퍼지 및 지능시스템학회 논문지 2006, Vol. 16, No. 1, pp. 94-100

B^+-tree를 이용한 XML 색인기법

Indexing of XML with B^+-tree

권국봉* · 홍동권*

Guk-Bong Kwon and Dong-Kweon Hong

* 계명대학교 컴퓨터 공학과

요 약

인터넷을 바탕으로 하는 컴퓨팅 패러다임의 변환은 인터넷에서 디지털 정보 교환의 표준으로 확고한 자리를 굳힌 XML의 사용을 가속화시키고 있다. 이로 인해 XML 데이터의 양이 기하급수적으로 증가하고 보다 효율적으로 XML 데이터를 저장하고 질의하기 위한 연구가 활발히 진행되고 있다. 본 논문에서는 대용량의 데이터 중심 XML을 효과적으로 관리하기 위한 방안으로 그래프 중심의 색인 방법을 연구, 제안한다. 논문에서 제안한 XML 색인은 XML 데이터와 XML 구조 정보를 크게 3개의 구성 성분으로 표현한 후, 메인 메모리 자료구조로 표현된 각 그래프들을 노드 아이디를 키로 사용하여 B+트리에 각 노드를 사상하고 B+트리를 디스크에 저장하여 이들 색인 모델에 대해 지속성을 부여하였다. 본 논문에서 제안한 색인 방법을 통해 XML 데이터의 크기에 따라 질의 처리시간이 선형적으로 증가하는 결과를 얻을 수 있었다.

Abstract

Computing paradigm shift to internet-based one has accelerated the use of XML in diverse applications. This phenomena has made the explosive increases of XML data and it triggered many active researches on maintaining very huge amount of XML data in turn. In this paper we present a persistent graph-based XML indexing for data-centric XML data. In our approach we use 3 graphs to represent XML indexes and XML data itself. They are schema graph, data graph and data graph index. And then we have mapped those graphs to B+-trees to give them the persistency. With our approach we can achieve linear query execution time with the increase of XML sizes.

Key words : XML indexing, data-centric XML, XML query processing

1. 서 론

인터넷의 사용과 정보의 양이 급증하면서 대용량의 정보를 보다 효과적으로 활용하는 다양한 방법의 연구가 활발히 진행 되고 있다. 웹에서 널리 사용되고 있는 HTML(HyperText Markup Language)은 문서에서 데이터의 의미보다 데이터의 표현에 중점을 두고 있어 많은 응용 분야들이 요구하는 데이터의 의미를 표현하는 표현 방법으로는 기능이 부족하다는 단점이 있다. 이에 웹 발전의 주도적인 역할을 하고 있는 W3C(World Wide Web Consortium)에서는 1996년 차세대 웹 문서의 표준으로 XML(Extensible Markup Language)을 제안하였으며 XML 관련 기술들은 현재 다양한 분야에서 폭넓게 활용되고 있다[1].

XML은 최근 20~30년 동안 기술적, 상업적으로 급속한 발전을 이룬 관계형 데이터베이스의 구조적인 테이블과는 다른 데이터 모델을 사용한다. 이러한 데이터의 반구조적인 특성과 XML 질의어인 W3C XQuery의 데이터 모델의 차이점으로 인하여 XML 데이터를 기존의 데이터 모델로 표현하는 것은 데이터의 변환, 복잡한 연산의 성능 저하와 같은 문제점을 보이고 있다. 이와 같은 이유로 반구조적인 문서로 표현되는 XML을 효과적으로 저장, 검색, 색인 하는 기술의 필요성이 날로 증가하고 있으며 XML 전용 데이터베이스 (native XML database) 분야에서 다양한 연구가 활발히 진행되고 있다. 따라서 본 논문에서는 DTD (또는 XML Schema, 이하 DTD라 칭함)에 나타난 XML 데이터의 구조 정보를 사용하여 XML을 효율적으로 저장, 관리, 검색할 수 있는 XML 전용 데이터베이스의 색인 모델을 구현하고 그 성능을 평가한다. 본 논문의 구성은 다음과 같다. 2장에서는 XML 데이터의 색인 모델에 대한 기존 연구들을 살펴보고, 3장에서는 효율적인 검색을 위한 색인 모델을 제안하고 색인 모델의 구현을 설명한다. 4장에서는 예제 XML 데이터로 본 논문에서 제안한 색인모델로 색인을 구축하고 질의 수행 실험 결과를 바탕으로 성능평가를 한다. 마지막으로 5장에서는 결론과 향후 연구방향을 제시한다.

