2001년 한국정보처리학회 추계 학술발표논문집 제8권 제2호

웹 출판 프레임워크를 지원하는 XQL기반 XML 문서 검색 시스템 설계 및 구현

문종환*, 김철원*
*호남대학교 컴퓨터공학과
e-mail:pobimoon@korea.com

A Design and Implementation of XML Document Retrieval System Based on XQL supporting Web Publishing Framework

Jong-Hwan Moon*, Cheol-Won Kim*
*Dept of Computer Engineering, Honam University

요 약

정보의 구조적 표현 가능하고 인터넷을 기반으로 하는 정보교환의 매개체로써 다양한 응용분야에 확산되고 있는 XML(eXtensible Markup Language)은 차세대 인터넷 전자문서 표준으로 주목받고 있다. 최근들어 기존 문서를 XML로 변환하거나 신규 문서를 XML로 작성하는 사례가 늘면서 대량의 XML문서가 생성되고 있으며 이에 따라 대량의 XML문서를 효율적으로 검색하기 위한 XML 검색 시스템이 요구되어지고 있다.

본 논문에서는 내용기반, 구조기반, 속성기반 검색을 지원하는 XML 문서의 질의언어로 제안되어진 XQL과 XML 문서를 분석하는 문서 구조 처리기, 사용자 입력 질의를 실행하기 위한 질의 언어 처리기를 가지는 XML 문서검색 시스템을 제시하고 검색된 문서의 결과를 웹 출판 프레임워크(Web Publishing Framework)인 Cocoon을 적용하여 다른 포맷의 문서로 전환 가능한 시스템을 설계 및 구현하였다.

1. 서론

최근 웹의 발전으로 인하여 인터넷의 사용과 정보의 양이 급증하고 있으며, 인터넷 상의 정보를 효과적으로 사용하고자 하는 연구가 진행중이다. 지금까지는 인터넷상의 대부분의 정보가 HTML 문서로 한정되어 있었으나 HTML은 단지 문서의 재현을 위한 정보를 하나의 DTD(Document Type Definition)를 사용하여 나타내기 때문에 각 문서의 엘리먼트를 의미 있는 정보로 표현하는 기능이 부족하고 효과적인 문서검색이 어렵다. 따라서, 인터넷 상의 문서들을 저장하고 플랫폼이 독립적이며 구조화된 문서의 전송 및 교환을 용이하도록 하는 표준이 필요하게 되었다. 그로 인해 기존의 구조화된 문서표준인 SGML(Standard Generalized Markup Language)의 너무나 복잡하다는 단점과 HTML이 구조화된 문서를 표현하기에는 부족하다는 단점을 보완하여 1998년 W3C(World Wide Web Consortium)에 의해 차세대 웹 문서의 표준으로 XML (eXtensible Markup Language)을 제안하였다.[3]

XML은 현재 W3C에서 제안된 국제 표준의 전자문서 메타언어[3][4][5]로서 개발자와 사용자가 쉽게 접근할 수 있는 설계기법과 단순기법을 채택하고 있으며, 웹에서 구조화된 문서를 표준화된 텍스트 형식으로 전송하도록 하여 문서를 구성하는 각 요소들의 독립성을 보장하게 함으로써 문서의 호환성, 내용의 독립성, 요소변경의 용이성 등의 특성을 제공한다.

XML의 응용분야는 다양하며 현재 인터넷 웹 문서뿐만 아니라 전자도서관, CSCW(Computer Supported Cooperative Work), 그리고 CALS(Commerce At the Light Speed)를 포함한 다양한 분야에서 XML을 활용하고자 폭 넓은 연구를 하고 있으며, 수학 분야의 MathML (Mathematical Markup Language), 채널기술의 CDF(Channel Definition Format), 이동 통신에서의 WML(Wireless Markup Language)등의 응용사례가 늘고 있다.[4]

이와 같이 전 세계적으로 XML에 대한 관심이 고조되고 실제 많은 분야에서 활용되고 있고 있으며 향후 정보의 생성, 재사용, 처리 및 지속성, 이식성 등 XML 문서를 효과적으로 관리하고 검색할 수 있는 시스템에 관한 많은 연구들이 진행되어지고 있는 중이다.

본 논문에서는 웹 상의 분산된 XML 문서를 병합하여 파싱한 후 문서의 구조정보를 트리 구조로 구성하고 질의언어로 제안되어진 XQL을 이용하여 질의함으로써 XML 문서에 대한 구조 검색과 내용 검색이 가능하도록 설계하였다. 또한 그 결과를 웹 출판 프레임워크인 Cocoon을 이용하여 여러 가지 포맷의 문서로 변환하여 제공하므로써 사용자의 편리성과 동시에 여러 개의 컨텐츠 운용 가능성을 제시하였다. 이를 이용한 응용분야로는 향후 XML 문서의 검색과 관리, 문서의 재사용 분야 등에서 활용될 수 있는 요소기술로서 기대된다.

