정보학연구 제4권 제4호 2001년 12월(pp.89-100)
The Journal of Information Technology, Vol.4, No.4

구조 검색을 위한 XML 문서 저장 시스템

임산송* · 현득창** · 정회경***

요 약

XML(eXtensible Markup Language)은 W3C(World Wide Web Consortium)에서 표준으로 제정, 발표한 대표적인 전자문서 표준이다. XML 문서는 구조화된 정보를 체계적으로 생성하고 전송할 수 있으며, 기존의 파일 형태 정보에 비하여 의미적인 정보 단위를 구조로 표현하고 이러한 구조 정보를 이용해 문서의 관리 및 검색, 저장에 이용할 수 있다. 이에 본 논문에서는 XML의 구조적 정보를 이용하여 저장 검색하기 위한 XML 저장 시스템을 설계 및 구현하였다. 문서의 기본 단위인 엘리먼트(element) 단위로 모델링(modeling)하여 저장하였고, 저장된 XML 정보를 구조 단위로 검색 할 수 있도록 모델링 하였다. 또한 DTD(Document Type Definition)와 인스턴스(instance)에 대하여 스키마(schema)를 생성하여 다양한 문서에 대한 구조를 효과적으로 관리, 저장할 수 있도록 하였다.

Ⅰ. 서론

증가되고 있는 대용량의 멀티미디어 데이터와 전자도서관(digital library) 구축 및 전자상거래(e-Commerce) 등에서 대량의 전자문서 정보를 저장하고 관리하거나 구축에 대한 요구가 발생하면서 대량의 문서 정보를 효율적으로 저장하고 검색하며 관리할 수 있는 정보 서비스의 요구가 점차 증가되고 있다. 이에 XML은 W3C에서 인터넷상에서 전자문서의 효율적인 처리를 위해 전자문서 표준으로 제정하였다[1,2]. XML 문서는 구조화된 정보를 체계적으로 생성하고 전송할 수 있으며, 기존의 파일 형태의 정보에 비하여 의미적인 정보 단위를 구조로 표현하고 이러한 구조 정보를 이용해 문서의 관리 및 검색, 저장에 이용할 수 있는 장점이 있다. 따라서 대량의 구조화된 XML 문서 정보를 이용하여 문서를 보다 효율적으로 사용, 관리하기 위한 연구가 필요하며, XML 문서를 이용한 다양한 응용 프로그램에 대한 지원과 문서의 공유 등을 지원하기 위해 데이터베이스에 기반한 XML 저장 시스템이 요구된다.

이에 본 논문에서는 XML 문서의 구조 정보를 이용한 모델링을 통하여 DTD나 인스턴스의 데이터 정보를 정확하게 표현하고 엘리먼트, 엔티티(entity), 애트리뷰트(attribute) 등 각 노드에 대한 정보를 추출하여 데이터베이스에 효율적으로 저장하고 관리할 수 있는 XML 문서 저장 시스템을 설계 및 구현한다. 또한, 인스턴스에 대한 저장과 색인을 위하여 DOM(Document Object Model)을 이용하여 추출된 각 노드에 색인 값을 부여하기 위한 깊이 우선 탐색 방법을

* 배재대학교 컴퓨터공학과 박사과정
** 극동대학교 정보통신공학부 조교수
*** 배재대학교 정보통신공학부 부교수

적용하고, 색인용 트리를 구성하여 다양한 문서에 대한 구조를 효과적으로 관리, 저장할 수 있다.

II. XML 문서의 데이터 모델링

본 장에서는 XML 문서를 데이터베이스에 저장하기 위해 XML 문서의 논리적 구조를 표현하기 위한 논리적 구조 모델링과 XML 문서의 인스턴스를 저장하기 위한 물리적 구조 저장 모델링에 대하여 설명한다.

2.1 XML 문서의 논리적 구조 모델링

XML 문서가 포함하고 있는 많은 특성들을 데이터베이스에 손실 없이 표현하고 저장하기 위한 데이터 모델은 DTD 독립 스키마 모델(DTD Independent Schema Model)과 XML 문서 데이터 자체의 구조를 표현해 주는 DTD 의존 스키마 모델(DTD Dependent Schema Model)로 분류 할 수 있다[3,4,5].

