제18회 한국정보처리학회 추계학술발표대회 논문집 제9권 제2호 (2002.11)

분산환경에서 데이터 동기화를 위한 이동 에이전트 시스템

국윤규*, 전병국**, 최영근*
*광운대학교 컴퓨터과학과
**원주대학 사무자동학과
e-mail : ykkook@cs.kwangwoon.ac.kr

Mobile Agent System for Data Synchronization in Distributed Environment

Youn-Gyou Kook*, Byung-Guk Chun**, Young-Keun Choi*
*Dept of Computer Science, Kwang-Woon University
**Dept of Office Automation, Won-Ju College

요약

컴퓨터와 정보기술 발달은 독립적인 e-비즈니스 솔루션들을 대거 등장하게 하고, 그로 인하여 분산 환경내 이종 시스템간의 원활한 정보의 공유와 활용을 위하여 동기화 필요성이 대두되고 있다. 이러한 독립된 플랫폼 상의 e-비즈니스 솔루션들간의 정보 공유를 위한 데이터 동기화는 중요한 문제이다. 따라서 본 논문에서는 각 e-비즈니스 프로세스들의 상호 운용성 보장과 분산처리 시스템 환경의 클라이언트/서버 구조의 문제를 해결할 수 있는 이동에이전트 시스템을 이용한 데이터 동기화를 언급한다.

1. 서론

더욱 복잡한 정보기술 환경으로, 독립적인 e-비즈니스 솔루션들이 원활한 정보 활용과 인터넷 기반 애플리케이션과의 상호 운용 등으로 인한 정보 공유 필요성이 확대되어 감에 따라 e-비즈니스 프로세스의 데이터 동기화가 필요하다.[8]

분산 환경에서 이종 시스템간의 데이터 동기화는 데이터 일관성을 위하여 동기화의 절차와 충돌에 따른 데이터 우선순위를 결정하는 동기화 정책을 설정해야 한다. 또한 동기화 정책을 반영하는 동기화 시스템은 데이터 갱신과 삭제에 따른 자율성 보장과 분산되어 운용되는 플랫폼에 독립적이어야 하며, 네트워크 부하와 트래픽을 감안하여 이동이 원활한 경량 시스템이 되어야 한다. 이러한 시스템에 적합한 이동 에이전트 시스템은 플랫폼에 독립적이고, 이종 데이터를 동기화 시킬 수 있으며, 특성상 분산처리 시스템의 대안으로 인식되고 있다. 또한 이동 에이전트는 네트워크 부하와 트래픽을 감소시키며 원격지에서 수행할 실행코드와 데이터를 한 번에 이동하여 로컬 시스템에서 업무를 수행하므로 분산 시스템에 응용이 가능하다.[1][5][10][11]

※ 이 논문은 한국과학재단 2001년도 산학협력연구지원에 의하여 연구되었음. (I01-2001-000-00064-0)

따라서 본 논문에서는 각 e-비즈니스 프로세스들의 상호운용성을 보장하고 네트워크의 문제를 고려하면서 데이터 동기화를 할 수 있는 이동 에이전트 시스템에 대하여 기술한다.

이동 에이전트 시스템은 분산된 e-비즈니스 프로세스에 상주하면서 에이전트 시스템간의 통신을 할 수 있도록 한다. 에이전트 시스템은 각 프로세스에서 발생하는 데이터 갱신 정보를 프로세스 내의 데이터베이스에 반영하면서 이와 동시에 동기화가 필요한 프로세스의 에이전트 시스템에게 이에 대한 정보를 전달하여 동기화를 요구한다. 동기화 요구를 받은 에이전트 시스템은 전달받은 정보를 바탕으로 데이터의 정보를 갱신한다.

본 논문의 이동 에이전트 시스템은 코드와 지속적인 상태정보로 이루어지는 이동 에이전트와 e- 비즈니스 프로세스가 존재하는 호스트 내에서 에이전트를 생성, 실행, 전송, 종료시킬 수 있는 에이전트 서버로 구성된다. 에이전트는 목적지 에이전트 서버의 이름과 주소를 식별하여 이동하며, 다른 에이전트와 에이전트 소유자, 에이전트 관리자 등과 통신을 한다. 본 논문은 2장에서 이동 에이전트와 동기화 정책을 살펴보고, 3장에서 동기화 이동 에이전트 시스템을 설명하며, 마지막으로 결론으로 구성된다.

