韓國 컴퓨터情報學會 論文誌
第13卷 第4號, 2008. 7.

2008-13-4-3-5

모바일 데이터 브로드캐스팅을 위한 트리 기반의 인덱싱 방법

박 미 화*, 이 용 규**

A Tree-Based Indexing Method for Mobile Data Broadcasting

Mee Hwa Park*, Yong Kyu Lee**

요 약

무선 모바일 환경에서 통신 장비의 에너지와 전송 대역폭 효율을 위해 방송 기법이 널리 사용되고 있다. 기존에는 비계층적 데이터를 대상으로 한 인덱싱 연구들이 있었으나, 웹과 이동통신 환경에서 널리 사용되는 XML 데이터에 대한 방송 인덱싱 연구는 미미한 실정이다. 본 연구에서는 XML 문서에 대한 새로운 방송 인덱싱 방법으로 TOP 트리를 제안한다. TOP 트리는 XML 문서에 포함된 엘리먼트들을 같은 경로를 갖는 엘리먼트 그룹으로 분류한 후, 해당 그룹을 순서화된 고유 ID가 부여된 노드로 구성하고 엘리먼트 그룹 간의 관계를 간선으로 연결한 경로 요약 트리이다. 본 논문에서는 TOP 트리 기반 방송 스트림 생성 방법과 다중 경로 질의 처리 방법을 제안하고 실험을 통해 제안 방법의 우수성을 입증한다.

Abstract

In this mobile computing environment, data broadcasting is widely used to resolve the problem of limited power and bandwidth of mobile equipments. Most previous broadcast indexing methods concentrate on flat data. However, with the growing popularity of XML, an increasing amount of information is being stored and exchanged in the XML format. We propose a novel indexing method, called TOP tree(Tree Ordering based Path summary tree), for indexing XML document on mobile broadcast environments. TOP tree is a path summary tree which provides a concise structure summary at group level using global IDs and element information at local level using local IDs. Based on the TOP tree representation, we suggest a broadcast stream generation and query processing method that efficiently handles not only simple path queries but also multiple path queries. We have compared our indexing method with other indexing methods. Evaluation results show that our approaches can effectively improve the access time and tune-in time in a wireless broadcasting environment.

▸ Keyword : 이동 방송(Mobile Broadcast), 방송 인덱싱(Broadcast Indexing), XML 인덱스(XML Index)

• 제1저자 : 박미화 교신저자 : 이용규
• 접수일 : 2008. 4. 11, 심사일 : 2008. 6. 3, 심사완료일 : 2008. 7. 25.
* 동국대학교 컴퓨터공학과 박사과정 ** 동국대학교 컴퓨터공학과 교수

Ⅰ. 서 론

무선 통신 기술의 발달과 고성능 휴대용 단말기의 등장으로 사용자들이 자유로이 이동하면서 네트워크상의 정보를 접근할 수 있는 이동 컴퓨팅이 빠른 속도로 확산되고 있다. 이에 무선 통신 환경의 협소한 대역폭과 이동 단말기의 제한된 배터리 용량 문제를 보완하면서 사용자에게 효율적으로 정보를 전송할 수 있는 방송 기술이 널리 사용되고 있다[1][2][3][4][5].

이동 방송 환경에서 사용자가 필요한 데이터를 전송받기 위해서는 평균적으로 데이터가 방송될 때까지 기다리는 접근 시간과 실제로 데이터를 전송받는 튜닝시간이 소요된다. 정보 서비스를 제공하는 응용들에서 접근 시간은 서비스의 품질을 결정하는 중요한 요인이며 튜닝 시간은 이동 단말기의 배터리 소비와 밀접한 관련이 있다. 접근 시간과 튜닝 시간을 줄이기 위해 방송 시간정보를 제공하는 방송 인덱싱에 대한 다양한 연구들이 진행되고 있다[6][7][8][9][10][11].

기존의 방송 인덱싱 연구들은 방송될 데이터 항목의 키(key)값과 함께 방송 시간을 방송 스트림의 앞부분 또는 중간에 반복 배치함으로써 사용자의 튜닝시간을 줄이고자 하였다.

그러나 XML과 같이 문서의 내용과 함께 구조 정보를 같이 표현하는 구조화된 데이터의 사용이 증가되고 있어, 이러한 데이터들에 대한 방송 인덱싱 연구가 필요한 실정이다. 특히 XML은 인터넷상의 교환과 데이터 표현의 표준으로 자리잡고 있어 많은 정보가 XML 형태로 사용되고 있으며, XML의 연구 분야 또한 무선 환경으로 확대되고 있는 상황이므로 XML 데이터를 효율적으로 방송하기 위한 연구가 필요하다.

기존의 XML 방송 인덱싱 연구들은 단순 경로 질의(simple path query)를 지원하므로 '*'나 '?', '//'를 포함하는 질의나 '|'나 'AND', 'OR'를 포함하는 복합질의를 처리하는데 비효율적이다.

본 논문의 목적은 다양한 경로를 포함한 구조적 데이터에 대한 빠른 접근을 제공하면서도 인덱스의 크기를 감소시킴으로써 무선 통신 사용자의 접근 시간을 크게 회생하지 않는 새로운 방송 인덱싱 방법을 제안하고 평가하는 것이다.

본 논문에서 수행한 인덱싱 연구의 내용과 범위는 다음과 같다.

① XML 문서의 구조적 정보에 대한 빠른 접근을 제공하면서도 인덱스의 크기를 줄일 수 있는 트리 기반 인덱스를 제안한다.

