멀티서버 환경에서의 공간 뷰 생성 및 실체화 기법

김태연* 정보흥 조숙경 배해영
인하대학교 전자계산공학과
yeon211@hanmail.net

The Spatial View Creation and Materialization Technique in Multi-Server Environments

Tae-Yeon Kim, Bo-Heung Jung, Sook-Kyoung Cho, Hae-Young Bae
Dept. of Computer Science Engineering, Inha University

요 약

지금까지 구축 된 공간 데이터들을 각 서버끼리 공유하여 유사한 내용의 공간정보구축을 위한 중복투자를 막고 공간데이터의 활용도를 높이는 방안이 시급하다. 멀티서버 환경에서의 공간데이터의 공유는 데이터의 보안과 사용자 편의성을 고려하여 뷰로 제공되고 관리되어질 수 있다. 분산 된 대용량의 공간데이터에 대해 복잡한 공간연산 처리를 감안한다면 멀티서버 환경에서는 공간 뷰의 실체화가 필수적이다. 본 논문에서는 멀티서버 환경에서의 공간데이터의 유통을 지원하고 공유 된 공간데이터에 대한 사용자 편의성과 네트웍 부하를 고려한 공간 뷰 실체화 기법을 제안한다. 확장 된 공간데이터 디렉토리 서비스(Extended Spatial Data Directory Service)를 이용하여 각 서버가 공유하는 공간테이블에 대한 정보를 검색하고, 검색 된 정보를 바탕으로 로컬서버는 각 서버에 직접 접속하여 필요한 영역의 데이터를 얻어 뷰를 생성한다. 이때 공유 공간데이터가 대용량 데이터임을 고려하여 로컬서버는 공간 뷰 정의에 대한 술어를 분리하여 뷰 질의 처리를 각각의 원격서버로 분산시켜 네트웍 부하와 서버의 질의처리 비용을 줄이고 로컬 서버에 실체화 시킴으로써 뷰에 대한 질의처리시 각 서버로의 잦은 데이터의 접근을 줄여 사용자 응답시간을 최소화한다.

1.서론

현재 우리나라는 1995년부터 시작된 국가지리정보체계 구축계획과 공간정보의 생산,수집으로 이미 상당부분 구축되어 있고 현재도 구축되고 있다. 각 기관에서 해당 기관의 목적에 따라 공간정보 데이터베이스를 구축하고 사용중이다. 이러한 결과물들이 계속적으로 구축되고 있으나, 그 구축내용과 사용할 수 있는 방법에 대한 정보가 극히 제한적이기 때문에 결국 첨단정보로 활용되기보다는 어딘가에 보관되어 유통되지 못하고 사장되는 것이 현실이다. 수치데이터 구축에 상당히 많은 비용이 소요됨을 고려한다면, 이렇게 많은 비용을 들여 구축한 정보를 다양한 분야에서 공동으로 활용하는 것은 매우 중요하다. 유사한 내용의 공간정보구축을 위한 중복투자를 최소화하고 기 구축된 공간데이터의 활용도를 높이기 위한 방안이 대두되고 있다.[1]

공간 데이터베이스 시스템의 지리데이터는 일부 영역 단위로 만들어지는 것이 아니라 건물, 도로, 등고선, 행정경계, 지하철 노선 등과 같이 서로 다른 주제(Theme)를 가지는 공간 객체들의 집합으로 구성된다. 이와 같은 공간 객체를 관리하는 일반적인 방식은 각각의 주제별로 그 주제에 속하는 공간 객체들의 집합을 하나의 레이어로 관리하는 것이다. 사용자의 편의성과 보안상 목적을 위하여 공간데이터베이스 시스템은 레이어 전체가 아닌 사용자 요구에 따른 공간 뷰를 제공하는데, 공간 뷰는 공간데이터의 크기와 공간연산의 복잡도를 고려해서 실체화 되어 관리되어진다.[2]

[1]본 논문에서는 멀티서버 환경에서의 공간데이터의 유통을 지원하고 공유 된 공간데이터에 대한 사용자 편의성과 서버의 처리비용 및 전송부하의 최소화를 위해 로컬 서버에의 공간 뷰 실체화 기법을 제시한다. 본 기법은 멀티 공간데이터베이스 서버 환경에서의 공간데이터의 공유를 위해 LDAP을 이용한 확장 된 공간데이터베이스 디렉토리 서비스(Extended Spatial Data Directory Service)를 사용한다. 이를 통해 각각의 공간데이터베이스 서버는 인증 된 서버끼리 공간데이터를 공유하여 중복된 공간정보 구축을 피하고, 공간데이터의 활용도를 높인다.

