

All-IP 이동 통신망에서 HSS 시스템의 효과적인 백업과 실시간 이중화 기법

박성진^{1*}, 박형수²

¹한신대학교 컴퓨터공학부, ²(주)엔텔리아

Effective Backup and Real-Time Replication Techniques for HSS System in All-IP Mobile Networks

Seong-Jin Park^{1*} and Hyung-Soo Park²

¹School of Computer Engineering, Hanshin Univ., ²nTelia Inc.

요 약 디지털 이동통신 서비스의 가입자 정보를 실시간으로 관리하는 HSS(Home Subscriber Server) 시스템은 사용자의 콜처리를 위한 기본 정보뿐만 아니라 사용자 인증과 운영 등에 관한 추가적인 정보를 제어하기 위해 가입자 데이터베이스를 주기억장치에 상주시켜 운영하는 주기억 장치 데이터베이스(main memory database system)를 필요로 한다. 그러나, 신뢰성과 안정성이 중요시되는 HSS-DBS의 경우, 초기 로딩시간이 길고 메모리 손실시 그 영향이 훨씬 크기 때문에 보다 안전한 데이터의 저장 방식과 백업 기법이 필요하다. 본 논문에서는 HSS 시스템의 안전성과 성능을 고려한 효율적인 백업 이중화 기법을 제시하였다. 제안한 쉐도잉 백업 기법을 통해 위치 등록으로 인한 실시간 백업의 과부하를 최소화하고 일정 시간 간격을 두고 지연 회복시킴으로써 오류 발생 시 신속하게 복구할 수 있으며 원격 서버에 실시간으로 백업하는 백업 이중화 기법을 통해 안정적인 시스템 운영을 보장한다.

Abstract An HSS(Home Subscriber Server) system requires a main-memory database on main-memory unit for the real-time management of the subscriber information in the mobile communication service, in that the system controls not only basic data for handling calls of users, but also additional service data related to user authentication and operational data.

Nonetheless, HSS-DBS system, requiring the reliability and stability, need more secure data store method and a back-up technique because the system have a long startup time and the big problem on the failures of main-memory.

This paper proposes an efficient back-up replication technique, on the basis of enhancing the stability and performance of HSS system. The proposed shadowing back-up technique adopting the delayed recovery process, can help minimize the real-time back-up overloads by location registration, while the proposed backup replication method enables more stable system operations with replicating the data to remote server in real time.

Key Words : HSS, Main-memory database, HLR, Backup, Replication

1. 서론

이동 통신망에서는 기존의 고정된 유선망과는 다르게 가입자의 위치가 끊임없이 변하므로 위치정보(location information)의 지속적인 관리가 필요하며 HSS(Home

Subscriber Server) 시스템은 이러한 가입자의 위치 정보 관리를 기반으로 하여 호 처리 및 부가서비스 지원 등을 통해 IMT 2000 및 위성 이동통신 분야로 점차 그 활용성이 확대되고 있다. 이를 위하여 디지털 이동통신의 다양한 가입자 정보들의 효율적인 관리를 지원하는 전용 데

이 논문은 한신대학교 학술연구비 지원에 의하여 연구되었음.

*교신저자 : 박성진(sjpark@hs.ac.kr)

접수일 09년 01월 09일

수정일 09년 04월 20일

게재확정일 09년 04월 22일

이터베이스 시스템인 가입자 정보 데이터베이스 시스템(HSS-DBS; HSS DataBase System)의 개발이 필요하다[13,17,18].

디지털 이동통신 시스템의 가입자 정보를 관리하는 HSS-DBS는 주어진 시간적인 제약조건을 만족시켜주어야 함으로 가능한 빠른 접근과 신속한 처리가 이루어져야 한다. 일반적으로 데이터베이스를 디스크에 저장하여 운영하는 것보다 주기억 장치에 운영하는 것이 상대적으로 바람직하다. 이에 따라 빠른 처리를 요구하는 HSS-DBS는 주기억 장치 데이터베이스(main memory database system) 방식이 요구된다[3,5].

주기억장치의 가격 하락과 용량 증대로 인해 이동통신 분야이외에도 다양한 분야에서 주기억장치를 최대로 활용하여 데이터베이스 시스템의 성능을 향상시키는 연구가 활발히 진행되고 있다[1,8,12]. 위치 정보의 변경 및 접근에 대한 실시간 처리와 대용량의 가입자 정보 관리를 목적으로 하는 HSS 시스템에서는 이러한 주기억장치 데이터베이스의 활용이 바람직하다[17]. 따라서, HSS 시스템에서 보다 많은 수의 가입자를 수용하고 정보의 실시간 처리를 위해서는 주기억장치 데이터베이스 시스템 방식을 개선한 HSS-DBS 시스템의 개발이 필요하며 이를 위해서 다음 특성을 고려해야한다.

디스크 기반의 데이터베이스에 비하여 주기억장치 기반의 데이터베이스는 필요로 하는 다량의 데이터를 전부 메모리에 유지하기 때문에 초기 로딩시간이 길고 메모리 손실이 그 영향이 훨씬 크다[6]. 따라서, 신뢰성과 안정성이 중요시되는 HSS-DBS의 경우, 보다 안전한 데이터의 저장 방식과 백업 기법이 필요하다. 즉, 시스템의 오류로 인하여 HSS 시스템에서 관리하는 가입자의 위치 정보가 손실되는 경우 실시간 서비스에 막대한 차질을 유발하고 오류 발생 시 피해 정도와 피해 양이 훨씬 커지며 회복 방안이 마련되어있더라도 상당한 시간과 노력을 통해서 복구될 수 있다는 문제점이 있다. 따라서, 이러한 특성이 HSS-DBS 시스템 개발 시 효율적으로 고려되어 시스템 운영 방식에 반영될 필요가 있다.

