

## OLAP Server에 독립적인 OLAP Client의 구현

임상배\*, 김경창\*, 지원철\*\*

홍익대학교 전자계산학과\*

홍익대학교 산업공학과\*\*

### Abstract

정보기술의 발달과 경영패러다임의 변화로 데이터웨어하우스가 등장했다. 기존의 관계형 데이터베이스를 기반으로 하는 데이터웨어하우스에서는 사용자가 다양하고 복잡한 분석을 효율적으로 수행하기가 매우 어렵다. 따라서 새로운 데이터모델 즉 다차원 모델이 필요하게 되었다. OLAP(Online Analytical Processing)은 이런 다차원적인 기업 데이터를 사용자가 분석하고 Report 할 수 있게 만들고 관리하는 역할을 한다. 이러한 OLAP은 클라이언트/서버 구조를 가지는데 OLAP 제품별로 질의 방법이나 보고방식들이 다르다. 따라서 최종 사용자는 다른 Client 툴을 사용할 경우 새로운 질의 방법과 보고방식을 익혀야 한다. 그리고 커스텀화된 OLAP Client 툴을 사용할 경우 Server가 바뀌면 사용할 수 없게 되는 단점이 있다. 이런 문제의 원인은 OLAP Client가 Server에 의존적이기 때문이다. 본 논문에서는 이러한 문제점을 해결하기 위해서 OLAP의 특성과 OLAP Client가 지녀야 하는 특성을 알아보고 이러한 요구사항에 맞는 OLAP Server에 의존적이지 않은 통합된 OLAP Client를 제시하고 실제 구현하였다.

### 1. 서론

다차원 데이터분석(Multidimensional Data Analysis)라고 불리는 OLAP(Online Analytical Processing)은 그 자체가 의미하는 것과 같이 복잡한 실세계를 다차원을 통하여 분석하고자 하는 새롭고, 향상된 데이터 질의 방법이라고 할 수 있다. 경제적 환경이나 의사결정의 범위가 넓어지고 복잡해짐에 따라 비즈니스 사용자의 관점도 달라지게 되었다. 그들이 원하는 이상적인 데이터의 관점은 OLTP 수준의 상세 데이터에 접근하여 다양한 방

본 연구는 정보통신부 '98 국책기술개발사업의 지원으로 수행되었음

법 및 분석기법을 활용하여 데이터를 탐색하는 것일 것이다. OLAP 기술은 비즈니스 사용자의 이상적인 요구를 현재의 수준에서 충족시킬 수 있는 최적의 방법이며 의사결정을 위한 기반기술로 인식되고 있다. OLAP(Online Analytical Processing)은 1970년 관계형 데이터베이스구조를 제안한 E.F. Codd가 1993년 의사결정시스템에서 관계형 데이터베이스의 한계를 지적하며 OLAP의 개념을 소개하였다.[1] 이러한 OLAP은 구조적으로 크게 두 가지로 나누어 볼 수 있다. OLAP Server와 OLAP Client가 그것이다. 현재 OLAP에서의 Client는 Server에 의존적인 특성을 가지고 있다. 때문에 최종 사용자는 만약 기존의 것과 다른 OLAP Server를 사용할 경우 질의방법과 보고방식이 다르기 때문에 사용자는 새로운 사용방법을 익혀야 한다. 또한 대부분의 OLAP 제품에서 제공하는 OLE 기능을 사용하여 커스텀화된 Client 툴을 사용할 경우 Server에 의존적이기 때문에 Server가 바뀔 경우 사용할 수 없게 된다. 따라서 Server에 의존적이지 않은 OLAP Client가 필요하게 되었다. 본 논문에서는 OLAP의 특성을 알아보고 Client측면에서 갖추어야 할 기능에 대해 알아보고 기본적으로 요구되는 특징을 갖춘 OLAP Server에 의존적이지 않은 OLAP Client의 Interface의 설계와 구현에 대해 알아보도록 한다.

### 2. OLAP의 특성

OLAP은 최종 사용자에게 다음과 같은 분석적, 향해적인 작업을 지원하는 통합된 기업 데이터에 대한 동적인 다차원 분석의 기능적 특성을 지닌다.

