Journal of Korean
Data & Information Science Society
2003, Vol. 14, No.1 pp 37~44

On the Aggregation of Multi-dimensional Data using Data Cube and MDX

Jeong Yong Ahn[1] · Seok Ki Kim[2]

Abstract

One of the characteristics of both on-line analytical processing(OLAP) applications and decision support systems is to provide aggregated source data. The purpose of this study is to discuss on the aggregation of multi-dimensional data. In this paper, we (1) examine the SQL aggregate functions and the GROUP BY operator, (2) introduce the Data Cube and MDX, (3) present an example for the practical usage of the Data Cube and MDX using sample data.

Keywords : Aggregation, GROUP BY, Data Cube, MDX

1. 서론

현대 사회에서는 시간의 흐름에 따라 관측되어 누적된 시계열 데이터 또는 컴퓨터 기술에 기반하여 발생되는 데이터 등과 같은 대용량의 데이터를 흔히 접할 수 있다. 과거 수십 년간 통계청을 비롯한 많은 기관들에서 해마다 작성되어 누적된 통계 데이터는 그 양적인 면에 있어 매우 방대하며, 최근 정보통신 기술이 발전하면서 온라인 업무처리와 관련된 많은 양의 데이터가 양산되고 있다. 이들 데이터는 대용량이라는 특징 이외에도 다차원적(multi-dimensional)으로 구성된 경우가 많다. 예를 들어, 센서스 데이터는 년도, 지역, 조사항목 차원으로 구성되어 있으며, 상거래를 통하여 발생하는 데이터는 기간, 지역, 매장, 제품 등과 같은 차원으로 이루어진다.

이러한 데이터의 대부분은 데이터베이스(database)에 저장되어 관리되어지며, 데이터의 통합과 분석 과정을 거쳐 정보를 제공한다. 그러나 데이터를 분석하는 과정에 있어 데이터베이스의 기능만을 이용하는 것은 한계가 있다. 데이터베이스는 데이터를 종합, 분석, 합병하는 목적으로 설계되지 않아 의사결정에 필요한 분석적이고 다양한

1) Assistant Professor, Department of Computer Science and Statistics, Seonam University, 720 Kwangchi-dong, Namwon 590-711, Korea
E-mail : jyahn@tiger.seonam.ac.kr
2) Department of Computer Science and Statistics, Chonbuk National University, 664-14 Dukjin-dong, Jeonju 561-756, Korea
E-mail : sisyphus@mail.chonbuk.ac.kr

정보를 쉽게 제공하기 어려운 단점을 가지고 있기 때문이다.

일반적으로 대용량, 다차원 데이터의 분석은 어떤 속성들에 대한 합계, 평균 등과 같은 집계(aggregation) 통계량의 비교로부터 시작된다. 특히, 의사 결정 시스템이나 OLAP(on-line analytical processing) 시스템에서 데이터 집계에 대한 질의는 매우 빈번히 발생된다(Thomsen, 1997). 데이터베이스에서 데이터 집계는 질의 언어인 SQL(structured query language)에서 제공하는 sum, avg와 같은 집계 함수와 GROUP BY 연산자를 통하여 이루어진다. 그러나 GROUP BY 연산자를 이용하여 데이터를 분석하는 것은 여러 가지 문제점(2절 참조)을 가지고 있다. Gray 등(1997)은 그 해결책으로 GROUP BY 연산자와 ROLLUP, CUBE 연산자를 병행하여 이용하는 방법을 제안한다. 그러나 이 방법 역시 대용량 데이터인 경우에 질의 결과를 반환하는 시간이 많이 소요되는 단점을 여전히 가지고 있다. 이러한 문제점들은 데이터 큐브를 미리 생성해 놓고 다차원 질의 언어인 MDX(Multi-Dimensional eXpression)를 이용함으로서 대부분 해결할 수 있으나 이에 대한 연구는 찾아보기 힘들다.

