

공간 데이터베이스의 중복 제어를 위한 Floating Primary Copy 기법

최진오*
부산외국어대학교 컴퓨터공학과

Floating Primary Copy Method for Replica Control of Spatial Data

JinOh Choi

Dept. of Computer Engineering, Pusan University of Foreign Studies

E-mail : jochoi@taejo.pufs.ac.kr

요 약

GIS 응용에서 공간 객체들이 여러 사이트에 걸쳐 중복 저장되어 있다면, 이 객체들의 수정 시 일관성 제어를 위한 중복 제어 기법이 필요하다. 그러나, 일반적인 비즈니스 데이터를 위한 전통적인 중복 제어기법은 공간 데이터에는 그대로 사용될 수 없다. 공간 데이터의 수정은 긴 트랜잭션을 필요로 하며, 또한 일반 데이터와 달리 데이터간의 공간 종속성을 지니기 때문이다. 본 논문은 기존의 Primary Copy 중복 제어 기법을 확장하며, time-stamp에 기반하여 Primary Copy 사이트가 변경되는 Floating Primary Copy 기법을 제안한다. 본 논문에서 제안하는 중복 제어 기법은 Primary Copy 사이트를 동적으로 변경함으로써, 긴 트랜잭션의 중복 일관성 제어를 지원할 수 있다.

1. 서론

GIS 응용 시스템은 일반 응용 시스템과 다른 여러가지 특성을 가지고 있다. 먼저, 이 응용 시스템은 각 서비스 시설물에 대하여 행정 지역별로 별도의 기관에서 공간 데이터베이스를 관리므로 분산 데이터베이스 시스템이 요구된다. 즉, 분청과 지역청 모두 독립된 데이터베이스를 지니는 환경이다. 둘째, 이러한 구조에서 각 기관사이의 데이터의 중복이 불

가피하다. 셋째, 각 사이트에서의 공간 데이터 수정은 긴 작업 시간을 필요로 하는 긴 트랜잭션(Long Transaction)처리가 필요하다[1][3]. 이러한 특징을 가진 응용은 분산 환경에서의 긴 트랜잭션에 대한 중복 데이터베이스 일관성 제어를 필요로 한다.

이러한 구조를 가지는 응용 시스템에서, 각 사이트에서 수행되는 지도 수정 작업은 긴 트랜잭션이기 때문에 동시성 제어와 일관성 제어가 기존의 전통적

* 이 논문은 2001년도 부산외국어대학교 학술연구조성비에

의해 연구되었음

기법으로 해결되지 않는 문제점을 지닌다[1]. 본 논문에서 대상으로 하는 환경의 새로운 요구사항을 정리하면 다음과 같다. 첫째 분산환경에서 동시성이 보장되어야 한다. 즉, 오랫동안 기다려야 하는 트랜잭션이 없어야 하며 다른 사이트에서 긴 트랜잭션이 수행 중이더라도 중복 데이터에 대한 접근이 허용되어야 한다. 둘째, 긴 트랜잭션의 수행 결과가 취소되어서는 안 된다. 긴 트랜잭션의 수행 결과를 취소하기에는 작업한 내용과 시간이 너무 값비싸므로 이러한 상황이 발생되지 않도록 하여야 한다.

이러한 요구 사항을 만족하기위해서 본 논문에서는 primary copy 중복 일관성 제어 기법에 기초하여 확장하는 접근 방법을 취한다. 확장 내용은 첫째, 긴 시간 blocking을 피하고, 큰 공간데이터의 잠금 단위를 축소하기 위하여, 잠금 개념(locking concept)을 확장한다. 둘째, 긴 트랜잭션의 취소를 피하기 위해 점진적 변경 전파 프로토콜로 확장한다.

본 논문의 구성은 다음과 같다. 2장에서 관련 연구, 3장에서 잠금의 확장을, 4장에서 primary copy 기법의 확장 방법을 소개하며, 5장에서 구체적인 프로토콜을 설명한다. 6장에서 시스템 설계 내용과 프로토콜의 예를 보이고 7장에서 결론을 맺는다.

