

XMLMD: XML 웹서비스 기반의 클라이언트 OLAP API

배은주^U 김 명
이화여자대학교컴퓨터학과 고성능인터넷지식공학연구소
(ejbae, mkim)@ewha.ac.kr

XMLMD: XML Web Services-based Client OLAP API

Eunjoon Bae^U Myung Kim
Dept. of Computer Science & Engineering, Ewha Womans University

요 약

OLAP(On-Line Analytical Processing)은 데이터 웨어하우스에 저장된 정보를 다차원적으로 분석하여 그 결과를 온라인으로 사용자에게 제공하는 제반 기술로써, 비즈니스 인텔리전스의 핵심 기술 중 하나이다. OLAP 애플리케이션을 개발하기 위해서는 OLAP API를 필요로 하며, 주요 API들로는 OLE DB for OLAP, JOLAP, XML for Analysis (XML/A)를 들 수 있다. 이 중에서 XML/A는 XML 웹서비스를 지원하는 API로써 SOAP, XML, HTTP 등 개방형 인터넷 표준을 따른다. XML/A는 데이터를 XML 형태로 전송함으로써 서로 다른 플랫폼을 갖는 클라이언트와 서버간에 통신을 할 수 있도록 해 준다. 반면, OLAP 오브젝트나 분석 데이터를 서버에 요청하기 위해서 클라이언트는 동일한 형식의 XML 데이터를 매년 생성해야 하는 번거로움을 안고 있다. 본 연구에서는 윈도우즈 환경의 애플리케이션 개발자들이 이러한 번거로운 작업을 하지 않고 XML/A Provider의 웹서비스를 통해 편리한 다차원 데이터 분석을 지원하는 XMLMD(XML Multidimensional) API를 설계하고 구현하였다.

1. 서 론

OLAP은 온라인 상에서 다차원 데이터를 분석하는 기술로써, 정제된 비즈니스 데이터의 다차원적 집계 뷰(view)를 제공하는 역할을 한다. OLAP 애플리케이션의 예를 들면, 특정 유통업체의 제품별, 매장별 판매총액을 분석하여 그 결과를 분석가에게 제공하는 것을 들 수 있다. 이러한 애플리케이션은 최근들어 클라이언트/서버 환경 뿐 아니라 웹기반으로도 개발되고 있다. 웹 OLAP 애플리케이션은 웹 브라우저 이외에 별도의 소프트웨어를 필요로 하지 않아 사용이 간편하다.

웹 OLAP 애플리케이션은 다음과 같이 여러 형태로 개발되고 있다. 분석될 데이터를 HTML 파일로 미리 만드는 방법과 CGI형태로 사용자 쿼리에 의해 HTML 파일을 동적으로 만드는 방법이 있으며, 이외에 자바를 이용해 애플리케이션을 작성하는 방법이 있다[1]. 이와 같은 웹 OLAP 애플리케이션을 개발하려면 웹을 지원하는 OLAP API를 사용하게 된다.

OLAP API의 역할은 OLAP 서버와 클라이언트 사이의 연결 관리, OLAP 명령문 수행, OLAP 데이터 조회 등의 기능을 하는 메소드를 클라이언트 애플리케이션에게 제공하는 것이다. 대표적인 OLAP API의 종류로는 Microsoft사의 OLE DB for OLAP과 Oracle사의 JOLAP (Java-based OLAP API) 등이 있으며, 이 중에서 OLE DB for OLAP이 de facto 표준으로 자리잡음으로써[2] 많은 OLAP 제품들은 OLE DB for OLAP을 현재 지원하고 있다. 최근에는 OLE DB for OLAP의 디자인 개념을 재사용하면서 이기종간의 데이터 공유와 통신을 위한 XML 웹서비스를 지원하는 SOAP(Simple Object Access Protocol) 기반의 XML 통신 API인 XML for Analysis (XML/A)가 발표되었다.

* 본 연구는 한국과학재단 목적기초연구 (R04-2001-000-00191-0) 지원으로 수행되었음.

XML/A를 이용한 클라이언트 애플리케이션은 XML 소스를 만들어 전송함으로써 XML/A Provider에 분석 데이터를 요청한다. 전송하는 XML 소스는 SOAP 헤더와 호출 메소드, 매개변수와 속성 등으로 이루어진 긴 문장이다. 클라이언트 애플리케이션은 분석 데이터를 서버에 요청할 때마다 이러한 XML 소스를 생성해야 하는 번거로움을 안고 있다. 본 연구에서는 윈도우즈 환경의 클라이언트 애플리케이션 개발자들이 이러한 반복적인 작업을 하지 않고 다차원 데이터에 편리하게 접근할 수 있는 애플리케이션 수준의 인터페이스인 XMLMD(XML Multidimensional API)를 설계하고 구현하였다.

