Journal of Korea Multimedia Society Vol. 9, No. 3, March 2006(pp. 265-278)

2D-THI: XML데이터베이스를 위한 이차원 타입상속 계층색인

이 종 학†

요 약

본 논문에서는 XML 데이터베이스의 타입상속 계층에 대한 색인기법으로 이차원 색인구조를 이용하는 이차원 타입상속 색인기법인 2D-THI를 제안한다. XML 스키마는 타입상속을 지원하는 XML 문서를 위한 스키마 모델 중에 하나이다. 기존의 XML 데이터베이스를 위한 색인기법은 XML 스키마상의 타입상속 계층에 대한 XML 질의를 지원하지 못한다. 따라서 본 논문에서는 XML 질의의 타입상속 계층을 지원하기 위한 색인기법으로 다차원 파일구조를 이용하는 이차원 색인구조를 구성한다. 이차원 색인구조에서 한 축은 색인된 엘리먼트의 킷값 도메인으로 구성하고 다른 한 축은 타입상속 계층의 타입 식별자 도메인으로 구성한다. 이와 같은 이차원 색인구조를 이용함으로써 사용자 질의 패턴에 따라 두 도메인 사이에서 색인 엔트리들의 클러스터링 정도를 조정함으로써 질의처리의 성능을 향상시킬 수 있다. 본 논문에서 제안한 2D-THI의 성능평가를 위하여, 비용 모델을 개발하고 이를 통하여 2D-THI를 기존의 객체지향 데이터베이스에서 사용하고 있는 CH-index와 CG-tree와 같은 클래스 계층 색인기법들과 색인의 성능을 비교평가 한다. 성능평가의 결과로서, CH-index와 CG-tree에서는 특정 형태의 XML 질의의 경우에만 좋은 성능을 보인 반면, 본 논문에서 제안한 2D-THI에서는 주어진 질의 형태에 따라 최적의 질의처리 성능을 제공할 수 있음을 보인다.

2D-THI: Two-Dimensional Type Hierarchy Index for XML Databases

Jong-Hak Lee†

ABSTRACT

This paper presents a two-dimensional type inheritance hierarchy index(2D-THI) for XML databases. XML Schema is one of schema models for the XML documents supporting the type inheritance. The conventional indexing techniques for XML databases can not support XML queries on type inheritance hierarchies. We construct a two-dimensional index structure using multidimensional file organizations for supporting type inheritance hierarchy in XML queries. This indexing technique deals with the problem of clustering index entries in the two-dimensional domain space that consists of a key element domain and a type identifier domain based on the user query pattern. This index enhances query performance by adjusting the degree of clustering between the two domains. For performance evaluation, we have compared our proposed 2D-THI with the conventional class hierarchy indexing techniques in object-oriented databases such as CH-index and CG-tree through the cost model. As the result of the performance evaluations, we have verified that our proposed two-dimensional type inheritance indexing technique can efficiently support the query processing in XML databases according to the query types.

Key words: XML Documents(XML 문서), XML Schema(XML 스키마), XML Index(XML 색인)

※ 교신저자(Corresponding Author) : 이종학, 주소 : 경북 경산시 하양읍 금락1리 330(712-702) 전화 : 053)850-2746, FAX : 053)850-2704, E-mail : jhlee11@cu.ac.kr

접수일 : 2005년 10월 7일, 완료일 : 2005년 12월 7일

† 정회원, 대구가톨릭대학교 컴퓨터정보통신공학부 교수

1. 서 론

XML(eXtensible Markup Language)[1]은 인터넷의 급속한 발전과 더불어 대량의 정보를 효과적으로 표현 및 교환할 수 있는 새로운 데이터 표준 언어이다. 기존의 마크업 언어인 HTML (Hyper Text Markup Language)의 차세대 언어로서 사용이 간편하고 재사용성 및 확장성이 뛰어나다는 장점을 가지고 있다. 이러한 장점으로 인해 XML은 전자상거래, 전자 민원 서비스, 데이터베이스, 웹 문서 작성, 웹 사이트 개발, 전자문서 교환, 무선 인터넷 콘텐츠, 전자 서명과 암호화와 같은 정보 보호 등 많은 분야에서 활용되고 있다.

XML 데이터베이스는 XML 문서를 저장하고 검색하기 위한 데이터베이스이다[2]. 이러한 XML 데이터베이스를 정의하기 위한 스키마 정의어로서 DTD(Data Type Definition)와 XML 스키마[3, 4]가 있다. DTD는 엘리먼트의 구조를 재사용할 수 없는 등 데이터 타입이 제한적으로 사용되는 단점을 가지고 있다. 그래서 타입상속을 지원하는 XML 스키마가 W3C(World Wide Web Consortium)에 의해서 제안되었다. XML 스키마의 타입상속에 의해 정의된 XML 데이터베이스는 각 타입이 여러 개의 서브타입을 가질 수 있으므로 하나의 타입상속 계층(class inheritance hierarchy)을 형성한다. 따라서 XML 질의어는 이러한 데이터 모델상의 특징을 감안하여 질의의 대상을 하나의 타입 또는 특정 타입을 루트로 하는 타입상속 계층으로 지정할 수 있다.

데이터베이스의 색인구조는 탐색 조건에 따라 레코드들을 빠르고 효율적인 검색을 위하여 사용하는 접근 구조이다[5]. 따라서, XML 데이터베이스의 색인구조는 하나의 타입에 속한 엘리먼트들을 대상으로 하는 탐색뿐만 아니라 타입상속 계층에 속한 엘리먼트들을 대상으로 하는 탐색도 효율적으로 지원할 수 있어야 한다. 지금까지 XML 데이터베이스에서는 엘리먼트 중첩(nested)[6]에 관한 색인기법에 대한 연구는 활발하나 XML 스키마에 의한 타입상속 색인기법에 대한 연구는 미흡한 실정이다. XML 데이터베이스의 엘리먼트 중첩에 관한 색인기법으로는 DataGuide[7], 1-Index[8], Index Fabric[9], APEX[10] 등이 있다. 이러한 색인구조는 구조 요약(structural summary)[11]이나 경로 색인(path index)[7,12-14]를 이용하여 주어진 경로 표현식에 대하여 XML 데이터베이스의 관련 있는 부분만은 검색할 수 있도록 하여 XML 데이터의 검색 속도를 향상시키는 색인기법이다.

본 논문에서는 XML 스키마에 의한 타입상속 색인기법으로 타입상속에 대한 질의를 빠르게 처리할 수 있는 효율적인 이차원 타입상속 색인기법을 제안한다. 지금까지 상속 계층에 대한 색인기법으로는 기존의 객체지향 데이터베이스의 클래스 상속에 대한 색인기법으로 CH-index[13]와 CG-tree[14]가 대표적으로 사용되고 있다. 이러한 클래스 계층 색인기법들은 색인 엔트리들의 클러스터링이 하나의 속성에 의해 이루어지는 B^+-tree[15]와 같은 일차원 색인구조를 주로 사용하고 있다[13,14,16,17]. 일차원 색인구조에서는 클러스터링 특성이 하나의 속성에 의해서 독점되기 때문에 특정 형태의 탐색 질의만 효율적으로 지원하고 다른 형태는 효율적으로 지원하지 못하는 문제점을 가지고 있다.

본 논문에서 제안하는 이차원 타입상속 색인기법은 다차원 파일구조를 사용하여 킷값 도메인과 함께 타입 식별자 도메인으로 구성된 이차원 도메인 공간상의 색인 엔트리들의 클러스터링 문제를 다룬다. 이차원 타입상속 색인기법에서는 사용자 질의 패턴에 따른 최적의 이차원 타입상속 색인구조를 구성한다. 먼저, 사전에 분석한 사용자 질의 형태에 대한 정보를 이용하여 킷값 도메인과 타입 식별자 도메인 사이의 색인 엔트리들에 대한 클러스터링 정도를 구한다. 그리고 이러한 클러스터링 정도를 유지하도록 하는 이차원 색인구조의 영역 분할전략을 적용하여 색인구조를 구성한다. 이와 같은 색인구조는 이차원의 두 도메인 사이에서 색인 엔트리들의 클러스터링 정도를 주어진 질의 패턴에 적합하도록 조정하는 색인구조이다.

