

전자 상거래 시스템에서의 지연된 정보 동기화 방법*

이동주, 연종흠, 이상구

서울대학교 전기·컴퓨터공학부

Delayed Information Synchronization in e-Commerce Systems

Dongjoo Lee, Jongheum Yeon, Sang-goo Lee

요 약

전자 상거래에서 이용되는 전자카탈로그 정보는 다양한 이유에 의해서 중복 저장되기도 하는데, 이는 멀티데이터베이스 시스템에서의 글로벌 트랜잭션 보장 문제를 야기한다. 본 논문에서는 하나의 시스템에서만 카탈로그의 변경이 발생하고 변경 정보가 전파되는 특별한 상황에서의 동기화 문제 해결을 위한 로그 작성 및 관리 방법을 제안한다. 이를 만족하여 구현된 동기화 모듈은 지연된 동기화를 보장함을 보이고, 실제 환경에서 구현된 예를 통해서 제안한 방법이 유용함을 보인다.

Abstract

E-Catalogs are often duplicated in multiple database systems due to various reasons. This causes the problem of assuring ACID properties of global transactions in multi-database management systems. We suggest a log-based synchronization mechanism for a special situation that catalogs can be changed

* 본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT연구센터 육성·지원사업 (IITA-2008-C1090-0801-0031)의 연구결과로 수행되었음.

only in one system and the changes spread to other systems. We have defined several conditions that have to be satisfied to assure the synchronization of the information. We show that the information can be kept identical in a delayed manner if synchronization module fulfills the defined conditions. Finally we show that the suggested mechanism is useful by showing a real world application that uses this synchronization mechanism in e-commerce environment.

1. 서론

분산 시스템 환경에서는 동일한 정보가 여러 용도로 사용되기도 하고, 물리적, 기능적으로 독립된 여러 개의 데이터베이스에 중복되어 저장되기도 한다. 특히, 전자 카탈로그와 같이 다양한 용도로 여러 시스템에서 사용되는 정보는 시스템의 독립성 보장이나 성능 보장과 같은 다양한 이유로 중복되어 저장되는 경우가 빈번하다. 이같이 기능적, 형태적으로 서로 다른 멀티데이터베이스 환경에서 트랜잭션의 ACID 속성 보장과 각 시스템의 독립성 유지는 서로 상충되는 요구로 오래 전부터 해결하기 어려운 문제로 다루어져 왔다[1,2,3,4].

중복하여 저장된 정보는 지속적으로 동기화 되어 정보의 동일성이 유지되어야 할 필요가 있는데, 이를 위해 글로벌 트랜잭션을 보장하고자 할 때에는 시스템의 독립성을 해치게 된다. 따라서 ACID 속성 보장과 독립성 유지와의 적절한 절충 지점을 선택해야 한다.

본 논문에서는 정보의 변경이 하나의 시스템에서만 발생하고, 변경 정보가 이를 사용하는 다른 시스템으로 전파되는 특별한 상황에서, 정보 변경에 대한 로그 작성 및 전송을 이용해 정보의 동기화 문제를 해결한다. 특히 추상화된 수준에서 로그 작성 및 전송 규약을 정의하고, 주어진 조건을 만족하도록 로그 모듈이 구현되면 지연된 동기화가 이루어짐을 보인다. 실제 인터넷 상거래 시스템에 구현된 예를 통해서 제안하는 방법이 유용함을 보인다.

2. 관련연구

2.1 멀티데이터베이스[2]

멀티데이터베이스는 이형의 독립적인 데이터베이스상에서의 글로벌 트랜잭션을 관리하기 위한 데이터베이스 관리 시스템이다. 멀티데이터베이스에서의 ACID 속성은 각 시스템의 독립성 보장과 상충한다. 다음의 독립성은 멀티데이터베이스를 일반적인 분산 데이터베이스와 구별하게 한다.

A. 설계 독립성(Design autonomy)

글로벌 트랜잭션을 위해서 시스템의 설계가 변경되지 말아야 함을 의미한다.

B. 실행 독립성(Execution autonomy)

각각의 개별 시스템이 각 시스템에 대한 완전한 관리 능력을 가지고 있어야 함을 의미한다.

C. 통신 독립성(Communication autonomy)

각 개별 시스템이 다른 시스템에 대한 관리 정보를 알 필요가 없어야 함을 의미한다.

독립성의 수준은 글로벌 트랜잭션의 보장을 위해 적절한 수준에서 절충되는데, 이는 각 개별 시스템의 특성이나, 트랜잭션의 유형에 따라서 달라진다.