② 제안한 인덱스 트리를 이용하여 튜닝 시간을 줄이면서 접근 시간의 증가량을 최소화하는 방송 데이터 스트림을 구성한다.

③ 방송 스트림 상에서의 다양한 경로 질의 처리 방법을 제시한다.

논문의 나머지 구성은 다음과 같다. 2절에서는 관련연구에 관해 기술하고, 3절에서는 TOP 트리 기반의 새로운 XML 방송 인덱싱 기법을 제안하고 이를 이용한 방송 스트림 생성 방법과 생성된 방송 스트림에서의 다양한 질의 처리 방법을 예제를 이용하여 설명한다. 4절에서 제안된 인덱싱 기법에 대한 성능 평가와 결과 분석을 수행하고, 5절에서 결론을 맺는다.

Ⅱ. 관련연구

방송 인덱싱은 데이터가 방송될 시간 정보를 방송 스트림에 포함함으로써 방송으로 전달되는 데이터들을 클라이언트들이 적은 에너지 소모를 통해 찾을 수 있도록 하는 방법이다.

기존의 방송 인덱싱 연구들의 대부분은 단일 데이터 항목들을 다룬다[8][12][13][14][15]. 모바일 환경에서 XML 데이터의 방송을 위한 인덱싱 연구로는 경로 요약 인덱싱[6]과 분산 인덱싱[7]이 있다.

경로 요약 인덱싱은 동일한 패스의 중복이 많이 발생하는 XML 문서의 노드들을 그룹으로 묶어 패스의 중복을 제거함으로써 원본 XML 문서의 경로 요약본(path summary)을 만들고 XML 문서의 구조정보를 무선 방송 데이터의 색인으로 활용하는 방법이다. 경로 요약 인덱싱에서는 같은 경로를 갖는 데이터를 연이어 방송함으로써 단일 경로 질의에 유용하다는 장점이 있으나, XML 문서의 원문이 변형되므로 특정 응용 시스템에서만 활용될 수 있다는 단점이 있다. 또한, 인덱스가 방송 스트림에 한번만 나타나는 (1:1) 브로드캐스팅에 속해 중간에 방송을 청취하게 되어 인덱스를 듣지 못했을 경우 다음 브로드캐스팅 사이클까지 대기해야하는 단점이 있다.

분산 인덱싱은 XML 데이터 및 인덱스 정보를 부분적으로 반복 배치하여 스트림을 구성한다. 즉, XML 데이터와 색인 정보를 2-레벨로 구분하여 색인 및 데이터의 중복 배치 영역을 설정한다. XML 트리의 상위 h 레벨까지의 부분을 상위레벨이라고 부르며 나머지 h+1에서 H까지의 부분을 하위 레벨이라고 부른다. 상위 레벨에 해당되는 데이터 및 색인 정보를 전체적으로 혹은 부분적으로 반복한다. 분산 인덱싱은 h값에 따라 반복의 정도가 결정되며 최적의 성능을 나타내기 위한 h값의 설정이 필요하다.

기존의 XML 방송 인덱싱 연구들은 단순한 경로 질의 처리에는 유용하지만, XML 문서에 포함된 다양한 구조적 계층적 데이터 정보를 검색하는 다중 경로 질의와 부분 부합 질의를 처리하기에는 비효율적이다. 또한 XML 방송 인덱싱 방법의 활용성을 고려하여 XML 원문 구조를 변형하지 않는 인덱싱 연구가 필요하다.

III. TOP 트리 기반 방송 인덱싱

3.1 TOP 트리(Tree Ordering based Path-summary tree)

TOP 트리는 XML 문서에 포함된 엘리먼트들을 같은 경로를 갖는 엘리먼트 그룹으로 분류한 후, 해당 그룹을 순서화된 고유 ID가 부여된 노드로 구성하고 엘리먼트 그룹 간의 관계를 간선으로 연결한 트리이다.

본 논문에서 제안한 TOP 트리의 특징은 다음과 같다.

① TOP 트리는 모든 노드의 차수가 K인 균형 트리이다. TOP 트리의 차수 K는 해당 XML 문서의 DTD 트리가 갖는 최대 차수와 같다.
② 노드를 유일하게 구별하고 노드들 간의 관계를 쉽고 빠르게 구별하기 위해 ID기반 번호부여 방법[16]에 따라 ID를 부여한다.
③ TOP 트리는 경로 요약 트리이다. TOP 트리의 각 노드는 같은 경로를 갖는 엘리먼트들의 그룹이다. 이를 통해 인덱스 크기 증가를 최소화할 수 있으며 경로 질의를 효과적으로 처리할 수 있다.
④ XML 문서의 각 엘리먼트는 같은 경로를 갖는 TOP 트리 노드에 표현되며, 문서에 나타난 순서에 따라 노드 안에서의 지역 ID를 부여받는다. 각 엘리먼트는 해당 노드가 갖는 그룹ID와 각 엘리먼트가 갖는 지역ID를 이용해서 구별된다.

3.1.1 TOP 트리 구조

TOP 트리는 XML 문서에 나타난 엘리먼트의 정보를 저장한 노드와 엘리먼트의 관계를 표현하는 간선으로 구성된다.

대부분의 XML 인덱스 트리가 XML 문서에 포함된 엘리먼트들을 독립적인 노드로 구성함으로써 인덱스 트리의 크기가 크고 경로 질의를 효과적으로 처리하지 못함에 비해 TOP 트리는 같은 경로를 갖는 엘리먼트에 대한 정보를 하나의 노드에 표현함으로써 인덱스 트리의 크기를 줄일 수 있으며 경로 질의의 처리 시간을 단축시킬 수 있다.