2. 관련연구

2.1 XML 문서 질의어

기존의 XML 문서 저장 및 검색시스템에 대한 연구는 객체 지향 DBMS를 사용하는 eXcelon, POET XML Repository 등이 있으며, 관계형 DBMS를 사용하는 XDMS, Oracle8i 등이 있다. eXcelon은 단순성과 유연성과 같은 XML 장점을 살릴 수 있는 응용 프로그램을 개발하도록 하는 XML 데이터 서버이다. 그러나 eXcelon은 모든 데이터를 XML 이라는 하나의 논리적 관점에서 취급한다. XML 검색 엔진으로 Milner가 개발한 SCOOBS가 있다. 이것은 각 용어들을 엘리먼트들과 연결시켜 나타내고, 역 인덱스 파일 구조를 가진다.

현재 XML 문서를 검색하기 위해 질의어로 사용하기 위한 XQL은 XSL(eXtensible Stylesheet Language)의 패턴 언어를 확장한 형태로 XSL의 노드에 대한 색인을 추가했고, 질의와 패턴을 사용하는데 단순한 구문을 사용하여 간결하고 간단하다는 것이 특징이다. XML-QL은 현재 W3C에 Note로 제출된 상태이며, 대용량의 문서에서 데이터를 추출하고 , 여러 문서사이에 데이터를 주고받고, 한 문서에서 다른 문서로 내용을 변환하고, 문서간의 통합을 어떻게 할 것인가에 초점을 맞춘 언어이다.

XQL은 XML문서의 엘리먼트와 텍스트를 필터링하기 위한 표기법이다. XSL(eXtensible Stylesheet Language) 패턴 문법으로도 확장 가능하며, 특정 노드와 엘리먼트를 검색하고 지정하기 위한 간결하고 쉬운 표기법으로 문장이 간결하고 이해하기 쉬우며, 문법은 단순하게 한다는 설계목표를 가지고 만들어졌다. XQL은 쉽게 파싱될 수 있어야 하며 XML 문서 내에서 노드들을 식별할 수 있어야 하고 질의는 여러 개의 결과를 나타낼 수 있어야 하는 여러 가지의 설계 표를 가지고 만들어졌다. XQL은 문서 내에서의 특정 연관된 노드들의 집합인 정보를 검색하기 위한 표기법이다. 이 표준에서는 출력에 대한 형식을 지정하고 있지 않으며, 질의에 대한 결과는 한 개의 노드, 노드 목록, XML 문서, 배열 또는 다른 구조가 될 수 있다. 일부 구현에서는 질의에 대한 결과가 XML 문서 또는 트리 구조가 될 수도 있다. 본 논문에서는 XML 검색 질의어로 XQL을 사용하여 검색할 수 있도록 한다.

2.2 웹 출판 프레임워크(Cocoon)

Cocoon은 웹 컨텐츠를 제공하기 위하여 W3C 기술인 DOM, XML, XSL을 이용한 자바 기반의 웹 출판 프레임워크이다. Cocoon은 서버측에서 XML을 처리하여 웹정보를 생성하고, 생성된 결과를 클라이언트 쪽에 전달하는 서버측 XML 처리기법을 이용한 시스템으로 클라이언트에서 독립적인 시스템을 구성할 수 있는 장점을 가지고 있다. Cocoon은 문서를 내용, 스타일, 로직(Logic)의 3개의 층으로 완전히 분리하고 3개의 층이 독립하여 설계되고, 생성되며 독립적으로 관리할 수 있도록 함으로써 시스템 관리 비용과 오버헤드를 줄이고, 문서의 재 이용을 늘리면서 시간의 소비를 줄일 수 있도록 한 시스템이다.