본 논문의 구성은 다음과 같다. 제 2절에서는 관련 연구로서 XML 데이터베이스와 기존의 객체지향 데이터베이스의 클래스 상속 계층에 대한 색인기법들을 소개한다. 제 3절에서는 XML 데이터베이스의 타입상속 계층구조에 대한 색인기법으로 이차원 타입상속 색인기법을 제안한다. 그리고 제 4절에서는 성능 평가를 위한 비용 모델과 함께 성능 평가의 결과를 제시한다. 마지막으로 제 5절에서 결론을 기술한다.

2. 관련 연구

본 절에서는 XML 타입상속 계층구조에 대한 색인기법을 논하기 위하여 필요한 기본 개념들을 소개한다. 먼저, XML 데이터베이스의 구조적 정의를 위해서 제안된 XML 스키마와 XML 스키마 그래프에 대하여 기술한다. 그리고 XML 타입상속 색인기법으로 사용되고 있는 기존의 객체지향 데이터베이스의 클래스 상속 색인기법들에 관해서 기술한다.

XML 문서의 구조를 정의하기 위하여 제안된 초기의 XML 문서 정의어인 DTD는 엘리먼트 중첩에 관한 정의만 있을 뿐, 엘리먼트 상속에 관한 정의가 없음으로 인하여 엘리먼트의 구조를 재사용할 수 없다는 한계점을 가지고 있다. 따라서 W3C에서는 이를 보완하기 위해서 XML 스키마이라는 새로운 XML 데이터베이스 정의어 표준규약을 정의하였다. 그림 1은 루트(root) 엘리먼트를 동물로 가지는 XML 스키마 문서의 예이다. 그림 1에서 동물 엘리먼트의 타입은 동물명, 수명이라는 엘리먼트를 가지는 복합 타입인 동물타입을 가진다. 그리고 척추동물 타입은 타입상속을 나타내는 문법인 <extension base="동물타입">을 통해서 동물타입 안에 정의해 놓은 엘리먼트들을 상속 받아 동물명, 수명 엘리먼트를 포함하여 자신이 정의하는 경추개수, 요추개수 엘

```
<element name="동물" Type="동물타입"/>
<element name="척추동물" Type="척추동물타입"/>
<element name="무척추동물" Type="무척추동물타입"/>
<element name="포유류" Type="포유류타입"/>
<element name="어류" Type="어류타입"/>
<element name="파충류" Type="파충류타입"/>
  <complexType name="동물타입">
    <complexContent>
      <sequence>
        <element name="동물명" 타입="string"/>
        <element name="수명" 타입="integer"/>
      </sequence>
    </complexContent>
  </complexType>
  <complexType name="척추동물타입">
    <complexContent>
      <extension base="동물타입"/>
        <sequence>
          <element name="경추개수" 타입="string"/>
          <element name="요추개수" 타입="integer"/>
        </sequence>
      </extension>
    </complexContent>
  </complexType>
  <complexType name="무척추동물타입">
    <complexContent>
      <extension base="동물타입"/>
      </extension>
    </complexContent>
  </complexType>
```

```
<complexType name="무척추동물타입">
  <complexContent>
    <extension base="동물타입"/>
    </extension>
  </complexContent>
</complexType>

<complexType name="포유류타입">
  <complexContent>
    <extension base="척추동물타입"/>
    </extension>
  </complexContent>
</complexType>

<complexType name="어류타입">
  <complexContent>
    <extension base="척추동물타입"/>
    </extension>
  </complexContent>
</complexType>

<complexType name="파충류타입">
  <complexContent>
    <extension base="척추동물타입"/>
    </extention>
  </complexContent>
</complexType>
```

그림 1: XML 스키마 문서의 예

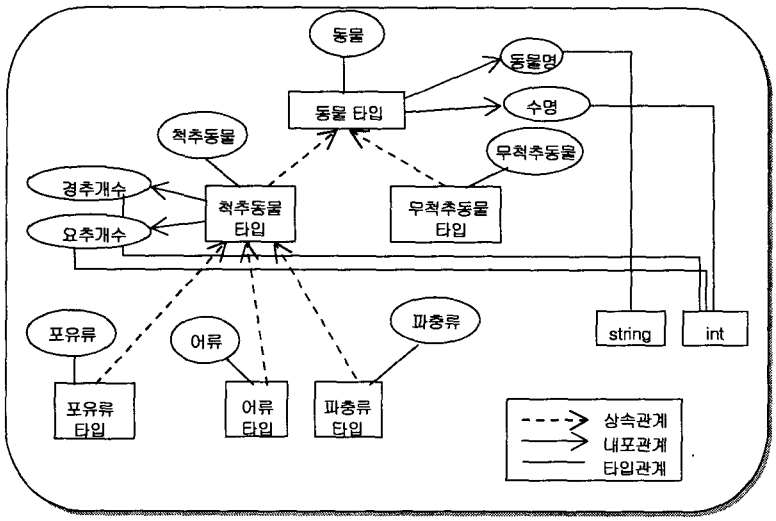


그림 2. 스키마 그래프의 예

리먼트들을 추가로 가진다.

그림 1의 XML 스키마 문서를 XML 스키마 그래프로 표현하면 그림 2와 같다. XML 스키마 그래프에서 엘리먼트는 타원으로, 타입은 사각형으로 나타낸다. 타입을 정의하는 엘리먼트들은 화살표가 있는 실선으로 나타낸다. 타입간의 상속관계는 화살표가 있는 점선으로 나타내며, 화살의 머리 쪽이 부모 타입이고 꼬리 쪽이 자식 타입이다. 그리고 엘리먼트의 타입은 화살표가 없는 실선으로 나타낸다. 그림 2에서 보면 동물 타입은 동물명, 수명이라는 엘리먼트를 가지며, 척추동물 타입과 무척추동물 타입은 모두 동물 타입으로부터 상속된 타입임을 보여준다. 또한, 포유류 타입, 어류 타입과 파충류 타입은 척추동물 타입으로부터 상속된 타입이다. 그러므로 척추동물 타입의 엘리먼트는 자신의 경추개수, 요추개수 엘리먼트와 동물 타입에서 상속받은 동물명, 수명 엘리먼트를 가지게 된다.

앞에서 살펴본 바와 같이 XML 스키마에서는 타입상속이 가능하다. 하지만 지금까지 XML 스키마의 타입상속에 대한 색인기법에 대한 연구가 미흡하다. 따라서 본 논문에서는 먼저 XML 데이터베이스의 타입상속과 개념과 유사한 기존의 객체지향 데이터베이스의 클래스 상속 계층구조에서 채택하고 있는 색인기법들로 키 클러스터링 색인구조인 CH-index[13]와 클래스 클러스터링 색인구조인 CG-tree[14]소개한다.

대부분의 객체지향 데이터베이스 시스템에서 채택하고 있는 CH-index는 클래스 상속 계층을 이루는 클래스들 각각에 개별적으로 색인을 유지하는 것이 아니라, 클래스 계층을 이루는 모든 클래스들에 대한 하나의 색인을 유지시키는 색인기법이다. 즉, B^+-tree[15]의 단말 색인 페이지에 클래스 계층내의 모든 객체에 대한 색인 엔트리를 가질 수 있도록 B^+-tree를 확장함으로서 색인 엔트리들을 각 킷값별로 클러스터링하는 색인구조이다.

그림 3은 CH-index의 구조를 나타낸다. CH-index의 비단말 페이지의 구조는 그림 3(a)와 같이 B^+-tree와 동일한 구조이지만, 단말 페이지는 그림 3(b)와 같이 B^+-tree와는 다른 색인 레코드들로 구성된다. CH-index의 단말 페이지의 가장 큰 특징은 동일한 킷값을 가지는 클래스 계층내의 모든 객체들에 대한 색인 엔트리들과 함께 이들을 클래스별로 구분해두기 위한 클래스 디렉토리라는 필드를 가진다는 점이다. 클래스 디렉토리는 색인될 속성의 킷값을 갖는 객체를 포함하는 클래스의 수가 몇 개인가를 나타내는 클래스의 개수(no. classes) 필드, 각 클래스에 대한 클래스 식별자(class-id) 필드와 상대주소(offset) 필드를 가진다. 상대주소 필드는 각 클래스별 색인 엔트리(oid)들에 대한 색인 레코드 내에서의 상대주소를 나타낸다. 만약, 색인 레코드의 크기가

