

DW에서의 질의어처리 성능향상을 위한 데이터 구조화 방법

이덕근*, 오미화**, 조재훈***, 최인수****

A Data Structuring Technique for Performance Enhancement of Query Processing in the Data Warehouses

Deok-Keun Lee*, Mi-Hwa Oh**, Jae-Hun Cho***, In-Soo Choi****

요약

OLAP(On-Line Analytical Processing) 시스템은 사용자가 다양한 각도에서 정보에 접근하여 대화식으로 정보를 분석할 수 있는 의사결정 지원 시스템이다. 그러나 기존 OLAP 시스템의 구축방법에는 사용자 요구사항의 변경에 따른 복잡한 MDX(Multidimensional Expressions) 쿼리문 사용으로 시간과 비용이 증가되는 비효율성이 생기게 된다. 이와 같은 문제를 해결하고자 사실테이블 내에 유니트 컬럼 매트릭스를 추가하는 새로운 개념의 데이터 구성 방법을 제안하고 있다. 이러한 데이터 구성 방법을 사용함으로써 OLAP 시스템에서의 처리 시간과 비용을 줄일 수 있었다

Abstract

An OLAP(On-Line Analytical Processing) system is the decision support tool with which a user can analyze the information interactively in the various aspects. However, the traditional existing construction of an OLAP system has the inefficiency problem of increasing the processing time and cost caused by the use of complex MDX(Multidimensional Expressions) queries. In an attempt to solve this problem, a new concept of data structuring technique, where a unit column whose elements are all 1 is added to the fact table, was suggested. With the data structuring technique, we can reduce the processing time and cost in OLAP systems.

▶ Keyword : OLAP, MDX, Unit Column

• 제1저자 : 이덕근

• 접수일 : 2005.01.04, 심사완료일 : 2005.02.28

* 송실대학교 산업·정보시스템공학과 박사수료 ** 송실대학교 산업·정보시스템공학과 박사과정

*** 송실대학교 산업·정보시스템공학과 석사과정 **** 송실대학교 산업·정보시스템공학과 교수 재직

I. 서론

프로오늘날 기업 조직 내부에서 발생하는 정보의 수요는 기업간의 시장 경쟁이 치열해 짐에 따라 증폭되고 있으며, 정보수요를 일일이 만족시킬만큼 기업의 정보시스템이 충분한 능력을 갖고 있는 것도 아니다. 또한 정보를 얻기 위해 능동적으로 테이블을 검색하고 분석해 보려는 욕구가 증가하고 있다. 이러한 환경변화에 대응하기 위하여 새로운 데이터 모델링 기법[11]이나 새로운 시스템 개발방법론[12] 등이 연구되기 시작했고, 특히 데이터 웨어하우스(Data Warehousing)와 OLAP(On-Line Analytical Processing: 온라인분석처리)와 같은 새로운 개념[1]이 등장되고 있다.

데이터웨어하우스(Data Warehouse, 이하 DW로 기록)는 "다년간 기업의 운영시스템에 집적된 데이터를 주제영역별로 통합하여 별도의 프로그램 없이 다양한 각도에서의 분석을 가능케 하는 시스템"이다[2]. 보다 실무적인 측면에서의 DW는 "의사결정을 위해 필요한 분석 정보를 자동으로 모아두고, 결합시켜 사용자가 원하는 시점에 원하는 형태로 제공하는 시스템"이라는 정의가 보편적으로 사용되고 있다[3].

OLAP은 사용자가 의사 결정에 필요한 지식을 찾아내기 위해 대량의 데이터를 쉽게 분석할 수 있도록 도와주는 데이터베이스 응용이다[4]. 의사 결정에 있어서는 개별적인 레코드들보다 레코드들을 요약한 경향이 중요하기 때문에 상당수의 OLAP 질의들이 데이터를 요약하는데 사용되는 집계(aggregate)를 포함하고 있다[5,6,7,8]. OLAP에서는 다차원 데이터 모델을 사용하는데, 이 모델은 데이터를 분석의 대상이 되는 측정값(measure)들과 측정값을 결정하는 차원(dimension)으로 구분한다. 그리고 각 차원은 다차원 배열의 하나의 축(axis)으로 대응시키고 측정값은 배열의 셀(cell)에 저장된 값으로 대응시킨다. 다차원 데이터 모델은 차원들의 값의 변화에 따른 측정값의 변화를 분석하는 OLAP 사용자의 논리적 사고 방식에 적합하다고 알려져 있다[4].