접수일자 : 2006년 1월 6일
완료일자 : 2006년 2월 3일
감사의 글 : 본 연구는 계명대학교 비사연구기금으로 이루어졌음

2. 관련 연구

XML 문서 색인에 대한 연구는 기존의 일반 문서 또는 SGML 문서의 색인에 대한 연구 결과를 많이 활용하고 있다. 호주 RMIT에서 제시한 SCL(Simple Concordance List)모델은 정의된 문서 계층과 정의된 마크업 스키마로부터 독립을 제공한다. 이 모델은 텍스트 간격들을 다루며 문서 구조에 대한 질의를 지원하기 위해 계층적인 관계보다 오히려 포함관계를 사용한다. 이 모델은 SC-list 데이터 타입에서 중첩된 정보를 허용하기 때문에 리스트의 리스트와 같은 순환

구조를 다룰 수 있다[2, 3]. 그러나 SCL 구조는 색인어를 포함하는 엘리먼트를 찾지 못하는 단점을 가진다. 즉, 특정 엘리먼트의 조상, 형제들을 알 수 없으며 포함하고 있는 엘리먼트가 몇 번째 자손에 해당하는지 알 수 없다.

추상화 기반 방법은 구조화된 문서의 문법적 구조를 부모와 자식의 관계를 나타낼 수 있는 트리로 표현 하여 문서 구조를 나타내고, 문서를 읽지 않고 색인 내에서 경로 표현을 구하기 위해서 문서 구조의 특정 형태를 취하는 방법이다. 즉, 실제 문서들의 구조를 추상화한 색인 구조인 추상화를 사용한 구조 검색 색인을 설계 하는 것이다. 추상화는 여러 가지 방법이 있고 활용하려는 응용에 맞게 결정해야 한다[4]. 추상화 색인의 경로 표현을 통해 문서에서 부모, 자식, 자손들을 찾을 수 있고 색인의 크기를 줄일 수 있으나 완전한 구조 검색 즉, 추상화 되지 않은 문서 구조의 검색을 위해서는 문서 전체를 읽어야 하는 오버헤드가 있다. 대표적인 것이 XML infoset이다[1]. Index Fabric[5]은 XML 색인 방법에서 경로를 데이터 그래프 내에서 나타나는 레이블들을 알파벳으로 하는 문자열로 취급하여 문자열 탐색에 유용한 구조인 패트리샤 트라이 (Patricia trie)에 경로를 저장하는 방법이 제안되었다. 패트리샤 트라이는 주기억장치 자료구조이므로 영속성 (Persistency)을 제공하기 위하여 Patricia 트라이를 디스크에 저장하기 위한 기법도 같이 제안하고 있다. 이때, 색인 탐색 시의 입출력 횟수를 줄이기 위해 Patricia trie에 대한 다단계 색인을 구축한다[5]. APEX[6]는 적응성 있는 경로 인덱스를 사용하는 방법(APEX: Adaptive Path Index for XML Data)으로 데이터 마이닝 기법을 이용해 자주 사용되는 경로를 빠르게 찾을 수 있도록 했다. 질의의 워크로드(workload)가 달라지면 색인의 구조를 점진적으로 변경하는 색인 그래프에 대해 설명하고 있다[6]. 하지만 APEX는 워크로드에 기반을 둠으로 기본적으로 부정확한 색인 그래프를 유지하고 있으며 자주 사용되지 않는 질의를 처리하기 위해서는 질의 수행 시 색인 그래프를 갱신해야 하는 문제점을 가지고 있다. Vist[15]는 모든 XML을 대상으로 한 것이 아니라 데이터 중심의 구조적 XML (structured XML)의 색인을 제안하고 있다. 현재 많은 응용이 다루고 있는 데이터의 상당 부분이 문서 중심의 XML 데이터가 아니라 데이터 중심의 XML 데이터라는 인식을 하고 이 방법은 XML 데이터와 XML 질의를 시퀀스로 표현하고 XML 데이터에서 XML 질의의 답을 찾는 것을 서브시퀀스 매칭 문제로 표현하고 있으며 색인을 저장하기 위하여 B+tree를 활용하고 있다.

3. 그래픽 방식의 색인 모델

본 논문에서 다루는 XML 데이터베이스는 같은 DTD를 사용하는 XML 문서의 집합으로 XML 문서의 컬렉션과 같다. XML 컬렉션에 들어있는 각각의 XML 문서는 구조적 혹은 반 구조적인 데이터 중심 (data-centric)의 문서이며, 색인은 XML 문서의 컬렉션 단위로 생성되고 관리된다. XML 문서에 대한 연산은 값을 기준으로 하는 데이터 중심의 연산이 대부분을 차지하며 XML 색인은 데이터 중심의 연산들을 효율적으로 수행 할 수 있게 구성된다.

