

웹기반에서 XML을 이용한 데이터공유 연구

Study for Data Sharing Method with XML in Web Based

한 흥구*, 전일선, 정만수, 박 호균, 우 성구 최 성
남서울대학교 컴퓨터학과, 한국정보통신기술사협회

Heung Ku Han*, Ilsun Jun, ManSoo Jung, HoGyun Park, SungGu Woo, Sung Choi
Dept. of Computer Science, Namseoul University, Iccpeak

요 약

전세계는 웹을 이용하여 모든 사람들과 통신이 가능한 시대이다. 많은 컴퓨터에 연결되어있는 수많은 자료를 플랫폼과 응용 프로그램에 종속되지 않고 사용할 수 있다면 우리들은 많은 자료를 효율적으로 어느 곳에서든지 활용할 수 있을 것이다. 또한 개발환경이 인터넷과 인트라넷의 급격한 증가로 인해 개발자들은 분산된 환경에서 작업을 수행하는 일이 많아지면서 기존의 HTML을 이용한 문서의 공유나 교환은 HTML의 제한점들로 인해 많은 부담이 된다.

본 논문에서는 공유되어진 XML데이터의 출력 인터페이스를 작성하여 효율적인 인터페이스를 제안하였으며, 분산환경에서의 정보 공유와 교환을 위해 HTML의 제한점들을 보완하도록 XML을 연구하였다.

1. 서론

오늘날 전세계는 웹을 이용하여 모든 사람들과 통신이 가능한 시대가 되었다. 상호 운영 방식의 수많은 레이어 상에서 웹 통신을 가능하게 하는 것은 바로 널리 공인된 표준들으로써, 이는 웹의 최대 가능성에 있어서 결정적인 역할을 담당한다. 또한 개발환경의 변화는 기존의 사무실 환경에서 네트워크와 인터넷의 급속한 증가에 힘입어 점차 분산화되어 가는 추세이고, 이와 더불어 발생하는 다양한 문제점들 역시 유발되고 있다. 개발팀원들이나 시스템 분석가나 유지보수자 등의 측면에서 정보 공유 역시 커다란 문제로 나타나고 있다.

인터넷상의 대부분의 정보와 분산된 개발 환경은 인트라넷이나 인터넷을 기반으로 하여 서로의 정보를 HTML(Hyper Text Markup Language)을 이용하여 표현하고 공유하고 있다. 하지만 최근에는 HTML의

단점을 극복하고 SGML(Standard Generalized Markup Language)의 장점을 취한 XML(eXtensible Markup Language) 표준이 등장하면서 인터넷상에서 XML문서의 활용에 대한 관심이 대폭 고조되고 있는 추세이며, 이에 대한 연구도 활발히 진행되고 있는 실정이다. XML은 HTML의 문서의 구조가 고정되어 있어 확장성이 없고 문서 유효성을 검사 할 수 가 없는 단점들을 보완하였고, SGML의 부분집합이기 때문에 HTML이 갖는 문서 구조의 확장성과 문서의 포맷 검사기능에 대한 문제점을 해결하였다. 또한 XML표준 구조 자체가 유연하기 때문에 XML 표준은 기존의 데이터베이스와 호환성을 가질 수 있다. 기존 데이터베이스의 데이터를 XML파일 형식으로 변환하고 서로 다른 웹 응용프로그램에서 이 데이터베이스를 공유하여 사용할 수 있다면 효율적인 작업이 될 것이다.

본 논문에서는 공유되어진 XML데이터의 출력 인터

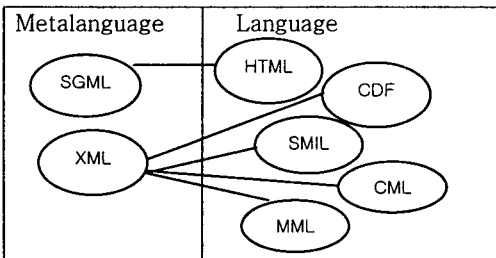
페이스를 작성하여 효율적인 인터페이스를 제안하였었으며, 기존에 사용하여 오던 여러 종류의 데이터베이스를 XML형식으로 변환하고 이 XML파일을 웹에서 공유하여 여러 가지 다양한 형태로 출력 할 수 있도록 연구하였다.

본 논문의 구성은 다음과 같다. 2장에서는 메타데이터로서의 XML과 컴포넌트로서의 XML에 대한 개괄적인 내용과 본 연구에 필요한 관련연구들을 기술한다. 3장에서는 XML을 이용한 데이터 공유방법 설계에 대해 알아보고, 4장에서는 공유되어진 XML데이터의 출력 인터페이스를 제안하였다. 5장에서는 XML Data Query 를 알아보고, 마지막으로 6장에서는 결론을 맺는다.