DW에서는 다차원 분석을 지원하기 위하여 눈송이(snow-flake), 별모양(star) 설계기법을 사용하며, 다수의 SBT(subject-based tables)들로 구성된다[9]. SBT는 다시 작은 크기의 차원(dimension) 테이블과 대용량 팩트

(fact) 테이블로 구성된다. 차원 테이블 정보는 팩트 테이블 정보 분석의 기반 데이터라고 할 수 있으며, 대부분의 DW에서는 분석을 위하여 누적 관리된다. 기존 DW 운영시스템은 개별적으로 고유의 기능만을 수행하여, 실시간으로 일회성 데이터만을 다룰 뿐만 아니라, 무엇보다도 컬럼(column) 위주의 쿼리만이 가능함으로 분석상의 한계를 가지고 있었다. 의사결정 지원시스템의 기반이 되는 DW는 이러한 분석의 한계를 극복하고자 데이터 통합을 통한, 데이터를 이용한 주제지향적 (subject-oriented) 분석이나 복잡한 MDX(Multidimensional Expressions) 쿼리문 혹은 함수를 사용한다.

본 연구에서는 DW 환경에서 OLAP 데이터 분석을 위해 사실테이블(fact table)을 중심으로 효율적인 다차원 분석을 위한 데이터 처리 방법을 제시한다. 그리고 이 방법을 실제 사용 데이터에 적용하여 복잡성과 제한성이 최소화됨을 증명한다. 즉, 복잡한 MDX나 함수 사용 없이 단순 수식을 가지고 원하는 결과값을 쉽게 도출 가능함을 보인다. 또한 DW에서 호환성 있는 데이터 관리를 위해 데이터 분석 시 발생하는 문제점을 정의하고 이를 해결하기 위한 방안들을 소개하고자 한다.

II. 기존 큐브 상에서의 다차원 분석의 문제점

DW 환경에서 다차원 분석을 위해 큐브 생성에 대한 많은 연구가 이루어지고 있으며 생성된 큐브를 분석하기 위한 피보팅(pivoting), 슬라이싱(slicing), 다이싱(dicing), MDX 등 다양한 방법이 제시되고 있다.

또한 분석의 방법으로 측정값의 집계 분석 이외에 별도의 셀이나 구성원에 적용하는 방법도 많이 활용되고 있다. 즉 별도의 셀이나 구성원을 사용하여 의미없는 원시 데이터를 업무지표로 모델링하여 분석 기능을 향상시킬 수 있다. 가령 제품 판매에 대한 큐브가 있을 때 마케팅 부서에서 각 매장별 혹은 제품별로 판매이익을 도출하고자 한다면 큐브의 원시데이터인 측정값의 집계만으로는 판매이익을 알 수 없다. 별도의 계산된 구성원(calculated members)에 판매금액과 판매비용을 이용하여 매장별 혹은 제품별 판매이익을

산정하게 하면 마케팅 부서에서는 계산된 구성원에 적용된 값을 활용할 수 있다. 즉 계산된 구성원에 수식 (2.1)을 적용하게 되면 <표 1>과 같은 판매이익을 이끌어 낼 수 있다(10).

$$\begin{aligned} & (\text{measure}).(\text{판매이익}) \text{ AS} \\ & (\text{measure}).(\text{판매금액}) - (\text{measure}).(\text{판매비용}) \\ & \dots\dots\dots (2.1) \end{aligned}$$

표 1. 제품별 판매이익
Table 1. Margin of Each Product

제품	측정값	판매 금액	판매 비용	판매이익
Product 1		40000	20000	20000
Product 2		30000	15000	15000

이렇게 유용하게 활용 가능한 계산된 구성원에도 몇 가지 문제점이 있다. 즉 계산된 구성원에 적용되는 MDX식의 효율성을 감소시킬 뿐만 아니라 최종사용자로 하여금 사용상에 불편을 초래 할 수 있다는 것이다.