TOP 트리 노드에 저장되는 정보는 다음과 같다.

① GID(Group ID) : 같은 경로와 같은 이름을 갖는 엘리먼트의 집합을 구별하기 위한 그룹의 고유번호이며, 트리 번호 부여 방법[16]에 의해 부여된다.
② Name : 같은 경로를 갖는 엘리먼트의 이름
③ Sibling ID : 노드의 오른쪽 형제 노드의 GID
④ 엘리먼트 단위(Element Unit) : 같은 경로와 같은 이름을 갖는 엘리먼트들의 정보를 저장한 단위, 엘리먼트 단위는 해당 엘리먼트가 문서에 나타난 순서대로 노드에 저장된다.

엘리먼트 단위가 포함한 정보는 다음과 같다.

① LID(Local ID) : 엘리먼트 단위가 노드에 저장되는 순서에 따라 부여된 고유번호, 같은 노드에 저장된 엘리먼트 단위들은 서로 다른 LID를 갖는다.
② PID(Parent ID) : 엘리먼트 단위에 저장된 해당 엘리먼트의 부모에 대한 엘리먼트 단위의 LID.
③ Start 위치 : XML 문서에 포함된 엘리먼트의 방송 시작 시간 정보
④ End 위치 : XML 문서에 포함된 엘리먼트의 방송 종료 시간 정보

TOP 트리의 노드 정보를 그림으로 표현하면 다음과 같다.

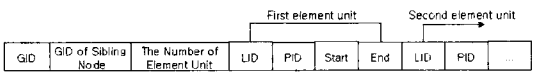

그림 1. TOP 트리의 노드 정보
Fig 1. Node Information of TOP tree

3.1.2 경로 요약 정보

TOP 트리는 XML 문서의 경로 요약 정보를 추출하기 위해 DTD를 활용한다. <그림 2>는 예제로 사용된 XML 문서의 DTD를 보이고 있다. 예제 문서에는 동일한 경로와 이름을 갖는 엘리먼트들이 있다. '/SigmodRecord/issue' 엘리먼트는 67개가 존재하며, 총 라인수가 16248개이다. 또한 issue 엘리먼트의 자손 중에 article 엘리먼트가 10개 이상 존재하므로 XML 문서에 포함된 엘리먼트들을 인덱스 트리의 노드로 구성할 경우, 인덱스 트리의 크기가 커지게 됨은 자명한 일이다. 따라서 본 논문에서는 <그림 2>와 같이 XML 문서의 DTD 정보를 이용하여 DTD 트리를 구성하고 이를 이용하여 방송 인덱스 트리인 TOP 트리를 생성한다.

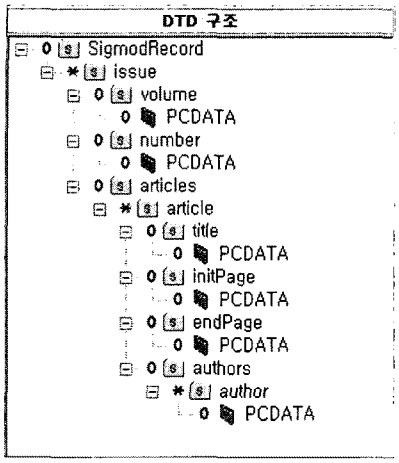

그림 2. SigmodRecord.xml에 대한 DTD 트리
Fig 2. DTD tree of SigmodRecord.xml

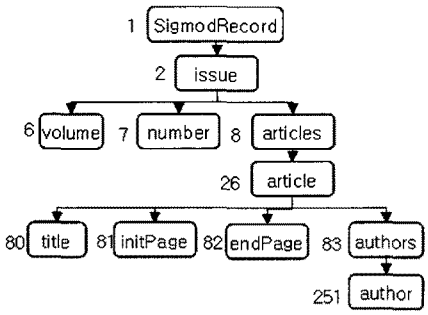

그림 3. 〈그림 2〉에 대한 TOP 트리
Fig 3. TOP tree of 〈Fig 2〉

3.1.3 TOP 트리 구성 예

〈그림 2〉에 대한 TOP 트리를 구성하면 〈그림 3〉과 같으며, TOP 트리의 각 노드 정보는 다음과 같다.

[1, SigmodRecord,\0, {(1, \0, 10, 16248) }]
[2, issue,\0,{(1, 1, 11, 56), (2, 1, 58, 110), …}]
[6, volume,7,{(1, 1, 12, 12), (2, 2, 59, 59),…}]
[7, number,8,{(1, 1, 13, 13), (2, 2, 60, 60),…}]
[8, articles,\0,{(1, 1, 14, 55), (2, 2, 61, 109),…}]
이후 생략.....

각 노드 정보에서 '\0'은 오른쪽 형제 노드가 없다는 표시이다. Start와 End 정보는 해당 엘리먼트가 나타난 시작 라인 번호와 끝 라인 번호를 기입하였다.

3.1.4 TOP 트리 연산

TOP 트리는 트리 번호 부여 방법[16]에 의해 각 노드에 고유번호를 부여함으로써 특정 노드의 GID만으로 부모 자식 관계에 있는 노드들의 GID를 계산할 수 있다. 〈표 1〉은 주어진 노드들과 부모, 자식, 형제, 조상, 후손 관계를 갖는 노드들의 GID를 구하는 함수들로 무선 단말에서 경로질의를 처리할 때 유용하게 사용된다.