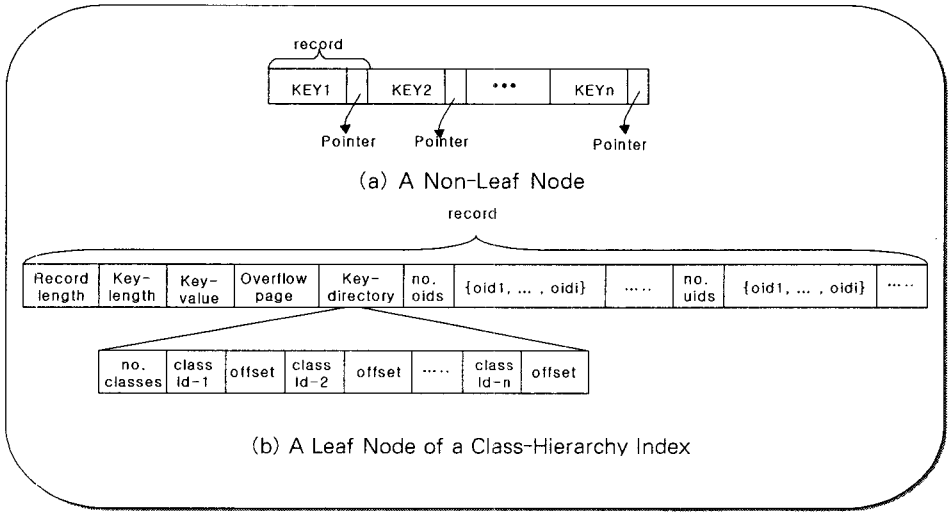


그림 3. CH-index의 비단말 페이지와 단말 페이지의 구조

색인 페이지의 크기보다 크게 되면 오버플로우 페이지(overflow page)를 할당하고 색인 레코드 안에 오버플로우 페이지 포인터 필드를 사용하여 이 페이지를 가리키게 한다.

CG-tree는 색인 엔트리들을 각 클래스별로 분리하여 별개의 단말 색인 페이지 리스트로 구성함으로서 먼저 색인 엔트리들을 클래스별로 클러스터링하는 색인구조이다. 즉, 단말 페이지는 클래스 개수만큼의 클래스별 이중 연결 리스트(doubly linked list)로 구성된다. CH-index에서 단말 페이지에 가지고 있는 클래스 디렉토리라는 필드를 포함하지 않고, 그 대신 비단말 페이지들의 최하위 계층에 디렉토리 페이지(directory page)라는 특수한 페이지들을 가진다.

그림 4는 CG-tree의 클래스 디렉토리 페이지의 구조를 나타낸다. 디렉토리 페이지를 구성하는 각 디렉토리 레코드에는 각 클래스별로 단말 색인 페이지를 가리키는 포인터들을 가지고 있다. 그림 4에서 클래스 디렉토리 레코드 R_i의 j번째 엔트리($R_i.c_j$로 표기)는 이중 연결리스트로 구성된 클래스 C_j의 단말 색인 페이지들 중에서 킷값이 $[K_i, K_{i+1})$에 속하는 객체에 대한 색인 엔트리들을 포함하고 있는 최초의 단말 색인 페이지를 가리키는 포인터이다.

객체지향 데이터베이스에서의 질의는 주어진 하나의 클래스에 속하는 객체 인스턴스들에 대한 질의 혹은 클래스 계층 내에서 어느 클래스에 속하는 객체 인스턴스들에 대한 질의로 다른 클래스 범위로 주어질 수 있다. 그러나 앞에서 살펴본 두 색인구조의 성능을 살펴보면 CH-index에서는 하나의 클래스를 대상으로 킷값에 대한 범위 질의(range queries)를 처리하는 경우에 색인 성능이 비효율적으로 되며, CG-tree에서는 같은 클래스 계층을 대상으로 하는 부합질의(match queries) 경우에 색인 성능이 비효율적이 된다.

이와 같은 기존의 객체지향 데이터베이스의 클래스 상속계층 색인기법들을 XML 데이터베이스의 타입상속 색인기법으로 사용할 경우에도 B$^+$-tree와 같은 일차원 색인구조를 이용함으로써 특정 형태의 질의에 대해서만 효율적인 성능을 나타내며, 그 외의 질의 형태들에 대해서는 비효율적으로 된다. 따라서

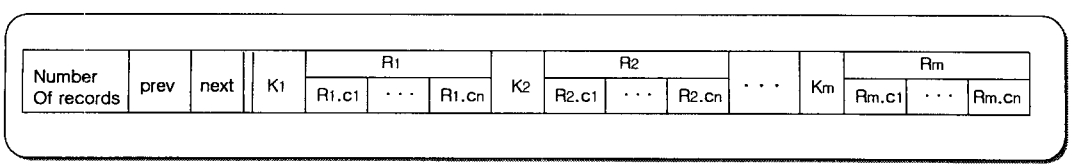


그림 4. CG-tree의 디렉토리 페이지 구조

본 논문에서는 이러한 문제점을 해결하기 위하여 이차원 색인구조를 이용하여 주어지는 질의 유형에 따라 동적으로 색인구조를 구성할 수 있는 XML 데이터베이스의 타입상속 계층구조에 대한 효율적인 색인구조를 제안한다.

3. XML 데이터베이스의 타입상속 색인기법

본 절에서는 XML 데이터베이스의 타입상속 계층구조에 대한 이차원 타입상속 색인구조를 제안한다. 제 3.1절에서는 킷값 도메인과 타입 식별자 도메인으로 구성된 이차원 타입상속 색인구조를 소개하고, 제 3.2절에서는 이차원 타입상속 색인구조의 특징으로 최적의 클러스터링 조건을 소개한다. 제 3.3절과 제 3.4절에서는 다차원 화일구조의 하나인 계층 그리드 화일[18,19]을 이용한 이차원 타입상속 색인구조의 구성 방법과 조작 알고리즘을 기술한다.

3.1 이차원 타입상속 색인구조

XML 스키마의 타입상속 계층에 대한 색인구조로 다차원 파일구조에서 두 개의 차원을 이용하여 한 도메인은 색인될 엘리먼트의 킷값으로, 다른 한 도메인은 타입상속 계층구조에 포함된 타입들의 식별자들로서 이차원 색인구조를 구성하고, 앞으로 이를 2D-THI (2-Dimensional Type Hierarchy Index)라 한다.

XML 데이터베이스에서의 타입상속 계층에 대한 색인기법을 논하기 위하여 XML 스키마를 따르는 XML 데이터베이스의 타입상속 계층구조에 대한 질의를 다음과 같은 세 가지 형태로 분류한다. 여기서, 타입 T^*는 임의의 타입 T와 그의 모든 서브 타입들을 원소로 하는 집합이다. 예를 들어 동물*는 집합{동물, 척추동물, 포유류, 어류, 파충류, 무척추동물}이며, 척추동물*는 집합{포유류, 어류, 파충류}이다.

1. STR(Single Type Range) 형태의 질의: 특정한 하나의 타입 T를 대상으로 하는 범위 질의

2. THM(Type Hierarchy Match) 형태의 질의: 특정 타입 T의 타입 집합 T^*를 대상으로 하는 부합 질의

3. THR(Type Hierarchy Range) 형태의 질의: 특정 타입 T의 타입 집합 T^*를 대상으로 하는 범위 질의

Robinson[20]에 의하면, 사용자가 요구하는 모든 질의는 도메인 공간내의 영역들로 표현할 수 있고, 이 영역은 도메인을 구성하는 축에 대한 구간들의 곱으로 표현되며, 이러한 영역을 질의 영역(query region)이라고 한다. 그 정의에 의하면 타입상속 계층을 대상으로 하나의 키 엘리먼트에 대한 질의 조건으로 주어지는 질의는 킷값 도메인과 타입 식별자 도메인으로 구성된 이차원 도메인 공간상의 이차원 질의 영역으로 매핑할 수 있다.

2D-THI에서 타입 식별자 도메인은 색인을 구성할 타입상속 계층상의 타입 식별자들이 루트로부터 깊이우선 탐색(depth first search) 순서로 나열되도록 구성한다. 이렇게 구성함으로써 임의의 타입 T에 대해서 그의 서브 타입들로 구성된 타입 집합 T^*에 포함된 모든 타입 식별자들이 타입 식별자 도메인 상에서 연속된 구간이 되도록 한다.