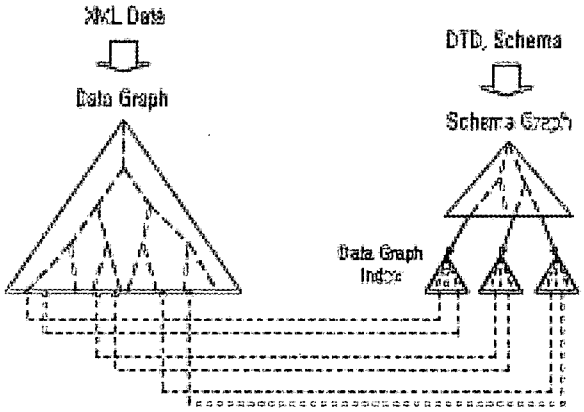

그림 1. 색인 모델의 전체 구조

3.1 스키마 그래프의 구조 및 구현

스키마 그래프는 색인 구현에서 가장 먼저 생성되는 그래프로서 DTD를 사용하여 XML 데이터의 구조 정보를 표현한다. DTD가 유효한(valid) XML 문서의 모든 가능한 형태를 결정하므로 스키마 그래프를 이용하여 경로(path)를 사용한 XML 질의에서 경로의 타당성을 빨리 검사할 수 있으며 경로와 관련된 데이터의 검색을 효율적으로 실행할 수 있게 도와준다. 스키마 그래프는 그림 2와 같은 XML DTD를 그림 3과 같은 트리형태의 그래프로 표현한 것이며 생성과정은 다음과 같다. 먼저 XML 데이터의 DTD를 Xerces의 SAX, DOM 파서와 JDOM을 이용하여 분석하고 그 다음 분석된 DTD의 내용을 그래프로 표현한다. 이때 표현된 그림 3과 같은 트리는 메모리에서 표현된 자료구조이므로 영속성 (지속성, persistency)을 위하여 하드 디스크에 저장하는 방법이 필요하다. 본 연구에서는 지속성을 위해 XML 엘리먼트 노드 아이디를 기준으로 그래프를 B+트리에 저장한다. 지속성을 위한 스키마 그래프의 디스크 표현 방법은 그림 4와 같다. 각 노드마다 자신의 고유한 노드 아이디(NID)와 부모의

```
<!ELEMENT dblp (article | mastersthesis)*>
<!ENTITY % field
"author|editor|title|year|journal|volume|month|url|ee|cdrom|
school">
<!ELEMENT article (%field;)*>
<!ATTLIST article
key CDATA #REQUIRED reviewid CDATA #IMPLIED
rating CDATA #IMPLIED mdate CDATA #IMPLIED>
<!ELEMENT mastersthesis (%field;)*>
<!ATTLIST mastersthesis
key CDATA #REQUIRED mdate CDATA #IMPLIED>
<!ELEMENT author (#PCDATA)> <!ELEMENT editor
(#PCDATA)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT year (#PCDATA)>
<!ELEMENT journal (#PCDATA)>
<!ELEMENT volume (#PCDATA)>
<!ELEMENT month (#PCDATA)>
<!ELEMENT url (#PCDATA)>
<!ELEMENT ee (#PCDATA)>
<!ELEMENT school (#PCDATA)>
<!ELEMENT cdrom (#PCDATA)>
```

그림 2. XML DTD 예제

노드 아이디(PNID), 자식의 노드 개수(CCNID), 자식의 노드 아이디(CNID) 등을 표시한다. 또한 단말노드에는 데이터 그래프와의 연결을 위해 단말 노드마다 각 단말 노드 구조와 일치하는 데이터 그래프의 노드에 대한 노드 아이디(NID)와 노드 아이디의 텍스트 값으로 구성된 데이터 그래프 인덱스가 연결되어 있어 질의에 사용 되는 XML 데이터의 노드에 대한 빠른 접근을 만들어 준다.

그림 4는 예제 스키마 그래프 그림 3에서 노드 아이디가 2번인 mastersthesis를 포함하고 있는 정보를 나타낸 것인데 각 노드는 각 노드 헤더에 자신의 고유한 노드 아이디, 엘리먼트 자리수, 부모노드의 아이디, 자식노드의 개수, 애트리뷰트의 존재 유무 등의 정보를 포함하고 있으며 그 뒤로 엘리먼트의 이름이 있고 그 다음 자식 노드의 아이디 정보를 포함하고 있다.