(그림 1)은 상점에 대한 수준별 구성원을 나타낸 것이다. 판매액을 측정값이라고 할 때, 우리가 State 수준별 상점의 평균 판매액을 살펴보고자 한다면 Group 1, 2, 3의 평균을 구하면 된다. 즉 State 수준 구성원의 각 평균 판매액은 City 수준의 Group 1, Group 2, Group 3 구성원을 통해 산정할 수 있다. 그러나 일반적으로 MDX식을 통해 계산된 구성원을 구성하게 되면 Group 0 구성원을 통해 State 수준 전체 구성원을 대상으로 한 평균 판매액이 산정되게 된다. 예를 들면 계산된 구성원에 State 수준 구성원의 각 평균 판매액을 표현하면 'Group 0의 총 판매액 / Group 0의 구성원의 수'로 표현할 수 있을 것이다. 그러나 이렇게 산정된 값은 평균 판매액이 모두 같기 때문에 실질적인 분석을 위해서는 아무런 의미가 없는 값이 된다. 우리가 State별 구성원의 각 평균 판매액이 의미있는 값으로 하기 위해서는 Group 1, Group 2, Group 3의 구성원 수를 통해 CA, OR, WA의 평균 판매액을 산정해야 한다. 즉 State 수준의 CA는 'Group 1의 총 판매액 / Group 1의 구성원의 수'로 표현된 MDX 식을 통해 우리가 원하고자 하는 값을 도출해야 한다는 것이다. 마찬가지로 방법으로 OR과 WA도 'Group(2,3)의 총 판매액 / Group(2,3)의 구성원의 수'를 통해 State 수준별 상점의 평균 판매액을 산정할 수 있다.

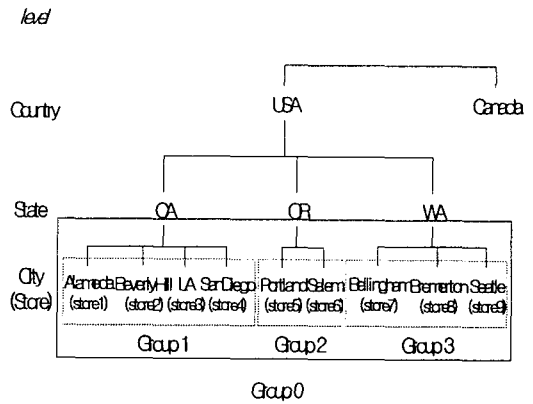


그림 1. Store 차원의 수준별 구성원
Fig 1. Dimension Hierarchy for City, State, and Country

이와 같은 분석 시 필요한 기본 개념을 바탕으로 MDX식을 포함하는 계산된 구성원을 활용할 때 다음과 같은 문제점이 발생할 수 있다.

먼저 차원 피보팅을 통해 데이터 분석을 하려고 할 때 기존에 생성되어 있는 계산된 구성원이 의미 없는 값으로 바뀔 수 있다. OLAP에서 가장 기본적이고 중요한 분석방법이 차원의 전환을 통해 측정값을 분석하는 차원 피보팅이다. 수식(2.1)처럼 측정값만의 수식이 적용된 계산된 구성원은 피보팅을 통한 분석 시에도 계산된 구성원이 의미 있는 값으로 전환이 될 수 있을 것이다. 그러나 MDX수식에 차원의 구성원이 포함되었을 경우 차원 피보팅을 하게 되면 계산된 구성원 값은 의미 없는 값이 된다. 가령 앞서 본 예에서의 차원 피보팅 즉 제품에 대한 판매이익을 상점에 대한 판매이익으로 차원 피보팅을 하면 이 계산된 구성원은 의미 있는 값으로 분석자에게 전해진다. 그러나 제품별 평균 판매액을 계산된 구성원에 적용한 값은 상점별 평균 판매액으로 차원 피보팅을 하게 되면 기존에 생성되었던 계산된 구성원 값은 의미를 가지지 못하게 되므로 차원 피보팅을 할 때마다 별도의 계산된 구성원을 생성해야 하는 문제점이 있다.