본 논문에서 제안하는 기법은 멀티서버 환경에서 원격 서버에 있는 공간데이터에 대한 공간 뷰의 요청 시 Extended Spatial Data Directory Service 서버를 이용하여 공간테이블 정보를 검색하고, 뷰 생성 정의문에서 조건이 되는 술어를 우선 분리하여 분리 된 가장 간단한 셀렉션 질의를 서버로 요구하여 필요한 영역의 데이터만을 얻는다. 로컬서버는 각각의 서버로부터 전송받은 결과 데이터들을 이용하여 최종 연산을 수행하고, 생성 된 공간뷰를 실체화 시킨다. 이때, 뷰 정의문에 제시 된 어트리뷰트만이 아니라 공간테이블의 모든 어트리뷰트를 실체화시킨다. 이는 같은 영역에 대한 다른 뷰 정의시 데이터의 중복 저장을 방지하여 로컬서버내의 공간데이터의 활용도를 높이기 위함이다. 그리고 원격 서버에서의 1차적인 질의 처리 후 데이터의 전송은 대용량 공간데이터에 대한 크기를 줄여 전송 부하를 최소화하는 장점을 갖는다. 또한 로컬 서버의 공간 뷰 실체화는 공간 뷰 생성에 따른 원격서버로의 잦은 데이터 접근을 방지하여 각 서버간의 처리비용을 줄이고, 사용자 질의처리에 대한 응답시간을 최소화하여 사용자의 만족도를 높인다.

본 논문의 구성은 다음과 같다. 2장에서는 공간 뷰의 개념과 LDAP(LightWeight Directory Access Protocol)에 대해 기술하고, 3장에서는 멀티서버 환경에서의 공간 뷰 실체화 기법에 대해서 제안한다 마지막으로 4장에서는 결론을 맺는다.

2.관련연구

2.1 공간 뷰

지리 정보 시스템(GIS: Geographic Information System)에서 공간 객

[1] 본 연구는 정보통신부의 대학 S/W 연구센터 지원사업의 연구 결과임

체를 모델링 할 때 다양한 사용자 요구 조건들이 있으며, 특히 사용자의 관점(User's perspective view)에 따른 서로 다른 공간 표현(spatial representation)을 지원하는 것이 매우 중요하다. 즉, 공간 데이터베이스에서는 동일한 공간 객체가 사용자들의 관점에 따라 상이한 공간 표현을 가질 수 있다.[4]

공간 뷰 역시 관계형 DBMS에서의 전통적인 뷰와 같이 사용자에게 동일한 데이터를 다양하게 보여주는 기능, 질의에 대한 매크로 기능, 그리고 보안 기능등을 제공한다. 기존의 뷰와 다른 점은 공간 뷰는 공간 테이블에 포함 된 공간 속성을 이용하여 정의하고, 데이터베이스의 일부를 보여주는 윈도우 기능 뿐만 아니라, 데이터베이스에 저장 된 공간 객체에 대해 다양한 공간연산을 통해 사용자 요구를 만족시키는 새로운 객체를 생성할 수 있다.

따라서 공간 뷰의 유지 및 관리를 위해서는 기본테이블로부터 유도된 공간객체에 대한 관리 및 유도 된 객체에 대해 공간연산을 통한 새로운 객체 생성에 대한 고려를 해야 한다. 사용자의 관점에 따른 다양한 공간 표현을 위한 공간연산의 제공은 공간 뷰를 기존의 뷰와는 다른 방법으로 관리해야 한다.

2.2 LDAP
(LightWeight Directory Access Protocol)

디렉토리 서비스는 다양한 색인, 캐싱, 디스크 접근 기술들을 통하여 네트워크의 다양한 지원과 관련된 정보를 논리적으로 접근 할 수 있도록 지원한다. 디렉토리 서비스는 정보를 빠르게 읽을 수 있도록 고안된 특별한 형식의 데이터베이스이다. 특정분야의 디렉토리 서비스의 이용 및 개발 요구가 높아짐에 따라 개발 된 X.500은 최초의 일반적인 목적의 디렉토리 시스템이었고 다양한 질의를 사용하는 강력한 검색기능을 제공하였을 뿐만 아니라 서버와 데이터의 분산이 용이하고, 특히 운영체제나 네트웍, 응용프로그램에 독립적으로 사용될 수 있는 표준이었다. 하지만, DAP(X.500의 directory client access protocol)은 양적인 면에서 너무 방대했고 구현상의 복잡성 때문에 DAP의 기능을 대부분 지원하면서도 낮은 사용빈도수의 기능은 축소시키거나 제거한 LDAP이 등장하였다. LDAP은 단순하면서도 속도에 최적화된 프로토콜을 사용함으로써 기업의 정보 등을 포함한 다양한 정보를 저장할 수 있으며 이러한 정보를 검색, 변경,삭제할 수 있다.[4]