표 1은 이러한 타입 식별자 도메인 구성의 예를 보여준다. 표 1은 그림 2의 XML 스키마 그래프에서 타입 식별자들을 깊이우선 탐색 순서로 타입 식별자 도메인을 구성한 것이다. 표 1에서 동물*의 범위는 동물의 도메인 값 0을 비롯하여 모든 타입들을 포함하게 되므로 0~5가 되고, 척추동물*의 범위는 1~4가 된다. 이처럼 모든 타입 T^*에 포함된 타입 식별자들은 표 1처럼 타입 식별자 도메인 상에 연속된 구간을 가지게 됨을 알 수 있다.

표 1. 타입 식별자 도메인 구성의 예

타입식별자	동물	척추동물	포유류	어류	파충류	무척추동물
도메인 값	0	1	2	3	4	5
T^*의 범위	0~5	1~4	2~2	3~3	4~4	5 ~ 5

다차원 도메인 공간상에서 한 영역의 형태는 도메인을 구성하는 각 축에 대한 구간비로 표현할 수 있다. 따라서 다차원 색인구조에서는 사용자 질의 유형에 나타나는 질의 영역들의 형태에 대한 정보를 기반으로 질의 영역들에 의해 교차하는 페이지 영역들의 개수가 최소로 되는 *페이지 영역의 최적 구간비*를 결정하고, 색인 페이지의 구조가 이와 같은 최적의 구간비를 갖도록 하는 페이지 영역분할 전략을 사용함으로써 최적의 다차원 색인구조를 구성할 수 있다

[12]. 이와 같이 이차원 타입상속 색인기법에서도 사용자 질의 유형에 대한 정보를 기반으로 최적의 이차원 타입 색인구조를 구성할 수 있다.

3.2 최적의 2D-THI

다차원 색인구조에서는 다차원 도메인 공간을 구성하는 페이지 영역의 모양(구간비)에 따라 질의 영역에 의해서 교차되는 페이지 영역의 개수가 달라지는 특징이 있다. 즉, 주어진 질의 영역의 모양과 도메인 공간의 분할 상태를 나타내는 페이지 영역의 모양이 같아질수록 액세스해야할 페이지의 개수가 적게 된다. 참고문헌[12]에서는 이러한 특징을 이용하여 데이터의 균일 분포와 비균일 분포 각각에 대하여 주어진 질의 영역들에 대해 페이지 영역의 평균 액세스 횟수를 최소로 하는 페이지 영역의 최적 구간비를 계산하는 방법을 제안하였다. XML 스키마의 타입상속 계층구조에 대한 이차원 색인구조인 2D-THI에서는 이와 같은 방법을 이용하여 페이지 영역의 최적 구간비를 계산한다.

색인구조를 구성하는 도메인공간상에서 데이터가 균일하게 분포할 경우에는 도메인을 구성하는 페이지 영역들의 크기가 일정하게 되며, 주어진 질의 영역들에 의해 교차되는 페이지 영역들의 개수를 최소로 하는 페이지 영역의 최적 구간비는 모든 질의 영역들에 대해 각 축별로 구간 크기를 더한 값의 비로서 계산할 수 있다[12].

하지만 도메인공간상에서 데이터가 비균일하게 분포할 경우에는 도메인 공간의 위치에 따라 색인 엔트리의 밀집도가 다르기 때문에 페이지 영역의 크기가 위치에 따라 달라진다. 즉, 밀집도가 높은 곳에서는 밀집도가 낮은 곳에 비하여 많은 페이지가 할당되므로 각 페이지 영역의 크기는 작아지게 된다. 따라서 비균일 분포의 경우에는 질의 영역에 의해 교차되는 페이지 영역의 개수는 질의 영역의 크기뿐만 아니라 질의 영역이 주어진 위치의 데이터 밀집도에도 비례하게 되므로 밀집도를 고려해 주어야 한다. 각 질의 영역의 크기에 대해서 위치에 따른 데이터 밀집도를 가중치(weight)로 곱한 질의 영역의 형태를 정규화된 질의 영역(normalized query region)이라 하고, 이러한 질의 영역의 정규화를 통해서 페이지 영역의 최적 구간비를 계산한다. 표 2는 이와 같은 결과를 주어진 n개의 질의 영역 $q_i(x)$ X $q_i(y)(i=1, \ldots, n)$에 대해서 데이터 분포에 따른 페이지 영역의 최적 구간비를 표로 나타낸 것이다.

표 2. 데이터의 분포에 따른 페이지 영역의 최적 구간비

데이터의 분포	페이지 영역의 최적 구간비(p(x):p(y))
균일 분포	$\sum_{i=1}^{n} q_i(x) : \sum_{i=1}^{n} q_i(y)$
비균일 분포	$\sum_{i=1}^{n} q_i(x)\sqrt{d_i} : \sum_{i=1}^{n} q_i(y)\sqrt{d_i}$ (d_i : 데이터 밀집도)

따라서, 본 논문에서는 표 2에서 정의된 최적 구간비를 이용하여 최적의 2D-THI를 구성한다. 즉, 사용자 질의 유형으로 주어지는 다양한 XML 질의들에 의해 이차원 도메인 공간상에 표현되는 질의 영역들을 이용하여 페이지 영역의 최적 구간비를 구하고, 이와 같은 구간비를 가지는 페이지 영역들로서 2D-THI를 구성한다.

3.3 2D-THI의 구성

본 논문에서는 다차원 색인구조의 하나인 계층 그리드 파일[18]을 이용하여 XML 스키마의 타입상속 계층구조에 대한 2D-THI를 구성한다. 2D-THI는 디렉토리와 색인 페이지로 구성된다. 디렉토리는 다단계의 균형된 트리 구조를 가지며 디렉토리 페이지의 구조는 그림 5(a)와 같다. 디렉토리의 최하위 단계에 있는 디렉토리 레코드는 색인 페이지를 가리키는 포인터를 가지며, 그 색인 페이지가 할당된 페이지 영역을 리젼 벡터(region vector)를 이용하여 표현한다. 리젼 벡터는 해당 페이지 영역의 킷값 도메인과 클래스 식별자 도메인에 대한 구간을 해쉬값으로 표현한 것이다. 색인 페이지는 디렉토리 레코드에 의해서 표현된 영역내에 속하는 색인 레코드만을 저장한다. 그리고 다단계의 디렉토리 구조는 재귀적으로 구성된다. 즉, 상위 단계의 디렉토리 레코드는 차 하위 단계의 디렉토리 페이지를 가리키는 포인터를 가지며, 그 디렉토리 페이지가 가리키는 영역을 표현한다.

2D-THI의 색인 페이지의 구조는 그림 5(b)와 같다. 색인 페이지의 각 색인 레코드에는 타입 식별자 값(type-id value) 필드, 킷값(key-value) 필드, 이 두 값을 갖는 엘리먼트의 개수(No. Eids) 필드와 이들

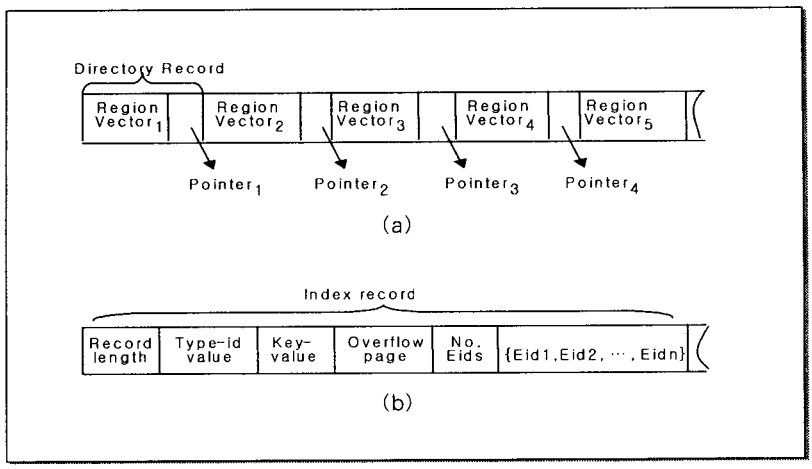


그림 5. 2D-THI의 디렉토리와 색인 페이지의 구조 : (a) 타입 상속 색인구조 디렉토리 페이지의 구조, (b) 타입 상속 색인구조 색인 페이지의 구조

엘리먼트에 대한 색인 엔트리(Eids)들의 리스트 필드가 있다. 그리고 레코드의 크기가 페이지의 크기보다 크게 될 경우에 할당하게 되는 오버플로우 페이지(overflow page)를 가리키는 포인터가 있다.

