

전자상거래에서 패키지 상품판매를 지원하는 트랜잭션 관리기법

최 희 영[†] · 황 부 현^{††} · 허 기 태^{†††}

요 약

전자상거래 시스템에서 패키지 상품의 구매가 이루어지기 위해서는 패키지 상품을 구성하는 상품들이 구비되어 있어야 할뿐만 아니라 충분한 수량이 확보되어 있어야만 구매자의 요구가 있을 때 이를 실시간으로 처리할 수 있게 된다. 그리고 다수의 사용자가 동시에 가상쇼핑물에 접근하여 패키지 상품을 구매하고자 할 때에도 상품에 대한 동시성 제어가 효율적으로 이루어져야만 한다. 본 논문에서는 고정형 패키지과 맞춤형 패키지상품 구매 시 상품선택과 구매가 효율적으로 이루어지도록 하기 위해서 모든 상품에 대한 실 수량과 판매가능 수량정보를 유지토록 함으로써 구매자의 요구를 실시간으로 처리할 수 있도록 하였고, 트랜잭션의 처리율이 향상되도록 하였다. 그리고 트랜잭션의 동시성 제어를 효율적으로 하기 위해서 잠금 을 기반으로 한 동시성 제어 알고리즘을 제시하였다.

Transaction Management Technique to Support for Package Goods Sale in Electronic Commerce

Hee-Young Choi[†] · Bu-Hyun Hwang^{††} · Gi-Taek Hur^{†††}

ABSTRACT

In electronic commerce systems, for the purchase of package goods to be processed in realtime, the stock of their components has to be enough. When many purchasers access the electronic commerce systems and buy the package goods, the efficient concurrency control mechanism is required for improving the performance of the systems. In this paper, we propose a mechanism that processes the order of package products in realtime and improves the processing rate of transactions for buying ready-made package products and custom-made package products. The mechanism maintains the actual and saleable quantity for each component of a package good. The proposed mechanism is based on a new efficient concurrency control algorithm based on locking

1. 서 론

최근 웹 기술과 사용하기 편리한 웹브라우저의 출현으로 인터넷 사용이 급격히 증가하면서 인터넷을 비즈니스 측면에서 활용하고자 하는 노력이 계속되고 있으며,

그 일환으로 쇼핑물을 이용한 전자상거래(Electronic Commerce : EC)가 점차 확산되고 있다[1-3]. 쇼핑물은 다양한 형태로 사용자에게 상품정보를 제공해 주고 소비자의 취향과 가격에 맞는 상품을 선별하여 조언해 주거나 원하는 상품의 구매를 가능토록 하기 위해서 필요한 정보를 제공해 준다[1-3, 7].

* 본 논문은 한국과학재단 1999년도 특정 기초연구비(1999-2-30300-006-3)지원에 의하여 연구되었음
† 정 회 원 : 전남대학교 전산학과 시간강사
†† 정 회 원 : 전남대학교 전산학과 교수, 전남대학교 정보통신연구소
††† 총신회원 : 동신대학교 컴퓨터학과 교수
논문접수 : 2000년 1월 3일, 심사완료 : 2000년 8월 24일

대부분의 쇼핑물 시스템들은 공급자의 상품을 위탁 받아 상품을 진열하여 거래를 대행해 주고, 사용자에게 상품을 판매한다[10]. 쇼핑물은 사용자에게 다양한

상품을 제공하기 위해 공급자들로부터 위탁받은 여러 상품들이 모여서 하나의 상품을 구성하는 패키지(package)상품을 판매한다. 패키지 상품은 두 종류이상의 상품이 모여서 하나의 상품으로 구성된 상품이다. 이러한 패키지 상품은 구성하는 상품의 요소가 하나라도 만족하지 않으면 상품의 가치를 잃어버리기 때문에 수량이 한정된 패키지 상품을 판매하는 쇼핑몰과 공급자 사이에서 일관된 상품의 수량을 제공해야 하는데, 만약 상품의 수량이 불일치하면 쇼핑몰은 사용자에게 없는 상품을 판매하는 것과 같으므로 쇼핑몰과 공급자 사이에서 상품 수량 정보를 일관성있게 유지할 수 있어야 한다.

현재 모든 쇼핑몰들은 지불만 되면 거래가 이루어지도록 하고 있다[14, 15]. 그러나 지불이 된 상태에서 패키지 상품을 구성하는 하나의 상품이라도 없거나 또는 사용자가 시간과 노력을 들여 쇼핑을 한 상품들을 실제 구입하려고 할 경우 쇼핑몰에서 품절이라고 한다면 사용자들은 그 쇼핑몰을 신뢰하지 않을 것이며, 상품을 거래하는 거래율도 떨어질 것이다. 대부분의 쇼핑몰에서 제공하는 패키지 상품은 여행용 상품으로 사용자와 실시간으로 거래가 이루어지지 않고 있으며, 사용자가 주문하면 e-mail이나 전화로 통보를 해주고 상품 수량에 제한을 두지 않고 거래를 하고 있다. 그러나 공급자가 쇼핑몰에 제공하는 상품은 한정되고, 동일한 패키지 상품을 구매하려는 이용자가 증가함에 따라 쇼핑몰에 접근하는 순서대로 거래를 수행하도록 한다면 구매에서부터 지불까지의 시간으로 인해 많은 지연이 발생하게 된다[5, 6, 12]. 따라서 쇼핑몰에 거래되는 연산들이 인터리브하게 수행하도록 하기 위해서는 트랜잭션 단위로 처리를 해 주는데, 이 트랜잭션들이 접근하는 상품의 수량이 일관되게 유지하도록 하기 위한 동시성제어와 상품의 주문한 결과를 즉시 사용자에게 실시간으로 제공해 주기 위한 트랜잭션 관리를 해 주어야 한다[5, 6].

기존 분산 시스템에서 트랜잭션 관리는 2단계 잠금규약(2PL : Two-Phase Locking Protocol), 타임스탬프순서화 규약(TSO : Time Stamp Ordering Protocol), 직렬화 그래프(SG : Serialization Graph)등을 통하여 트랜잭션의 직렬성을 보장하고, 데이터베이스의 일관성을 유지하고 있다[8]. 타임스탬프 방법은 클럭을 동기화시켜야 하는 문제 때문에 일반적으로 많이 사용되는 방법은 2단계 잠금규약이다. 그러나 전자상거래 트랜

잭션은 쇼핑에서부터 구매까지의 과정을 사용자와 대화형태로 이루어지므로 트랜잭션의 수행기간이 길기 때문에 기존의 2단계 잠금 규약을 적용하면 다음과 같은 문제가 발생한다. 첫째, 관독과 쓰기잠금을 상품의 수량에 설정하여 트랜잭션을 처리하면 잠금을 설정한 상품에 대해서 다른 트랜잭션이 접근할 수 없기 때문에 트랜잭션이 지연으로 인해 동시성을 저하시킨다. 둘째, 패키지상품을 구성하는 요소상품이 한정된 수량일 때 모든 요소상품이 전부 만족해야 하므로 철회율이 높다. 셋째, 트랜잭션 지연으로 인해 성능이 저하된다.

본 논문에서는 동시성을 향상시키고 트랜잭션 지연으로 인해 성능의 저하를 감소시키기 위해서 수행하는 트랜잭션을 쇼핑·구매트랜잭션이라 하고, 이 트랜잭션을 장바구니 부트랜잭션과 구매 부트랜잭션으로 구분하여 수행한다. 장바구니 부트랜잭션은 사용자가 상품을 쇼핑하고 구매하고자 하는 상품의 수량을 선택하는 단계이다. 구매 부트랜잭션은 선택된 상품에 대해 지불이 이루어지고, 지불된 상품의 수량을 갱신하기 위한 트랜잭션이다. 쇼핑몰은 각 상품에 대해서 실수량과 판매가능한 수량값을 유지하고 사용자가 선택한 상품들의 수량을 데이터베이스에서 일관되게 갱신하기 위해서 잠금을 기반 한 동시성 제어기법을 제안하고자 한다.