표 1. TOP 트리 연산
Table 1. Operations on TOP tree

연 산	설 명
G_child(i, j)	GID가 i인 노드의 j번째 자식 노드의 GID를 반환
G_parent(i)	GID가 i인 노드의 부모 노드의 GID를 반환
G_descendents(i, j)	GID가 i인 노드가 GID가 j인 노드의 자손인지를 판단한 후 결과 값을 반환
G_ancestors(i, j)	GID가 i인 노드가 GID가 j인 노드의 조상인지를 판단한 후 결과 값을 반환

$$k*(i-1)+j+1 \quad \cdots\cdots (3.1)$$

G_parent(i)는 차수가 k인 TOP 트리에서 수식 (3.2)를 이용하여 GID가 i인 노드의 부모 노드의 GID를 계산 한 후 반환하는 함수이다.

$$\left\lceil \frac{i-2}{k}+1 \right\rceil \quad \cdots\cdots (3.2)$$

G_descendents(i, j)는 〈알고리즘 1〉에 따라 차수가 k인 TOP 트리에서 GID가 i인 노드가 GID가 j인 노드의 자손 노드인지를 판단 한 후 결과 값을 돌려주는 함수이다.

```
Algorithm G_descendents
Input : i, j
Output : true or false
{
 if ( i > j ) {
  set lvl = ⌈ log_k j ⌉
  set min_id = G_child(j, 1)
  set max_id = G_child(j, k)
  for each h = [ lvl to Max_level - 1],
    if i >= min_id and i <= max_id,
          return true
      set min_id = G_child(min_id, 1),
      set max_id = G_child(max_id, k).
 }
 return false
}
```

알고리즘 1. G_descendents 알고리즘
Algorithm 1. Algorithm of the G_descendents

```
Algorithm G_ancestors
Input : i, j
Output : true or false
{
 if  i < j {
  set level = ⌈ log_k j ⌉
  set pat_id = G_parent(j)
  for each h = [ 1 to level ]
     if i = pat_id,  return true
     set pat_id = G_parent(pat_id)
 }
 return false
}
```

알고리즘 2. G_ancestors 알고리즘
Algorithm 2. Algorithm of the G_ancestors

G_ancestors(i, j)는 〈알고리즘 2〉에 따라 차수가 k인 TOP 트리에서 GID가 i인 노드가 GID가 j인 노드의 부모 노드인지를 판단 한 후 결과 값을 돌려주는 함수이다.

3.2 TOP 트리 기반 방송 스트림 생성

TOP 트리 기반 방송 스트림은 컨트롤 인덱스 스트림, TOP 인덱스 스트림, XML 원문 데이터 스트림으로 구성된다.

컨트롤 인덱스 스트림은 TOP Tree를 BFS로 순회하여 각 노드의 GID와 Name을 추출하고 TOP 트리 스트림의 각 노드가 방송될 오프셋 정보로 구성된 스트림을 생성한 것이다. 컨트롤 인덱스 스트림의 인덱스 항목은 〈GID, Name, TOP 트리 스트림의 해당 노드에 대한 방송 위치 정보〉로 구성된다.

방송을 이용하는 사용자들은 컨트롤 인덱스 스트림을 수신받아 자신이 원하는 데이터에 대한 TOP 트리 스트림의 방송 시간을 알아 낸 뒤, 해당 노드 정보를 청취한 후, 실제 데이터가 방송되는 시점을 알 수 있다.

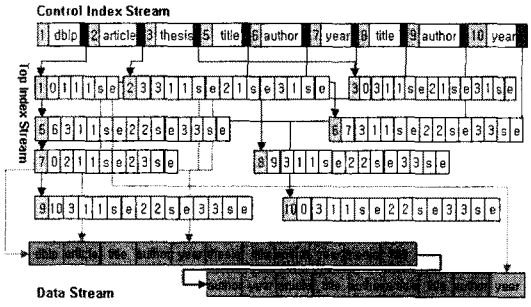

그림 4. 〈그림 3〉에 대한 TOP 트리 기반 방송 스트림
Fig 4. TOP tree based Broadcasting Stream of the 〈Fig 3〉

컨트롤 인덱스를 이용할 경우, 모든 질의는 컨트롤 인덱스에서 질의 결과에 포함될 노드들의 GID를 찾은 후 해당 노드들에 대한 인덱스가 방송될 TOP 트리 방송 스트림 내의 위치로 바로 이동하고, 해당 데이터가 방송될 시간까지 대기 상태로 있다가 수신하는 과정으로 처리된다. 다중 경로 질의 또는 부분 부합 질의도 컨트롤 인덱스에 대한 검색만으로 질의 결과에 포함될 노드들의 GID를 찾을 수 있으므로, 사용자들의 접근 시간을 줄일 수 있다.

TOP 인덱스 스트림은 TOP 트리를 BFS로 순회하여 생성한 방송 스트림으로 XML 데이터들 중 같은 경로를 갖는 엘리먼트들의 방송 시간 정보를 한 번에 제공한다.

XML 원문 데이터 스트림은 XML 문서를 원문 변형 없이 방송스트림으로 구성한 것이다. 〈그림 3〉의 예에 대한 TOP 트리 기반 방송 스트림을 (1:1) 방식으로 구성하면 〈그림 4〉와 같다.

