAS 는 XML 질

의를 위한 룰 기반의 선언적 질의 언어로서 XMAS 의 룰 형식은 CONSTRUCT *head* WHERE *body* 으로 <그림 2>는 두 개의 문서를 우편번호로 조인하는 예이다.

```
CONSTRUCT
<big_neighborhoods_med>
  <big_nerghborhood_med>
   $N $S
  </> {$N, $S}
</>
WHERE
<neighborhoods>
$N:  <neighborhood>
        <zip>$Z</>
     </>
</>
</>
IN "http://www.npaci.edu/DICE
/MIX/tutorial/neighborhoods.xml"
AND
WHERE
<schools>
$S:  <school>
        <zip>$Z1</>
     </>
</>
</>
IN "http://www.npaci.edu/DICE/
MIX/tutorial/schools.xml"
AND $Z=$Z1
```

<그림 2> XMAS 질의 언어 예

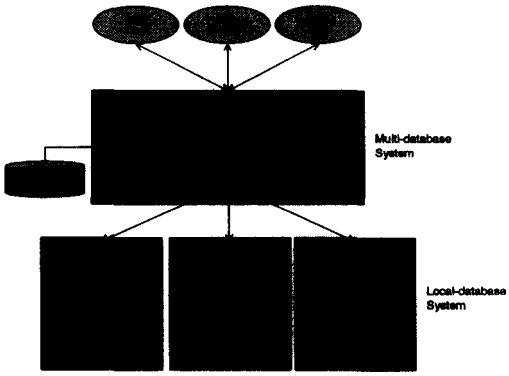
XMAS 장점으로는 사용자 질의(XMAS)만을 이용하여 쉽게 통합할 수 있고, 변수 바인딩(variable binding)을 통하여 원하는 데이터를 정확히 변환할 수 있으나 통합 대상이 문서(WWW 이나 DBMS 와 같은 정보 시스템이 아니다)이고, 통합 대상이 되는 문서 전체를 가져오게 되며 사용자 정의 함수를 통한 유연한 의미 충돌의 해결 방법을 제시하지 못하는 등의 단점이 있다.

2.2 UniSQL/M

UniSQL/M[UniSQL, 1994]은 미디에이터 방식의 정보 통합 시스템은 아니나 DBMS 의 통합 방법 제공 및 XMF 의 기본 아이디어를 도출하였다는 점에서 비교하였다. UniSQL/M 은 한 개 이상의 DBMS 로부터 데이터를 실체화(materialization) 시켜 통합하는 데이터베이스 관리 시스템으로 <그림 3>과 같은 구조를 가진다.

마스터(Master) 프로세스는 UniSQL/M 클라이언트와 UniSQL/M 데이터베이스 서버 사이, UniSQL/M 데이터베이스 서버와 UniSQL/M 로컬 데이터베이스 드라이버간의 통신을 초기화하는 역학을 수행하며, 서버나 드라이버 수에 상관없이 각 호스트마다 하나의 프로세스가 실행 실행된다.

드라이버(Driver)는 글로벌 데이터 베이스 엔진과 로컬 데이터베이스 엔진의 질의 변환을 위한 모듈로서 각 로컬 데이터베이스 서버마다 드라이버가 하나씩 구동된다. 마지막으로 서버(Server)는 로컬 데이터베이스와 글로벌데이터베이스 구동에 필요한 서버 역할을 담당한다.



<그림 3> UniSQL/M 의 구조

UniSQL/M 은 통합을 위하여 레지스터 정의, 프록시 정의, 그리고 가상 클래스의 정의 등을 수행해야 한다. 레지스터 정의에서는 글로벌 데이터베이스와 로컬 데이터베이스간의 연결을 설정하기 위해 UniSQL/M 에게 로컬 데이터베이스에 대한 정보(호스트 이름,데이터베이스 타입, 사용자 명, 패스워드 등)를 제공하며, 프록시 정의에서는 글로벌 데이터베이스 스키마와 로컬 데이터베이스의 테이블이나 클래스간의 일대일 매핑을 하여 UniSQL/M 을 통하여 접근할 수 있는 인터페이스로 로컬 데이터베이스의 테이블이나 클래스를 만드는 작업을 수행한다. 마지막으로 가상 클래스에서는 각각의 로컬 데이터베이스에 있는 테이블이나 클래스간에 존재하는 유사한 데이터에 대한 동일한 뷰를 정의하기 위해 새로운 형태의 클래스를 형성하는 것으로써 사용자는 가상 클래스를 이용하여 질의하거나 이용하게 된다.

UniSQL/M 장점으로는 단일화된 언어(SQL)로서 통합을 하게 되며, 유연한 충돌 문제 해결 방안을 제공하나 데이터 통합만 이루어지며, 드라이버를 지원하지 못하는 DBMS 나 정보 시스템과의 통합은 힘들다는 단점 또한 가지고 있다.

2.3 시스템 비교

XMF, MIX, 그리고 UniSQL/M 의 시스템을 정보 통합의 관점에서 비교하면 다음과 같다.