2. XML 관련 연구

2.1 메타데이터로서의 XML

XML은 객체지향 기술의 중요한 바탕을 구성한다. 지금까지 발전되어온 객체지향 기술들은 하나하나의 객체들은 각각 DB, 자바클래스, COM/CORBA 객체들로 만들어 졌지만, 이들 객체를 저장하고 읽어 들일 수 있는 메타데이터 형식이나 표준방법은 존재하지 않는다.



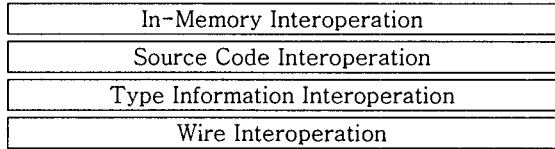
<그림1. 메타데이터로서의 XML>

XML은 완벽한 메타데이터 포맷이다. 프로그래머가 XML을 메타데이터 포맷으로 사용 하기 위해서는 데이터의 형식을 정의하기위한 적절한 DTD를 정의하고, 여기에 맞추어 XML데이터를 작성하고, 또한 데이터 작성을 위해서는 DOM을 지원하는 XML 파서를 사용해 데이터를 생성하기만 하면 되므로 XML은 프로그래머가 수행해야 할 작업을 대폭적으로 줄여준다.

2.2 컴포넌트로서의 XML

XML은 원래 HTML을 확장하기 위한 솔루션으로 설계되었지만, 서로 다른 종류의 컴포넌트 기반 시스템을 통합하는 기술로도 많이 사용되고 있다. 컴포넌트는 근본적으로 상호운용을 위한 것이다. 상호운용

정도에 따른 계층모델은 그림2에서 4가지 등급을 보여주며, XML은 기본적으로 데이터와 메시지 교환을 위한 최소 전송 표현을 정의한다. XML은 타입 정보 표현을 임의로 규정하지 않으며 XML 데이터스트림을 규정하기 위한 표준 메커니즘인 DTD(Document Type Definition)를 제공한다.



<그림2. 컴포넌트 상호 운용의 네 기능>

2.3 XML과 HTML 및 SGML의 관계

XML은 HTML같은 고정된 형식이 아닌 확장이 가능한 언어이다. HTML은 태그가 한정되어 있는 반면 XML은 문서의 내용에 관련된 태그를 사용자가 직접 정의할 수 있으며 그 태그를 다른 사람들이 사용하도록 할 수 있다. XML은 본질적으로 다른 언어를 기술하기 위한 언어, 즉 메타언어이다.

XML은 SGML의 간략화 된 버전으로, SGML의 실용적인 기능만을 모은 부분집합이라 할 수 있다. XML은 SGML의 기능 중 80%의 장점과 20%의 복잡성을 가지고 있다. XML 개발자들은 20%의 복잡성을 제거하기 위해 노력하였고, 그 결과 SGML 표준 안은 500 페이지에 달하는 반면 XML 표준 안은 30페이지 정도 밖에 되지 않는다. XML은 SGML의 장점과 일반성을 최대한 수용하는 한편, SGML의 특정 부분을 발췌/요약하여 전문가와 일반인 모두 쉽게 배워 웹에서의 구현을 편리하게 할 수 있는 언어이다.

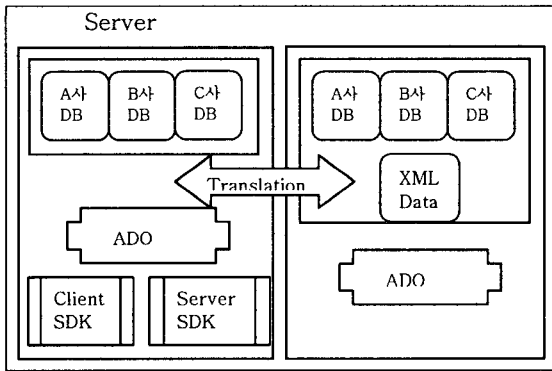
2.4 데이터 변환 및 공유의 필요성

웹상의 많은 HTML문서들이 XML문서로 변환될 가능성이 많으며, 쉽게 활용할 수 있는 시스템이 요구되어지고 있다. 이런 웹 시스템은 간단하게 XML파서와 브라우저로 구성된다. XML파서는 문서가 잘 구성되어 있는지, 유효한 문서인지를 검사하게 된다. 이는 XML문서의 오류가 발생하여 문서의 내용이 변경될 수도 있고, 응용프로그램에서도 오류를 발생시킬 수 있기 때문에 검사가 필요하다. 또한 브라우저는 XML문서를 Stylesheet정보에 의해서 클라이언트 측 사용자에게 시각적인 화면을 제공한다.