TOP 트리 기반 방송 스트림을 (1:1) 방식으로 구성할 경우, 컨트롤 인덱스 스트림, TOP 인덱스 스트림, 데이터 스트림이 차례로 방송하게 된다. TOP 트리 기반 방송 스트림을 (1:m) 방식으로 구성할 경우에는 방송 스트림에 컨트롤 인덱스와 TOP 인덱스 스트림이 m회 반복된다. TOP 트리 기반 방송 스트림을 분산 방송 방식으로 구성할 경우에는 컨트롤 인덱스만 m회 반복된다. TOP 인덱스 스트림과 데이터 스트림은 m개의 조각으로 분리되어 한 방송 주기 동안 흩어져서 방송된다.

3.3 TOP 트리 기반 방송에서의 질의 처리

이동 단말기에서의 질의는 XPath나 XQuery와 같은 표준 XML 질의 언어로 표현된다고 가정한다. 다음은 XPath 형태로 표현된 다양한 경로 질의 예제이다.

- Q1 : /dblp/article/title
- Q2 : /dblp/*/title
- Q3 : //article/title[year] | //article/year[title]

Q1 질의를 처리하기 위해서 이동 단말은 루트 노드에서부터 G_child() 함수를 이용하여 컨트롤 인덱스 스트림에서 경로가 '/dblp/article/title'인 노드의 GID가 5임을 찾고 해당 노드에 대한 TOP 인덱스 스트림이 방송될 때까지 대기 모드로 작동한다. TOP 인덱스 스트림의 GID 5번 노드에서 각 데이터 항목이 방송될 위치를 파악한 후 데이터의 방송 시간까지 다시 대기 모드로 동작한다.

Q2 질의를 처리하기 위해서 이동 단말은 컨트롤 인덱스

스트림과 G_descendents() 함수를 이용하여 Name이 'title'인 노드들중에서 dblp 노드의 후손 노드들의 GID가 5와 8을 찾는다. 해당 노드에 대한 TOP 인덱스 스트림이 방송될 때까지 대기 모드로 작동한 후 TOP 인덱스 스트림의 GID 5번과 8번 노드에서 각 데이터 항목이 방송될 위치를 파악한다.

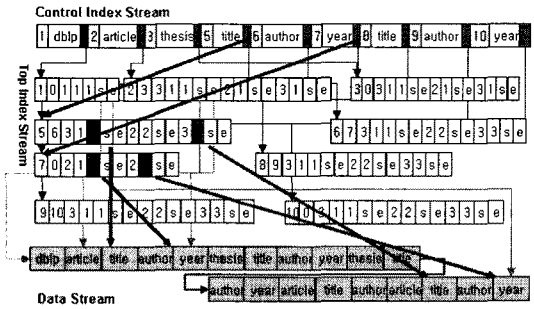

그림 5. Q3 질의를 처리하기 위한 이동 단말의 접근 절차
Fig 5. Access Steps for Q3 query processing

Q3 질의를 처리하기 위해서 이동 단말은 컨트롤 인덱스 스트림에서 모든 article노드를 찾고 article 노드의 자식들 중 이름이 title인 노드와 year인 노드에 대한 TOP 인덱스 스트림이 방송될 때까지 대기 모드로 작동한 후 TOP 인덱스 스트림의 5번과 7번 노드에서 각 엘리먼트 단위의 PID가 같은 데이터 항목이 방송될 위치를 파악한다.

IV. 인덱싱 성능평가

4.1 성능평가에 사용된 기호

본 절에서는 제안 기법과 본 연구와의 비교분석을 위해 해석적 방법에 의한 성능 분석 결과를 기술한다. 〈표 2〉는 분석을 위해 사용된 기호를 나타낸다.

표 2. 성능 평가에 사용된 기호
Table 2. Notations

용 어	설 명
k	TOP 트리의 최대 차수
K	XML Data 트리의 최대 차수
H	XML Data 트리와 TOP 트리의 높이
l	TOP 트리 노드 당 평균 엘리먼트 수
D	XML Data 트리의 평균 노드 크기
C	컨트롤 인덱스 트리의 평균 노드 크기
W	데이터를 청취에 소요되는 시간
Y	XML 인덱스 트리의 노드 크기

XML 데이터 트리가 차수 K이고, 높이가 H, XML 데이터 트리의 평균 노드 크기가 D인 완전 K-원 트리로 표현된다고 하면, XML 데이터 스트림의 크기는 다음과 같이 표현될 수 있다.

$$Sizeof DATA = D \times \sum_{i=1}^{H} K^i = D \times \frac{K^H - 1}{K-1} \quad \cdots\cdots (4.1)$$

Control Index는 TOP 트리 인덱스와 같은 차수와 높이를 갖는 K-원 트리이므로 TOP 트리가 차수 k, 높이 H인 완전 k-원 트리로 표현된다고 하면, Control Index 스트림의 크기는 다음과 같이 표현된다.

$$Sizeof\ Control\ Index = C \times \sum_{i=1}^{h} k^i = C \times \frac{k^h - 1}{k-1} \quad \cdots\cdots (4.2)$$

TOP 인덱스 트리의 노드 크기는 각 노드당 평균 l개의 엘리먼트를 표현한다고 가정할 경우, 'C×l'로 표현할 수 있다. C는 컨트롤 인덱스 트리의 노드 크기로서, 하나의 엘리먼트를 가리키기 위해 필요한 크기를 갖는다. 차수 k, 높이 H인 완전 k-원 트리로 표현된 TOP 인덱스 스트림의 크기는 다음과 같다.

$$Sizedf\ TOPIndex = C \times l \times \sum_{i=1}^{h} k^i = C \times l \times \frac{k^h - 1}{k-1} \quad \cdots\cdots (4.3)$$

4.2 제안 방법의 접근 시간과 튜닝 시간

〈표 3〉과 〈표 4〉, 〈표 5〉는 제안된 1:1, 1:m, 분산 TOP 인덱싱 방법에 대한 인덱스 스트림의 크기와 데이터 스트림의 크기, 접근 시간과 튜닝시간을 기술한 것이다.