본 논문의 구성은 다음과 같다. 2절에서는 OLAP과 OLAP API에 대하여 기술하고, 3절에서는 XMLMD의 근간이 되는 XML/A, 4절에서는 본 논문에서 제안하는 XMLMD의 구조와 구현에 대하여 기술한다. 5절에서는 XMLMD를 이용한 웹 애플리케이션을 소개하며 6절에서 결론 및 향후과제를 설명하기로 한다.

2. OLAP과 OLAP API

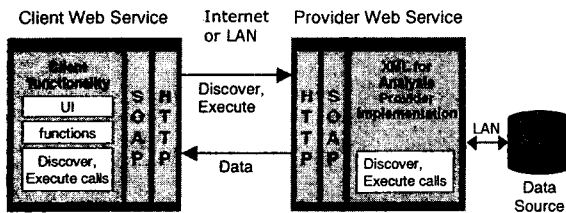
데이터 웨어하우스에 저장된 데이터는 다수의 차원정보(제품, 매장, 기간 등)와 측정값(매출액, 매출수량 등)으로 구성된다. OLAP의 주요 역할은 이러한 데이터를 다차원적으로 집계 연산하여 그 결과를 사용자에게 제공하는 것이다. 온라인 서비스를 위해 대부분의 OLAP 서버들은 부분적 집계 결과들을 사전에 계산하여 저장해 두는데 이를 큐브(cube)라고 부른다. 사용자는 큐브의 일부분을 슬라이스 하여(예제: 냉장고 제품의 기간별 매출액) 볼 수도 있고, 큐브의 일부를 떼어내는 디스 연산을 통해 데이터 분석을 할 수도 있다. 또한 각 차원은 계층구조를 가질 수 있으며 사용자는 계층구조의 레벨을 따라 내려가는 드릴다운(전체제품=>가전제품=>각 가전제품) 연산이나 반대 방향의 톨업 연산을 통해 데이터를 분석하게 된다[3].

OLAP API는 생성된 큐브 상에서 슬라이스, 다이스, 드릴다운, 롤업 기능을 하는 메소드를 OLAP 애플리케이션에게 제공한다. 예를 들어, OLE DB for OLAP은 SQL과 유사한 MDX (Multidimensional Expressions) 언어를 제공한다. MDX를 사용하면 Select 절과 From 절을 통해 분석하려는 큐브와 차원을 선택할 수 있고, Where 절을 써서 큐브의 단면을 슬라이스할 수 있다[4]. OLAP API는 이외에도 OLAP 서버와 클라이언트의 연결을 관리하고, OLAP 오브젝트를 검색할 수 있는 기능의 메소드를 클라이언트 애플리케이션에 제공한다.

3. XML for Analysis (XML/A)

XML/A는 웹을 통해 데이터 공급자와 클라이언트 애플리케이션 간의 데이터 액세스 상호 작용을 표준화하기 위해 설계된 OLAP API로서 인터넷 또는 인터넷 상에서 다차원 데이터를 보편적으로 사용할 수 있는 표준 데이터 액세스 API를 제공한다[5]. XML/A는 SOAP, XML, HTTP 등의 개방형 인터넷 표준과 함께 틀, 프로그래밍 언어, 하드웨어 플랫폼, 장치 등에 독립적인 구현을 지원하며, de facto 표준 OLAP API인 OLE DB for OLAP의 디자인 개념을 재사용 하고 있다.

XML/A의 구조는 [그림 1]과 같이 표현될 수 있다. 클라이언트는 SOAP과 HTTP 프로토콜을 사용하여 서버의 웹 서비스 URL과 함께 Discover, Execute 메소드를 호출한다. 서버는 XML/A Provider 인스턴스를 만들어서 Discover 및 Execute 메소드를 처리하고, 요청받은 데이터를 XML로 패키징화 한 후에 클라이언트로 보낸다.



