

그룹화를 통한 이동 에이전트 기반 모바일 데이터베이스 관리 시스템

신성욱* 정동원* 이정욱** 김창화*** 백두권*

고려대학교 컴퓨터학과, 건국대학교 컴퓨터과, 강릉대학교 컴퓨터학과

{stnoble, withimp, ljo}@software.korea.ac.kr, kch@kangnung.ac.kr, baik@software.korea.ac.kr,

A Mobile Agent-based Mobile Database Management System by Grouping

SungWook Shin* DongWon Jeong* JeongOog Lee** Changhwa Kim*** Doo-Kwon Baik*

Dept. of Computer Science & Engineering, Korea University*

Dept. of Computer Science, Konkuk University**

Dept. of Computer Science, Kangnung University***

요약

무선 네트워크의 폭발적인 증가와 함께 이동 통신 기기의 발전은 기존의 유선 일변도의 사용방식을 무선 환경으로 급속히 확장, 이전시키고 있다. 무선 환경의 발전에 따라 모바일 데이터베이스 사용의 필요성은 급격히 증대되고 있으며 이에 따른 많은 문제점도 발생하고 있다. 현재의 모바일 데이터베이스는 동기화를 위하여 하나의 동기화 서버가 다수의 모바일 데이터베이스를 관리하는 중앙 집중형 방식으로, 모바일 데이터베이스들간의 동기화와 데이터 집중에 문제가 발생한다. 또한 고장허용에 대한 해결책은 전혀 고려되어 있지 않다. 이러한 문제점을 해결하기 위해서 본 논문에서는 이동 에이전트를 이용한 모바일 데이터베이스 관리 시스템을 제안한다. 이 시스템은 모바일 데이터베이스의 분류-배치를 통하여 네트워크 효율을 높이며 고장허용 관리 기능을 통한 좀더 신뢰성 있는 서비스를 제공한다.

1. 서론

최근 폭발적인 네트워크의 발전은 무선 네트워크 부분에서도 일어나고 있다. 무선 네트워크의 근간이 되는 무선 대역폭의 확장과 함께 무선 장비들의 발전으로 통신환경과 기기에 제한적이던 무선 환경이 대폭 개선되고 있다[6]. 이에 따라서 지금까지 한정적이던 무선 기기의 데이터 처리량이 기하급수적으로 증가하고 있으며 단순한 단말기의 역할에서 하나의 독립적인 데이터 처리를 요구하는 디바이스로 발전하고 있다. 이러한 요구에 따라 무선 기기에서 사용되는 모바일 데이터베이스의 요구는 폭발적으로 증가하고 있으며 현재 여러 제품이 상용화 되었다.

현재의 모바일 데이터베이스는 중앙의 데이터베이스와 각 모바일 데이터베이스 사이에 동기화 서버가 존재하여 전체의 동기화를 이루는 방식을 취하고 있다. 이러한 구조는 동기화 서버에 대한 많은 이동 기기의 잦은 접속으로 인한 과부하가 발생할 수 있으며 모바일 데이터베이스와의 동기화를 위해서 불필요한 자원이 소모된다. 또한 모바일 디바이스간의 동기화를 위해 별도의 동기화 서버가 필요하므로 자원의 낭비는 물론 융통성 있는 모바일 데이터베이스의 운영에 제약을 가지게 된다. 그리고 무선 환경의 특성인 예측할 수 없는 사고상황에 대응할 수 있는 고장허용(fault-tolerance)과 비연결 동작(non-connection operation)에 대한 연구가 미흡한 실정이다.

이러한 문제점을 해결하기 위하여 이동 에이전트를 기반으로 하는 모바일 데이터베이스 관리 시스템을 제안한다. 이 논문의 구성은 다음과 같다. 2장에서는 이 시스템을 구성하는데 사용된 요소 기술에 대해서 기술한다. 3장에서는 시스템에 대한 구조적인 설명과 함께 각 기능에 대하여 설명한다. 마지막으로 4장에서는 연구 성과와 함께 앞으로 추가되어야 할 연구 분야에 대하여 설명한다.

2. 관련 연구

이장에서는 제안하는 관리 시스템의 기본이 되는 요소기술의 연구에 대해서 기술한다. 2.1절에서는 제안하는 관리 시스템의

모바일 데이터베이스의 기본적인 구조에 대한 연구에 대하여 기술하며 2.2절 비연결 계획은 제안한 시스템의 고장허용을 제공하는 기본 구조로서 사용한다.

