

이질적인 GIS 데이터 소스의 상호운용을 지원하는 CORBA기반의 표준 데이터 제공자 설계

(Design of CORBA based Standard Data Provider for Interoperability
of heterogeneous GIS Data Sources)

김민석*, 안경환**, 홍봉희***

(Kim MinSeok*, An KyoungHwan**, Hong Bong-Hee***)

초 록

기존에는 이질적인 GIS 데이터 소스간의 통합을 위해 래퍼(wrapper)기술을 사용했다. 래퍼(wrapper)는 외부 클라이언트 응용프로그램에게 이질적인 데이터 소스들의 통합 데이터 모델을 제공함으로써 데이터와 지리정보처리의 공유를 가능하게 한다. 그러나 데이터 소스들의 통합은 표준 서비스를 포함한 상호운용성을 지원하는 것은 아니다. 표준을 따르는 상호운용을 위해서는 표준화된 데이터 모델과 서비스가 필요한데 OGC (OpenGIS Consortium)의 OpenGIS(Open Geodata Interoperability Specification)가 이를 제공하고 있다. 분산환경에서 상호운용을 위한 기존방법은 각 데이터 소스별로 데이터 제공자를 개발하여 상호운용을 지원하였다. 따라서 중복 구현에 대한 문제와 구현에 따른 노력과 비용이 상당 부분 존재하였다. 본 논문에서는 데이터 제공자에서 공통부분을 추출 및 컴포넌트화하고 데이터 소스에 종속되는 모듈은 CORBA 구현객체로 작성하여 제시하였다. 따라서 각 데이터 소스들은 표준 데이터 제공자를 통해서 기존의 데이터 지원 기능유지와 상호운

용성을 달성할 수 있다. 클라이언트는 OpenGIS Feature모델을 기반으로 한 표준 인터페이스로 데이터 소스에 접근한다. 데이터 소스는 표준 데이터 제공자와 정의된 인터페이스를 바탕으로 CORBA의 구현 객체 서버를 작성함으로써 중복구현 및 구현부담을 줄일 수 있다.

키 워 드

표준 데이터 제공자, 표준 데이터 접근 모델, 통합 인터페이스, 컴포넌트, Wrapper

1. 서 론

네트워크 기술과 분산 컴퓨팅 기술의 발전으로 외부 응용프로그램들이 미들웨어를 이용하여 다양한 데이터 소스들에 대해 상호운용을 지원하는 연구가 활발히 진행되고 있다[BID96, EUC95, DIC98]. 상호운용이란 표준화된 지리 공간 데이터의 공유뿐만아니라 표준 API들을 인터페이스로 정의하여 지리정보처리의 공유까지 가능하게 해주는 것을 말한다[OGG98]. 따라서 외부 응용프로그램들은 미들웨어내의 개방된 표준 인터페이스 사용으로 이질적인 데이터 소스들에 대해서 상호운용성을 얻을 수 있다.

* 부산대학교 GIS학과 석사과정

** 부산대학교 컴퓨터공학과 박사과정

*** 부산대학교 컴퓨터공학과 교수

여러 데이터 소스들은 상호운용성을 지원하기 위해 고유 데이터 접근 모델을 표준 데이터 접근 모델로 변환을 해야 한다. 이를 위해 래퍼(wrapper)기술이 필요하다[TH97]. 래퍼(wrapper)는 서로 다른 데이터 소스들의 지리 데이터 표현과 해석을 통합된 접근모델로 외부에 제공한다. 따라서 래퍼(wrapper)는 데이터 접근 제공자 역할을 한다[OGG98]. 기존에는 상호운용성을 지원하기 위해 데이터 소스 수 만큼의 데이터 제공자를 개발해야만 했다. 즉, GIS 데이터 소스가 교체되거나 추가될 때마다 데이터 제공자가 구축되어야 한다. 이로 인해 각 데이터 소스의 데이터 제공자의 개발은 중복된 비용과 개발 부담을 개발자들에게 주고 있다. 본 논문에서는 데이터 제공자의 공통모듈을 추출하여 컴포넌트화 하고 데이터 소스에 종속되는 모듈은 CORBA의 구현객체로 캡슐화 한다.

기존 GIS 소프트웨어는 개발 및 운용 환경에 극히 종속적이어서 응용프로그램 개발시 사용자의 요구 사항에 대하여 능동적으로 대처하기가 어려운 실정이다. 이에 따라 현재 산업계 전반에 걸쳐 표준 개발 환경을 지원하는 컴포넌트에 대한 요구가 발생하고 있다. 따라서 환경 독립성을 갖춘 컴포넌트화 된 GIS 소프트웨어를 확보하고 플러그앤플레이(plug-and-play)를 달성하기 위해 표준 데이터 제공자의 구성요소를 컴포넌트화 하였다.

표준 데이터 제공자는 분산객체 표준 기술인 OMG(Object Management Group)의 CORBA를 이용하여 구현되고 통합 인터페이스를 통해 데이터 소스에 접근한다. 통합된 표준 데이터 제공자는 데이터 접근 컴포넌트와 지역 통합 인터페이스로 나누어진다. 데이터 접근 컴포넌트는 질의처리를 위한 질의어를 변환하고 질의실행결과를 개방된 표준 인터페이스로 접근할 수 있게 지원한다. 클라이언트에게 제공되는 개방된 API는 OpenGIS에서 정의하는 표준 인터페이스를 사용하였다. 질의를 위한 지역 데이터 소스의

스키마 정보에 대한 접근도 지원한다. 지역 통합 인터페이스는 데이터 소스에 종속되는 모듈들을 CORBA의 구현객체로 작성하고 인터페이스를 표준 데이터 제공자에게 지원한다.