본 연구에서는 대용량, 다차원 데이터의 집계 방법에 대해 살펴보고, 효율적인 방안으로 데이터 큐브(data cube)와 MDX의 이용을 제안하고자 한다. 데이터 큐브는 자주 이용되는 데이터를 미리 계산해 두고 필요시 이것을 이용하는 방법으로, 적절한 데이터 큐브의 설계는 OLAP 시스템의 효율성에 매우 중요한 사항이다(Cheung 등, 1999; Harinarayan 등, 1996). 2절에서는 본 연구에서 이용할 예제 데이터에 대해 간단히 소개하고, SQL 집계 함수와 GROUP BY, CUBE, ROLLUP 등의 연산자를 이용한 데이터 집계 방법과 그 문제점을 기술한다. 3절에서는 데이터 큐브의 생성과 MDX의 사용법에 대해 살펴보고, 4절에서 예제 데이터를 이용하여 데이터 큐브와 MDX의 응용예를 제시한다.

2. SQL 집계 연산자

SQL의 GROUP BY, CUBE, ROLLUP 연산자와 MDX를 비교, 설명하기 위하여 본 연구에서는 FoodMart 데이터베이스를 약간 변형하여 이용한다. 이 데이터베이스는 식품 체인점들의 판매와 재고에 대한 정보를 포함하고 있는 데이터이며, 다차원적으로 구성되어 있어 GROUP BY, MDX 등을 비교하기 유용한 형태이다. 대용량 데이터의 질의는 TPC-D 데이터를 이용하는 것이 일반적이다. 그러나 본 연구는 다차원 데이터의 질의에 대한 반응 시간의 정확한 비교보다는 MDX의 활용성에 대해 살펴보는 것이 목적이므로 조금 더 현실적인 FoodMart 데이터를 이용하고자 한다.

FoodMart 데이터는 여러 개의 테이블로 구성되어 있으며, 본 연구에서 이용하게 될 주요 테이블의 필드는 <표 1>과 같이 설계되어 있다.

<표 1> FoodMart 데이터베이스 주요 테이블의 필드 구성

테이블 명	sale_fact	product	store	time_by_day
필드	product_id time_id customer_id store_id store_sales store_cost :	product_class_id product_id brand_name product_name gross_weight units_per_case :	store_id region_id store_name store_address store_city store_country :	time_id the_day the_month the_year week_of_year month_of_year :

GROUP BY 연산자는 특정 필드의 값에 따라 데이터를 그룹 짓는 기능을 하며, 일반적으로 SQL에서 제공하는 집계 함수들과 함께 사용되어 데이터의 요약 정보를 제공해준다. 예를 들어, <표 1>의 테이블에서 기간, 지역별 총 판매량의 통계는 다음과 같은 질의를 통해 산출할 수 있으며, 질의 결과는 <표 2>와 같다.

```
SELECT      the_year, store_country, SUM(store_sales) as SumSale
FROM        sales
GROUP BY    the_year, store_country
```

<표 2> GROUP BY 질의 결과

the_year	store_country	SumSale
1997	USA	2260952
1998	USA	2203233
1998	Canada	392181
1998	Mexico	1721174

<표 2>에서 볼 수 있는 바와 같이 GROUP BY를 이용하는 질의의 결과는 그룹화하기 위해 선택된 필드(또는 차원)들에 대한 집계 통계량을 제공하는 역할을 한다. 그러나 이 결과는 Gray 등(1997)에서 지적하는 바와 같이, (1) 히스토그램적인 정보를 표현할 수 없으며, (2) Roll-up과 Drill-down 문제, (3) 교차 테이블 표현의 어려움 등과 같은 문제점을 가지고 있다. 이러한 문제점은 ROLLUP, CUBE 등의 연산자를 함께 이용하면 어느 정도 해결할 수 있다. ROLLUP, CUBE 등의 연산자는 GROUP BY 연산자와 함께 이용되어 요약된 데이터에 대한 2차적인 요약을 할 수 있다. ROLLUP, CUBE 연산자의 사용 예는 다음과 같다.

```
SELECT      the_year, store_country, SUM(store_sales) as SumSale
FROM        sales
GROUP BY    the_year, store_country WITH ROLLUP
```

<표 3> ROLLUP 연산자를 이용한 질의 결과

the_year	store_country	SumSale
1997	USA	2260952
1997	total	2260952
1998	Canada	392181
1998	Mexico	1721174
1998	USA	2203233
1998	total	4316588
total	total	6577540

```
SELECT      the_year, store_country, SUM(store_sales) as SumSale
FROM        sales
GROUP BY    the_year, store_country WITH CUBE
```