본 논문에서는 위와 같은 문제점들을 해결할 수 있도록 HSS 시스템의 안전성과 성능을 고려한 효율적인 백업 이중화 기법을 제시하였다. 백업 기법으로는 HSS 데이터베이스에서 발생하는 트랜잭션의 로그 정보를 활용한 웨도잉 백업 기법을 통해 실시간으로 백업을 수행하는 방법을 제시하였다. 제안 기법은 위치 등록으로 인한 실시간 백업의 부하를 최소화하고 일정 시간 간격을 두고 지연 회복함으로써 오류 발생 시 신속하게 복구할 수 있는 장점이 있다. 또한, 어떠한 상황에서도 데이터의 완전한 복구를 위하여 네트워크를 통한 원격 서버에 실시간으로

백업하는 백업 이중화 기법을 통해 안정적인 시스템 운영을 보장하였다.

2. 백업 방법에 관한 연구

2.1 주기억장치 데이터베이스 시스템에서의 백업

HSS-DBS의 경우, 실시간 접근성을 위하여 주기억장치 데이터베이스 시스템 방식을 도입하여야 한다. 그러나, 이 경우 주기억 데이터베이스 시스템의 휘발성 기억 장치의 특성상 효과적인 백업과 회복 전략이 필요하며 이동통신 서비스의 특성을 고려할 때 신뢰성과 안정성 강화 차원에서 보다 강력하고 효율적인 백업 기법이 필요하다.

주기억장치 데이터베이스 시스템에서 전원 차단이나 운영체제 등의 시스템 오류가 발생할 경우, 주기억장치의 데이터베이스 내용은 휘발성 장치의 특성으로 인해 손실된다. 백업은 이러한 갑작스러운 오류가 발생할 경우에 고장발생 시점 이전으로 시스템을 복구시키기 위한 회복(recovery)을 위하여 수행한다[4,7]. 기존의 디스크 기반의 데이터베이스 시스템에서는 백업 기법으로서 주로 로깅(logging)과 체크포인트팅(check-pointing)을 이용한 회복 기법을 주로 사용한다[2].

로깅이란 시스템이 정상적으로 동작하는 상황에서 트랜잭션의 수행 연산 및 데이터 수정 내용에 관한 상세한 정보를 비휘발성 장치에 기록하는 작업을 말한다. 체크포인트팅이란 오류 복구를 위하여 시스템이 재시동하는 경우 복구 과정이 오랜 시간동안 진행되어야하는 점을 개선하기 위하여 미리 주기적으로 해당 시점의 데이터베이스 정보를 비휘발성 장치에 기록함으로써 신속하게 일관성(consistency)을 유지하는 데이터베이스 상태로 재구성하기 위한 기법이다. 로깅과 체크포인트팅 기법을 이용한 백업이 수행된 주기억장치 데이터베이스의 경우, 오류 발생 후 재시동될때 트랜잭션 일관성을 유지하는 가장 최근의 데이터베이스 상태로 재구성해주는 회복 기능이 요구된다[9].

주기억 시스템에서의 회복 기법의 예로는 MM-DBMS와 Dali 시스템에서의 회복 기법을 들 수 있다. MM-DBMS는 윈스코신 대학에서 개발된 주기억장치 데이터베이스 시스템으로 관계형 데이터베이스 형식으로 별도의 회복 프로세스를 사용하며 로그 버퍼 영역으로서 비휘발성 메모리를 사용하고 있다[11].

Dali 저장시스템은 AT&T 벨연구소에서 개발한 주기억장치 저장시스템으로 디스크 영역을 주기억장치 영역

과 직접적으로 연결하는 기억장치-매핑(memory-mapped)된 구조를 사용한다[8,14]. 이 방법에서는 디스크에 저장, 관리되는 데이터베이스의 구조와 주기억장치에 상주하는 데이터베이스의 구조를 동일하게 통합관리가 가능하다.

초기 HSS-DBS 시스템의 경우, 트랜잭션 로그 등의 개념없이 별도의 독립된 백업 프로세스를 통해 갱신된 메모리 SLD(System Loading Data)의 내용을 직접 페이지 단위로 지속적으로 디스크 SLD로 백업하는 방법을 사용하였다. 그 뒤, 디스크 기반의 일반적인 DBMS 백업 기법인 로깅 기법과 체크포인트 개념 등을 그대로 적용한 HSS-DBS 시스템이 개발되었고 최근 일부 연구의 경우, HSS 시스템안의 트랜잭션 유형별로 일부 백업을 제한하는 등의 차별화된 백업 전략을 적용한 백업 방법을 제안하고 있다[16]. 즉, 이러한 제한적 백업방법은 위치등록의 경우에는 지연백업을, 가입자 등록의 경우에는 실시간 백업함으로써 중요한 가입자 등록 정보는 안전하게 유지관리하는 방법으로 위치등록 정보의 경우에는 손실이 되더라도 오류 발생 전 위치정보를 사용하여 호를 성공시킬 가능성이 높다는 점에 기반한다.

