

# ODBMS 상에서 OpenGIS SFCORBA 정보 저장소의 구현

송창빈 김기홍 유승원 차상균

서울대학교 전기공학부

{tsangbi, next, mokmon, chask}@kdb.snu.ac.kr

## Implementation of OpenGIS SFCORBA Interfaces on Top of ODBMS

Changbin Song Kihong Kim Seungwon Yoo Sang K. Cha

School of Electrical Engineering, Seoul National University

### 요약

사회 경제적으로 지리 정보에 대한 관심이 커지고 있으며 인터넷 기술의 보급으로 일반인들도 지리 정보를 쉽게 접근할 수 있게 되었다. OpenGIS는 지리 정보를 표현하고 공유하는데 필요한 데이터 모델 및 상호 연동을 위한 표준이며 CORBA, OLE/COM, SQL 세가지 환경에 대해 구현 명세기가 있다. 본 논문은 ODBMS를 저장 시스템으로써 사용하고 CORBA를 분산 객체 기술로 활용하여 OpenGIS SFCORBA(Simple Features Specification for CORBA) 인터페이스를 구현하는 방법을 설명한다. 특히 ODBMS와 SFCORBA 사이의 인터페이스를 변환하는 Wrapper의 역할, 자동 생성 방법, 구현 등을 중심으로 기술한다.

## 1 서론

사회 경제적으로 중요성이 부각되고 있는 GIS(Geographic Information System)는 수치 지도 정보를 근간으로 토지, 자원, 시설물, 환경, 통계 등의 관련 정보를 체계적으로 관리하는 것이다. 특히 우리나라의 경우 국토 개발이 서구에 비해 빠른 속도로 진행되었기 때문에 지리 정보의 갱신이 빈번하므로 더욱 관심이 높아지고 있다.

한편 인터넷 기술이 빠른 속도로 일반화 됨에 따라 전문가에게 국한되어 있던 지리 정보 사용 계층이 일반 사용자에게로 확산되고 있다[1]. 점차 넓어지는 사용자 계층과 방대한 지리 데이터를 공유하기 위해 OGC(OpenGIS Consortium, Inc.)는 지리 정보를 표현하고 공유하는데 필요한 데이터 모델 및 상호 연동을 위한 표준인 OpenGIS(Open Geodata Interoperability Specification)[2]를 제정하고 있다. OpenGIS 표준은 지리 정보 소프트웨어를 개발할 때 준수해야 하는 형상 정보, 지리 정보, OpenGIS 서비스, 정보 공동체 등 14개의 주제로 나누어진 추상 명세서를 정의하고 있으며 이를 기존에 쓰이고 있는 분산 기술인 CORBA, OLE/COM, SQL 세가지 환경에 대한 구현 명세서도 공개하였다.[3] 이 중 SQL 환경을 기반으로 OpenGIS 표준을 지원하는 제품들이 속속 발표되고 있으나[4, 5] CORBA와 OLE/COM 기반의 구현은 아직 프로토타입 수준이다.

OpenGIS 표준은 다양한 저장 시스템과 분산 환경에서 구현할 수 있는데 본 논문에서는 ODBMS를 저장 시스템으로 사용하고 CORBA를 기반으로한 SFCORBA(OpenGIS Simple Features Specification for CORBA) 인터페이스의 구현에 대해 연구한다. OpenGIS의 지리 데이터 모델은 객체 지향 패러다임에 기반을 두고 있기 때문에 ODBMS에 객체를 저장하고 CORBA를 통해 객체를 전송하는 것이 가장 효과적인 구현 방법이다. 하지만 OpenGIS 인터페이스는 동적 타입 바인딩을 통해 정보에 접근하는 방법을 취하고 있으나 ODBMS는 컴파일시 타입을 바인

딩하기 때문에 이들 간의 차이를 변환해주는 Wrapper가 필요하다. 본 논문에서는 Wrapper의 구현 방법과 자동 생성 방법에 대해 설명한다.

관련 논문 [6]에서는 OpenGIS 사용자를 위해, 구체적인 데이터 모델과 인터페이스, 클라이언트 코드의 예를 자세히 보이는 대신 SFCORBA의 구현은 거의 설명되지 않았으나 본 논문에서는 구현상의 문제점과 해결 방법을 설명한다.

## 2 프로그래밍 인터페이스

SFCORBA는 추상 명세서를 CORBA 환경으로 매핑한 구현 명세서이며 현재 공개된 1.0 버전에서는 Feature, Geometry, Spatial Reference System이 정의되어 있다. CORBA에서는 IDL(Interface Definition Language)[7]를 통해 서버에서 제공하는 서비스를 기술하는데 SFCORBA에는 모든 인터페이스가 이것으로 정의되어 있다.