- 차원계층을 통해 전 차원에 걸친 계산과 모델링의 적용
- 시계열 경향분석
- 슬라이스 & 다이스
- 고수준에서 저수준으로의 드릴 다운

- 새로운 관점에서의 차원비교를 위한 회전

이러한 특성을 지닌 OLAP에 대한 여러 가지 규칙이 있다. 1993년 E.F.Codd는 OLAP 시스템이 따라야 하는 12가지 규칙을 제시하였다. Gartner Group은 여기에 9가지의 규칙을 추가하였고 IRI Software는 보완적인 규칙 3가지를 덧붙였다.[2]

이러한 규칙에 기반한 OLAP은 데이터베이스의 크기와 복잡도에 관계없이 일관적으로 질의에 대한 빠른 응답을 제공한다. 다중사용자, 클라이언트/서버 환경으로 구현되며 OLAP은 다양한 "What-if" 데이터 모델 시나리오에서 주제 중심으로 계획된 역사성을 띤 데이터에 대한 분석뿐만 아니라 비교성을 지닌 개인화 된 뷰를 통해 기업정보를 사용자가 종합하는 것을 도와준다. 이러한 것은 OLAP Server의 사용으로 달성될 수 있다.

## 2.1 OLAP Server

다차원 자료구조를 운용하고 지원하도록 특별히 고안된 고성능의 다중사용자의 자료처리 엔진이다. Server의 디자인과 자료구조는 기존의 관계형 데이터베이스의 정형적인 관계 기반의 원시자료의 변환과 빠르고 쉬운 계산뿐만 아니라 어느 방향에서의 비정형 정보 탐색도 신속하게 할 수 있도록 최적화 되어 있다.

## 2.2 OLAP Client

최종 사용자 응용프로그램으로 OLAP Server에 슬라이스들을 요청할 수 있고 최종 사용자가 질의를 수정하거나 선택하는 것, 2차원이나 다차원 형태의 결과보고 및 보고 내용에 대한 랭킹, 계산 등의 기능을 제공한다.

### 2.2.1 OLAP Client의 기본 기능

OLAP Client에서 기본적으로 제공해야 하는 기능은 다음과 같다.[3]

#### 1) 다차원 관점 제공(Multidimensional view)

데이터를 논리적인 다차원으로 구성하고 각 질의 결과를 크로스 테이블 형태로 보여줌으로써 복잡한 데이터 관계를 쉽게 이해하고 분석할 수 있게 해준다.

#### 2) 드릴 다운(Drill down)

차원끼리의 계층구조나 차원 내부의 계층구조를 따라서, 집계 데이터로부터 좀 더 상세한 수준의 데이터로 질의를 이동시키는 것을 의미한다.

#### 3) 롤 업(Roll up)

상세한 수준의 데이터로부터 상위수준의 집계 데이터로 질의를 이동시키는 것을 의미한다.

#### 4) 슬라이싱(Slicing)/다이싱(Dicing)

N개의 차원으로 이루어진 다차원 데이터에서 하나의 slice란,(N-1)이나 또는 그 미만의 차원으로 이루어진 부분집합을 말한다. 또한 dice란, 각 차원에 제한 조건을 줌으로써 얻어진 다차원 데이터의 부분집합을 의미한다.

#### 5) 피보팅(Pivoting)

질의를 통해 생성된 하나의 뷰에서 차원의 위치를 서로 바꾸어줌으로써, 관점이 다른 여러 개의 다른 뷰를 생성할 수 있는 기능이다.

