

미들웨어 : OLE DB를 기반으로 한 공간 데이터 변경 프로토콜의 설계 및 구현

박 정하*, 김 동현, 홍 봉희

부산대학교 컴퓨터공학과

{pjh0083*, pusrover, bhhong} @hyowon.pusan.ac.kr

The Design and Implementation of Update Protocol for Spatial Data based on the Middleware : OLE DB

Joung-Ha Park*, Dong-Hyun Kim, Bong-Hee Hong
Dept. of Computer Engineering, Pusan National University

요 약

지리정보시스템 클라이언트는 Open API를 이용한 개방형(Open) 구조를 지향하고 있으며 OpenGIS는 다양한 데이터 소스간의 상호운용을 지원하는 개방형 지리정보시스템의 표준으로 자리를 잡아가고 있다. 따라서 기존의 폐쇄환경(Closed Environment)에서 행해졌던 공간 데이터 동시 수정의 연구는 OLE DB와 같은 개방환경(Open Environment)에서 연구될 필요가 있다. 기존의 기법을 OLE DB에 적용할 때 문제점으로 제기되는 것은 첫째, 공간데이터 수정을 위한 변경 제어가 서버측에 위치함으로써 서버에 따라 재작성 되어야 하는 문제가 있고, 둘째, 서버의 데이터를 중복 저장하고 있는 클라이언트 데이터에 대한 일관성 유지를 위해 클라이언트의 변경 내용은 다른 클라이언트에게 통보되어야 하지만 OLE DB의 통보 메커니즘은 Rowset을 공유하는 동일 클라이언트 내에서의 통보만 가능한 구조이다.

본 논문은 공간데이터베이스 서버에 의존적인 서버부분의 기능을 미들웨어에서 구현함으로써 서버 모듈의 재작성 문제를 해결하고 서로 다른 클라이언트간의 변경 전파 프로토콜을 제시하여 서버와 독립적으로 운용할 수 있는 공간데이터 변경을 위한 OLE DB 서비스 제공자 프로토콜을 설계하고 구현한 결과를 보인다.

1. 서론

클라이언트/서버 GIS 환경에서 공간데이터 수정 작업은 대화식의 긴 트랜잭션이므로 잠금에 의한 긴 시간 대기문제가 발생하지만 영역잠금과 변경 전파 프로토콜을 이용하여 작업의 동시성을 높일 수 있다 [4]. 그러나 지리정보시스템 클라이언트는 Open API를 이용한 개방형(Open) 구조를 지향하고 있으며 OpenGIS는 다양한 데이터 소스간의 상호운용을 지원하는 개방형 지리정보시스템의 표준으로 자리를 잡아가고 있다. 따라서 [4, 5]와 같이 폐쇄환경(Closed environment)이 아닌 OLE DB와 같은 개방환경(Open environment)에서 공간 데이터 동시 수정을 위한 연구가 필요하다.

[4, 5]에서 제시한 공간데이터 변경 기법을 OLE DB에 적용할 때 다음과 같은 문제점이 있다. 첫째, 변경전파 프로토콜의 수행을 위한 서버측 모듈은 서버에 의존적인 모듈로서 서버에 따라 재작성 되어야 한다. 즉 협동 트랜잭션을 위한 조정자가 서버측에 위치함으로써 서버 의존적인 모듈의 구현이 필요하게 된다. 둘째, OLE DB의 변경 전파 방법은 Rowset을 공유하는 동일 클라이언트 내에서의 변경전파만 가능한 구조이므로 서버와 독립적으로 클라이언트간 변경 전파를 수행하기 위해서는 새로운 프로토콜 제정이 필요하다.

본 논문에서는 동시성 제어 및 변경 전파 작업을 OLE DB의 서비스 제공자에서 수행할 수 있는 구조를 제시하여 서버측 제어모듈의 재작성 문제를 해결한다. 그리고 Rowset형태로 중복 저장된 공간 데이터의 일관성 유지를 위해서 조정자 클라이언트에서 참여 클라이언트들로 공간데이터 변경 전파를 위한 메커니즘을 제시한다.