2.1.1 DTD 독립 스키마 모델링

DTD 독립 스키마 모델은 XML DTD와 인스턴스 사이의 관계에 대한 정보, DTD 자체에 정의된 엘리먼트, 애트리뷰트, 엔티티, 표기법(Notation)등 정보 관리를 목적으로 한다. 또한 DTD 독립 스키마 모델은 잘 구성된 XML(Well-formed XML) 문서와 같이 특정 DTD와 무관하게 임의의 DTD를 따르는 인스턴스나 DTD를 따르지 않는 문서의 인스턴스라도 공통적으로 동일한 구조를 갖는 스키마를 의미하며, DTD 독립 스키마 모델에 정의된 클래스로부터 상속관계를 받는 방식을 통하여 새로운 DTD 모델이 생성된다. DTD 독립 스키마 모델링은 엘리먼트 모델, 에트리뷰트 모델, 엔티티 참조 모델, XML 저장기 모델, DTD 모델, 문서모델 그리고 문서형 정의 모델(DocTypeDef Model)로 구성된다. 엘리먼트, 애트리뷰트, 엔티티 참조에 대한 모델은 한 문서를 구성하고 있는 문서 내부의 구성요소가 포함해야 할 정보를 표현하기 위한 부분이며, XML 저장기 모델은 XML 문서의 구조 정보를 정의하는 DTD와 DTD 구조를 따르는 인스턴스 사이의 관계 정보를 관리하게 된다. DTD 모델은 XML DTD에서 정의된 엘리먼트, 애트리뷰트와 엔티티 그리고 표기법에 대한 정보와 DTD의 식별자로서 사용되게 되는 공용 식별자(Public ID)와 시스템 식별자(System ID) 등의 정보를 관리하기 위해 ID를 둔다. 문서형 정의 모델은 문서에서 doctype으로 선언된 엔티티나 표기법 등에 대한 처리를 위한 부분이며, 문서 모델링은 XML의 인스턴스에 대한 정보를 관리하기 위하여 인스턴스의 루트 엘리먼트에 대한 정보와 인스턴스 내부에 포함한 엔티티 선언이나 표기법 선언에 대한 정보를 갖는 객체들에 대한 처리 정보를 가지게 된다[3,6].

2.1.2 DTD 의존 스키마 모델링

DTD 의존 스키마 모델에서는 특정 DTD가 가지는 모든 정보들이 스키마로써 표현되고 실제 인스턴스를 저장하기 위한 기반을 제공한다. DTD 의존 스키마 모델링은 특정 DTD에 의존하여 생성된 클래스들의 집합으로서 DTD에서 엘리먼트, 애트리뷰트, 엔티티 참조의 정의로 문서의 구조를 표현하는 것과 같이 데이터베이스 안에서 엘리먼트, 애트리뷰트, 엔티티 참조 클래스를 이용하여 문서 구조를 표현한다[3,6].

2.2 XML 문서의 물리적 구조 모델링

XML 문서의 물리적 구조 모델은 XML 인스턴스가 삽입 되었을 때, 실제 인스턴스를 어떻게 저장하는가를 의미하는 부분이다. XML 저장 방법에는 실제 인스턴스의 내용에 대한 저장과 관리를 인스턴스 단위로 할 것인가, 혹은 인스턴스를 구조 단위인 엘리먼트 단위로 나누어 저장할 것인가에 따라 단편화 모델과 분할 모델의 두 가지로 나뉜다.

단편화 모델은 인스턴스를 구조적 단위인 엘리먼트 단위로 나누어 저장하는 방식이 아닌 인스턴스 자체를 저장하는 모델로써 저장된 문서를 이용한 문서의 재구성 보다는 검색 위주의 저장이 요구되는 곳에 적합한 모델링이다.

단편화 모델의 장점은 저장이 용이하며, 문서를 재구성할 필요가 없기 때문에 검색시간이 빠르다. 단점으로는 문서 전체 정보를 한꺼번에 보고자 할 때, 데이터를 각 단말 요소들이 각각 가지고 있으면 이를 모두 검색, 접근하여 나타내어야 하므로 심각한 속도 저하를 가져올 수 있으며, 한 구조 내에서 두개 이상의 단말 노드 내용이 중복될 때 서로 다른 구조 정보에 같은 내용이 여러 번 중복될 수 있다[3,6,7].

분할 모델은 인스턴스를 구조적 단위인 엘리먼트 단위로 나누어 저장하는 방식이다. 대부분 XML 문서 저장 시스템은 분할 모델을 적용하고 있는데, 그 이유는 저장기에서 문서 관리 시스템 등으로의 확장 시 시스템 구조변경을 최소화 할 수 있기 때문이다.

분할 모델의 장점은 엘리먼트 단위로 나뉘어 저장되기 때문에 문서의 재구성이 용이하며, 한 문서에 대해 동시에 여러 사용자가 편집할 수 있고, 문서의 일부분만 재 편집하거나 추출한 문서를 다시 통합하는 과정에서 정보의 손실 없이 교환이 가능해진다. 단점으로는 엘리먼트 단위로 나뉘어 저장한다면 데이터베이스에 저장하는데 시간이 많이 걸릴 수 있으며, 저장 구조가 복잡해진다. 또한 분할된 엘리먼트로 부터 문서를 재구성해야 하기 때문에 검색 결과를 생성하는데 오랜 시간이 걸린다[4,6].

III. XML 문서 저장 관리기 설계

본 장에서는 XML 문서 논리적 구조인 엘리먼트, 애트리뷰트, 그리고 엔티티에 대한 모델링과 XML 문서 저장 시스템의 설계와 각 모듈을 설명한다.