2.1 데이터베이스에 통합된 에이전트 환경

데이터베이스에 통합된 에이전트 환경은 DBMS 내부에 에이전트가 동작할 수 있는 실행환경[1](agent run-time environment)를 작성하는 것이다. 통합환경에서 에이전트는 DBMS가 가지는 기능들을 자신의 기능처럼 사용할 수 있다. 통합환경이 가져야 할 요소로서는 DBMS, 실행환경, 에이전트를 생성할 수 있는 에이전트 운전자(dispatcher), 관찰자(observer), 그리고 결과를 수집하는 종료선(finish-line)으로 구성된다. 통합 환경은 DBMS의 기본 기능을 에이전트가 공유함으로써 시스템 자원의 소모를 줄이고 직접적인 DBMS 기능의 사용으로 다른 외부 인터페이스를 사용하는 것보다 빠른 데이터 처리가 가능하다. 또한 외부 스키마에 쉽게 적용할 수 있는 기능을 제공한다.

2.2 모바일 데이터베이스의 비연결 계획

각각의 모바일 데이터베이스는 하나의 ad-hoc 네트워크를 구성하여 동작한다. ad-hoc 네트워크는 중앙 집중화 된 관리나 표준화된 지원 서비스의 도움 없이 임시 망을 구성하는 무선 이동 호스트들의 집합으로 백본 호스트나 다른 이동 호스트로의 연결을 제공하기 위한 고정된 제어 장치를 가지고 있지 않다. 분산운영과 다이나믹한 네트워크 형태, 불규칙한 링크용량 그리고 소전력 기기에 사용된다는 특성을 가지고 있다. ad-hoc 네트워크를 제공하면 배치와 재구성이 쉬우며 뛰어난 적응성을 가진 네트워크를 구성할 수 있다.

하나의 그룹에 속한 이동기기의 예측된 부재에 대처하는 방법에는 basic sign-off, check-out, 그리고 relaxed check-out의 세가지 방법[2]이 있다. basic sign-off는 부재중인 멤버를 ad-hoc 네트워크의 다른 멤버들은 손상이 아닌 일시적인 disconnect로 인식하며 부재중 멤버는 자신의 데이터베이스를

읽기 전용 방식으로 사용한다. check-out과 relaxed check-out은 자신이 지정한 (check-out) 데이터베이스의 부분에 대해서는 자신만이 접근할 수 있는 권한을 주는 방식으로 relaxed check-out은 다른 멤버에게 지정된 데이터에 대한 읽기 권한만을 주는 방식이다.

3. 이동 에이전트 기반의 모바일 데이터베이스 관리 시스템

3.1 system 개요

이번 절에서는 제안하는 시스템의 전체적인 구성과 각 부분의 역할에 대해서 기술하며 각 부분에 대한 상세한 설명은 다음절에서 설명한다. 이 시스템은 정적 데이터베이스의 데이터 일부를 사용하는 모바일 데이터베이스들을 대상으로 하며 정적 데이터베이스와 모바일 데이터베이스 사이의 데이터 동기화 (backup)는 주기적으로 발생한다고 가정한다.

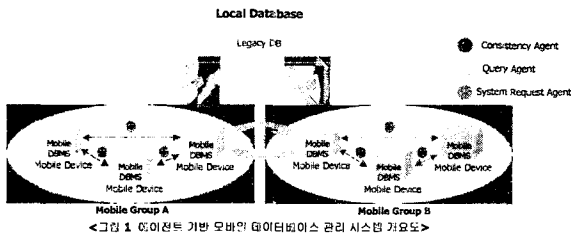


그림 1은 시스템 전체에 대한 개요도이다. 정적 데이터베이스는 단일 혹은 분산 데이터베이스로서 하나의 통합된 정보를 모바일 데이터베이스에게 제공한다. 또한 주기적인 동기화뿐만 아니라 필요에 따라 시스템 요청 에이전트를 이용하여 모바일 그룹간의 동기화를 이룬다. 각 모바일 그룹은 그룹 안에서 동기화를 유지하면서 역시 동기화가 이루어진 다른 그룹과 질의 에이전트를 통하여 필요한 데이터를 얻고 전체 시스템의 동기화를 처리한다.

3.2 모바일 데이터베이스 분류-배치 (mobile database grouping)

모바일 데이터베이스의 성능이 향상되고는 있지만 기존의 정적인 데이터베이스 (static database)와 비교하여 작은 데이터 용량과 처리 능력을 가진 것은 확실하다. 이러한 환경에서 모바일 데이터베이스는 정적 데이터베이스의 선택적인 일부만을 복사하여 사용하게 된다[2]. 이는 각 모바일 데이터베이스가 자신들이 주로 사용하는 데이터의 종류를 구별할 수 있다는 것이다. 이러한 사실에 근거하여 각 모바일 데이터베이스를 각각의 사용에 따라 분류하여 배치한다. 분류된 모바일 데이터베이스는 해당 모바일 그룹 안에 배치되며 그룹별로 데이터의 일관성 유지와 고장허용 정책이 적용된다.