2. 관련 연구

2.1 이동 에이전트

이동 에이전트 시스템은 '네트워크로 연결된 독립성이 높은 다수의 독립적인 소프트웨어 프로세서(에이전트)가 대등한 관계를 유지하며 하나의 시스템으로서의 기능을 수행하는 시스템'으로 정의할 수 있다.[6]

이동 에이전트는 주어진 작업을 완수하기 위하여 다른 서버에 파견되는 프로세스이다. 파견된 이동 에이전트는 소스 서버와 독립적이며 자율적으로 진행한다. 이동 에이전트가 목적지 에이전트 서버에 도착하면, 에이전트의 자격을 인증한다. 인증을 부여받은 에이전트는 실행을 시작하게 된다. 이동 에이전트는 새로운 에이전트를 생성하여 다른 에이전트 서버에 전송할 수 있거나, 다른 에이전트와 상호작용할 수 있다. 작업 완료한 이동 에이전트는 파견한 서버나 다른 에이전트 서버에 그 결과를 전달한다.[9] 그림 1은 이동 에이전트의 특성을 표현한 것이다.

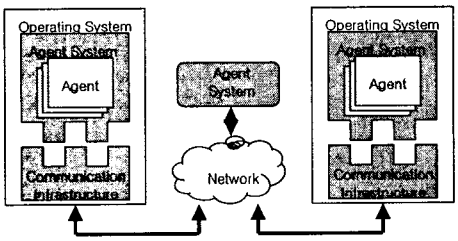

그림 2 이동 에이전트의 흐름도

따라서 이러한 이동 에이전트 시스템은 클라이언트/서버 시스템의 결점을 해결할 수 있는 분산처리 시스템의 대안으로서 인식되고 있다. 이동 에이전트 시스템은 플랫폼에 독립적이므로 다양한 이종 시스템에 적용될 수 있으며, 이동 에이전트가 상황에 따라 동적으로 변화가 가능하므로 유연성이 뛰어나다.

이동 에이전트 시스템은 Aglets, Ara, Concordia, Mole, Odyssey 등 많은 시스템이 개발되고 있고 현재 분산처리 시스템에 직접 적용하기 위해 많은 연구가 진행되고 있다. 또한 이동 에이전트를 이용한 전자 상거래를 비롯하여 네트워크 관리, 정보검색, 정보보안, 시스템 통합 등 많은 분야에서 이동 에이전트를 적용한 연구가 활발히 진행되고 있다.[3]

2.2 동기화 정책

분산 환경에서 이종 시스템간의 데이터 동기화는 데이터 동기화 절차를 통해서 데이터 일관성을 유지한다. 데이터가 갱신된 시스템은 동기화가 필요한 시스템들에게 해당 데이터의 갱신을 요구할 때, 먼저 갱신된 시스템의 데이터 트랜잭션 처리를 할 것인지 아니면 다른 시스템들에게 먼저 데이터 트랜잭션 처리를 요구할 것인지, 동기화 절차에 대한 정책이 필요하다. 또한 데이터를 삭제한 시스템에서 동기화를 요구할 때, 동기화가 필요한 다른 시스템은 해당 데이터의 정보를 갱신한다면 두 시스템간에 데이터가 불일치 되므로 이에 대한 해결 방법이 요구된다.

그리고 데이터 동기화 트랜잭션 처리에 있어서 동기화 요구 시스템간의 충돌 발생 여부를 감지해야 한다. 두 시스템이 동일한 튜플에 서로 다른 데이터 값을 삽입하는 삽입 충돌, 한 시스템은 튜플을 삭제하고 다른 시스템은 튜플을 갱신하는 삭제 충돌, 두 시스템이 동일한 데이터 필드에 서로 다른 데이터 값을 갱신하는 갱신 충돌이 있다.[13] 두 시스템간의 동기화로 인한 상호간의 충돌을 탐지하고 해결 해야한다.

그러므로 분산 환경에서 이종 시스템간의 데이터 동기화 시스템은 데이터 일관성을 위하여 동기화의 절차와 충돌에 따른 데이터 우선순위를 결정하는 동기화 정책을 설정해야 하고, 데이터 갱신과 삭제에 따른 자율성을 확보해야 한다. 또한 분산 환경에서 이종 시스템간의 데이터 동기화를 위하여 동기화 시스템은 플랫폼에 독립적으로 운용되어야 하며, 이동이 원활하도록 최적화를 통해 경량 시스템이 되어야 한다.