3. 로그 기반 지연된 동기화 방법

본 장에서는 전자 카탈로그의 변경 전파에 의한 지연된 동기화 방법에 대해서 기술한다.

3.1 전자 상거래 환경에서의 동기화

정보의 동기화는 다양한 이유에 의해서 필요한데, 조달청[5]에서의 정보 시스템은 좋은 예가 될 수 있다. 조달청은 물품 정보(카탈로그)를 관리하는 국가 기관이다. 다양한 기능을 수행하며 이를 지원하기 위한 여러 정보 시스템이 존재한다.

카탈로그 관리 시스템은 카탈로그의 저장,

변경, 삭제 등의 기본 관리를 수행한다. 쇼핑물 시스템은 국가 기관의 물품 구매를 지원하고, 지능형 검색 시스템은 온톨로지를 활용하여 카탈로그에 대한 의미 검색을 지원한다[6]. 이외에도 다양한 기능을 지원하는 여러 시스템들이 존재한다.

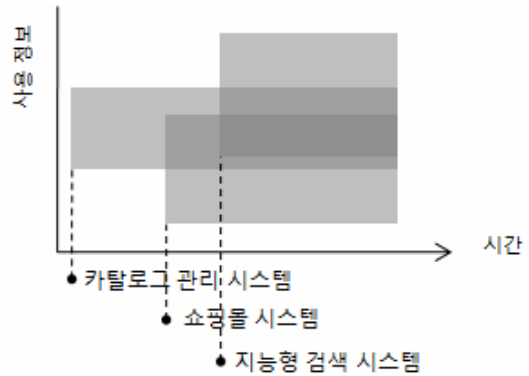


그림 1 조달청의 정보 시스템의 구축 시점 및 공유 정보

그림 1과 같이 이들은 서로 다른 시기에 구축되었으며, 일부 정보는 공유하지만, 많은 부분에 있어서 구조적, 기능적으로 다른 정보를 활용한다. 특히, 시스템의 구축과 관리의 주체가 다르므로 인해서 멀티데이터베이스 환경을 만드는 다음의 제약을 가진다.

제약 1. 강한 실행 독립성: 조달청 내의

정보 시스템은 각 시스템의 실행이 다른 시스템의 실행에 영향을 받아서는 안 된다.

중복되어 저장된 카탈로그 데이터에 대한 변경은 중복된 정보를 갱신하도록 하는 글로벌 트랜잭션을 유발하고 이는 각 시스템

의 독립성을 해친다. 따라서 글로벌 트랜잭션을 보장하기보다는 시스템의 독립성을 최대한 보장할 수 있는 동기화 방법이 필요하다.

3.2 로그 작성 및 전송

카탈로그 정보는 카탈로그 관리 시스템에서만 업데이트가 발생하고 다른 시스템에서는 읽기만 발생한다. 이는 동기화 시스템을 좀더 간단하게 한다.

다음과 같이 세 개의 시스템에서 데이터 x 에 대한 쓰기 및 읽기가 발생한다고 보자.

S_{n1}		S_m		S_{n2}	
		$T_{m,1}$	$w(x)$ $r(x)$		
$T_{n1,1}$	$w(x)$			$T_{n2,1}$	$r(x)$
$T_{n1,2}$	$r(x)$			$T_{n2,2}$	$w(x)$

S_m 이 변경의 주체가 되고, S_{n1} , S_{n2} 로 변경이 전파 된다면, 항상 S_{n1} 와 S_{n2} 에서의 $w(x)$ 는 S_m 에서보다 늦다. 그러나 위와 같이 $r(x)$ 를 포함한 트랜잭션이 $w(x)$ 를 포함한 트랜잭션보다 선행하는 경우에는 글로벌 직렬화가 깨지게 된다. 이를 위해서는 $w(x)$ 를 항상 $r(x)$ 앞에 오도록 응용 수준에서 $r(x)$ 에 대한 읽기 작업 전에 x 에 대한 변경을 확인하고, 변경이 발생했을 때 $w(x)$ 를 수행하도록 해야 한다. 그러나 이는 변경 확인을 위한 다른 시스템 정보 확인이라는 과도한 오버헤드를 유발한다. 또한, 만일 S_m 의 일시적인 장애로 변경 확인 및 처리가 불가능한 경우

에는 장애가 복구되기까지 대기하지 않는 한 여전히 글로벌 직렬화를 보장할 수 없다. 따라서 이 같은 상황에서는 강한 글로벌 트랜잭션의 보장보다는 지연된 동기화가 적합하다.