본 논문의 구성은 다음과 같다. 2장에서는 기존연구에서 제시된 전자상거래 트랜잭션에 관한 연구와 기존 쇼핑몰에서 제한된 상품을 관리하는데 기존의 트랜잭션 관리 방법인 2단계 잠금 규약을 적용했을 때 발생하는 문제점을 살펴본다. 3장에서는 본 논문에서 사용한 전자상거래 모델과 트랜잭션 처리모형에 관해 기술하고, 4장에서는 제안한 알고리즘에 필요한 가정과 자료 구조에 기반한 효율적인 트랜잭션 동시성 제어 방법을 제시한다. 5장에서는 트랜잭션 대기 시간과 거래율에 따른 성능분석을 하였고, 6장에서는 결론과 추후 연구되어야 할 방향을 제시하였다.

2. 관련연구 및 문제정의

본 장에서는 인터넷 기반 한 전자상거래 시스템에서 트랜잭션 스케줄링 방법들에 대해 살펴보고, 방법에서 발생하는 문제점을 살펴본다. 먼저, 기존의 분산 환경에서 사용되는 2단계 잠금 규약과 전자상거래 시스템 환경에서 제안된 연구들에 대하여 살펴본다.

트랜잭션 스케줄링을 위해 2단계 잠금 규약을 사용할 때 각 트랜잭션은 접근하려는 데이터 항목에 해당 잠금을 얻어야 하며, 만일 충돌 관계에 있는 잠금이 이미 설정되어 있다면 그 잠금이 해제되기까지 기다려야 한다. 트랜잭션의 수행 중 일단 잠금을 해제하기 시작하면 다시 얻을 수 없다[7]. 이 방법을 한정된 패키지 상품을 판매하는 전자상거래 시스템 환경에 적용하면 다음과 같은 문제점이 발생한다. 전자상거래 시스템에 2단계 잠금 규약을 사용하면 전자상거래 트랜잭션의 특성이 사용자 상호작용(interaction)이 결부되기 때문에 기존 트랜잭션과 달리 장기간 지속될 수 있다. 이러한 경우 기존의 트랜잭션의 동시성 제어(예, 2PL) 기법은 불필요하게 오랜 기간의 잠금을 유지하여 시스템 전체의 동시성을 저하시킬 수 있다. 특히 쇼핑몰을 이용한 사용자 수와 감안 할 때, 오랜 기간 잠금에 의해 시스템 저하 영향은 매우 클 것이다. 전자상거래 트랜잭션 정확성(correctness)은 이와 같이 장기간 상호작용에서 발생 할 수 있는 동시성 저하 문제를 해결하여야 한다.

지금까지는 기존의 분산환경에서 사용되는 2단계 잠금 방법을 전자상거래 시스템 환경에 적용할 경우에 발생하는 문제점을 살펴보았다. 다음으로는 인터넷을 기반한 전자상거래 시스템 환경에서 잠금을 이용한 TCML-EC(Transaction Concurrency Control Method based on Lock in Electronic Commerce System)방법[6]에 대해 살펴본다.

TCML-EC 방법에서는 연계된 상품을 판매하는 전자상거래시스템이 일관성을 유지할 수 있도록 2단계 잠금 규약을 기반으로 한 동시성제어 방법을 제안하였다. 그리고 잠금의 사용으로 인하여 발생할 수 있는 트랜잭션 수행의 지연을 줄이기 위해서 검사 잠금과 할당 잠금의 새로운 잠금을 도입하고 교착상태 탐지 방법을 제안하였다. 데이터베이스에 저장된 각 상품은 공급자 시스템이 보유하고 있는 수량과 상품을 요구하는 트랜잭션에게 상품판매가 가능한지를 나타내는 Available 항목으로 구성하였다. 검사 잠금(Check_Lock)과 할당 잠금(Allocation_Lock)은 다른 잠금 모드와 양립가능하여 이 잠금들이 설정되었을 때 다른 모드의 잠금 설정을 허용한다. 검사 잠금은 트랜잭션 T_i 가 상품 x 에 대해 요구한 수량을 할당할 수 있는지를 검사하기 위해서 데이터 아이템 x 와 Available(x)에 요구되어지는 잠금이다. 또한 할당 잠금은 T_i 가 상품 x 에 대해 요구

한 수량을 할당하기 위해 데이터 항목 x 와 Available(x)에 요구되어지는 잠금이다. 검사 잠금은 트랜잭션의 철회/대기 및 할당을 결정하는 잠금으로 트랜잭션이 요구한 수량만큼 상품을 할당가능 할 때 할당 잠금으로 전환한다. 그리고 트랜잭션의 모든 상품들이 할당 잠금을 얻었을 때 쓰기 잠금으로 변환하여 실수량을 갱신한다. 이때 쓰기 잠금은 모든 잠금과 양립가능하지 않기 때문에 쓰기 잠금으로 인해 다른 트랜잭션이 지연된다. 이에 대한 예는 다음과 같다.

【예 1】 TCML-EC 방법에서 발생하는 트랜잭션 지연

쇼핑몰에서 제공하는 연계된 상품은 다음과 같이 구성되어 있고, 데이터베이스에 유지하고 있는 요소상품마다 실수량과 Available한 수량이 각각 10개씩 있다고 가정하자. 다음 예에서 C연산과 A연산의 표기는 C(검사하고자 하는 상품명, 사용자 요구수량), A(할당할 상품명, 할당받은 상품수량)이다.

연계된 상품 A : 비행기티켓(a), 숙박(b),

연계된 상품 B : 비행기티켓(a), 숙박(b), 자동차렌트(c)

연계된 상품 C : 비행기티켓(a), 숙박(b), 관광권(d)

$T_1 : R_1(a) C_1(a,5) A_1(a,5) R_1(b) C_1(b,5) A_1(b,5)$
 $W_1(a) W_1(b)$

$T_2 : R_2(a) C_2(a,6) A_2(a,6) R_2(b) C_2(b,6) A_2(b,6)$
 $R_2(c) C_2^6(c,6) A_2^6(c,6) W_2(a) W_2(b) W_2(c)$

$T_3 : R_3(a) C_3(a,3) A_3(a,3) R_3(b,5) C_3(b,5) A_3(b,5) R_3(d) C_3$
 $(d,5) A_3(d,5) W_3(a) W_3(b) W_3(d)$

(여기에서 R(a)는 상품 a에 대한 실수량 항목을 판독연산, C(a,5)는 상품 a에 대해서 요구한 5개 수량이 만족한가를 검사연산, A(a,5)는 수량이 만족한 상품 a를 5개 할당연산, W(a)는 상품 a에 대한 실수량 항목을 갱신하는 연산)

TCML-EC 방법에서 전자상거래 트랜잭션 T_1 , T_2 , T_3 는 각각 연계된 상품 A, B, C를 주문한다. 각 상품을 구성하는 연계된 상품이 만족한가를 검사하기 위해 쇼핑몰은 공급자 시스템으로 하나의 이동에이전트를 파견한다. 이동에이전트는 각 공급자 시스템에서 상품의 실수량과 Available수량을 검사하고 다음 공급자 시스템으로 이동한다. 수량검사는 상품에 검사잠금을 설정하여 상품의 수량과 비교하게 된다. 트랜잭션 T_3 은 a 상품을 3개 할당 받아 Available 항목의 수량이 7로 갱신되고, 트랜잭션 T_1 은 a 상품을 5개 구매하기

위해서 먼저 판독한 실수량 10과 비교하여 만족하므로 Available항목과 비교하게 된다. 따라서 Available 항목의 수량은 2로 갱신된다. 트랜잭션 T_2 도 역시 a 상품의 구매하기 위해서 수량 6개를 주문한다. 그러나 검사잠금에서 상품의 실수량은 만족하지만 Available 항목의 수량이 만족하지 않으므로 수량이 만족할 때까지 대기 상태가 된다. 상품의 실수량 갱신은 연계된 모든 상품이 할당잠금을 다 가지고 있으면 쓰기잠금으로 전환하여 수량을 갱신한다. 그러나 a 상품에 대해 T_1 과 T_2 가 판독잠금을 소유하고 있으므로 충돌로 인해 트랜잭션 T_3 은 판독잠금을 해제할 때까지 기다리게 된다. 따라서 현재 할당받은 상품을 구매할 수 있는 수량인데도 다른 사용자가 구매를 마칠 때까지 기다려야 하는 문제가 발생한다. □

예를 통해서 알 수 있는바와 같이 인터넷을 기반으로 한 전자상거래 쇼핑물 환경에 기존의 트랜잭션 관리 기법을 그대로 사용하는 것이 너무 제약적이고, 쓰기 잠금이 설정된 패키지 상품에 대해서는 다른 트랜잭션들이 접근할 수 없기 때문에 동시에 동일한 상품을 구입하는 사용자 수가 많을수록 기존 2단계 잠금 규약에서 제공되는 연산의 충돌 관계로 인해 트랜잭션이 지연됨에 따라 동시성이 저하된다. 따라서 이를 해결할 수 있는 효율적인 트랜잭션 관리 기법이 필요하다.