3.1 문서 모델링

3.1.1 엘리먼트 모델링

XML에서 엘리먼트는 문서의 구조를 정의할 때 구조 표현의 단위로 사용되는 아주 중요한 구성요소이다. 엘리먼트 모델에서는 엘리먼트의 트리상의 위치 정보, 속성 정보, 자식 노드의 순서 관계 등을 포함하여야 한다.

XML 엘리먼트 구문 모델은 크게 단말 엘리먼트와 비단말 엘리먼트로 구분되고 여기에 EMPTY 엘리먼트와 ANY 엘리먼트를 추가 할 수 있다. 단말 엘리먼트는 #PCDATA와 같은 데이터 타입을 가질 수 있으며 엔티티 참조와 속성을 가진다. 비단말 엘리먼트는 하나 이상의 하위 엘리먼트를 가질 수 있고, 이 하위 엘리먼트는 단말 엘리먼트처럼 #PCDATA와 엔티티 참조, 애트리뷰트를 가진다. EMPTY 엘리먼트는 애트리뷰트 선언을 이용하여 참조 데이터의

근원을 식별하기 때문에 애트리뷰트를 갖는다. ANY 엘리먼트는 단말노드와 같은 하위 클래스들을 가질 수 있으나 ANY는 어떠한 내용이 와도 수용할 수 있기 때문에 단말 엘리먼트와 다르게 새로운 하위 클래스들을 둔다.

3.1.2 속성 모델링

속성은 특정 엘리먼트의 특징을 정의하는 것으로 속성 이름과 타입, 값에 대한 관리가 필요하다. XML 문서의 속성 타입으로 문자열, 토큰화된(tokenized) 타입, 나열형(enumerated) 타입으로 나뉠 수 있다. 본 논문에서는 속성에 대한 모델링의 선언값이 문자열로 선언이 된 것인가 아니면 숫자형으로 된 것이냐를 분리하여 모델링 하였다.

속성 클래스는 속성의 이름과 선언한 값을 위한 이름, 선언 값 필드와 XML 엘리먼트의 이름, 선언 값의 삽입, 삭제, 변경 연산을 포함하며, 숫자형 속성 클래스는 속성 값을 숫자로 가지기 위해 정수형의 리스트로 선언된 값 필드와 속성값의 삽입, 삭제, 변경 연산을 가지고, 문자열 속성 클래스는 속성 값을 문자열로 관리하기 위해 문자열 형태의 리스트 형으로 선언된 값 필드와 속성 값의 삽입, 삭제, 변경 연산을 가진다.

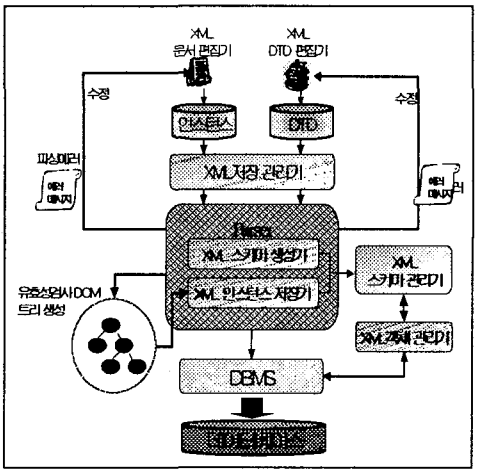
3.1.3 엔티티 모델링

엔티티는 문서 내에서 참조할 수 있는 문자 집합의 단위로 본 논문에서는 세가지 방법으로 분류하여 모델링 하였다. 엘리먼트는 엔티티 참조를 할 수 있는데, 이러한 엔티티 참조는 내부 엔티티 참조와 외부 엔티티 참조 그리고 비 XML 외부 엔티티 참조로 구성된다. 외부 엔티티 참조에는 엔티티 집합(Entity Set)을 포함하고 있으며 이 엔티티 집합은 개별 파일로 구성되어있다. 비 XML 외부 엔티티 참조는 참조되는 내용이 외부 파일로 존재한다.

엔티티 참조 클래스에서는 엔티티 이름과 그 엔티티를 참조하고 있는 엘리먼트 리스트와 엔티티 타입 그리고 참조 엘리먼트에 대한 정보를 속성으로 갖는다.

비 XML 외부 엔티티 참조에서는 참조되는 내용이 외부 데이터 파일로 존재하므로, 해당 파일은 엔티티 정의(Entity Definition)을 통해 얻을 수 있게 하였다.