변경된 정보에 대해 로그를 작성하고 동기화 시스템이 다음과 같은 가정을 만족하면 지연된 동기화가 가능하다. 여기서 변경에 대한 주체가 되는 시스템을 S_m , 변경이 전파 되는 시스템을 S_{fi} 로 표현한다($i = 1, 2, \dots, n$).

가정 1. If there is $w(x)$ then $l(x)$ follows right after $w(x)$ in S_m .

이는 항상 변경에 대한 로그 $l(x)$ 가 $w(x)$ 와 함께 트랜잭션에 포함되어 ACID 속성을 만족함을 의미한다. 이같이 변경의 주체가 되는 시스템에서 로그가 작성되고, 로컬 직렬화가 된 시스템에서의 로그는 항상 변경과 동일한 순서를 가진다. 따라서 다음의 두 가정과 함께 다른 시스템에서의 순차성도 보장할 수 있다.

가정 2. If $l_1(x)$ precedes $l_2(x)$ in S_m then $l_1(x)$ precedes $l_2(x)$ in S_{fi} .

가정 3. If $l_1(x)$ precedes $l_2(x)$ then $w_1(x)$ precedes $w_2(x)$ in S_{fi} .

가정 2는 변경된 정보에 대한 기록이 순차적으로 전송됨을 의미하고, 가정 3은 전송된 로그가 순차적으로 반영됨을 의미한다.

그러나 위의 세가지 가정은 동일 로그가 두 번 전송되어 시스템에 두 번 반영되었을

때 동일한 정보로 유지된다는 것을 보장하지 못한다. 예를 들어 다음과 같은 작업이 두 번 발생하면 시스템간의 공유 정보의 내용이 달라진다.

```
w(x): x = x+1;
```

로그의 전송은 정상적으로 이루어 졌으나, 통신 환경이나 시스템의 이상에 의해서 로그 전송여부에 대한 확인이 정상적으로 이루어지지 않았을 때에는 로그의 전송이 2회 이상 발생할 수 있다. 따라서 로그 관리 시스템은 다음과 같은 가정을 만족해야 한다.

가정 4. Multiple $w(x)$ makes same result with one $w(x)$ in S_{fi} .

이는 로그 반영에 있어서의 멱등성을 의미하고 이를 위해서는 다음과 같이 내용 수준에서의 갱신이 되도록 로그의 형태를 작성해야 한다.

```
w(x): x = 11;
```

로그의 전송과 변경의 전파보다 변경 로그의 생성이 빠르다면, 정보의 동일성을 보장할 수 없다. 따라서 다음과 같은 가정이 추가되어야 한다.

가정 5. Creation of logs is much slower then delivery and processing of logs.

이같이 다섯 가지 가정을 만족하여 수행된 지연된 동기화는 그림 2에서와 같이 변경 정보가 모두 전파되지 않았을 때에는 글로벌 직렬화를 보장하지 못한다. 그러나 동일 정보를 공유하는 시점에는 항상 글로벌 직렬화를 보장한다. 변경의 전송과 반영 속

도가 매우 빨라 즉시적으로 로그가 처리되는 경우에는 글로벌 트랜잭션을 보장한 경우와 동일한 효과를 얻을 수 있다.

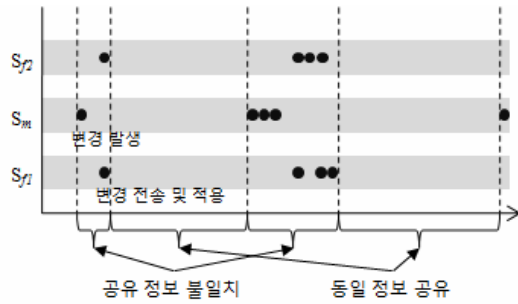


그림 2 지연된 동기화

3.3 로그 자료 구조

전파 대상이 되는 시스템으로의 변경 전송 여부와 변경 반영 여부를 함께 기록하여 동기화 시점을 확인하도록 다음과 같은 로그 자료 구조가 필요하다.

```
<id, time, eid, content, tsSfi, csSfi, tsSj2, csSj2>
```

*time*은 로그 전송 및 전파의 순차성 보장에 이용된다. ts_{Sfi} , cs_{Sfi} 는 S_{fi} 로의 로그 전송 및 반영 여부를 기록하여 동기화 시점을 확인할 수 있게 한다. 만일 작성된 모든 로그에 대해서 반영 여부가 참이면 이는 현재 모든 시스템이 동일한 정보를 공유한다는 것을 의미한다.

