
모바일 환경에서 다중 큐 기반의 동기화 시스템 구현 및 성능 비교

김홍기* · 김동현**

Implementation And Performance Evaluation of a Synchronization System based on the
multi-queue on Mobile Environments

Hong-Ki Kim* · Dong Hyun Kim**

본 연구는 국토해양부 첨단도시개발사업 - 지능형국토정보기술혁신 사업과제의
연구비지원(07국토정보C05)에 의해 수행되었음

요 약

GIS 서비스는 최적의 서비스를 위해 최신의 시공간 데이터를 제공해야 한다. 시공간 데이터의 양방향 동기화 기법은 최신의 데이터를 제공하기 위해 모바일 단말기를 이용하여 현장에서 변경된 시공간 데이터를 수집하고, 무선 네트워크를 이용하여 서버로 수집된 데이터를 갱신한다. 그러나 여러 클라이언트의 동기화 작업을 순차적으로 수행하기 때문에 동기화 요청 순서가 늦은 클라이언트들이 오랜 대기시간을 가진다. 이 논문에서는 다중 큐를 이용하여 여러 클라이언트의 동기화 요청이나 대용량의 동기화 작업으로 발생하는 대기 시간을 감소시킬 수 있는 동시 동기화 기법을 제안한다. 또한, 제안한 기법과 순차적 동기화 기법을 각각 구현하여 성능을 비교한 결과 제안한 동시 동기화 기법이 순차적인 기법보다 31%정도 대기시간이 감소함을 확인하였다.

ABSTRACT

The GIS service has to provide latest spatio-temporal data for the effective service. The two-way synchronization scheme collects updated spatio-temporal data in the field and synchronizes with a server by using the wireless network. However, since the two-way synchronization scheme processes sequentially synchronizing work of clients, it has the problem that the synchronization of multiple clients takes long time. In this paper, we propose the concurrent synchronizations scheme based on the multi-queue in order to reduce the waiting time of multiple clients. We also build both schemes and evaluate the performance test. The results show that the proposed scheme reduce the waiting time by 31%.

키워드

동기화, 모바일 클라이언트, 모바일 GIS, 시공간데이터, 무선 네트워크

Key word

synchronization, mobile clients, mobile GIS, spatio-temporal data, wireless network

* 정회원 : 동서대학교 유비쿼터스IT 대학원

접수일자 : 2010. 08. 20

** 정회원 : 동서대학교 컴퓨터정보공학부 교수 (교신저자, pusrover@dongseo.ac.kr) 심사완료일자 : 2010. 09. 07

I. 서 론

모바일 GIS 시스템은 최적의 서비스를 위해 공간 정보의 주기적인 동기화가 필요하다. 기존의 기법은 공간 정보를 CD나 유선 네트워크를 이용하여 동기화하였다. 그러나 CD나 유선 네트워크를 이용한 동기화는 최신 공간 정보의 실시간 전파가 불가능하고 동기화 연산을 위해 사용자가 직접 작업을 수행하는 불편함이 있다. 최적의 서비스를 제공하기 위해서는 사용자의 불편을 최소화하고, 실시간으로 최신의 공간 정보를 동기화할 수 있는 기술이 필요하다.

ActMAP시스템[1]과 양방향 동기화 기법[2]은 최신 공간 정보를 무선 네트워크를 이용하여 신속하게 전파할 수 있다. 그러나 ActMAP시스템과 양방향 동기화 기법은 다수의 클라이언트에 대해서 갱신된 공간 정보의 동기화 작업을 순차적으로 처리한다. 순차적으로 동기화를 수행할 경우 처리시간이 긴 클라이언트 이후에 동기화를 수행하는 다른 클라이언트들이 오랜 시간 대기하는 문제가 발생한다.

이 논문에서는 대용량 데이터의 집중된 동기화 요청으로 인해 발생하는 대기 시간을 줄이기 위해 다중 큐를 이용한 양방향 동기화 기법을 제안한다. 제안하는 동기화 기법은 다중 큐를 이용하여 변경충돌이 없는 클라이언트들의 동기화 작업을 동시에 수행한다. 또한 변경충돌은 없으나 동기화 되는 영역이 같은 클라이언트들은 순차적으로 처리하여 변경된 데이터의 일관성을 보장한다. 또한 제안한 기법을 이용한 다양한 환경에서 실험을 통한 성능평가를 수행한다.