3. 웹기반에서 XML을이용한 데이터공유방법 설계

3.1 off-line 상에서의 데이터 변환

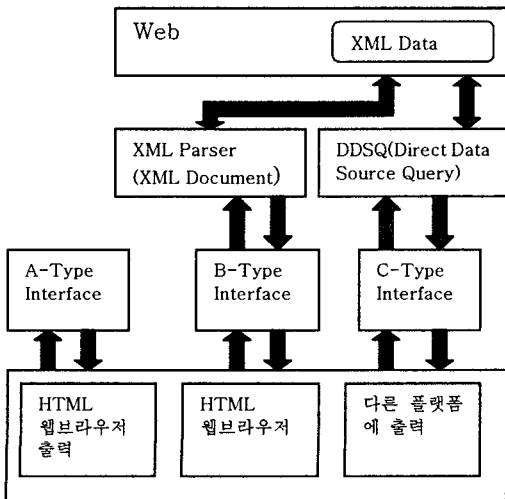
서로 다른 응용프로그램 혹은 도구에 의해 생성되어진 데이터소스를 XML포맷 형식으로 변환하여 재사용이 가능한 XML파일로 웹 서버에 저장한다. 각각의 데이터베이스에서 데이터를 ADO개체를 이용하여 XML데이터 포맷 형식의 데이터를 서버에 저장한다. 이렇게 Translation 되어진 XML Data는 웹에서 많은 사용자에게 공유될 수 있다.



<그림3. ADO를 이용한 XML Data Translation>

3.2 on-line상에서의 데이터 공유

XML Data는 응용 프로그램과 운영체제의 영향을 받지 않으므로 사용자 혹은 개발자가 평소에 사용하는 다양한 인터넷 프로그래밍 도구를 사용하면 된다. 이것은 기존의 인터넷 프로그램은 응용 프로그램에 대한 운영체제의 종속성과 데이터베이스의 호환성 문제로 그에 맞는 인터페이스를 작성하는 것과는 많은 차이가 있음을 의미한다.



<그림4. on-line상에서의 데이터공유>

오늘날 모든 시스템 환경은 인터넷으로 모든 자원이 공유되어 동시에 접근하여 작업이 이루어지는 시대이다. 그런데 어떤 시스템에서는 작동이 안되어 새로이 출력 인터페이스를 만들거나 시스템을 교체해야 한다면 많은 불편과 고충이 따를 것이다. 웹에 공유되어있는 XML Data를 다양한 방법으로 표현하여 그 특징과 차이점을 살펴보면 다음과 같다.

첫째, 공유된 XML Data에 접근하는 방법은 두가지 형태로 구분해 볼 수 있다. 즉 클라이언트 측에 있는 사용자가 인터페이스에서 임의 자료를 사용하고자 할 때 DOM 파서에 원하는 쿼리를 Request 했을때 XML Data가 XML Document 형태로 파싱되어 그 쿼리 결과를 브라우저에 표시할 수 있다. 또한 클라이언트에서 DOM 파서를 사용하지 않고도 바로 공유 XML Data에 접근하여 원하는 형태로 그 결과를 되돌려줄 수도 있다. 이 단계의 중간 매체로 DDSQ(Direct Data Source Query)로 표현 할 수 있다.

둘째, 되돌려지는 쿼리의 결과를 어떤 형식으로 출력 인터페이스를 작성하는 가에 따라 다양한 작업을 할 수 있다.

4. 출력 인터페이스

4.1 A-Type Interface

XML 데이터를 DSO개체를 이용하여 HTML로 작성한 인터페이스이다. ASP에서 ASP스크립트로 클라이언트에서 요구하는 쿼리를 DDSQ형식으로 공유되어있는 XML Data로 접근한 후 그 결과를 출력할 수 있다. 작성되어진 출력 인터페이스는 웹 데이터베이스의 개념을 도입하여 사용자들이 많이 사용하는 레코드셋 개체 즉 커서의 이동을 나타내 보았다. 여기서는 레코드셋의 이동과 데이터의 추가 뿐 아니라 다양한 매소드를 사용하여 출력할 수 있다.

4.2 B-Type Interface

XML데이터와 XSL문서가 같은 파일 내에 위치한 인터페이스이다. XML에서는 출력 인터페이스로 CSS와 XSL형식의 스타일 시트를 가지고 있어 XML 데이터를 다양한 형태로 출력 할 수 있다. 이번 출력 인터페이스는 XML Data를 HTML파일에 직접 입력함과 동시에 출력하기 위한 HTML 태그를 사용하여 브라우저에 테이블 형태로 나타내는 인터페이스로 데이터의 이동, 추가, 검색 등의 기능을 포함할 수가 없어

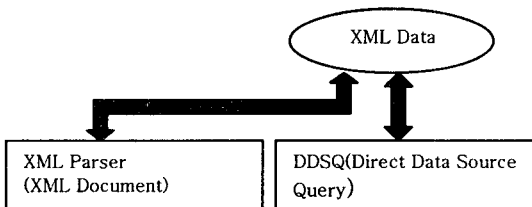
다양하게 작업하는 데는 한계가 있고, 재활용의 개념과는 거리가 멀다.