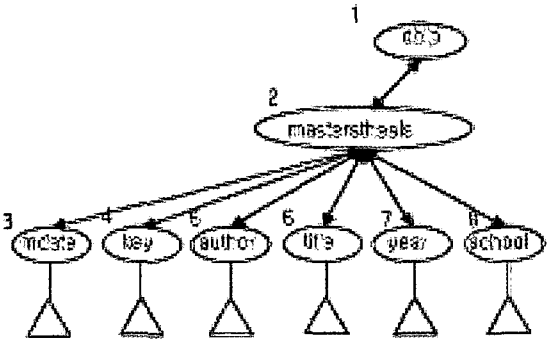

그림 3. DTD 일부분을 트리 형태로 변환한 모습

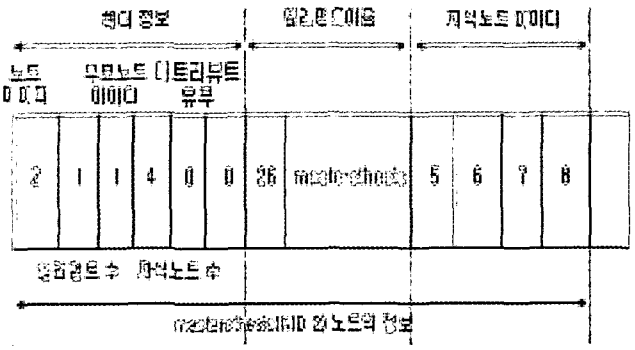

그림 4. 스키마 그래프의 디스크 저장 정보

3.2 데이터 그래프의 구조 및 구현

데이터 그래프는 XML 데이터의 내용을 담고 있는 그래프로서 스키마 그래프가 만들어지고 난후 스키마 그래프의 구조 정보를 이용해 XML 데이터를 데이터 그래프로 만든다. 데이터 그래프는 스키마 그래프와 비슷하게 트리형태의 그래프로 표현되는데 단말노드에 데이터 그래프 인덱스를 가지고 있는 스키마 그래프와는 달리 데이터 그래프는 단말노드에 각 노드의 애트리뷰트, 엘리먼트의 값을 가지고 있다. 데이터 그래프는 단말노드를 제외하고는 그림 4의 스키마 그래프 표현과 동일한 노드 정보를 가진다.