본 논문의 구성은 다음과 같다. 2장에서는 본 논문과 관련된 기존연구와의 관계와 차이를 기술하였고 3장에서는 공통 컴포넌트화할 수 있는 모듈과 종속되는 모듈간의 추출근거를 마련하기 위한 기존 데이터 제공자들의 구조를 살펴보고 4장에서는 논문에서 설계한 시스템 구조와 구성내용을 기술하였다. 5장에서는 결론 및 향후연구 방향에 대해서 제시하였다.

2. 관련연구

2.1 wrapper기술

래퍼(wrapper)기술은 구조적 방법을 적용한 시스템을 객체지향적 시스템으로 전환하거나 다른 용도의 시스템으로 전환하는데 사용되는 소프트웨어 모듈이다[ILD96]. 본 논문에서의 적용은 상호 이질적이고 독립적인 기존 시스템의 상호운용성을 지원하기 위한 데이터 소스의 데이터 접근 모델을 외부에 일관되고 표준화된 데이터 접근 모델로 전해주는 모듈로 사용하였다. 래퍼(wrapper)가 제공하는 첫번째 기능으로는 데이터 소스의 능력에 대한 표현이 외부에 대해 투명하도록 제공한다. 두번째로 외부에 대한 일관된 표준 인터페이스를 제공한다. 세번째로 질의서비스 측면에서 질의를 수행하고 그 결과 전달방법이 일관된 접근 모델로 이루어진다. 래퍼(wrapper)기술은 적용분야에 따라 여러 분야로 나눌 수 있는데 그 중에서 데이터 변환 분야로 본 논문에서는 사용되었다[QTS96]. 데이터 변환은 데이터 접근 모델 변환을 말한다. 적용사례는 서로 다른 시스템들을 통합할 때 래퍼(wrapper)를 사용하며 이것은 표준화된 서비스를 지원하는 것은 아니지만 기존 시스템을 유지

하면서 사용할 수 있게 해주는 장점을 지니고 있다[BID96].

2.2 OpenGIS의 서비스 모델

최근에는 CORBA를 이용하여 DBMS들을 포함한 여러 데이터 소스들을 통합 및 상호운용하려는 연구가 활발히 이루어지고 있다[EUC95]. 그렇지만 기존의 연구들은 표준 인터페이스를 이용하지 않는 구조로 다양한 시스템간의 상호운용을 지원하지 못한다. 이러한 문제점을 극복하기 위해서는 상호운용 지리정보처리를 위한 개방형 표준으로 자리잡고 있는 OpenGIS(Open Geodata Interoperability Specification)의 이용이 필요하다. OpenGIS는 공간 데이터와 공간연산에 대한 분산접근을 위한 소프트웨어 기본설계의 포괄적 명세이며, OpenGIS에 부합하는 소프트웨어간에 상호운용이 가능하도록 해주는 공통된 표준 인터페이스 명세이다[OGG98]. OpenGIS는 상호운용의 기본이 되는 공통적인 기본지리 정보타입 및 지리정보를 접근, 관리, 표현 공유할 수 있도록 서비스 명세를 정의하고 있으며 지리정보의 생산자(provider)와 소비자(consumer)간의 공간정보유통 방법을 제시하고 있다. 또한 OGC(OpenGIS Consortium)에서는 이를 구현하기 위한 기술로서 OLE/COM, CORBA, SQL(ODBC)을 이용한 구현 명세를 내놓고있다[OGA98, OGS98]. 이들 구현 명세를 보면 OpenGIS 추상명세의 부분집합으로 이루어져 있으며, IDL(Interface Definition Language)과 같은 인터페이스 기술언어로서 표준 인터페이스를 정의하고 있다. 본 논문에서는 한 플랫폼에 종속 받지 않으며, 구현언어, 객체의 위치등에도 독립적인 CORBA를 이용한 OpenGIS의 표준 인터페이스 구현기술 기반으로 표준 데이터 제공자를 제시한다.

3. 기존 데이터 제공자의 구조

외부 응용프로그램에 개방된 API를 제공하는 데이터 제공자들의 공통모듈과 종속모듈들을 정의하기 위해 본 논문에서는 기존방법을 사용한 두 가지의 데이터 제공자를 예로 제시하였다. 데이터 제공자는 오라클과 고딕 데이터 소스를 대상으로 한다. 두 데이터 소스를 대상으로 한 이유는 서로 비교할 수 있는 특성들을 갖추고 있기 때문이다. 오라클은 관계형 DBMS를 기반으로 지리 데이터를 지원하며 질의어(SQL)가 지원된다. 반면에 고딕은 객체지향 DBMS를 기반으로 하고 질의어가 아닌 API를 지원한다. 데이터 제공자의 서비스는 질의처리를 대상으로 하며 OpenGIS의 CORBA 구현명세의 피처(feature)모델 인터페이스를 사용하여 클라이언트에게 제공한다[DIC98]. 데이터 제공자에서 데이터 소스에 종속되는 모듈은 해당 데이터 소스의 접근모델에 일치하게 작성한다. 전체적으로 데이터 제공자는 클라이언트에게 개방된 인터페이스를 제공하는 CORBA의 구현객체로 캡슐화 된다. 클라이언트가 사용하는 표준 인터페이스는 피처모델 인터페이스 중 QueryableContainerFeatureCollection(QCFC) 인터페이스를 사용한다. QCFC는 FeatureCollection을 생성하고 접근 조작하는 ContainerFeatureCollection과 질의를 실행하여 결과를 반환하는 QueryEvaluator인터페이스로 구성된다.