접근 시간은 사용자가 청취를 시작한 시점부터 수신을 마칠 때까지 소요되는 시간으로 인덱스를 청취하기 위해 기다리는 시간과 원하는 데이터가 방송될 때까지 대기하는 시간을 더해서 구한다.

튜닝 시간은 모바일 단말이 실제 수신에 사용된 시간으로 컨트롤 인덱스의 방송 시간을 청취하기 위해 하나의 버켓을 수신하는 시간과 컨트롤 인덱스를 검색하기 위해 청취하는 시간 h, 해당되는 TOP 인덱스 스트림의 노드를 수신하는 시간, 그리고 원하는 데이터를 청취하는 시간을 합한 것과 같다.

표 3. 1:1 TOP Index 방법의 스트림 크기와 시간
Table 3. Size of Broadcasting Streams, Access Time and Tune-in time of 1:1 TOP index

기 준	크 기
데이터스트림	*SizeofDATA*
인덱스스트림	*SizeofControlIndex+SizeofTOPIndex*
방송스트림	*SizeofControlIndex+SizeofTOPIndex +SizeofDATA*
평균접근시간	*SizeofControlIndex+SizeofTOPIndex +SizeofDATA*
평균튜닝시간	*2+h+W*

표 4. 1:m TOP Index 방법의 스트림 크기와 시간
Table 4. Size of Broadcasting Streams, Access Time and Tune-in time of 1:m TOP index

기 준	크 기
데이터스트림	*SizeofDATA*
인덱스스트림	*SizeofControlIndex+SizeofTOPIndex*
방송스트림	*m*(SizeofControlIndex+SizeofTOPIndex)+SizeofDATA*
평균접근시간	*0.5*(m+1)(SizeofControlIndex+SizeofTOPIndex+SizeofDATA*1/m)*
평균튜닝시간	*2+h+W*

표 5. 분산 TOP Index 방법의 스트림 크기와 시간
Table 5. Size of Broadcasting Streams, Access Time and Tune-in time of distributed TOP index

기 준	크 기
데이터스트림	*SizeofDATA*
인덱스스트림	*SizeofControlIndex + SizeofTOPIndex*
방송스트림	*m*SizeofControlIndex+SizeofTOPIndex+SizeofDATA*
평균접근시간	*0.5*(m+1)(SizeofControlIndex + (SizeofTOPIndex + SizeofDATA)*1/m)*
평균튜닝시간	*2+h+W*

4.3 기존 연구와의 비교

본 절에서는 기존 연구[7]와의 성능 비교를 위해 인덱스 스트림의 크기와 접근 시간을 비교한다.

XML 데이터 트리가 차수 K이고, 높이가 H, XML 데이터 트리의 평균 노드 크기가 D인 완전 K-원 트리로 표현된다고 하면, XML 데이터 스트림의 크기는 수식 (4.1)과 같이 표현될 수 있다.

[7]에서 제안한 분산 인덱싱 방법은 인덱스를 상위 부분과 하위 부분으로 나누어 상위부분을 반복 방송하는 방법이다. 상위 부분을 HI 인덱스라 하고 하위 부분을 LI 인덱스라 한다. HI 인덱스의 높이 n은 $0 \le n \le H$ 을 만족하는 값이어야 한다. HI의 높이가 n이면 LI의 높이는 H-n, 하위 인덱스 LI의 개수는 Kn이 된다. 상위 인덱스 HI도 하위 인덱스의 개수만큼 반복 방송되므로 [7]에서 제안한 인덱스 스트림은 Kn개의 HI와 Kn개의 LI 인덱스로 구성된다. 따라서 [7]의 분산 인덱스의 크기는 다음 수식과 같다.

$$\begin{aligned} &Sizeof D\_\in dex \\ &= Y \times K^n \times (Sizedf HI + Sizeof LI) \\ &= Y \times K^n \times \left(\frac{K^n-1}{K-1} + \frac{K^{H-n+1}-1}{K-1}\right) \end{aligned} \quad \cdots\cdots\cdots (4.4)$$

[7]의 스트림 크기와 접근시간, 튜닝시간은 다음과 같다.

표 6. 기존 방법[7]의 스트림 크기와 시간 분석
Table 6. Size of Broadcasting Streams, Access Time and Tune-in time of [7]

기 준	크 기
데이터스트림	*SizeofDATA*
인덱스스트림	*SizeofD_Index*
방송스트림	*SizeofD_Index+SizeofDATA*

평균접근시간	$0.5*(K^n+1)/K^n*(SizeofD\_Index+SizeofDATA)$
평균튜닝시간	$2+SizeofD\_Index/K^n+W$

각 방법의 인덱스 스트림의 크기와 데이터 스트림의 크기, 접근 시간과 튜닝시간을 비교하기 위해 다음과 같이 가정하였다.

