

# 이동 클라이언트의 공간 데이터 변경을 위한 대리 트랜잭션 모델

문선희<sup>○</sup>      반재훈      홍봉희  
부산대학교      컴퓨터공학과

## Surrogate Transaction Model for Update Spatial Data of Mobile Clients

Sun-Hee Moon<sup>○</sup>      Chae-Hoon Ban      Bong-Hee Hong  
Dept. of Computer Engineering, Pusan National University

### 요 약

이동 컴퓨팅 환경이 급속하게 발전하면서 GIS 분야에서도 이동 클라이언트를 이용한 다양한 서비스에 대한 연구가 진행되고 있다. 특히, 이동 클라이언트를 이용한 공간 데이터 변경의 경우에 동시성 제어를 수행해야 하는데 기존 클라이언트-서버 환경에서 사용되는 잠금 기법은 잠금 획득을 위한 긴 대기시간 때문에 적합하지 않다.

본 논문에서는 이동 컴퓨팅 환경에서의 공간 데이터 변경을 위한 3-계층 구조를 제시하며 이 구조에서의 대리 트랜잭션 모델을 정의한다. 또한, 이동 클라이언트의 공간 데이터 변경 트랜잭션의 동시성 제어를 위해 전통적인 낙관 기법을 확장하여 공간 데이터에 적합한 확인 기법을 제시하며 동시성 제어에서 트랜잭션 직렬화 보장의 예를 보인다.

## 1. 서론

이동 컴퓨팅 환경이 급속하게 발전하면서 사용자에게 언제, 어디서나 필요한 컴퓨팅 환경을 제공하기 위해 휴대폰이나 PDA와 같은 이동 단말기에서 제공되는 서비스에 대한 연구가 활발히 진행중이다(1). 특히 GIS 분야에서 이동 단말기는 실시간으로 정확한 데이터의 입력 및 검증을 위해 다양하게 사용될 수 있다.

본 논문에서는 이동 단말기에서의 공간 데이터 변경을 위해 3-계층 구조인 S-S-M(Server-Surrogate PC-Mobile Client) 환경을 제시한다. S-S-M 구조는 공간 데이터베이스를 가진 서버와 공간 데이터 변경을 담당하는 이동 클라이언트 및 이동 클라이언트의 변경 트랜잭션을 위임하여 수행하는 대리 PC로 구성된다.

이 환경에서 공간 데이터 변경 트랜잭션의 동시성 제어를 위해 기존의 클라이언트 서버 모델에서 적용되는 잠금 기법을 적용하는 경우에는 잠금 획득을 위한 긴 대기 시간으로 인해 시스템의 성능이 현저히 떨어지게 된다. 따라서 본 논문에서는 S-S-M 환경을 고려하여 전통적인 낙관적 병행 기법인 확인 기법을 확장하고 이 기법을 적용시키기 위한 트랜잭션 모델을 제시한다. 또, 확인 기법을 적용한 트랜잭션 모델에서 트랜잭션간 인터리빙 관계에 대한 직렬성 보장을 예를 들어 설명하였다.

본 논문의 구성은 다음과 같다. 2장에서 본 논문과 관련된 연구를 소개하고 3장에서는 S-S-M 환경에서의 공간 데이터 변경을 위한 대리 트랜잭션 모델을 제시한다. 4장에서는 대리 트랜잭션 모델에서의 동시성 제어를 위한 확장된 확인 기법 및 직렬화 예제를 설명하며 5

장에서는 결론을 기술한다.

## 2. 관련연구

기존의 LAN 환경과는 달리 이동 컴퓨팅 환경은 예측할 수 없는 끊김 등을 고려한 계층 구조가 적합하다. Client-Agent-Server의 3-계층 구조는 서버의 데이터를 이동 클라이언트에 저장하는데 시간을 절약하고 통신으로 인한 단말기의 배터리 소모를 최소화 한다(2). 이 구조에서는 잠재적으로 무한정의 대기시간(unbounded delay)을 가지는 2PL(two phase locking)의 방법보다 이동 컴퓨팅 환경이 가지는 이동성을 보장하여 잦은 끊김을 고려한 낙관적인 동시성 유지 방법이 효과적이다(3)(4).