데이터 그래프의 생성 과정은 스키마 그래프의 생성 과정과 유사하며 다음과 같은 생성 과정을 거친다. 첫째, XML 데이터를 Xerces의 SAX, DOM 파서와 JDOM을 이용하여 분석하고 둘째, 파싱된 XML 데이터를 트리 형태의 그래프로 표현한 다음 셋째, 지속성을 위해 노드 아이디를 키로 하여 B+트리의 각 단말노드에 저장한다.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<!DOCTYPE dblp SYSTEM "dblp.dtd">
<dblp>
 <mastersthesis mdate="2002-01-03"
                key="ms/Brown92">
    <author>Kurt P. Brown</author>
    <title>
     PRPL: A Database Workload, v1.3.
    </title>
    <year>1992</year>
    <school>
    Univ. of Wisconsin-Madison
    </school>
 </mastersthesis>
 <article mdate="2002-01-03"
          key="tr/dec/SRC1997-018">
    <editor>Paul R. McJones</editor>
    <title>
       The 1995 SQL Reunion, May 29, 1995.
    </title>
    <journal>
     Digital System Research Center
    </journal>
    <volume>SRC1997-018</volume>
    <year>1997</year>
    <ee>db/labs/dec/SRC1997-018.html</ee>
    <ee>http://www.mcjones.org/System_R/</ee>
    <cdrom>decTR/src1997-018.pdf</cdrom>
 </article>
 <mastersthesis mdate="2002-01-03"
                key="ms/Yurek97">
   <author>Tolga Yurek</author>
   <title>
   Efficient View at  Data Warehouses.
   </title>
   <year>1997</year>
   <school>University of California</school>
 </mastersthesis>
</dblp>
```

그림 5. XML 데이터 예제

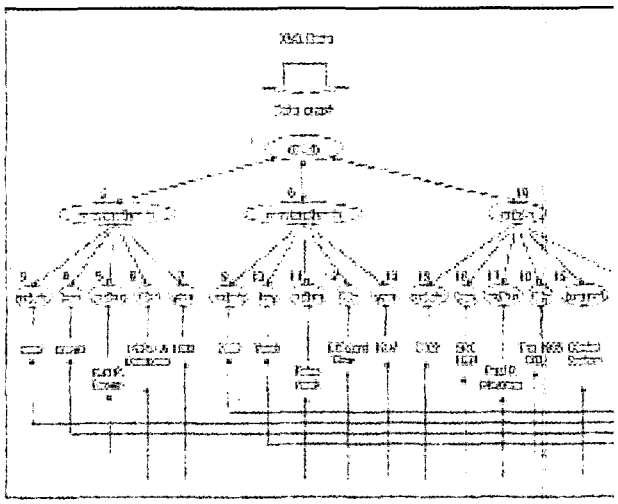

그림 6. 트리형태로 변환된 XML 데이터

3.3 데이터 그래프 인덱스의 구조 및 구현

데이터 그래프 인덱스는 스키마 그래프의 구조와 일치하는 데이터 그래프의 노드를 노드의 텍스트 값, 노드의 아이디를 기준으로 정렬해둔 B+트리 기반의 그래프이다. 데이터 그래프 인덱스는 B+트리와 유사한 형태로 표현이 되는데 스키마 그래프가 생성되고 나면 각각의 단말노드에 데이터 그래프 인덱스의 생성이 초기화 되고 데이터 그래프가 생성되는 과정에 데이터 그래프 생성자로부터 데이터 그래프의 노드에 대한 정보를 넘겨받아 생성된다.

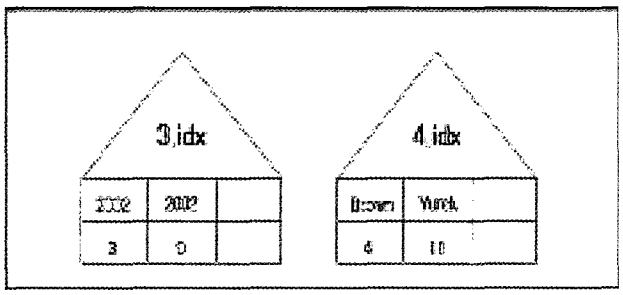

그림 7. 데이터 그래프의 인덱스

그림 7은 예제 스키마 그래프 그림 6의 3번 단말 노드에 연결된 데이터 그래프 인덱스를 나타내고 있다. 예제 스키마 그래프 그림 6의 3번 단말 노드는 masterstthesis의 애트리뷰트인 mdate를 나타내는 것으로 3번 단말 노드와 구조정보가 일치하는 데이터 그래프의 단말 노드들에 대한 텍스트 값과 노드 아이디를 짝으로 하여 텍스트 값을 기준으로 정렬하여 저장 한다. 이렇게 스키마 그래프의 각 단말노드에 스키마 그래프의 단말 노드와 구조 정보가 일치하는 데이터 그래프의 노드들에 대한 정보를 저장하여 둠으로써 XML 데이터에서 스키마 그래프의 노드와 구조정보가 일치하는 데이터 그래프 노드들에 대해서 빠른 임의의 접근이 가능 하게 된다

.앞에서 제안된 색인에서 데이터 그래프 인덱스는 B+트리로 구성이 되어 있기 때문에 바로 디스크에 저장 할 수 있고 나머지 데이터 그래프, 스키마 그래프를 디스크 저장하기 위해서 그림 8과 같이 트리의 노드 아이디를 고유한 키로 구분하여 B+트리의 단말 노드에 각 노드를 사상하여 B+트리로 저장한다.

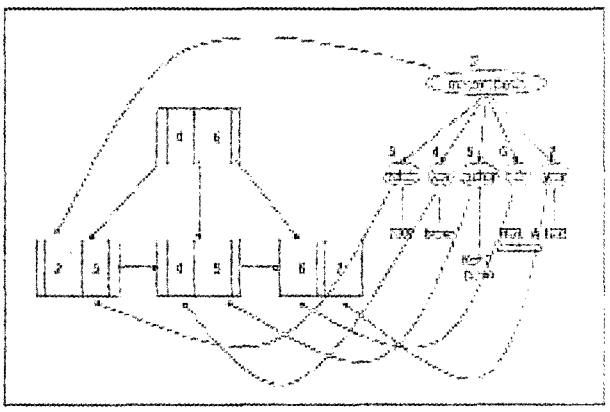