다음으로 다중 차원을 적용했을 때 발생하는 문제점이다. 보통 계산된 구성원은 하나의 차원에 여러 개의 측정값을 적용해 수식을 완성한다. 그러나 2개 이상의 다중 차원을 적용하게 되면 MDX식이 복잡해 질 뿐만 아니라 계산된 구성원을 적용하는데도 어려움이 따른다. 예를 들어 정부에서 중소기업 지원 사업을 추진하고 있다고 하자. 이때 지원 사업에 대해 지역별 혹은 분야별 지원현황을 확인하고자 한다면, 이러한 요구사항을 분석하기 위해서는 지역관련 차원과

지원분야관련 차원을 고려한 지원기업을 파악해야한다. 하지만 별도의 큐브를 생성하지 않는 이상 이 요구사항을 해결하기는 쉽지가 않다. 설사 해결한다고 하더라도 매우 복잡하게 될 것이 분명하다.

위와 같은 문제점이 발생하는 하는 이유는 일반적으로 OLAP에서 수식은 열(column)과 열의 조합으로 생성되기 때문이다. 앞서 본 예에서는 열들의 계산은 물론 행(row)단위로 묶어 계산을 해야 하기 때문에 여러 문제가 발생하게 되는 것이다. 이런 문제를 해결하기 위해서 본 연구에서는 큐브 생성단계에서 행을 미리 계산하고, 데이터를 분석할 때 효율적으로 적용하기 위해 DW의 사실테이블을 새롭게 구성하는 방법을 제시하고자한다.

III. 효율적인 다차원 분석을 위한 DW 구성

본 연구에서 제시하는 DW구성 방법을 채택하면 계산된 구성원을 사용하는 다차원 분석에서 발생 할 수 있는 여러 문제점을 해결할 수 있게 된다. OLAP에서 열과 열 계산에 주로 사용되던 계산된 구성원을 행의 계산에도 쉽게 적용시키기 위해 본 방법에서는 DW에 새로운 열을 추가하여 집계 처리 시 행의 계산을 미리 처리하고 있다. 이렇게 새롭게 추가한 열을 활용한 선 계산을 함으로써 기존에 계산된 구성원에 발생했던 차원 피보팅과 다중 차원 적용 문제점을 해결할 수 있게되는 것이다.

3.1 DW 구성

본 연구에서 제시하고 있는 구성방법의 효율성을 증명하기 위해 다음과 같이 실제 사용하는 데이터를 바탕으로 OLAP을 구성해 보고자 한다.

(그림 2)에서는 부품·소재통합연구단(Inter-research Consortium, 이하 iCON으로 기록)의 2004년도 종합기술 지원사업에 대한 기업의 지원요청을 별모양 스키마 구조로 나타내고 있다. 차원은 시간(time), 기업분야(organ), 지역별 요청기업(regionsupport), 기술별 요청기업(techsupport)으로 구성되어 있고, 사실테이블은 차원테이블의 기본키를 참조하고 총 매출액과 R/D투자액을 측정값(measure)으로 한다.

그리고 본 연구에서 제시한 방법에 따라 추가한 열을 측정값에 포함하고 그 값은 모두 '1'의 값을 가지게 한다. 그리고 우리는 이러한 열을 'Unit Column'이라고 하겠다. 이렇게 추가된 열은 집계 처리를 할 때 행의 계산값으로 선 계산된다.

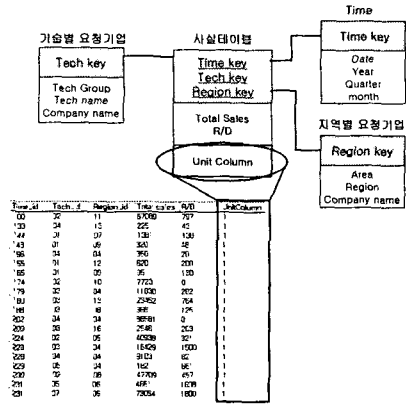


그림 2. 부품·소재통합연구단 종합기술지원사업 스키마 및 데이터
Fig 2. iCON's Base Table for Measuring Data and Lookup Tables for Techniques, Times, and Regions

3.2. 문제점 해결

먼저 위에서 살펴본 DW를 바탕으로 큐브를 생성하여 분석 리포트를 작성해 보겠다.