이외에도 SUM, Top N 등과 같은 의사결정을 지원하기 위한 여러 가지 기능들이 갖추어져야 한다.[4]

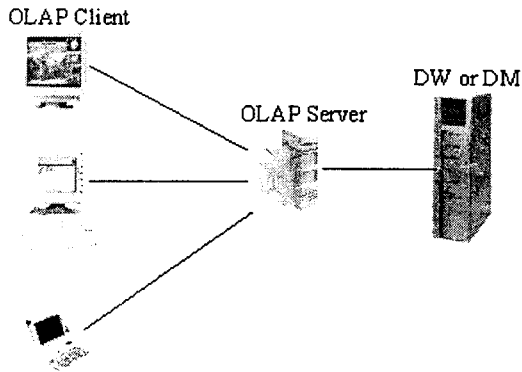
## 3. 독립적 OLAP Client의 필요성

현재의 OLAP 제품들의 Client를 보면 비슷한 기능을 제공하지만 그 사용방법에 있어서는 상이한 면을 보인다. 질의하는 방법이나 결과를 보고서양식으로 보는 방법, 기타 기능을 적용하는 방법들이 제품마다 다르므로 최종사용자가 사용할 경우 불편함을 느낄 수 있다. 또한 OLE기능을 사용하여 커스텀화된 Client 툴을 사용할 때도 단지 하나의 Server만을 지원하므로 문제점이 발생한다. 예를 들면 OLAP Server로 Informix의 Metacube를 사용한다면 Client 툴로 Metacube explorer를 사용해야 하고 MicroSoft의 Plato를 사용한다면 Microsoft의 Client 툴을 사용해야 한다. 이러한 문제점들은 모두 Client 툴이 Server에 의존적이기 때문이다. 따라서 여러 벤더들의 서버를 지원 할 수 있는 통합된 독립적인 Client 툴이 필요하다. 통합된 OLAP Client 툴이 있다면 OLAP Server에 상관없이 사용자는 동일한 GUI에서 질의를 던지고 항상 같은 방법으로 질의처리 결과에 대한 보고서를 작성하고 여러 가지 기능을 수행하게 할 수 있게 된다.

### 3.1 독립적 OLAP Client 툴의 설계

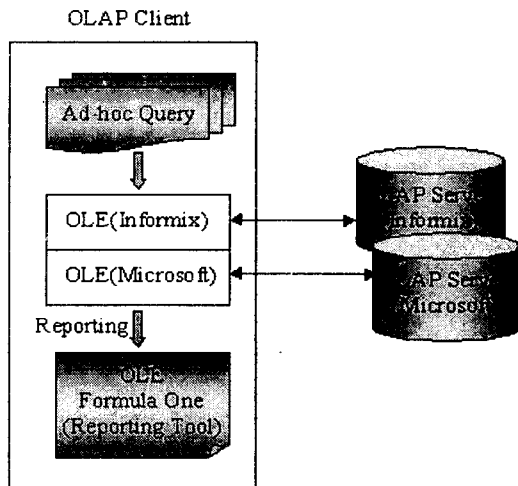
기본적인 OLAP의 구조는 <그림.1>과 같다. 물론 이런 구조가 절대적인 것은 아니다. OLAP Server가 Client쪽에 있는 2-tier 구조 일 수도 있고 Client와 DW(or DM)의 중간에 위치하는 3-tier 구조 일 수도 있다. 독립적인 툴을 만들기 위한 방법으로는 본 연구에서는 먼저 Client 툴에서 기본적으로 제공되어야 하는 기능들에 대한 최종 사용자

인터페이스를 만들고 OLAP 제품에서 제공되는 OLE기능을 사용하여 인터페이스에 맞게 기능들을 프로그래밍 한 후 각각의 OLAP Server에 질의를 던지고 그 수행결과를 Formula One이라는 리포팅 툴을 사용하여 질의처리결과를 표시하는 방법을 사용하였다.



<그림.1> OLAP의 구조

본 연구에서는 Informix와 Microsoft의 Server에 사용할 수 있는 Client 툴을 목표로 하고 있는 현재는 Informix를 지원하는 것까지 구현되어있다. Client 툴의 구조는 <그림.2>와 같다.

