

XBench 벤치마크 성능평가 및 분석 : 순수 관계형 모델과 객체 관계형 모델 비교

김재욱^o 신진호, 이상원
성균관대학교 정보통신공학부
{wodnr^o, sjho14, swlee}@skku.edu

Performance Evaluation and Analysis of XBench Benchmark : A comparison of Pure Relational Model and Object-Relational Model

Jae-Uk Kim^o Jin-Ho Shin, Sang-Won Lee
School of Information & Communication Engineering, Sungkyunkwan University

요 약

현재 XML은 데이터 표현과 교환을 위한 표준으로 자리잡아 가고 있으며, 다양한 응용분야에 도입되어 사용되고 있다. 따라서 데이터베이스 분야에서도 XML문서를 효과적으로 저장하고 검색하기 위한 연구가 활발히 진행되고 있으며, 여러 상용 DBMS에서 XML을 지원하고 있다. 그리고 최근에는 XML 데이터베이스의 성능을 평가하기 위해 다양한 종류의 XML 벤치마크가 제안되고 있는데, 본 논문에서는 특정 상용 객체 관계형 DBMS를 사용해서 XBench 벤치마크를 순수 관계형 기능과 객체 관계형 기능을 사용하여 모델링하고 성능 측정된 결과를 보이고, 이를 통해 각각의 데이터베이스에 대해 성능을 평가하고, 장단점을 비교 분석한다.

1. 서 론

1.1 XML의 개요

최근에 인터넷 환경에서 데이터의 저장과 전송을 위한 표준으로서 XML(eXtensible Markup Language)이 부각되고 있는 상황이며, 지속적으로 XML의 사용이 증가하고 있는 추세에 있다. XML은 1998년에 발표된 이후 가능한 모든 분야에서 시도되고 있는 다양한 어휘집(MathML, SVG 등)들과 더불어 급속도로 퍼져나갔다. 이러한 어휘집과 함께 사용하는 경우 문제가 되는 것은 XML문서가 어떤 어휘집 안에서 사용되고 있는지 그리고 그 어휘집에 잘 맞는지를 어떻게 판단해야 할지를 결정해야 한다는 것이다. 따라서 W3C는 XML 유효성 검증 언어를 제공하기 위해 강력한 데이터 정의와 요소들을 정의할 수 있는 XML 스키마를 발표하게 된다[1].

1.2 본 논문의 목적

본 논문에서는 O사의 상용 객체 관계형 데이터베이스와 순수 관계형 데이터베이스를 구축하여 XBench 벤치마크[2]를 적용하여 성능평가를 하기 위한 환경구축과 데이터의 효율적인 저장, 조작, 검색을 위한 방안에 대해 논의한다. 이들 각각의 시스템의 상대적 특성들은 데이터베이스 성능에 대해 중요한 영향을 끼치게 된다. 따라서 각각의 데이터베이스에 대해 벤치마크를 수행하

고, 시스템 상의 여러 가지 특성과 성능에 대한 비교 분석이 필요하다. 또한 XQuery의 테스트 질의를 O사의 데이터베이스에서 실행할 수 있도록 재 정의하고, 각각의 질의들을 수행하여 얻은 결과에 대해 분석 및 보고함으로써 각각의 데이터베이스에 대한 성능을 비교 분석하는데 그 목적이 있다.

1.3 논문의 구성

본 논문은 다음과 같이 구성되어 있다. 2장과 3장에서 XBench 벤치마크의 소개와 O사의 데이터베이스에서 XML 지원현황에 대해 서술한다. 4장에서는 각 데이터베이스에 대한 성능 실험에 대해 서술하고, 5장에서는 실험을 통해 얻은 결과로 성능평가 및 분석을 한다. 마지막 6장에서는 결론을 서술한다.

2. XBench 벤치마크

XML 벤치마크에는 XMach, Xmark, XOO7, XBench 등이 있는데, 본 논문에서는 다양한 스키마와 데이터를 가지고 있고, 실험에 가장 적합하다고 판단한 XBench를 사용하여 성능평가 한다. 실험에 사용된 XBench 벤치마크의 데이터는 두 가지 특성으로 구분된다. 첫 번째 특성은 애플리케이션 특성이고, 두 번째 특성은 데이터 특성이다. 세부적으로 애플리케이션 특성은 데이터 중심(Data-centric(DC)) 이거나 텍스트 중심(Text-

centric(TC))으로 구분된다. 데이터 특성 또한 두 가지 특성으로 구분할 수 있는데, 첫 번째는 하나의 XML파일로 이루어진 단일 문서(Single document(SD))이고, 두 번째는 여러 개의 XML 문서로 이루어진 다수 문서(Multiple document(MD))로 구분된다.