3.1 오라클 데이터 제공자의 구조

오라클 데이터 제공자는 질의어를 통한 질의처리 서비스를 클라이언트에게 제공한다. 질의처리는 오라클에서 제공하는 Embedded SQL을 이용하며 영역질의를 대상으로 한다. 데이터 제공자의 구조를 그림 1에서 나타내었다. 그림 1에서는 질의처리를 대상으로 한 데이터 제공자를 나타내었는데 클라이언트의 질의실행 및 결

과 데이터 접근은 Open API인 CORBA 구현객체의 인터페이스를 통해 이루어진다. 질의처리 서비스를 수행하기 위해 필요한 모듈로는 질의를 하기 위한 스키마 처리 모듈과 질의어 변환 및 실행모듈 그리고 질의결과 접근 모듈로 구성된다. 각각의 모듈 설계는 질의처리와 스키마 처리에 대한 전체 구조로 나타내었다. 스키마 질의 처리를 따로 분리한 이유는 질의처리방법이 다르며 그 결과를 저장하는 구조도 다르기 때문이다.

질의결과는 관계형 데이터베이스의 레코드 집합으로 생성되며 이것은 기하정보와 속성정보로 구분된다. 기하정보는 하나의 레이어내에서 유일 객체 식별자인 GID단위의 결과객체로 생성된다[OSC97, PCP97]. 질의결과에 대한 표준 지리 데이터인 피처로의 변환은 데이터 제공자에서 이루어진다. 스키마 처리는 Feature모델 인터페이스 중에서 FeatureType을 사용하여 처리된다.

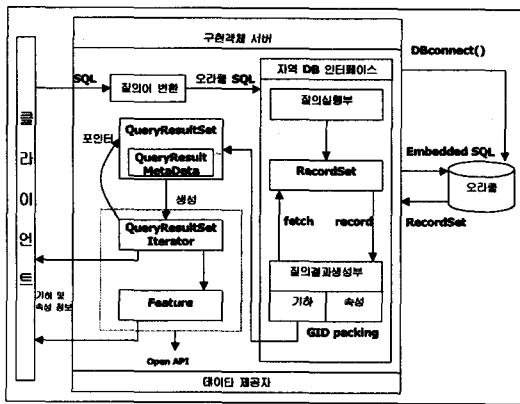


그림 1. 오라클 데이터 제공자 (질의처리) 전체 구조

3.2 고딕 데이터 제공자의 구조

고딕에서 외부 모듈과의 연결은 소켓통신을 사용해야만 한다. 따라서 오라클과는 다르게 데이터 제공자와 고딕간의 통신모듈이 추가된다. 그

림 2의 고딕 데이터 제공자는 오라클과 같이 클라이언트가 인터페이스를 통해서 질의처리와 결과를 피처로 얻는 데이터 접근 모듈을 포함하고 있다. 그리고 질의실행을 맡아서 결과를 넘겨주는 고딕의 API로 구성되며 종속모듈 역할을 하는 지역 DB 인터페이스가 있다. 지역 DB 인터페이스는 데이터 제공자와 고딕사이의 인터페이스 역할을 하며 소켓통신으로 연결된다. 질의를 수행할 때, 고딕이 질의어를 지원하고 있지 않기 때문에 클라이언트로부터 얻은 질의어를 질의인자로 분해한다. 질의인자는 질의를 실행하기 위해 질의어를 파싱(parsing)하고 그 결과가 데이터 구조로 생성된 것이다. 질의인자가 고딕에 전해지면 지역 DB 인터페이스에 의해 수행된 고유의 질의결과를 얻는다. 질의결과는 표준 데이터 접근 모델로 변환하여 클라이언트에게 제공된다. 고딕의 질의 실행 후 결과 형태는 OID의 집합이다. OID를 통해서 지리 데이터의 기하와 속성을 구할 수 있다. 스키마처리는 질의처리와 같이 고딕의 API를 사용하여 스키마 결과를 얻는다. 질의실행결과 얻어진 데이터는 오라클에서처럼 개방된 표준 인터페이스인 Feature Type을 사용하여 클라이언트에게 제공된다.