3.4 2D-THI의 조작 알고리즘

이차원 타입상속 색인구조인 2D-THI의 삽입, 삭제 및 검색을 위한 조작 연산의 알고리즘은 계층 그리드 파일에서와 동일하며 단지 삽입 연산에서 색인 페이지의 용량이 오버플로우되면 페이지 영역을 분할하는 영역분할 전략만 다르다. 따라서 본 절에서는 페이지 영역의 구간비가 3.3절에서 정의한 최적 구간비에 근접하도록 하는 영역분할 전략을 제시한다. 2D-THI는 색인 엔트리가 삽입되고 삭제되는 상황에 따라서 색인 페이지가 분할과 병합을 반복 수행함으로써 동적 변화에 적응한다[18]. 새로운 엔트리가 삽입되는 경우에는 다단계의 디렉토리를 루트로부터 최하위 디렉토리까지 탐색하여 그 색인 엔트리가 속하는 페이지 영역을 찾게 되고, 그 영역에 할당된 색인 페이지에 색인 엔트리를 삽입하게 된다. 이러한 결과로 색인 페이지의 용량이 오버플로우 되면, 해당 영역은 같은 크기를 갖는 두 개의 영역으로 분할되고 새로운 색인 페이지가 하나 더 할당되어서 기존의 색인 페이지에 있던 레코드들은 두 색인 페이지에 분산된다.

그림 6은 각 레코드가 하나의 색인 엔트리를 가지며 색인 페이지의 용량이 3이라고 가정했을 때, 계층 그리드 화일의 영역분할 과정을 보여주고 있다. 그림 6의 (a)는 전체 도메인 공간상에 3개의 레코드가 존재하는 초기 상태를 나타낸다. 여기에 새로운 색인 엔트리의 삽입으로 레코드를 하나 더 생성하면 색인 페이지의 용량이 초과되므로 전체 도메인 공간은 두 개의 영역으로 분할되고, 새로운 색인 페이지가 할당된다. 그리고 레코드들은 분할 경계 값에 따라 두 개의 색인 페이지에 분산된다(그림 6(b)). 그림 6의 (c)와 (d)는 연속된 색인 엔트리의 삽입으로 페이지 영역의 연속된 분할 상태를 나타낸다. 이와 반대로 색인 엔트리들의 삭제 시에는 분할 과정의 역순으로 병합하게 된다. 계층 그리드 파일에서는 그림 6과 같이 두 축을 번갈아가면서 분할시키는 전략을 가정하고 있다. 그러나 2D-THI에서는 페이지 영역을 분할할 때 분할 축을 임의로 선택할 수 있으며, 분할 축을 선택하는 방법에 따라 페이지 영역의 모양을 결정할 수 있다. 따라서 본 논문에서는 분할되는 페이지 영역의 분할 축으로 분할된 영역의 구간비가 페이지 영역의 최적 구간비 O에 가깝게 되는 축을 선택함으로써, 색인 레코드의 지속적인 삽입으로 인한 연속된 분할 시에 도메인 공간내의 모든 페이지 영역들의 구간비를 O에 가깝도록 한다.

X축과 Y축으로 구성된 2D-THI의 도메인공간에서 색인 페이지의 용량초과로 분할이 요구된 페이지 영역의 크기를 p(x) X p(y)라고 할 때, X축을 분할한 경우 분할된 한쪽 페이지 영역의 구간비는 p(y)/(p(x)/2)이고, Y축을 분할한 경우는(p(y)/2)/ p(x)가

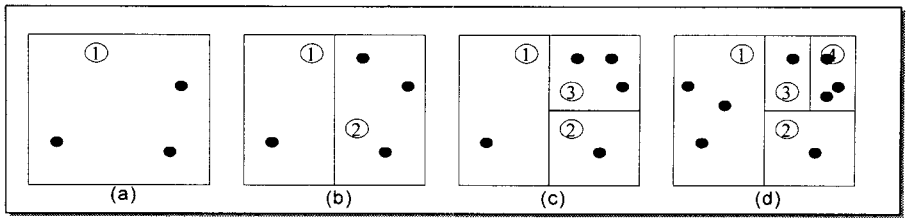


그림 6. 2D-THI의 영역분할

된다. 따라서 이 두 가지 경우에 대해서 분할된 페이지 영역의 구간비가 최적 구간비 O에 더 가깝게 되는 경우를 택하면 된다. 즉, 분할될 페이지 영역의 구간비 p(y)/p(x)가 최적 구간비 O보다 작으면 X축을 분할하고 그렇지 않으면 Y축을 분할한다.

4. 성능평가

본 절에서는 기존의 객체지향 데이터베이스에서 널리 사용되고 있는 클래스 상속 색인구조인 CH-index와 CG-tree를 XML 데이터베이스에 적용한 경우와 본 논문에서 제안한 이차원 타입상속 색인구조인 2D-THI의 성능을 비교 분석한다. 먼저 성능평가 모델을 기술한 다음 성능평가의 결과를 기술한다.

4.1 성능평가 모델

본 논문에서는 데이터베이스의 액세스 구조를 위한 여러 분석 모델에 의한 연구에서 일반적으로 사용되고 있는 가정들을 다음과 같이 사용한다.

(a) 모든 킷값들은 같은 길이를 가진다. 이 가정으로 각 색인구조에서 비단말 레코드들의 길이를 동일하게 한다.

(b) 각 킷값에 대해서 그 값을 가지는 엘리먼트가 있는 타입들의 개수는 일정하다. 이 가정으로 CH-index의 각 단말 색인 페이지의 색인 레코드에서 타입 개수 필드의 값을 모두 일정하게 하며, CG-tree의 타입 디렉토리 레코드의 길이를 모두 일정하게 한다.

(c) 킷값들은 타입상속 계층 내에서 균일 분포한다. 이것은 2D-THI에서 색인 페이지를 구성하는 각 색인 레코드들의 길이를 모두 일정하게 하고, 가정 (a)와 함께 사용하여 CG-tree의 각 타입별 단말 색인 페이지의 색인 레코드의 길이를 모두 일정하게 한다. 그리고 가정 (a), (b)와 함께 사용하여 CH-index에서 단말 색인 페이지를 구성하는 각 색인 레코드들의 길이를 모두 일정하게 한다.

다음 표 3은 비용 모델에서 데이터베이스의 특성을 반영하기 위하여 사용된 매개변수들이고, 표 4는 각 색인구조의 특성을 반영하기 위하여 사용된 매개변수들이다.

표 3. 데이터베이스 특성과 관련된 매개변수

매개변수	의 미
N	타입상속 계층 전체의 엘리먼트 수
D_i	타입 T_i에서 색인된 속성이 가지는 서로 다른 킷값의 개수
N_i	타입 T_i의 엘리먼트 수
E_i	타입 T_i에서 같은 킷값을 가지는 엘리먼트들의 평균 개수(N_i/D_i)
nt	타입상속 계층을 이루는 타입의 개수

표 4. 색인구조의 특성과 관련된 매개변수

매개변수	의 미
HRL	CH-index 색인 레코드의 평균 길이
GRL	CG-tree 색인 레코드의 평균 길이
GDL	CG-tree 클래스 디렉토리 레코드의 길이
DRL	2D-THI 색인 레코드의 평균 길이
RL	CH-index와 CG-tree 비단말 레코드의 길이
DDL	2D-THI 디렉토리 레코드의 길이
T	각 킷값을 갖는 엘리먼트가 있는 타입의 평균 개수
EN	CH-index 또는 CG-tree의 비단말 페이지가 가지는 레코드의 평균 개수
EDN	2D-THI의 디렉토리 페이지가 가지는 레코드의 평균 개수

비용 모델에 의한 각 색인구조의 성능평가를 위하여 실험 모델을 다음과 같이 설정한다. 타입상속 계층구조는 전체 타입의 개수로서 31개의 타입(T_1, T_2,

..., T_{31})으로 구성된 균형된 이진트리 형태를 사용한다. 각 타입은 3,000개의 엘리먼트로 구성하여 총 엘리먼트 N은 31 X 3,000 = 93,000개이다. 타입 T_i (i = 1, 2, ···, 31)에서 동일한 킷값을 갖는 엘리먼트들의 평균 개수인 K_i의 값으로 1, 2, 5, 10을 사용한다. K_i의 값은 각 색인구조의 단말 색인 페이지를 구성하는 색인 레코드의 크기를 결정함으로써 단말 색인 페이지의 블로킹 인수를 결정한다. 그리고 각 타입 T_i에 있는 엘리먼트가 3,000개로 일정하므로 각 타입이 가지는 서로 다른 킷값의 개수 D_i는 3,000, 1,500 900, 300으로 한정된다. 또한 성능 비교를 위해 사용된 질의유형으로는 STR 형태의 질의, THM 형태의 질의, 그리고 THR 형태의 질의만으로 주어지는 각각의 경우와 여러 질의 형태들이 혼합되어 주어지는 경우를 사용한다.