[그림 1] XML/A의 구조

XML/A는 Discover와 Execute 2개의 메소드를 제공한다. Discover 메소드는 데이터 원본의 목록 또는 특정 데이터 원본에 대한 큐브, 차원, 측정값, 레벨 등의 OLAP 오브젝트들을 검색하는데 쓰인다. 검색할 데이터는 메소드의 매개변수를 사용하여 선택한다. Execute 메소드는 MDX와 같은 OLAP 명령문을 사용하여 큐브를 슬라이스, 다이스, 드릴다운, 롤업하여 다차원 데이터를 검색할 때 쓰인다.

[그림 2]는 Discover와 Execute의 구문이다. RequestType은 큐브, 차원과 같은 검색할 OLAP 오브젝트의 형태를 지정하고, Restrictions는 검색 대상에 대한 특정 제한조건을 나타낸다. Properties는 데이터 원본의 정보나 디스플레이 형식 등의 속성을 할당하고, Command는 MDX OLAP 명령문이다. 수행된 결과는 Result 항목에 XML 형태로 반환된다.

```

Discover (
  [in] RequestType As EnumString,
  [in] Restrictions As Restrictions,
  [in] Properties As Properties,
  [out] Result As Rowset)

Execute (
  [in] Command As Command,
  [in] Properties As Properties,
  [out] Result As Resultset)
    
```

[그림 2] Discover, Execute 메소드 구문

XML/A의 사용 예를 살펴보자. "FoodMart 2000" 카탈로그의 "Sales" 큐브에 속한 차원 종류를 알아보려면 [그림 3]과 같이

Discover 메소드를 사용하여 XML 소스를 작성한 후에 이를 XML/A Provider에 전송하면 된다.

```

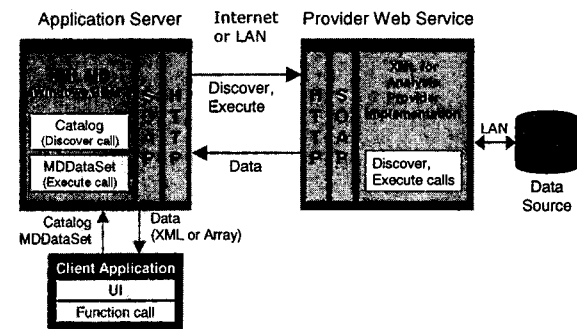
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <SOAP-ENV:Body>
    <Discover xmlns="urn:schemas-microsoft-com:xml-analysis" SOAP-
      Env:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      <RequestType>MDSHEMA_DIMENSTIONS</RequestType>
      <Restrictions>
        <RestrictionList>
          <CatalogName>FoodMart 2000</CatalogName>
          <CubeName>Sales</CubeName>
        </RestrictionList>
      </Restrictions>
      <Properties>
        <PropertyList>
          <DataSourceInfo>Provider=MSOLAP;Data Source=local</DataSourceInfo>
          <Format>Tabular</Format>
          <Content>SchemaData</Content>
        </PropertyList>
      </Properties>
    </Discover>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
    
```

[그림 3] XML/A의 Discover 메소드 소스

[그림 3]과 같이 XML 소스의 앞 부분에는 SOAP 헤더가 있고, OLAP 오브젝트인 "차원" 정보를 검색하기 위한 Discover 메소드는 박스 안에 나타나 있다. 예제에 대한 Discover 매개변수 값으로, RequestType 변수에 "MDSHEMA_DIMENSTIONS"을 지정하고, Restrictions 변수에는 <CATALOG_NAME>에 "FoodMart 2000", <CUBE_NAME>에 "Sales"를 지정한다. Properties 변수에는 데이터 원본 정보와 반환데이터 형식의 값을 지정한다. XML 소스를 생성한 후 XML/A Provider에 요청을 하면 서버는 "Sales" 큐브의 차원정보 목록을 테이블 형식의 XML로 만들어 클라이언트에게 제공한다.

4. XMLMD (XML Multidimensional)

이제 본 연구에서 제안하는 XMLMD를 소개하기로 한다. XMLMD는 윈도우즈 플랫폼에서 XML/A Provider를 통해 다차원 데이터를 쉽게 접근할 수 있는 클라이언트 OLAP API이다. XML/A는 XML 웹 서비스를 기반으로 하기 때문에, 다양한 하드웨어 플랫폼과 언어를 지원하는 장점을 갖고 있다. 반면 특정 플랫폼에 익숙한 클라이언트 애플리케이션 개발자들한테는 데이터와 서비스에 관한 구체적인 내용을 포함하는 XML 소스를 생성해야 하는 불편함을 준다. XMLMD는 이러한 불편함을 최소화하면서 클라이언트가 전송할 XML 코드를 조건에 맞게 자동으로 생성하고, 반환된 XML 결과물을 배열변수에 할당하여 데이터 항목을 쉽게 접근할 수 있도록 설계되고 구현되었다.