3. 동기화 이동 에이전트 시스템 (Synchronization - Mobile Agent System)

3.1 시스템 구성요소

동기화 이동 에이전트 시스템(S-MAS)의 특징은 플랫폼에 독립적이고, 이동이 원활하며, 자율적으로 동기화를 진행한다. 동기화 이동 에이전트 시스템(S-MAS)은 그림 2에서와 같이 시스템 관리, 정책 조정 및 관리, 에이전트 관리 부분으로 구성된다.

3.1.1 시스템 관리 모듈

시스템 관리에서는 동기화에 필요한 정책들을 설정한다. 동기화에 필요한 정책은 데이터베이스에 접근하기 위한 JDBC와 동기화하기 위한 테이블의 정보, 또한 에이전트 시스템에서 생성된 동기화 이동 에이전트의 이주 정책과 시스템간의 데이터 일관성을 위한 우선순위를 설정해야 한다.

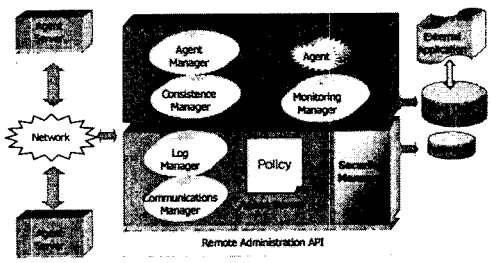

그림 3 동기화 이동 에이전트 시스템 (Synchronization-MAS)

시스템 관리자는 이동 에이전트 시스템이 Java로 이루어져 있기 때문에 데이터베이스에 접근하기 위한 JDBC(Java Database Connectivity)를 설정해야 한다. JDBC는 관계형 데이터베이스에 접근하고 조작하는 Java 표준 명세로서 사용되는 데이터베이스의 제공업체에 따라 JDBC를 설정해야 하며 데이터베이스간의 동기화가 이루어질 테이블의 정보를 명시해야 하고, 동기화를 위한 시스템간의 이동 에이전트 이주 정책을 설정해야 한다. 이동 에이전트의 이주 방법은 동기화가 요구되는 시스템들에게 동시 전송을 하는 브로드캐스팅 이주와 여정에 따라 전송되는 유니캐스팅 이주가 있다. 마지막으로 데이터 일관성을 위하여 충돌에 따른 데이터 우선순위 정책 설정이 필요하다. 시스템간의 데이터 삽입 충돌, 삭제 충돌, 갱신 충돌에 대하여 트랜잭션 시간 기반 우선 순위와 시스템 중요도에 따른 우선 순위를 설정할 수 있다. 이동 에이전트의 전송에 따른 보안관련 정책 설정은 본 논문의 연구 방향의 일환으로서 향후 과제로서 계속적인 연구가 필요하다.

3.1.2 정책 조정 및 관리 모듈

시스템 관리에서 설정된 정책들은 동기화가 이루어질 시스템들의 통신 관리자(Communication Manager)를 통해 조정되고, 에이전트 시스템에서 발생하는 모든 이벤트는 로그 관리자(Log Manager)에 의해 기록된다.

통신 관리자는 동기화가 필요한 에이전트 시스템들과의 동기화 정책을 조정한다. 데이터베이스 관련하여 시스템에 설치된 JDBC 정보와 동기화가 필요한 테이블의 정보를 전송하며 각 시스템의 시간 정보(Time Stamp)를 전송한다. 또한 이동 에이전트들의 이동 경로인 여정(Itinerary)과 데이터 일관성을 위한 우선 순위에 대한 에이전트 시스템간의 조정이 필요하다.

로그 관리자는 동기화 이동 에이전트 시스템에서 발생하는 이벤트에 대한 로그를 관리한다. 이동 에이전트의 접근과 인증, 데이터베이스 갱신에 따른 에이전트 생성 및 파견 등의 로그를 기록한다.

보안 관리자는 원격지에서 파견된 이동 에이전트의 인증을 통해 시스템의 접근을 허용한다. 만일 파견된 이동 에이전트마다 시스템 접근에 따른 다른 등급을 주기 위해서는 이동 에이전트에 대한 보안 및 인증 부분이 설계되어야 하지만, 본 논문에서는 기본적인 보안 문제 해결만을 위해 Java에서 제공되는 보안관리자를 구성한다.