3.2 전체 시스템 구조



(그림 1) XML 저장 시스템의 구조

그림 1은 전체 시스템 구조를 보여주고 있다. 인스턴스와 DTD가 입력되면 공용 구문 검색기를 사용하여 문서 유효성 검사를 수행한다. 이때 유효한 문서이면 문서의 파싱 정보가 생성되어 스키마 생성기에 보내어져 스키마를 생성한다. 오류가 발생하면 오류 메시지를 출력하게 된다. 그리고 유효성 검사를 마친 문서에 대해서는 DOM 인터페이스를 이용하여 파싱 트리를 생성하고 각각의 테이블 정보를 추출하여 데이

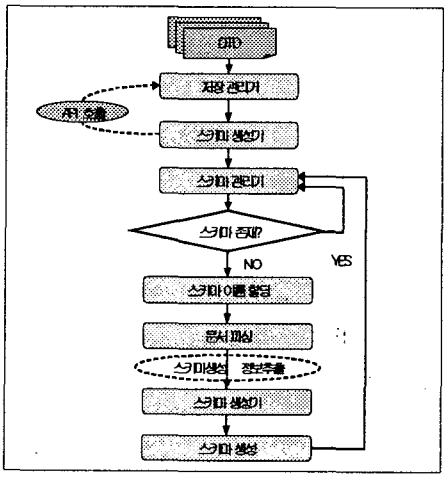
터베이스에 저장한다. 본 시스템에서는 DTD의 내용과 인스턴스의 내용을 DTD와 content 테이블에 저장하도록 설계되었다. 각 모듈들에 대한 설명은 다음과 같다.

- 저장 관리기 모듈은 각 모듈들에 대한 통합 인터페이스를 제공하는 것으로 이는 DTD 독립 스키마의 생성부터 새로운 DTD에 대한 스키마 생성, 새로운 XML DTD나 인스턴스에 대한 정보 반환 등의 기능을 수행한다.
- XML 스키마 관리기 모듈은 DTD, 인스턴스, 분산된 저장소에 대한 포괄적인 정보를 관리하는 것으로, 특정 스키마에 대한 저장소가 어디에 있는지, 특정 XML 인스턴스가 어떤 저장소에 있는지 등에 대한 각 DTD의 스키마 이름, 각 엘리먼트에 대응되는 클래스들의 이름, 저장소 ID 등을 할당하고 관리한다.
- XML 객체 관리기 모듈은 찾고자 하는 XML 객체에 대한 조건을 받아 스키마 관리기를 통해 질의해야 할 저장소의 범위를 결정한다. 결정된 질의 대상 저장소에 적합한 질의 문을 생성하고, 각 저장소에 대한 질의 문을 객체 관리기에 전달한다.
- 스키마 생성기 모듈은 입력된 DTD를 파싱하여 스키마 생성에 필요한 정보를 추출하고 일반 클래스에서 제공하는 엘리먼트, 애트리뷰트, 엔티티에 대한 클래스를 기반으로 스키마를 생성한다. 그리고, 생성된 스키마에 대한 정보를 스키마 관리기에 등록한다.
- 인스턴스 저장기 모듈은 인스턴스를 미리 생성되어 있는 DBMS의 스키마에 복합 객체 형태로 저장한다. 그리고, 인스턴스에 대한 정보를 스키마 관리기에 등록한다.
- 인스턴스 관리기 모듈은 인스턴스 저장기에서 생성된 인스턴스에 대한 정보를 관리하며 또한 인스턴스를 파싱하여 생성된 인스턴스 객체 트리에 대한 관리기능을 한다.
- 단순 질의 검색 모듈에서는 저장된 인스턴스에 대한 정보를 검색할 수 있게 해주는 모듈로서 질의를 생성하면 저장 관리기가 스키마 생성기로부터 ID를 받아 객체 관리기에서는 질의 수행 결과로 인스턴스를 가져온다.

3.3 XML 문서 저장 관리기 모듈 설계

본 절에서는 XML 문서 저장 관리기의 주요한 기능인 스키마 생성 모듈과 인스턴스 저장 모듈, 단순 질의 검색 모듈에 대하여 설명한다.

3.3.1 스키마 생성 모듈



(그림 2) 스키마 생성 과정

그림 2는 스키마 생성 과정을 보여주는 그림으로, 새로운 문서를 저장하기 위해 새로운 스

키마의 생성이 필요하고, 새로운 DTD를 입력받아 저장 관리기가 스키마 생성기의 API를 호출한다. 스키마 생성기는 스키마 관리기를 호출하며, 이때 새로 생성하려는 스키마가 이미 존재하는지에 대한 검사를 하고 존재하지 않으면 스키마 이름을 할당받는다. 파서를 이용하여 문서를 파싱한 후 스키마 생성에 필요한 정보를 추출하여 스키마 생성기에서 스키마를 생성한다. 생성된 스키마는 스키마 관리기에 보내져 관리하게 된다.