부품·소재통합연구단은 2004년도 지원사업에 요청한 기업을 기술별 분포로 일목요연하게 확인하기를 원한다. 이 분석 보고서를 보기 위해서는 기술별 요청기업 차원을 축으로 요청기업의 수를 집계해서 리포트에 나타내어야 한다. 기술별 요청기업의 수를 나타낼 MDX식은 다음과 같이 표현될 수 있다.

count(descendants((techsupport).currentmember.children,(techsupport).(company name))) (3.1)

수식(3.1)은 기술별 요청기업 차원의 수준(level)별로서 기업의 수를 카운트(count)하는 수식으로 기술별 요청기업의 수를 알아볼 수 있다. 이런 복잡한 MDX식을 계산된 구성원 '기술별 지원업체'에 적용한 리포트를 (그림 3)을 통해 확인할 수 있다.

Tech Group	Tech Name	MeasuresLevel		Rd Invest	tech별지원업체
		Sales1			
All TechSupport	All TechSupport 합계	4,367,311		83,571	128
기술개발	기술개발 합계	1,975,932		33,614	79
	개발	927,576		8,246	29
	기술	188,752		7,699	31
	자율지	659,564		17,667	19
연구인사	연구인사 합계	186,245		23,664	37
	연구	134,112		12,533	14
	연구	52,137		11,129	23
협력설류	협력설류 합계	2,205,130		26,293	12
	설류	21,484		842	2
	협력	2,183,646		25,451	10

그림 3. 기술별 지원업체 리포트

Fig 3. Technically Classified Asking Companies Report for iCON's Support

그러나 기술별 요청기업 차원에서 지역별 요청기업 차원으로 피보팅을 할 때 문제가 발생하는데, 지역별 요청기업 차원으로 피보팅을 하면 기존에 생성한 계산된 구성원인 '기술별 지원업체'는 의미 없는 값으로 바뀌기 때문에 새로운 계산된 구성원을 생성해야 한다. 새롭게 생성한 계산된 구성원 '지역별 지원업체'에 적용된 MDX식은 다음과 같다.

count(descendants((regionsupport).currentmember.children,(regionsupport).(company name))) (3.2)

지역별 요청기업 차원의 수준(level)별로 기업의 수를 MDX식을 통해 지역별 요청기업의 수를 알 수 있다. 계산된 구성원 '지역별 지원업체'에 적용한 리포트는 (그림 4)의 ①이다. (그림 3)의 ②를 통해 보는바와 같이 생성한 '기술별 지원업체' 계산된 구성원은 차원 피보팅을 하게 되면 '128'이라는 의미 없는 값으로 바뀌는 것을 확인할 수 있다.

Area	Region	MeasuresLevel		Rd Invest	region별지원업체	tech별지원업체
		Sales1				
All RegionSupport	All RegionSupport 합계	4,367,311		83,571	128	128
경상	경상 합계	812,736		7,862	35	128
	경남	125,940		2,642	12	128
	경북	27,642		1,321	5	128
	대구	106,444		1,421	6	128
	부산	535,212		1,986	10	128
경인	경인 합계	697,000		35,715	55	128
	경기	470,950		28,181	40	128
	인천	416,050		1,534	15	128
서울권	서울권 합계	2,137,018		24,335	14	128
	경기도	3,879		75	1	128
	서울	2,133,140		24,260	13	128
전라	전라 합계	21,547		1,553	2	128
	광주	1,671		467	2	128
	전북	19,876		1,086	3	128
충청	충청 합계	509,009		14,305	16	128
	충남	13,390		1,180	7	128
	충북	495,619		13,125	9	128
영남	영남 합계	81,533		5,248	7	128
	울산	434,106		5,676	4	128

그림 4. 지역별 지원업체 리포트

Fig 4. Regionally Classified Asking Companies Report for iCON's Support

즉 기존분석 방법으로는 각 차원마다 의미 있는 계산된

구성원을 보기 위해서는 새로운 계산된 구성원을 생성해야만 한다.