3.1.3 에이전트 관리 모듈

데이터 동기화에 관련된 에이전트들을 관리하는 부분으로서 설정된 정책에 따른 에이전트의 관리, 데이터 갱신과 삭제에 따른 모니터 관리, 동기화 에이전트의 생성 및 파견 관리, 원격지에서 파견된 동기화 에이전트의 인증 관리, 동기화를 위한 데이터 일관성 관리 등의 역할을 수행한다.

에이전트 관리자(Agent Manager)는 시스템 관리 부분에서 설정된 정책에 따라 에이전트를 관리하고, 원격지에서 파견된 이동 에이전트의 인증을 담당한다. 데이터를 모니터링하면서 갱신과 삭제에 따른 이벤트가 발생하면 이동 에이전트를 생성하여 정책에 따라 파견한다. 또한 원격지에서 파견된 이동 에이전트의 시스템 접근에 대한 인증을 위하여 보안 관리자와의 통신을 통하여 인증 처리를 하게된다.

감시 에이전트(Monitoring Manager)는 데이터의 갱신과 삭제에 따른 이벤트 발생의 경우에 에이전트 관리자에게 이벤트 발생에 대한 정보를 전달한다.

일관성 에이전트(Consistence Manager)는 파견된 이동 에이전트와의 통신을 통한 데이터 일관성 여부를 판단하여 데이터베이스 접근 여부를 결정하여 에이전트 관리자에게 결과를 전달한다. 일관성 에이전트는 시스템의 우선 순위 정책에 따라 데이터 충돌을 피하기 위한 판단을 하여 에이전트 관리자가 파견된 에이전트의 시스템 접근여부에 대한 결정을 한다.

동기화 에이전트(Sync Agent)는 데이터 갱신 및 삭제에 따른 이벤트 발생으로 모니터링 에이전트가 정보를 전달하면 에이전트 관리자에 의해 생성되어 원격지로 파견된다. 동기화 에이전트는 이주 정책에 따른 여정에 의하여 이동을 하며 데이터 충돌을 방지하기 위한 우선 순위 정책을 따른다. 원격지 에이전트 시스템에서 인증을 받고 실행한 이동 에이전트는 그 결과 값을 획득하여 여정에 따라 다음 호스트로 이동한다.

3.2 시스템 설계

동기화를 위한 이동 에이전트 시스템은 데이터 갱신 및 삭제에 따른 이동 에이전트 생성과 파견을 위한 이주 메커니즘과 원격지 이동 에이전트 시스템으로의 이동 후 인증과 실행, 데이터 충돌 방지 등을 위한 실행 메커니즘으로 구분할 수 있다.

그림 3은 데이터 갱신 및 삭제에 따른 이동 에이전트의 생성 및 실행, 상태에 대한 결과를 순차적으로 도식화한 것이다.

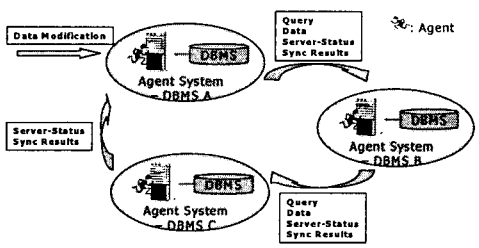

그림 4 이동 에이전트 실행 순차도

3.2.1 이동 에이전트 이주 메커니즘

외부 프로세스에 의한 데이터 변경을 감시한 모니터링 관리자는 에이전트 관리자에게 동기화를 요구한다. 에이전트 관리자는 동기화 정책에 따라 동기화 대상 에이전트 시스템의 보안 관리자에게 접근 허용을 인증 받고, 통신 관리자의 정책 전송과 동기화 이동 에이전트를 목적지에 파견한다.

동기화 이동 에이전트는 데이터 변경 및 상태 정보 등과 함께 이주한다. 이주에 따른 경로 조정 및 보장은 동기화 정책에 따라 이루어진다. 그림 4는 이러한 이동 에이전트 이주 메커니즘을 보여준다.