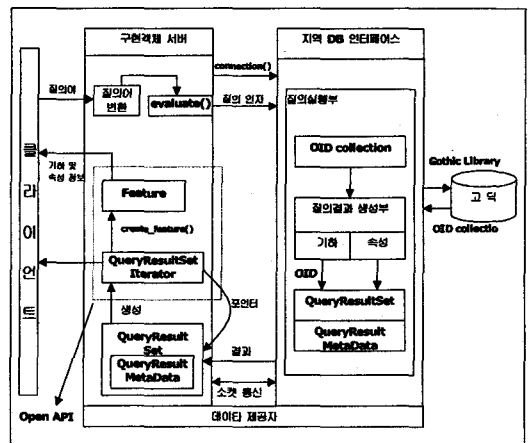


그림 2. 고딕 데이터 제공자 (질의처리) 전체 구조

3.3 비교 분석

앞에서 다룬 두 데이터 제공자를 비교해보면 공통점과 차이점을 알 수가 있다. 공통점은 첫째로 질의를 실행하기 전에 질의어를 지역 데이터 소스에서 수행 가능한 질의로 변환해야 한다. 개방된 공통의 API를 제공하기 위해 데이터 제공자는 클라이언트와 데이터 소스 사이의 래퍼(wrapper) 역할을 해야 하기 때문이다. 그러므로 모든 데이터 제공자는 클라이언트의 데이터 접근 모델과 지역 데이터 소스의 데이터 접근 모델 사이에서 변환기능을 담당해야 한다. 둘째는 질의처리가 지원되는 데이터 소스들이며 모두 DBMS의 자체 질의처리기(QueryProcessor)를 통해 질의를 실행한다. 세번째로 클라이언트의 질의결과에 대한 접근은 클라이언트에게 상호운용을 지원해야 하므로 표준 데이터 접근모델로 이루어져야 한다. 표준 데이터 접근 모델은 OpenGIS에서 정하고 있는 Feature모델을 사용하였다[OGA98, OGC98]. 클라이언트는 지역 데이터소스에 종속된 질의결과를 ResultSetIterator 인터페이스를 통해 접근한다. 질의결과는 ResultsetIterator를 통해 직접 속성이나 기하를 얻거나 필요한 경우에 피처(feature)로 생성되어 클라이언트에게 제공되며 데이터 소스의 데이터 접근모델에 관계없이 접근 가능하다. 따라서 데이터 소스로부터 생성된 질의 결과에 대한 접근은 표준 데이터 접근 모델의 변환을 통해서 이루어져야 한다. 차이점은 첫째로 질의처리에서 질의어 지원 유무이다. 질의 처리를 하는데 있어 질의어가 지원되지 않을 경우는 질의어에서 질의요소를 추출한 후에 데이터 소스의 API를 사용하여 질의처리를 수행한다. 질의요소는 질의 인자와 같은 개념으로 질의어를 파싱한 결과에서 생성된 데이터 구조이다. 반면 질의어가 지원될 경우는 지역 질의어를 생성 후에 DBMS에서 처리한다. 둘째로 고덕과 오라클의 데이터 모델이 각각 관계형 모델과 객체지향형 모델을 기반으로 하기 때문에 질의결과 생성방법이 서로

다르다.

스키마는 질의변환 없이 요청 메시지만 보내면 지역 데이터 소스에서 처리하여 피처모델에서 제공되는 FeatureType으로 생성되어 접근된다. 앞의 분석을 토대로 데이터 제공자에서 공통으로 처리되는 모듈은 표준 데이터 접근 모듈들과 지역 질의어 생성 모듈이다. 여기서 공통으로 처리 되어야할 모듈 중에 데이터 소스에 종속되는 모듈이 있다. 스키마 질의와 일반질의를 수행하는 모듈, 처리결과 변환모듈 등이 해당된다. 지금까지 살펴본 공통으로 처리되는 모듈과 종속 처리되는 모듈을 통해 하나의 표준이라고 할 수 있는 표준 데이터 제공자를 설계하여 제시한다.

4. 표준 데이터 제공자의 설계

표준 데이터제공자는 다양한 데이터소스들의 상호운용성을 지원하기 위해 각 데이터 소스별로 제공되는 데이터 제공자를 통합 표준화하여 위로는 표준화된 개방형 API를 제공하고 아래로는 통합된 인터페이스를 제공한다. 통합 인터페이스 구현부분이 각 데이터 소스별 종속모듈로 구현됨으로써 독립되고 표준화된 래퍼(Wrapper) 역할을 한다. 표준 데이터 제공자는 클라이언트를 위한 데이터 접근 컴포넌트, 그리고 데이터 소스를 위한 지역 스키마 접근 컴포넌트와 데이터 소스 종속모듈을 통합하기 위한 지역통합 인터페이스로 구성된다.

4.1 시스템 전체 구조

전체 시스템의 구조는 아래의 그림 3과 같이 3개의 계층으로 구성된다. 첫번째가 표준 서비스를 제공받는 클라이언트 계층이고 두번째가 데이터 소스와는 독립적이며 클라이언트에게는 표준화된 데이터 접근 모델을 제공하는 데이터 접근 제공자(Data Access Provider)인 표준 데

이타 제공자 계층이다. 마지막 계층은 데이터를 소유하고 있으면서 데이터 원천을 제공하는 데이터 소스 계층이다. 클라이언트와 표준 데이터 제공자 사이는 개방된 API를 통해 연결이 되고 표준 데이터 제공자와 데이터 소스는 표준 데이터 제공자에서 정의하는 인터페이스를 통해 연결된다.