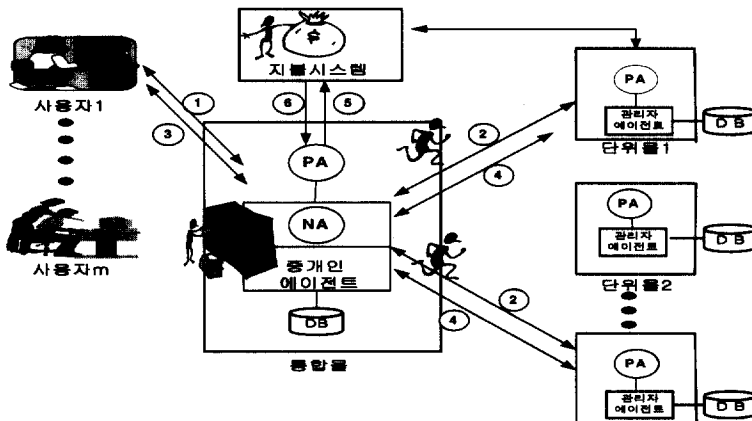
3. 전자상거래를 위한 시스템 모델

본 논문에서 제안하는 전자상거래 시스템 모델은 사용자, 통합물, 단위물, 지불시스템으로 구성된다. 사용

자는 물품구매를 위해 웹 사이트를 통해 통합물로 접근하거나, 단위물로 접근하여 거래를 행할 수 있다. 통합물은 다수의 물들을 하나로 통합하는 물으로써 단위물에서 위탁한 상품들을 판매해 주거나 여러 단위물에서 제공된 상품들로 구성된 패키지 상품들을 판매한다. 단위물은 직접 사용자와 거래를 하며, 상품들을 위탁해서 판매하는 물이다. 지불시스템은 사용자가 요구한 상품의 대금을 지불하는 시스템으로 은행 또는 신용카드회사이다.

제안하고 있는 전자상거래 시스템 특징은 사용자가 통합물을 통해 다수의 단위물에서 판매하는 상품들을 비교 구매할 수 있다는 점이다. 또한 하나의 통합물을 통해 연결된 다수의 단위물에 주문하여 상품을 구입할 수 있으며, 상품의 대금은 통합물에서 원-스톱지불시스템으로 처리할 수 있다. 또한 사용자에게 다양한 상품 정보를 보여줄 수도 있다. 원-스톱 지불 시스템이란 상품 구매에 대한 결제를 무 통장 입금이나 지로용지를 통해 지불하는 것이 아니라, 상품구매와 함께 결제까지 한꺼번에 가능한 시스템이다[13]. 사용자는 수량이 한정된 상품을 통합물과 단위물을 통해 구입할 수 있으며, 단위물은 다수의 사용자와 독자적으로 주문 및 지불처리를 할 수 있다.

(그림 1)에서 통합물 시스템은 고객에게 하나의 물인 것처럼 보여지고 단위물에서 제공되는 상품정보의 목록을 요약해서 유지한다. 유지되는 상품정보는 분야별로 색인되어 있다. 예를 들어 여행상품, 채소류, 의류, 유아용품, 과일류등의 상품의 목록을 데이터 베이스에 유지하고 있다.



(그림 1) 에이전트를 기반으로 한 전자상거래 시스템 모델

통합물에서 수행되는 트랜잭션처리는 사용자가 선택한 패키지 요소상품이 전부 만족하는지를 결정하기 위해서 요소 상품을 판매하는 단위물에 상품을 요구하여 전부 만족하면 장바구니에 등록시키는 일을 수행하며, 또한 구매요청에 따라 단위물이 가지는 상품의 수량을 갱신시키고 은행과 연계시켜 지불이 이루어지도록 하기 위한 트랜잭션으로 쇼핑·구매트랜잭션이라 한다. 이 시스템은 사용자와 단위물 시스템과의 거래를 중재하고 트랜잭션 처리를 관리해 주는 중개인 에이전트(Broker Agent : BA)[7]와 지불을 담당하는 지불에이전트(Payment Agent : PA), 단위물에 있는 상품을 검색하여 통합물에 등록시키고 사용자의 요구에 맞는 상품을 검색해 주는 검색 에이전트(Navigation Agent : NA)모듈로 나누어진다. 따라서 여러 요소 상품들로 구성된 패키지 상품을 통합물에서 제공함으로써 사용자는 요소 상품들을 구매하기 위해서 여러 사이트를 방문할 필요 없이 더 편리하게 거래를 행할 수 있다.

단위물(Unit Mall :UM)은 통합물(Integrated Mall :IM)을 통해 구매자와 거래를 할 수 있으며, 독립적인 물의 운영이 가능하고 통합물을 거치지 않고 구매자와 직접 거래를 할 수 있는 백화점 형태의 물이나 전문형의 물이다. 각 단위물의 관리자 에이전트(Management Agent : MA)는 취급하는 상품 정보를 데이터 베이스에 유지하며, 통합물에서 상품의 자세한 정보를 요구했을시 상품 정보를 제공한다. 그리고 구매자에게 판매할 상품을 전시하고, 통합물과 상품판매의 계약대행을 체결한다. 또한 통합물의 요청에 따라 상품에 대한 자세한 정보를 제공하고, 제한된 수량을 가지는 해당 상품의 재고를 관리한다. 따라서 주기적으로 통합물에 변동되는 상품과 품절 상품에 대한 정보를 통합물에 제공한다. 즉 사용자와의 거래는 단위물 데이터베이스에 들어 있는 상품 정보를 기준으로 수행한다. 데이터 베이스에 유지하는 상품 정보는 상품명, 상품코드, 판매가, 할인율, 실수량, 판매가능한 수량등에 대한 정보를 제공한다. 각 단위물이 통합물을 거치지 않고 직접 사용자와 거래를 할 경우에 지불은 단위물의 지불에이전트(Payment Agent : PA)에 의해 수행한다.

제안하는 모델에서는 전자상거래 트랜잭션을 쇼핑·구매트랜잭션이라 한다. 쇼핑·구매 트랜잭션은 사용자가 통합물에 제출한 구매 요구를 단위물 시스템에 반영하기 위해 존재하는 데이터 베이스 연산들의 집합으로 구성된 원자적인 작업단위이다. 이 트랜잭션은 장

바구니 부트랜잭션과 구매 부트랜잭션으로 구분하였다. 장바구니 부트랜잭션은 사용자가 구매하고자 하는 패키지 상품과 수량을 선택하면 통합물의 중개인 에이전트에 의해서 각 요소상품으로 분류하고, 해당 단위물의 관리자에이전트에게 요구한 수량의 여부를 받아 장바구니에 할당가능한지를 검사하는 트랜잭션이다. 구매 부 트랜잭션은 장바구니의 패키지 상품을 사용자가 구매의사를 하였을 때 단위물의 상품의 수량을 갱신하고 결제하는 트랜잭션이다. 상품구매 절차는 (그림 1)에서 번호순으로 수행한다.

4. 패키지 상품을 지원한 트랜잭션 관리 알고리즘

본 장에서는 패키지 상품을 판매하는 통합물에서 잠금을 이용한 동시성제어를 위한 알고리즘에 필요한 과정과 자료구조를 기술한다. 패키지 상품이란 두 개의 요소상품이 모여서 구성된 하나의 상품으로 고정형 패키지상품과 맞춤형 패키지상품의 종류로 나눈다. 고정형 패키지 상품은 요소상품의 종류가 미리 정해져 있는 상품으로 사용자가 임의로 요소상품을 변경할 수 없다. 맞춤형 패키지 상품은 사용자가 원하는 요소상품의 목록을 보고 선택해서 패키지상품을 구성한다.

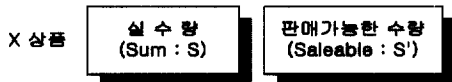
4.1 가정 및 자료구조

본 논문에서 사용하고 있는 가정은 다음과 같다.