이 논문의 구성은 다음과 같다. 2장에서는 관련연구를 기술하고 3장에서는 순차적 동기화의 문제점을 기술한다. 4장에서는 다중 큐를 이용한 동기화 기법을 제시한다. 5장에서는 제안하는 동기화 기법과 순차적 동기화 기법의 성능을 비교한다. 마지막으로 6장에서는 결론 및 향후 해결해야 될 과제에 대해서 설명한다.

II. 관련연구

ActMAP[1]은 차량용 네비게이션 시스템에 최신의 공간 정보를 신속하게 전파하기 위해 개발되었다. ActMAP은 기존의 CD를 통한 전파방식인 오프라인

방식과 무선 네트워크를 이용하는 온라인 방식의 변경전파를 모두 지원한다. 오프라인 방식은 CD를 이용하여 네비게이션에 사용되는 대용량의 지도 데이터를 배포한다. 온라인 방식은 CD를 통해 배포된 지도 데이터에 변경된 정보들을 무선 네트워크를 이용하여 전파한다. 그러나 ActMAP 시스템은 서버에서 모바일 단말기로 단방향 변경전파만 가능하기 때문에 서버의 갱신 주기에 따라 공간 정보의 최신성이 결정되는 문제가 있다.

양방향 동기화 기법[2]은 모바일 단말기와 서버간 양방향으로 공간데이터를 변경하기 위한 기법이다. 먼저 모바일 단말기를 이용하여 현장에서 변경된 공간 정보를 수집한다. 수집된 공간 정보는 무선 네트워크를 이용하여 서버와 동기화한다. 서버에 갱신된 공간 정보는 무선 네트워크로 연결된 다른 클라이언트들로 전파한다. 그러나 양방향 동기화 기법은 데이터 일관성 유지를 위해 다수 클라이언트의 동기화 작업을 순차적으로 처리하기 때문에 긴 작업시간을 요구하는 동기화 요청이 있는 경우 동기화 요청 작업의 평균 대기시간이 증가하는 문제가 있다.

III. 문제정의

모바일 환경은 다수의 클라이언트들이 동시에 접속하여 동기화를 요청할 수 있다. 동시에 접속된 클라이언트들에 대해서 양방향 동기화 기법은 동기화 요청 순서대로 그림 1과 같이 순차적으로 처리한다. 그러나 순차적 처리방법은 클라이언트가 긴 대기 시간을 가지는 문제가 발생할 수 있다.

그림 1의 a)는 MC₁, MC₂, MC₃, MC₄가 동기화를 요청하였을 때 순차적인 동기화 처리로 인해 MC₂, MC₃, MC₄로 갈수록 동기화를 위해 대기해야하는 시간이 점점 더 증가하는 경우를 보여준다. 그림 1의 b)는 처리시간이 긴 동기화작업으로 인해 MC₂가 오랜 대기시간을 가지게 되는 경우를 보여준다. 이 논문에서는 이러한 대기 시간을 해결하기 위해 변경충돌이 없는 클라이언트의 동기화를 동시에 처리하여 불필요한 대기 시간을 감소시키하고자 한다.

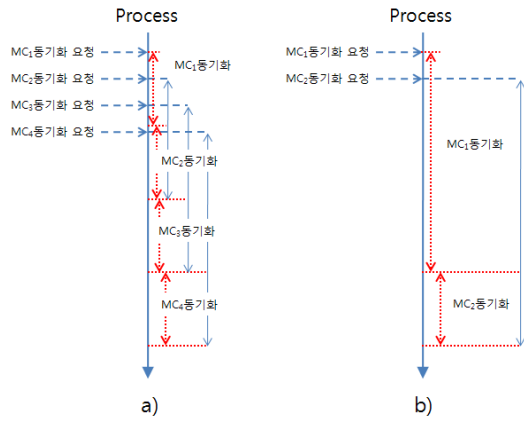


그림 1 문제정의
Fig. 1 Problem definition

IV. 다중 큐를 이용한 동기화 프로토콜

기본적인 아이디어는 다중 큐를 사용하여 변경충돌이 없는 클라이언트의 동기화 작업을 분산하고, 이를 동시에 처리하는 것이다.