[그림 4] XMLMD 구조

[그림 4]는 XMLMD의 구조이다. 클라이언트에서 XML로 작성하여 데이터를 교환하는 부분을 COM+ 환경의 Catalog, MDDataset 오브젝트로 캡슐화하였다. 이들을 사용하여 클라이언트는 VB 형식의 메소드 호출을 통해 XML 또는 배열 형태로 원하는 데이터를 쉽게 얻을 수 있다. XMLMD를 사용하면 배열변수로 반환되는 결과 항목을 제어하기 쉬울 뿐 아니라, XML/A Provider를 통해 교환하는 XML 데이터 구조를 모르더라도 쉽게 OLAP 애플리케이션을 개발할 수 있다.

XMLMD는 Catalog, MDDataset 두 개의 오브젝트로 구성된다. Catalog 오브젝트는 데이터베이스 Catalog, 큐브, 차원, 측정값, 레벨, 멤버 등의 OLAP 오브젝트를 조회할 때 사용하고, MDDataset 오브젝트는 MDX문을 수행하여 그에 해당하는 큐브 데이터를 조회할 때 사용한다. Catalog 오브젝트는 DBServer, Format, Content의 3개의 속성과 Discover(), GetCatalogs(), GetCubes(), GetDimensions(), GetMeasures(), GetMembers(), GetLevels()의 7개의 메소드로 구성되어 있고, MDDataset 오브젝트는 DBServer, Format, Content, AxisFormat의 4개의 속성과 Execute(), GetDataSet()의 2개의 메소드로 구성되어 있다.

OLAP 오브젝트의 구조는 다음과 같다. DB 카타로그는 여러 개의 큐브를 갖고, 큐브는 다수의 차원과 측정값으로 구성되며, 차원은 계층구조와 레벨, 멤버들을 갖는다. Catalog 오브젝트의 각 메소드들은 이 구조에 따른 매개변수를 가지며, MDDataset 오브젝트는 기본적으로 행과 열의 2차원 형태로 큐브 데이터를 반환하도록 설계되었다. [그림 5]와 [그림 6]은 Catalog와 MDDataset 오브젝트의 속성과 메소드 목록이다.

Catalog: OLAP 오브젝트를 검색
DBServer: DB 서버의 위치 (default: local)
Format: 반환되는 결과집합의 형식 (default: Tabular)
Content: 결과로 반환되는 데이터의 형태 (default: SchemaData)
Discover(): XML/A의 Discover 메소드 수행
 다양한 카타로그 큐브 차원 측정값 멤버 레벨을 가져오는 메소드이다.
GetCatalogs(ByRef Catalogs)
GetCubes(ByRef Cubes, ByRef CubeType, ByRef CubeDesc, Catalog)
GetDimensions(ByRef Dimensions, ByRef DimUName, ByRef DimCaption, Catalog, Cube)
GetMeasures(ByRef Measures, ByRef MeaUName, ByRef MeaCaption, Catalog, Cube)
GetMembers(ByRef Members, ByRef LevelNumber, ByRef LevelUName, ByRef MemberUName, ByRef ParentUName, ByRef MemberCaption, Catalog, Cube, Dimension, Optional LevelNum)
GetLevels(ByRef Levels, ByRef LevelNumber, ByRef LevelUName, ByRef LevelCaption, ByRef LevelCardinality, Catalog, Cube, Dimension)

[그림 5] Catalog 오브젝트의 속성과 메소드

MDDataset: OLAP 큐브 데이터를 검색
DBServer: DB 서버의 위치 (default: local)
Format: 반환되는 결과집합의 형식 (default: Multidimensional)
Content: 결과로 반환되는 데이터의 형태 (default: Data)
AxisFormat: 축의 형식 (default: TupleFormat)
Execute(): XML/A의 Execute 메소드 수행
GetDataSet(Command, Catalog, ByRef Axis0Caption, ByRef Axis0LNum, ByRef Axis1Caption, ByRef Axis1LNum, ByRef CellValue, ByRef CellFmtValue): MDX문의 수행결과를 반환