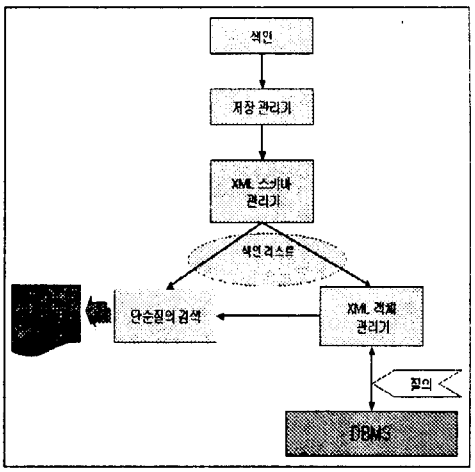
3.3.2 인스턴스 저장 모듈

인스턴스 저장기는 저장 관리기로부터 인스턴스를 받아 저장하는 모듈이다. 인스턴스가 입력되면 일반 스키마와 파서를 이용하여 파싱한 후 DOM 인터페이스를 이용하여 트리를 생성하고 깊이 우선 탐색 방법을 통해 인덱싱(indexing)한다. 이때 깊이 우선 탐색 방법을 이용하여 문서에 대한 구조 검색 시 노드의 시작 값과 종료 값을 인덱스로 가지게 되어 자식, 조상 노드에 대한 구조적 검색을 가능하게 해준다.

3.3.3 단순질의 검색 모듈

저장된 XML 문서의 검색을 위하여 검색기 모듈을 가지게 되는데, 그림 3은 검색 흐름도를 보인다.

색인 이벤트가 발생하게 되면, 저장 관리기 모듈이 XML 스키마 관리기 모듈에게 이를 요청하게 된다. 이때 스키마 관리기는 색인을 위한 리스트를 단순 질의 검색부와 XML 객체 관리기에게 전달한다. 스키마 관리기로부터 받은 리스트를 XML 객체 관리기는 DBMS에 적절한 질의를 전달하게 되며, 그 수행 결과를 다시 XML 객체 관리기가 받아 단순 질의 검색부로 그 결과를 보내주게 된다.



(그림 3) 검색 흐름도

3.4 문서 저장 시스템 저장 구조

표 1은 문서의 논리적 구조 모델링을 이용한 문서 저장 시스템의 테이블 구조를 보여준다.

테이블 구성은 XmlReposit, DTD, EID, Document, Element, Attribute, Content, ExternalFiles로 구성되어 있으며, 각 테이블의 역할은 다음과 같다.

- XmlReposit 테이블은 저장기 테이블들의 최상위로서 현재 저장기의 버전(Version), 제작자(Creator), 저장된 DTD의 총 수(DtdNum)를 표시해 주는 기능을 가지고 있다.
- DTD 테이블은 새로운 DTD의 저장을 위해 저장 시스템이 DTD 테이블을 새롭게 생성하게 되는데, DTD 테이블은 동일한 DTD를 가지고 생성되는 문서들의 최상위로서 DTD를 식별하는 식별자(DtdId)와 DTD의 URL(DtdURL), 공용 DTD 인지를 나타내는 System, 그리고 현재 이 DTD를 이용하여 생성된 문서의 총 개수를 가지고 있어 이후 생성될 문서의 DTD 식별자를 할당하

는데 사용되며, 실제 DTD의 내용을 저장하는 content를 가지게 된다.

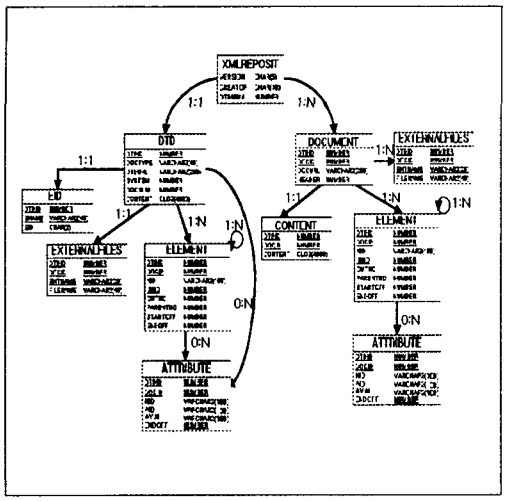
내용이 저장되는 테이블로서 문서 식별자와 DTD 식별자 및 문서의 구조정보를 알 수

〈표 1〉 저장 테이블 구조

XmlReposit	DTD	EID	Document	Element	Attribute	Content	Externalfile
DtdNum	DtdId	Eid	DocURL	DtdID	DtdID	content	Entname
Version	DtdUrl	DocID	DtdID	DocID	DocID	DtdId	DtdID
Creator	DocNum	Ename	DocID	NID	NID	DocID	DocID
	Content		Header	Inno	AID		FileName
	System			Outno	AVAL		
				PatentNo	EndOff		
				StartOff			
				EndOff			