Feature 관련 인터페이스로는 Feature, FeatureType, FeatureCollection을 중심으로 Iterator와 Factory 관련 인터페이스들이 정의된다. Feature는 공간 및 비공간 속성과 식별자를 가지는 객체를 지칭하며 다음의 IDL 인터페이스로 정의되어 있다.

```
interface Feature {
    readonly attribute FeatureType feature_type;
    Geometry get_geometry(...);
    any get_property(in string name);
    void set_property(in string name, in any value);
    ...
}
```

FeatureType은 Feature의 메타 정보를 검색하는 인터페이스를 제공하며 FeatureCollection은 Feature들의 참조를 저장 관리한다. FeatureCollection에서 상속받은 ContainerFeatureCollection은 참조 대신 Feature 자체를 저장 관리한다. FeatureFactory, FeatureTypeFactory 등의 Factory는 각각 Feature 및

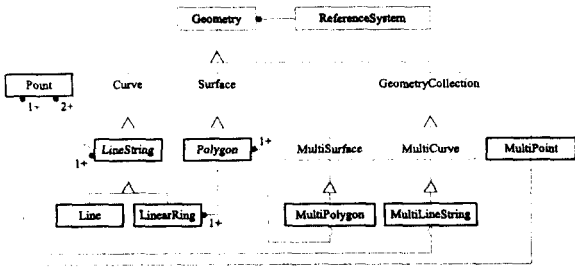


그림 1: Geometry 모델의 계층 구조

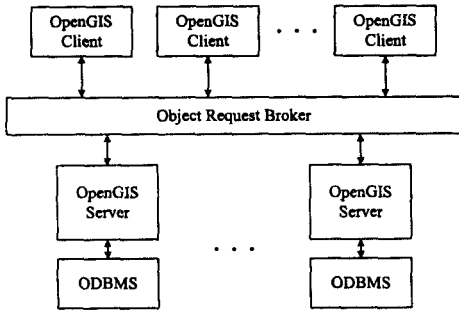


그림 2: OpenGIS 시스템 구조

FeatureType 등을 생성하는 인터페이스를 제공한다. Feature는 ContainerFeatureCollection이나 Query\_Evaluator의 질의 인터페이스를 통해 접근할 수 있다.

그림 1은 Geometry 모델의 계층 구조를 나타낸 것이다. Geometry를 베이스 클래스로 하여 Point, Curve, Surface, GeometryCollection이 정의되며 각각 0, 1, 2차원 및 collection을 나타낸다. LineString, Line, LinearRing이 Curve로부터 상속받아 1차원 정보를 나타내며 마찬가지로 2차원, collection 정보는 Surface, GeometryCollection으로부터 상속받아 정의된다. 그림에서 굵은 선은 실제 객체를 생성할 수 있는 타입임을 나타낸다.

### 3 시스템 구조

SFCORBA 인터페이스는 CORBA IDL로 정의되었기 때문에 OpenGIS 정보 저장소의 구조는 일반적인 CORBA 클라이언트/서버 환경과 비슷하다. 그림 2는 ODBMS에 저장되어 있는 지리 정보에 SFCORBA 인터페이스를 통해 접근할 때의 구조를 나타낸다.

OpenGIS 서버는 SFCORBA에서 정의된 인터페이스를 구현한 CORBA 서버에 해당한다. 서버는 클라이언트의 요구를 특정 ODBMS에 맞게 변환하여 실행한 후 결과를 돌려 주는 역할을 한다. 개별 Feature 인터페이스는 서버에서 생성되는 CORBA의 구현 객체(object implementation)를 통해 실제로 구현된다. 그리고, OpenGIS 클라이언트는 사용자가 작성한 CORBA 응용 프로그램이다.

이런 구조는 CORBA의 특성상 다양한 환경 및 언어를 사용하여 구현할 수 있다. 예를 들어 UNIX에 있는 ODBMS를 위한 서버를 C++로 구축하고 클라이언트는 MS-Windows 상에서 C++

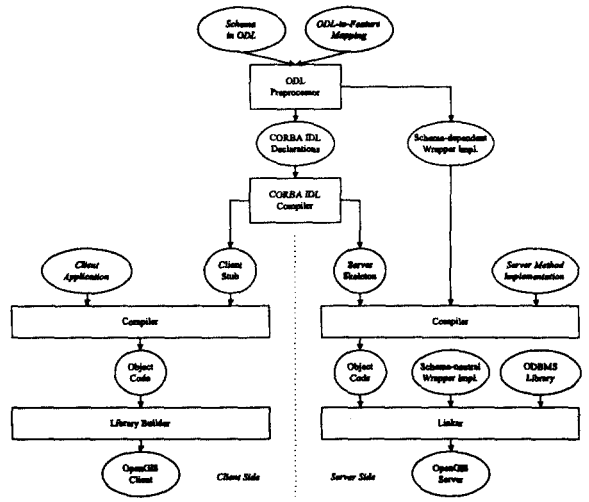


그림 3: 자동 생성 과정

나 Java로 작성할 수도 있다. 또한 CORBA는 인터넷 기본 프로토콜인 TCP/IP 상에서 구현된 IIOP 프로토콜을 사용하므로 인터넷 환경에서의 사용에도 아무런 제약이 없다는 장점이 있다.