현재 웹의 대부분은 HTML로 작성되어 있고 보통 태그(tag)로 정의된 프리젠테이션의 형태이며 서버쪽의 프로그램과 클라이언트의 프로그램 사이의 정보와 태그가 분리되어 있지 않다. Cocoon은 문서를 내용과 스타일, 로직으로 서로 분리 가능한 XML파일이 독립되어 작업할 수 있는 환경을 제공하며 그들을 다시 병합하여 표현하는 XSL 변환을 적용한다. 실질적으로 정적 혹은 동적으로 생성되는 XML 문서를 HTML문서로 자동 변환해 주는 작업을 하며 XML 문서에 XSL-FO(eXtensible Stylesheet Language - Formatting Objects)를 적용한 PDF, 무선인터넷 WAP에서 사용하는 WML, XHTML 등으로 문서 포맷을 변환하여 여러 형태의 문서 표현이 가능하도록 하며, XML과 웹에서 여러 가지 형태의 프리젠테이션을 위한 XSL 등을 적용한 동적 XML을 클라이언트에 제공한다. 이러한 시스템은 동시에 여러 개의 컨텐츠를 운영하거나 여러 개의 프리젠테이션을 제공하는 시스템에서의 관리비용 절감과 여러 형태의 문서 표현을 제공하는 웹사이트의 구축이 가능하고 서로 분리된 작업을 최소한으로 축소시켜 관리의 효율성을 제공한다.[6]

3. XML 문서 검색 시스템 설계 및 구현

본 논문의 문서 검색 시스템은 자바로 구축하였으며, JDK1.3, SUN사의 DOM과 SAX를 지원하는 파서와 XQL을 기반으로 설계하였다. 그림 1은 본 논문에서 제안한 XML 문서 검색 시스템의 전체적인 구성도 이다.

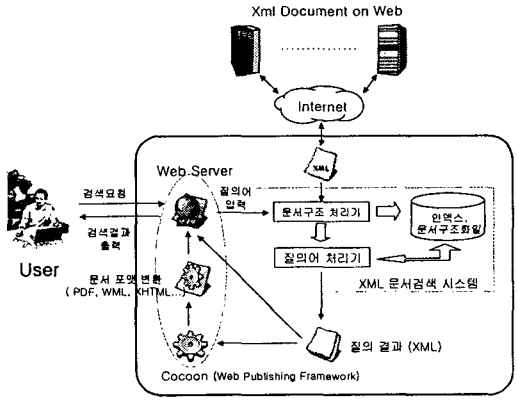


그림 1. Conceptual Diagram of XML Document Search System

3.1 XML 문서 검색 시스템 구성

웹 상의 분산된 Well-formed XML 문서를 입력으로 취하고 문서 구조인 처리기에 의해 파싱한 후 내부 데이터 구조인 인덱스와 문서구조 정보를 구성하고 그 정보를 XML 질의어 처리기의 문서 검색을 위해 저장한다. 문서가 입력될 경우에는 문서 단위로 구분하여 인덱스 구조를 등록하고, 각각의 인덱스 구조를 생성한 후에 유지 관리하게 된다. 검색된 결과에 대한 구조 정보는 인덱스 정보와 함께 스택에 유지되어지며, 결과에 대한 질의를 계속하여 처리할 수 있도록 인덱스 정보를 계속 유지한다.

3.2 문서 구조 처리기

이 처리기는 XML 문서를 파싱하면서 인덱스 구조정보를 생성시킨다. Well-formed XML 문서를 입력받아 스택과 구조정보 및 텍스트 내용정보 등에 관하여 초기화 한 후, start document, end document, start element, end element, 텍스트(내용)들을 분류하여, XML 문서를 인덱스된 데이터 구조로 변환시킨다. 각 엘리먼트를 노드로 하며, 트리 진행순서는 입력된 문서의 순서에 따라 진행하고, 동시에 스택을 운영한다. 그림 2에서 트리 구조는 XML 문서의 구조를 의미하며, 노드 안의 번호는 트리 진행순서이다. 이것은 XML 문서의 순서와 일치하며 화살표는 세 종류로서 PUSH, POP/PUSH, POP으로 구성되며, 이것은 스택을 운영할 때의 상태를 의미한다.

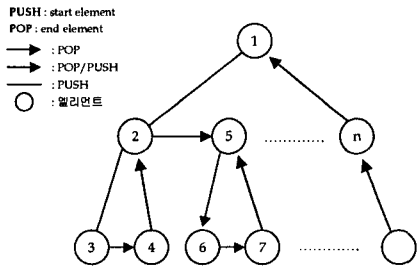


그림 2. Processing order of XML Document Structure by Stack

그리고 그림 3과 같이 각 노드에 따라 구조화 정보를 결정하며, 인덱스 및 구조정보를 가진 데이터 구조를 생성한다.(empty 엘리먼트 또는 속성정보 생략 가능) 노드는 엘리먼트를 나타내며, 각 노드의 숫자는 레벨과 부모노드, 자식노드, 형제노드와 순서의 의미를 동시에 갖고 있다. 또한 각 엘리먼트는 자신의 엘리먼트 이름과 여러 개의 속성과 컨텐트(내용)인 텍스트를 가질 수 있다. 속성 검색을 위해 속성은 내부 인덱스 정보를 구성하고 있으며, 텍스트 검색을 위해 텍스트도 단어 단위의 인덱스 정보를 갖고 있다.