4.1 용어정의

이 논문에서 사용되는 용어는 다음과 같다.

- LST(Last Sync Time) : 클라이언트가 서버와 최종 동기화를 한 시간으로 계층, 분할 별로 관리
- LUT(Last Update Time) : 서버의 데이터가 최종적으로 갱신된 시간으로 계층, 분할 별로 관리
- DO(Delta Object) : LST이후 서버에서 변경된 데이터
- DOS(Delta Object Set) : DO들의 집합
- TDO(Temporary Delta Object) : 최종 동기화 시간 이후 클라이언트에서 변경된 데이터
- TDOS(Temporary Delta Object Set) : TDO들의 집합
- CR(Copy Region) : 클라이언트가 복사해서 가지고 있는 데이터영역 중 동기화를 요청하는 계층, 분할의 집합

4.2 양방향 동기화 프로토콜

그림 2는 M_1 , M_2 순서로 동기화를 요청하였을 때 수행되는 동기화 프로토콜을 보여준다.

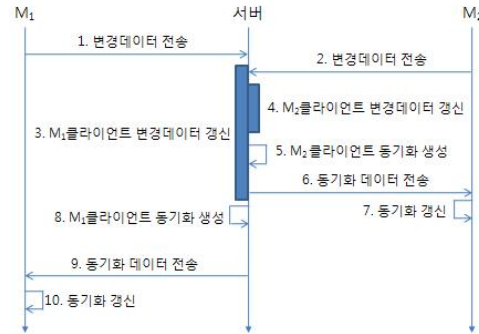


그림 2 동시 동기화 프로토콜
Fig. 2 Concurrent synchronization protocol

M_1 과 M_2 는 각각의 TDOS를 서버로 동기화 하기 위해 전송한다. M_1 의 TDOS는 서버와 변경충돌 검사 후 다중 큐에 삽입한다. 만약 변경충돌이 발생하면 M_1 의 동기화는 취소된다. M_2 의 TDOS는 서버와 변경충돌 검사를 수행하면서 M_1 의 TDOS와 변경충돌이 발생하는지도 검사한다. 변경충돌이 없다면 다중 큐에 삽입하고, 변경충돌이 발생하면 동기화를 취소한다. 다중 큐에서 TDOS 갱신이 완료되면 클라이언트로 전송할 DOS를 생성한다. 그림 2는 M_2 의 TDOS가 M_1 의 TDOS보다 처리 시간이 짧은 경우로 M_2 의 DOS가 먼저 생성되어 전송된다. DOS를 전송받은 M_2 는 DOS를 이용하여 데이터베이스를 갱신하고 동기화를 완료한다. 이후 TDOS 처리가 완료된 M_1 은 DOS를 전송받아 데이터베이스를 갱신하고 동기화를 완료한다.

V. 성능평가

5.1 시스템 구조

그림 3은 다중 큐를 이용한 동시 동기화 시스템의 구성도를 보여준다. 모바일 클라이언트가 동기화를 요청하면, 통신 관리자로 동기화 요청메시지가 수신된다. 통신 관리자는 수신한 메시지를 동기화 관리자로 전달한다. 동기화 관리자는 전달받은 메시지를 분석하고, 클라이언트로부터 전송받은 TDOS가 있으면 변경충돌 관리자로 전달한다. 클라이언트로부터 전송된 TDOS가 없는 경우는 동기화 어댑터로 전송해야하는 DOS를 요청한다.