2. 관련 연구

기존의 중복 제어 방법은 크게 두 가지로 나뉜다. 첫째, 비동기식 제어(asynchronous control) 방법은 primary copy 방법으로 대표되는데, 모든 중복 사이트들이 가용하지 않아도 수정할 수 있으며 late update propagation을 지원하므로 변경의 성능(update performance)이 좋다는 장점이 있는 반면 한동안 전체 시스템의 무결성이 깨질 수 있다는 단점(concurrency anomalies)이 있다. 둘째, 동기식 제어(synchronous control) 방법으로서, 모든 수정을 전체 분산 사이트들에 걸쳐 동기화 한다. 이 접근 방법은 완벽한 무결성을 제공하지만 제약 조건이 너무 엄격하여 성능이 떨어지고 통신 부하(overhead)가 큰 단점이 있다. 그래서 정족수 기반 기법(quorum-based scheme), causality 기법 등의 변형이 제시되었다.

이러한 기존의 중복 제어 기법들은 전통적 관계

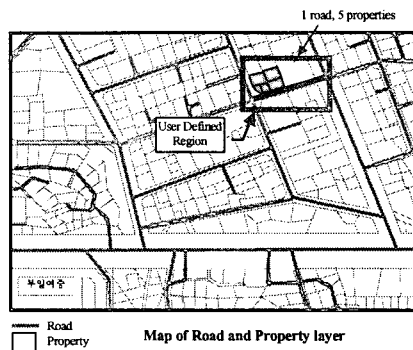
형 데이터베이스 응용에는 충분한 해결방안을 제시하였으나 공간 데이터베이스와 같은 새로운 요구사항을 가진 응용에서는 기존의 기법들을 그대로 적용할 수 없는 문제점이 있다. 동기식 제어방법은 잠금을 사용하기 때문에 긴 트랜잭션의 실행은 너무 오랫동안 다른 트랜잭션들을 기다리게 하는 문제점이 있다. 비동기식 제어 방법도 변경 전파 시 변경 상충이 발생할 수 있으며 이로 인한 수정 내용이 취소될 수 있다.

[1]은 본 논문과 동일한 문제를 다루지만, [1]에서 제시한 협동 작업 프로토콜은 동기식 중복 제어 방식에 기초하여, 모든 사이트가 중복 일관성 제어에 책임을 지도록 한다. 이것은 자신이 수정한 내용을 책임지고 다른 모든 사이트로 전파해야 하기 때문에 다른 사이트들이 종료될 때까지 불필요한 대기를 하여야 하는 문제점이 있었다. 이 논문에서는 timestamp에 기반 한 primary copy 개념을 도입하여 이러한 문제점을 해결하고자 한다.

3. 잠금의 확장

이 장에서는 기존 primary copy에서 사용하는 잠금 기법을 확장하여 blocking 조건을 완화한다.

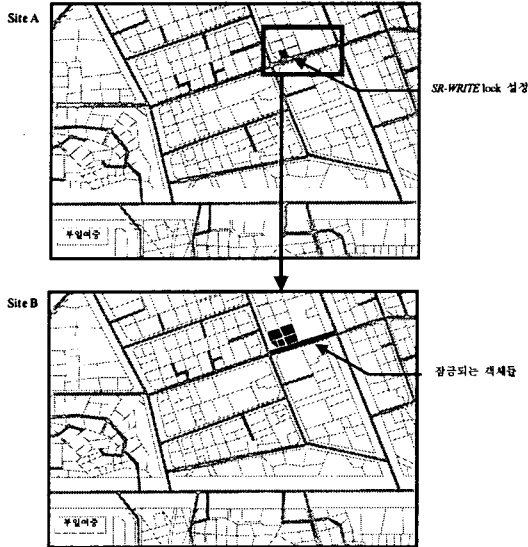
영역 잠금(Region READ Lock): 전체 데이터 집합 중에서, 트랜잭션 T_i 를 실행하는 사용자가 MBR로 정의한 부분 영역(sub region)에, 완전히 포함되는 객체들에 대하여 설정하는 잠금. 하나의 객체 혹은 객체 그룹에 대하여 향후 SR-WRITE lock을 설정할 것임을 암시.



[그림 1] 영역 잠금의 예

SR-WRITE 잠금(Spatial Relationship-bound WRITE

Lock) : 트랜잭션 T_i 의 영역 잠금이 걸린 객체 집합들에 설정하는 잠금. 잠금을 걸고자 하는 객체뿐만 아니라 그 객체와 공간관련성 관계[1]에 있는 객체도 수정중이라면 wait해야 하는 lock 모드.