이러한 문제점을 해결하기 위해 계산된 구성원에 MDX식을 사용해 행의 계산을 별도로 적용하는 것이 아닌, 본 연구에서 제시한 것처럼 큐브 생성과정에서 행의 계산을 위해 새로운 열을 추가해서 큐브를 생성하면 차원 피보팅을 할 때마다 발생하는 문제점을 해결할 수 있을 뿐 아니라 복잡한 MDX식도 구현할 필요도 없게 된다.

(그림 5)는 추가한 열을 적용해 큐브 생성 후 기술별 요청업체에 관한 분석 리포트이다.

Tech Group	Tech Name	MeasuresLevel		Rd Invest	Unit Column
		Sales1			
All TechSupport	All TechSupport 합계	4,367,311		83,571	128
기술개발	기술개발 합계	1,975,932		33,614	79
	개발	927,576		8,246	29
	기술	188,752		7,699	31
	자율지	659,564		17,667	19
연구인사	연구인사 합계	186,245		23,664	37
	연구	134,112		12,533	14
	연구	52,137		11,129	23
협력설류	협력설류 합계	2,205,130		26,293	12
	설류	21,484		842	2
	협력	2,183,646		25,451	10

그림 5. 제시된 방법을 적용한 기술별 지원업체 리포트

Fig 5. Technically Classified Asking Companies Report for iCON's Support Using Unit Method

계산된 구성원 생성 없이 지원업체의 수를 알 수 있다. 기술별 지원업체를 지역별 지원업체에 대한 차원 피보팅을 해도 별다른 조작이나 계산된 구성원 없이 피보팅만으로 쉽게 리포트를 생성할 수 있다는 것을 (그림 6)을 통해 확인할 수 있다.

Area	Region	MeasuresLevel		Rd Invest	Unit Column
		Sales1			
All RegionSupport	All RegionSupport 합계	4,367,311		83,571	128
경상	경상 합계	812,736		7,862	35
	경남	125,940		2,642	12
	경북	27,642		1,321	5
	대구	106,444		1,421	6
	부산	535,212		1,986	10
경인	경인 합계	697,000		35,715	55
	경기	470,950		28,181	40
	인천	416,050		1,534	15
서울권	서울권 합계	2,137,018		24,335	14
	경기도	3,879		75	1
	서울	2,133,140		24,260	13
전라	전라 합계	21,547		1,553	2
	광주	1,671		467	2
	전북	19,876		1,086	3
충청	충청 합계	509,009		14,305	16
	충남	13,390		1,180	7
	충북	495,619		13,125	9
영남	영남 합계	81,533		5,248	7
	울산	434,106		5,676	4

그림 6. 제시된 방법을 적용한 지역별 지원업체 리포트

Fig 6. Regionally Classified Asking Companies Report for iCON's Support Using Unit Column Method

다음은 다중 차원을 적용할 때 발생하는 문제점도 제시된 방법으로 해결할 수 있다. 가령 부품·소재통합연구단은 기술별/지역별 두 차원에 대한 요청기업 분포를 확인하고자 할 때 기존 계산된 구성원을 적용한 방법으로는 문제점이 발생한다. (그림 7)에서 보는 것처럼 기존에 계산된 구성원은 기술별 혹은 지역별 지원업체 하나의 차원에 MDX식이

적용되었기 때문에 두 차원이 고려된 분포를 확인할 수 없다. 또한 하나의 계산된 구성원에 다중 차원을 적용하는 MDX식은 복잡할 뿐 아니라 수식을 완성하기도 어렵다.

Area	Tech Group	Sales	Rd Invest	region별지출액	tech그룹당액
A. TechSupport	AC RegionSupport	4,967,311	63,571	128	128
	기술지원	872,736	1,853	36	128
	개발	693,600	26,715	53	128
	지원	2,197,018	24,302	14	128
	지원	21,547	1,553	5	128
	개발	530,600	14,303	19	128
	AC RegionSupport	1,976,929	33,614	170	128
	기술지원	782,008	7,418	36	78
	개발	671,658	13,776	55	78
	지원	4,937	573	14	78
기술지원	AC RegionSupport	7,059	297	5	78
	개발	186,249	23,664	129	37
	기술지원	12,517	247	36	37
	개발	178,719	15,851	55	37
	지원	25,233	802	14	37
	지원	1,158	414	5	37
	개발	6,565	6,362	18	37
	AC RegionSupport	2,205,130	28,292	128	12
	개발	8,151	17,776	36	12
	지원	79,553	2,676	63	12
전자사업	AC RegionSupport	2,103,179	22,962	14	12
	개발	14,229	842	5	12
	지원	3,674	411	11	12