이 논문의 구성은 다음과 같다. 2장에서는 관련 연구를 기술하고, 3장에서 OLE DB를 이용한 공간데이터 수정의 문제점을 제시한다. 그리고 4장에서 공간 데이터 수정을 위한 OLE DB 서비스 제공자 구조와 변경 프로토콜을 제시하고, 5장에서는 구현 내용을 기술한다. 마지막으로 6장에서는 결론 및 향후 연구 과제를 제시한다.

2. 관련연구

[3]은 OLE DB를 통한 다양한 형태의 데이터 소스에 대한 접근방법을 제시하고 하였고, 다수의 클라이언트가 데이터 소스의 일부 데이터를 동적으로 중복 저장할 경우 수정에 의한 클라이언트 데이터에 대한 일관성 유지를 위한 통보 메커니즘과 인터페이스를 제시했다.

[4]는 클라이언트/서버 환경에서 캐쉬 공간데이터의 잠진적 변경전파 방법을 제시하였다. 여기서는 기존의 잠금기법을 확장한

영역잠금 기법을 사용하여 클라이언트의 동시성을 지원하고 캐쉬 공간데이터의 잠진적 변경전과 프로토콜을 제시하였다. [4]에서는 협동 트랜잭션 수행을 위한 조정자가 서버쪽에 위치함으로써 각각의 Server에 따라 재작성 되어야 하는 문제가 있고 Open API를 이용한 개방형 구조가 아닌 폐쇄환경에서 구현이 이루어져 개방환경에 바로 적용하기에는 어려움이 있다. 따라서 본 논문에서 서버와 독립적인 공간데이터 수정 서비스 구조를 제시한다.

[6]에서는 OpenGIS의 OLE/COM 구현명세를 분석하여 GIS서버간의 상호운용을 지원하는 OLE DB 데이터 제공자를 설계하고 구현하였다. 그러나 여기서는 GIS 서버의 발견(Discover)과 접근(Access)을 통해 상호운용을 지원하지만 공간데이터 수정기능이 없는 문제점이 있다.

COM(Component Object Model)은 서로 동적으로 교신할 수 있고 새로운 기능을 지원하기 위해 모일 수 있는 어플리케이션 구성요소 개발을 위한 표준으로서 COM의 확장 가능한 객체 구조를 이용하여 다양한 분야에서 사용된다[3]. OLE DB는 OLE/COM기반으로 데이터베이스를 접목하여 다양한 형태로 구축되어 있는 데이터 소스에 대한 하나의 일관된 인터페이스를 제공하는 COM 인터페이스의 집합이다. 그러나 OLE DB에서는 공간 데이터에 대한 접근과 조작에 대한 인터페이스를 제공하지는 않는다.

3. OLE DB를 이용한 공간 데이터 수정의 문제점

이 장에서는 OLE DB를 이용하여 공간데이터의 수정을 수행할 때의 문제점을 설명한다.

3.1 긴 시간 대기 문제(Long Wait)

OLE DB에서의 잠금기법은 Rowset의 Single Row에 대한 DBPROP_LOCKMODE를 설정함으로써 쓰기잠금을 제공한다. 그리고 OLE DB에서 제공하는 Transaction관련 인터페이스는 GIS응용과 같은 긴 트랜잭션(Long duration transaction)을 고려하지 않은 OLTP 트랜잭션이다. 기존의 Rowset의 Single Row에 대한 Read/Write잠금과 트랜잭션으로는 서버에 동시에 접속하여 지도 수정작업을 하는 클라이언트의 동시성 지원이 어렵고 장시간 대기문제를 야기시킨다.

[4]는 영역잠금에 기초한 확장된 잠금기법과 변경 전과 프로토콜을 통한 일관성 유지기법을 도입하여 클라이언트/서버 환경에서 구현을 하였다. 본 논문은 긴 시간 대기문제를 [4]에서 제시한 확장된 잠금기법과 변경 전과 프로토콜을 도입하여 OLE DB에서 발생하는 긴 시간 대기문제를 해결한다.