본 논문에서는 <표 1>과 같은 XBench 벤치마크 데이터를 생성하여 O사의 객체 관계형 데이터베이스와 순수 관계형 데이터베이스에 적용하여 테스트한다. 단 TC/SD는 CLOB Type으로 선언되어야 한다. 이는 순수 관계형 데이터베이스의 특성상 적합하지 않으므로 테스트하지 않는다.

<표 1> XML 데이터 특성.

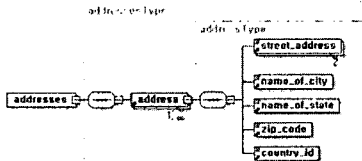
	SD	MD
TC	전자 도서관	뉴스 기사, 전자 도서관
DC	전자상거래 카탈로그	업무처리 데이터

3. O사 데이터베이스

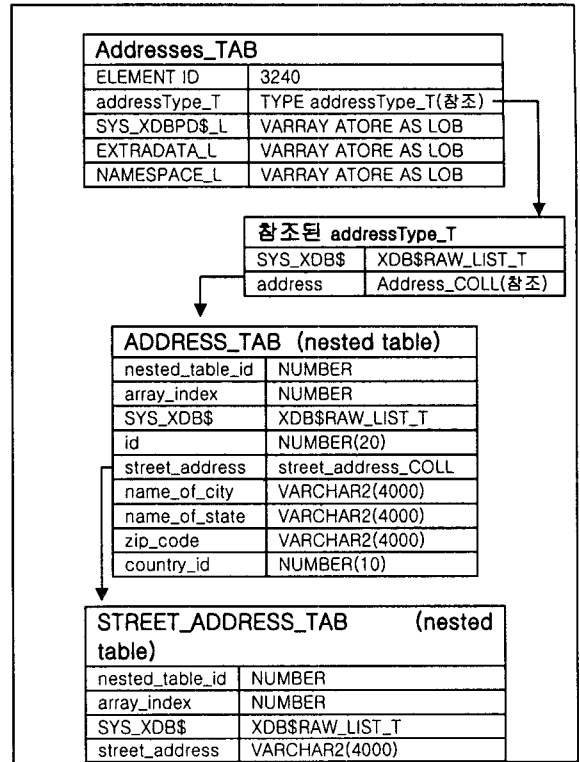
O사의 상용 DBMS에서는 XML 문서의 저장과 검색 및 효율적인 관리를 할 수 있도록 객체 관계형 데이터베이스의 기능들을 확장하여 XML 데이터베이스에 적용함으로써 XML 데이터베이스의 기능을 보장한다. O사의 XML 데이터베이스에서는 W3C XML 데이터 모델을 완벽하게 수용한 XML 저장소를 지원하고, XPath 와 기존의 SQL 질의를 사용하여 데이터에 대한 표준 접근 방식을 사용한다. <그림 1, 2>는 DC/MD 스키마중 DCMDAddr 스키마를 예를 들어 객체 관계형 데이터베이스에서 어떻게 모델링 되는지를 보인 것이다. <그림 3>은 <그림 1>의 스키마를 관계형 데이터베이스로 구축하는 것을 보인 것이다.

4. 성능 실험

이번 장에서는 상용 DBMS인 O사의 객체 관계형 데이터베이스와 순수 관계형 데이터베이스의 성능을 테스트



<그림 1> DCMDAddr 스키마 구성 예



<그림 2> 객체 관계형 모델링

하기 위해 XBench 벤치마크를 적용하여 성능 테스트를 위한 환경 구축에 대해 서술한다.

4.1 실험 환경

모든 실험환경은 팬티엄 2.0GHz, 1GB 주기억장치, 40GB 하드디스크, Red Hat Linux AS 3 에서 실험되었다.

4.2 질의 문

XBench에서 성능평가에 사용되는 XQuery 질의는 모두 20개의 종류로 구성되어 있다. 이 질의 들은 O사에서 지원하지 않는 질의 형태이기 때문에 실행 가능한 질의 형태로 번역하여 실행하였다[3]. <표 3>은 XQuery를 XPath와 SQL로 번역한 하나의 예이다.