IV. 결론

DW 환경에 다차원 분석을 이용하는 OLAP시스템은 사용자가 데이터를 여러 관점에서 다양한 방식으로 비교 분석하여 필요한 데이터를 업무 분야별로 구조화함으로써 분석 항목의 범위를 명확히 하고, 경영자에게 전략적인 방향을 제시해 줄 수 있다. 그러나 기존 방식에서 분석을 위한 OLAP의 설계 및 구축 단계에서 사용자의 잦은 요구사항의 변경에 대해 별도의 계산된 구성원을 생성해야 하는 문제점과 2개 이상의 차원을 적용하게 되면 MDX식이 복잡해 질 뿐만 아니라 계산된 구성원의 모델링과 구현이 반복되어 전체 시간과 비용이 증가하는 비효율성이 생기게 된다. 기존 방식의 문제점은 OLAP에서의 수식이 열(column)들의 조합으로 생성되어 있기 때문이다. 이러한 근본적인 문제를 해결하기 위해 본 연구에서는 실제 사례를 바탕으로 사용자의 정확한 요구사항을 도출하기 위해 큐브 생성단계에서 행(row)을 집계하여 데이터 분석을 보다 효율적으로 하기 위해 DW의 사실테이블을 새롭게 재구성한 형태의 OLAP을 제안하였다. 제안된 OLAP 구축을 통해 사용자의 요구사항을 보다 더 정확히 파악할 수 있게 되고, 데이터에 대한 정확한 이해와 파악이 가능함을 보였다. 결과적으로 재구성된 사실테이블을 바탕으로 OLAP를 사용함으로써 다차원 분석에 필요한 시간과 비용을 줄일 수 있게 된다.

이러한 점에서 제안하는 모델은 대량의 데이터에 대해 다수의 집계 연산이 처리되는 환경에서 효율적으로 동작할 것으로 예상된다. 제안한 집계 처리 모델은 OLAP에서의 집계 연산 처리 분야, 예를 들어 데이터 큐브 계산에서 유용하게 사용될 우수한 정형적인 기초를 제공할 것으로 예상되며, 이 부분에 대한 자세한 내용은 앞으로의 연구과제가 되리라 생각한다.

그림 7. 지역별/기술별 지원업체 리포트
Fig 7. Regionally and Technically Classified Asking Companies Report for iCON's Support

이와 같은 문제점(그림 8)에서 보는 것처럼 리포트의 행과 열에 지역별 지원업체와 기술별 지원업체 차원을 적용하고 페이지(page)차원에 추가한 열을 측정값으로 적용하여 발생한 문제점을 해결할 수 있다.

Area	Tech Group	Sales	Rd Invest	region별지출액	tech그룹당액
A. TechSupport	AC RegionSupport	4,967,311	63,571	128	128
	기술지원	872,736	1,853	36	128
	개발	693,600	26,715	53	128
	지원	2,197,018	24,302	14	128
	지원	21,547	1,553	5	128
	개발	530,600	14,303	19	128
	AC RegionSupport	1,976,929	33,614	170	128
	기술지원	782,008	7,418	36	78
	개발	671,658	13,776	55	78
	지원	4,937	573	14	78
기술지원	AC RegionSupport	7,059	297	5	78
	개발	186,249	23,664	129	37
	기술지원	12,517	247	36	37
	개발	178,719	15,851	55	37
	지원	25,233	802	14	37
	지원	1,158	414	5	37
	개발	6,565	6,362	18	37
	AC RegionSupport	2,205,130	28,292	128	12
	개발	8,151	17,776	36	12
	지원	79,553	2,676	63	12
전자사업	AC RegionSupport	2,103,179	22,962	14	12
	개발	14,229	842	5	12
	지원	3,674	411	11	12