- XMF : 서로 다른 데이터 모델을 가진 정보 시스템이 관리하는 정보 자원을 XML 데이터 모델로 변환하여 글로벌 스키마 형태로 통합한 후, 질의 언어로 질의하는 검색 시스템
- MIX : 서로 다른 구조를 가진 XML 문서를 질의 언어를 이용하여 동적으로 XML 문서로 재 가공하는 문서 통합/변환 시스템
- UniSQL/M : 서로 다른 데이터 모델을 가진 DBMS 가 관리하는 데이터를 동적으로 통합하여 질의할 수 있는 DBMS

<표 1> 정보 통합 시스템 비교표

	XMF	MIX	UniSQL/M
통합 대상	WWW, DBMS 등과 같은 시스템 내이 관리하는 정보 자원	WWW 의 XML 문서 및 GIS 시스템, DBMS 와 같은 정보 자원을 XML 문서화 하여 통합	DBMS 가 관리하는 데이터 (드라이버에 따라 파일의 데이터가 통합 대상이 될 수 있음)
공통 데이터 모델	XML	XML	객체-관계 데이터 모델

래퍼	질의 변환 및 결과 변환, 정보 자원에 대한 연결 방법(JDBC, GET/POST) 제공	질의 변환 및 결과 변환 연결 방법 : GET	질의 변환 및 결과 변환 연결 방법 : socket
질의어	XPath(가상 문서에 대하여 질의)	XMAS(를 통하여 가상 문서 생성)	SQL(을 통하여 뷰 설정 및 실체화)
통합 방법 기술	XML 형태로 통합 중재 규칙 기술	질의어 자체에 통합 방법 기술	질의어 자체에 통합 방법에 대한 기술
글로벌 스키마	DTD, XML 스키마	- (DTD 추론 수행)	SQL
구조적 충돌 해결	O (통합 중재 규칙)	O (XMAS)	O (SQL)
의미적 충돌 해결	O (사용자 정의 함수)	O (XMAS 의 기능에 제한적)	O (사용자 정의 함수)

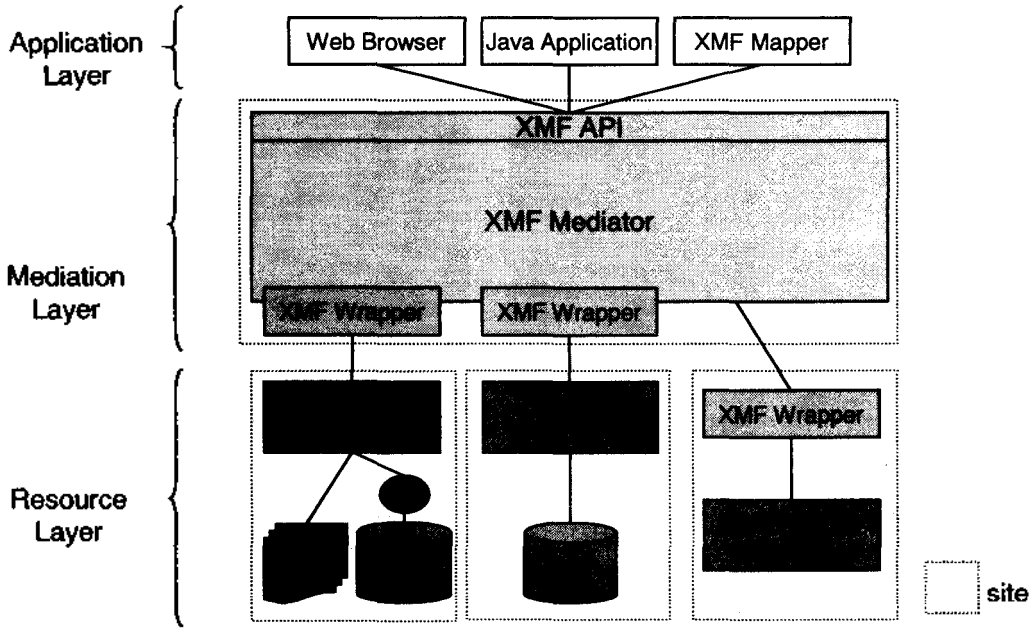
3. XMF

3.1 전체 구조

본 논문에서 제안하는 인터넷 정보 중재 시스템인 XMF 는 XML 기반의 정보 통합 도구로서 전형적인 미디에이터-래퍼 구조[Wiederhod, 1992]를 가진다. XMF 미디에이터는 사용자와 래퍼 사이에 위치하면서 중재자 역할을 수행하는 미들웨어(middle-ware)로서 사용자는 XMF 의 미디에이터를 통해 분산된 호스트에 존재하는 정보 자원을 검색할 수 있다.

<그림 4>는 XMF 의 계층 구조를 나타낸 것으로 XMF 는 응용 계층, 중재 계층, 그리고 자원 계층의 3 단계 계층 구조를 가진다. DBMS, 또는 WWW 과 같이 통합하고자 하는 시스템은 모두 자원 계층에 속하게 되며, 자원 계층에는 주로 통합하고자 하는 정보 자원을 위한 시스템들이 위치하게 된다. 중재 계층은 XMF 가 위치하는 곳으로 통합에 대한 다양한 구성 요소가 있으며, 중재 계층의 미디에이터에는 모든 데이터 관리 및 처리를 XML 기반으로 수행한다. 또한 자원 계층의 데이터 모델을 중재 계층의 데이터 모델로 변환하기 위하여 래퍼가 존재하게 되는데, 래퍼는 미디에이터와 같은 사이트 내에 위치하기도 하며, 다른 사이트에 위치 할 수도 있다.