4.3 C-Type Interface

다른 곳에 위치한 XML데이터와 XSL을 결합시켜 출력하는 인터페이스이다. 위의 출력 인터페이스는 작성하는 방법은 다르나 일반적으로 간단하게 사용하는 형식이다. 그러나 사용되어지는 플랫폼에 자유롭지 못하고, XML의 장점인 다양한 출력 인터페이스의 XSL을 효율적으로 사용하지 못한 경우라고 할 수 있다. 웹에 공유되어 있는 XML Data를 플랫폼에 자유롭게 하기위해 새로운 XSL 스타일시트를 작성하여 저장한다. 화면에 출력하기 위해 HTML에서 각각의 XML Data와 XSL를 연결하여 화면에 그 결과를 출력한다. 공유되어있는 XML Data의 XML포맷을 텍스트 형태로 나타내거나 테이블 형태로 출력을 하는 XSL파일, 필드명의 오름차순으로 정렬하여 나타내는 XSL 파일 등을 작성하여 그 결과를 화면에 브라우저하는 인터페이스이다.

5. XML Data Query

XML 데이터는 XML 파서를 이용하여 데이터를 사용하는 방법과 직접 XML 데이터로 접근하여 데이터를 가지고 가는 방법이 있다. 다음 <그림5.>는 공유되어 있는 XML데이터를 XML Parser와 DDSQ형식으로 원하는 데이터를 커서 형태로 되돌리는 그림이다.



<그림5. XML 데이터에서 원하는 데이터쿼리>

5.1 XML 파서를 이용한 Query

XML 파서를 이용하여 원하는 커서만을 추출하려고 할 때 기존의 SQL문에 원하는 데이터 형식의 쿼리를 만들어 수행하면 된다. 이때 기존의 데이터베이스 소스에서 공유하고자 하는 XML 데이터로 변환하고자 하는 경우에도 원하는 Query문을 생성하여 변환할 수도 있다. 데이터베이스에서 항상 기본 테이블을 사용자의 화면과 출력하기위한 중간 단계로 Query를 작성하여 실행한 결과로서 진행된다. 이 Query는 사용되는 도구마다 약간의 차이점이 있다. 그래서 나온

기본 표준이 SQL문이다. 구조적 질의 언어는 모든 데이터베이스의 기본 질의 형식으로 사용된다. 이미 생성되어진 고유 XML데이터 내용의 변경, 데이터 요소나 속성 목록 변경 등을 하고자 할 때 Notepad를 사용하거나 MS XML Notepad를 사용할 수 있다.

5.2 Direct Data Source Object을 통한 Query

Microsoft사에서 ODBC를 대체하기 위해 설계된 COM(Component object Model)은 인터페이스의 모음으로 DAO(Data Access Object), RDO(Remote Data Object) 그리고 ADO(ActiveX Data Object)등이 있다.

6. 결론 및 향후 연구

데이터로서의 XML은 SGML 및 HTML의 장점을 모두 수용한 것이라고 할 수 있으며, 웹 브라우저에 표시하기위한 것이라기보다는 어떠한 종류의 응용프로그램과도 통합될 수 있다. 하지만 XML은 데이터뿐만 아니라 태그들 까지도 저장해야 하기 때문에 대용량의 자료를 저장하는 데에는 부적합 할 수도 있다.

향후 연구방향은 현재의 최적 솔루션을 확보하는 것도 중요하지만 장기적인 안목을 가지고 XML기반 제반 요소기술 및 각종 솔루션 개발이 필요하다.

[참고문헌]

- [1] 이원석, "XML Introduction", "http://dblab.comeng.chungnam.ac.kr/~dolphin/xml".
- [2] 장근실, 유철중, 장옥배. "JAVA 프로그램의 정보공유를 위한 XML DTD 설계", 제 26회 정보과학회 추계학술대회 논문집
- [3] Mark Johnson, "XML for the absolute beginner", JavaWorld, April 1999.
- [4] 김현기 외 3명, "XML 문서 구조검색을 위한 저장시스템 설계", 정보과학회 학술발표 논문집(B) 제 26권 제1호, 1999
- [5] XML Linking Language (XLINK) "http://www.w3.org/TR/xlink", Working Draft
- [6] 정희경, "XML 가이드", 그림, 1998
- [7] 송계순, "웹환경에서 XML을 이용한 데이터 공유 방법 설계 및 구현", 숙명여대 정보통신 대학원, 2000.08
- [8] Richard Light, "차세대 웹의 혁명 XML", 대림, 1998