4.2 성능평가 결과

본 절에서는 성능평가의 결과를 기술한다. 제 4.1절의 성능평가 모델에서 가정한 것처럼 타입상속 계층의 모든 타입에서 색인 된 엘리먼트의 값들이 킷값 도메인 상에서 균일하게 분포할 경우 각 색인구조에 대하여 STR 형태의 질의 유형, THM 형태의 질의 유형, THR 형태의 질의 유형, 그리고 혼합 형태의 질의 유형이 주어질 경우에 대해서 탐색 비용을 모델링하고 이 비용 모델을 통해서 성능을 비교 분석한다. 다음 표 5는 본 논문에서 질의 처리를 위한 색인탐색의 비용 C를 모델링 하는데 사용된 매개변수들이다.

표 5. 색인탐색의 비용을 모델링 하는데 사용된 매개변수

매개변수	의 미
H	단말 색인 페이지를 제외한 색인구조의 높이
HB	CH-index의 단말 색인 페이지의 블로킹 인수
GB	CG-tree의 단말 색인 페이지의 블로킹 인수
TB	2D-THI의 단말 색인 페이지의 블로킹 인수
RKN	범위 질의에 나타나는 킷값의 개수
TQT	타입상속 계층에 대한 질의에서 질의의 대상이 되는 타입의 개수
DKN	하나의 단말 색인 페이지에 포함된 서로 다른 킷값의 개수
TN	하나의 단말 색인 페이지에 포함된 타입 식별자의 개수

(1) STR 형태의 질의 유형이 주어지는 경우

STR 형태의 질의 유형이 주어는 경우에 2D-THI에서는 타입 식별자 도메인을 이루는 X축에 대해서만 계속해서 분할하게 되며, X축에 대하여 더 이상 분할할 수 없을 때 킷값 도메인을 이루는 Y축에 대하여 분할하게 된다. 따라서 하나의 단말 색인 페이지에는 X축에 대해 하나의 타입 식별자 값만이 할당되고, Y축에 대해서는 블로킹 인수 TB개만큼의 킷값이 할당되어 분할된 페이지 영역의 구간비가 1:TB로 된다. 즉, 이 경우 2D-THI는 단말 색인 페이지들을 타입상속 계층을 이루는 각 타입별로 분리 저장하여 구성되는 CG-tree와 같은 색인구조가 된다.

모든 색인구조에서 각 색인 레코드에는 하나의 킷값이 할당되므로 단말 색인 페이지에 할당되는 서로 다른 킷값의 개수 DKN은 각 색인구조의 블로킹 인수와 같게 된다. 그리고 단말 색인 페이지에 오버플로우가 발생하면 DKN의 값은 1이 된다. 따라서 STR 형태의 질의에서 색인탐색의 비용 C는 각 색인구조의 높이 h와 질의 범위에 주어진 킷값의 개수 RKN를 DKN로 나눈 값에 1을 더한 값이 된다. 그리고 단말 색인 페이지에 오버플로우가 발생했을 경우(DKN=1)에는 색인 레코드의 길이를 X라 할 때 하나의 킷값에 대해 $\lceil X/PS \rceil$가 된다. 그러므로 각 색인구조에 대해 색인탐색의 비용 C는 다음과 같이 계산할 수 있다.

<CH-index의 경우>

$$DKN = \begin{cases} HB & (HB \geq 1\text{인 경우}) \\ 1 & (\text{그 외의 경우}) \end{cases}$$

$$C = \begin{cases} h + RKN/DKN + 1 & (DKN > 1\text{인 경우}) \\ h + RKN \times HRL/PS & (DKN = 1\text{인 경우}) \end{cases}$$

<CG-tree의 경우>

$$DKN = \begin{cases} GB & (GB \geq 1\text{인 경우}) \\ 1 & (\text{그 외의 경우}) \end{cases}$$

$$C = \begin{cases} h + RKN/DKN + 1 & (DKN > 1\text{인 경우}) \\ h + RKN \times GRL/PS & (DKN = 1\text{인 경우}) \end{cases}$$

<2D-THI의 경우>

$$DKN = \begin{cases} TB & (TB \geq 1\text{인 경우}) \\ 1 & (\text{그 외의 경우}) \end{cases}$$

$$C = \begin{cases} h + RKN/UK + 1 & (DKN > 1\text{인 경우}) \\ h + RKN \times DRL/PS & (DKN = 1\text{인 경우}) \end{cases}$$

그림 7은 타입 T_i에서 같은 킷값을 가지는 엘리먼트들의 개수가 두 개일 때, 각 색인구조별 질의에 나타나는 킷값 범위의 크기에 따라 액세스해야 할 색인 페이지의 수를 그래프로 나타낸 것이다. CH-index에

서는 한 색인 레코드에 타입상속 계층의 모든 타입에서 같은 킷값을 가지는 색인 엔트리들을 함께 저장하기 때문에 STR 형태의 질의 유형과 같이 하나의 특정 타입 T를 대상으로 하는 범위 질의가 주어질 경우에 액세스해야 할 색인 페이지의 액세스의 수는 킷값 범위에 나타나는 킷값의 개수에 따라 급격하게 증가한다. 반면에 CG-tree는 단말 색인 페이지들을 타입상속 계층을 이루는 각 타입별로 분리하여 저장하므로 STR 형태의 질의 유형이 주어지더라도 단말 색인 페이지에 포함된 서로 다른 킷값의 개수 DKN이 매우 크게 되어 액세스해야 할 색인 페이지의 수는 킷값 범위에 나타나는 킷값의 개수에 따라 매우 완만하게 증가한다. 한편, 본 논문에서 제안한 2D-THI에서는 영역분할 전략에 의해서 CG-tree와 같은 구조가 되어 그림 7에서 보는 바와 같이 킷값 범위의 크기에 따라 완만하게 증가 한다.

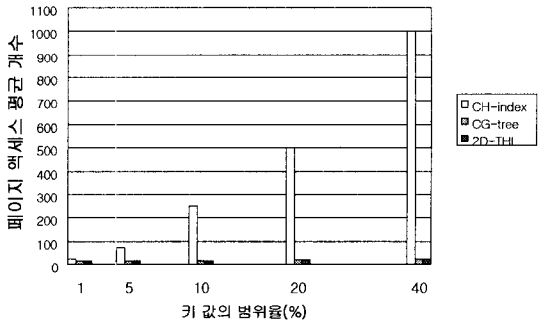


그림 7. STR 형태의 질의 유형에 대한 성능 비교

(2) THM 형태의 질의 유형이 주어지는 경우

THM 형태의 질의 유형이 주어는 경우, 2D-THI에서는 색인 페이지의 영역분할 전략에 의해 킷값 도메인을 이루는 Y축에 대해서만 계속해서 분할하게 되며, 더 이상 분할할 수 없을 때 타입 식별자 도메인을 이루는 X축에 대하여 분할하게 된다. 따라서 하나의 단말 색인 페이지에는 Y축에 대해 하나의 킷값만이 할당되고, X축에 대해서는 블로킹 인수 TB개 만큼의 타입 식별자 값이 할당되어 분할된 페이지 영역의 구간비가 TB:1로 된다. 즉, 이 경우 2D-THI는 한 색인 레코드에 타입상속 계층의 모든 타입에서 같은 킷값을 가지는 색인 엔트리를 함께 저장하는 CH-index와 같은 색인구조가 된다. 그러므로 이 색인구조를 이용한 질의 처리의 경우 색인탐색 비용 C는 질의의 대상이 되는 타입의 개수에 상관없이 일정하게 된다. 각 색인구조의 비용 모델은 다음과 같다.