- 패키지 상품은 통합물을 통해서만 구매가능하다.
- 패키지를 구성하는 각 상품들이 하나라도 만족하지 않으면 그 트랜잭션은 철회된다.
- 쇼핑·구매 트랜잭션은 장바구니 부트랜잭션에서 선택된 상품들만 구매 부트랜잭션에서 구매할 수 있다.
- 각 단위물에서 사용자가 장바구니에 할당받은 상품의 수량을 취소하는 룰을 k%라 한다.
- 통합물에서 단위물로 보내는 데이터에 대한 통신지연은 없다고 가정한다.

제안하는 알고리즘을 위해 통합물과 단위물에 다음과 같은 자료구조를 유지한다. 먼저 단위물에서 유지되어야 자료구조는 각 상품의 수량정보를 '실수량' 항목과 '판매가능한 수량' 항목으로 분류하였다. 실수량 항목은 데이터베이스에 유지되어 있는 상품 수량을 나타내고, 판매가능한 수량 항목은 현재 시스템에 등록되어 있는 상품의 실수량에서 아직 완료되지 않은 트

랜잭션에 할당된 수량을 더한 값이다. 사용자들이 장바구니에 담은 상품은 다시 꺼내놓을 수도 있기 때문에 100% 구매한다는 보장을 할 수 없으므로 사용자들이 장바구니의 상품을 취소하는 물을 조사하여 통계처리 한 결과가 k%라 할 경우 각 단위물에서 판매가능한 수량을 실수량보다 k%만큼 더 사용자에게 제공함으로써 정해진 시간 안에 k%의 취소율만큼 더 거래율을 향상시킬 수 있다. 따라서 제안하는 알고리즘에서는 k%의 수량만큼 더 판매가능한 수량을 초기화시키고, 트랜잭션이 실행후에 실수량이 영(0)인 상품은 더 이상 불필요한 연산이 수행되지 않도록 판매가능한 수량을 영(0)으로 초기화 시킨다. 이에 대한 자료구조는 (그림 2)와 같다.



(그림 2) 각 상품 데이터 항목의 구성

통합몰은 패키지의 요소상품에 대한 쇼핑정보 테이블을 유지한다. 이 테이블은 현재 통합몰에서 수행하고 있는 트랜잭션번호(Tr-id), 사용자가 요구한 패키지 상품명(Name), 패키지상품을 구성한 요소상품(Sub-name), 판매가능한 수량의 만족여부를 나타내기 위해서 모드 M1, 실수량의 갱신 여부를 나타내기 위해서 M2의 모드(Mode)의 값을 유지하며 (그림 3)과 같다.

Tr_id (트랜잭션번호)	Name (패키지 명)	Sub_name (요소상품 명)	Mode(모드)	
			M1	M2

(그림 3) 쇼핑정보 테이블

쇼핑정보테이블에 등록되는 경우는 사용자가 통합몰로 패키지 상품을 요구했을시 통합몰에서 패키지 상품별로 구성되는 요소상품을 테이블엔트리에 삽입하고 각 단위물에 상품을 요구한다. 단위물에서는 요소상품의 판매여부를 통합몰로 보내면 통합몰에서는 모든 요소상품이 만족해야 하므로 그 결과 값을 테이블의 모드 값에 등록한다. 모드값은 '0'과 '1'의 두 개의 상태값을 가질 수 있다. '1'이 등록되는 경우는 단위물에서 요소상품이 있다는 것이고, '0'은 단위물에서 상품이 없는 경우이다. 따라서 패키지상품의 구입여부는 모드값에 따라 결정된다. 요소상품의 모드값이 하나라도 '0'의 값

을 가지면 그 패키지 상품은 취소(returned)되고, 테이블에서 삭제된다. 따라서 이 테이블의 자료는 트랜잭션이 완료 또는 철회될때 까지 유지된다.

4.2 동시성제어

이 장에서는 통합몰과 단위물이 상품 수량에 대한 정보를 일관되게 유지하고 상거래에 불필요한 트랜잭션의 지연을 줄이기 위해 잠금을 이용한 동시성제어 기법으로 CCL-EC (Concurrency Control method using Locking for Electronic Commerce)방법을 제안한다. CCL-EC알고리즘은 두가지 방법으로 스케줄링할 수 있다. 하나는 고정형 패키지 상품으로 요소상품의 구성이 미리 고정되어 사용자가 임의대로 바꿀 수 없는 상품이다. 다른 하나는 맞춤형 패키지 상품이다. 이 상품은 통합몰에서 제공하는 요소상품의 목록을 보고 사용자가 임의대로 패키지를 구성하는 상품이다. 먼저 고정형 패키지 상품의 경우는 다음과 같은 통합몰 서버알고리즘으로 구성한다.

● 통합몰서버 알고리즘

통합몰에서 수행되는 쇼핑·구매 트랜잭션의 동시성제어를 위한 알고리즘은 다음과 같다. 통합몰의 중개인 에이전트는 구매자 에이전트와 계속적인 연결을 유지하는 동안 통합몰에서 수행되는 작업은 크게 2가지로 구분할 수 있다. 하나는 패키지 상품을 구성하는 요소상품 목록에 따라 중개인 에이전트가 단위물로 장바구니 부트랜잭션을 제출하거나 사용자의 구매요구에 따라 쇼핑정보 테이블의 모드 값을 참조하여 구매 부트랜잭션을 각 단위물로 제출하는 일이다. 다른 하나는 각 단위물 관리자 에이전트에서 보내는 수량 갱신 여부에 따른 테이블을 유지하여 모드의 값과 지불의 결과에 따라 트랜잭션을 완료 또는 취소시키는 것을 행한다.

CCL-EC방법에서 고정형 패키지상품은 통합몰의 중개인 에이전트에 의해서 요소상품을 판매하는 단위물로 연결하여 판매가능한 수량을 갱신하는데 동시에 2개 이상의 트랜잭션의 수행으로 인하여 발생하는 데이터의 불 일치성이 발생하기 때문에 요소상품의 수량검사와 갱신을 위해 배타적인 잠금을 사용한다. 이 잠금은 판매가능한 수량과 실수량을 판독하여 갱신하고 쓰는 연산을 행한다.

임의의 쇼핑·구매트랜잭션이 시작하기전 각 상품의

수량에 저장된 값을 실 수량보다 K% 만큼 더 많이 판매가능한 수량을 조정하고, 트랜잭션이 시작하면 장바구니 부트랜잭션에서 선택한 수량의 갱신은 판매가능한 수량에서 이루어지고, 구매 부 트랜잭션에서 갱신한 수량은 실수량에서 이루어진다. 그러나 동시에 서로 다른 통합물이 단위물의 요소상품을 구매하고자 한다면 단위물의 관리자 에이전트는 먼저 잠금을 요구한 트랜잭션에게 잠금을 할당하게 되는데, 이 잠금은 모든 단위물의 요소상품의 실수량을 갱신할 때까지 소유한다면 트랜잭션이 지연되므로 배타적인 잠금을 설정하여 수량 갱신 후 바로 잠금을 해제하는 방법을 사용한다. 그러나 데이터베이스의 일관성을 유지하기 위해서 하나의 단위물에서 정상적인 수량의 갱신이 이루어지지 않았다면 이 패키지 상품은 취소되어야 하므로 현재 요소상품의 수량을 정상적으로 갱신하였는가의 정보는 통합물의 쇼핑정보테이블에서 유지한다. 장바구니 부트랜잭션과 구매 부트랜잭션은 테이블의 모드인 M1의 값이 모두 '1'이면 장바구니에 할당하고, M2의 값이 모두 '1'이면 증가인 에이전트는 지불에이전트를 생성하여 지불트랜잭션을 수행하도록 한다. 따라서 통합물에서 유지하는 쇼핑정보 테이블의 모드 값에 따라 장바구니 할당과 구매가 결정되므로 이를 검사하는 알고리즘은 알고리즘1과 같다.

/ UM : 단위물, M1: 판매가능한 수량 모드, M2: 실수량 모드, P_list(Y) : 패키지 상품의 요소상품, k: 요소상품의 개수, P: 주문한 수량*/*

Shopping and Ordering transaction SOT request the Mode_Type

case requested Mode_Type

M1 : { IF (M1=='1')
THEN cancel the requested P of element
data_item to the UM;
ENDIF;
}

delete package data_item at the Table

M2 : { IF (M2=='1')
THEN cancel the requested P of element
data_item to the UM;
ENDIF;
}

delete package data_item at the Table

(알고리즘 1) 쇼핑정보테이블의 모드 검사

쇼핑·구매트랜잭션의 철회는 패키지 상품의 대금이 실패되는 경우와 사용자의 요구에 의해서 철회되는 경우이다. 한 사용자는 여러 종류의 패키지 상품을 구입할 수 있으므로 하나의 패키지 상품이 만족하지 않는다고 해서 전체 트랜잭션을 철회시키지 않고 수량이 만족한 패키지 상품만을 수행할 수 있도록 하였다. 통합물에서 수행되는 고정형 패키지 상품에 대한 CCL-EC 알고리즘은 알고리즘2와 같다.