- EID 테이블은 DTD가 새로운 DTD 식별자를 가지고 생성 되는데, DTD에서 발생하는 모든 엘리먼트 이름들에 대해서 각각의 EID를 가지게 된다. EID는 2바이트 크기를 가지며, 발생하는 엘리먼트들에 대해 순차적으로 할당한다.
- Document 테이블은 하나의 문서에 대한 최상위 역할을 하기 위한 것이다. 어떤 DTD를 가지는 문서인지를 알기 위해 DTD 식별자와 현 문서에 새롭게 할당된 문서 식별자, 이 문서의 URL, 그리고 문서에서 문서 트리를 나타내는 이전 부분인 Header를 가지게 된다.
- Element 테이블은 실제적인 문서의 구조 정보를 지니게 되는 테이블들이다. DTD 식별자와 문서 식별자를 사용하여 어떠한 문서의 일부분인지를 알 수 있게 되며, 문서의 구조적 정보를 나타내는 NID, Inno, Outno가 있고, 부모 노드를 찾을 수 있도록 부모에 대한 Inno인 ParentNo와 이 엘리먼트의 실제 내용의 시작 포인터(StartOff)와 끝 포인터(EndOff)가 있다.
- Attribute 테이블은 저장되는 애트리뷰트의 있는 NID와 그 NID에 따르는 각각 애트리뷰트의 값을 저장하는 AVAL과 엘리먼트에 대응되는 식별자 값 AID로 구성된다.
- Content 테이블은 문서의 실제 내용이 저장되는 테이블로서 어떤 DTD의 어떤 문서의 내용인지를 나타내기 위해서 DTD 식별자와 문서 식별자를 가지고 있고, 실제 내용을 content에 저장한다.
- ExternalFiles 테이블은 외부 파일에 대한 정보를 저장하는 테이블로 외부파일의 엔터티 이름과 외부 파일명을 갖고 있다.

그림 4는 테이블들의 상호 연관 관계를 나타내는 테이블 관계도이다.



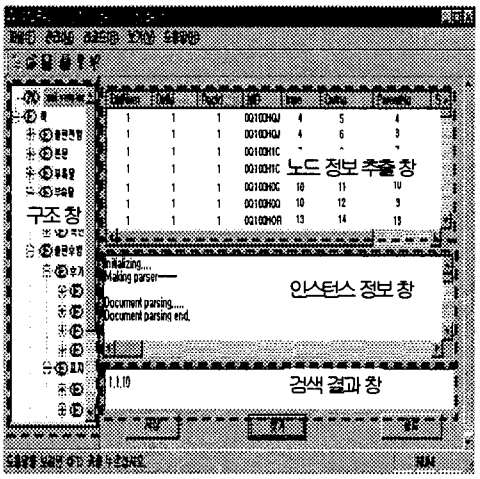
(그림 4) 테이블 관계도

Ⅳ. 시스템 구현 및 고찰

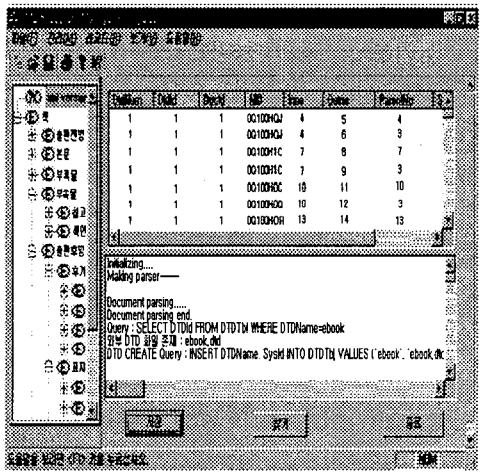
4.1 구현

XML 저장 시스템의 구현 환경으로는 운영체제로 Windows NT 4.0과 오라클 8.0.5과 Visual C++6.0을 이용하여 구현하였다. 또한 문서의 유효성 검사를 위해 공용 구문 검색기를 사용하였다. DOM 인터페이스를 이용하여 인스턴스 트리를 생성하여 깊이 우선 탐색 방법으로 인덱스를 생성하였다.

XML 문서 저장 시스템은 DTD나 XML 문서를 열어 유효성 검사를 한 후 각각의 값들을 추출하여 보여준다. 그림 5는 XML 문서 저장 시스템의 사용자 인터페이스(User Interface)이다. 사용자 인터페이스의 구성은 4개의 창으로 구성되어진다. XML 문서의 구조를 보여주는 구조창과 유효성을 검사 한 문서가 가지는 각 노드의 정보를 표현해 주는 노드 정보 추출 창, 저장 될 인스턴스를 보여주는 인스턴스 정보 창 그리고 마지막으로 질의어 처리에 대한 결과를 보여주는 검색 결과 창으로 나누어 볼 수 있다.