2.2 백업 방법의 문제점

HSS 시스템에서 주요 기능은 사용자의 위치정보 갱신을 위한 위치 등록 기능과 호 처리를 위한 라우팅 정보 요구 기능이 대표적이며 전체 기능의 90% 이상의 비중을 차지한다[16]. 이중에서도 특히, 위치 변경에 따른 가입자 위치정보의 변경은 보다 빈번하게 발생하는 주요 기능이다.

운용자 혹은 고객 관리센터로부터의 요구에 의하여 이루어지는 신규 가입자의 등록 및 기존 가입자 해지, 부가서비스의 등록 및 삭제 등의 가입자 관련 운용 기능은 관련된 정보가 절대로 손실되지 않아야 되는 기능이다. 신규 가입자의 등록과 같은 가입자 관련 운용 기능은 HSS 시스템의 운용자에 의하여 혹은 HSS 시스템과 연결된 고객관리 센터의 요구에 의하여 그 기능이 수행되고 오류로 인하여 HSS 시스템이 재 시동되었을 경우에도 정보가 유지되어야 가입자들에 대한 지속적인 서비스가 가능하다. 이러한 점에서 HSS-DBS 시스템은 시스템의 신속한 데이터 접근을 지원하면서도 높은 수준의 신뢰성과 안정성을 보장해야하는 시스템적인 특성을 가지고 있다.

기존의 주기억 장치를 사용하는 HSS 데이터베이스 시스템의 백업 방법의 문제점은 다음과 같다.

첫째, 주기억장치 데이터베이스 시스템에서 주기억장치에 데이터를 관리하더라도 백업은 기본적으로 디스크에 관련 정보를 안정적으로 저장해야 하기 때문에 디스크 작업에 따른 오버헤드를 줄이기 위한 방법이 요구된

다. HSS 시스템이 재 시동되기 이전의 위치정보를 정상적으로 사용하기 위해서는 각 가입자의 가장 최근 위치정보를 백업하여 사용하는 것이 바람직하다. 그러나, 위치정보에 대해 지속적으로 백업을 수행할 경우에 HSS 시스템의 전체 성능에 매우 심각한 영향을 미치게 된다. 이러한 데이터의 지속적인 실시간 백업이 시스템에 과부하를 주기 때문에 이원화된 형태로 로그의 실시간 백업이 바람직하며 오류 발생 시 백업 로그를 이용한 신속한 복구가 이루어져야하는데 백업 로그의 양이 대량인 경우가 역시 복구 시간이 많이 소요되는 문제점이 있다. 그리고, 이를 해결하기 위해 기존의 체크포인팅 기법을 그대로 적용할 경우 발생 빈도는 감소할지라도 또다시 데이터의 디스크 실시간 백업이 발생하게 된다. 따라서, 백업 로그의 양을 적정 수준으로 관리하기 위하여 실시간으로 데이터가 아닌 로그를 디스크에 저장하고 그림자와 같이 주기억장치 데이터베이스의 또 다른 메모리 버전에 로그 연산을 재실행함으로써 백업 로그 정보를 감소시키는 변형된 체크포인트 방법이 바람직하며 이를 위해 웨도잉 백업 기법을 제안하였다.

둘째, 디스크에 실시간 백업한 백업 로그의 경우에도 미디어 손상 등에 의해서 손실될 수가 있으며 이 경우 시스템의 안정적 운영에 큰 차질이 올 수 있다. 따라서, 하드웨어를 신뢰성 있는 장애허용 컴퓨터를 사용할 뿐 아니라 데이터베이스의 이중화를 통해 백업 로그를 이중화된 원격 데이터베이스에 반영함으로써 미디어 오류에 대비하여야 한다[10]. 이를 위해 백업 이중화 기법을 제안하였으며 이를 통해 여러 오류 발생에 따른 복구과정중에도 원격으로 서비스가 중단없이 이루어지도록 할 수 있다.

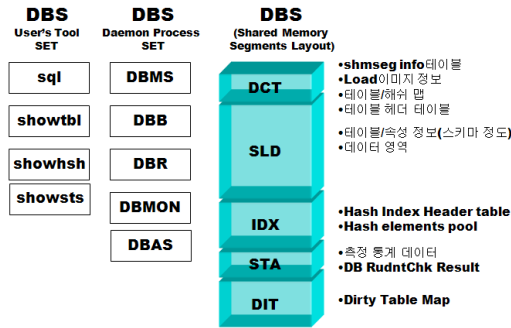
그리고 HSS 시스템이 특정 지역을 기반으로 하기 때문에 등록에 따른 로그의 유형이 비슷하고 동일 사용자의 잦은 위치 등록의 경우, 로그중 최종 위치 정보만을 체크포인트 시에 데이터베이스에 반영하는 것이 바람직하다. 이를 위한 로그 필터링 개념을 적용하였다.