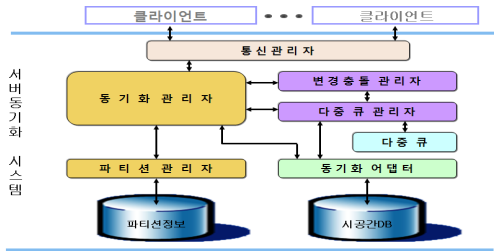


그림 3 동시 동기화 시스템 구성도
Fig. 3 Concurrent synchronization system configuration

변경충돌 관리자는 TDOS의 변경충돌 여부를 검사하고 충돌이 없으면 다중 큐 관리자로 TDOS를 전달한다. 만약 변경충돌이 발생하면 동기화 관리자는 변경충돌 발생을 알리고 동기화 관리자는 변경충돌이 발생한 클라이언트의 동기화를 취소한다. 다중 큐 관리자는 TDOS를 큐에 삽입하여 처리한다.

5.2 성능비교 요소

다중 큐를 이용한 동기화 시스템과 순차적 동기화 시스템의 성능 비교 요소는 표 1과 같다.

표 1 성능비교 요소
Table. 1 Performance comparison factors

파라미터	동기화 데이터 수	클라이언트 입력 패턴
설정값	1000	LS 패턴 (1:2, 1:4, 1:6)
	5000	SL 패턴 (2:1, 4:1, 6:1)
	10000	랜덤

클라이언트들의 동기화 데이터 수에 따라 순차적 동기화와 다중 큐 동기화의 대기시간의 평균을 측정한다. 동기화를 요청하는 클라이언트들은 처리시간이 짧은 경우와 처리시간이 긴 경우로 구분된다. 다중 큐를 이용한 동기화 기법은 이러한 클라이언트들의 동기화 순서와 비율에 따라 성능이 달라질 수 있다. 성능 비교를 위해서 동기화 시간이 긴 클라이언트는 L형으로, 동기화 시간이 짧은 클라이언트는 S형으로 정의하고 동기화 요청 순서에 따라 LS패턴과 SL패턴으로 구분한다. 그리고 실제 동기화 상황에서 L형이 S형보다 발생빈도가 적기 때문에 각각의 패턴에서 L형과 S형의 발생 비율을 1:2, 1:4, 1:6 형태로 S형의 발생 빈도를 변경하여 성능을 비교한다.

5.3 실험 결과

이 절에서는 표1에서 정의한 성능비교 요소를 이용하여 다중 큐를 이용한 동기화 시스템과 순차적 동기화 시스템의 성능을 비교하였다. 모든 클라이언트의 변경 데이터는 충돌이 없고, 중복영역이 없다는 가정에서 실험을 수행 하였다.

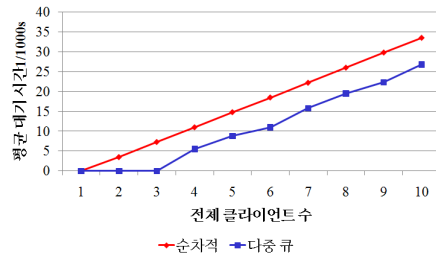


그림 4 변경 데이터 수가 1000인 경우
Fig. 4 Number of update date is 1000

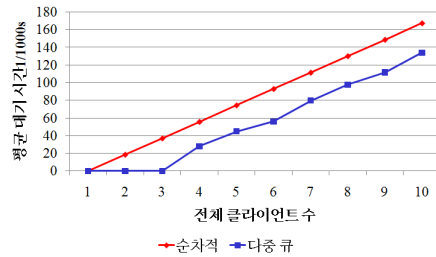


그림 5 변경 데이터 수가 5000인 경우
Fig. 5 Number of update date is 5000

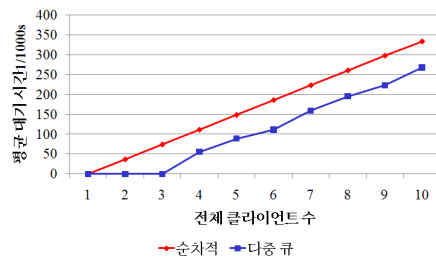


그림 6 변경 데이터 수가 10000인 경우
Fig. 6 Number of update date is 10000

그림 4, 5, 6은 각각 동기화를 요청한 클라이언트의 동기화 데이터의 수가 1000, 5000, 10000일 때 순차적 동기화 기법과 다중 큐를 이용한 동기화 기법의 평균대기시간을 측정한 결과이다. 이 실험을 통해 데이터 수가 동일한 경우 순차적 동기화 보다 다중 큐를 이용한 동기화가

대기시간이 짧음을 확인할 수 있다.