3.2 공간데이터베이스 서버에 따른 재작성 문제

[4]에서 제시한 잠진적 변경 전과 프로토콜 수행을 위한 조정자는 서버에 의존적인 모듈로서 서버에 따라 재작성 되어야 한다. 이와 같은 재작성으로 인해 고려되어야 하는 문제점들은 다음과 같다. 첫째, 공간데이터베이스 서버에 적합한 데이터 제공자는 OpenGIS Simple Feature Spec에 따른다면 공간데이터 수정기능이 없다. 기존의 데이터 제공자를 재사용하면서 공간데이터 수정기능을 제공하려면 별도의 모듈을 서비스 형태로 제공해야 한다. 둘째, 서버마다 변경 전과 프로토콜 수행을 위해 별도의 모듈을 작성하게 되므로 비효율적이다. 따라서 서버에 의존적인 모듈구현을 미들웨어 수준에서 구현함으로써 서버와 독립적인 컴포넌트 구조의 제시가 필요하다.

3.3 OLE DB의 통보 인터페이스

클라이언트는 실행시간에 서버에 요청한 데이터를 Rowset형태로

중복 저장한다. 각 클라이언트에서 중복 저장하고 있는 데이터는 클라이언트의 요구에 따라 변경될 수 있고 각 클라이언트에 중복 저장된 데이터에 대한 일관성 유지가 필요하다. OLE DB에서 일관성 유지를 위해 통보기법을 사용하는데 Rowset을 공유하는 클라이언트 내에서의 변경전과만 가능하다. 그러나 서버의 데이터를 동적으로 중복 저장하고 있는 클라이언트간의 일관성 유지를 위해서는 변경된 내용이 클라이언트로 전파되어야 하므로 이 논문에서 클라이언트간 변경 전과 구조와 프로토콜을 제시한다.

4. OLE DB 공간 데이터 서비스 제공자

이 장에서는 서버와 독립적으로 공간데이터의 변경 전과를 수행하는 미들웨어 구조를 설명한다.

그림 1은 [4]에서 변경 전과를 수행하는 제어 모듈의 위치와 OLE DB 서비스 제공자에서의 위치를 비교한 것이다. 그림 1-[1]은 변경 전과의 조정자(coordinator)가 서버측에 존재하기 때문에 서버가 변경되면 서버에 따라 제어 모듈을 재작성 해야 하는 문제가 발생하므로 본 논문에서는 그림 1-[2]와 같이 서버측의 변경 전과 프로토콜을 미들웨어에서 구현하여 서버에 의존적인 부분을 제거하여 재작성 문제를 해결한다.

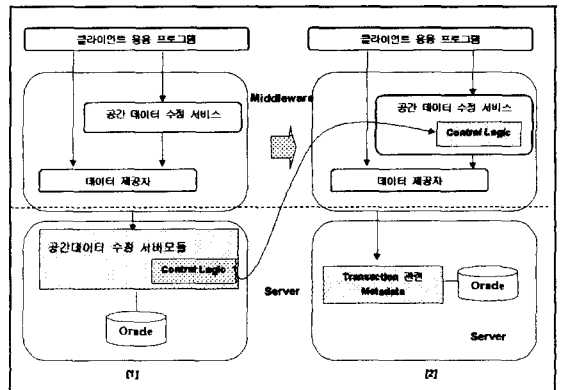


그림 1. 잠진적 변경 전과 제어의 위치

기존의 방법은 변경 전과를 수행하는 조정자가 서버가 되지만 미들웨어를 이용한 구조에서는 트랜잭션을 수행하는 각 클라이언트가 조정자가 된다. 이때 서버는 변경 전과와 관련된 메타정보를 유지하는 역할을 수행한다. 이때 공간데이터의 수정을 수행하는 서비스 제공자는 데이터 제공자를 통한 서버의 메타정보를 이용하여 수정작업을 수행하므로 서버가 바뀌어도 재사용이 가능하게 된다. 그리고 클라이언트에게 필요한 각 기능들을 COM 모델을 이용하여 컴포넌트화 함으로써 이후에 사용자의 요구에 맞게 재사용 가능하게 한다.