3. HSS 데이터베이스 시스템 구조

3.1 시스템 프로세스 구성

HSS-DBS는 그림 1과 같이 세 부분으로 구성된다. 첫째는 사용자가 DBS의 상태 및 정보를 확인할 수 있는 사용자 툴이다. SQL은 데이터베이스 내 정보를 질의하는 질의 툴이다. showtbl(show tables)은 릴레이션 정보를 확인하는 툴이고 showhsh(show hash index)는 데이터베이

스 인덱스를 구성하는 해쉬 인덱스에 대한 정보를 확인할 수 있다. showsts(show statistics)는 DBMS를 통해 수행되는 데이터 검색, 변경, 삽입, 삭제에 대한 통계 데이터를 출력하는 명령어 실행 틀이다.



[그림 1] HSS-DBS 시스템 구성

두 번째, DBS의 핵심이라 할 수 있는 데몬 프로세스들은 5개의 프로세스로 구성된다. 데이터베이스에 저장되어지는 정보들의 처리와 저장의 효율성은 고도로 최적화되어야 하므로 이를 관리하는 것은 매우 복잡하고 어렵다. 따라서, 데이터 관리를 전문으로 하는 DBMS(Data Base Management System) 프로세스를 두어 사용자 어플리케이션에게서 복잡한 데이터베이스를 독립시키고 정형화된 기호로서 접근하도록 하였다. 또한, DBMS는 데이터베이스 시스템의 구동(Startup)을 수행한다. 구동 로직의 복잡성을 DBMS 프로세스로부터 분리시키기 위해서 `dbloader` 프로그램이 별도로 존재한다. 이 프로그램은 DBMS의 데이터베이스 기동 시 실행되어 디스크에 저장된 데이터베이스를 시스템의 공유 주기억장치에 로딩하도록 한다. 주기억장치 데이터베이스 시스템은 수시로 메모리에 변화가 발생하는데 이에 따른 메모리안의 데이터베이스와 디스크안의 DB파일의 차이가 있을 수 있다. 그래서 `DBR(Database Recovery)` 프로세스는 이들의 일치성(Consistency)을 위해 메모리에 일어나는 변화를 실시간으로 디스크에 쓰기 작업을 수행 한다. 기본 메커니즘은 메모리에 있는 블록에 변화가 생기면 변화된 부분에 플래그를 세팅하여 공유 메모리에 있는 DB를 디스크상의 DB와 일치시킨다. 여기서, 사용되는 플래그는 1byte 당 4096byte로 매핑되어 있다. 데이터베이스를 검색하여 운영자가 그 정보를 알기 쉽게 보여주는 것이 `MMI(Man Machine Interface)`인데 여기에는 특정 국번에 등록되어 있는 가입자를 모두 보여주는 기능이 있다. 이런 벌크 선택(bulk selection) 기능은 DBMS가 아닌 `DBBS(Data Base Bulk Selection)`가 함으로써 DBMS의 부하를 줄일 수 있다.

데이터베이스에 발생할 수 있는 고장에 대비하여 운영자는 주기억장치의 데이터베이스를 `DAT(Direct Access Tape)`로 백업해야 한다. 이러한 백업 관련 작업을 지원하는 프로세스가 `DBB(Data Base Backup)`이다. 데이터의 백업은 DB영역에 대해서만 수행되어지며, 백업된 `DAT`의 복구역시 `DBB`프로세스에 의해서 수행되어 진다. `DBMON(DataBase MONitor)`은 주기억 장치상의 데이터 베이스상태(Data Dictionary, Data Directory, Hash Index 등) 검증을 수행하는 프로세스이다.

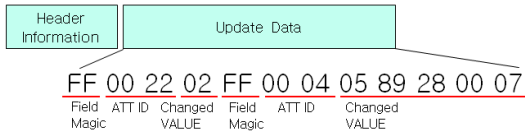
`DBS`가 수행될 때 사용되는 데이터 정보 영역은 매우 중요하다. `DCT(Data Dictionary)` 영역은 주기억장치상 데이터베이스의 물리적 구성 정보가 저장 된다. 데이터베이스가 처음 만들어 질 때 `dbloader`에 의해 생성되어지며, 항상 읽기 전용 모드를 가지는 영역이다. `SLD`는 실제 데이터가 저장되어지는 영역으로서, 이 영역에 대한 변경 사항은 `DBR`에 의해서 실시간으로 디스크상의 `SLD` 파일에 반영되어 진다. `IDX(Index)`는 데이터 접근 속도 향상을 위한 키 값들의 색인 영역이다. 색인은 선형 해쉬(Linear Hash)방식을 지원한다. 이 영역에는 해쉬 구성 정보와 `SLD`안의 데이터들에 대한 해쉬 색인 정보들이 저장되어 진다. `DIT(Dirty Flag)` 영역은 `SLD`상의 변경된 페이지에 대한 더티 플래그 값 저장 영역이다. `SLD`에 변경 사항이 발생하면 해당 페이지(4K)에 대한 더티 플래그값이 설정되어 진다. `DBR`은 더티영역을 주기적으로 감시함으로써 `Flag`가 설정된 `SLD` 페이지를 디스크로 백업 한다. `STA(Statistics Data)`는 `DBS`에서 수행되는 질의에 대한 통계 데이터 영역이다.