3.4 동기화 프로세스

동기화 프로세스는 그림 3에서와 같이 크게 A. 로그 작성 단계, B. 로그 전송 단계, C. 로그 반영 단계로 구분된다.

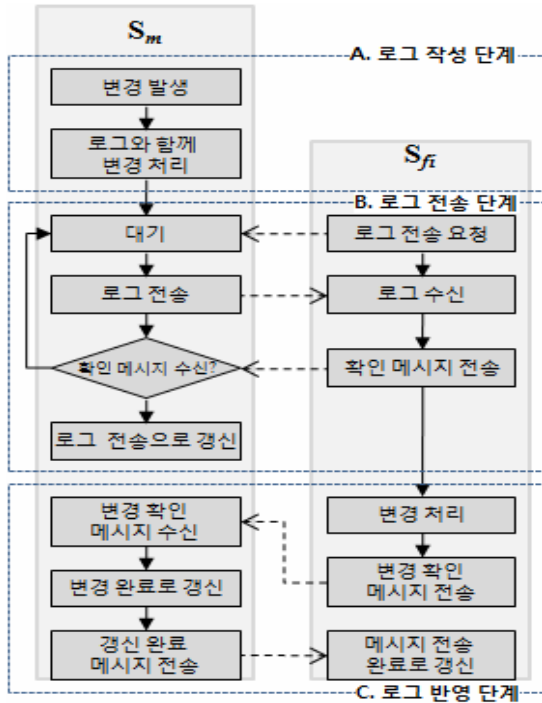


그림 3 동기화 프로세스

A. 로그 작성 단계

로그 작성 단계에는 변경된 정보에 대한 로그가 3.3에서 정의된 로그 자료구조로 저장된다. 이는 S_m 에서만 발생하는 것으로 로컬 트랜잭션을 보장한다.

B. 로그 전송 단계

변경 정보 전송은 S_{fi} 에서의 주기적 변경 정보 전송 요청에 의해 처리된다. 이 같은 전송은 S_m 이 전송 확인 메시지 수신으로 1회 이상의 정상적 로그 전달을 보장함으로써 모든 변경에 대한 로그 전송을 보장한다.

C. 로그 반영 단계

전송된 로그는 각각의 S_{fi} 에서 처리된다. 이때에 각 시스템에서 가지는 저장 구조로 가공 저장한다. 정보 반영이 완료되면 반영 완료 메시지를 전송하고, S_m 으로부터 반영 완료에 대한 수신 확인 메시지를 전송 받으면 해당 로그 상태를 메시지 전송 완료로 갱신한다. 수신확인 메시지를 전송 받지 못한 경우 이후 메시지를 재 전송하고, 수신 확인 메시지를 전송 받을 때까지 계속하여 전달한다. 이 같은 방법으로 S_m 에 정보 반영 상태가 갱신되는 것을 보장한다.

3.5 동기화 시스템 구현

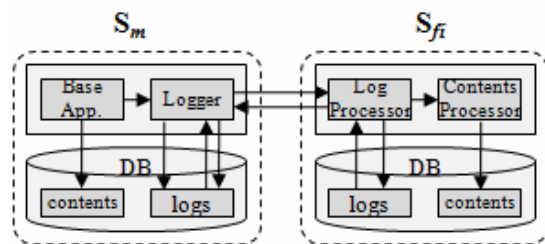


그림 4 동기화를 위한 로그처리 시스템

그림 4는 동기화를 위한 변경 주체 시스템과 변경 전파 시스템에서의 데이터 베이스와 응용을 나타낸다. 각 화살표는 정보의 이동을 나타낸다.

A. 컴포넌트

① Base Application

카탈로그 정보가 처리될 때 해당 정보에 대한 변경 이력을 Logger에 전달하고 동일

한 트랜잭션에 묶일 수 있도록 한다.

② Logger

변경된 정보를 데이터베이스에 저장하고, 다른 시스템으로 변경 정보를 전달한다. Logger는 변경 정보를 전달 받는 시스템과의 로그 전송이 확실히 되었는가와 변경에 대한 처리 여부를 전송 받아 로그에 기록한다.

③ Log Processor

전달 받은 로그를 데이터베이스에 저장하고, S_m 과의 확인 메시지 전송을 처리한다.

④ Contents Processor

전달 받은 변경 정보를 각 시스템에서 사용하는 방식대로 가공하여 저장한다.