그림 8. 그래프를 B+트리에 저장

3.4 색인의 생성 및 질의 처리

XML 데이터에서 본 논문에서 제안하는 색인을 생성하는 방법은 다음과 같다. 첫째, XML DTD나 XML Schema를 스키마 그래프로 변환 한다. 둘째 DTD를 기준으로 XML 데이터를 데이터 그래프로 변환하여 표현 하고, 셋째 DTD의 각 엘리먼트마다 해당 되는 데이터 그래프의 노드 값을 기준으로 데이터 그래프 인덱스를 만든다. 마지막으로 이렇게 만들어진 각각의 그래프들을 지속성을 부여하기 위해 이들 그래프들을 B+트리의 단말노드에 사상시켜 디스크에 저장한다. 디스크에 저장된 각 그래프들은 XML 데이터의 질의에 다음과 같은 방법으로 사용된다. 질의에 사용되는 XML 데이터의 범위가 결정이 되면 스키마 그래프에서 구조정보를 얻어서 질의에 사용되는 데이터 그래프에 대한 인덱스를 얻고, 그 후 각각의 데이터 그래프 노드에 대한 접근을 한다. 각각의 데이터 그래프 노드에 접근 후 질의의 종류에 따라 노드에 포함되어 있는 정보를 이용하여 부모, 자식, 형제 등으로 이동하면서 질의를 수행 하게 된다. 이때 각각의 그래프는 디스크에 B+트리 형태로 저장되어 있는 상태이기 때문에 1-3회의 디스크 입출력으로 디스크의 그래프에 접근 가능하다.

4. 구현 환경 및 성능 평가

본 논문에서 구현한 색인의 기능 및 성능은 초기 XML 전용 시스템인 Kweelt[11]의 일부분으로 구현하여 그 성능을 평가하였다. Kweelt는 XML 데이터를 파일 형태로 저장하고 XML 질의는 파일을 메모리에 로드하여 DOM을 사용하는 방식으로 되어 있다. 구현 환경 플랫폼으로는 Linux Kernel 2.4.25를 사용하였으며 썬 JDK 1.4.2를 기반으로 하였다. 사용하는 주요 패키지로는 XML 데이터를 그래프로 만들기 위하여 XML 데이터를 파싱하고 조작하기 위해 Apache Software Foundation Xerces의 SAX, DOM 파서와 JDOM을 사용하였고 메모리상의 그래프를 디스크에 저장하기 위한 B+트리의 구현을 위해서는 GiST 패키지를 사용하였다[10].

성능 평가를 위해서는 다음과 같은 2가지 성능 평가 척도를 사용했다.

(1) XML 데이터의 구조에 따른 성능 측정

본 논문에서 제안한 XML 데이터의 색인 방법이 구조적 또는 반구조적인 XML 데이터에서 인덱스를 사용하지 않는 기존의 Kweelt보다 얼마나 좋은 성능을 낼 수 있는가를 알아보기 위해 연산 소요 시간을 XML 데이터의 구조에 따라 측정했다.

(2) XML 데이터의 크기에 따른 성능 측정

본 논문에서 제안한 XML 데이터의 색인 방법이 XML 데이터 파일의 크기가 커짐에도 좋은 성능을 낼 수 있는가를 알아보기 위해 소요시간을 XML 데이터의 크기에 따라 측정했다.

실험에 사용한 XML 데이터는 모두 2종류이다. 첫째는 auction 데이터이며 또 다른 데이터는 dblp 데이터이다[12,13]. auction 데이터는 관계형 데이터베이스의 테이블과 같이 구조적인 데이터이고 dblp 데이터는 반구조적인 (semi-structured) 데이터이다. 실험은 각각의 파일 사이즈를 1M, 2M, 4M, 10M, 20M, 40M, 100M로 변경하면서 측정하여 다양한 XML 데이터의 크기에 따른 색인의 성능을 잘 알아볼 수 있게 했다.

표 1. XML 데이터 파일의 크기에 따른 색인의 크기

크기(M)	auction(M)	dblp(M)
1	1.3	1.5
2	2.5	3.3
4	5.2	6.5
10	13	16
20	26	31
40	53	62
100	124	151

각 XML 데이터 파일을 본 논문에서 제안하는 색인 방법을 사용하여 색인으로 만들었을 때 색인의 크기는 <표 1>에서 보는 것과 같이 원래 XML 데이터 크기의 약 1.3배에서 1.6배 정도가 된다. 데이터 그래프가 원래 XML 파일이 가지고 있는 모든 내용을 전부 가지고 있으므로 XML 파일을 따로 저장할 필요가 없다는 것을 고려하면 실제 색인 때문에 추가적으로 늘어난 크기는 0.3배에서 0.6배 정도이다. 이런 결과에서 본다면 색인이 차지하는 공간은 지금의 컴퓨팅 환경에서 전혀 부담이 되지 않는 것으로 판단된다.

XML 질의에는 다양한 종류의 연산이 있다. 본 논문에서는 XML 질의어가 가지는 다양한 기능을 대표할 수 있는 Kweelt의 2가지 종류의 연산을 선택하였다. 첫 번째 연산은 그림 9에서 표현된 Kweelt 연산식으로 auction XML 데이터에서 99년 3월 한당동안 등록된 품목의 개수를 알아보는 연산이다.