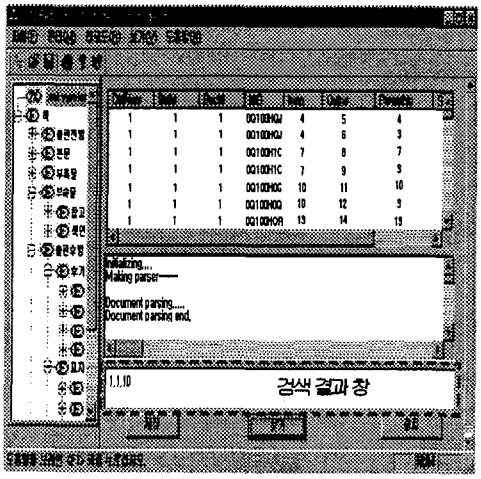
(그림 5) XML 문서 저장 시스템 사용자 인터페이스



(그림 6) 저장 시스템에 검증된 문서 정보 추출

그림 6은 문서의 유효성 검사를 통하여 추출되어진 노드 정보들과 인스턴스에 대한 내용을 보여주는 검증된 문서 정보 추출 그림이다.

이렇게 추출된 노드 정보와 인스턴스를 데이터베이스에 저장하며, 저장된 데이터에 대해 필요시 질의어를 입력받아 처리하게 되는데 이의 처리 예를 그림 7에 보인다.



(그림 7) 검색 결과 창

4.2 고찰

본 시스템은 구조 정보를 가지는 XML 문서에 대한 특성을 그대로 유지하면서 이를 효과적으로 저장하고 검색할 수 있는 XML 문서 저장 시스템을 설계 및 구현하였다.

본 논문에서는 XML 문서의 데이터 모델링을 DTD에 대한 정보를 유지하기 위한 부분과 인스턴스에 대한 정보를 유지하는 부분으로 나누어진 모델링을 반영하였으며, 특히 인스턴스에 대한 정보를 유지하는 부분은 특정 DTD의 구조를 분석하여 동적으로 생성되게 함으로서 XML 인스턴스를 효과적으로 관리할 수 있도록 하였다. 따라서, 기존의 XML 문서 저장 시스템에 비해 본 시스템이 가지는 특징 및 장점은 다음과 같다.

첫째, 시스템의 설계 방향을 구조적 문서의 효율적인 저장에 큰 의미를 가지면서 엘리먼트 단위로 저장을 목적으로 하기 때문에 문서의 재구성 측면에 있어서 효율적이다.

둘째, 객체 지향 모델링을 통한 저장이 이루어지므로 기존 관계형 데이터베이스가 테이블로 모든 데이터를 모델링 하므로 계층 구조를 가지는 트리로 표현될 수 있는 XML 문서를 모델링하기에는 적합하지 않은 단점을 극복할 수 있다. 그리고 저장된 내용을 추출하여 문서의 재구성 시 다수의 테이블에 대한 조인 연산 비용의 증가에 대한 부담을 해소하여 주었다.

셋째, 본 시스템에서는 질의어 처리를 가능하게 하기 위해 시스템 구성에 있어서 단순 질의 검색 모듈을 두어 SQL(Standard Query Language) 기능을 지원하여 질의를 할 수 있도록 하였다.

넷째, 엘리먼트 단위로 문서 저장을 하는 방법을 사용하였기 때문에 문서 저장 시스템에서 문서 관리 시스템으로서의 확장에 용이한 장점을 갖는다.

그러나, 문서를 구조 단위로 나누어 저장하기 때문에 문서 전체를 저장하는 방식에 비하여 저장 구조가 복잡해지며, 문서의 검색에 있어서 검색 시간이 많이 걸린다.

V. 결론 및 향후 연구과제

최근 증가되고 있는 대용량의 멀티미디어 데이터와 전자도서관 구축 및 전자상거래 등에서 대량의 전자문서 정보를 저장하고 관리하거나

구축해야 할 요구가 발생하고 있다. XML은 대표적인 전자 문서 표준으로 전자문서의 논리적 계층구조를 정확히 정의할 수 있으며, 멀티미디어 및 하이퍼미디어 정보의 표현도 가능하게 하고 있다. 또한 XML 문서는 구조화된 정보를 체계적으로 생성하고 전송할 수 있으며, 의미적인 정보 단위를 구조로 표현하고 이러한 구조 정보를 이용해 문서의 관리 및 검색, 저장에 적합하다.

본 논문에서 제시된 모델링의 기본 원리는 XML 문서가 가지고 있는 문법 구조를 따르는 구조 정보 모델링을 이용하여 DTD나 문서의 인스턴스 데이터 정보를 쉽게 표현해 줄뿐만 아니라 그들 사이의 관계, 즉 엘리먼트, 애트리뷰트, 엔티티 등의 각 노드에 따라 모델링을 설계하였다. 그리고 구조적 문서의 가장 기본이 되는 엘리먼트 단위로 나누어 저장하였으며, XML 문서의 효율적인 저장뿐 아니라 이를 관리할 수 있는 XML 저장 관리 시스템을 설계하여 DBMS에서 제공하는 다양한 제반 기능들을 기반으로 대용량의 XML 문서 처리 및 공유와 부분적인 XML 객체의 추출 및 관리를 가능하게 하였다. 그리고 인덱스 정보를 갖는 트리 노드를 구현함으로써 저장된 문서에 대한 정보를 기반으로 SQL을 지원하여 트리 인덱스를 이용하여 해당 노드에 대한 정보와 내용을 추출할 수 있는 검색 기능을 제공하고 있다. 따라서 본 시스템은 전자도서관의 구축기반 기술로 사용될 수 있으며, 전자상거래, EDI 등의 분야에서 전송 및 수신 문서에 대한 데이터 저장소로 사용될 수 있다.