이러한 낙관적 기법은 유연성 있는 동시성 유지 기법이지만 일관 충돌이 발생하면 복귀는 불가피하다. 따라서 관계형 데이터베이스에서는 긴 트랜잭션을 서브-트랜잭션으로 나누어 부분 복귀(partial rollback)함으로써 복귀 비용을 최소화하였다(5).

## 3. 대리 트랜잭션 모델

이 장에서는 본 논문에서 고려하는 환경을 정의하며 공간 데이터 변경을 위한 트랜잭션을 정의한다.

### 3.1 환경 정의

본 논문은 그림 1의 S-S-M(Server-Surrogate PC-Mobile Client) 환경을 연구 대상으로 한다. S-S-M은 단일 GIS 서버와 여러 대리

PC 및 이동 클라이언트로 구성된 환경으로써, 대리 PC와 이동 클라이언트는 1:1의 관계이다. 이동 클라이언트의 지도 데이터 초기화는 대리 PC를 통해서만 가능하다. 즉, 대리 PC는 유선 네트워크를 통해 서버의 데이터를 다운로드하고 이동 클라이언트는 대리 PC와 싱크(sync)하여 대리 PC로 전송된 데이터를 다운로드 받는다. 초기화 후, 이동 클라이언트는 대리 PC와 연결을 끊은 상태에서 공간 데이터를 삽입, 삭제, 수정하며 변경된 결과는 대리 PC를 통해 일괄처리 방식으로 서버에 전달한다.

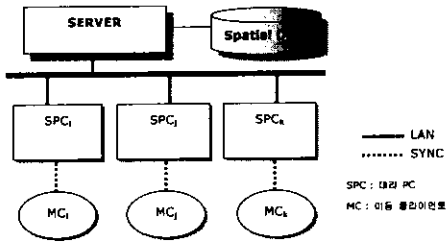


그림 1. S-S-M 환경

### 3.2 트랜잭션 정의

공간 데이터 변경을 위해 본 논문에서는 이동 클라이언트의 서버와 대리 PC 사이의 트랜잭션과 대리 PC와 이동 클라이언트 사이의 트랜잭션 및 이동 클라이언트의 가상 원료를 다음과 같이 정의한다.

#### 대리 트랜잭션(Surrogate Transaction, ST)

이동 클라이언트와 서버의 통신을 위해 대리 PC에서 수행하는 트랜잭션으로 다음과 같은 작업을 포함한다.

- 서버의 공간 데이터를 이동 클라이언트에 전송
- 이동 클라이언트가 변경한 공간 데이터를 서버에 전송
- 서버에서 검증한 공간 데이터 변경 완료 및 복귀를 이동 클라이언트에 전달

#### 이동 트랜잭션(Mobile Transaction, MT)

접속을 끊은 이동 클라이언트의 공간 데이터 변경 트랜잭션이며 다음의 작업으로 구성된다.

- 대리 PC와 싱크하여 공간 데이터를 이동 클라이언트로 다운로드
- 이동 클라이언트의 공간 데이터 변경(삽입, 삭제, 수정)
- 공간 데이터 변경 작업을 대리 PC로 업로드

#### 가상 원료(Virtual Commit, VC)

하나의 트랜잭션이 완료되어야 새로운 트랜잭션을 시작할 수 있다. 따라서, 이동 클라이언트에서의 공간 데이터 변경 트랜잭션을 가상으로 완료하여 새로운 트랜잭션을 시작할 수 있게 한다.