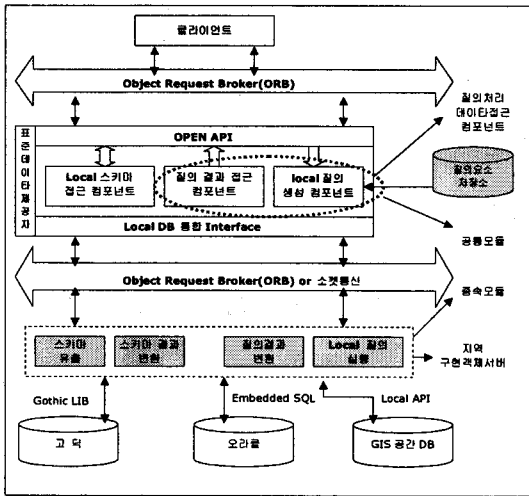


그림 3. 전체 시스템 구조

4.2 표준 데이터 제공자의 구성

표준 데이터 제공자는 데이터 제공자들 중에 독립적으로 수행할 수 있는 공통모듈이 추출된 컴포넌트들로 구성되어 있다. 이렇게 함으로써 데이터 제공자의 독립화가 가능하다. 이를 위해 기존 데이터 제공자의 공통모듈과 종속모듈을 분리하여야 한다. 이들을 구분하는 기준을 표준 데이터 접근 모델의 사용 유무로 하였다. 데이터 제공자가 서비스할 때 다루어지는 데이터가 표준 지리 데이터를 사용하는지에 따라 처리 모듈이 종속 또는 독립화 될 수 있기 때문이다. 그림 3에서 종속되는 부분을 음영으로 표시하였다. 표준 지리 데이터를 사용하는 공통모듈은 질의어 변환모듈과 표준 지리 데이터를 생성하

는 모듈, 접근 모듈 등이 있다. 이 모듈들을 하나의 데이터 제공자의 구성요소인 데이터 접근 컴포넌트로 정의하였다.

데이터 접근 컴포넌트는 OpenGIS Feature 모델에서 정의하는 서비스 인터페이스를 클라이언트에게 제공한다. 서비스로는 FeatureCollection, FeatureCollectionFactory, QueryEvaluator 인터페이스가 있다. 기존의 데이터 제공자에서 공통으로 통합될 수 없는 모듈, 즉 데이터 소스에 공통으로 종속되는 모듈의 인터페이스를 통합된 인터페이스로 정의하고 각 데이터 소스별로 인터페이스 구현 부분을 작성하도록 하였다. 따라서 표준 데이터 제공자는 데이터 접근 컴포넌트와 지역 통합 인터페이스로 구분하여 나눌 수가 있다. 표준 데이터 제공자 설계에서는 앞에서 사용한 질의결과 접근에 대한 Result SetIterator를 표준 데이터 제공자의 독립성을 위해서 FeatureCollection으로 대체하였다. 그래서 독립된 표준 데이터 제공자가 구성된다. 새로운 데이터 소스가 추가될 때 지역 통합 인터페이스 구현 부분만을 작성함으로써 데이터 소스의 구현 비용을 줄일 수 있다. 인터페이스내의 함수들은 지역 데이터 소스에 종속된 지리 데이터를 표준 지리 데이터인 피처객체로 변환하여 데이터 접근 컴포넌트에게 제공한다.

표준 데이터 제공자를 구성하는 방법으로 CORBA를 이용한 방법과 소켓을 통한 방법이 있다. 두 방법의 공통점은 여러 데이터 소스를 통합하기 위한 인터페이스가 제공된다는 것이다. 독립적인 데이터 접근 컴포넌트와 종속모듈과의 분리가 가능하다. 또한 서로 구현에 대한 투명성이 제공된다. 그리고 데이터 소스의 접속 정보와 질의요소 저장소에 정보를 등록하고 인터페이스 구현부분을 작성만하면 데이터 소스의 실시간 추가가 가능하다. 차이점은 CORBA에서는 위치의 투명성이 보장되지만 소켓통신보다는 성능면에서 뒤떨어진다. 소켓을 이용한 구조는

CORBA보다는 투명성이 부족하지만 성능면에서는 앞선다[DIC98].

4.3 데이터 접근 컴포넌트

데이터 접근 컴포넌트는 클라이언트에게 메타 정보와 질의처리 서비스를 제공하는 컴포넌트이다. 세부적으로는 클라이언트로부터 얻어 해당 지역 데이터 소스에 맞는 질의로 변환해서 전달하거나 메타 데이터를 요청하고 결과를 표준 데이터 접근 모델의 피처로 얻어 클라이언트가 접근할 수 있도록 인터페이스를 제공한다.

4.3.1 질의처리 데이터 접근 컴포넌트

질의처리 데이터 접근 컴포넌트는 질의를 수행하여 결과를 표준 데이터 접근 모델로 제공한다. 질의처리 데이터 접근 컴포넌트는 지역 질의 생성 컴포넌트와 질의결과 접근 컴포넌트로 구성된다. 지역 질의 생성 컴포넌트는 클라이언트에서 요구된 질의어를 해당 데이터 소스의 질의어로 변환한다. 질의결과 접근 컴포넌트는 질의결과를 피처객체로 얻고 이들을 FeatureCollection으로 생성해서 접근할 수 있도록 해주는 컴포넌트이다.

a) 지역 질의 생성 컴포넌트

지역 질의 생성 컴포넌트는 질의 파싱 모듈, 질의요소 변환 모듈, 지역 질의어 생성 모듈로 구성된다. 질의 파싱 모듈은 클라이언트가 요구한 질의어를 파서 트리(Parser Tree)로 생성한다. 질의 파싱 컴포넌트는 파싱된 질의요소들을 질의요소 저장소를 참조하여 해당 데이터 소스의 질의요소로 변환한다. 변환된 질의인자에 대해서 질의어 지원 유무에 따라 지역 질의 요소를 전달하거나 또는 지역 질의요소를 지역 질의어로 생성하는 지역 질의어 생성 컴포넌트가 있다. 질의요소 저장소는 질의요소를 데이터 소스