마지막으로 응용 계층에는 XMF 를 이용하는 다양한 응용들이 위치하게 된다. XMF 는 일반 사용자들이 XMF 를 이용하여 프로그래밍 할 수 있도록 API 를 제공하기 때문에 자바 프로그래밍을 통하여 응용을 구현할 수 있으며, 기존적으로 XMF 를 웹에서 사용하기 위한 서블릿을 위한 라이브러리를 이용하여 웹으로도 사용할 수 있다. 또한 XMF 관리자가 사용하는 XMF 매퍼는 통합하고자 하는 정보 자원에 대한 중재 규칙을 위한 사용자 인터페이스이다. 즉, 모든 XMF 를 이용하는 응용 프로그램들은 XMF 의 응용 계층에 위치하게 된다.



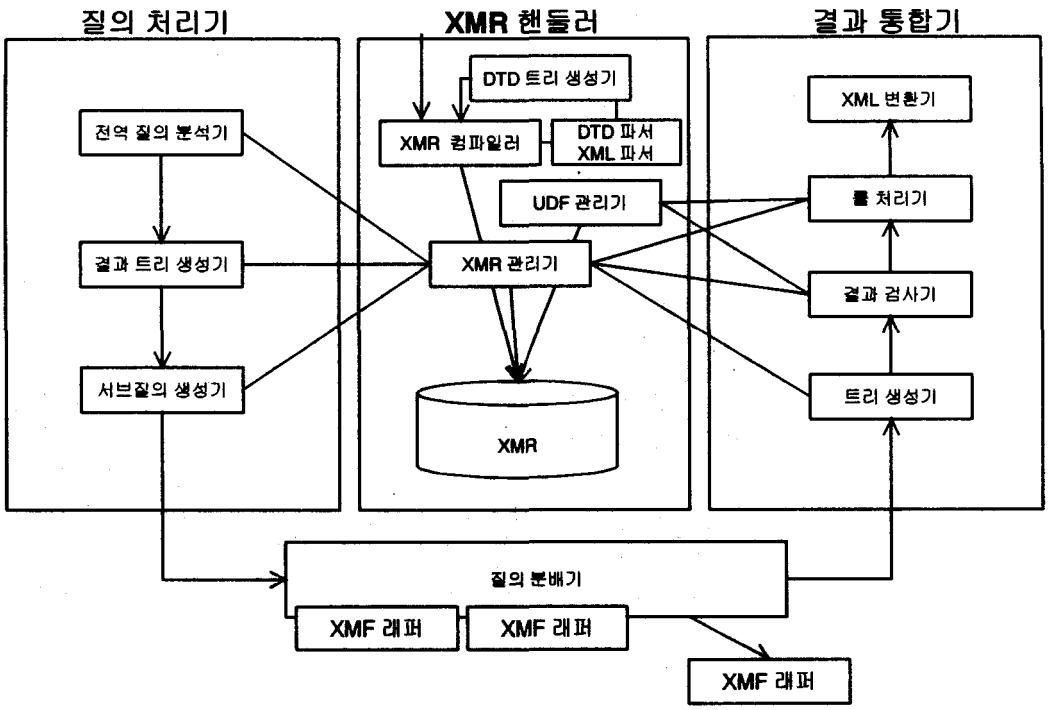
<그림 4> XMF 의 계층 구조

3.2 미디에이터

미디에이터는 사용자 질의를 입력 받아 각 래퍼로 사용자 질의를 재가공 및 분배한 후, 그 결과를 받아 통합하는 기능을 가지는 모듈이다.

<그림 5>는 XMF 의 미디에이터 구조로서 XMF 미디에이터는 크게 질의 처리기(Query Processor), XMR 핸들러(XMR Handler), 그리고 결과 통합기(Result Integrator)로 구성되어 있다.

질의 처리기는 사용자 질의를 입력 받아, 사용자 질의가 올바른 질의인지 검사를 한 후, 올바른 질의이면 각각의 정보 자원 시스템에서 이해할 수 있는 질의(decomposed query)로 변환을 하게 된다. 이때 사용하는 사용자 질의와 변환된 질의는 모두 XPath 형태의 질의 언어를 사용한다. XPath 를 사용하는 가장 큰 이유는 (i) 현재 W3C 에서 공식적으로 사용하는 XML 에 대한 질의 언어는 XPath 가 유일하며, (ii) XPath 를 지원하는 시스템이 증가하고 있어 추후 XMF 의 정보 통합 대상이 되는 시스템에서 XPath 를 지원하게 되면 통합이 용이하기 때문이다.



<그림 5> XMF 미디에이터 구조

결과 통합기는 이렇게 래퍼에게 전달된 서브 질의들에 대한 XML 문서 형태의 부분 결과들을 얻어와 매핑 룰에 따라 통합하여 하나의 XML 문서로서 사용자에게 질의 결과를 제공한다. 각 래퍼가 제공하게 되는 서브 질의 결과들은 지역 스키마와 정의에 따른 문서 구조를 이루고 있으며, 매핑 룰을 참조하여 해당 지역 스키마에 대응하는 전역 스키마를 참조하여 통합한다.