OpenGIS 서버는 SFCORBA 인터페이스를 개별 ODBMS의 인터페이스에 맞게 변환해 주어야 한다. SFCORBA에서 ODBMS OML 인터페이스로의 변환을 담당하는 것은 Wrapper 객체이다. Wrapper 객체는 앞 절에서 설명한 CORBA 구현 객체의 실체이며 CORBA와 ODBMS 사이의 데이터 타입 변환을 하며 스키마 개발자가 ODBMS 상에서 구현한 사용자 정의 메소드를 호출하는 역할을 한다.

한편, SFCORBA 인터페이스는 타입을 실행시에 바인딩하는 문법을 기반으로 하고 있다. 하지만 ODBMS 인터페이스는 일반적으로 ODMG OML C++ 바인딩이며[8] C++은 컴파일시에 타입이 바인딩되기 때문에 Feature의 get/set\_property()와 같이 이름을 매개변수로 받아 값을 처리하는 인터페이스를 동적으로 처리할 수가 없다. 그러므로 컴파일할 때 가능한 모든 경우에 대해 처리하도록 코드를 작성해야 한다.

하지만 DBMS의 스키마가 바뀔 때마다 바뀌는 사용자 정의 타입에 대해 스키마 개발자가 서버를 매번 고쳐야 하는 것은 무리가 있으므로 본 연구에서는 스키마가 주어진 경우 자동으로 OpenGIS 서버를 생성하도록 전처리기를 개발하였다. 참고로 OpenGIS에는 스키마 정의에 대한 언급이 없기 때문에 본 연구에서는 ODMG ODL의 문법에 공간 데이터 타입을 추가한 언어를 사용한다.

다음 그림 3은 스키마 개발자가 정의한 ODL에서 Wrapper 객체를 생성하고 동적으로 타입을 바인딩하는 부분의 코드를 생성해내는 ODL 전처리기를 나타낸다. 왼쪽 부분은 클라이언트 측에서 응용 프로그램을 생성하는 과정을 나타내며 오른쪽 부분은 Wrapper 객체를 자동으로 생성하여 스키마 개발자가 구현한 메소드와 함께 서버를 생성하는 과정을 나타낸다.

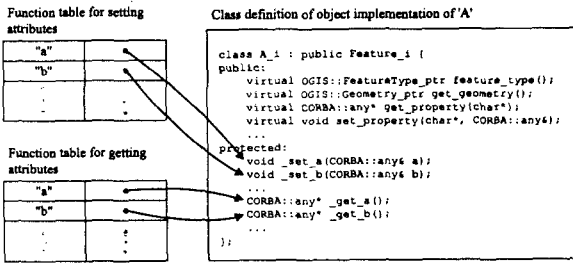


그림 4: Function Table의 예

#### 4 구현

본 연구에서는 ODBMS로는 Objectivity/DB[9]를 사용하였으며 CORBA로는 Orbix MT[10]를 사용하였다. Objectivity/DB에서 지리 정보 데이터를 다루기 위하여 본 연구실에서 개발한 SDBX(Spatial DataBase eXtension), 즉 공간 데이터 타입/연산자, 공간 인덱스 라이브러리를 활용하였다. SDBX는 OpenGIS Geometry 모델에 정의된 모든 타입 및 공간 연산자를 지원하며 공간 인덱스로는 R/R\*-tree를 지원한다.

Wrapper에서 이름을 매개변수로 하여 값을 처리하기 위하여 ODL 전처리 과정에서 생성되는 정보에 대해 살펴보자. 만약 'A'라는 인터페이스에 'a'와 'b'라는 속성이 정의되었다면 ODL 전처리기는 그림 4와 같은 C++ 클래스를 생성해 낸다. Feature\_i는 Feature 구현에 필요한 공통적인 인터페이스나 변수들이 선언되어 있으며 Feature\_i로부터 상속을 받은 A\_i라는 클래스에는 Feature 인터페이스의 구현 이외에도 .set\_a()와 .get\_a()와 같이 속성을 읽고 쓰는 메소드를 구현이 자동으로 생성된다. 그리고, 각 인터페이스마다 속성을 읽고 쓰기 위해 테이블을 두개씩 만든다. 이 테이블은 그림 4에서와 같이 속성의 이름과 해당 속성을 읽는(혹은 쓰는) 메소드에 대한 포인터를 저장한다. 이렇게 C++ 클래스와 테이블을 생성하면 get/set\_property()는 다음과 같이 간단히 구현될 수 있다. lookupGetFunctionMap(name)은 name에 해당하는 메소드 포인터를 돌려주는 함수이다.

```

CORBA::any* A_i::get_property(char* name) {
    return (this->lookupGetFunctionMap(name))(); }
void A_i::set_property(char* name, CORBA::any& value) {
    (this->lookupSetFunctionMap(name))(value); }
    
```