에 따른 지역 질의요소로 매핑하기 위한 저장소이다. 그림 4는 지역 질의어 생성과정을 나타내었다.

b) 질의결과 접근 컴포넌트

질의결과 접근 컴포넌트는 종속모듈에서 수행되어 생성된 질의 결과를 Feature Collection의 표준 데이터 접근 모델로 클라이언트에게 제공한다. 제공되는 서비스는 표준 지리 데이터에서 기하 및 속성 정보를 얻도록 해준다. 질의결과 접근 컴포넌트는 FeatureCollection을 제공하기 위해 통합 인터페이스를 사용하는데, 이때 인터페이스는 표준 지리 데이터인 피처 객체를 제공한다.

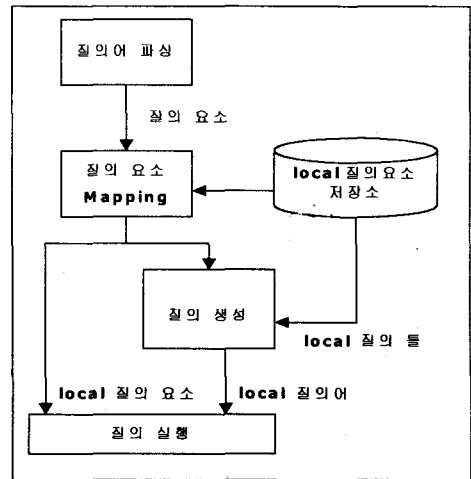


그림 4. 지역 질의어 생성 컴포넌트

4.3.2 지역 스키마 접근 컴포넌트

클라이언트는 지역 스키마 정보를 FeatureType인터페이스를 통해 접근할 수 있다. 지역 스키마 접근 컴포넌트는 스키마 요청 메시지를 지역 스키마 접근 컴포넌트를 통해 해당 지역 데이터 소스에 전달한다. 그리고 레이어의 메타정보인 각 레이어에 대한 타입, 스키마, 속성정의 정보등을 Feature Type으로 생성하여 제공한다. 스키마 정보를 얻기 위해서는 데이터

소스의 서버명을 통해서 접근한다.

4.4. 지역 통합 인터페이스

그림 5. 통합 인터페이스 IDL

```
interface LocalServerIntegation {  
  
    struct GeomConstraint {  
        spatial_op oper;  
        geometry geom;  
        integer geo_type;  
    };  
  
    struct Elements {  
        string select_list;  
        string layer_list;  
        GeomConstraint constraint;  
    };  
  
    union QueryStr (long){  
        case 1 : Elements elements;  
        case 2 : string Query;  
    };  
  
    struct DBInfo {  
        string DB_name;  
        string IP;  
        string user;  
        string passwd;  
        string qelement_repository;  
        string port;  
    };  
  
    void Local_Evaluate (in QueryStr que  
        in DBInfo dbinfo);  
  
    Feature Feature_fetch (in long point  
  
    void Local_Schema (in DBInfo dbinfo)  
  
    FeatureType FeatureType_fetch (in l  
        pointer);  
}
```

지역 통합 인터페이스는 기존 데이터 제공자 내의 지역 데이터 소스들과의 연결을 위한 종속 모듈 인터페이스들을 하나의 통합된 인터페이스로 정의한 것이다. 따라서 표준 데이터 제공자가 독립된 컴포넌트로 작성될 수 있게 해주는 인터페이스이다. 이 인터페이스를 통해 데이터

접근 컴포넌트는 데이터 소스에 대한 표준화된 데이터 접근이 가능하다.

지역 통합 인터페이스의 구현부분은 각 데이터 소스별로 작성되어지며 CORBA의 구현객체로 작성된 지역 구현 객체 서버이다. 지역 구현 객체 서버는 질의를 수행하고 결과를 생성, 변환하는 모듈과 스키마를 생성, 변환하는 모듈로 나누어진다. 질의처리 결과는 피처(feature)로 생성되고 스키마는 FeatureType으로 생성된다. 지역 구현 객체 서버에서의 주요기능은 종속된 지리 데이터를 표준 지리 데이터로 변환이다.

4.4.1 통합 인터페이스 선정방법

데이터 소스들간의 통합 인터페이스를 선정하는데 있어 인터페이스 구현측면과 적용측면에서의 선정 방법을 제시하였다. 첫째로 적용측면에서는 특정 데이터 소스로서의 매핑 비용을 최소화 할 수 있는데 이것은 인터페이스가 넓게 보편적으로 다른 데이터 소스들에 대해 적용되어야 하는 원칙에 어긋날 수가 있다. 적용을 위한 인터페이스 선정은 다양한 데이터 소스들의 특성을 고려해야 한다.

둘째로 구현측면에서는 데이터 소스의 질의 처리 능력을 고려해야 한다. 이것은 질의처리능력에 따라 정의될 인터페이스 레벨정도가 달라지기 때문이다. 즉 공간 DBMS 같은 데이터 소스는 자체 DB엔진을 사용하여 질의처리를 수행한다. 공간 DBMS 인터페이스 구현모듈은 질의처리를 수행하는 DB 인터페이스를 표준 데이터 제공자에게 제공 할 수 있다. 그러나 자체 응용 프로그램에 의존되어있는 파일시스템과 같은 데이터 소스에서는 질의처리를 위한 API가 지원되지 않는다면 데이터 소스에 접근할 수 있는 기능 등의 기본 입출력 함수들에 대한 인터페이스를 구현해야 한다. 따라서 표준 데이터 제공자는 이렇게 정의된 인터페이스에서 제공하는 함수들을 사용하여 질의처리 모듈을 구성한다.