/ 통합물서버 알고리즘 */*

/ SOT(Shopping and Ordering Transaction) :쇼핑·구매 트랜잭션, S'=S+S*K% (K% : 취소율과 반품률, S' : 판매가능한 수량), S : 실수량, P : 선택한수량, Ti : i 번째 트랜잭션, Bank :대금지불하는 은행, TABLE(Tr_id, Name, Sub_name, M1,M2), P_list[Y]패키지 상품의 구성 목록, UM: 단위물, IM:통합물 */*

SOT request a new button

case request type

ordeing : request package data_item P_list[Y] and its S' to the UM;

IF(the message sended from the UM =='ACK')

THEN M1='1' ;

ELSE M2='0' ;

invoke the ModeCheck Procedure ;

break;

cancel : invoke the ModeCheck Procedure ;

break

purchase : request each Xi of transaction the S of S_List[Xi] to the UM;

IF(the message sended from the UM =='ACK')

THEN (M1=='1');

ELSE M2='0';

}

The payment Agent excute for payment ;

IF(the message sended from Payment System is 'NAK')

THEN send the 'failed payment' message to the user; // 지불실패됨

invoke the ModeCheck Procedure ;

ELSE send the 'completed payment' message to the UM;

break;

(알고리즘 2) CCL - EC 알고리즘

맞춤형 패키지 상품은 사용자가 통합물에서 제공하는 목록을 확인하고 각 요소상품을 선택하여 구성한

다. 선택한 요소상품이 구매 가능한가를 확인하기 위해서 통합몰에 제출되면 통합몰 중개인 에이전트는 사용자가 선택한 요소상품을 해당 단위물로 분류하여 제출한다. 나머지는 고정형 패키지 상품과 동일하다.

*send the user select element data_item to the IM
invoke the Fix-type algorithm*

(알고리즘 3) 맞춤형 패키지 알고리즘

● 단위물 서버 알고리즘

단위물에서 수행되는 쇼핑·구매트랜잭션의 동시성제어를 위한 알고리즘은 다음과 같다. 단위물에서 수행되는 작업은 통합몰에서 요구한 판매가능한 수량과 실수량의 갱신요구에 응답하는 것이고, 단위물 자체에서 수행되는 연산들과 통합몰에서 요구한 연산들을 데이터베이스의 일관성을 유지할 수 있는 동시성제어를 하는 일이다. 단위물이 통합몰의 중개인 에이전트에서 수량을 갱신하는 요구가 들어오면 단위물의 관리자 에이전트는 현재 수행하는 연산과 충돌이 발생하지 않으면 잠금을 설정한다. 장바구니 부트랜잭션에서는 판매가능한 수량을, 구매 부트랜잭션에서는 실수량을 갱신하고 해당 통합몰 서버에게 응답한다.

단위물로 사용자가 직접 접속하여 거래를 요청한 트랜잭션들은 패키지 상품이 아니기 때문에 쇼핑정보 테이블을 유지하지 않는다. 먼저 장바구니 부트랜잭션은 판매가능한 수량이 만족하면 장바구니에 할당하고 구매 부트랜잭션에서는 실수량이 만족하면 지불시스템에게 지불대금을 의뢰한다. 지불시스템에서 지불이 정상적으로 마쳤다면 쇼핑·구매트랜잭션은 완료(commit)된다. 단위물에서 상품의 판매되는 과정은 ordering, cancel, purchase모듈로 구성한다. 쇼핑 부트랜잭션에서 사용자가 상품의 카탈로그와 상품의 수량을 보는 단계는 데이터 베이스에 있는 정보를 단지 보기만 하고 데이터 베이스의 일관성에 영향을 미치지 않으므로 생략한다. 알고리즘 4에서 제공되는 각 모듈의 기능은 다음과 같다.

ordering 모듈은 사용자가 패키지 상품을 선택하면 그 상품을 구성하는 상품의 수량이 할당가능한가를 사용자에게 제공해 주기 위해서 배타적인 잠금을 사용해서 상품의 판매가능한 수량을 검사하고 갱신하는 연산들을 수행한다.

cancel 모듈은 선택한 상품의 수량을 취소할 경우에 수행되는 연산들로서 사용자가 장바구니에 담은 패키지 상품을 취소할 때 수행된다. 이 때 취소 연산을 수행하기 위한 상품들은 통합몰에서 유지되는 쇼핑정보 테이블의 모드가 '1'인 상품들이다.

purchase 모듈은 사용자가 장바구니의 상품들을 구입하기 위해서 장바구니의 상품 목록에 잠금을 설정하고 실수량을 갱신하는 연산을 수행하는 모듈이다. 그러나 정해진 시간동안 사용자의 거래율을 더 높이기 위해서 통합몰에서는 사용자가 상품을 취소하는 취소율을 주기적으로 조사하여 그 취소율 k% 만큼 더 판매가능한 수량에 적용하므로 실수량이 음수값을 가진다면 없는 상품을 판매하는 것과 같다. 따라서 음수값이 나온 상품은 취소연산을 수행한다. 이 때 취소 연산을 수행하기 위한 패키지 상품은 통합몰에서 유지되는 테이블의 모드가 '1'인 상품들이고 단위물로 직접거래를 요구한 트랜잭션은 장바구니에 할당된 상품의 수량을 취소시킨다.

/ 단위물에서의 동시성제어 알고리즘 */*

/ SOT(Shopping and Ordering Transaction) : 쇼핑·구매 트랜잭션, PT(Payment Transaction) : 지불트랜잭션, S_List [Xi]: 장바구니 안의 상품리스트, S' = S' + S*K% (K% : 취소율과 반품률, S' : 판매가능한 수량) S : 실수량, P : 선택한수량, saleable_number : 판매가능한 수량. read_number : 실수량 */*

SOT requested a new type ;

case requested type
ordering :

IF(there is no conflict with setting lock) //사용자가 선택한 상품이 가지는 판매가능한 수량요구
THEN

{ get a Lock;
read S' ;
IF (S' ≠ 0 and P ≤ S')
THEN update saleable_number; // 주문한 수량을 판매가능한 수량으로부터 빼준다.
S_list[Xi] = S_list[Xi] + X(P);
ELSE send S' and S to the user;
Unlock ; }

ELSE block;
break;

cancel :

case requested update data_item X and its S' or S;
saleable_number : IF(there is no conflict with setting lock)
THEN get a Lock;


```

                update saleable_number;
                Unlock;
            ELSE block;
            break;
real_number: IF(there is no conflict with setting
lock)
            THEN get a Lock;
                update read_number;
                Unlock
            ELSE block ;
            break;
purchase :
    IF(there is no conflict with setting lock)
    THEN get a Lock ;
        update real_number;
        IF(S < 0)
        THEN case requested goods type ;
            package_goods : send the 'pack-
age goods' message to the IM;
            break;
            element_goods : send the 'ele-
ment goods' message to the user;
            cancel the requested data_item;
            break;
        ELSEIF(S=0)
            THEN send the 'NAK' message to the IM;
                S'=-0;
            ELSE case requested goods type;
                package : send the 'package goods'
                message to the IM;
                break;
                goods : request the payment ;
                IF(the message sended from
                payment system=='NAK' )
                THEN cencel the requested
                transaction Ti ;// 지불실패됨
                send the 'failed payment'
                message to the user;
                transaction abort;
                ELSE commit and send the
                'completed payment' mess-
                age to the user;
            Unlock;
        break;
    
```

(알고리즘 4) 단위물에서의 CCL-EC 알고리즘

【예제 2】 CCL-EC 알고리즘의 예

통합물에서 제공되는 패키지상품의 형(type)은 고정형과 맞춤형 패키지 상품으로 구성되어 있는데 맞춤형 패키지 상품은 상품의 선택한 후의 과정은 고정형 패키지 상품의 과정과 동일하므로 이 예제에서는 다루지

않는다.