그 외에 property\_def\_exists와 같이 FeatureType 인터페이스에도 문자열로써 메타 정보를 접근하는 인터페이스가 있으며 Factory 관련 인터페이스의 경우에도 동적 타입 바인딩이 필요하다. 이들 구현할 때에도 Feature와 마찬가지로 ODL 전처리기를 통해 필요한 정보를 자동으로 생성해 낸다.

그림 5은 구현된 OpenGIS 서버를 이용하여 만든 Java 응용 프로그램을 보여준다. 이 클라이언트는 Java 애플릿 형태로 웹 브라우저 상에서 실행된다. 그림에서는 가운데 질의 사각형과 겹치는(overlap) 데이터를 찾은 결과이다. 데이터는 서울시 전체 도로 데이터를 사용하였으며 화면에는 동호대교 복단이 보이고 있다.

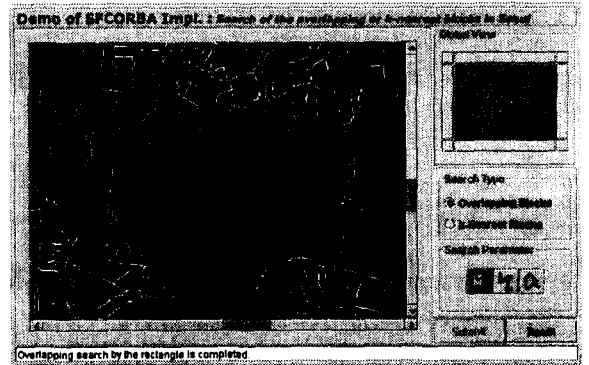


그림 5: Java 애플릿으로 만든 응용 프로그램

#### 5 결론

본 논문에서는 ODBMS에 저장된 지리 정보를 SFCORBA 인터페이스로 접근하기 위해 필요한 Wrapper와 그것을 자동으로 생성하는 과정에 대하여 설명하였다.

현재, 본 연구실에서 이미 개발한 SDBC[11]의 구조를 SFCORBA에 적용하여 캐시 기능을 이용한 성능 향상을 꾀하고 있으며, 서버 측에서 수행할 필요가 없는 메소드를 클라이언트 측에서 실행하는 클라이언트 메소드의 지원 방안을 연구하고 있다. 또한 Query\_Evaluator에서 질의 처리를 효율적으로 하는 방안에 대하여 연구하고 있다.

#### 참고문헌

- [1] "Yahoo! maps." <http://maps.yahoo.com/py/maps.py>.
- [2] K. Buehler and L. McKee, *The OpenGIS Guide: Introduction to Interoperable Geoprocessing*. Open GIS Consortium, Inc., 1996.
- [3] Open GIS Consortium, Inc., *OpenGIS Simple Features Specification For CORBA, OLE/COM, SQL*, 1998.
- [4] ESRI Inc., *SDE for Oracle Version 3.0.2*, 1998.
- [5] Oracle Corp., *Oracle8 Spatial Cartridge*, 1998.
- [6] 김기홍, 송창빈, and 유승원, "Web과 CORBA를 이용한 OpenGIS 정보 저장소 접근," 정보과학회지, 1999.
- [7] J. Siegel, ed., *CORBA: Fundamentals and Programming*. John Wiley & Sons, Inc, 1996.
- [8] R. G. G. Cattell and D. K. Barry, eds., *The Object Database Standard: ODMG 2.0*. Morgan Kaufmann, 1997.
- [9] Objectivity Inc., *Using Objectivity/C++*, 1996.
- [10] IONA Tech. Ltd., *Orbix Programming Guide*, 1996.
- [11] S. K. Cha, K. Kim, C. Song, J. Kim, Y. Kwon, and J. Jun, "A middleware for transparent access to multiple spatial object databases," in *Proceedings of InterOp'97*, pp. 84-99, Dec. 1997.

본 논문은 국가지리 정보 시스템 기술 개발 과제 중 OODBMS 인터페이스 과제(GI-04-03)를 수행하면서 얻어진 연구 결과의 일부입니다.