- XML 데이터 트리는 30%의 가상노드를 갖는 K-원 트리이다.
- [7]의 분산 레벨 n은 최적튜닝시간을 갖는 H/2로 설정한다.
- 다중 방송 회수와 분산 방송할 회수 m은 같은 값을 갖도록 설정한다.
- TOP 트리의 최대 차수 k는 2보다 같거나 크면서 XML 데이터 트리의 최대 차수 K와 같거나 작은 값을 갖도록 설정한다.
- TOP 트리의 높이와 XML 데이터 트리의 높이는 H이다.
- TOP 트리의 노드당 평균 엘리먼트 수는 ceil(DATA 트리의 노드 수/TOPIndex 트리의 노드 수)로 설정한다.
- TOP 트리를 제외한 모든 트리의 노드 크기는 동일하다고 가정하며, 간단하게 1로 표현한다. 즉, D=C=Y=1로 설정한다.
- 튜닝 시간에 포함되는 데이터 수신 시간 W는 각 방법에서 동일한 값이므로, 계산에 포함하지 않는다.

비교를 위해 4≤K≤8, 4≤H≤6, 3≤m≤9 인 모든 경우에 대한 평균접근시간을 측정하였다.

〈그림 6〉은 4≤K≤8, 4≤H≤6, 3≤m≤9인 모든 경우에 대한 각 방법의 평균접근시간을 보인다. 그림을 통해 XML 데이터 트리의 차수가 4≤K≤8일 경우에 제안한 1:1 TOPIndex 방법과 분산 TOPIndex 방법 이 D_Index[7] 방법과 비교했을 때 평균 접근 시간이 짧음을 알 수 있다. 제안한 1:1 TOPIndex와 1:M TOPIndex, 분산 TOPIndex 중에서는 분산 TOPIndex 방법의 접근 시간이 적음을 알 수 있다. 반면, 1:M TOPIndex 방법은 반복횟수가 증가함에 따라 접근시간 증가량이 커서 접근시간이 가장 오래 걸리는 것으로 나타났다.

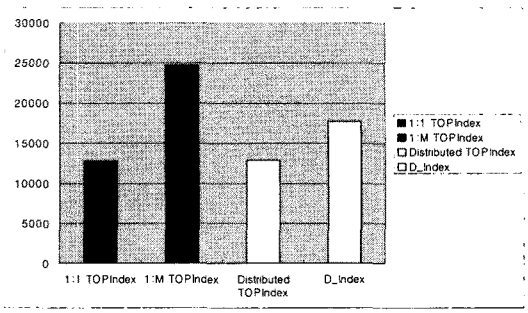

그림 6. 4≤K≤8, 4≤H≤6, 3≤m≤9인 모든 경우의 평균접근시간
Fig 6. Average Access Time with 4≤K≤8, 4≤H≤6, 3≤m≤9

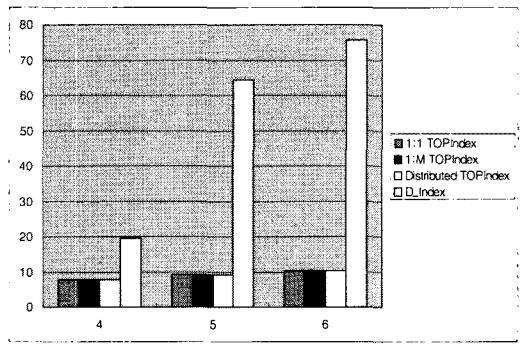

그림 7. K=4, 높이 H(4≤H≤6)에 따른 평균튜닝시간
Fig 7. Average Tune-in Time with K=4, 4≤H≤6

〈그림 7〉은 K=4, 높이 H(4≤H≤6)에 따른 각 방법의 평균튜닝시간을 보인다. D_Index 방법은 K = 4이고, 분할 높이를 H/2로 설정했을 때 튜닝시간을 최소로 갖는다. (1:1) TOPIndex 방법과 (1:M) TOPIndex 방법, 분산 TOPIndex 방법의 튜닝 시간은 동일하므로, 트리 높이 H에 따라 조금씩 상승한 값을 갖게 된다.

TOP 트리를 이용한 인덱싱 방법의 튜닝 시간이 [7]과 비교하여 현저하게 적은 것은 TOP 트리 인덱스가 ID 기반의 번호 부여 방법을 이용해서 인덱스 노드의 GID를 부여하기 때문이다. ID 기반 방법은 Child()와 Parent() 연산을 통해 쉽게 부모/자식 관계에 있는 노드의 GID를 계산 할 수 있기 때문에 경로 질의에서는 트리 높이만큼의 노드만 검색하면 원하는 경로에 있는 노드를 검색할 수 있게 된다. 반면, [7]의 방법은 원하는 데이터가 방송되는 시간을 검색하기 위해 반복 방송되는 HI와 LI 인덱스를 반드시 한번은 읽어야 하므로, 제안하는 방법보다 튜닝 시간이 길다.

그래프 분석을 통해 제안된 방법이 기존 방법[7]보다 접근시간과 튜닝시간에서 우수함을 보였다. 또한 인덱스의 크기와 방송스트림의 크기 비교를 통해 제안 방법이 기존의 방법

보다 향상된 방법임을 보였다.

다중 경로 질의 처리 능력을 비교할 경우에도 [7]의 방법에서는 다중 질의를 여러 개의 서브 질의로 분할하여 처리한 후 질의 결과를 취합하는 반면, 제안 방법에서는 컨트롤 인덱스를 이용하여 서브 질의에 대한 결과 값이 방송될 위치를 한 번에 파악할 수 있으므로, 제안 방법이 다중 경로질의에 대해 더 효율적임을 알 수 있다.

Ⅵ. 결론

본 논문에서는 모바일 방송 환경에서의 단일 데이터 항목을 다루는 기존의 인덱싱 연구와 달리 문서의 구조 정보를 포함하는 계층적인 XML 문서를 방송하기 위한 인덱싱 방법을 제안하였다.