통합물에서 제공되는 고정형 패키지 상품은 다음과 같이 구성되어 있고 각 패키지의 요소상품은 각 단위 물에서 판매한다고 가정하자. 각 단위물에서 유지하는 상품의 실수량과 판매가능한 수량의 실행 전 상태는 <표 1>과 같다.

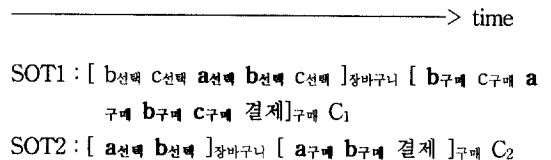
- 통합물
- 단위물
- 상품 A : 비행기티켓(a), 숙박(b), 단위물1 : a 상품
- 상품 B : 숙박(b), 관광(c) 단위물2 : b 상품
- 상품 C : 비행기티켓(a), 숙박(b), 관광(c) 단위물3 : c 상품

<표 1> 트랜잭션을 수행하기 전의 데이터베이스 상태

수량	상품명	실행 전 상태		
		a	b	c1
실 수 량		10	10	10
판매가능한 수량		12	12	12

각 단위물에서 유지되는 판매가능한 수량은 사용자 들이 장바구니에 담은 상품을 취소하는 물이 20%라 가정하면 실수량과 판매가능한 수량의 초기값은 <표 1>과 같다. 상품의 구입은 사용자가 선택한 패키지 상품의 수량만큼 각 요소상품도 동일한 수량을 구입한다고 하자.

user1 : Package B(5) Package C(3)
 user2 : Package A(5)



- b선택 : 상품 b의 판매가능한 수량을 read와 write가 한쌍으로써 동시에 수행
- b구매 : 상품 b의 실수량을 갱신(read와 write가 동시에 수행)

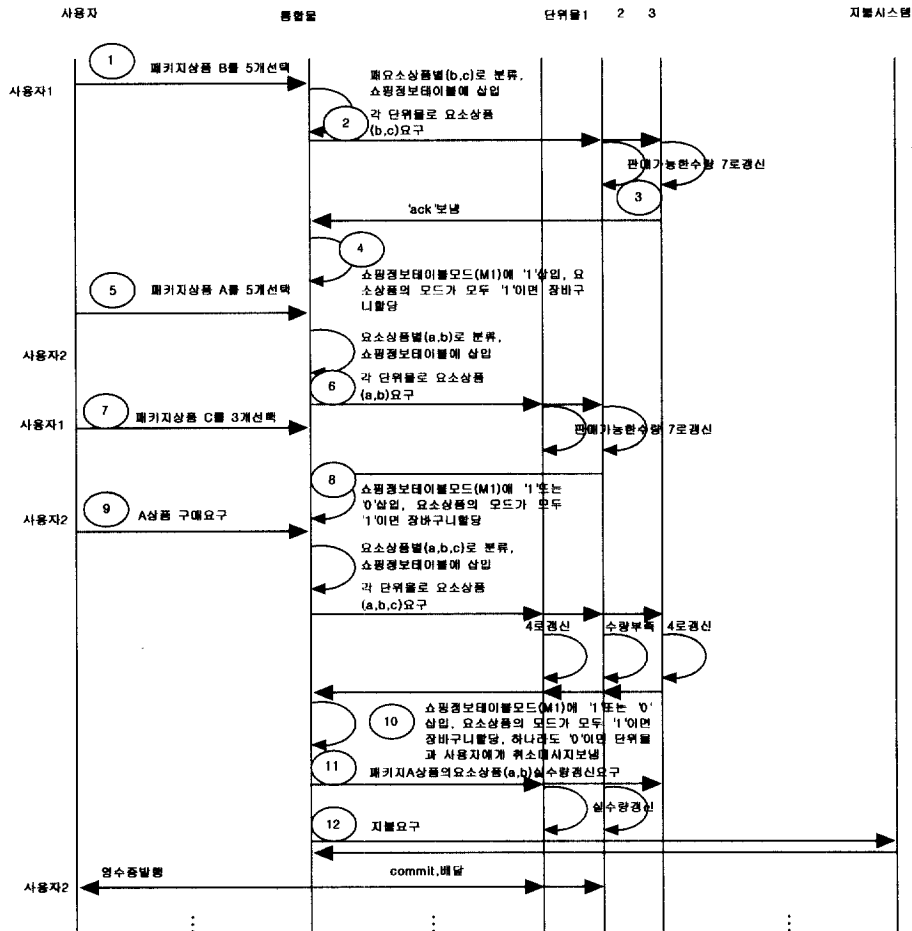
쇼핑·구매트랜잭션 SOT1과 SOT2가 장바구니 부트 랜잭션, 구매 부트랜잭션 단위로 수행된다면 위의 스 캐줄링에서 진한부분의 연산은 충돌이 발생하는 연산 이다. 기존 TCML-EC방법에서는 잠금을 해제하는 단 계가 트랜잭션의 마지막 부분에서 이루어지기 때문에 너무 오랫동안 잠금을 소유하고 있어서 잠금으로 인해

다른 트랜잭션의 수행을 지연시키는 문제가 발생하게 된다. 제안하는 CCL-EC방법은 일관된 데이터베이스를 유지하기 위해서 장바구니 부트랜잭션에서는 장바구니에 상품을 할당하기 위해서 판매가능한 수량을 갱신하는 동안에만 잠금을 설정하고 해제시키며, 구매 부트랜잭션에서는 실수량을 갱신할 때만 잠금을 설정하고 해제시킴에 따라 TCML-EC에서 발생하는 트랜잭션 지연으로 인해 동시성을 감소하는 문제를 해결할 수 있었다. 그러나 패키지상품은 각 단위물에서 요소상품이 정상적으로 수량을 갱신하였다 하더라도 다른 단위물의 요소상품이 전부 만족해야 하므로 통합물에서 유지되는 쇼핑정보테이블의 모드 값에 따라 결정된다. 쇼핑·구매트랜잭션 SOT1과 SOT2가 통합물과 단위물에 <표 2> (그림 4)의 ⑧번 단계 쇼핑정보서 수행되는

단계는 (그림 4)와 같다. (그림 4)의 통합물에서 유지되는 쇼핑정보테이블은 <표 2>와 같고, 쇼핑·구매트랜잭션 SOT1과 SOT2를 실행한 후의 테이블의 모드상태는 <표 3>과 같다.

<표 2> 그림 4의 ⑧번 단계 쇼핑정보 테이블 모드 상태

Tr-id	패키지 상품명	구성상품	모드	
			M1	M2
T1	B	b	1	
		c	1	
	C	a	1	
		b	0	
T2	A	a	.	
		b	.	



(그림 4) 쇼핑·구매트랜잭션 수행 단계

〈표 3〉 트랜잭션 T1과 T2의 실수량 갱신후의 쇼핑정보 테이블의 상태

Tr-id	패키지 상품명	구성상품	모드	
			M1	M2
T1	B	b	1	1
		c	1	1
T2	A	a	1	1
		b	1	1

〈표 4〉 쇼핑·구매트랜잭션 실행 후 데이터베이스의 상태

수량	상품명	실행 후 상태		
		a	b	c
실 수량		5	0	5
판매가능한 수량		7	2	7

· 실수량이 0인 제품은 판매가능한 수량도 0으로 초기화 한다.

5. 성능분석

본 장에서는 제안한 CCL-EC 방법의 성능을 분석하기 위한 매개변수를 기술하고 2단계 잠금으로부터 트랜잭션이 대기되는 시간과 거래율에 따른 성능을 분석한다. 그리고 CCL-EC 방법의 성능을 분석하기 위하여 참고문헌 [6]에서의 TCML-EC 방법을 비교대상으로 한다. TCML-EC 방법은 쇼핑물로 제출된 트랜잭션을 수행하기 위해서 이동 에이전트를 각 공급자 물로 보내서 판독과 쓰기잠금 외에 검사 잠금과 할당잠금을 이용해서 트랜잭션을 수행한 후 그 결과를 쇼핑물로 알려주는 방법이다. TCML-EC 방법은 2단계 잠금 규약을 기본으로 한다. 본 장에서 성능분석을 위해 사용되는 매개변수는 다음과 같다.