<표 4> CUBE 연산자를 이용한 질의 결과

the_year	store_country	SumSale
1997	USA	2260952
1997	total	2260952
1998	Canada	392181
1998	Mexico	1721174
1998	USA	2203233
1998	total	4316588
total	total	6577540
total	Canada	392181
total	Mexico	1721174
total	USA	4464185

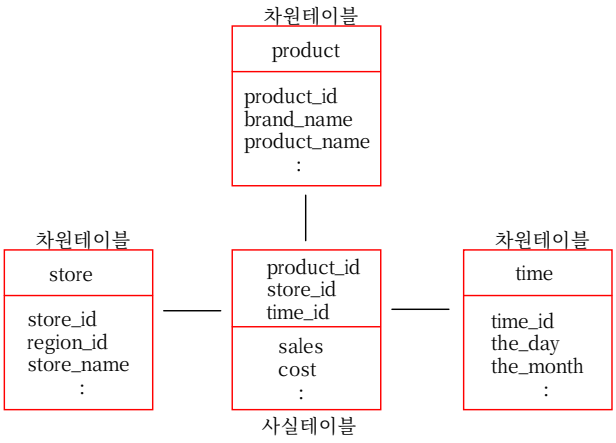
<표 3>과 <표 4>는 ROLLUP 연산자와 CUBE 연산자를 이용한 질의의 결과이다. 이 결과들은 GROUP BY 연산자만을 이용했을 때보다 집계 결과가 확실히 더 유용한 형태이며, total 값들을 제공함으로서 히스토그램적인 정보를 표현해 주고 있다. 그러나 이러한 연산자를 사용한다고 해도 여전히 (1) Roll-up과 Drill-down 등이 자유롭지 못하고, (2) 반응 시간이 많이 소요되며, (3) 일반 사용자들에게 정보를 제공하고자 할 때 인터페이스 개발이 어렵다는 문제점을 가지고 있다.

3. 데이터 큐브와 MDX

데이터 큐브(data cube)는 분석이 필요한 데이터를 다차원적으로 조직화해 정리해 놓은 것으로 OLAP 시스템에서 주로 이용되는 개념이다. OLAP은 사용자들에게 데이터의 흐름을 다차원적인 구조로 빠르게 보여주면서 몇몇 미리 계산된 값들을 제공함으로서 데이터에 대한 통계적인 요약 정보를 쉽게 제공해 주는 기술이다(Lenz와 Shoshani, 1997; Shoshani, 1997). 데이터의 다차원적인 구조는 데이터에 대한 비교를 가능하게 해줄 수 있기 때문에 의사 결정에 있어 매우 유용하게 활용될 수 있으며, 이에 대한 많은 연구가 이루어지고 있다(안정용 등, 2000; Chaudhuri 등, 1997)

2절에서 살펴본 CUBE 연산자는 데이터 큐브를 생성하는 연산자이다. 그러나 CUBE 연산자를 SQL 구문 안에서 이용하게 되면, SQL 문장이 실행되는 시점에 데이터 큐브가 생성되어 결과를 반환하게 되며, 2절에서 언급한 여러 가지 문제점들이 발생한다. 이러한 문제점들을 해결하기 위하여 OLAP에서는 다차원 데이터 큐브를 미리 생성해 놓고, 다차원 데이터의 질의 표준 언어를 이용하여 정보를 가져오는 방식을 사용한다.

OLAP 데이터 모델에서 다차원 데이터는 주로 스타 스키마(star-schema)라는 관계형 데이터베이스 설계 기법을 이용하여 표현한다. 스타 스키마는 <그림 1>과 같이 사실 테이블(fact table)과 차원 테이블(dimensional table)로 분류된다. 사실 테이블은 분석을 요하는 변수 차원의 항목들을 포함하고 있는 테이블이며, 차원 테이블은 사실 테이블의 변수들을 살펴보기 위한 범주형 속성의 계층적인 정보를 포함한다. 스키마가 완성되면 데이터베이스에 테이블과 데이터 큐브를 생성한다. 생성된 큐브에서 데이터의 질의는 MDX를 이용한다.