TOP 트리 인덱스는 모바일 방송 환경에서의 접근 시간 증가를 막기 위해 경로 요약 정보와 ID 기반의 순서화 방법을 적용하여 구성하였다. 또한 컨트롤 인덱스를 이용하여 방송 스트림을 구성함으로써 다양한 경로 질의를 효율적으로 처리할 수 있음을 보였다.

실험 및 분석 과정에서는 기존 연구와의 분석적 접근 방법에 의한 성능 비교를 수행하였다. 이를 통해, 제안 방법의 우수성을 알 수 있었다.

향후에는 TOPIndex를 활용하여 여러 XML 문서를 효율적으로 방송하는 방법에 대한 연구를 수행할 것이다.

본 논문에서 제안한 새로운 인덱스 구조는 이동 통신 기술이 발달함으로써 증가할 것으로 예상되는 이동 정보 서비스 이용자들의 서비스 만족도 향상에 기여할 것으로 기대된다.

참고문헌

[1] 최성환, 정성원, 이송이, "이동컴퓨팅환경에서 데이타의 접근 빈도 및 시맨틱 관계를 고려한 방송 방법," 한국정보과학회 논문지, 제30권 제5호, pp. 476-493, 2003.

[2] 이상돈, "효율적인 다중 데이터 접근을 위한 방송 스케쥴 생성," 한국정보과학회 논문지, 제29권 제4호, pp. 285-296, 2002.

[3] S. Acharya, R. Alonso, M. J. Franklin, and S. Zdonik, "Broadcast Disks: data management for asymmetric communications Environment," In Proceedings of the ACM SIGMOD Conference on Management of Data, pp. 199-210, 1995.

[4] A. R. Hurson, Y. C. Chehadeh, and L. L. Miller, "Object Organization on a Single Broadcast Channel in a Global Information Sharing Environment," In Proceedings of the 24th Conference on EUROMICRO, vol. 2, pp. 1021-1028, 1998.

[5] S. Jiang and N. H. Vaidya. "Scheduling data broadcast to impatient users," In Proceedings of the ACM International Workshop on Data Engineering for Wireless and Mobile Access, pp. 52-59. 1999.

[6] 박상현, 최재호, 이상근, "모바일 무선 네트워크 환경에서 에너지 효율적인 XML 데이터 방송 기법," 한국정보과학회 2005년 추계학술대회 논문집, vol. 32, no. 2, pp 7-9, 2005.

[7] 정연돈, 이지연, "무선 XML 스트림을 위한 색인 기법," 한국정보과학회 논문지 , 제32권 제3호, pp. 416-428, 2005.

[8] 정연돈, 이지연, "B2V-Tree: 무선 데이터 스트림에서 부분 부합 질의를 위한 색인 기법," 한국정보과학회 논문지, 제32권 제3호, pp. 285-296, 2005.

[9] 김충수, 정연돈, "무선 XML 데이터 방송을 위한 에너지 효율적인 스트리밍 기법," 데이타베이스연구회 논문지, 제21권 제3호, pp 35-48, 2005.

[10] 김충수, 박창섭, 정연돈, "이동 컴퓨팅 환경에서 XML 데이타의 에너지 효율적인 방송," 한국정보과학회 논문지, 제33권 제1호, pp. 117-128, 2006.

[11] Alvin T. S. Chan, Hong Va Leong, and Eugene Y. C. Wong, "Xstream:A Middleware for Streaming XML Contents over Wireless Environments," IEEE Transactions on Software Engineering, vol. 30, no. 12, pp. 918-935, 2004.

[12] T. Imielinski, S. Viswanathan, and B. R. Badrinath. "Power Efficiency Filtering of Data on Air," In Proceedings of the International Conference on Extending Database Technology, pp 245-258, 1994.

[13] W. C. Lee and D. L. Lee. "Using Signature Techniques for Information Filtering in Wireless and Mobile Environments," Special Issue on Database and Mobile Computing, Journal on Distributed and Parallel Databases, pp. 205-227, 1996.

[14] T. Imielinski, S. Viswanathan, and B. R. Badrinath, "Data on Air: Organization and Access," IEEE Transactions on Knowledge and Data Engineering, vol. 9, no. 3, pp. 353-372, 1997.

[15] Y. D. Chung and M. H. Kim, "Energy-Efficient Indexing for Wireless Broadcast Data," KIST Technical Paper CS/TR-98-120, 1998.

[16] Y. K. Lee, S. J. Yoo, K. Yoon, and P. Bruce Berra, "Index Structures for Structured Documents," In Proceedings of the First ACM International Conference on Digital Libraries, pp. 91-99, 1996.

저 자 소 개

박미화

(E-mail: meehwap@dgu.edu)

1997년 : 동국대학교 컴퓨터공학과 (학사)

1999년 : 동국대학교 컴퓨터공학과 (석사)

2008년 : 동국대학교 컴퓨터공학과 (박사)

관심분야 : 데이터베이스 시스템, 모바일 데이터베이스, 정보검색

이용규

(E-mail: yklee@dgu.edu)

1986년 : 동국대학교 전자계산학과 (학사)

1988년 : 한국과학기술원 전산학과 (석사)

1996년 : Syracuse University (전산학박사)

1978년~83년: 정보통신부 국가공무원

1988년~93년: 한국국방연구원 선임연구원

1996년~97년: 한국통신 선임연구원

2002년~03년: 콜로라도대학 컴퓨터학과방문교수

1997년~현재: 동국대학교 컴퓨터공학과 교수

관심분야 : 데이터베이스 시스템, 모바일 데이터베이스, XML, 정보검색, e-비즈니스 시스템