- n : 트랜잭션이 접근하는 평균데이터 항목수
- k% : 장바구니에 할당된 상품을 취소하는 평균률
- P_{μ} : 접근하는 데이터 항목이 메모리에 있을 확률 (mem_size/db_size)
- io_time : 데이터를 입출력하기 위해 디스크에 접근하는 시간
- T_r : 읽기 연산의 평균수행시간
- T_w : 쓰기 연산의 평균수행시간
- T_0 : 트랜잭션의 연산 제출시간
- T_{move} : 이동에이전트가 한 시스템으로 이동하기 위한 평균시간

디스크의 입출력을 하지 않는 각 연산의 CPU 처리 시간은 매우 짧으므로 입출력 시간을 제외한 각 연산

의 처리시간을 동일한 것으로 본다. 성능분석을 위해 기본적으로 필요한 수식들은 다음과 같다. 트랜잭션은 접근하는 각 데이터 항목에 반드시 읽기연산과 쓰기연산을 수행한다. 비교하려는 두 방법에서 읽기/쓰기연산은 접근하는 데이터 항목이 메모리에 없을 경우 디스크 입출력이 발생한다. 따라서 트랜잭션에서 수행되는 읽기/쓰기연산시간의 합은 RW_T 이라 할 때 다음과 같다.

$$RW_T = n \times \{(1 - P_{\mu}) \times io_time + 2 \times T_{cpu_time}\}$$

TCML-EC 방법에서 트랜잭션에 수행되는 검사연산과 할당연산의 수행시간의 합은 다음과 같다.

$$\begin{aligned} & \text{트랜잭션에서 수행되는 검사연산 시간의 합} \\ & = n \times T_{cpu_time} \end{aligned}$$

$$\begin{aligned} & \text{트랜잭션에서 수행되는 할당연산 시간의 합} \\ & = n \times T_{cpu_time} \end{aligned}$$

5.1 트랜잭션의 대기시간의 분석

제안한 CCL-EC 방법에서는 잠금을 이용해서 일관성을 유지하였는데, 기존 2단계 잠금을 기본으로 한 TCML-EC 방법은 데이터 항목에 검사 잠금과 할당 잠금을 추가하여 트랜잭션을 관리하였다. 트랜잭션 대기 시간은 임의의 데이터 항목에 잠금이 설정되었을 때, 설정된 잠금과 충돌관계인 잠금을 요구하는 트랜잭션이 설정된 잠금이 해제되기를 기다리는 시간을 의미한다. 본 절에서는 [6]과 CCL-EC에서 트랜잭션의 대기의 주 요인인 잠금 보유시간의 평균을 계산해서 트랜잭션의 대기시간을 분석한다.

다음은 쓰기잠금이 존재하는 데이터 항목 x에 대하여 충돌이 발생하였을 때, 대기하는 트랜잭션이 오직 하나인 경우이다. TCML-EC 방법에서 트랜잭션의 연산처리시간은 검사연산과 할당연산 수행시간 만큼 많은 시간을 필요로 하므로 다음과 같다.

$$\begin{aligned} T_{processing_time} & = n \times T_r + (T_c + T_a) + T_w \\ & = n \times \{(1 - P_{\mu}) \times io_time + T_{move}\} + (2n+1) \\ & \quad \times T_{cpu_time} \end{aligned}$$

CCL-EC 방법에서 트랜잭션의 연산처리시간은 평균 상품의 데이터의 수량에 읽기/쓰기연산의 시간 만큼

필요하므로 다음과 같다.

$$T_{\text{processing_time}} = RW_T$$

따라서 TCML-EC에서는 트랜잭션 T_i 이 쓰기 잠금을 소유한 데이터 항목에 대하여, 트랜잭션 T_j 가 접근하는 데이터 항목 중 k 번째로 접근하는 경우, 트랜잭션 T_j 가 k 번째로 접근하는 데이터 항목으로 인하여 대기하는 시간은 다음과 같다.

$$\begin{aligned} T_j(k)_{\text{wait_time}} &= T_{i_start_time} + T_{i_processing_time} \\ &\quad - (T_{j_start_time} + T_{j(k-1)_{\text{processing_time}}} + T_{\text{move}}) \\ &= T_{i_start_time} - T_{j_start_time} + (n-k+1) \\ &\quad \times \{(1-P_\mu) \times io_time + T_{\text{move}} + T_{\text{cpu_time}}\} - T_{\text{move}} + T_{\text{cpu_time}} \end{aligned}$$

즉, 트랜잭션 T_j 가 접근하는 데이터 항목 중에서 k 번째 데이터 항목에 접근하였을 때, T_j 는 트랜잭션 T_i 가 앞으로 접근해야 할 데이터 항목을 위해 이동하고, 읽기 잠금과 쓰기 잠금을 얻고 해제할 때까지의 모든 과정동안 대기하여야만 한다.

반면 CCL-EC방법에서는 트랜잭션 T_i 가 쓰기 잠금을 소유한 데이터 항목에 대하여, 트랜잭션 T_j 가 k 번째로 접근하는 경우, 트랜잭션 T_j 가 k 번째 데이터 항목으로 인하여 대기하는 시간은 다음과 같다.

$$T_j(k)_{\text{wait_time}} = T_{\text{cpu_time}}$$

즉 CCL-EC방법에서는 트랜잭션이 접근하는 모든 데이터 항목에 대해서 동시에 잠금을 요구하고 쓰기 연산 또한 동시에 수행되므로, 트랜잭션 T_i 은 쓰기연산만 잠금을 보유한다. 따라서 T_j 는 T_i 의 쓰기연산 시간 동안 대기한다. 그러므로 임의의 트랜잭션이 쓰기연산을 소유한 데이터항목에 대하여 단지 하나의 트랜잭션만이 k 번째 데이터 항목을 접근하면서 충돌하는 경우 TCML-EC방법이 CCL-EC방법보다 $T_{i_start_time} - T_{j_start_time} + (n-k+1) \times \{(1-P_\mu) \times io_time + T_{\text{move}} + T_{\text{cpu_time}}\} - T_{\text{move}}$ 시간만큼 더 대기한다.

다음은 동시에 많은 트랜잭션들이 수행되어 쓰기잠금이 존재하는 데이터 항목에 충돌이 발생하였을 때, 쓰기잠금이 존재하는 데이터 항목에는 이미 많은 트랜잭션들이 대기하고 있으며 대기하는 트랜잭션들은 여러 데이터 항목에서 충돌하여 대기하는 경우이다.

쓰기 잠금이 존재하는 데이터 항목 x 에 대하여, $m-1$

개의 트랜잭션이 대기하고, 각 트랜잭션은 1개의 데이터항목과 충돌하여 대기하는 경우, m 번째 트랜잭션이 데이터 항목 x 에 대하여 잠금을 얻기 위하여 대기하는 시간은 다음과 같이 나타낼 수 있다.

$$T_m(k)_{\text{wait_time}} = \sum_{j=1}^{m-1} \sum_{k=1}^l T_j(k)_{\text{wait_time}}$$

임의의 트랜잭션이 쓰기잠금을 소유한 데이터 항목에 대하여 단지 하나의 트랜잭션만이 k 번째 데이터 항목을 수행하면서 충돌하는 경우 트랜잭션의 대기시간 $T_j(k)_{\text{wait_time}}$ 가 CCL-EC 방법보다 TCML-EC방법이 더 크므로 $T_j(k)_{\text{wait_time}}$ 의 합으로 나타나는 쓰기 잠금이 존재하는 데이터 항목 X 에 대한 임의의 트랜잭션의 대기시간 또한 CCL-EC 방법보다 TCML-EC방법이 더 크다.

5.2 거래율에 따른 분석

임의의 시간동안 상품이 거래되는 거래율은 앞 절의 분석에서와 같이 트랜잭션의 연산충돌로 인해 대기시간이 적을수록 거래율은 높아질 것이다. 그러나 사용자가 장바구니에 할당받은 상품을 다른 사용자가 사지 못하게 가지고 있다가 취소를 한다면 취소하는 물 만큼 거래율이 떨어지게 된다. 따라서 임의의 시간 T_{time} 동안 상품의 평균 취소율이 $k\%$ 라면 TCML-EC 방법에서는 $T \times k\%$ 개의 트랜잭션이 철회된다. CCL-EC방법에서는 판매가능한 수량을 $k\%$ 만큼 더 추가하여 사용자의 장바구니에 할당하여 트랜잭션을 처리하므로 더 거래율을 향상시킬 수 있다. 그러나 $k\%$ 의 취소율이 발생하지 않는 경우는 TCML-EC방법과 CCL-EC의 거래율은 동일하나, 취소율은 CCL-EC방법이 $k\%$ 만큼 더 발생한다.