[그림 2] SR-WRITE 잠금의 예

이러한 잠금 기법을 이용한 몇가지 용어를 정의한다.

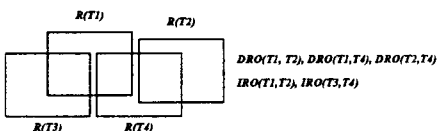
Mid-merge Delta : 트랜잭션의 시작 또는 직전의 mid-merge point로부터 현재의 mid-merge point까지 수정한 내용

- $MD_i(T_x)$: T_x 트랜잭션의 i 번째 mid-merge delta

Region lock area : 한 트랜잭션에 의해 영역 잠금이 설정된 객체들을 완전히 포함하는 MBR(Minimum Boundary Rectangular).

- $R(T_x)$: T_x 트랜잭션의 영역 잠금 MBR

region Overlap Relationship : 영역 잠금들 사이의 중첩관계. 직접 중첩 관계(DRO : Direct Region Overlap)와 간접 중첩 관계(IRO : Indirect Region Overlap)로 나뉜다.



[그림 3] region overlap relationship

- $DRO(T_x, T_y)$: 트랜잭션 T_x 와 T_y 는 영역 잠금의

관계가 직접 중첩 관계임.

- $IRO(T_x, T_y)$: 트랜잭션 T_x 와 T_y 는 영역 잠금의 관계가 간접 중첩 관계임.

Region Overlap Areas : 한 트랜잭션이 설정한 영역 잠금과 중첩 관계를 가지는 모든 영역 잠금들의 영역 합. 트랜잭션들의 시작과 종료에 의해 동적으로 변경됨.

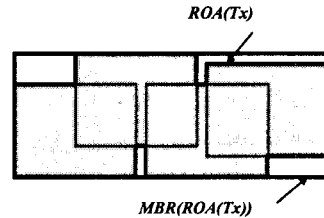
- $ROA(T_x)$: T_x 트랜잭션과 직접 또는 간접 중첩

관계를 가지는 모든 영역 잠금들의 영역 합.

- $MBR(ROA(T_x))$: $ROA(T_x)$ 의 MBR

Region Overlap Transactions

- $ROT(T_x)$: $ROA(T_x)$ 내의 모든 관련 트랜잭션들 (각 영역 잠금을 정의한 트랜잭션들)의 집합



[그림 4] Region Overlap Areas

4. Primary Copy 제어 방법의 확장

비동기식 중복 일관성 제어기법은 동기식 제어 기법에 비해 많은 제약사항이 완화되었지만 긴 트랜잭션을 지원하기 위해서는 제약 사항을 더 완화하는 확장이 필요하다. 즉, primary copy가 고정되어 있기 때문에 primary copy에 부하가 집중되고 primary copy가 고장이 날 경우 다른 트랜잭션들의 수정이 불가능해진다. 또한 primary copy에 잠금이 오랫동안 설정되므로 하나의 긴 트랜잭션이 다른 트랜잭션들의 수정을 오랫동안 기다리게 만드는 문제가 있다. 본 논문에서는 primary copy 사이트가 time-stamp에 의해 결정되고 변경되도록 확장하고 영역 잠금과 협동 작업 프로토콜을 도입하여 긴 트랜잭션을 지원한다.

Primary Copy Transaction : $ROT(T_x)$ 중에서 $TS(T_i)$ 가 최소인 트랜잭션 또는 그 트랜잭션이 실행되는 사이트.

- $PCT(ROT(T_x)) : PCT(ROT(T_x)) = \{ T_i \mid \text{Min}(TS(T_i)) \}$

Semi-primary Copy Transactions : $ROT(T_x)$ 중에서 primary copy transaction을 제외한 나머지 모든 트랜잭션들.

- $SCT(ROT(T_x))$: T_x 트랜잭션과 동일한 ROA 를 가지는 ROT 중에서 $TS(T_i)$ 값이 최소가 아닌 트랜잭션 또는 그 사이트.

Time Stamp : 한 트랜잭션이 영역 잠금을 정의한 시각.

- $TS(T_x)$: 트랜잭션 T_x 의 time stamp

본 논문에서 제시하는 확장된 primary copy 중복 일관성 제어 방법을 부동(floating) primary copy 중복 일관성 제어 기법이라 부른다.