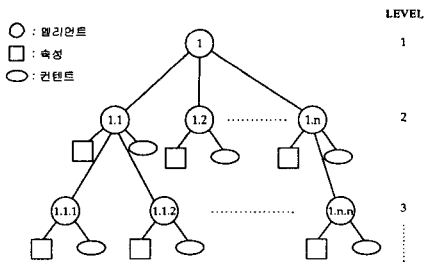


그림 3. Index and Structure Information of Nodes

3.3 XQL 질의어 처리기

XQL 질의어 처리기에서는 사용자로부터 질의어를 입력받고, 질의어에 대한 구문분석을 처리한 후 엘리먼트 검색, 속성검색, 컨텐트(텍스트)검색에 대하여 분류하고 기존의 인덱스 정보와 구조 정보를 구성하였던 인덱스 문서구조 파일을 참조하여 검색한다. 속성검색의 경우 속성 이름과 속성값에 대하여 인덱스정보를 검색하며, 컨텐트 검색의 경우 단어 단위로 인덱스된 정보를 검색한다.

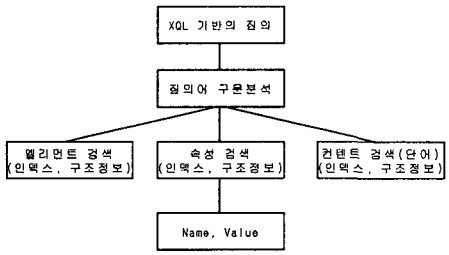


그림 4. XQL Query Processor

질의어 처리기의 순서는 다음과 같다.

(1) XQL 질의어 입력
(2) XQL 질의문 구문분석기에 의해 파싱
(3) 파싱 트리 생성
(4) 기존의 인덱스정보 및 구조정보로부터 엘리먼트 검색(노드 단위의 id 검색), 속성 검색(Name과 값에 따라 검색), 컨텐트 검색(단어 단위로 검색)
(5) 검색 여러개의 중간 결과 값 작성과 필요시 (4)번의 처리과정 반복
(6) 최종 처리결과 XML 문서로 출력

3.4 질의어의 예와 출력 결과

1) 입력 XML 문서

그림 5의 XBook을 상위 엘리먼트로 가지는 XML 문서를 이용하여 XQL 질의어에 대한 검색을 한다. 여기서는 간단한 예로 XBook 엘리먼트 1가지만을 보여주고 있다.

```
<XBook>
    <title internal="en-ko" grade="2">
                                    자바와 XML</title>
    <image>89-7914-107-6.gif</image>
    <author>브렛 맥래프린</author>
    <publisher>한빛미디어</publisher>
    <date>2001-03-31</date>
    <page>528</page>
    <note>
        <bookindex>
            <chapter id="1">1장. 소개</chapter>
            <chapter id="2">2장 XML문의...</chapter>
            <chapter id="3">3장. XML 분석</chapter>
            .....
            <chapter id="16">부록B. SAX ...</chapter>
        </bookindex>
        <content> 최근 인터넷 개발자가 ..</content>
    </note>
</XBook>
<XBook>
.....
</XBook>
```

그림 5. Example of XML Input Document

2) 질의어 예와 출력 결과 문서

여기서는 질의어 검색을 엘리먼트 검색, 속성 검색, 내용검색으로 분류하여 출력 결과를 테스트한다.

(1) 속성검색 : XQL 질의어 입력인 //XBook/title[@grade = "2"] 은 모든 XBook 엘리먼트를 찾고 바로 다음 자식 노드에 있는 title 엘리먼트 중에서 속성 이름이 grade이면서 속성값이 2인 title 엘리먼트들을 출력한다.

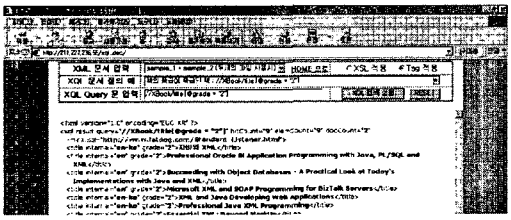


그림 6. Execution screen of XQL Attribute Search

(2) 엘리먼트 검색 : XQL 질의어 입력인 //XBook/title은 모든 XBook 엘리먼트를 찾고 바로 다음 자식 노드에 있는 title 엘리먼트들을 모두 출력한다.