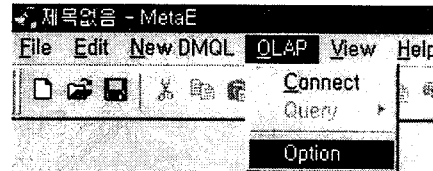


<그림.2> 통합된 OLAP Client의 구조

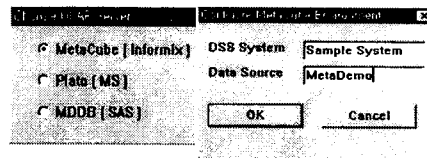
<그림.2>에서 알 수 있듯이 OLAP Client는 사용자 인터페이스를 통해 비정형 질의를 입력받고 이런 질의를 어떤 OLAP Server에 던질 것인지는 사용자가 이미 설정해 놓은 OLAP Server의 종류에 따라 결정된다. 해당 OLAP Server에서 질의 결과가 돌아오면 Formula One이라는 리포팅 툴을 사용하여 결과보고를 하게 된다.

### 3.2 독립적 OLAP Client 툴의 구현

우선 OLAP Client에서 어떤 데이터베이스에 접속할 지를 설정해야 한다. 접속할 데이터베이스 소스는 윈도우즈의 "32비트 ODBC" 통해 설정한다. Client 툴을 실행시키면 <그림.3>과 같은 메뉴가 보인다. Option 부분을 선택하면 <그림.4>이 나타나며 사용자의 OLAP Server를 선택하고 설정하게 된다



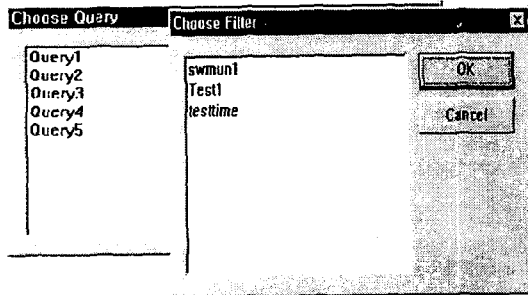
<그림.3> OLAP Server설정 메뉴



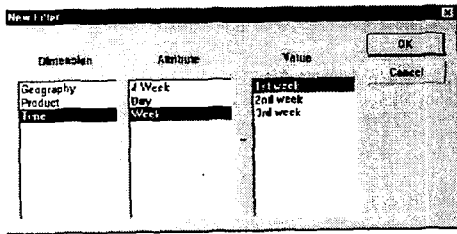
<그림.4> OLAP Server 설정

그 다음 메뉴의 Connect를 선택하게 되면 선택한 Data source에 대해 사용자 인증을 한 후 접속을 하게 된다. 접속이 되면 Query라는 메뉴가 활성화 되는데 사용자가 자주 사용하는 질의는 저장해놓고 사용할 필요가 있으므로 저장된 질의를 사용하는 경우와 새로운 질의를 만드는 경우 두 가지를 두었다. 저장해 놓은 질의를 선택하면 데이터베이스에 저장된 질의들의 목록이 나타난다. 질의를 선택하면 질의에 필터를 적용할지를 선택하게 된다. 필터는 검색될 데이터의 범위를 제한하는 역할을 한다. 이런 필터는 최종 사용자로 하여금 좀 더 세밀한 질의를 만들 수 있게 된다. 거의 대부분의 필터는 SQL문의 where 절에 해당한다. 그리고 필터 또한 자주 사용하는 필터는 저장해 놓을 필요가 있다. 예를 들면 "A지역의 B품목의 판매실적은 얼마나?"라는 질의와 "A지역의 C매장의 판매실적은 얼마나?"라는 질의를 차례대로 실행한다면 그리고 최종 사용자가 두 질의의 결과 중 1998년이라는 특정연도의 결과를 원한다면 두 질의에 1998년이라는 필터를 적용하면 된다. 그렇다면 1998년이라는 필터를 만들어 저장해놓고 사용한다면 1998년에 해당하는 결과를 보고 싶을 때 다시 필터를 만드는 수고를 덜게된다. <그림.5>에 이와 같은 과정들이 나타나 있다. 사용자가 저장되어 있는 질의에 이미 만들어 놓은 필터가 아닌 새로운 필터를 만들고 이를 질의에 적용해야하는 경우도 있다. 만약 최종 사용자가 "A지역의 B품목에 대한 판매실적은 얼마인

가?"라는 질의를 만들어 이를 저장했다고 가정해 보자. 만약 최종 사용자가 "A지역 B품목의 10월에 판매실적이 얼마인가?" 라는 질의를 하고 싶다면 이미 저장되어 있는 질의를 선택하고 10월에 해당하는 필터를 만들고 이를 적용하면 되는 것이다. 실행과정은 <그림.6>과 같다.