Primary copy 사이트의 결정과 변경은 아래의 조건을 따른다.

1. 트랜잭션 T_x 가 영역 잠금을 설정할 때, $ROT(T_x)$ 중에서 time-stamp값이 최소인 트랜잭션이 primary copy 트랜잭션이 됨.

2. $PCT(ROT(T_x))$ 이 완료 후 종료할 경우, $ROT(T_x) \neq \emptyset$ 이면 $SCT(ROT(T_x))$ 중에서 $TS(T_x)$ 가 최소인 트랜잭션이 새로운 primary copy 트랜잭션이 됨

Primary copy 트랜잭션을 중심으로 한 협동 작업 프로토콜은 다음과 같다.

1. 한 트랜잭션 T_x 가 한 ROT 에 참여(영역 잠금을 설정)할 때 primary copy 트랜잭션으로부터 time-stamp를 부여 받고 semi-primary copy 트랜잭션이 됨. 단, ROT 가 존재하지 않을 경우 자신이 primary copy가 됨.

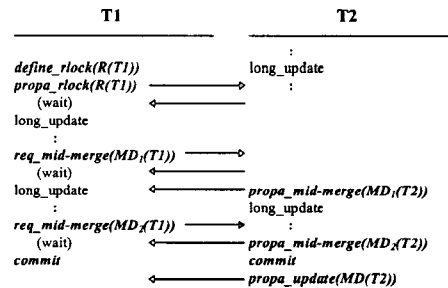
2. Semi-primary copy는 mid-merge를 수행하고자 할 때 primary copy에게 mid-merge delta를 전파함. Primary copy는 전파 받은 mid-merge가 충돌이 없을 경우 변경을 확정하고 다른 사이트로 전파하여 중복 일관성 제어를 수행함. 충돌이 있을 경우, time-stamp 우선 순위에 의해 충돌을 해결함.

3. Primary copy의 mid-merge 수행도 semi-primary copy와 동일하게 실행함.

이 장에서는 분산 트랜잭션들이 영역 잠금을 설정하고 primary copy로 mid-merge를 전파하는 등의 프로토콜을 정의하고 설명한다.

본 논문에서 제안하는 트랜잭션 모델은 5단계의 트랜잭션 연산을 수행한다. 각 단계의 트랜잭션 연산에 대한 설명은 다음과 같으며 그림 5에서 예를 보이고 있다.

1. **define_rlock**:작업 예정영역에 영역 잠금 지정.
2. **propa_rlock**:중복 사이트들로 영역 잠금 전파. 중복 사이트들의 응답을 받아 트랜잭션 모드 결정.
3. **propa_mid-merge/req_mid-merge**:primary copy로 mid-merge 요청.
4. **commit**:기 트랜잭션의 완료. region lock을 해제
5. **propa_update**: ROT 에 자신 이외의 사이트가 존재하지 않을 경우 primary copy가 수정된 모든 내용을 중복 사이트로 전파하여 중복 일관성 제어를 수행함.



[그림 5] 트랜잭션 연산의 예

그림 6은 위 프로토콜 연산 중에서 propa_mid-merge와 req_mid-merge의 알고리즘을 보이고 있다.

```

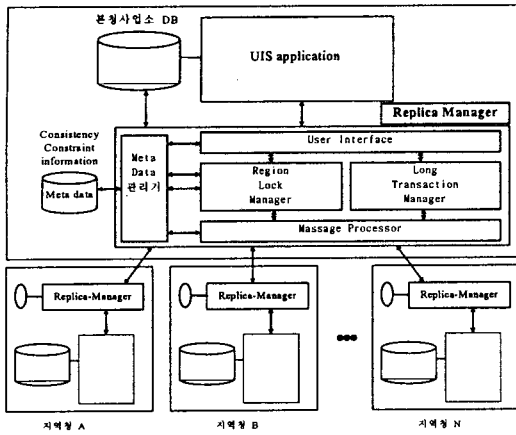
propa_mid-merge(Md(Ty)) {
// SCT들의 mid-merge 요청을 처리하는 PCT 연산.
//Tx:PCT, Ty:mid-merge를 요청한 SCT, T:Ty를 제외한 SCT
if (DRO(Tx,Ti) 인 Ti가 존재하면)
    send Md(Ty) to Ti
}

req_mid-merge(Md(Ty)) {
// PCT에게 mid-merge를 요청하는 SCT 연산.
//Tx:PCT, Ty:mid-merge를 요청한 SCT, T:Ty를 제외한 SCT
if (DRO(Tx,Ti)|| IRO(Tx,Ty))
    result=send Md(Ty) to Tx //PCT에게 변경 요청
if(result==conflict) rollback
else commit
}
    
```