XMR 핸들러는 질의 처리 과정과 결과 통합 과정에서 수행하는 일련의 질의 분해 및 통합 과정에서 필요한 메타 데이터를 관리하며, 질의 처리기와 결과 통합기의 요구에 따라 적절한 매핑 룰과 통합 관련 메타 데이터를 제공하는 기능을 수행한다.

3.3 래퍼

래퍼[김경일, 2000; 김태현, 1999]의 수행 업무는 미디에이터에서 전달 받은 질의(decomposed query)를 각각의 정보 시스템에서 처리할 수 있는 형태의 질의 언어로 변환하여, 해당 시스템에게 질의하는 일과 질의 결과를 XML 형태로 변환하는 일인 질의 변환과 결과 변환이 가장 주된 일이다.

<표 2>에서 제시하는 것은 XMF 에서

자원하는 정보 시스템에 따른 래퍼 타입이다. 현재 5 가지의 타입을 지원하고 있으며, 각 타입에 따른 래퍼의 구조는 <그림 6>과 같다.

<표 2> 래퍼 타입

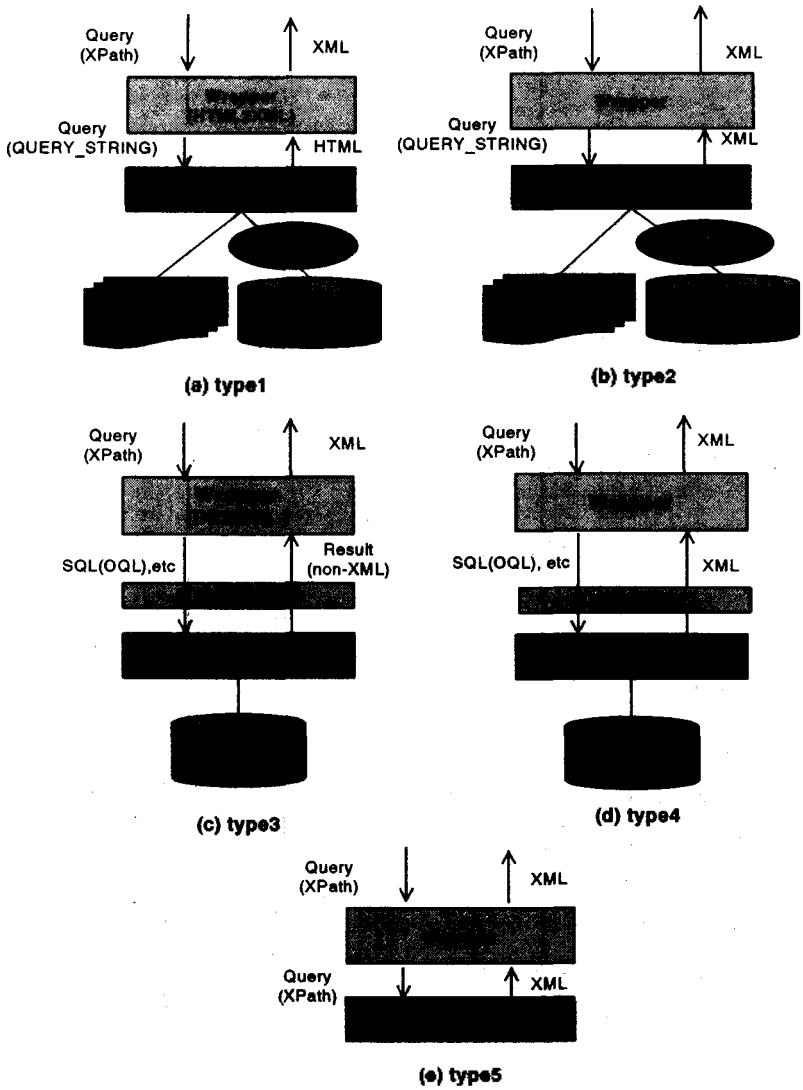
	타입 1	타입 2	타입 3	타입 4	타입 5
시스템 형태	WWW	WWW	DBMS	DBMS	WWW
접속 방법	HTTP	HTTP	JDBC	JDBC	HTTP
질의	Query String	Query String	SQL	SQL	XPath
결과 형태	HTML	XML	데이터베이스	XML	XML

타입 1 은 일반적인 웹 응용에 대한 래퍼의 예이다. 이러한 웹 응용은 일반적으로 HTTP GET/POST 를 이용한 파라미터 전달을 통하여 질의 결과를 생성해낸다. 이를 지원하기 위해서는 서브 질의 결과인 XPath 를 파라미터 단위로 변환해주는 모듈이 필요하게 되며, 결과도 일반적인 HTML 형식이므로 이를 XHTML 또는 XML 로 변환해줄 수 있는 래퍼가 필요하다. 타입 2 는 근래에 들어 나타나고 있는 웹 응용의 형식으로 질의 결과에 대한 마임 타입(mime type)이 text/xml 또는 application/xml 의 경우에 해당한다. 즉, 일반적인 웹 응용을 작성하되 질의 결과는 XML 로 보이는 예이다. 이러한 경우에는 질의 변환 모듈만 추가적으로 필요하게 된다.

