

XML에 기반한 Local 검색 시스템의 설계 및 구현

김병진⁰ 김두현 홍도석 김용성
전북대학교 컴퓨터학과
(darkh9, kdh, capcom)@mail.chonbuk.ac.kr, yskim@moak.chonbuk.ac.kr

Design and Implementation of Local Search System Based XML

Byoung-Jin Kim⁰ Doo-Hyun Kim Do-Seok Hong Yong-Sung Kim
Dept. of Computer Science, Chonbuk National University

요 약

본 논문에서는 방대한 양의 정보를 관리하고 검색하기 위한 데이터베이스와 검색엔진을 사용하지 않고 정보를 효율적으로 처리할 수 있는 XML 기반의 Local 검색 시스템을 설계하고 구현한다.

CD-ROM이나 하드디스크와 같은 보조기억장치에 저장되어 있는 문서 파일을 효율적으로 관리하고 저장하기 위해서 인덱스를 XML 문서로 작성한다. 이러한 XML 문서의 태그를 이용해서 사용자가 원하는 문서 파일을 스크립트(Script) 언어를 사용해서 검색한다. 본 논문에서 제안한 시스템을 이용하면 빠른 검색시간과 효율적으로 문서들을 관리할 수 있다.

1. 서 론

정보 산업(Information Technology : IT)의 급속한 성장에 따라 정보에 대한 중요성과 필요성이 인식되고 있다. 이러한 정보는 인터넷이 국내외적으로 급속하게 보급되면서 자료의 양이 기하급수적으로 증가하고 있으며 사용자는 자신이 원하는 정보를 얻기가 점점 어려워지고 복잡해지고 있다. 이러한 문제점을 해결하기 위해서 각종 검색 엔진을 통한 검색서비스의 증가로 편의성을 제공하고 있으며 그 성능도 많이 향상되고 있다. 또한, 사용자는 자신에게 필요한 정보가 증가하면서 개인이 소유한 정보를 관리하기 위해서 CD-ROM이나 하드디스크와 같은 보조기억장치에 정보를 저장하고 있다.

그러나 이러한 보조기억장치에 존재하는 정보를 개인이 활용하기에는 여러 가지 문제점이 있다. 정보가 파일 단위로 존재하기 때문에 원하는 정보를 빠른 시간 내에 찾기가 어렵고, 정보를 조직적으로 관리하기가 힘들다.

이러한 문제점을 해결하기 위한 방법중의 하나가 정보를 XML 문서의 형태로 관리하는 것이다. XML은 문서의 구조정보를 제공할 뿐만 아니라, XML 태그는 데이터를 해석하는데 사용할 수 있기 때문에 데이터로서 XML의 역할도 중요성이 인식되고 있다. 따라서 구조 정보를 내포하고 있는 데이터로서의 XML 문서를 효과적으로 관리하는 구조와 질의어 설계 및 처리에 대한 연구가 많이 진행되고 있다.

본 논문에서는 XML 태그를 기반으로 하여 사용자의 정보들을 보다 효율적인 검색할 수 있는 Local 검색 시스템을 제안한다. 사용자의 정보에 따라서 검색에 적합한 DTD를 설계하고, 정보를 입력하면 자동으로 XML 파일을 생성한다. 또한, XML 태그정보를 이용하여 효율적인 러컬 검색 알고리즘을 제시한다. 제안된 방법은 데이터 베이스와 검색 엔진이 필요하지 않으며, XML의 태그를 해석하기 때문에 보조기억장치에 존재하는 정보를 빠른 시간 내에 검색할 수 있다.

2. 관련 연구

최근 컴퓨터 정보 통신 기술의 발달로 인한 정보의

양과 급증으로 산재되어 있는 정보들을 효과적으로 찾고자 하는 검색엔진을 통한 검색 서비스의 발달과 습득한 정보들에 대한 정보 처리의 중요성이 증가하고 있다.