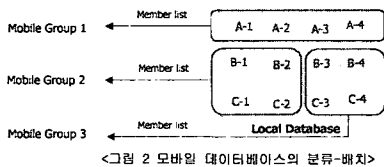


그림 2는 모바일 데이터베이스 분류-배치의 개념을 도식적으로 보여준다. 각각의 모바일 데이터베이스는 정적인 중앙의 데이터베이스의 일부분을 사용한다. 이러한 모바일 데이터베이스의 패턴을 분석하여 정적 데이터베이스를 일정한 그룹으로 분류한다. 모바일 데이터베이스의 참여 빈도 혹은 정적인 중앙의

데이터베이스의 분류에 따라 각 정적 데이터베이스를 분류한다. 이를 기반으로 각 정적 데이터베이스 그룹에 속하는 모바일 데이터베이스를 하나의 모바일 그룹으로 배치한다. 이때 정적 데이터베이스는 각 모바일 데이터베이스에 그룹의 멤버에 대한 정보를 제공한다. 모바일 그룹 안에서 각 모바일 데이터베이스는 ad-hoc 네트워크를 구성하여 개개의 데이터베이스 처리에 대한 데이터를 교환하며 각 모바일 그룹은 그룹 안에서 데이터의 일관성 유지와 고장허용 관리 기능을 제공한다.

모바일 데이터베이스를 분류-배치함으로써 각 모바일 그룹은 자신들이 주로 사용하는 정적 데이터베이스의 한정된 일부분만을 관리할 수 있으며 일관성 유지에 소요되는 부하를 줄일 수가 있다. 또한 각 모바일 그룹 안에서는 데이터의 일관성이 유지되기 때문에 현재 사용하는 모바일 데이터베이스와 비교하여 단순한 모니터 기능이 아닌 데이터베이스의 기능 활용을 극대화 할 수 있다. 추가적으로 중앙 집중적인 동기화 서버를 사용할 필요가 없이 각 모바일 데이터베이스는 동등한 위치를 가지는 ad-hoc 네트워크를 구성함으로써 각 이동 기기에 할당되는 부하의 균등 분할과 함께 고장허용 관리 기능을 제공할 수 있는 기본 구조를 제공한다.

3.3 모바일 그룹에서의 일관성 유지

각 모바일 데이터베이스는 DBMS 기능과 함께 에이전트 실행 환경 (run-time environment)을 제공한다[1]. 각각의 모바일 데이터베이스의 실행환경은 에이전트가 동작하고, 새로운 에이전트를 생성할 수 있는 환경을 제공한다. 모바일 그룹은 그룹 멤버에 대한 정보를 분류-배치과정에서 획득하고 있고 이를 일관성 유지의 기준 정보로 사용한다. 모바일 데이터베이스에서의 일관성 유지는 분산된 데이터베이스가 독립적인 데이터베이스로 작동하는데 가장 기본적인 중요 요소이다[6].

현재의 모바일 데이터베이스는 정적 데이터베이스와 모바일 데이터베이스 사이에 하나의 동기화 서버를 두어 전체 시스템의 동기화를 유지한다. 이러한 구조는 모바일 데이터베이스의 수가 많아질 경우 동기화 서버에 집중된 접속으로 과부하가 발생한다. 또한 시스템 전체의 효율이 떨어질 수 있고 중앙의 동기화 서버의 이상은 전체 시스템의 이상을 야기하는 신뢰적이지 못한 구조이다. 모바일 데이터베이스의 비연결 처리를 위해서는 자신이 가지고 있는 모든 데이터에 대해서 정적 데이터베이스와 동기화 작업을 요구함으로써 시스템의 효율을 극도로 떨어뜨린다. 모바일 데이터베이스간 동기화는 중앙의 동기화 서버를 통해서 이루어진다. 일부 모바일 데이터베이스 업체에서는 모바일 데이터베이스간의 동기화를 위하여 모바일 데이터베이스간 동기화 서버를 제공한다. 이러한 경우 모바일 데이터베이스의 증가에 따라 기하급수적으로 많은 동기화 서버가 필요하게되므로 자원의 낭비와 함께 유동적인 모바일 데이터베이스 운영에 장애가 된다.