클라이언트에 Rowset형태로 중복 저장된 서버의 데이터는 클라이언트에 의해 변경되는데 각 클라이언트에 중복 저장된 데이터의 일관성 유지를 위해 변경된 내용은 전파되어야 한다 [4].

본 논문에서는 서비스 제공자 내에 통신관리자(Communication Manager)를 두어 클라이언트간의 잠진적 변경 전과를 담당한다. 여기서 각 클라이언트에서 변경된 내용(Delta)과 클라이언트들의 상태정보는 서버의 시스템 테이블로 정보를 유지하고 있으며 데이터 제공자를 통해 접근이 가능하다. 그림 2는 클라이언트간 변경 전과를 수행하는 통신모듈의 내부 구조도를 나타낸다. 각

통신모듈에는 수신모듈과 송신모듈이 있고 변경된 정보(Delta)는 Factory모듈에서 Packet Stream을 생성하여 참여 클라이언트로 전송된다. 클라이언트간 통신의 제어는 서비스 내의 수정 프로토콜 관리자가 수행한다.

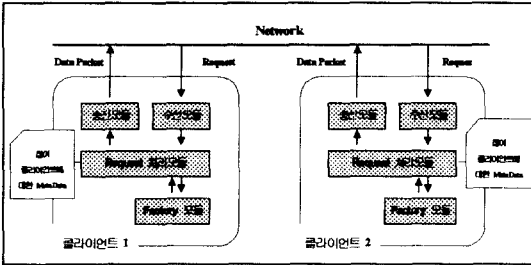


그림 2. 클라이언트간 변경 전파 구조도

5. 구현

그림 3은 본 논문의 미들웨어와 전체 시스템 구조이다. 서버는 Oracle Spatial Cartridge를 대상으로 구현하였으며, 미들웨어인 OLE DB 공간데이터 서비스 컴포넌트는 Windows98/WindowsNT환경에서 구현하였다. 클라이언트는 OLE DB 데이터 사용자로서 미들웨어를 통해 서버에 접근하고 공간데이터 수정을 수행한다.

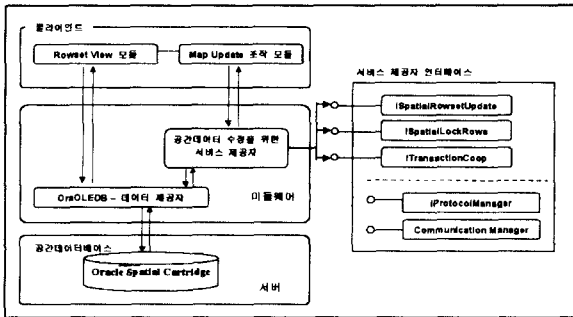


그림 3. 미들웨어 및 전체 시스템 구조

Oracle의 트랜잭션 관리기와는 별도로 미들웨어에서 트랜잭션 관리기(Transaction Manager)를 두었다. Oracle의 트랜잭션은 서버내의 지역 트랜잭션을 관리할 때 사용하는 것이므로 이 논문에서는 미들웨어의 트랜잭션들을 전역적으로 관리하는 별도의 트랜잭션 관리기를 구현하였다.

또한 Lock Manager는 OLE DB에서 제공하는 기본적인 Read/Write 잠금모드와는 별도로 영역잠금에 기초한 확장된 잠금을 관리한다. 실제 서버의 데이터에 Write잠금을 설정하고 관리하는 것은 Oracle 서버지만 이 논문에서 사용하는 변경 전파 프로토콜에 의한 확장된 잠금을 설정하는 것은 미들웨어에서 구현한 별도의 Lock Manager이다.