3.2 동기화 방식

상용 HSS 데이터베이스의 변경 대상이 되는 데이터를 2가지 유형으로 나누어 저장하고 각기 다른 동기화 기법을 적용하였다. 첫 번째는 예1)과 같은 영구 데이터 유형(`P-DATA; Permanent DATA`)으로 가입자의 생성, 삭제 그리고 프로파일 데이터의 변경에 관한 것으로 빈도수는 많지 않으나 실시간으로 백업 시스템에 반영되어야한다. 이들은 외부장치 연동 블록, `MMI` 블록 등에 의해서 유발되는 것으로 데이터의 유실이 허용되지 않는다. `P-DATA`에 속하는 데이터 필드로 부가 서비스 상태 및 정보, 가입자 번호 등이 있다. 두 번째는 예2)와 같은 임시 데이터 유형(`T-DATA; Temporary DATA`)으로 가입자 위치 등록 데이터의 변경에 관한 것이다. 이들은 빈도수가 많고 `MAP` 블록들에 의해서 유발되는 것으로 데이터의 유실이 허용된다. 이 정보 필드로는 가입자의 상태, 위치 정보, 위치/변경 시간 등이 있다. 이러한 두 가지 유형의 데이터에 대해 각기 다른 동기화 기법을 아래와 같이 적용하였다.

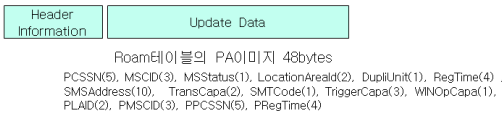
List Encoding 메시드는 P-DATA 영역 중에 있는 필드들이 변경될 때, 변경되는 필드 리스트를 함께 열거하여 동기화 처리를 수행하도록 하는 방법이다. 이 방법은 각 필드마다 필드 식별자(ATT ID)를 표시하여 각 필드를 독립적으로 인식할 수 있도록 처리하였다.

예1) update roam set cfu=02, pcssn=0589280007;



PA Only Encoding 메시드는 T-DATA 영역에 속하는 데이터 필드가 변경되어질 때, T-DATA에 속하는 모든 필드를 동시에 변경하는 방식이다. 이 방식은 모든 필드를 변경함에 따라 필드 식별자를 생략하고 미리 정해진 순서에 따라 동기화 절차가 수행된다.

예2) update roam set pcssn=0589280007, mscid=088e99, locationareaid=0000, regtime=1004,smsaddress=00000000000000000000, triggercapa=cafe11,pmscid=088e99, ppcssn=0528890007



동기화를 위해 저장되는 로그의 경우, 변경되는 데이터의 유형에 따라 유실되어서는 안 되는 P-DATA인 경우에는 List Encoding 메시지를, 빈도가 높고 중요도가 상대적으로 떨어지는 T-DATA인 경우에는 PA Only Encoding 메시지를 각기 적용함으로써 저장되는 로그정보의 공간을 효율적으로 활용하도록 하였다.

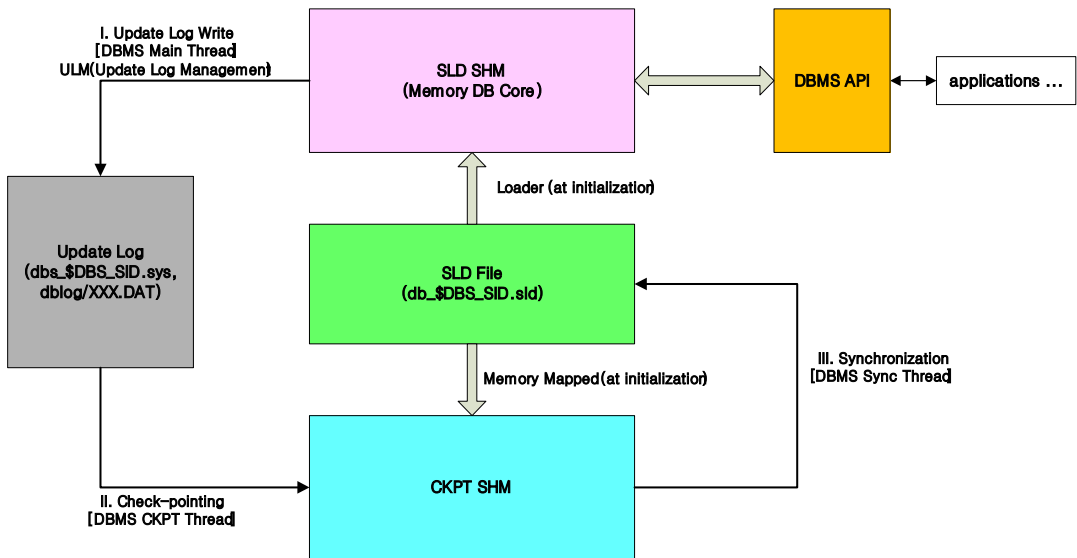
4. HSS-DBS 웨도잉백업 이중화기법

HSS-DBS에서 발생하는 오류 상황에 대비하여 실시간 로그 백업과 체크포인트를 통한 로그의 지연 반응을 통해 시스템 성능에 큰 영향을 주지 않으면서 백업을 수행하고 오류 발생 시 어떠한 경우에도 신속하고 완전하게 복구할 수 있는 웨도잉 백업 이중화 기법을 제안하였다.

4.1 웨도잉 백업 기법

주기억장치 데이터베이스 시스템에서 반드시 고려해야 하는 문제점의 하나는 파손시의 회복이다. HSS 시스템에서도 가입자의 추가 및 삭제, 부가서비스 등록 및 해지 등의 기능은 시스템의 오류로 인하여 손실되어서는 안 되는 정보이므로 안정적인 백업과 이중화 기능이 필요하다.