<표 2> 실험 결과 (초)

	TCSD		DCSD		DCMD	
	XDB	RDB	XDB	RDB	XDB	RDB
Q5	0.24	72.93	0.10	0.11	0.11	0.06
Q8	1.26	48.38	0.08	0.13	0.18	0.04
Q12	0.20	97.83	0.15	0.14	0.18	0.06
Q14	8.93	1014.18	9.52	22.66	657.06	9.14
Q17	-	-	54.79	9.21	5.50	1.01

```

ADDRESSES_TAB
CREATE TABLE addresses_tab (
  address_id      NUMBER NOT NULL ,
  name_of_city    VARCHAR2 (1000) NULL ,
  name_of_state   VARCHAR2 (1000) NULL ,
  zip_code        VARCHAR2 (1000) NULL ,
  country_id      NUMBER NULL,
  constraint addresses_tab_pk_1
    primary key (address_id)
);

STREET_ADDRESS_TAB
CREATE TABLE street_address_tab (
  address_id      NUMBER NOT NULL,
  array_index     NUMBER NOT NULL,
  street_address  VARCHAR2 (1000) NOT NULL ,
  address_id      NUMBER NOT NULL,
  constraint street_address_tab_pk_1
    primary key (address_id, array_index),
  constraint hw_tab_fk_1
    foreign key (address_id, array_index) references
    addresses_tab (address_id, array_index)
);
    
```

<그림3> 순수 관계형 모델링

<표 3>질의 변역

XQuery (원본 XQuery)
for \$a in input()/order[@id="2"] return \$a/order_lines/order_line[1]
XPath (객체 관계형 기능)
select extract(O.object_value, '/order/order_lines/order_line[1]') from order_tab O where existsNode(O.object_value, '/order[@id=2]') = 1;
SQL (순수 관계형 기능)
select XMLELEMENT("order_line", XMLATTRIBUTES(order_line_id as "id"), XMLELEMENT("item_id", item_id), XMLELEMENT("quantity_of_item", quantity_of_item), XMLELEMENT("discount_rate", discount_rate), XMLELEMENT("special_instructions", special_instructions)) from order_line_tab where order_id=2 and order_line_id=1;

5. 성능 평가 및 분석

이번 절에서는 각각의 질의에 대해 성능을 평가하고 실험 결과에 대해서 보고한다.

5.1 실험 1 : 문서 구조와 물리적 저장

실험 1은 다음의 두 가지 관점에서 데이터가 정확히 저장되었는지를 보장하는 실험이다.

첫 번째로 텍스트 검색에 대한 성능을 실험하고, 두 번째로 텍스트 검색에서 얻어진 XML 데이터가 올바르게 재구성 되었는지를 실험한다.

5.2 실험 2 : 경로 표현과 불규칙성

실험 2는 문서의 불규칙성과 불확실한 경로 표현을 사용하여 데이터에 접근하는 것에 초점을 맞추어 실험한다.

5.3 실험 결과

DC/SD와 DC/MD 스키마의 성능은 결과에서 보이는 것과 같이 객체 관계형 데이터베이스와 순수 관계형 데이터베이스는 몇몇을 제외하고 비슷한 성능을 보이고 있다. 또한 성능이 차이 나는 것은 질의 최적화와 스키마 최적화로 충분히 극복할 수 있는 문제들이다. 그러나 TC/SD의 경우 순수 관계형 데이터베이스로 모델링하는 과정에서 테이블의 수가 너무 많아지고, 각각의 질의에 대한 수행계획에 있어서 조인이 상당히 많이 발생하는 문제로 객체 관계형 데이터베이스의 성능이 월등이 좋은 것으로 나타났다. 또한 XPath 는 단순하고 직관적인 반면 SQL 질의로 번역하는 과정에서 너무 복잡해지고, 어렵다는 것과 조인에 의한 카디션 곱 때문에 객체 관계형 데이터베이스의 결과와 같은 결과 셋을 출력할 수 있도록 SQL 질의를 만들기가 상당히 힘들다는 단점이 있다.

6. 결론

본 논문에서는 XBench를 이용하여 객체 관계형 데이터베이스와 순수 관계형 데이터베이스의 기능성에 대해서 성능평가 하였다. XBench 벤치마크를 통해 다양한 형태의 XML 데이터에 대한 포괄적인 테스트를 할 수 있었다.

또한 이러한 데이터를 통해 O사의 객체 관계형 데이터베이스와 순수 관계형 데이터베이스의 성능 평가를 함으로써 두 데이터베이스의 장점과 단점을 파악할 수 있었다. 따라서 앞으로의 연구 방향은 실험을 통해 알 수 있었던 각 데이터베이스의 단점을 보완하여 성능향상에 대한 연구 및 분석을 해야 하고, XML 스키마의 재정의에 의해 성능에 어떤 영향을 주는지를 연구해야 한다.

참고 문헌

[1] Duckett, Jon, Professional XML Schemas, 2003
 [2] Benjamin Bin Yao and M. Tamer Özsu, Evaluation of DBMSs Using XBench Benchmark, TR-CS-2003-24, University of Waterloo, August 2003.
 [3] Ravi Murthy, Sandeepan Banerjee, XML Schemas in Oracle XML DB, VLDB 2003 in Berlin, Sep 2003