면서 발생하는 문제이다.

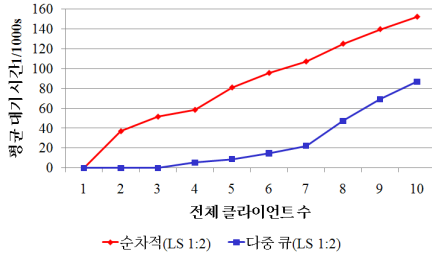


그림 7 LS(1:2)패턴의 평균대기 시간
FIG. 7 Waiting time of the LS(1:2) pattern

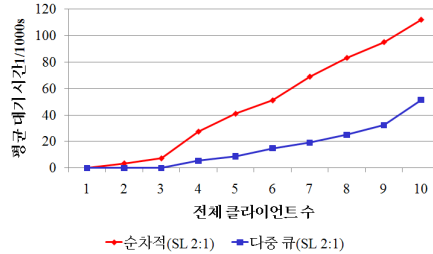


그림 10 SL(1:2)패턴의 평균대기 시간
FIG. 10 Waiting time of the SL(1:2) pattern

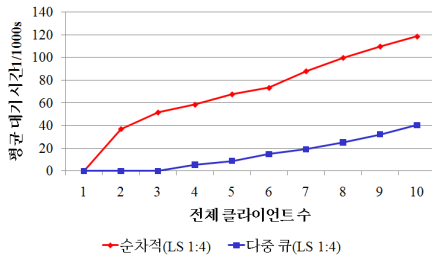


그림 8 LS(1:4)패턴의 평균대기 시간
FIG. 8 Waiting time of the LS(1:4) pattern

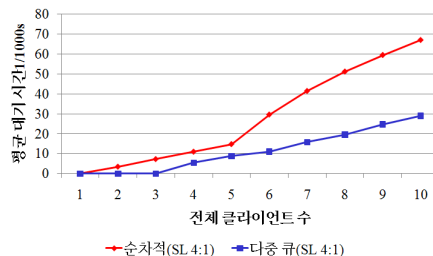


그림 11 SL(1:4)패턴의 평균대기 시간
FIG. 11 Waiting time of the SL(1:4) pattern

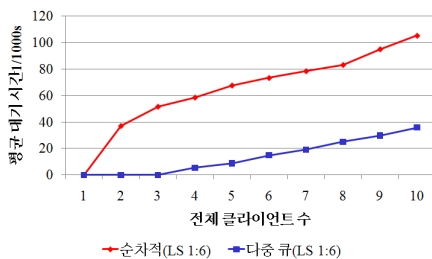


그림 9 LS(1:6)패턴의 평균대기 시간
FIG. 9 Waiting time of the LS(1:6) pattern

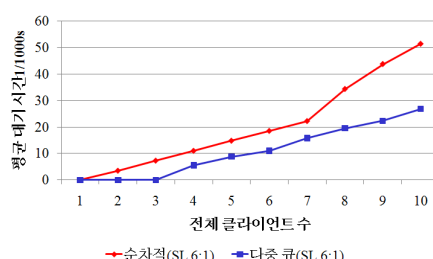


그림 12 SL(1:6)패턴의 평균대기 시간
FIG. 12 Waiting time of the SL(1:6) pattern

그림 7, 8, 9는 동기화를 요청한 클라이언트들이 LS패턴으로 동기화를 요청한 경우의 실험으로 L형 클라이언트와 S형 클라이언트의 발생 비율을 1:2, 1:4, 1:6하여 실험한 결과이다. 이 실험의 결과 순차적 동기화 기법은 LS패턴에서 대기시간 문제가 크게 나타남을 확인할 수 있었다. 다중 큐를 이용한 경우도 1:2 비율을 가지는 결과를 확인하면 클라이언트의 수가 7이 되는 구간부터 대기시간이 급증하게 되는데 이는 다중 큐가 3개의 큐로 구성된 상태에서 모두 큐에 L형의 클라이언트가 존재하

그림 10, 11, 12는 SL패턴으로 동기화를 요청한 경우의 실험으로 S형 클라이언트와 L형 클라이언트의 발생 비율을 1:2, 1:4, 1:6으로 하여 실험한 결과이다. SL패턴은 L형의 클라이언트가 요청되기 이전까지는 순차적 동기화와 다중 큐를 이용한 동기화간에 대기시간차이가 크게 발생하지 않음을 확인할 수 있다.