기존 방식 중 백업 프로세스의 방법을 그대로 사용하는 경우 정보의 변경과 백업이 별개의 프로세스에 의하여 별도로 이루어지기 때문에 오류에 의한 재시동이 일어났을 경우 정보의 불일치가 일어날 위험성을 가지고



[그림 2] HMDDB의 기본 백업구조

있다[15]. 또한, 트랜잭션 개념을 도입하여 로깅 및 주기적인 백업에 의하여 저장된 디스크 SLD를 사용하여 회복을 수행하는 방법은 시스템이 재 시동되었을 경우에 정보의 불일치가 나타나지 않고 데이터베이스가 일관성 있는 상태로 회복될 수 있는 장점이 있는 반면에 기존의 엔진 프로세스 구조를 많이 수정 및 보완해야 하는 구현상의 어려움이 있다.

특히, 최근 연구 중 위치 정보의 변경과 가입자 운용 기능에 따른 정보의 변경에 대하여 갱신 플래그의 값을 달리함으로써 백업 프로세스가 해당 갱신 페이지를 디스크에 저장하는 시점을 조절하는 방법[16]의 경우, 위치 정보는 한가할 때 주기적인 백업을 하고 신규 가입과 같은 관련 운용작업 내용은 즉시 실시간 백업하기 때문에 정보의 불일치를 가져올 수 있으며 복구 시에 다수 사용자의 다량의 위치이동 정보의 일관성을 재설정하는 것이 또 다른 시스템 부하를 유발할 수 있다.

본 논문에서 제안하는 백업 과정은 다음과 같은 순서에 의하여 이루어진다. 먼저, 응용 프로세스의 위치정보 변경 요구를 받아 엔진 프로세스는 데이터베이스 갱신 기능을 수행한 후 갱신 트랜잭션 정보를 디스크상의 로그에 기록하며 백업 프로세스는 체크포인트링 시점마다 REDO 작업을 통한 백업 로그를 SLD 디스크에 저장된 이전 데이터베이스 정보에 반영하여 일관성을 유지한다.

HMDB의 기본 백업구조는 그림 2와 같다. 그림 2에서 보듯이 HMDB는 2개의 공유 메모리인 SLD SHM(System Loading Data SHared Memory)의 내용과 CKPT SHM(CHeck Point SHared Memory)을 가지고 있으며 초기화 시에, 로더는 SLD 파일을 SLD SHM으로 로딩 시킨 뒤, CKPT SHM을 구성하고, SLD 파일을 이에 메모리 매핑시킨다. HMDBMS는 여러가지 스레드(thread)로 구성되어 동작한다.

• 로깅

SLD SHM의 변화를 실시간으로 디스크의 SLD 파일에 쓰기를 하는 것은 성능에 심각한 영향을 미치므로 로그를 활용한 웨딩 백업을 수행한다. SLD SHM에 변화가 발생했을 때, 메인-스레드는 트랜잭션 정보를 앞에서 설명한 로그 형식으로 변환하여 실시간으로 저장한다. 먼저, dbs_\$DBS_SID.sys 파일을 메모리 매핑시키고 이 메모리에 로그를 실시간적으로 저장하게 되는데 dbs_\$DBS_SID.sys는 50개의 2Mbyte 로그 저장용 버퍼들로 이루어지며 이 파일은 메모리 매핑되어 있어 디스크 입출력 없이 고속으로 접근이 가능하다. 버퍼는 자유리스트(free-list)와 더티-리스트(dirty-list) 두 종류의 연결 리스트로 구성되어 더티-리스트는 현재 로그가 저장 중

이거나 2Mbyte 용량을 다 써버린 버퍼들로 구성되며 소진될 경우를 대비해 아카이브-스레드(archive-thread)는 기본적으로 2초에 한 번씩 더티-리스트에서 버퍼를 하나씩 꺼내어 이를 dblog 디렉토리 밑에 XXX.DAT 파일로 저장하게 된다.(당연히 XXX.DAT파일의 크기도 2Mbyte이며 이름은 항상 dblog_\$DBS_SID_xxxx.DAT이다. 여기서 xxx는 파일 번호로 00000~99999의 범위를 가진다). 그리고, 저장된 버퍼를 다시 자유-리스트에 넣어 둔다. 이로써 로그는 두 부분에 저장되게 된다. 즉, 메모리 매핑된 방식으로 인하여 고속 저장을 담당하는 dbs_\$DBS_SID.sys 파일이 1차 저장 장소이고, 아카이브-스레드에 의해서 생성된 dblog/XXX.DAT 파일이 2차 저장 장소이다. 이러한 2단계 저장방식에서 사용되는 1차 저장장소인 메모리 버퍼이고 2차 저장장소인 디스크 버퍼 이 둘 모두를 합쳐서 로그 버퍼풀(log buffer pool)이라고 부른다. 모든 로그는 앞에서 언급한 2단계 저장방식에 따라 찾을 수 있어야 한다.