<그림 1> 스타 스키마

MDX는 다차원 데이터를 질의하기 위한 언어로, 마이크로소프트사에 의해 제안된 OLE DB for OLAP API에 포함되어 있다. MDX는 집계 함수, 시계열 함수 등 다양한 함수를 제공하고 있으며, 많은 기업들의 호응을 얻고 있어 다차원 질의 언어의 표준으로 받아들여지고 있다(Thomsen 등, 1999). SQL과 비슷하게 MDX 질의는 SELECT, FROM, WHERE 절을 포함하고 있으며, 기본 형식은 다음과 같다.

```
SELECT      { 축 차원 요소들 } on columns,
            { 축 차원 요소들 } on rows
FROM        큐브 이름
WHERE       슬라이서 차원
```

<표 5> Sales 큐브

Store	Time	Sales	Cost
Downtown	June-1998	1200	1000
Downtown	July-1998	1300	1050
Uptown	June-1998	1000	800
Uptown	July-1998	1000	900

예를 들어, Sales 큐브가 <표 5>와 같이 구성되어 있다고 할 때, 다음 MDX 질의는 <표 6>과 같은 결과를 보여준다.

```
SELECT
            { [Members].[Sales], [Members].[Cost] } on columns,
            { [Time].[June-1998], [Time].[July-1998] } on rows
FROM        Sales
WHERE       ( Store].[Downtown] )
```

<표 6> MDX 질의 결과

		Measures	
		Sales	Cost
Time	June-1998	1200	1000
	July-1998	1300	1050

4. 데이터 큐브와 MDX의 이용 예

데이터베이스를 이용하는 시스템에서 데이터는 여러 개의 테이블에 나누어 보관되는 것이 일반적이다. 이러한 테이블들로부터 정보를 추출하기 위해서는 조인(join) 연산이 불가피하기 때문에 질의 결과를 반환하는 시간이 많이 소요된다. 반응 시간 문제는 통계적인 관점에서 표본을 추출하여 이용하는 방법을 통하여 해결할 수도 있겠으나 다차원 비교 분석의 특성으로 볼 때 이 방법은 유용하지 않은 것으로 생각된다. 또 조인 연산이 많이 발생되지 않는다고 해도, 대용량 데이터인 경우에 집계 시간이 상당히 필요하게 되는 문제점도 있다.

<표 1>의 sales_fact 테이블에는 대략 백만 개의 레코드가 저장되어 있다. <표 1>의 테이블을 이용했을 때의 GROUP BY 연산자와 MDX 질의의 대략적인 반응 시간을 비교한 결과를 <표 7>에서 보여주고 있다. 물론 이 결과는 컴퓨터 환경에 따라 다르기 때문에 절대적인 평가는 어렵겠지만 반응 시간에 대한 상대적인 차이를 가늠해볼 수 있다. MDX를 이용한 질의는 반응 시간이 거의 소요되지 않음을 알 수 있으며, 이러한 결과는 다차원 데이터 큐브를 미리 생성해 놓은 상태에서 질의가 이루어

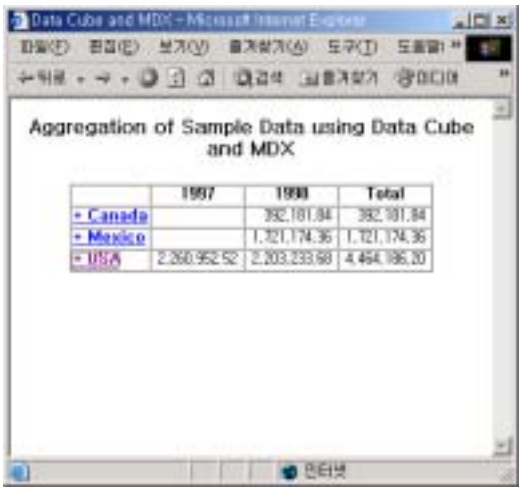
지기 때문에 당연한 현상으로 받아들일 수 있다.

<표 7> 질의 반응 시간

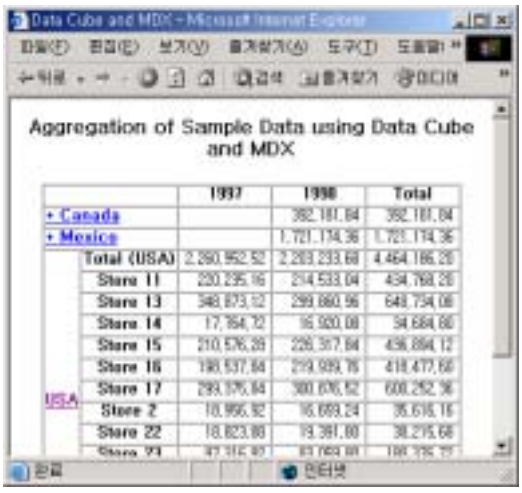
조인 테이블 수	SQL	MDX
2	≅ 3 sec	≤ 1 sec
3	≅ 6 sec	≤ 1 sec
4	≅ 10 sec	≤ 1 sec