타입 3 과 타입 4 는 정보 자원이 DBMS 의 경우로서 SQL 또는 OQL 과 같은 별도의 구조적 질의 언어를 사용하는 경우이다. 타입 3 은 완전한 DBMS 로서 SQL 과 관계형 데이터 모델을 따르는 질의 결과만을 제공하는데 반하여 타입 4 는 DBMS 위에 XML 모듈을 추가하여 질의 결과가 XML 인 경우이다. 타입 3 의 경우에는 전역 질의로서의 XPath 를 해당 정보 자원의 지역 질의, 즉 SQL 로 변환하는 모듈과 질의 결과를 XML 로 변환하는 모듈이 필요하며, 타입 4 의 경우에는 XPath 를 SQL 로 변환하는 모듈이 필요하다.

타입 5 는 완전한 XML 응용으로서 별도의 래퍼를 필요치 않고 곧바로 미디에이터에 연결할 수 있는 경우에 해당한다.

현재 지원하고 있는 5 가지의 타입 외에도 SOAP(Simple Object Access Protocol)을 지원함으로써 보다 다양한 형태의 웹에 대한 통합을 지원할 예정이며, XMF 의 래퍼 타입을 추가할 수 있도록 설계 및 구현함으로써 래퍼만 추가시켜 주면 다양한 형태의 정보 시스템에 대한 통합이 가능하도록 한 것이 특징이라 볼 수 있다.



<그림 6> XMF 의 래퍼 구조

4. 통합 중재 규칙

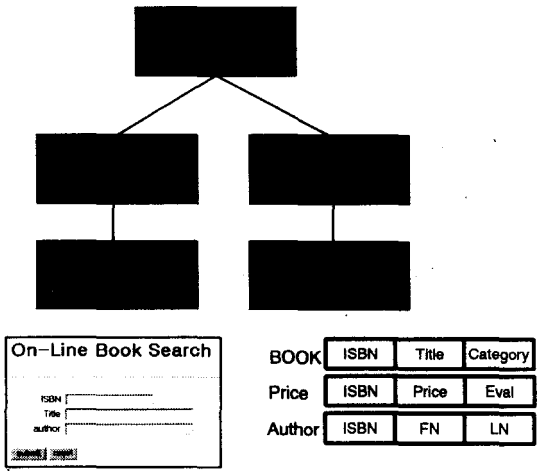
XMF 의 통합 중재 규칙(XMR; XMF Mediation Rule)은 XMF 의 질의 처리 및 결과 통합시에 필요한 스키마와 각 스키마 간의 관계를 정의한다. 이러한 중재 규칙에 대한 요구사항은 다음과 같다.

(1) 통합 중재 규칙은 질의 처리와 결과 통합에 동시에 사용할 수 있어야 한다.

(2) 통합 중재 규칙은 통합하고자 하는 시스템의 지역 시키마와 통합된 후의 모습인 광역 스키마와의 관계를 명시할 수 있어야 한다.

(3) 통합 중재 규칙에는 통합시 발생할 수 있는 충돌 문제(구조적 충돌 문제와 의미적 충돌 문제)[NIIP; Kim, 1995a]를 해결할 수 있어야 한다.

본 연구에서는 이와 같은 요구사항을 충족하기 위해서 XML 을 공통 데이터 모델로 이용하며, 각 정보 자원의 메타 데이터를 XML 로 표현한다. 또한 광역 스키마와 지역 스키마 와의 관계를 모두 XML 로 표현한다.

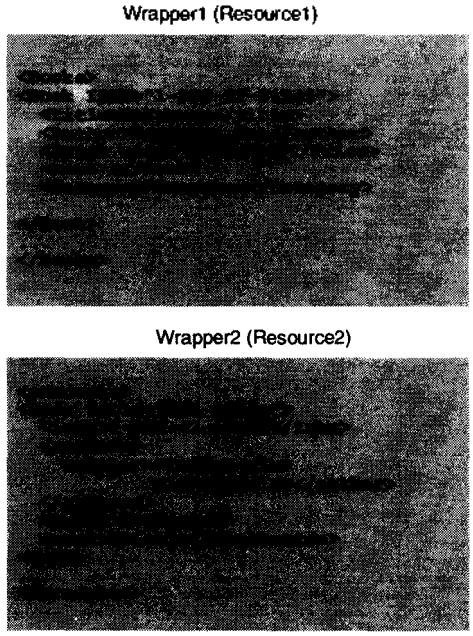


<그림 7> 정보시스템 환경

예를 들어 <그림 7>과 같은 온라인 도서에 검색 서비스가 있다고 가정해 보자. 각 시스템은 정보 시스템이 WWW 와 DBMS 로 각각 상이하며, 스키마 형태도 상이하다.

Resource1 의 경우에는 웹에 연동되어 검색할 수 있으며 그 결과는 XML 형태로 전달한다고 가정해 보다. 이에 따라 XMF 의 래퍼 타입 중에서는 타입 2 번에 해당하게 된다. 또한 Resource 2 의 경우에는 타입 3 에 해당하게 된다.

이와 같은 래퍼가 설정이 되면 다음 그림과 같은 래퍼의 결과를 가정할 수 있다.