• 체크포인트링

SLD SHM의 변화에 따른 로그 정보를 2단계 저장방식에 따라 메모리 버퍼와 디스크 버퍼에 저장함을 보였지만 결국 최종적인 목적은 이 로그 정보를 가지고 SLD file을 갱신하는 것이다. 이는 결국 디스크 입출력을 요구하는데 이를 효율적으로 구현하기 위해 체크포인트링 개념을 도입하였다. 체크포인트링 CKPT라는 것은 dbs_\$DBS_SID.sys파일을 포함한 로그 버퍼풀에서 로그를 추출하여 이를 CKPT SHM에 계속해서 반영하여 쓰는 기능이다. 체크포인트-스레드는 기본적으로 2초마다 한 번씩 동작하여 ckpt_rb_area_t의 pos에 해당하는 update-log부터 시작하여 wrpos에 이르기까지 모든 update-log를 CKPT SHM에 쓰기 기능을 수행한다. HMDBMS의 스레드중 하나인 동기-스레드는 기본 5초마다 한 번씩 동작하여 CKPT SHM의 메모리 페이지중 체크-스레드에 의해서 변경된 페이지(4096 byte)를 SLD file로 쓰기 기능을 수행한다.

첫 번째 ULM(Update Log Management) 과정을 통해 SLD SHM의 메모리 데이터베이스 내용의 변경 사항을 비휘발성 장치에 로그 형태로 저장함으로써 오류 발생 시 메인 메모리 데이터베이스가 손실되더라도 SLD 파일의 이전 데이터베이스 버전과 로그 정보를 사용하여 복구가 가능하도록 하는 기본적인 백업 작업이 수행된다.

두 번째 과정으로 체크포인트-스레드에 의해 일정시간 간격을 두고 체크포인트 시점마다 SLD 파일의 초기 버전인 CKPT SHM 데이터베이스에 수정 로그를 재실행(Redo)함으로써 HSS 시스템은 두 개의 주기역장치 데이

터베이스를 갖게 된다. 이는 오류 발생시, SHM들이 이원화되어 독립적으로 운영되었을 경우, 가장 최근 체크포인트 시점이후의 변경 로그만 반영하면 신속하게 기존 데이터베이스의 대체가 가능하기 때문에 이상 발생 시 신속한 대응이 가능하고 성능 향상을 목적으로 체크포인트 간격을 조정함으로써 SLD SHM의 실시간 웨도잉 버전으로도 활용할 수 있다.

세 번째인 동기화 과정은 CKPT SHM 웨도잉 데이터베이스 버전을 디스크상의 SLD 파일에 동기화시키는 단계로 기존 SLD SHM과의 직접적인 인터페이싱없이 독립적 수행을 통해 백업함으로써 실시간 서비스에 부하를 주지 않으면서 완전한 데이터 백업이 이루어지도록 한다.

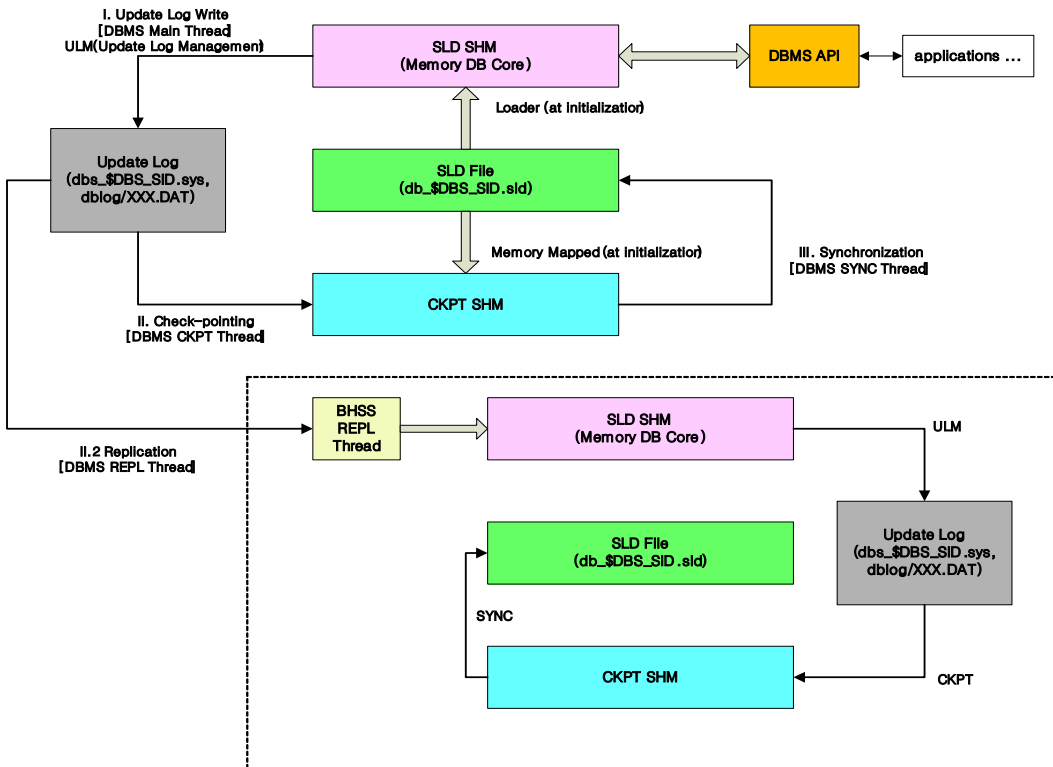
4.2 백업 이중화 기법

그림 2에서 설명한 HMDb의 기본 백업의 확장된 형태로 백업 이중화의 기본 구조는 다음 그림 3과 같다. 그림 2의 첫 번째 단계를 통해 생성된 수정 로그 정보가 이중화-스레드에 의해 백업 B-HSS(Backup HSS)시스템에 전달되어 C-HSS(Commercial HSS)에서 이루어진 데이터 수정 작업과 백업 작업이 똑 같은 방식으로 재실행됨에 의해 원격 이중화 시스템의 운영이 이루어진다. 백업 이

중화 기법을 통해서 시스템 서비스의 중단없는 안정적인 지원이 가능하게 됨으로써 시스템의 신뢰성을 크게 향상시킬 수 있다.