<그림 2>와 <그림 3>은 데이터 큐브와 MDX를 이용하는 예이다. 그림과 같이 웹 브라우저를 통하여 일반 사용자들에게 정보를 제공하는 인터페이스 개발은 간단한 프로그래밍 작업을 통하여 쉽게 구현될 수 있다. <그림 3>은 <그림 2>에서 Roll-up과 Drill-down 기능을 활용하는 예이다. 데이터 큐브에서 정보는 다음과 같은 MDX 질의를 통하여 추출할 수 있으며, 이러한 기능의 활용을 통하여 판매량의 전체적인 현황 및 기간, 지역(또는 매장)별 판매 현황을 쉽게 파악할 수 있다.

```
SELECT
        [Time].[The Year].members on columns,
        NON EMPTY [Store].[Store Country].[USA].children on rows
FROM    FoodMartSalesCube
```



<그림 2> 판매량에 대한 MDX 질의



<그림 3> Roll-up과 Drill-down 이용

5. 결론

의사결정의 범위가 넓어지고 복잡해짐에 따라 사용자들은 데이터의 다차원적인 비교 정보를 활용하고자 한다. 다차원 정보는 사용자들에 의해 이해되는 실제 차원(예를 들어, 기간, 제품, 지역 등)을 반영하는 정보이기 때문에 특히 비즈니스 분야에서 그

중요성이 증가하고 있다.

본 연구에서는 SQL 집계 함수와 GROUP BY, CUBE, ROLLUP 등의 연산자를 이용한 데이터 집계 방법과 그 문제점에 대해 살펴보고, 대용량, 다차원 데이터의 효율적인 집계 방안으로 데이터 큐브와 MDX의 활용을 제안하였다. SQL 집계 연산자를 이용하는 방법과 비교하여 볼 때 데이터 큐브와 MDX의 활용은 더 많은 정보와 편리성을 제공해 줄 수 있으며, 다양한 각도에서 데이터를 분석하고 의사 결정에 쉽게 활용할 수 있는 이점이 있다.

참고문헌

1. 안정용, 한경수 (2000). On-Line Analytical Processing and Research Problems for Statisticians, 한국통계학회논문집, 제7권, 제2호, 457-463.
2. Chaudhuri, S. and Dayal, U. (1997). An Overview of Data Warehouses and OLAP Technology, *ACM SIGMOD Record,* Vol. 26, No. 1, 65-74.
3. Cheung, D. W., Zhou, B., Kao, B., Lu, H., Lam T. W. and Ting, H. F. (1999). Requirement-Based Data Cube Schema Design, *Proceedings of the Information and Knowledge Management*, http://www.csis.hku.hk/~dcheung/publication.html
4. Gray, J., Chaudhuri, S., Bosworth, A., Layman, A., Reichart, D. and Venkatao, M. (1997). Data Cube: A Relational Aggregation Operator Generalizing Group-By, Cross-Tab, and Sub-Totals, *Data Mining and Knowledge Discovery*, Vol. 1, No. 1, 29-53.
5. Harinarayan, V., Rajaraman, A. and Ullman J. D. (1996). Implementing Data Cubes Efficiently, *Proceedings of the ACM SIGMOD on Management of Data*, 205-216.
6. Lenz, H. J. and Shoshani, A. (1997). Summarizability in OLAP and Statistical Data Bases, *Proceedings of International Conference on Statistical and Scientific Database Management,* http://www.lbl.gov/~arie/papers/
7. Shoshani, A. (1997). OLAP and Statistical Databases: Similarities and Differences, *Proceedings of the ACM Symposium on Principles of Database Systems(PODS),* 185-196.
8. Thomsen, E. (1997). *OLAP Solutions: building multidimensional information systems*, John Wiley & Sons, New York.
9. Thomsen, E., Spofford, G. and Chase, D. (1999). *Microsoft OLAP Solutions*, John Wiley & Sons, New York.

[2002년 11월 접수, 2003년 2월 채택]