그림 2는 단일 클라이언트의 공간 데이터의 변경 과정을 나타낸다. 대리 PC가 서버에 지도 데이터를 요청하면서 대리 트랜잭션이 시작되고 서버는 대리 트랜잭션의 시작을 로그에 저장한다.(그림 2-①) 서버의 로그에는 대리 트랜잭션의 시작과 검증단계 및 완료 단계에 대한 정보가 저장된다. 이동 트랜잭션은 대리 PC에 저장된 서버의 데이터를 이동 클라이언트가 요구하는 순간부터 시작된다. 이동 트랜잭션이 시작되면 이동 클라이언트는 대리 PC와 접속을 끊고 장시간 외부에서 공간 데이터의 변경 작업을 수행한다. 이동 트랜잭션은 공간 데이터의 삽입, 삭제, 수정과 같은 연산들로 구성되고 변경 작업이 끝난 이동 트랜잭션은 가상 완료되어(그림 2-②) 새로운 트랜잭션이 시작될 수 있게 한다. 가상 완료된 이동 트랜잭션은 대리 PC로 옮겨지고

대리 PC가 이동 클라이언트의 요청을 대신하여 이동 트랜잭션을 대리 트랜잭션 단위로 서버에 전송한다.

서버에 전송된 대리 트랜잭션은 이동 트랜잭션의 수정 연산 단위로 검증단계를 밟으며 대리 트랜잭션의 검증단계를 로그에 저장한다.(그림 2-③) 검증단계에서 트랜잭션간 충돌이 발생하는 경우 대리 트랜잭션과 이동 트랜잭션의 변경 작업은 복귀된다. 서버는 대리 트랜잭션의 검증이 완전히 끝나면 대리 PC에게 대리 트랜잭션의 완료 및 복귀 여부를 전달하고 대리 트랜잭션의 완료를 로그에 저장한다.(그림 2-④) 복귀된 클라이언트는 변경 트랜잭션을 재시작 하거나 트랜잭션의 재시작을 그만둘 수 있다.

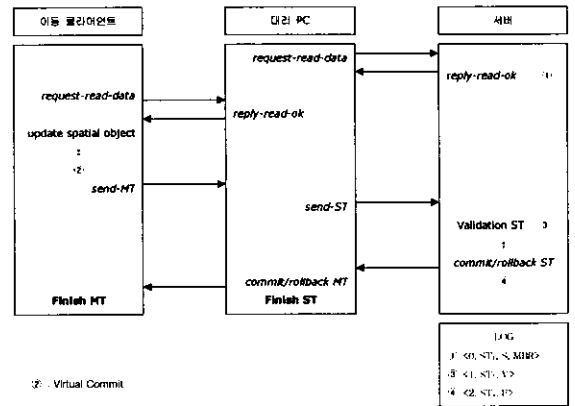


그림 2. S-S-M의 단일 트랜잭션 시나리오

### 4. 확인 기법을 이용한 직렬성 유지

이 장에서는 앞에서 정의한 트랜잭션의 직렬화를 보장하기 위한 방법인 확장된 확인 기법을 설명하고 이를 이용해 직렬성을 유지하는 예를 보인다.

#### 4.1 확장된 확인 기법

서버는 전통적인 확인 기법으로 여러 변경 트랜잭션의 동시성을 유지한다. 트랜잭션은 판독 단계, 검증단계, 기록 단계의 3단계를 거쳐 처리되며 검증단계에서는 트랜잭션간 직렬 가능성 위배 여부와 변경한 데이터의 충돌 여부 검증이 필요하다. 전통적인 확인 기법에서의 트랜잭션간 간섭 배제되는 조건은 표 1과 같다(6).

표 1. 전통적인 확인 기법에서의 트랜잭션간 간섭 배제 조건

가정 :  $T_i < T_j$  (트랜잭션  $T_i$ 가  $T_j$ 보다 먼저 시작함)

조건 1  
Finish( $T_i$ ) < Start( $T_j$ )

조건 2

Start( $T_i$ ) < Finish( $T_j$ ) < Validation( $T_j$ )  
Write-Set( $T_i$ ) ∩ Write-Set( $T_j$ ) = ∅  
Write-Set( $T_i$ ) ∩ Write-Set( $T_j$ ) = ∅

(Write-Set : 변경하는 데이터들, Read-Set : 읽은 데이터들)

공간 데이터에는 일반 데이터와 달리 공간 데이터간의 disjoints, meets, insides, covers, equals, overlaps의 관계인 공간 관련성이 존재한다. 따라서 기존의 확인 기법을 확장한 새로운 충돌 검증이 필요하다. 공간 데이터 변경에서 충돌 발생 조건은 표 1에서의 트랜잭션 간섭 배제 조건을 포함하며 조건 3을 만족해야 한다. 즉, 트랜잭션간 수정하는 영역이 겹치면 충돌이 발생할 수 있다.