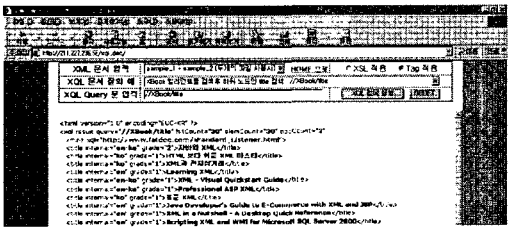


그림 7. Excution screen of XQL Element Search

(3) 내용검색 : XQL 질의어 입력인 //XBook/title[. &= "JAVA"] 는 모든 XBook 엘리먼트를 찾고 바로 다음 자식 노드에 있는 title 엘리먼트의 내용 중에 'JAVA'가 포함된 노드만을 출력한다.

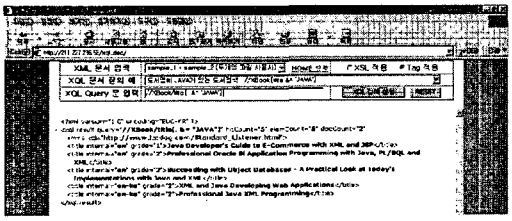
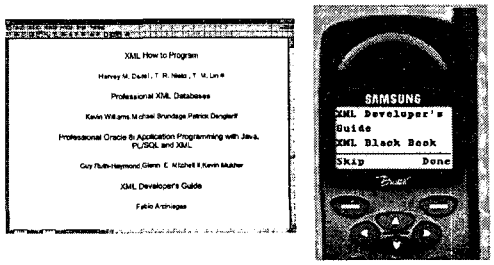


그림 8. Execution screen of XQL Content search

3) 질의 결과의 Cocoon 적용 예

그림 9는 검색 시스템에 의해 검색된 결과를 웹 출판 프레임워크인 Cocoon에 적용하여 PDF 문서 포맷과 WML 문서 포맷으로 출력한 결과이다.



(a) PDF (b) WML

그림 9. Example of Web Publish Framework

4. 결론 및 향후과제

컴퓨터를 이용한 문서의 처리 및 이 기종 시스템간의 정보 교환은 그 중요성이 계속 증가되고 있으며, 이에 대한 XML 문서에 대한 관리와 검색은 점차적으로 중요해 지고 있다. 특히 XML 문서는 구조화되어 있는 문서이기 때문에 문서 검색과 문서관리에 대단한 장점을 지니고 있고 문서 관리 및 검색에 대한 응용분야 또한 다양해 질 것이다.

따라서, 현재 XML 분야에 자체 기술 축적을 위해 본 논문에서는 XML 문서에 대한 구조와 표준 질의로 검색할 수 있는 시스템을 설계하였다. 웹 상에 분산된 XML 문서를 입력으로 하고 XML 문서구조 처리기에 의해 XML문서를 구조 분석하고, XQL 기반의 질의어 처리기에 의해 문서의 엘리먼트와 속성, 내용을 검색할 수 있도록 구성하였다. 또한 그 결과는 사용자가 원하는 형태의 문서 포맷이나 다른 형태의 컨텐츠로 제공 가능하도록 하였다. 본 논문에서 구현한 XML 문서 검색 시스템은 XML 검색엔진 또는 XML 기반 데이터베이스 응용분야에 광범위하게 적용될 수 있는 기술이며, 특히 전자상거래 분야의 상품 카달로그 관리나 전자박물관, 전자도서관 등에 응용이 가능할 것으로 예상된다. 향후 과제로는 온라인 상에서 대규모의 XML 데이터를 효율적으로 이용하면서 분산된 XML 문서를 저장 및 관리할 수 있는 환경으로 확장시킬 필요가 있다.

[참 고 문 헌]

[1] Jonathan Robie, Joe Lapp and David Schach, "XML Query Language(XQL)", http://www.w3.org/Style/XSL/Group/1998/09/XQL-propose.html, Sep, 1998.

[2] Jonathan Robie, "The Design of XQL", http://www.texcel.no/whitepapers/xql-design.html, 1998.

[3] Extensible Markup Language(XML)1.0, W3C Recmmendation, http://www.w3.org/TR/1998/REC-xml-19980210, 1998.

[4] Elliotte Rusty Harold, XML Bible, IDG BOOKS, 1999.

[5] Steven holzner, XML complete, Mcgraw-Hill, 1999.

[6] Apache XML Project: *Cocoon Documentation* http://xml.apache.org/Cocoon/index.html 1999.