<CH-index의 경우>

$$C = \begin{cases} h+1 & (HB \geq 1\text{인 경우}) \\ h + HRL/PS & (\text{그 외의 경우}) \end{cases}$$

<CG-tree의 경우>

$$C = \begin{cases} h + TQT & (GB \geq 1\text{인 경우}) \\ h + TQT \times GRL/PS & (\text{그 외의 경우}) \end{cases}$$

<2D-THI의 경우>

$$C = \begin{cases} h + TQT/TB & (TB \geq 1\text{인 경우}) \\ h + TQT \times DRL/PS & (\text{그 외의 경우}) \end{cases}$$

그림 8은 THM 형태의 질의 유형이 주어지는 경우, 각 색인구조별 질의에 나타나는 타입의 개수에 따라 액세스해야 할 색인 페이지의 수를 그래프로 나타낸 것이다. CG-tree의 경우 각 단말 색인 페이지에는 하나의 타입에 속하는 색인 엔트리들만 저장되어 있으므로, THM 형태의 질의 유형과 같이 특정 타입 T의 모든 타입상속 계층을 대상으로 하는 부합 질의가 주어지면 액세스할 색인 페이지의 개수는 타입의 개수에 비례하게 되어 그림 8에서와 같이 급격하게 증가 한다. 반면에 CH-index에서는 타입의 개수에 상관없이 거의 일정하게 나타나며 2D-THI에서도 영역분할 전략에 의해서 CH-index와 같은 구조가 되어 거의 일정하나 이차원으로 구성된 디렉토리 엔트리의 크기로 인하여 질의의 대상이 되는 타입의 개수가 많이짐에 따라 액세스해야 할 색인 페이지의 개수가 약간 증가함을 보인다.

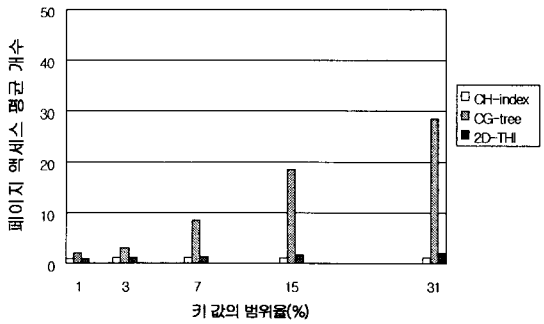


그림 8. THM 형태의 질의 유형에 대한 성능 비교

(3) THR 형태의 질의 유형이 주어지는 경우

THR 형태의 질의 유형이 주어지는 경우에 2D-THI에서는 색인 페이지의 영역분할 전략에 의해서 분할될 페이지 영역의 구간비 p(y)/p(x)가 페이

지 영역의 최적 구간비 O보다 작으면 X축을 분할하고, 그렇지 않으면 Y축을 분할함으로써 페이지 영역들의 구간비가 *TQT:RKN*에 근접한다. 각 색인구조의 색인탐색 비용 *C*는 다음과 같다.

< CH-index의 경우 >

$$DKN = \begin{cases} HB & (HB \geq 1\text{인 경우}) \\ 1 & (\text{그 외의 경우}) \end{cases}$$

$$C = \begin{cases} h + RKN/DKN & (DKN > 1\text{인 경우}) \\ h + RKN \times HRL/PS & (DKN = 1\text{인 경우}) \end{cases}$$

<CG-tree의 경우>

$$DKN = \begin{cases} GB & (GB \geq 1\text{인 경우}) \\ 1 & (\text{그 외의 경우}) \end{cases}$$

$$C = \begin{cases} h + (RKN/DKN + 1) \times TQT & (DKN > 1\text{인 경우}) \\ h + RKN \times GRL/PS \times TQT & (DKN = 1\text{인 경우}) \end{cases}$$

<2D-THI의 경우>

$$DKN = \begin{cases} \sqrt{\frac{RKN}{TQT} \times TB} & (TB \geq 1,\ TB > \frac{RKN}{TQT},\ TB > \frac{TQT}{RKN}\text{인 경우}) \\ TB & (TB \geq 1,\ TB \leq \frac{RKN}{TQT}\text{인 경우}) \\ 1 & (\text{그 외의 경우}) \end{cases}$$

$$TN = \begin{cases} \sqrt{\frac{TQT}{RKN} \times TB} & (TB \geq 1,\ TB > \frac{TQT}{RKN},\ TB > \frac{RKN}{TQT}\text{인 경우}) \\ TB & (TB \geq 1,\ TB \leq \frac{RKN}{TQT}\text{인 경우}) \\ 1 & (\text{그 외의 경우}) \end{cases}$$

$$C = \begin{cases} h + (\frac{RKN}{DKN} + 1)(\frac{TQT}{TN} + 1) & (DKN > 1,\ TN > 1\text{인 경우}) \\ h + (\frac{RKN}{DKN} + 1) \times TQT & (DKN > 1,\ TN = 1\text{인 경우}) \\ h + RKN \times (\frac{TQT}{TN} + 1) & (DKN = 1,\ TN > 1\text{인 경우}) \\ h + RKN \times TQT \times DRL/PS & (DKN = 1,\ TN = 1\text{인 경우}) \end{cases}$$

그림 9는 THR 형태의 질의 유형이 주어지는 경우, 각 질의를 만족하는 킷값의 범위가 전체 범위의 10%인 경우에 대해서, 각 색인구조별 질의에 나타나는 타입의 개수에 따라 액세스해야 할 색인 페이지의 수를 그래프로 나타낸 것이다. 그림 9에서 보는 바와 같이 CH-index에서는 질의의 대상이 되는 타입의 개수에 상관없이 질의에 주어지는 킷값의 범위에 따라 일정한 값을 가지는 반면에, CG-tree에서는 질의의 대상이 되는 타입의 개수에 비례하고, 그 비례의 정도가 킷값의 범위가 커짐에 따라 증가한다. 2D-THI에서는 질의의 대상이 되는 타입의 개수와 킷값의 범위에 따라 영역분할 전략에 의해서 최적의 색인구조를 구성함으로써 다른 두 가지 색인구조에 비하여 항상 성능이 좋게 된다.

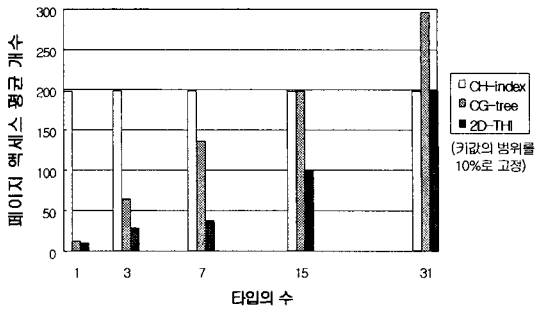


그림 9. THR 형태의 질의 유형에 대한 성능 비교

(4) 혼합 질의 유형이 주어지는 경우

STR, THM, THR 형태의 질의가 혼합되어 주어지는 경우 질의 영역의 크기를 각각 $RKN_i \times TQT_i(i=1, \cdots, n)$이라 하면, 2D-THI에서는 페이지 영역의 구간비가 모든 질의 영역의 각 축별로 구간 크기를 단순히 더한 값의 비가 되는 최적 구간비인 $\sum_{i=1}^{n} RKN_i : \sum_{i=1}^{n} TQT_i$에 근접하게 된다. 이 혼합 질의 유형의 경우 색인탐색의 비용모델은 THR 형태의 질의 유형이 주어지는 경우와 동일하다.

그림 10은 혼합 질의 유형이 주어지는 경우에 각 색인구조별 액세스되는 색인 페이지의 개수를 나타낸다. 실험에 사용한 질의 유형은 타입 식별자 도메인의 구간 크기로 1, 2, 4, 8, 16, 32인 여섯 가지 각각에 대해서, 킷값 도메인의 구간 크기로 전체 킷값 개수의 10%이내의 범위에서 임의의 여섯 가지를 선택하여 총 36개의 질의 형태들로 구성하였다. 그림 10의 혼합 질의 유형 1은 중복된 킷값의 개수가 20개인 경우로서 킷값 도메인의 구간 크기가 1, 2, 3, 6, 12, 25의 여섯 가지로 주어진 경우이며, 페이지 영역의 최적 구성비가 1.29로 계산된다. 그리고 혼합 질의 유형 2는 중복된 킷값의 개수가 세 개인 경우로서 킷값 도메인의 구간 크기가 1, 6, 12, 24, 62, 125의 여섯 가지로 주어진 경우이고 페이지 영역의 최적 구간비가 0.27로 계산된다. 그림 10에서 알 수 있듯이 두 가지 혼합 질의 유형 모두에 대해서 최적 구간비의 페이지 영역으로 구성된 2D-THI는 최적의 성능을 보인다. 그리고 최적 구간비가 1보다 크게 되는 혼합 질의 유형 1에서는 CH-index가 CG-tree보다 더 좋은 성능을 보이며, 최적 구간비가 1보다 작게 되는 혼합 질의 유형 2에서는 그 반대임을 잘 나타내고 있다.