그림 8. 제시된 방법을 적용한 지역별/기술별 지원업체 리포트
Fig 8. Regionally and Technically Classified Asking Companies Report for iCON's Support using Unit Column Method

이처럼 앞에서 제기된 다차원 분석에서 계산된 구성원을 적용했을 때 발생하는 문제점을 DW에 본 연구에서 제시하는 바와 같이 구성을 새롭게 하여 큐브 생성을 함으로써 사용자가 쉽고 편리하게 다차원 분석을 할 수 있다

참고문헌

- [1] 이종호, 주상호, "데이터 웨어하우징의 구축과 발전동향." 공주대학교 생산기술연구소 논문집 제6권, pp. 111-120. 1998
- [2] Inman, W.H., "Building the Data Warehouse," 2nd Edition, John Willy and Sons, 1996.
- [3] Oracle, "Data Warehousing Concept Paper," Oracle Korea, 1997.
- [4] Chaudhuri, S. and Dayal, U., "An Over view of Data Warehousing and OLAP Technology," ACM SIGMOD Record, Vol. 26, No. 1, pp. 65-74, March. 1997.
- [5] Agarwal, S., Agrawal, R., Deshpande, P.M. et al., "On the Computation of Multidimensional Aggregations," In Proc. Int'l Conf. on Very Large Data Bases, pp. 506-521, Mumbai (Bombay), India, September. 1996.
- [6] Kotidis, Y. and Roussopoulos, N., "An Alternative Storage Organization for ROLAP Aggregate Views Based on Cubetrees," In Proc. Int'l Conf. on Management of Data, pp. 249-258. ACM SIGMOD, Seattle, Washington, June 1998.
- [7] Li, J., Rotem, D., and Srivastava, J., "Aggregation Algorithms for Very Large Compressed Data Warehouses," In Proc. Int'l Conf. on Very Large Data Bases, pp. 651-662, Edinburgh, Scotland, UK, September. 1999.
- [8] Zhao, Y., Deshpande, P.M., and Naughton, J F., "An Array-Based Algorithm for Simultaneous Multidimensional Aggregates," In Proc. Int'l Conf. on Management of Data, pp. 159-170, ACM SIGMOD, Tucson, Arizona, June 1997.
- [9] 박인영, 이성재, 조윤희, 박재순, 박길주, "DW 차원정보 변경에 따른 Historical 데이터 관리 방법," 한국통신 통신망연구소 통합망관리 연구팀, 2001.
- [10] 권오주, "contact OLAP Solutions + α " 대림 2001.
- [11] 노태호, 최인수, "주관투영도를 활용한 실세계의 추상화," 한국OA학회 제7권, pp. 20-26, 2002
- [12] 최정우, 최인수, "웹 상에서의 비정형 시스템의 자율적 설계," 한국컴퓨터정보학회 제8권, pp. 64-74, 2003

저자소개



이 덕 군

1985년 경기대학교 경영학과 졸업(학사)
 1988년 숭실대학교 중소기업대학원 졸업(공학석사)
 2004년 숭실대학교 산업·정보시스템공학과 수료(공학박사)
 2001년~현재 부품소재통합연구단 소장
 <관심분야> MIS, 정보시스템 분석·설계, DW, OLAP



오 미 화

2002년 성결대학교 멀티미디어 정보통신학과 졸업(학사)
 2004년 숭실대학교 MIS 졸업(공학석사)
 2004년~현재 숭실대학교 산업·정보시스템공학과 재학(공학박사)
 <관심분야> MIS, 정보시스템 분석·설계, DW, OLAP



조 재 훈

1999년 숭실대학교 산업·정보시스템공학과 졸업(학사)
 2003년~현재 숭실대학교 산업·정보시스템공학과 재학(공학석사)
 <관심분야> MIS, 정보시스템 분석·설계, DW, OLAP, ERP



최 인 수

1985년 서울대학교 산업공학 졸업(공학박사)
 1975년 영국 College of Librarianship, Wales (Diploma) 객원교수
 1990년 日本 東洋大學 經營學部 객원교수
 1980년~현재 숭실대학교 산업·정보시스템공학과 교수
 <관심분야> 객체지향데이터베이스, 정보시스템 분석·설계, ERP, MIS, DW, OLAP