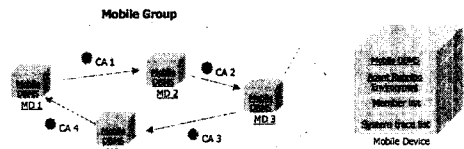


그림 3은 모바일 그룹 안에서 일관성을 유지하는 가장 간단한 예로서 단일 트랜잭션에 대하여 어떠한 손상도 발생하지 않는다고 가정한다. 각 모바일 데이터베이스는 DBMS와 에이전트의 실행환경을 가지며 두 개의 list를 유지한다.

- member list : 그룹에 속한 멤버들에 대한 정보를 가지며 전송시킬 모바일 데이터베이스를 선택하는 데이터 중에 하나로 사용

- System trace list : 정적 데이터베이스에서 분리된 이후부터 모바일 데이터베이스의 트랜잭션 정보를 저장하여 고장허용 관리와 정적 데이터베이스로의 백업에 사용
 그룹 안의 모바일 데이터베이스에서 SQL 질의(update, delete, insert)에 의한 데이터의 변화가 발생하면 해당 질의와 발생 시간, 자신의 정보를 포함한 이동 에이전트를 전송한다. 이동 에이전트를 전송 받은 모바일 데이터베이스는 자신의 데이터를 갱신하고 member list를 기반으로 다른 모바일 데이터베이스로 전송한다. 이러한 단계를 반복하여 그룹의 멤버를 모두 거치면서 일관성을 유지한다. 이 논문에서는 그룹 안에 일관성 유지를 위한 정책과 구현 방법에 대한 세부적인 설명은 하지 않는다.

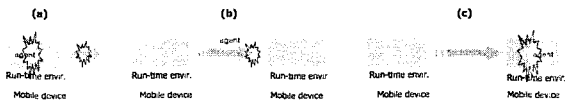
모바일 그룹을 사용한 일관성의 유지는 전체 모바일 데이터베이스를 분할하여 관리함으로써 그룹단위의 처리가 가능하다. 그룹단위의 처리로 하나의 모바일 데이터베이스 또는 하나의 모바일 그룹의 오류가 다른 부분의 처리에는 영향을 미치지 않는 신뢰성 있는 시스템의 구현이 가능하다. 또한 그룹간에 질의 에이전트를 사용하여 자신이 관리하지 않는 데이터에 대한 일관성 있는 처리를 허용한다. 모바일 그룹을 사용한 일관성 유지는 좀더 신뢰성 있고 확장 가능한 융통성 있는 구조를 제공한다.

3.4 고장 허용 (fault-tolerance)

무선 통신 환경의 특성상 물리적 또는 시스템적 요소로 인한 잦은 고장이 발생할 수 있다. 따라서 이러한 상황에 대처할 수 있는 고장허용의 관리 능력은 신뢰성 있는 시스템을 구축하기 위한 가장 중요한 요소 중의 하나이다. 현재의 모바일 데이터베이스 시스템은 고장허용에 대하여 고려를 거의 하고 있지 않다. 모바일 데이터베이스의 고장허용에서 고려되어 할 점은 무선 통신의 특성인 고비용의 데이터 전송, 일관성 유지를 위한 빠른 처리 등이 있다. 이 시스템에서 고장허용을 관리하기 위해서 다음 두 가지 종류의 에이전트를 사용한다.

- (1) 모니터 에이전트 : 모바일 데이터베이스나 이동 기기의 상황을 주기적으로 점검하여 정상 상태로 복귀되었을 때 손상시에 중지되었던 작업을 수행하고 손상 관리 에이전트를 호출한다.
- (2) 손상 관리 에이전트 : trace list를 정상상태의 다른 모바일 데이터베이스와 비교하여 손상상태에서 누락된 정보를 반환함으로써 일관성을 유지한다.

손상의 종류에는 크게 연결 손상 (connection fail)과 기기 손상 (device fail)로 구분할 수 있다[3]. 이를 보다 상세히 분류하면, 연결 손상은 수신 손상, 송신 손상으로 구분되며 기기 손상은 에이전트, 에이전트 실행환경 및 이동 기기의 손상으로 구분된다.



< 표 4 모바일 데이터베이스에서의 fault-tolerance >

모바일 데이터베이스에서의 손상은 그림 4와 같이 세가지 경우로 분류된다.