```
LET $bid := document("bids.xml")//bid_tuple
        [bid_date .>=. '99-03-01' AND bid_date .<=.
'99-03-31']
RETURN
    <item_count>
       NUMFORMAT("#####", count($bid) )
    </item_count>
결과
<?xml version="1.0"?>
<item_count>
  171
</item_count>
```

그림 9. XML 데이터에 대한 질의

```
LET $thesis := document("dblp.xml")//mastersthesis
        [year .>=. '1992' AND year .<=. '1995']
RETURN
    <thesis_count>
       NUMFORMAT("#####", count($thesis) )
    </thesis_count>
결과
<?xml version="1.0"?>
<article_count>
  168
</article_count>
```

그림 10. XML 데이터에 대한 질의 2

2번째 연산은 그림 10의 연산식으로 dblp XML 데이터에서 1992년에서 1995년 사이에 발표된 석사 학위 논문의 수를 알아보는 범위 연산이다. 각각의 질의는 데이터베이스에 가해지는 질의 중 많은 부분을 차지하고 있는 연산이며 색인의 성능을 가장 잘 알아볼 수 있는 연산이다. 각각의 질의에 대한 질의 수행 결과와 수행 시간은 표 2와 표 3과 같다. 실험에서 수행시간은 Kweelt의 클라이언트에서 같은 조건으로 측정한 연산 반응시간이다.

표 2. XML 데이터의 크기에 따른 질의 1의 수행 시간 비교

문서크기(M)	수행시간(sec)		질의 결과 수
	Kweelt	본 논문의 방법	
1	21	4	171
2	57	6	340
4	147	11	682
10	523	24	1715
20	2658	45	3438
40	■	81	6854
100	■	171	17584

표 5. XML 데이터의 크기에 따른 질의 2의 수행 시간 비교

문서크기(M)	수행시간(sec)		질의 결과 수
	Kweelt	본 논문의 방법	
1	24	5	168
2	59	7	338
4	155	13	677
10	540	28	1668
20	2850	47	3357
40	■	89	6711
100	■	179	16702

표 2와 표 3의 질의 수행 결과를 보면 본 논문에서 제안한 색인 기법을 Kweelt에 구현한 경우와 색인을 사용하지 않는 Kweelt의 성능에 많은 차이가 있음을 알 수 있다. 또 색인을 사용하지 않는 Kweelt의 경우 대용량의 XML 데이터는 처리가 불가능함을 볼 수 있으며 본 논문에서 제안한 색인의 경우 문서의 크기가 커짐에 따라 질의 수행 시간이 선형적으로 증가함을 알 수 있다.

5. 결론 및 향후 연구 방향

인터넷을 바탕으로 하는 컴퓨팅 패러다임의 변환은 디지털 정보 교환의 표준으로 확고한 자리를 굳힌 XML의 사용을 가속화시켰다. 이로 인해 XML 데이터의 양이 기하급수

적으로 증가함으로써 보다 효율적으로 XML 데이터를 저장하고 질의하기 위한 연구가 활발히 진행되고 있다. 본 논문에서는 대용량의 데이터 중심 XML을 효과적으로 관리하기 위한 방안으로 그래프 중심의 색인 방법을 연구 구현하였다. 본 논문에서 제안한 색인 기법은 XML 데이터의 구조 정보를 크게 3개의 구성 성분으로 나타 내었다. 첫째, DTD를 공유하는 XML 데이터의 구조 정보를 표현하는 스키마 그래프 둘째, XML 데이터를 표현하는 데이터 그래프 셋째, 데이터 그래프를 색인화한 데이터 그래프 인덱스로 이루어진다. 데이터 그래프는 XML 데이터의 구조 정보를 잘 표현할 수 있는 트리 형태로 표현되고 이를 깊이우선 탐색 기법을 이용하여 노드를 방문하면서 각 노드마다 고유한 노드의 아이디를 부여하며 이를 사용하여 노드의 구조 정보를 색인 한다. 스키마 그래프는 XML 데이터의 DTD를 사용하여 XML 데이터의 구조를 트리 형태로 표현하고 각 단말 노드에 구조 정보와 일치하는 데이터 그래프의 노드 아이디 색인을 연결하여 데이터 그래프 탐색이 가능하게 한다. 마지막으로 각 그래프마다 노드 아이디를 키로 사용하는 B+트리에 각 노드를 사상하고 B+트리를 디스크에 저장하여 이들 색인 모델에 대해 지속성을 부여하였다.

본 논문에서 제안한 색인 기법을 통해 XML 데이터의 구조를 이용한 다양한 구조적 질의를 효과적으로 처리 하여 XML 데이터에 대한 효율적인 검색을 지원할 수 있게 되었다. 또한 XML 데이터의 크기가 커짐에 따라 기하급수적으로 늘어나던 질의 처리시간을 XML 데이터의 크기가 커짐에 따라 선형적으로 늘어날 수 있게 됨으로써 대용량의 XML 데이터에 대해서도 빠른 처리가 가능하게 되었다.

향후 연구로서는 본 논문에서 제안한 색인모델을 XML 데이터의 갱신이 발생하는 동적인 환경에 적용하기 위한 연구가 필요하고 나아가 제안한 색인 기법을 XML 데이터의 질의최적화에 활용할 수 있는 연구가 필요하다.

참 고 문 헌

[1] Bray, T., Paoli, J., Sperberg-McQueen, C., "Extensible Markup Language(XML) 1.0," http://www.w3c.org/TR/1998/REC-xml-19980219/.

[2] Tuong Dao, Ron Sacks-Davis, James A. Thom, "An Indexing Scheme for Structured Documents and its Implementation.", Proceedings of the Fifth International Conference on Database Systems for Advanced Applications(DASFAA '97), pp.125-134, 1997.