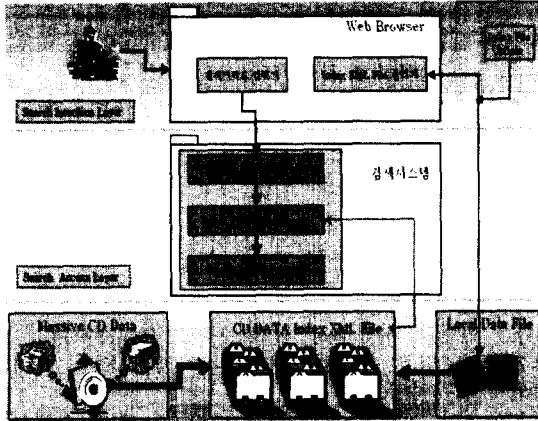
이러한 정보 환경의 변화는 사용자에게 쉽고 간편함을 제공하지만 데이터들은 과다하게 증가함으로써 복잡한 형태를 지닌다. 따라서 복잡한 정보를 처리하기 위한 노력으로 대용량 데이터베이스의 설계 및 자료의 저장구조 분석 방법 등의 연구로 이를 활용한 디지털 도서관, 전자출판, 전자 상거래 등의 다양한 분야에 활용되어지고 있다. 또한 이러한 대량의 문서를 구조화하여 관리하거나 표준화된 방법으로 처리하기 위한 대안으로 XML이 이용되어지고 있다[1, 2]. W3C에서 제안된 SGML을 축약한 XML은 다양한 구조의 문서나 데이터의 표현을 간결하고 다양하게 담을 수 있다. 따라서 이렇게 구조화된 XML 문서는 다양하고 대량의 데이터에 대해 간결한 표현과 많은 정보들을 담을 수 있어 전자상거래용 문서에서 메타데이터 관리에 이르는 그 크기와 복잡성이 다양한 구조화된 문서와 정보들 처리에 있어 다양하고 광범위한 이용이 가능하다[3,4].

또한 대다수의 검색 시스템들에서 많이 쓰이는 3-계층 모델은 각각의 모듈이 다른 곳에서 개발되어도 응용하는 데는 어려움이 없다[5]. 3-계층(3-Tier)중 데이터 베이스 계층은 데이터를 저장하는 곳으로 구조화 된 정보, 즉 데이터 베이스나 XML문서와 같은 어떠한 구조로도 구성할 수 있으며 논리적으로 구조화되어 있기 때문에 다른 계층 모듈에 비해 활용도가 넓다. 현재 대다수 시스템의 경우 3-계층구조의 데이터 베이스 시스템이나 웹 문서로부터 정보를 추출하고 수집한다[6, 7]. 수집된 결과들은 HTML 태그들을 통해 변환된 후 클라이언트의 웹 브라우저에 보여진다. 그러나 문서가 데이터 타입과 같은 원본 그대로의 정보를 가지고 있는 XML 문서로 구조화되어 있다면 기존의 시스템보다 데이터 처리 방법이 간단한 시스템의 설계가 가능하다. 각 시스템은 DTD를 참조하여 이루어져 있기 때문에 본 논문에서는 이 DTD 정보를 기반으로 한 XML 기반의 Local 검색 시스

템을 설계 구현하였다.

3. 시스템의 설계

본 논문에서 제안하는 Local 검색 시스템의 구조는 [그림 1]과 같이 3-계층으로 나누어지고, 4가지 구성요소로 구성된다.



[그림 1] 전체 시스템 구성도

- i) 정보 입력기 : 사용자가 새로운 정보가 생성되면 이를 입력하는 기능을 한다.
- ii) 웹 브라우저 : 사용자가 정보를 검색하거나 검색된 정보를 보여주는 기능을 한다.
- iii) 검색 시스템 : 사용자의 요구에 따라 정보를 검색하는 기능을 한다. 검색 형태는 파일의 제목, 날짜, 저자, 내용 등이 존재한다.
- iv) 파일 저장소 : 실제 정보가 저장되는 곳이다.

4. 시스템의 구현

로컬 검색 시스템을 구현하기 위해서 정보의 검색에 사용되는 정보를 추출해서 XML DTD를 설계하고, 이 정보를 자동으로 XML 소스로 만드는 주는 XML 소스 변환기를 만든다. 또한 검색엔진을 사용하지 않고 검색이 가능한 알고리즘을 제안한다.