B. 구현된 시스템 및 평가

조달청 동기화 시스템은 Java[7]를 기반으로 구현되었다. J2EE 아키텍처로 각 시스템에서 컴포넌트가 구현되었으며, 정보 전송 및 로그 갱신 모듈은 모두 데몬으로 구현되어 배경 프로세스로 동작한다. 통신은 HTTP 프로토콜을 이용하고, 변경 로그는 XML[8] 형태로 전송된다. 데이터는 각 시스템이 가지는 키 값에 의해 암호화 되어 전송되고 이후 XML로 해석되어 처리된다. 이는 공개된 통신 환경에서도 보안이 유지 되도록 한다.

조달청에서의 시스템간 정보 동기화 시스

템은 2004년부터 2007년까지 3년간 수정 보완되었으며, 현재 지속적인 정보의 갱신을 무리 없이 처리하고 있다. 조달청 업무 프로세스의 특성상 정보 갱신 작업은 모두 결제 후에 반영되는 형태가 되는데, 변경 정보는 짧은 시간 동안 일괄적으로 처리된다. 2004년 구축 이후 한달 간의 모니터링 결과 매일 1000여건의 카탈로그 변경 이력이 로그 전송 및 반영까지 합쳐서 20분 내의 시간에 처리되었다. 이는 전체 운영시간에 비해 매우 짧은 시간 동안만이 정보 불일치를 나타냄을 의미한다. 이는 본 로그 기반 지연된 동기화 방법이 이같이 특정한 상황에서 유용하게 사용될 수 있음을 의미한다.

4. 결론

본 논문에서는 전자 카탈로그와 같이 하나의 시스템에서 변경이 발생하고, 다른 시스템으로 변경이 전파되는 특별한 상황에서의 로그 기반의 지연된 동기화 방법을 제시하였다. 멀티데이터 베이스의 특성상 각 시스템의 독립성을 유지하도록 하는 제약 안에서 다섯 가지 가정을 제시하고 이를 충족하도록 구현된 모듈은 지연된 동기화를 수행함을 보였다. 또한 조달청에 구현된 시스템과 이에 대한 분석을 통해 제안한 방법이 유용함을 보였다.

참고문헌

- [1] Garcia-Molina, H., “Global Consistency Constraints Considered Harmful for Heterogeneous Database Systems”, in: Proceedings of the First International Workshop on Interoperability in Multidatabase Systems, Kyoto, Japan, April 1991, pp. 248-250
- [2] Breitbart, Y., Garcia-Molina, H., and Silberschatz, A. 1992. “Overview of Multidatabase Transaction Management” The VLDB Journal 1, 2 (Oct. 1992), 181-240.
- [3] Rastogi, R., Korth, H.F., Silberchatz, A., “Exploiting Transaction Semantics in Multidatabase Systems”, in: Proceedings of the International Conference on Distributed Computing Systems, Vancouver, Canada, June 1995, pp. 101-109
- [4] Morpain, C., Cart, M., Ferrie, J., Pons, J.F., “Maintaining Database Consistency in Presence of Value Dependencies in Multidatabase Systems”, in: ACM SIGMOD Record, Montreal, Canada, June 1996, pp. 459-468
- [5] <http://www.pps.go.kr/>
- [6] Lee, T., Chun, J., Shim, J., and Lee, S. 2006. “An Ontology-Based Product Recommender System for B2B Marketplaces”. Int. J. Electron. Commerce 11, 2 (Dec. 2006)
- [7] <http://java.sun.com/>
- [8] <http://www.w3.org/XML/>

저자소개

이동주(e-mail: therocks@europa.snu.ac.kr)는 2003년 서울대학교 응용생물화학부에서 학사를 취득하고, 2003년부터 현재까지 서울대학교 대학원 컴퓨터공학과 석박사 통합과정에 재학 중이다. 관심분야는 전자 상거래, 시멘틱 웹, 개인화 추천 및 상황인지 서비스, 자연어 처리이다.

연종흠(e-mail: jonghm@europa.snu.ac.kr)은 2008년 서울대학교 컴퓨터공학과 학사를 취득하고, 2008년 현재 서울대학교 대학원 컴퓨터공학과 석사과정에 재학 중이다. 관심분야는 전자 상거래, 개인화 추천 및 상황인지 서비스이다.

이상구(e-mail: sglee@snu.ac.kr)는 1985년 서울대학교 계산통계학과에서 학사를 취득하고, 1987년 Northwestern University, 컴퓨터공학과에서 석사를 취득하였으며, 1990년 Northwestern University, 컴퓨터공학과에서 박사를 취득하였다. 1992년부터 현재까지 서울대학교 교수로 재직 중이다. 관심분야는 전자 상거래, 시멘틱 웹 서비스, 유비쿼터스 웹 서비스, 상황인지 서비스이다.