- (1) 송신 손상 또는 에이전트 실행환경, 에이전트의 손상
 에이전트 기반 모바일 데이터베이스 시스템에서는 락커 페턴을 적용한 체크포인트(checkpoint) 기법[7]을 사용하여 이동 에이전트의 이동 전에 자신의 데이터를 저장한다. 네트워크 상황이나 기타 이유로 인하여 송신이 손상 됐을 경우 네트워크 상황을 파악하는 모니터 에이전트를 생성한다. 모니터 에이전트는 전송하려고 하는 모바일 데이터베이스와 이동 에이전트에 대한 정보를 가지고 주기적으로 전송 상태를 확인한다. 모니터 에이전트의 생성과 함께 시스템은 다른 모바일 데이터베이스를 선택하여 전송을 시도한다. 일정 시간 이상의 송신 손상이 발생

하거나 이동 기기, 에이전트 실행환경의 이상이 발생하였을 경우 시스템은 더 이상 일관성을 유지하기 힘들다고 판단하고 손상 관리 에이전트를 생성한다. 이 에이전트는 해당 모바일 데이터베이스의 변경을 금지시키고 비연결 처리만을 허용한다. 또한 주기적으로 상태를 확인하고 이상 상태가 회복되었을 경우 system trace list를 다른 모바일 에이전트로 전송하여 이상 기간동안 진행된 트랜잭션에 대한 동기화를 이룬다. 동기화가 이루어진 후에 데이터베이스에 대한 제한을 풀고 정상적인 동작을 실행하게 된다.

(2) 수신측의 이동 기기에 손상

수신측의 손상으로 전송이 불가능 할 경우 time-out을 적용하여 일정 시간을 시도한 뒤 응답이 없을 경우 송신측 모바일 데이터베이스는 모니터 에이전트를 생성하여 상대방의 이상 상태를 모니터링하면서 다른 모바일 데이터베이스를 선택하여 전송한다.

(3) 송신측의 에이전트 실행환경이나 혹은 실행환경의 손상
 제한한 시스템에서 모바일 데이터베이스는 점검 단위의 전송이 무사히 이루어졌을 경우 점검 노드에 해당하는 송신 모바일 데이터베이스에 전송 완료 메시지를 송신한다. 이 경우에는 빠른 일관성의 유지를 위하여 exactly-once property[4]를 고려하지 않는다. 일정 시간동안 전송 완료 메시지가 도착하지 않으면 점검 노드는 점검 단위를 줄여서 전송을 실시한다. 점검단위가 1이 되었을 경우 수신 모바일 데이터베이스는 내부 손상이 있다고 인식하고 다른 모바일 데이터베이스로 전송한다. 고장허용 관리 기능은 일관성 유지의 중요한 요소로서 시스템의 신뢰성을 높이며 비연결 처리를 가능하도록 하여 준다.

4. 결론 및 향후 연구

본 연구의 목적은 모바일 데이터베이스를 신뢰성 있고 효율적으로 관리할 수 있는 시스템을 제안하는 것이다. 이 관리 시스템은 분류-배치를 통하여 시스템의 부하를 분산하며 중앙 집중적 동기화 서버의 미사용과 함께 기존의 모바일 데이터베이스 시스템에서 제공하지 않는 고장허용 관리 기능을 제공함으로써 좀더 신뢰성 있는 서비스를 제공한다. 또한 각 모바일 그룹별로 분리된 관리를 실시함으로써 정적 데이터베이스에 대한 데이터 백업을 쉽게 제공하도록 하였다.

향후 연구과제로는 모바일 그룹간, 정적 데이터베이스와 모바일 그룹간의 데이터 교환에 대한 연구가 필요하다.

5. 참고문헌

- [1] E. Weippl and J. Altmann "Mobile Database Agents f Building Data Warehouses" IEEE 11th Internationa Workshop, 2000
- [2] H. JoAnne and A. Divyakant. "Planned Disconnectio for Mobile Database". IEEE 11th international workshop 2000
- [3] P. Stefan and S. Andre. "Modeling Fault-Toleran Mobile Agent Execution as a Sequence of Agreeem Problem". IEEE 19th Symposium, 2000
- [4] J. Gray and A. reuter "Transaction Processing Concept and Techniques" Morgan Kaufman, 1993
- [5] D. Georgakopoulos and M. Rusinkiewicz. " Using Ticke to Enforce the Serialzability of multidatabase Transaction". IEEE , 1993
- [6] E. Pitoura and B. Bhargava. " Maintaining Consistenc of Data in Mobile Distributed Environments". the 15th International conference on Distributin Computing System, 1995
- [7] J. Kumar and M. Satyanarayanan. "Disconnect Operation in the Coda file system". ACM transactions o Computer system, 10, 1992