3. 멀티서버 환경에서의 공간 뷰 실체화 기법

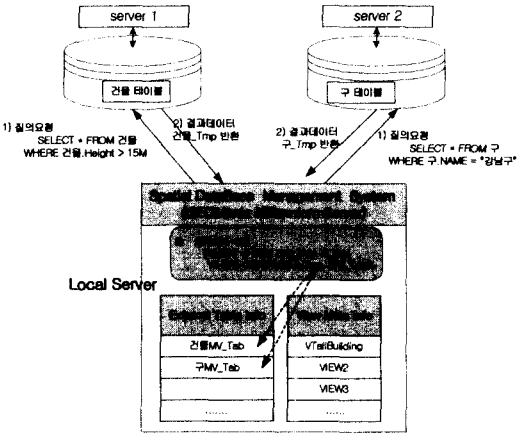
본 장에서는 멀티서버 환경에서의 원격서버의 데이터를 이용한 공간 뷰 생성 기법에 대해 자세히 서술한다. 이를 위해 먼저 사용되는 자료 구조에 대해 설명하고, 세부적으로 공간 뷰 생성 기법에 대해 설명한다.

원격 서버의 공간 데이터를 이용한 공간 뷰의 생성은 다음과 같은 단계로 구성되며, 각 단계는 첫째, 로컬 서버는 뷰 생성 정의문에 따라 조건이 되는 술어를 단순히 각각의 테이블에 따라 분리한다.그리고 Extended Spatial Data Directory Service Server에게 필요한 공간데이터를 소유한 서버의 검색을 요청한다. 둘째, 검색을 마친 Extended Spatial Data Directory Service 서버는 필요한 공간테이블을 갖고 있는 원격서버 주소 리스트를 반환한다. 셋째, 로컬 서버는 검색 된 서버가 여러 개 일 경우는 가장 근거리 서버에 직접 접속하여 분리한 술어 질의를 각각의 서버에 보낸다. 넷째, 원격 서버는 로컬 서버에서 요청한 질의를 처리하여 결과 데이터를 로컬 서버로 전송한다. 마지막으로 로컬 서버는 전송 받은 테이블들을 뷰 정의문에 따라 최종 연산 후 결과를 서버 내에 실체화 시킨다. 이 때, 해당 뷰 뿐만 아니라 같은 영역에 대한 다른 뷰 정의시 데이터의 중복 저장을 방지하기 위해 뷰 정의문에 제시 된 어트리뷰트만이 아니라 공간테이블의 모든 어트리뷰트를 저장시킨다.

다음의 [그림3_1]은 본 논문의 효율적인 이해를 위해 [예1]과 같은 일반적인 공간 뷰 생성 요청 시 각 서버간 뷰 생성 처리과정을 흐름도로 표현한 것이다.

[예1]

```
CREATE Spatial_View VTallBuilding AS
SELECT 건물.Name, 건물.Height from 건물, 구
WHERE 건물.Height > 15M AND 구.Name = " 강남구"
AND CONTAINS(구.Obj, 건물.Obj);
```



[그림3_1] 멀티서버 환경에서의 각 서버간 공간 뷰 실체화 과정

3.1 공간데이터 공유를 통한 공간 뷰 생성 시 필요한 각 서버간 자료구조

본 논문에서 제안하는 기법을 지원하기 위해서는 각 서버 간 공간데이터 공유를 위한 메타테이블의 관리 및 자료구조, Extended Spatial Data Directory Service Server에서 로컬서버로의 전송패킷 구조, 로컬 서버에서의 공간 뷰 저장구조 및 공간 뷰 관리를 위한 메타테이블에 대한 정의가 필요하다.

3.1.1 Extended Spatial Data Directory Service Server

각 서버간 공간데이터의 공유를 위해 Extended Spatial Data Directory Service 서버에서는 각 서버가 공유하고 있는 공간 테이블들을 관리하고 테이블 리스트를 공개한다.

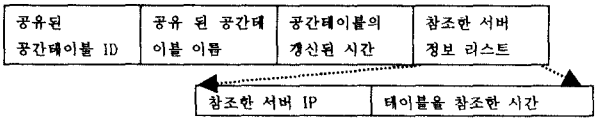
다음은 로컬 서버의 공간테이블 검색에 요구 시 필요한 공간테이블을 소유한 서버 리스트를 반환하는 결과 패킷의 자료구조이다.