조건 3

$$MBR(T_i) \cap MBR(T_j) \neq \emptyset$$

$MBR(T)$  = 트랜잭션 T가 수정하는 영역

충돌의 종류에는 그림 3과 같이 동일 객체 변경 및 변경한 객체간 공간 관련 위배가 있다. 그림 3-a에서 클라이언트<sub>1</sub>과 클라이언트<sub>2</sub>가 같은 객체 SO<sub>1</sub>을 수정하였고 그림 3-b에서는 두 이동 클라이언트가 같은 객체를 수정하지 않았지만 수정한 결과로 SO<sub>1</sub>과 SO<sub>2</sub>가 의도하지 않은 overlap 되어버려 공간 관련성이 위배되는 충돌을 보여준다.

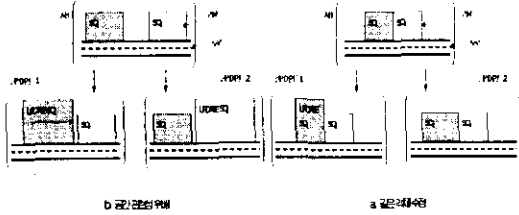


그림 3. 충돌 종류

그림 4는 두 대리 트랜잭션간 충돌 감지 단계의 예이다. 일단 대리 트랜잭션의 영역이 겹치면 충돌이 발생할 가능성이 있다. 충돌 감지는 두 대리 트랜잭션의 영역 비교한 후 대리 트랜잭션의 영역이 겹치면 이동 트랜잭션의 영역을 비교로 이루어진다. 이동 트랜잭션의 영역이 겹치면 이동 트랜잭션 내의 공간 데이터 변경 연산이 실제로 충돌하는지 정제 단계를 거친다. 이동 트랜잭션의 연산에서 충돌이 발생하면 해당 트랜잭션은 복귀한다.

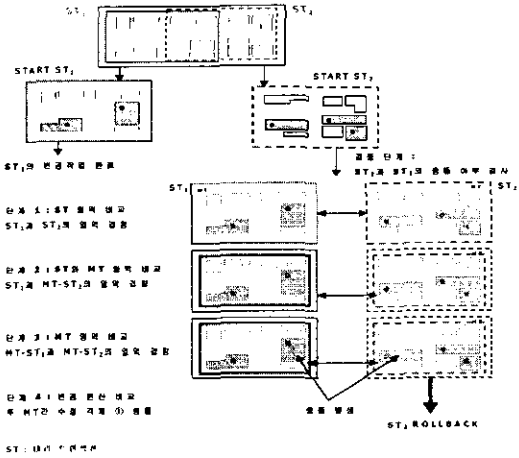


그림 4. 충돌 감지의 단계의 예

4.2 직렬화 보장의 예

동시에 중복된 영역을 수정하는 클라이언트의 대리 트랜잭션 시작 및 완료 시점에 따라 트랜잭션간 인터리빙 관계를 먼저 시작한 트랜잭션이 먼저 완료하는 경우(경우-1)와 나중에 시작한 트랜잭션이 먼저 완료하는 경우(경우-2)로 나누었다[4]. 경우-1에서 먼저 시작한 트랜잭션이 먼저 완료되므로 나중에 시작한 트랜잭션은 지연 없이 검증 단계를 거쳐 완료 또는 취소된다. 반대로 경우-2와 같이 먼저 시작한 트랜잭션보다 나중에 시작한 트랜잭션이 일찍 검증단계에 들어가면 직렬성 보장을 위해 먼저 시작한 트랜잭션이 검증단계에 들어올 때까지 트랜잭션의 완료를 지연시켰다[4]. 그러나, S-S-M 환경에서는 이러한 지연의 완료가 무한정 길어질 수 있으므로 지연 완료를 해결하기 위하여 검증단계에 들어가는 트랜잭션은 즉시 완료하되, 이미 완료