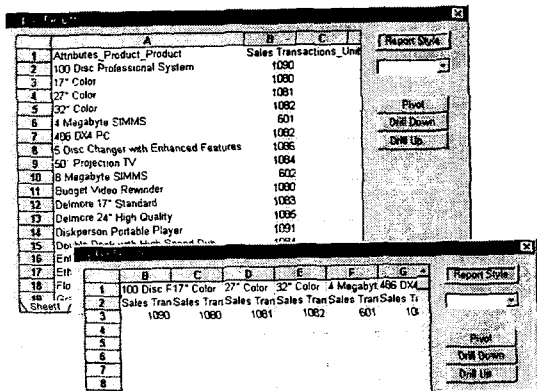


<그림.5> 질의 선택하고 저장되어 있는 필터 적용



<그림.6> 필터를 만들어 적용

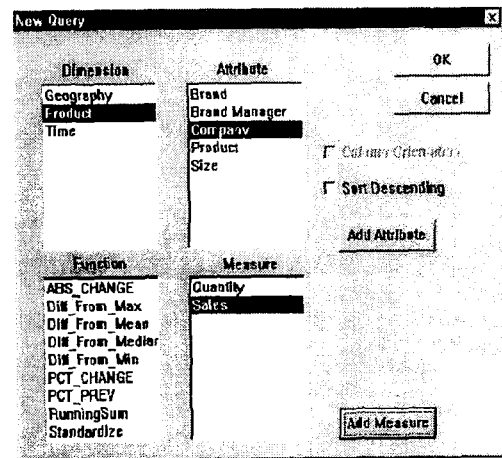
이와 같은 과정을 마치면 질의에 해당하는 결과가 리포팅 틀인 Fomular One을 사용하여 <그림.7>과 같이 표시된다. Pivot 버튼을 누르면 질의결과를 피보팅 하여 보여준다. Pivot 기능은 최종 사용자가 질의에 대한 결과를 다양한 관점에서 분석할 수 있도록 해준다.



<그림.7> 질의결과보고와 피보팅

새로 질의를 만들어서 수행하는 경우는 <그림.8>과 같이 각 차원에서 질의하고 싶은 속성들을 선택한 다음 Measure를 선택한다. Measure는 트랜잭션을 평가하는 방법으로 여기에도 필터를 적용할 수

있다. 적용방법은 저장된 질의를 수행하는 것과 동일하다. 질의 결과를 보여주는 것도 같은 방식을 취한다. Formula One의 다양한 기능을 사용하여 리포팅 양식을 바꿀 수도 있다. 물론 도표를 만들거나 차트화 하는 것도 가능하다. 현재는 단순한 여러 가지의 리포팅 양식을 만들 수 있는 단계이다.



<그림.8> 새로운 질의를 정의

#### 4. 결론

본 연구에서는 OLAP Server에 독립적인 통합된 OLAP Client를 설계하고 구현하였다. OLE 인터페이스를 통하여 Informix의 Metacube에 질의를 던지고 그 결과를 얻어 올 수 있었다. 이러한 방법을 사용한다면 OLE를 지원하는 다른 OLAP Server에도 질의를 수행하고 결과를 얻어 올 수 있다.

#### 5. 참고문헌

- [1].E.F. Codd, S.B. Codd and C.T. Salley, "Providing OLAP to User-Analysis: An Mandete", Arbor Software White Paters, 1995
- [2].Frank A. Buytendijk. OLAP: playing for keeps -- Maintenance and control aspects of OLAP applications Computable, July 1995.
- [3].Kenana Systems corporations, "An Introduction To Multidimensional Database Technology",Kenana Systems Corporation White Paper, 1995
- [4].Red Brick Systems,"Decision-Markers, Business Data and RISQL", A Red Brick White Paper, 1996