그림 13은 L형과 S형을 랜덤으로 발생시킬 경우의 실험 결과이다. 이 실험의 결과는 1000번 이상 수행한 실험 결과의 평균값으로 실제 서비스에서 이러한 그래프로

성능이 나타나지는 않는다. 그러나 이 결과로 다중 큐를 이용한 동기화 기법이 순차적 동기화 기법보다 대기시간이 짧을 것을 확인할 수 있다.

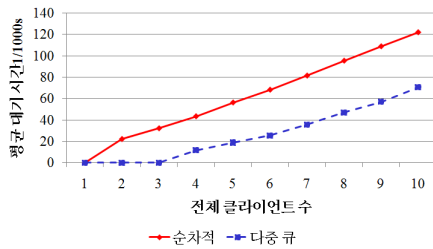


그림 13 랜덤패턴의 평균대기 시간
FIG. 13 Waiting time of the random pattern

VI. 결론

기존의 모바일 기반 시공간데이터 동기화 시스템은 다수의 클라이언트로부터 요청된 동기화 작업을 순차적으로 처리하여 요청순서가 늦은 클라이언트들이 오랜 대기시간을 가지는 문제가 있다. 이 논문에서는 이러한 대기시간을 감소시키기 위해 다중 큐를 이용한 동기화 시스템을 제안하였다. 다중 큐를 이용한 동기화 기법은 변경충돌이 없는 클라이언트들의 동기화 작업을 동시에 처리하여 불필요한 대기 시간을 감소시킨다. 실제 시스템을 구현하여 순차적 동기화 기법과의 성능비교를 통해 순차적 동기화 기법보다 대기 시간이 감소함을 확인하였다. 향후 다중 큐의 동시처리의 효율성과 오류가 발생한 경우의 회복에 대한 연구가 필요하다.

참고문헌

- [1] "ActMAP White Paper and Interfaces to the FeedMAP framework," white paper, 2007
- [2] 김홍기, 임창우, 이상신, 조대수, 김동현, "모바일 GIS DB를 위한 양방향 동기화 프로토콜 설계," 한국해양정보통신학회 2008 춘계종합학술대회, Vol.12, No.1, pp.183-186, 2008
- [3] Mi-Seon Choi, Young-Kuk Kim, Juno Chang, "Transaction-Centric Split Synchronization Mechanism

- for Mobile E-Business Applications," IEEE, 2005
- [4] 이혜진, 김진석, "모바일 환경에서 공간데이터 동기화 시스템 설계," 한국정보과학회, 2004년 가을 학술발표논문집 제31권 제 2호, 2004. 10, pp. 184~186.
- [5] 최진오, 홍봉희, "중복된 공간 데이터 관리를 위한 공동작업 트랜잭션 모델," 한국정보과학회, 한국정보과학회 학술발표논문집, 한국정보과학회 1997년 봄 학술발표논문집 제 24권 제1호, 1997. 4, pp. 88~92.
- [6] Shuai hao, Hock Beng Lim, "Data Synchronization in Distributed and Constrained Mobile Sensor Networks," UIC2007, LNCS 4611, pp.673~683, 2007
- [7] Mi-seon Choi, Young-Kuk Kim, Juno Chang, "Transaction-Centric Split Synchronization Mechanism for Mobile E-Business Applications," IEEE,2005

저자소개

김홍기(Hong-Ki Kim)

한국해양정보통신학회 논문지
제 14권 제4호 참조

김동현(Dong Hyuni Kim)

한국해양정보통신학회 논문지
제 14권 제4호 참조