된 트랜잭션이 다른 트랜잭션과 충돌이 발생하면 보상 트랜잭션을 이용하여 완료된 트랜잭션을 복귀시킴으로써 트랜잭션간 직렬성을 보장한다[7].

그림 5에서 경우-1에서와 같이 먼저 시작한 대리 트랜잭션 먼저 완료하는 경우를 예를 들어 설명한다. ST<sub>1</sub>는 먼저 시작한 ST<sub>1</sub>의 완료를 기다릴 필요 없이 바로 검증단계에 들어가며 ST<sub>1</sub>과 충돌이 발생하면 늦게 시작한 ST<sub>2</sub>는 취소된다.

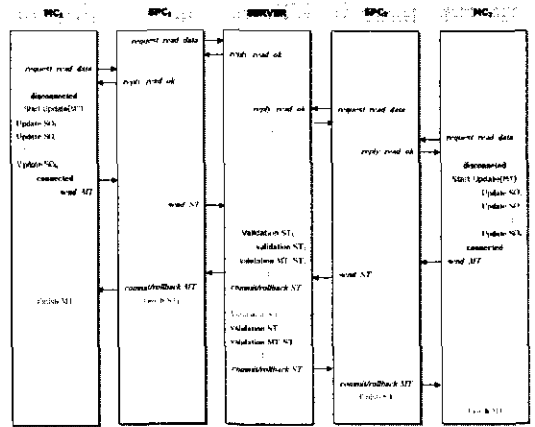


그림 5. 두 클라이언트 트랜잭션의 직렬성 보장의 예

5. 결론 및 향후 연구

본 논문에서는 이동 컴퓨팅 환경에서 공간 데이터 변경을 위한 3-계층 구조의 S-S-M 환경을 연구 대상으로 하였다. 이 S-S-M 환경에서 공간 데이터 변경을 위한 이동 트랜잭션 및 대리 트랜잭션을 정의하고 이 환경에서의 트랜잭션 모델을 제시하였다. 트랜잭션의 동시성을 제어하기 위하여 전통적인 낙관 기법의 확인 기법을 확장하고 이 트랜잭션 모델에서 클라이언트 트랜잭션간 인터리빙 관계에 대한 직렬성 보장을 예를 들어 설명하였다.

참고 문헌

- (1). G. H. Forman, J. Zahorjan, "The Challenges of Mobile Computing", IEEE Computer, Apr. 1994
- (2). Keith K. S. Lee, Y. H. Chin, "A new replication strategy for unforeseeable disconnection under agent-based mobile computing system", Parallel and Distributed Systems, 1998. Proceedings, 1998 International Conference on , 1998.
- (3). Eliezer Levy, Henry F. Korth and Abraham Silberschatz, "An optimistic commit protocol for distributed transaction management", Proceedings of the 1991 ACM SIGMOD international conference on Management of data, 1991, Pages 88 - 97
- (4). H. T. Kung, "On optimistic methods for concurrency control", ACM Trans. Database Syst. 6, 2 (Jun. 1981), Pages 213 - 226.
- (5). J. Eliot, B. Moss, "Log-Based Recovery for Nested Transactions", Proceeding of the 13<sup>th</sup> VLDB Conference, Brighton 1987, pp. 427-423, 1987.
- (6). Abraham Silberschatz, Henry F. Korth, S. Sudarshan, "Database System Concepts", third edition
- (7). Henry F. Korth, Eliezer Levy, Abraham Silberschatz, "A Formal Approach to Recovery by Compensating Transactions", Proceedings of the 16<sup>th</sup> VLDB Conference, PP. 95-106, 1990