4.1 문서에 대한 XML DTD 설계

본 논문에서 제안하는 검색 엔진을 사용하지 않고 XML 태그를 이용해서 검색하기 위한 XML DTD 구조는 다음과 같다.

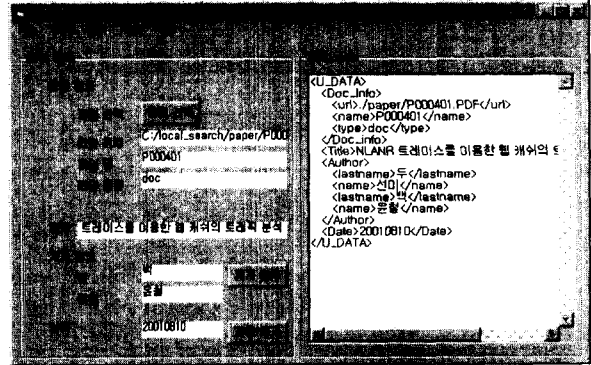
```
<!DOCTYPE U_date [
<!ELEMENT U_date(Doc_info, Title, Author*, Date)>
<!ELEMENT Doc_info(url, name, type)>
<!ELEMENT url (#PCDATA)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT type (#PCDATA)>
<!ELEMENT Title (#PCDATA)>
<!ELEMENT Author (lastname, name)>
<!ELEMENT lastname (#PCDATA)>
<!ELEMENT name (#PCDATA)>
<!ELEMENT Date (#PCDATA)>
]>
```

[그림 2] XML DTD 구조

사용자의 문서 파일을 XML 태그를 이용해서 검색하기 위해서 문서 파일의 검색에 많이 사용되는 제목, 사용자 정보, 생성 날짜, 파일명, 파일 확장자, 그리고 하이퍼링크에 필요한 URI 등의 정보를 인덱스로 두어 검색을 한다.

4.2 XML 소스 변환기

사용자가 문서 파일의 정보 입력하면 XML 태그를 이용해서 정보를 검색하기 위해서 4.1에서 제안한 Paper DTD 구조의 XML 문서를 만들어야 한다. [그림 3]과 같은 XML 소스 변환기에서 작성된 XML 소스는 XML 파일에 추가된다.



[그림 3] XML 소스 변환기

4.3 검색 알고리즘

검색 엔진을 사용하지 않고 Local에서 XML 태그를 검색하기 위해서 script 언어를 사용한다. 본 논문에서는 javascript 언어를 사용했다. 또한 XML 문서에서 하나의 파일 정보를 레코드셋(RecordSet)으로 인식하기 위해서 XML 태그 중 ID 속성을 이용한다.

본 논문에서 제안하는 XML 기반의 local 검색 알고리즘은 다음과 같다.

```
//제목 검색 알고리즘
입력 : 사용자가 입력한 제목 키워드
출력 : 제목에 사용자가 입력한 키워드가 포함된 문서들
U_keyword = Search_String
//XML 문서의 처음으로 이동
XML_ID.recordset.moveFirst()
while(!XML_ID.recordset.EOF) { //마지막까지 반복
    TitleString = XML_ID.recordset("Title").value;
    //키워드가 제목에 존재할 때
    if (TitleString.indexOf(U_keyword)>=0) {
        R_HTML = 제목과 저자 이름에 URL 링크
        R_HTML = R_HTML + "<p>"
    }
}
XML_ID.recordset.moveNext() //다음 정보로 이동
//레이어에 결과를 같은 페이지에 출력
if(R_HTML == "") {
    ResultDiv.innerHTML = "검색된 문서가 없습니다."
}
else {
    ResultDiv.innerHTML = R_HTML;
```

}

5. 실험 및 성능 평가

본 논문에서는 로컬의 CD-ROM에 존재하는 문서들의 정보들을 검색하기 위해서 정보처리학회 제 28회 춘계학술 발표회의 논문을 대상으로 실험을 했다.