<그림 8> 래퍼의 결과

즉, XMF 의 래퍼를 통하여 전달되어 온 결과는 모두 XML 형태이며, 이 XML 형태를 사용 XMF 통합 중재 규칙에 따라 통합하면 된다.

XMF 통합 중재 규칙은 다음과 같은 (i) 데이터 등록 부분 – 각 정보 시스템에 대한

접근 방법(정보 자원 이름, 위치 정보, ID, 패스워드 등)과 의미 충돌시 충돌 문제 해결을 위한 사용자 정의 함수(User Defined Function)을 등록 하는 부분 (ii) 스키마 기술 부분 – 통합하고자 하는 로컬 스키마에 해당하는 정보 시스템의 DTD 및 글로벌 스키마에 해당하는 DTD 의 등록과 같은 스키마 등록 (iii) 매핑 룰 부분 – 각 로컬 스키마와 글로벌 스키마와의 매핑 관계를 기술하는 부분 등의 세 부분으로 구성되어 있다.

<그림 9>의 예는 <그림 7>의 환경에서 통합하기 위한 통합 중재 규칙이다. <resigtry> 부분에는 통합하고자 하는 정보 자원에 대한 래퍼의 기술이다. 만약 통합하고자 하는 정보 자원에 대한 래퍼가 같은 호스트라면 CONNECTION 값이 LOCAL 이며, 다른 호스트로 연결되면 REMOTE 가 된다. 또한 REMOTE 의 경우에는 어떤 컴퓨터에 있는 래퍼인지를 명시해 주어야 한다. 그리고 함수를 등록 하는 부분이 있는데, 각각의 함수는 arithmetic, concatenate, convert 가 된다.

```
<?xml version="1.0"?>
<xmr>
  <registry>
   <sources operation="merge">
    <source name="ls1" connection="LOCAL" type="2" />
    <source name="ls2" connection="REMOTE" METHOD="POST" type="3">
      <url>http://xmf.foo.com/type3.cgi</url>
      <query name="xmfquery"></query>
    </source>
   </sources>
   <dest name="gs" />
   <function name="arithmetic" sclass="division.class" dclass="multiplication.class" />
  <function name="concatenate" sclass="SpilitString.class" dclass="MergeString.class" />
  <function name="convert" sclass="ToNumber.class" dclass="ToChar.class" />
  </registry>
  <schemas>
    <schema  name="gs"  ref="catalog.dtd"/>
    <schema  name="ls1" ref="book.dtd"/>
    <schema  name="ls2" ref="product.dtd"/>
  </schemas>
  <maps>
    <map dest="gs:/catalog"/>
    <map dest="gs:/catalog/book"/>
    <map dest="gs:/catalog/book/@ISBN" source="ls1:/Books/Book/@ISBN"/>
    <map dest="gs:/catalog/book/@ISBN" source="ls2:/products/Book/isbn"/>
```

```
    <map dest="gs:/catalog/book/title" source="ls1:/Books/book/title"/>
    <map dest="gs:/catalog/book/title" source="ls2:/products/Book/@title"/>
    <map dest="gs:/catalog/book/authors"/>
    <map dest="gs:/catalog/book/authors/author" source="ls1:/Books/book/Author"/>
    <map dest="gs:/catalog/book/authors/author"
         source="concatenate(1, ls2:/products/Book/authors/author/ln)" />
    <map dest="gs:/catalog/book/authors/author"
         source="concatenate(2, ls2:/products/Book/authors/author/fn)" />
    <map dest="gs:/catalog/book/price" source="ls1:/Books/book/Price"/>
    <map dest="gs:/catalog/book/price"
         source="arithmetic(ls2:/products/Book/price, * 1200)" />
    <map dest="gs:/catalog/book/eval" source="convert(ls1:/Books/Book/eval)" />
    <map dest="gs:/catalog/book/eval" source="ls2:/products/Book/evaluation" />
    <map dest="gs:/catalog/book/category" source="ls1:/Books/Book/Category"/>
    <map dest="gs:/catalog/book/category" source="Computer"/>
  </maps>
</xmr>
```

<그림 9> XMF 통합 중재 규칙 예

각 함수에는 sclass 와 dclass 가 있는데, sclass 는 질의할 때 사용하는 클래스이며, dclass 는 결과 통합에 이용하는 클래스이다. 이렇게 클래스가 나뉘게 되는 이유는 한 정보 자원의 값이 12000 원의 의미를 가지고 있고, 다른 정보 자원의 값이 10 달러라는 의미를 가지고 있다고 하였을 때 환율이 1200 원: 1 달러라면 사실 12000 과 10 은 같은 수를 의미한다. 따라서 질의할 때 12000 원인 것을 찾는다면 10 달러인 것도 같이 찾아야 하며, 결과 통합할 경우에는 10 달러는 12000 원으로 변경되어야 한다. 즉 질의할 때와 결과 통합할 때에는 반대되는 역할의 함수들이 각각 필요하게 된다.

또한 <resigtry>는 정보 자원의 통합시 어떠한 형태로 통합할 것인가에 대한 연산자를 지원한다. 제공되는 연산자는 다음과 같다.