[그림 6] MDDataset 오브젝트의 속성과 메소드

XMLMD를 사용하여 OLAP 애플리케이션을 개발하는 예제를 살펴보자. 3절에서 사용한 예제를 그대로 사용하여, FoodMart 2000 카탈로그의 Sales 큐브에 속한 차원을 검색해 보기로 한다. 이 때 클라이언트 애플리케이션은 [그림 7]과 같이 코딩할 수 있으며 이는 [그림 3]의 XML코드와 비교해 보면 상당히 간단하다는 것을 알 수 있다. 큐브 차원 정보를 얻기 위해 Catalog 오브젝트의 GetDimensions 메소드가 호출되었고, 이 메소드는 내부적으로 XML 소스를 생성해 XML/A Provider에게 해당 데이터를 요청을 하고, 그 결과를 XML과 배열 형태로 반환한다.

XMLMD를 사용하여 큐브 데이터를 얻는 예제는 [그림 8]에 있다. MDDataset 오브젝트를 생성한 후 GetDataSet 메소드를 호출하면 된다.

```
Set objCat = Server.CreateObject("XMLMD.Catalog")
xmlResult = objCat.GetDimensions(arrDimensions, arrDimUName, arrDimCaption, "FoodMart 2000", "Sales")
```

[그림 7] XMLMD의 Catalog 오브젝트 소스

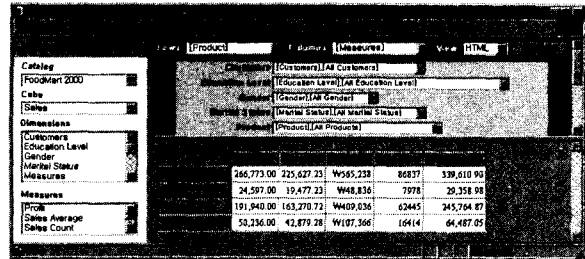
```
strCommand = "SELECT {[Measures].members } ON COLUMNS, {DrilldownLevelTop([Account].[All]].members , 3) } ON ROWS FROM [Budget] WHERE ([Category].[All Category],[Store].[All Stores],[Time].[1997])"
```

```
Set objCat = Server.CreateObject("XMLMD.MDDataset")
xmlResult = objCat.GetDataSet(strCommand, "FoodMart 2000", Axis0Caption, Axis0LNum, Axis1Caption, Axis1LNum, CellValue, CellFmtValue)
```

[그림 8] XMLMD의 MDDataset 오브젝트 소스

5. XMLMD를 이용한 웹 OLAP 애플리케이션

XMLMD를 이용하면 웹 OLAP 애플리케이션을 쉽게 만들 수 있는데, 그 예제로 [그림 9]와 같은 웹 기반 큐브 브라우저를 만들 수 있다. 왼쪽 프레임에서 DB 카탈로그와 큐브를 선택하고 오른쪽 상위 프레임에서 행과 열 차원을 선택한 후 실행 버튼을 누르면 큐브에서 슬라이스 연산을 실행한 후에 그 결과를 XML 또는 HTML 형식으로 보여준다. 이 애플리케이션의 모든 동작은 XMLMD가 제공하는 메소드들의 수행으로 이루어진다.



[그림 9] 웹 기반의 큐브 브라우저

6. 결론 및 향후 연구 과제

본 연구에서는 웹서비스 기반의 클라이언트 OLAP API인 "XMLMD"를 제안하고 구현하였다. 이는 윈도우즈 환경의 플랫폼에서 XML/A Provider와 통신하여 다차원 데이터 분석을 편리하게 해 주는 역할을 한다.

향후 연구 과제로는 큐브 데이터를 검색하기 위한 MDX문을 자동으로 생성하여 주는 메소드와 큐브, 차원 등의 OLAP 오브젝트를 생성, 수정, 삭제할 수 있는 메소드를 추가하여 XMLMD를 확장하는 것이다.

7. 참고문헌

- [1] The Data Administration Newsletter, "THE WAREHOUSE MEETS THE WEB", <http://www.tdan.com/i004fe04.htm>, 1999.
- [2] The OLAP Report, <http://www.olapreport.com>
- [3] Pilot White Paper, "An introduction to OLAP, Multidimensional Terminology and Technology", <http://www.pilotsw.com/olap/olap.htm#dsgl>, 1999.
- [4] MDX (Multidimensional Expressions), http://msdn.microsoft.com/library/default.asp?url=/library/en-us/olapdmd/agmdxbasics_04qg.asp
- [5] XML for Analysis Specification, <http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnxmlspec/html/xmlanalysis.asp>, 2001.