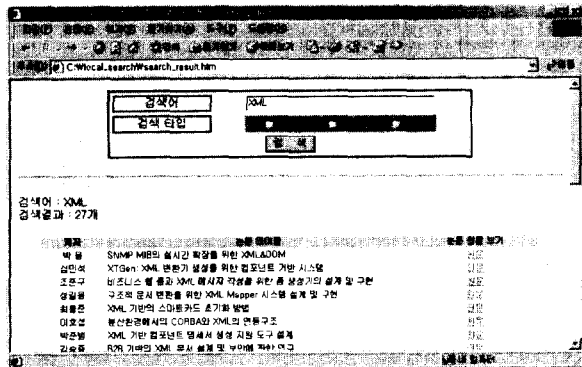
로컬 시스템에 존재하는 문서 파일을 검색하기 위해서 인덱스로 추출된 문서 정보들을 XML의 태그를 이용해서 표현한다.

4.2절에서 제시한 XML 소스 변환기를 이용하여 검색을 위한 XML 문서를 생성한 결과는 [그림 4]와 같다.



[그림 4] 검색을 위한 XML 문서

[그림 4]와 같은 XML 태그 정보를 이용해서 4.3에서 제안한 검색알고리즘을 적용해서 정보처리학회 제 28회 춘계학술 발표회의 논문 중 제목에 XML이 존재하는 논문을 검색한 결과는 [그림 5]와 같다.



[그림 5] 검색 결과

사용자의 정보가 계속 증가할 경우 XML 문서의 크기가 증가하여 Local 환경일지라도 검색 시간이 증가 때문에 XML 문서를 일정한 크기로 분리한다. 분리된 XML 문서를 인식할 때는 다음과 같은 태그를 사용하였다.

```
<XML ID="문서ID" SRC="XML 파일이름">
</XML>
```

XML 문서를 분리함으로써 검색이 단축되었다.

6. 결론 및 향후 연구 과제

인터넷을 이용한 정보의 양이 증가하면서 개인에게 필요한 정보의 양도 증가하고 있다. 하지만 많은 연구들이 전자 도서관과 같은 대규모의 데이터베이스를 이용하고, 검색 엔진을 사용하는 분야에 집중되고 있다.

따라서 본 논문에서는 CD-ROM이나 하드디스크와 같은 보조기억장치에 존재하는 개인의 정보를 데이터베이스나 검색엔진을 이용하지 않고 정보를 효율적으로 처리할 수 있는 XML 기반의 로컬 검색 시스템을 설계하고 구현하였다.

이를 위해서 정보 효율적인 관리하고 저장할 수 있는 인덱스를 XML 문서로 작성한다. 이러한 XML 문서의 태그를 이용해서 사용자가 원하는 문서 파일을 스크립트 (Script) 언어를 사용해서 검색한다. 또한, 정보의 양이 증가할 경우 검색 시간을 단축하기 위해서 XML 문서를 분리해서 저장하였다.

본 논문에서 제안한 시스템을 이용하면 빠른 검색시간과 효율적으로 문서들을 관리할 수 있으며, 향후 연구과제로는 XML 에디터를 이용해서 여러 타입의 정보를 관리하고, XML 문서의 분리시 효율적인 인덱스 구조를 제시해야 할 것이다.

참고문헌

- [1] Jon Bosak, "XML, Java, and the future of the Web", <http://sunsite.unc.edu/pub/sun-info/standards/xml/why/xmlapps.htm/>
- [2] Richard Lander, "XML: The New Markup Wave", http://www.csclub.uwa.terloo.ca.u/relander/XML/Wave/xml_mw.html/
- [3] W3C, "Extensible Markup Language(XML) 1.0" <http://www.w3.org/TR/1998/REC-xml-1998210/>, 1998
- [4] Dan Chang, Dan Harkey, Client/Server Data Access with Java and XML, pp 106-130, 11, 1997
- [5] D. D. Roure et al, Agents for Distributed Multimedia information Management, Proc. of 1 In-tl Conf. on Practical Applications of Intelligent Agents and Multi-Agents Technology, 4, 1996
- [6] Elliott Rusty Harold, JAVA Network Programming?, pp 302-340, 6, 1999
- [7] Frank Boumphrey, Professional XML Applications, pp 208-252, 09, 1999
- [8] Hiroshi Maruyama, Kent Tamura, Naohiko Uromoto, "XML and Java Developing Web Applications", pp 282-311, 8, 2000.
- [9] 이강찬외 2인, XML 표준화 동향, 정보과학회지, 제 19권 1호, 제140호, 01, 2001