테이블 이름	테이블소유 서버IP 리스트

[그림3_2] Extended Spatial Data Directory Service Server에서 로컬서버로의 전송 패킷 구조

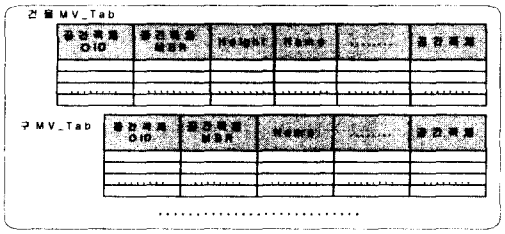
3.1.2 공간 뷰 실체화를 위한 각 서버간 자료구조

각 서버에서는 데이터의 일관성 유지를 위해 공유한 테이블에 대해 테이블을 참조한 서버정보와 참조한 시간, 그리고 테이블의 변경 시간을 관리한다. 로컬 서버는 전송 받은 데이터에 대해 뷰 정의문에 의해 처리 후 연산결과를 공간데이터의 특성 및 관리의 용이성을 고려하여 각 테이블 단위로 저장한다. 그리고 해당 뷰 뿐만 아니라 동일 영역에 대한 다른 공간 뷰 정의 시 서버 내 데이터의 활용도를 높이고 저장 공간을 절약하기 위해 뷰 정의문에 따른 연산결과에 대해 전체 어트리뷰트를 저장한다. 그리고 각각의 실체화 된 뷰에 대해 질의처리를 수행하기 위해서 저장한 데이터와 공간 뷰의 연관성을 관리하는 공간 뷰 메타 테이블을 관리한다.



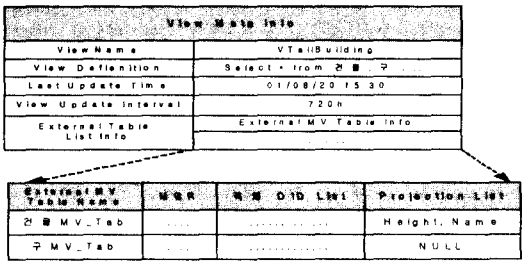
[그림3_3] 원격서버에서 관리하는 공간테이블을 참조한 테이블들의 정보를 위한 메타테이블 구조

1차적으로 술어 분리를 통한 질의를 원격 서버에 요구하여 필요한 영역의 데이터만을 전송받고 전송 받은 결과 테이블을 이용하여 뷰 정의문의 최종 연산을 수행하여 뷰를 생성한다. 이 때, 공간테이블이라는 특성 상 하나의 테이블로 관리하는 것은 무의미하므로, 각 테이블 별로 관리한다.



[그림3_4] 로컬서버에 저장 된 Materialized View Table 모습

다음은 [그림3_5]는 로컬서버에서 실체화 된 뷰를 관리하는 뷰 메타테이블이다. 뷰의 최근 갱신시간과 갱신주기를 통해 공간 뷰 데이터의 정확성을 보장하고 External Table List Info 구조체는 저장 된 데이터와 뷰와의 연관성을 관리한다.



[그림3_5] 로컬 서버의 뷰 메타 정보 관리 테이블

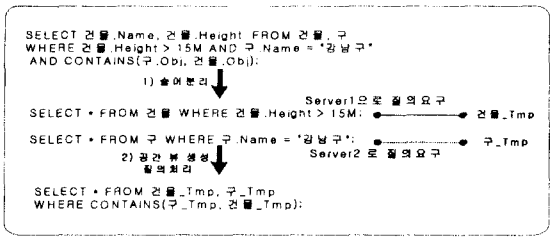
3.2 멀티서버 환경에서의 공간 뷰 생성 및 실체화 기법

본 논문에서 제안하는 멀티서버 환경에서의 공간데이터 공유를 통한 공간 뷰 실체화 기법은 다음과 같은 과정으로 처리된다.

[예1]과 같은 공간 뷰 생성문이 정의되어졌을 때 로컬서버는 뷰 생성 조건의 술어를 [예2]와 같이 분리한다.