이것은 질의처리를 데이터 소스측 모듈이 아닌 표준 데이터 제공자내 또는 클라이언트에서 처리해야 되는 경우이므로 성능면 뿐 만 아니라 구현방법에서도 비효율적이다. 질의처리는 데이터 소스에서 처리하여 결과만을 데이터 제공자로 넘겨주는 것이 효율적이므로 질의처리 구현은 종속모듈인 구현객체서버에서 작성된다. 따라서 질의처리능력과 방법이 다른 데이터 소스들이 지역 통합 인터페이스 정의로 인해 같은 처리방법과 처리능력을 가지게 된다.

4.4.2 통합 인터페이스의 정의

통합 인터페이스의 IDL은 그림 5와 같다. 인터페이스는 질의처리, 데이터소스 정보를 위한 데이터 구조와 질의처리 인터페이스로 구성된다. 질의처리 인터페이스는 질의실행, 결과를 얻는 함수들로 구성되어있다. 질의가 실행된 후 그 결과는 피쳐(feature)로 얻는다. 스키마처리에서도 스키마 결과를 FeatureType으로 얻는다. 통합 인터페이스를 통해서 표준 데이터 타입을 얻게 된다.

4.5 데이터 흐름 시나리오

클라이언트에서부터 선택된 데이터 소스까지의 데이터 흐름을 나타내는 시나리오이다. 질의처리와 스키마 유출을 위한 두 데이터 흐름 시나리오를 나타내었다.

4.5.1 질의처리 데이터 시나리오

질의처리를 위한 데이터 흐름을 그림 6에서 나타내었다. 클라이언트가 표준 데이터제공자에게 접속하여 질의어를 전달, 처리한 결과를 얻는 과정을 보여준다.

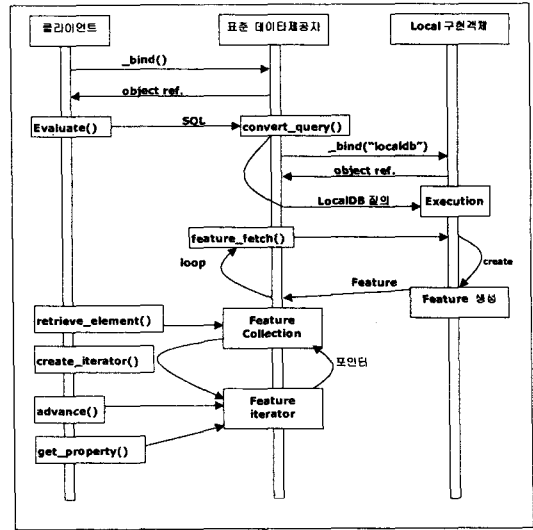


그림 6. 데이터 전체 흐름(질의처리)

먼저 클라이언트는 개방된 API를 사용하기 위한 표준 데이터 제공자 참조자를 얻는다. 클라이언트는 구현객체의 참조자를 통해 메소드를 호출하여 질의처리서비스를 제공 받는다. 지역 데이터 소스의 선정은 클라이언트에서 전달한 데이터 소스의 정보를 통해서 표준 데이터 제공자가 해당 지역 데이터 소스 구현 객체로 연결하여 이루어진다.

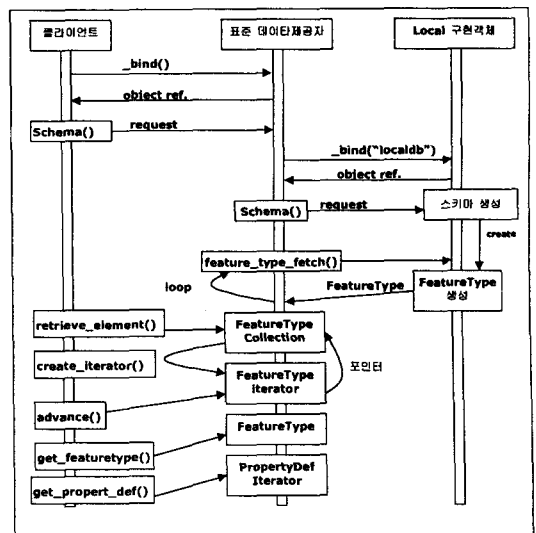


그림 7. 데이터 전체 흐름(스키마처리)

4.5.2 스키마 데이터 시나리오

스키마의 요구는 클라이언트가 질의서비스를 하기전에 제공받는 서비스이다. 그림 7은 스키마 정보의 요청과 결과 데이터를 접근하는 과정을 보여 주고 있다. 질의어가 아닌 스키마 요청 메시지 전달로 서비스가 이루어진다. 스키마를 제공받기 위한 데이터소스 선정은 질의처리와 같은 방법으로 이루어진다.