- merge : 각 정보 자원을 단순 병합 (중복 허용)
- join : 각 정보 자원을 설정한 키에 의하여 조인
- union : 각 정보 자원을 설정한 키에 의하여 유니온 (중복 제거)
- difference : 각 정보 자원을 차를 구하는 연산

<schema>는 각 정보 시스템의 스키마와 글로벌 스키마를 링크하게 된다. 즉, <schema> 부분에는 통합하고자 하는 정보 자원이 래퍼를 통하여 전달된 후의 XML 모습을 DTD 또는 XML 스키마 형태로 링크하게 된다. 예를 들어 통합하고자 하는 정보를 DTD 로 모델링 한 후 그 DTD 를

링크하게 되면 <schema name="ls1" ref="book.dtd"> 형태와 같이 DTD 파일 이름을 ref 어트리뷰트의 값으로 넣어주면 된다. 또한 만약에 통합하고자 하는 정보 시스템을 XML 스키마 형태로 모델링하면 <schema name="ls1" ref="book.xsd">와 같이 스키마 파일을 링크하면 된다. 즉, 각 정보 자원을 DTD 또는 XML 스키마의 형태로 모델링 한 후 파일 이름만 링크하게 되면 XMF 미디에이터에서는 그 파일을 로딩하여 파싱 및 구조 정보에 대한 분석을 하게 된다.

마지막으로 <maps> 엘리먼트는 통합하고자 하는 정보 자원에 대하여 어떻게 글로벌 스키마와 매핑시킬 것인가에 대하여 기술하는 부분이다. <map>의 어트리뷰트는 dest 와 source 가 있게 되는데 dest 는 글로벌 스키마를 의미하며 source 는 로컬 스키마의 정보 자원을 의미한다. dest 와 source 어트리뷰트는 XPath 의 생략형 문법(abbreviate syntax)를 변형하여 사용한다. 예를 들어 <map dest="gs:/catalog/boon/@ISBN"source="ls1:/Books/Book/@ISBN"">는 <schemas>와 <registry>에 ls1 이라고 정의된 정보 자원 중 /Books/Book/@ISBN 에 해당하는 값을 글로벌 스키마인 gs 의 /catalog/boon/@ISBN 에 매핑하는 것을 의미한다. 이러한 매핑을 통하여 통합 시 발생할 수 있는 구조적 충돌의 문제는 해결할 수 있게 된다. 만약 의미적 충돌 문제가 발생하게 되면 <registry>에 정의되어 있는 함수들을 이용하여 <그림 9>의 중재 규칙을 이용하여 해결하면 된다.

최종적으로 XMF 의 통합 중재 규칙에 의하여 통합된 모습은 <그림 10>과 같다.

<그림 10> 통합 후의 결과

5. 결론

인터넷 정보 자원에 대한 통합은 정보화 사회에서의 개별적인 정보 자원을 통합함으로써 새로운 정보의 추출 가능, 인터넷 정보 자원의 통합에 대한 요구 사항 증대, 그리고 기 구축된 정보 또는 데이터의 재사용 등에 있어 매우 중요하다고 볼 수 있다. 그러나, 인터넷 환경과 같이 다양한 형태의 정보 자원이 존재하며, 분산 환경 하에서 통합하고자 하는 각 시스템의 자치성을 보장하면서, 그리고 인터넷 환경에서의 모호한 정보 자원을 결함 없이(seamless) 통합하는 것은 매우 어려운 일이다.

본 논문에서는 인터넷 환경에서 이질적인 구조를 가진 정보에 대하여 XML 기반으로 통합하는 방안에 대하여 제안하고, 이를 바탕으로 구현하였다. XMF 는 인터넷 상의 이질적인 정보 자원을 XML 을 데이

터 모델로 이용하여 통합함으로써 사용자에게는 단일화 된 뷰(view)로 보이도록 하는 통합 시스템이다. 따라서 사용자에게는 단일화된 뷰 형태의 XML 로 인식될 수 있지만, 이는 서로 다른 인터넷 정보 자원으로부터 통합 절차에 따라 수행되어진 결과이다. 또한 XMF 의 통합 대상으로는 기존의 멀티 데이터베이스와는 다르게 DBMS 뿐만 아니라 다양한 인터넷 정보 자원까지 통합할 수 있으며, XML 형태를 따르는 정보 자원은 손쉽게 통합할 수 있다.

또한, XMF 의 일반 사용자는 XMF 를 이용하여 상이한 정보 자원들에게 일괄된 방법으로 질의할 수 있게 된다. 또한 질의 결과는 XML 형태로 전달됨으로써 데이터의 구조를 내포하고 있으며, 이에 따라 구조적 질의가 가능하게 된다.

XMF 의 향후 연구 과제로는 현재 맵 정보는 단순 평문 형태의 XML 로 되어 있기 때문에 쉽게 편집 할 수 있으나 보다 비쥬얼한 형태의 인터페이스를 이용하여 쉽게 맵을 구성할 수 있으며 부분적으로 자동화된 XMF 통합 중재 규칙을 생성할 수 있도록 할 예정이며, 현재 제공하고 있는 래퍼 외에도 다양한 프레임워크에서 사용할 수 있도록 SOAP 을 지원하는 래퍼를 추가할 예정이다. 또한 현재는 XPath 를 기반으로 질의 언어가 설계 및 구현되어 있는데, W3C 의 XML 에 대한 질의 언어인 Xquery 가 완성되면 XMF 에도 도입할 예정이다.