[3] Tuong Dao, "An Indexing Model for Structured Documents to Support Queries on Content, Structure and Attributes.", Proceedings of ADL '98, pp.88-97, 1998.

[4] Chow, J. H., Cheng, J., Chang, D., Xu, J., "Index Design for Structured Documents Based on Abstraction.", Proceedings of the 6th International Conference on Database Systems for Advanced Applications, pp.98-96, 1999.

[5] B. Cooper, N. Sample, M. J. Franlin, G. R. Hjaltason, and M. Shadmon. "A fast index for semi-structured data." In Proceedings of the Conference on Very Large Data Bases, 2001: 341-350.

[6] Chin-Wan Chung, Jun-Ki Min, and Kyuseok Shim. "APEX: An Adaptive Path Index for XML Data." In Proceedings of the ACM SIGMOD International Conference on the Management of Data, 2002: 121-132.

[7] H. Garcia-Molina, J. Hammer, K. Ireland, Y. Papakonstantinou, J. Ullman, and Jennifer Widom. "Integrating and Accessing Heterogeneous Information Source in TSIMMIS". In Proceedings of the AAAI Symposium on Information Gathering. pages 61-64, 3 1995.

[8] Jason McHugh, Serge Abiteboul, Roy Goldman, Dallan Quass, and Jennifer Widom. "Lore: A Database Management System for Semi-structured Data". SIGMOD Record, 26(3), 1997.

[9] Wolfgang Meier, eXist: An Open Source Native XML Database, "http://exist-db.org/".

[10] James Owen, Erik Voges, A Generic Indexing Mechanism For Persistent Java "http://people.cs.uct.ac.za/~evoges/web/".

[11] Arnaud Sahuguet, "Kweelt is a framework to query XML data", http://kweelt.sf.net/.

[12] Cover, R., "The XML cover pages," http://oasis-open.org/cover/xml.html/.

[13] CS Department University of Trier Home Page, "DBLP XML Document," http://www.informatik.uni-trier.de/ley/db/.

[14] Chen Qun, Andrew Lim, Kian Win Ong, "D(k)- Index: An Adaptive Structural Summary for Graph-Structured Data.", Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, San Diego, California, USA, pp. 134-144, 2003.

[15] Haixun Wang, Sanghyun Park, Wei Fan, Philip S. Yu, "ViST: A Dynamic Index Method for Querying XML Data by Tree Structures.", Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data, San Diego, California, USA, pp. 110-121, 2003.

저 자 소 개

권국봉(Guk-bong Kwon)
2002년 계명대학교 컴퓨터공학과(학사)
2004년 계명대학교 컴퓨터공학과(석사)

관심분야 : 리눅스, XQuery, 오라클, XML 데이터베이스
Phone : 016-584-1976
E-mail : gbkwon@kebi.com

홍동권(Dong-Kweon Hong)
1985년 : 경북대학교 전자공학과(학사)
1992년 : Universisty of Florida 전자계산학과(석사)
1995년 : Universisty of Florida 전자계산학과(박사)
1985년~1990년 : 한국전자통신연구원
1996년~1997년 : 한국전자통신연구원
2004년 : 정보처리학회논문지 제11권-D권 제 3호(6월)
1997년~현재 : 계명대학교 공학부 컴퓨터공학 전공 부 교수

관심분야 : 능동 실시간 데이터베이스, 병렬처리, 멀티 미디어 처리, XML 데이터베이스
Phone : 011-555-7070
E-mail : dkhong@kmu.ac.kr