기존의 시스템들과 비교해 볼 때, 본 시스템은 다양한 데이터 지원을 가능하게 해주면서 문서 재구성을 용이하게 해준다. 또한 문서 편집기 등과의 연계를 위해 시스템 확장에 용이하게 설계하였다. 또한 계층화된 엘리먼트의 포함관계 및 상관 관계를 고려한 엘리먼트의 구조 검색을 가능하게 해주는 특징을 가지고 있다.

이를 바탕으로 한 향후 연구방향은 본 논문에서 제시한 XML 문서뿐만 아니라 XSL(eXtensible Stylesheet Language), XSLT(eXtensible Stylesheet Language Transformations) 등 다양한 문서들에 대한 데이터 모델링이 요구되며, 간단한 SQL 기능을 지원하는 검색에서 벗어나 XML 질의 언어인 XQL(XML Query Language) 기능에 대한 지원이 필요할 것이며 이질적인 분산 데이터베이스 환경을 고려한 연구가 필요하다.

(본 논문은 과학재단 98특정기초연구과제 연구비에 의하여 연구되었음)

참고문헌

[1] W3C, "Extensible Markup Language 1.0," http://www.w3.org/TR/1998/REC-XML-19980210, 1998.

[2] 정회경, "WWW 문서 작성을 위한 차세대 언어 XML 가이드", 그린.

[3] 이원석, 대량의 구조화 문서 관리를 위한 SGML 저장 관리기의 설계 및 구현, 충남대학교 대학원, 2.1998

[4] Carl-Christian Kanne, Guido Moerkotte: Efficient Storage of XML Data., ICDE 2000: 1998

[5] Ralf Behrens: A Grammar Based Model for XML Schema Integration., BNCOD 2000: 172-190

[6] Takeyuki Shimura, Masatoshi Yoshikawa, Shunsuke Uemura: Storage and Retrieval of XML Documents Using Object-Relational Databases. DEXA 1999

[7] Sandeepan Banerjee, Vishu Krishnamurthy, Muralidhar Krishnaprasad, Ravi Murthy: Oracle8i - The XML Enabled Data Management System., ICDE 2000: 561-568

[8] 김현기, 이상기, 주종철 SGML/XML 문서관리 시스템의 설계 및 구현 한국 정보 처리학회 추계 학술 발표 논문집 제 5권 2호 pp1251-1254,1998

[9] Document Object Model (DOM), Level 2 Specification.W3C Candidate Recommendation. Available at http://www.w3.org/TR/1999/CR-DOM-Level-2-19991210

[10] Ronald Bourret, Java Packages for Transferring Data between XML Documents and Relational Databases XML-DBMS, Version 1.0, 1999 Technical University

[11] Alin Deutsch, Mary Fernandez, Dan Suciu, "Storing Semistructured Data with STORED," SIGMOD '99 Philadelphis PA, pp. 431-442, 1999.

[12] Chaitanya K. Baru, Vincent Chu, Amarnath Gupta, Bertram Ludscher, Richard Marciano, Yannis Papakonstantinou, Pavel Velikhov: XML-based Information Mediation for Digital Libraries., ACM DL 1999: 214-215

[13] P. Francois, Generalized SGML repositories : Requirements and modeling, Computer Standard & Interface, 1996

[14] Oracle Corporation, XML Support in Oracle 8 and beyond, chinal white paper, http://www.oracle.com/xml/documents

[15] 정희경 외 2인 공저, "SGML 가이드", 1997, 사이버 출판사

XML Document Repository System for structured retrieval

San-Song, Lim* · Deuk-Chang, Hyun** · Hoe-Kyung, Jung***

Abstract

XML (eXtensible Markup Language) is selected and published as a representative standard of electronic documents by W3C (World Wide Web Consortium). The structured information can be created and also transferred in XML documents. By utilizing XML, you can express the meaningful information unit as a structure comparing existed file typed information. With structured information, you can also manage, retrieve, and reposit documents. According to the above facts, in this paper, it is the purpose to design and implement XML documents repository system to reposit and retrieve using structured information of XML documents. As a model it was designed to be stored by element unit which is the basic unit of documents and was also designed to retrieve the stored XML information by structured unit. It was, especially, designed to manage and reposit the structure of various documents effectively through creating schema as to DTD(Document Type Definition) and instance.

* Dept. of Computer Engineering, Graduate School, Paichai Univ.,
** Division of Information Communication Engineering, KeukDong Univ.,
*** Division of Information Communication Engineering, Paichai Univ.