6. 결 론

지금까지 전자상거래에 관한 연구는 쇼핑물에 관한 연구, 인증에 관한 연구, 대금결제 서비스에 관한 연구, 에이전트를 기반으로 한 쇼핑물시스템에 관한 연구, 트랜잭션 처리에 관한 연구 등이 주로 이루어 졌다. 특히 사용자와 쇼핑물 사이에 보다 안전한 대금결제를 하기 위해서 전자지불시스템을 구축하는데 많은 초점을 두고 있다. 최근에는 사용자가 좀더 쉽고 효율적으로 상품을 구매하기 위해서 사용자와 공급자의 일일

대신하는 에이전트기술을 접목시킨 쇼핑물시스템이 출현되고 있다.

대부분 기존 쇼핑물에서 상품판매는 상품의 수량에 제한을 두지 않고 사용자가 무한대로 거래가 이루어지고 있으나 전자상거래를 이용하는 사용자의 수가 증가하고 상품의 특성과 성질에 따라 여러 개의 상품이 하나의 상품을 구성하는 패키지 상품도 제공을 해주어야 한다. 그러나 공급자가 제공하는 물량이 한정적이므로 상품수량을 고려한 트랜잭션 관리를 해야 한다. 상품 수량을 고려함에 따라 쇼핑물에서는 보다 더 사용자들에게 신뢰와 믿음을 줄 수 있는 거래를 유지한다.

쇼핑물에서 판매하는 패키지 상품의 수량이 제약성을 갖는 경우 패키지를 구성하는 요소상품이 전부 만족해야지 상품의 가치를 발휘할 수 있으므로 단위물과 통합물사이에 데이터가 일관되게 처리되어야 한다. 따라서 에이전트를 기반으로 한 한정된 패키지 상품을 판매하는 전자상거래 시스템을 구축할 때 상품 수량의 일관성을 유지하기 위한 트랜잭션 처리 기술이 필요하다.

기존 분산 데이터베이스를 관리하는 기법중 하나인 2단계 잠금 규약을 그대로 인터넷을 기반으로 한 쇼핑물시스템에 적용하면 많은 트랜잭션들이 배타적 잠금으로 인하여 지연됨으로써 동시성이 저하됨에 따라 거래되는 시간이 길어진다.

따라서 본 논문에서는 수량이 한정된 고정형 패키지 와 맞춤형 패키지 상품을 판매하는 전자상거래 시스템에서 각 상품의 수량 정보를 실수량과 판매가능한 수량값을 유지하여 쓰기 잠금으로 인하여 지연된 트랜잭션을 수행 가능토록 하고, 불특정 다수의 사용자들이 언제든지 쇼핑물에 접근하여 상거래를 행 할 수 있는 트랜잭션의 관리를 위해서 장바구니 부트랜잭션과 구매 부트랜잭션으로 나누어 수행할 수 있도록 하였다. 또한 이 두 트랜잭션들이 접근하는 수량의 항목을 따로 관리하여 수행함에 따라 쓰기 잠금으로 인해 발생하는 트랜잭션의 대기 시간을 줄일 수 있었고, 이 두 항목이 가지는 수량의 값이 일관된 값을 유지할 수 있도록 새로운 잠금을 사용하여 동시성 제어 알고리즘인 CCL-EC 알고리즘을 제안하여 CCL-EC 알고리즘에 의해 기존 2단계 잠금 규약에서 발생하는 트랜잭션 지연으로 인하여 발생하는 동시성의 저하 문제를 해결하였다. 그리고 사용자에게 상품의 구매 여부를 실시간으로 처리하도록 하였다. 앞으로의 연구방향은 본 논문에서 제안한 트랜잭션 관리 기법의 구현과 타입스택 프를 이용한 트랜잭션 관리 방법에 관한 연구가 이루어

어져야 할 것이며, 또한 동시에 만족하지 않는 트랜잭션들의 거래가 이루어질 수 있도록 더 효율적인 전략 기법에 관한 연구가 이루어져야 할 것이다.

참 고 문 헌

- [1] A. Chavez, D. Dreilinger, R. Guttman, and P. Maes, "A Real-Life Experiment in Creating an Agent Market-place," To appear in Proceedings of the Second International Conference on the Practical Application of Intelligence Agents and Multi-Agent of Intelligence Agents and Multi-Agent (PAAM'97) London, UK, April 1997.A.
- [2] J. G. Lee, E. S. Lee, "VEMA : Multi-Agent System for Electronic Commerce on Internet," Int'l Conf. on HCI(HCI'97), Aug. 1997.
- [3] J. G. Lee, E. S. Lee, Young Uk Song, and Jea Won Lee, "A Comparison Shopping Architecture over Multiple Malls : The Meta-Malls Architecture," Processings of International Conference on Electronic '98, April 1998
- [4] M. R. Genesereth and S.P. Ketchpel, Software Agents, Communications of the ACM, Vol.37, No.7, Jul. 1994, pp.48-53
- [5] 김성환, 황부현, "연계된 상품 판매를 위한 트랜잭션 관리", 한국정보과학회 호남 제주지부 학술발표 논문집 Vol.10, No.1, pp.246-250, August 29, 1998
- [6] 김성환, 김대인, 최희영, 황부현, "연계된 상품 판매를 위한 록 기반의 동시성 제어", 한국정보과학회 98 가을 학술발표 논문집 Vol.25, No.2, pp.51-53, October 31, 1998
- [7] J. J. Jung, D. Y. hwang, S. B. Jeon, G. S. Jo, "Agent-Based Framework for Brokerage Between Buyers and Sellers on Electronic Commerce," International conference on Electronic Commerce (ICEC'98), April 6-9, 1998, Seoul, Korea
- [8] Philp A. Bernstein, Vassos Hadzilacos, Nathan Goodman, "Concurrency Control and Recovery in Database Systems," Addison Wesley 1987.
- [9] Steven Ketchpel and Hector GarciaMolina. "Making trust explicit in distributed commerce transactions," In Proceedings of the 16th International Conference on Distributed Computing Systems, pp.270-281, 1996.
- [10] 손수호, "이동 에이전트를 이용한 위탁 판매에 관한 연구", HCI' 98 학술 대회발표 논문집, 1998.
- [11] D. Lange and D. Chang, IBM Aglet Workbench, IBM

Coporation White Paper, 1996, available at <http://www.tri.ibm.co.jp/aglets>

- [12] 김평철, "웹을 이용한 클라이언트-서버 데이터베이스 응용 개발", 한국정보과학회 '97동계데이터베이스 학술대회 튜토리얼, 1997
- [13] Jea Kyu Lee, Young Uk Song, Jae Won Lee, "A Comparison Shopping Architecture over Multiple Malls : The Meta Malls Architecture," International conference on Electronic Commerce '98, April 6-9, 1998, Seoul, Korea. pp.149-154, 1998.
- [14] <http://www.metaland.com>
- [15] <http://www.hyundaidept.com>



최희영

e-mail : hychoi@sunny.chonnam.ac.kr
 1987년 목포대학교 전산통계학과 (학사)
 1989년 전남대학교 대학원 전산통계학과(이학석사)
 1999년 전남대학교 대학원 전산통계학과(박사과정 수료)

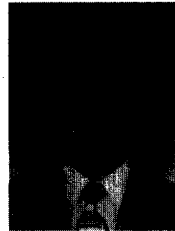
1997년~현재 전남대학교 전산학과 시간강사
 관심분야 : 분산시스템, 전자상거래, 트랜잭션 관리, 실시간 시스템, 에이전트



황부현

e-mail : bhhwang@chonnam.chonnam.ac.kr
 1978년 숭실대학교 전산학과 (학사)
 1980년 한국과학기술원 전산학과 (공학석사)
 1994년 한국과학기술원 전산학과 (공학박사)

1980년~현재 전남대학교 전산학과 교수
 관심분야 : 분산시스템, 분산 데이터베이스 보안, 객체지향 시스템, 전자상거래



허기택

e-mail : gthur@blue.dongshinu.ac.kr
 1984년 전남대학교 계산통계학과 (학사)
 1986년 전남대학교 대학원 계산통계학과(이학석사)
 1989년 광운대학교 대학원 전자계산학과(이학박사)

1989년~현재 동신대학교 컴퓨터학과 교수
 관심분야 : 멀티미디어, 정보통신, 전자상거래