[예2]에서와 같이 분리 된 술어 질의를 각각의 서버로 요청하여 결과 데이터를 전송 받는다. 이는 전송받아야 하는 데이터의 크기를 줄여 대용량 공간데이터의 전송에 따른 네트웍 부하를 최소화 하고 로컬서버의 질의처리 비용을 각각의 서버로 분산시켜 뷰 생성 시간을 최소화한다. 일반적인 관계형 릴레이션과 달리 서로 다른 두 공간 테이블을 하나의 테이블로 관리되는 것이 무의미하므로 뷰 생성을 위한 최종 연산 후 연산 결과를 각각의 테이블로써 [그림3_4]와 같이 저장한다. 이때, 뷰 정의문에서 제시한 어트리뷰트 뿐만 아니라 전송받은 테이블의 모든 어트리뷰트를 함께 저장하여 동일한 공간객체에 대해 다른 공간 뷰 정의 시 이미 저장 된 공간데이터를 사용하여 대용량 공간데이터의 중복저장을 방지한다.

[예2]



3.3 동일 테이블에 대한 다른 뷰 생성 요구 시

공간 뷰 생성 처리에 관한 일반적인 단계를 거쳐 각각의 서버로부터 전송받은 데이터로 최종연산을 수행하여 뷰 생성 조건에 부합되는 데이터들만을 추출한다. 이 데이터를 기존에 저장 된 테이블에 삽입연산을 이용하여 추가시킨다. 이때 테이블의 기본 키인 객체OID 를 검색하여 동일 OID 인 경우는, 즉 동일 객체인 경우는 삽입연산을 수행하지 않고 기존에 저장 된 데이터를 사용한다.

4. 결론

본 논문에서는 멀티서버 환경에서의 공간데이터 공유를 통한 공간 뷰 정의 시 각 서버로부터 전송 받은 데이터를 이용한 로컬서버에의 공간 뷰 실체화 기법을 제안하였다. 분산 된 공간데이터의 정보를 검색하기 위해 Extended Spatial Data Directory Service를 이용하였다 이를 위하여 Extended Spatial Data Directory Service 서버와 원격서버에서 관리해야 할 메타테이블을 정의하고, 로컬 서버에서의 외부 테이블 저장 방법과 공간 뷰 메타테이블을 정의하고 처리과정을 기술하였다. 제안 된 기법은 공간 뷰 정의문의 술어 분리를 통해 원격서버로의 단순한 질의 처리에 의한 결과 공간데이터를 로컬서버로 전송하고, 로컬서버에서는 얻은 데이터를 이용하여 공간 뷰 생성을 위한 최종 연산을 수행하여 실체화 시킨다. 실체화 시키는 데이터는 공간테이블의 특성 상 각각의 테이블로 관리하고 동일 영역에 대한 다른 공간 뷰 생성시 기존의 저장 된 데이터를 사용하기 위해 뷰 생성 조건을 만족하는 레코드의 모든 어트리뷰트를 저장한다. 따라서 다음과 같은 장점을 갖는다. 첫째, 원격서버에서의 질의처리 수행에 의해 한 단계 여과된 공간 데이터의 전송은 전송량에 따른 전송부하를 줄인다 둘째, 술어분리에 의한 원격서버로의 질의처리 요구는 로컬서버에서의 질의처리 비용을 분산시키는 장점을 갖고 있다. 셋째, 공간 뷰 생성 조건을 만족하는 어트리뷰트만이 아닌 해당 레코드의 모든 어트리뷰트의 저장은 동일 영역에 대한 다른 공간 뷰 생성시 기존의 저장 된 데이터를 사용함으로써 객체의 중복저장을 최소화한다. 넷째, 공간 뷰 관리를 위한 최근갱신시간과 갱신 주기는 공간 뷰의 데이터 정확도 중요 여부에 따라 원격서버 데이터와의 일관성을 유지할 수 있다. 마지막으로 로컬서버에 공간 뷰의 실체화는 각 서버로의 잦은 데이터 접근을 방지하여 통신부하를 줄이고, 질의처리에 대한 각 서버의 처리비용 제거에 따른 질의처리 응답시간을 최소화하여 사용자 만족도를 높일 수 있는 장점을 갖는다.

향후 연구과제로는 본 논문에서 제시한 공간 뷰 실체화 시 각 서버간의 최소의 부하로 유지 될 수 있는 일관성 유지 기법과 서버간 공간 뷰 공유 방안에 대한 연구가 이루어져야 한다.

참고문헌

[1]" GIS 정보유통을 위한 한국형 모델개발 연구" 국토지리원 1999.12.

[2] 김태연,정보홍,이재동,배해영" 서버 처리비용 분산을 위한 공간뷰 클라이언트 실체화 기법" 한국정보과학회 2001 봄 학술발표논문집(B) 제28권 제1호 p.211-213

[3] Lightweight Directory Access Protocol(v3)(RFC 2251)

[4] Claramunt C. and Mainguenaud M.," Identification of Definition Formalism for a Spatatial View" ,Advanced Geographical Modeling,1994