[그림 6] propa/req_mid-merge 연산

5. 프로토콜

6. 시스템 구조 설계



[그림 7] 시스템 구조

그림 7에서 시스템 구조를 보이고 있다. 본 논문에서 연구하고 설계한 내용은 Replica Manager와 메타 데이터이다. 메타 데이터 및 관리기는 분산 중복 정보와 프로토콜 실행 중 동적으로 변경되는 정보를 저장하고 질의하는 기능을 담당한다.

Replica Manager에서, 사용자 인터페이스(user interface)는 프로토콜의 실행 중에 사용자의 입력과 출력을 담당한다. 사용자 입력이란, 영역 잠금의 입력, 긴 트랜잭션의 mid-merge point 지정, 그리고, commit 실행 등이다. 그리고, 사용자 출력이란 mid-merge 결과를 primary copy로부터 통보 받고 다른 트랜잭션의 mid-merge 결과를 임시로 출력해보는 경우이다. 영역 잠금 관리기(region lock manager)는 자신의 영역 잠금과 다른 사이트의 영역 잠금들의 관계 정보를 설정하고 저장한다. 이 정보는 다른 사이트들의 수정을 허용할 지 금지할 지 결정하는데 이용된다. 긴 트랜잭션 관리기(long transaction manager)는 트랜잭션의 프로토콜 연산의 실행 순서를 제어하며 트랜잭션의 실행, 종료, 그리고 회복을 책임진다. 메시지 처리기(message processor) 분산 사이트 간의 메시지 패킷 전송을 책임지는 모듈이다.

7. 결론

본 논문은 지도를 수정할 때 발생하는 긴 트랜잭션들의 동시성과 중복 일관성을 제어 문제를 다루

었다. [1]의 연구에서 공동작업을 수행하는 모든 트랜잭션들이 동시에 종료되어야 한다는 문제점을 본 논문에서는 비동기식 중복 일관성 제어 방법인 primary copy 기법을 확장하여 해결하는 것을 보였다.

본 논문이 제안하는 기법은 영역 잠금을 이용하여 중첩 지역을 수정하는 트랜잭션들을 하나의 공동작업 단위로 보고, 이 중에서 time-stamp에 따라 primary copy가 결정되어 중복 제어를 수행하게 한다. 공동작업에 참여하는 트랜잭션들의 중간 수정 결과는 언제든지 primary copy에 병합되며, 이때 발생할 수 있는 변경 상충은 time-stamp 우선순위에 의해 primary copy가 해결하고 통보한다. Primary copy가 종료하고자 할 경우는 다른 사이트가 primary copy의 권한과 책임을 승계하며 최후의 트랜잭션이 종료된 후 primary copy에 병합되었던(충돌이 없음을 주목) 변경 내용을 분산 중복 사이트로 복사하여 중복 일관성 제어를 수행한다.

본 논문은 이러한 제어를 위해 용어를 정의하고 이 용어를 사용하여 프로토콜 연산을 개발하고 알고리즘을 제시하였다.

[참고문헌]

- [1] Jin-oh Choi, Young-san Shin, Bong-hee Hong, "Update Propagation of Replicated Data," in Lecture Notes in Computer Science 1677, pp.952-963
- [2] J.Gray, P.Helland, D.Shasha, "The Dangers of Replication and a Solution," in Proc. Int. Conf. on ACM SIGMOD, 1996, pp173-182
- [3]F.Bancilhon, W.Kim, H.Korth, "A Model of CAD Transactions," in Proc. Int. Conf. on VLDB, 1985, pp25-33
- [4] E.Pitoura, "A Replication Schema to Support Weak Connectivity in Mobile Information Systems," in Int. Workshop on DEXA, 1996, pp708-717
- [5] P.Bernsten, N.Goodman, "An Algorithm for Concurrency Control and Recovery in Replicated Distributed Databases," in ACM Tran. Database Systems, Vol9, no.4, 1984, pp596-615