5. 결론 및 향후 연구방향

본 논문에서는 다양한 GIS의 데이터 소스들의 상호운용을 지원하기 위해 사용되는 각 데이터 소스의 데이터 접근 제공자를 통합 표준화시킨 표준 데이터 제공자에 필요한 기술과 구성요소를 설계하였다. 표준 데이터 제공자는 데이터 소스에 대한 표준이면서 통합 인터페이스를 제공하고 외부 클라이언트에 대해서는 서비스를 위한 표준 지리 데이터와 개방된 표준 API를 제공하여 상호운용성을 지원한다. 데이터 소스에 대한 지역 통합 인터페이스는 이질적인 데이터 소스들의 특성을 CORBA객체로 캡슐화하여 투명성을 제공한다. 표준 API들은 데이터 제공자가 표준 데이터 접근 모델과 서비스들을 제공하는 API들이다.

표준 데이터 제공자 구성은 외부 인터페이스가 무엇을 지원하느냐에 따라 크게 두 부분으로 나누어진다. 데이터 접근 컴포넌트와 지역 통합 인터페이스가 해당된다. 구분의 기준은 사용되는 데이터에 따라서 나누었다. 데이터는 표준 지리 데이터와 지역 지리 데이터로 구분된다. 표준 데이터 제공자는 CORBA를 사용하여 설계하였다. CORBA는 이종의 분산된 환경하에서 응용프로그램들이 서로 통합할 수 있는 분산객체 시스템의 표준으로써 확장성과 이식성 때문에 선택되었다. 따라서 각 클라이언트 데이터 서버에 대한 인터페이스를 CORBA 구현객체

서버로 작성하였다. 개방된 API를 위한 표준 데이터 접근 모델은 OpenGIS의 표준 인터페이스 모델인 피쳐모델을 사용하여 표준 데이터와 서비스의 공유를 제공하였다.

지역 데이터 소스에 대한 종속모듈은 인터페이스로 정의하여 작성함으로써 클라이언트 프로그램이나 표준 데이터 제공자 코드를 변경하지 않고서도 GIS 데이터 소스의 실시간 추가가 가능하며 구현비용도 감소될 수 있다. 또한 클라이언트에서 필요로하는 표준 데이터 접근 모듈을 컴포넌트들로 구성하여 컴포넌트화함으로써 확장, 재사용, 변경등이 용이하다.

향후연구로 단기적으로 표준 데이터 제공자의 상세 설계가 진행되고 이를 이용한 시스템의 구현이 뒤따라야할 것이다. 그리고 구현에 따른 표준 데이터 제공자의 구성요소별 성능측정이 이루어져야할 것이다. 장기적으로는 CORBA기반에서의 성능향상을 위한 지리 데이터의 객체단위나 영역단위의 캐쉬기법에 관한 연구와 미들웨어내의 추가적 서비스를 위한 분산질의처리에 관한 연구가 진행 되어야할 것이다.

참고문헌

- [ILD96] Ling Lu, Ling Ling, and M.Tamer Ozsu, Interoperability in Large-scale Distributed Information Delivery Systems, 1996
- [BID96] Asuman Dogac Cevdet Dengi M.Tamer Ozsu, Building Interoperable Databases on Distributed Object Management Platforms, 1996
- [EUC95] Ebru Kil, Gokhan Ozhan, Cevdet Dengi, Nihan Kesim, etc, Experiences in Using CORBA for a Multidatabase Implementation, 1995
- [QTS95] Yannis Papakonstantinou, Ashish Gupta, Hector Garcia-Molna, Jeffrey Ullman, A Query Translation Schema for

Rapid Implementation of Wrappers, 1995

[TIH97] Yooshin Lee and Ling Liu, Calton Pu, Towards Interoperable Heterogeneous Information Systems: An Experiment Using the DIOM Approach, 1997

[PDI97] Yooshin Lee, Prototyping The DIOM Interoperable System, 1997

[DIC98] 안경환, 조대수, 홍봉희, 상호운용을 지원하는 CORBA 기반 개방형 지리정보시스템의 설계 및 구현, 1998

[OGG98] OpenGIS Consortium, Inc., The OpenGIS Guide, 1998

[OGA98] OpenGIS Consortium, Inc., The OpenGIS Abstract Specification Model, version3, 1998

[OGS98] OpenGIS Consortium, Inc., OpenGIS Simple Features Specification for CORBA, Revision 1.0, 1998

[OSC97] Oracle Corporation, Oracle8 Spatial Cartridge Users Guide and Reference, 1997

[PCP97] Oracle Corporation, Pro C/C++ Precompiler Release8.0 Programmers Guide, 1997

김민석

1998년 경성대학교 컴퓨터공학과 졸업(공학사)
 1998년-현재 부산대학교 대학원 GIS학과, 석사과정
 관심분야 : GIS, 공간데이터베이스, 분산 객체 기술, 개방형지리정보시스템

안경환

1997년 부산대학교 컴퓨터공학과 졸업(공학사)
 1999년 부산대학교 대학원 컴퓨터공학과 졸업(공학석사)
 1999년-현재 부산대학교 대학원 컴퓨터공학과, 박사과정
 관심분야 : GIS, 분산객체기술, 개방형 지리 정보시스템

홍봉희

1982년 서울대학교 전자계산기공학과 졸업(공학사)
 1984년 서울대학교 대학원 전자계산기공학과 졸업(공학석사)
 1988년 서울대학교 대학원 전자계산기공학과 졸업(공학박사)
 현재 부산대학교 공과대학 컴퓨터공학과 부교수
 관심분야 : 공간데이터베이스, GIS표준화, 병렬 GIS, 개방형지리정보시스템