참고 문헌

[김경일, 2000] 김경일, 이경하, 이강찬, 이규철, "XML RDB 게이트웨이의 설계 및 구현," *한국정보과학회 2000 가을 학술발표논문집(I)*, 제 27 권, 제 2 호, pp. 287-289, 2000.

[김태현, 1999] 김태현, 김경일, 이강찬, 이규철, " XML 기반의 정보 통합을 위한 OODB2XML 래퍼의 설계 및 구현", *한국정보과학회 '99 가을 학술발표논문집(I)*, 제 26 권 제 2 호, pp. 75-77, 1999.

[이강찬, 2000] 이강찬, 이경하, 이규철, "XMF 기반의 가상 멀티미디어 문서 생성", *2000 년도 추계학술발표 논문집*, 한국 멀티미디어학회, p81-84, 2000.

[이경하, 2000] 이경하, 조정수, 이강찬, 이규철, "XMF : XML 기반 분산 이질 정보 자원의 통합 프레임워크," *KDBC 2000 학술 발표논문집*, p262-270, 2000.

[Abiteboul, 2000] S.Abiteboul, P. Buneman, D. Suciu, "*Data On the Web*", Morgan Kaufmann Publishers, 2000.

[DOM, 1998] Vidur Apparao, Steve Byrne, Mike Champion, Scott Isaacs, Ian Jacobs, Arnaud Le Hors, Gavin Nicol, Jonathan Robie, Robert Sutor, Chris Wilson, Lauren Wood, "Document Object Model (DOM) Level 1", World Wide Web Consortium Recommendations, October 1998, Available at http://www.w3.org/TR/REC- DOM-Level-1/.

[Domenig, 1999] Ruxandra Domenig, Klaus R. Dittrich, "An Overview and Classification of Mediated Query System", *SIGMOD Record*, 28(3), pp. 63-72, 1999.

[Hector, 1999] Hector Garcia-Molina, Jeffery D. Ullman, Jennifer Widom, "*Database System Implementation*", Prentice Hall International, Inc, 1999

[Kim, 1995a] Won Kim, Injun Choi, Sunit K. Gala, Mark Scheevel, On Resolving Schematic Heterogeneity in Multidatabase Systems, Modern Database Systems, pp. 521-550, 1995.

[Kim, 1995b] Won Kim, "Introduction to Part 2: Technology for Interoperating Legacy Databases", *Modern Database System*, pp. 515-520, 1995

[MIX, 1999b] Chaitanya K. Baru, Vincent Chu, Amarnath Gupta, Bertram Ludscher, Richard Marciano, Yannis Papakonstantinou, Pavel Velikhov, " XML-based Information Mediation for Digital Libraries". *In proceeding of ACM DL*, pp. 214-215 1999.

[MIX, 1999a] C. Baru, A. Gupta, B. Ludascher, R. Marciano, Y. Papakonstantinou, and P. Velikhov. XML-Based Information Mediation with MIX, *In*

Proceeding of ACM SIGMOD, 1999.

[NIIIP] NIIIP(National Industrial Information Infrastructure Protocols), Available at http://www.niiip.org/

[Papakonstantinou, 1995]Y. Papakonstantinou, H. Garcia-Molina, J. Widom, "Object Exchange across Heterogeneous Information Sources", *In Proceeding of Data Engineering*, pp. 251-260, 1995.

[UniSQL, 1994] UniSQL/M Manual, UniSQL Inc., 1994.

[Wiederhold, 1992] Gio Wiederhold, "Mediators in the Architecture of Future Information Systems," *The IEEE Computer Magazine*, Mar, 1992.

[XML, 1998] Tim Bray and C.M. Sperberg-McQueen, "Extensible Markup Language (XML): Part I. Syntax", World Wide Web Consortium Recommendations, 1998. 2. 10, Available at http://www.w3.org/TR/REC-xml

[XMLSchema, 2000] David C. Fallside, "XML Schema Part 0: Primer", World Wide Web Consortium Candidate Recommendation, October 2000, Available at http://www .w3.org/TR/xmlschema-0/.

[XPath, 1999] James Clark, Steve DeRose, "XML Path Language (XPath) Version 1.0", World Wide Web Consortium Recommendations, Nov. 1999, Available at http://www.w3.org/TR/xpath.

저자 소개

이강찬 (chan@etri.re.kr)
충남대학교 컴퓨터공학과 졸업(학사, 석사, 박사)
현재 한국전자통신연구원 표준연구센터 선임연구원
관심 분야 : 데이터베이스, 정보 통합, XML, 미디에이터, WWW

이규철 (kclee@ce.cnu.ac.kr)
서울대학교 컴퓨터공학과 졸업(학사, 석사, 박사)
미국 Syracuse University, CASE Center 객원 교수
미국 IBM Almaden Research Center 객원 연구원
현재 충남대학교 공과대학 컴퓨터공학과 교수
관심 분야 : 데이터베이스, XML, 정보 통합, 멀티미디어 시스템, 전자상거래