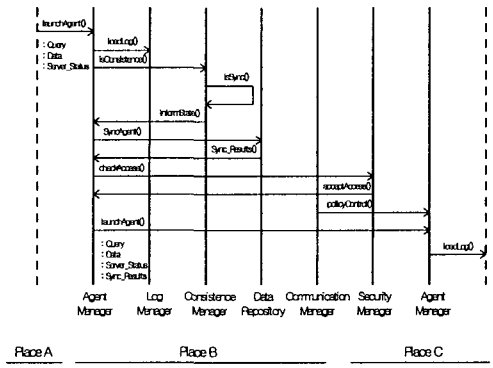

그림 5 이동 에이전트 동기화 메커니즘

3.2.2 이동 에이전트 동기화 메커니즘

동기화 요구를 받은 목적지 에이전트 관리자는 데이터 동기화를 위한 데이터 일관성에 대한 여부를 일관성 관리자에게 요구하여 동기화 정책에 따른 결과 상태에 따라 동기화 에이전트의 실행 여부를 판단한다. 동기화 에이전트에 의해 동기화를 진행한 결과는 에이전트 관리자에게 전달되어 다른 목적지로 파견시에 이 결과를 함께 이주시킨다. 그림 5는 이동 에이전트 동기화 메커니즘을 보여준다.

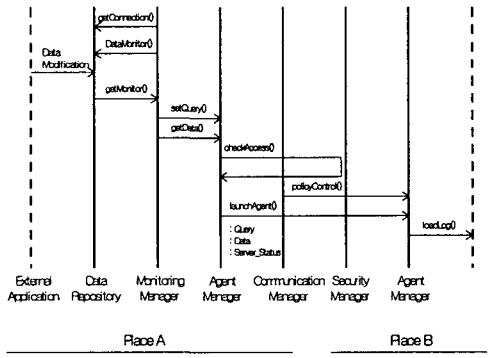

그림 6 이동 에이전트 이주 메커니즘

4. 결론

분산된 e-비즈니스 솔루션들의 정보 공유 필요성이 확대되어 감에 따라 e-비즈니스 프로세스의 데이터 동기화에 대한 연구가 활발히 진행되고 있다. 분산 환경에서 이종 시스템간의 데이터 동기화는 데이터 일관성을 위하여 동기화의 절차와 충돌에 따른 데이터 우선순위를 결정하는 동기화 정책을 설정해야 하고, 동기화 정책을 반영하는 동기화 시스템은 데이터 갱신과 삭제에 따른 자율성 보장과 분산되어 운용되는 플랫폼에 독립적이어야 하며, 네트워크 부하와 트래픽을 감안하여 이동이 원활한 경량 시스템이 되어야 한다. 따라서 분산처리 시스템의 대안으로 인식되고 있는 이동 에이전트 시스템을 이용하여 분산된 이종 데이터의 동기화에 대한 정책과 내용을 살펴보았다.

향후 연구 과제는 이러한 데이터 동기화 시스템을 바탕으로 이동 에이전트 시스템을 이용한 데이터 통합에 대한 연구가 필요하다. 정보기술 환경이 더욱 복잡해짐에 따라 독립적인 응용 솔루션 및 레거시 시스템들로 인하여 분산된 전사적 데이터 통합(EAI)을 위한 연구가 필요하다.

참고문헌

[1] Balazs Goldschmidt, Zoltan Laszlo, "Mobile Agents in a Distributed Heterogeneous Database System", IEEE, 2002

[2] Danny B. Lange, M. Oshima, "Programming and Deploying Java Mobile Agents with Aglets", Addson Wesley, 1998

[3] Giacomo Cabri, Letizia Leonardi, Franco Zambonilli, "Mobile-Agent Coordination Models for Internet Applications", IEEE, 2000

[4] Isao Kaji, Yongdong Tan, Kinji Mori, "Autonomous Data Synchronization in Heterogeneous Systems to Assure the Transaction", 1999

[5] Informix Software, "Cloudscape White Paper : Cloudscape LUCID Application Synchronization", April, 2001

[6] James E. White, "Mobile Agents White Paper" 1998

[7] J. Baumann et al., "Communication Concepts for Mobile Agent Systems", Pro. 1st Int'l Workshop on Mobile Agents, Germany, April, 1997

[8] Kenny K.F.Lee, "The Five Disciplines of ERP Software Implementation", ICMIT, 2000

[9] Mistsubishi Electric ITA, "Mobile Agent Computing", January, 1998

[10] Stavros Papastavrou, George Samaras, Evaggelia Pitoura, "Mobile Agent for World Wide Web Distributed Database Access", IEEE, 2000

[11] Wang Yan, Law Ken C.K. , "A Mobile Agent based System for Distributed Database Access on the Internet", IEEE, 2000

[13] 이상윤, 박순영, 이미영, 김명준, 이동 DBMS의 데이터 동기화기술 분석 , 데이터베이스연구, 17권 2001. 9