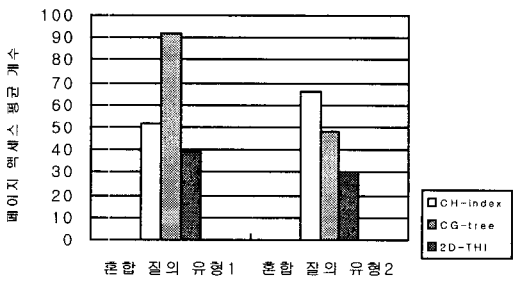


그림 10. 혼합 질의 유형에 대한 성능 비교

5. 결 론

본 논문에서는 XML 스키마에 의해 구조가 정의된 XML 데이터베이스의 타입상속 구조에 대한 색인 기법으로 다차원 색인구조를 이용하는 이차원의 타입상속 색인구조인 2D-THI를 제안하였다. 2D-THI는 키 도메인과 타입 식별자 도메인으로 이루어진 이차원 도메인 공간으로 색인구조를 구성한다. 그리고 사용자 질의 정보를 바탕으로 도메인 사이에서 색인 엔트리들의 클러스터링 정도를 주어진 질의 유형에 적합하도록 조정한다.

제안한 2D-THI의 성능평가를 위해서 균일 데이터 분포를 가정하여 비용 모델을 개발하고, 이를 이용하여 기존의 객체지향 데이터베이스의 클래스 상속 계층 색인구조인 CH-index와 CG-tree를 XML 데이터베이스에 적용하여 이들과 2D-THI를 비교 분석하였다.

성능평가의 결과로 XML 데이터베이스에 하나의 타입을 대상으로 하는 범위 질의인 STR 형태의 질의가 주어지는 경우에는 각 색인구조의 탐색 성능이 2D-THI와 CG-tree가 함께 CH-index보다 월등히 좋음을 보였고, 타입상속 계층을 대상으로 하는 부합 질의인 THM 형태의 질의가 주어지는 경우에는 각 색인구조의 탐색 성능이 2D-THI와 CH-index가 함께 CG-tree 보다 월등히 좋음을 보였다. 그리고 타입상속 계층을 대상으로 하는 범위 질의인 THR 형태의 질의가 주어지는 경우에는 2D-THI가 다른 두 색인구조보다 좋은 성능을 보였다. 마지막으로 혼합 질의가 주어지는 경우에도 2D-THI가 제일 좋은 성능을 보였다.

본 논문에서 제안한 2D-THI에서는 색인 페이지의 영역분할 전략에 의해서 타입상속 계층을 이루는 타입 집합에 대한 부합 질의 유형에 대해서는 CH-index와 같은 키 클러스터링 색인구조를 구성할 수 있으며, 하나의 특정 타입에 대한 범위 질의 유형에 대해서는 CG-tree와 같은 타입 클러스터링 색인구조를 구성할 수 있을 뿐만 아니라 질의 유형에 따라 다양한 형태의 클러스터링 색인구조로도 구성이 가능하다. 따라서 2D-THI를 이용하여 XML 데이터베이스의 타입상속 색인구조를 데이터베이스 응용 시스템에 주어지는 질의 유형에 따라 최적의 클러스터링 색인구조를 구성하면 색인탐색의 성능을 크게 향상시킬 수 있다.

참 고 문 헌

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

[2] W. Meier, "eXist: An Open Source native XML Database," *Web, Web-Services, and Database Systems, NODe 2002 Web- and Database-Related Workshops,* Revised Papers (Lecture Notes in Computer Science Vol. 2593), pp. 169-183, 2003.

[3] Migrating from XML DTD to XML Schema using UML, *http://www.rational.com/products/whitepapers /412.jsp.*

[4] XML Schema Part 0: Primer, *http://www.w3.org /TR /xmlschema-0/.*

[5] S. Finkelstein et al., "Physical Database Design for Relational Databases," *ACM Trans. on Database Systems,* Vol. 13, No. 1, pp. 91-128, Mar. 1988.

[6] A. Berglund et al., "XML Path Language (XPath) 2.0. W3C Working Draft 30 Apr. 2002," *http: //www.w3.org /TR /xpath20,* Working Draft, 2002.

[7] R. Goldman and J. Widom, "DataGuides: Enable Query Formulation and Optimization in Semistructured DataBases," In *Proc. Int'l Conf. on Very Large Data Bases,* Athens, Greece, pp. 436-445, Aug. 1997.

[8] T. Milo and D. Suciu, "Index Structures for

Path Expression," In *Proc. Int'l Conf. on* International Conference on Database Theory, Jerusalem, Israel, pp. 277-295, Jan. 1999.

[9] B. Cooper et al., "A Fast Index for Semistructured Data," In *Proc. Intl. Conf. on Very Large Data Bases*, Rome, Italy, pp. 341-350, Sept. 2001.

[10] C. W. Chung, J. K. Min, and K. Shim. "APEX: An Adaptive Path Index for XML Data," In *Proc. Intl. Conf. on Management of Data*, ACM SIGMOD, Madison, Wisconsin, pp. 121-132, June 2002.

[11] S. Nestorov et al., "Representative Objects: Concise Presentation of Semistructured, Hierarchical Data," In *Proc. Int'l Conf. on IEEE* International Conference on Data Engineering, Birmingham, U.K., pp. 79-90, Feb. 1997.

[12] J. H. Lee et al., "A Region Splitting Strategy for Physical Database Design of Multidimensional File Organizations," In *Proc. Int'l Conf. on Very Large Data Bases*, Athens, Greece, pp. 416-425, Aug. 1997.

[13] W. Kim et al., *Indexing Techniques for Object-Oriented Databases*, In book Object-Oriented Concepts, Databases, and Applications, (Kim, W. and Lochovsky, F. eds.), Addison-Wesley, 1989.

[14] C. Kilger and G. Moerkotte, "Indexing Multiple Sets," In *Proc. Int'l Conf. on Very Large Data Bases*, Santiago, Ghile, pp. 180-191, Sept. 1994.

[15] D. Comer, "The Ubiquitous B-tree," *ACM Computing Surveys*, New York, USA, Vol, 11, No. 2, pp. 121-137, June 1979.

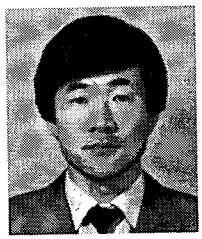
[16] C. C. Low et al., "H-Trees: A Dynamic Associative Search Index for OODB," In *Proc. Int'l Conf. on Management of Data*, ACM SIGMOD, pp. 134-143, San Diego, CA, June 1992.

[17] S. Ramaswamy and P. C. Kanellakis, "OODB Indexing by Class-Division," In *Proc. Int'l Conf. on Management of Data*, ACM SIGMOD, pp. 139-150, San Jose, CA, May 1995.

[18] K. Y. Whang and R. Krishnamurthy, "The Multilevel Grid File - A Dynamic Hierarchical Multidimensional File Structure," In *Proc. Intl. Conf. on Database Systems for Advanced Applications(DASFAA), Tokyo*, pp. 449-459, Apr. 1991.

[19] K. Y. Whang et al., "Dynamic Maintenance of Data Distribution for Selectivity Estimation," *The Very Large Data Bases Journal*, Santiago de Chile, Chile, Vol. 3, No. 1, pp. 29-51, Jan. 1994.

[20] J. T. Robinson, "The K-D-B-Tree: A Search Structure for Large Multidimensional Dynamic Indexes," In *Proc. Int'l Conf. on Management of Data, ACM SIGMOD*, Ann Arbor, Michigan, pp. 10-18, Apr. 1981.



이 종 학

1982년 경북대학교 전자공학과(전자계산 전공) 졸업(학사)
1984년 한국과학기술원 전산학과 졸업(공학석사)
1997년 한국과학기술원 전산학과 졸업(공학박사)
1991년 정보처리기술사
1984년~1987년 금성통신(주) 부설연구소 주임연구원
1987년~1998년 한국통신 연구개발본부 선임연구원
1998년~현재 대구가톨릭대학교 컴퓨터정보통신공학부 교수
관심분야 : 객체 데이터베이스, 다차원 파일구조, 물리적 데이터베이스 설계, XML 데이터베이스, 데이터 웨어하우스, 생물정보학 등