Spatial Update Interface는 기존 OLE DB에서 제공하는 Update 관련 인터페이스를 이용하여 공간객체 단위의 수정을 가능하게 하는 상위레벨의 공간데이터 수정 인터페이스를 제공한다.

Update Protocol Manager는 서비스 제공자의 핵심 부분으로서 변경 전파 프로토콜을 수행한다.

Communication Manager는 서버에서 유지하고 있는 정보(협동 트랜잭션에 참여하는 클라이언트 정보)를 바탕으로 각

클라이언트에게 실제 Delta 정보 등을 전달하는 역할을 수행한다.

그림 4는 Oracle의 데이터 제공자인 OraOLEDB를 이용하여 Rowset을 획득하고 서비스 제공자를 이용하는 지도 수정 접근과정을 나타내었다.

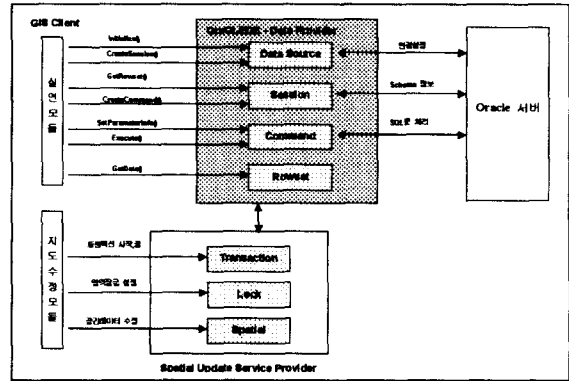


그림 4. 클라이언트의 지도 수정 접근 과정

6. 결론 및 향후 연구

본 논문에서는 OLE DB를 이용하여 GIS 공간 데이터 수정 트랜잭션의 동시성과 협동 트랜잭션을 지원하기 위한 기존의 연구방법을 OLE DB에 적용했을 때 발생하는 문제점을 기술하고 구현상의 문제와 해결방안을 제시하였다. 즉, 서버에 의존적인 기존의 공간데이터 변경기법을 OLE DB 서비스 제공자에서 구현함으로써 서버측 제어 모듈의 재작성 문제를 해결하였고 각 클라이언트에 Rowset형태로 저장된 서버 데이터의 일관성 유지를 위해 필요한 클라이언트간 변경 전파 메커니즘을 구현하였다.

향후 연구과제로 이 논문에서 제시된 서비스 제공자를 확장하여 분산환경에 적용하고 미들웨어 구조로 인해 예상되는 성능저하의 요인과 성능향상을 위한 연구가 필요하다.

7. 참고 문헌

- [1] Open GIS Consortium, OpenGIS Simple Features Specification for OLE/COM Revision 1.0, 1998
- [2] Microsoft Press, Microsoft OLE DB 2.0 Programmer's Reference and Data Access SDK, 1998
- [3] Jose A.Blakeley "Data Access for the Masses through OLE DB"
- [4] 신영상, 최진오, 홍봉희, "클라이언트-서버 환경에서 캐시 공간 데이터의 변경 전파", 한국정보과학회 '99 봄학술발표논문집(B), 제 26권 1호, pp 86-88.
- [5] 최진오, 홍봉희, "분산된 지리정보시스템에서 새로운 잠금기법을 이용한 중복된 공간 데이터의 변경 전파", 한국정보과학회 논문지 p1061 ~ 1072
- [6] 윤우진,한성룡,조대수,홍봉희, "OLE/COM을 기반으로 한 OpenGIS 미들웨어 설계" '99 한국 개방형 GIS연구회 p95 ~ p106
- [7] Microsoft Data Access SDK 2.1 MSDN Help, 1999
- [8] Oracle 8 Spatial Cartridge User's Guide and Reference Release 8.0.4
- [9] Oracle Provider for OLE DB User's Guide Release 8.1.5.2.0 Beta