백업 이중화(replication)의 기본 개념은 C-HSS와 B-HSS 간에 로그 정보를 전달하여, B-HSS가 실시간적으로 C-HSS와 같은 메모리 DB 내용을 가지도록 하는 것이다.

C-HSS와 B-HSS는 모두 각각의 이중화-스레드(Repl-thread)가 구동된다. 먼저, B-HSS의 이중화-스레드(recv thread)가 C-HSS의 이중화-스레드(send thread)에 TCP/IP 연결을 요구함으로써 백업 이중화가 시작된다. 연결 요구 시에, B-HSS는 C-HSS에 이중화 작업의 시작 위치 정보를 전송하는데, 이 정보는 B-HSS의 wrpos 정보이다. 이 위치 정보를 수신받은 C-HSS의 이중화-스레드는 로그 풀에서 해당 로그를 찾아서 B-HSS로 전송하기 시작한다. B-HSS는 C-HSS로부터 로그를 전송받아, 마치 C-HSS에서 애플리케이션으로부터 DBMS API를 통해 DB 트랜잭션을 요구받은 DBMS가 동작하듯이 동일하게 동작하게 된다. 이중화를 수행하기 위해 C-HSS에는 다음과 같은 종류의 상태 간에 전이가 이루어진다.



[그림 3] 백업 이중화 구조

C-HSS : <INIT, STOP, SEND_DISCONN, SEND_CONN1, SEND_CONN2>

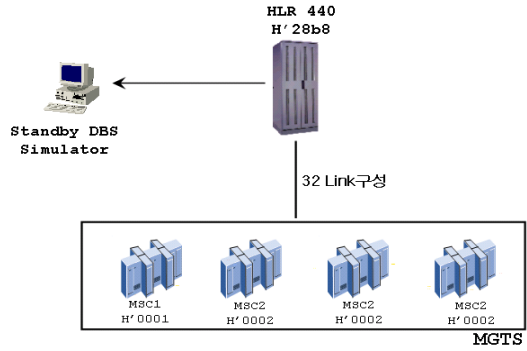
INIT는 최초의 상태이다. STOP 상태는 운영자의 명령에 의해 이중화가 중지되었을 때 전이되는 상태이다. INIT/STOP 상태에서는 운영자의 명령어를 수신하는 메시지 큐를 1초에 한 번씩 검색하게 되는데, 만약 메시지 큐에 있는 메시지가 이중화-시작(repl-start) 메시지라면 SEND_DISCONN 상태로 전이하게 된다. 만약 메시지 큐에 있는 메시지가 이중화-중지(repl-stop) 메시지라면 현 상태를 유지하게 된다. DBMS 초기화시에는 항상 이중화-스레드로 이중화-시작 메시지를 송신하므로 INIT_DISCONN 상태로 초기화시에 전이하게 된다. SEND_DISCONN 상태에서 B-HSS의 연결요구를 기다리게 된다. 이중화 연결은 두 가지 종류가 있는데 로그를 요구하는 연결이 이루어지면 SEND_CONN1 상태로 전이되고, SLD 다운로드를 요구하는 연결이 이루어지면 SEND_CONN2 상태로 전이된다. CONN1 또는 CONN2 상태에서 비정상 상황이 발생하면 항상 DISCONN 상태로 전이하여 다시 B-HSS의 연결요구를 기다리게 된다. B-HSS에서 이루어지는 상태 전이는 다음과 같은 상태들 사이에 이루어진다.

B-HSS : <INIT, STOP, RECV_DISCONN, RECV_CONN>

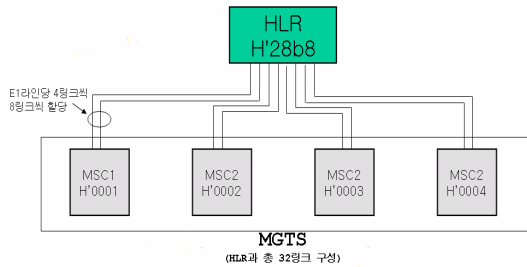
INIT/STOP 상태에서, 만약 메시지 큐에 있는 메시지가 이중화-시작 메시지라면 RECV_DISCONN 상태로 전이하게 된다. 만약, 메시지 큐에 있는 메시지가 이중화-중지 메시지라면 현 상태를 유지하게 된다. DBMS 초기화시에는 항상 이중화-스레드로부터 이중화-시작 메시지를 송신하므로 INIT_DISCONN 상태로 초기화시에 전이하게 된다. RECV_DISCONN 상태에서 C-HSS로 연결 요구를 했을 때(물론 연결요구는 B-HSS의 초기화 방식에 따라 수정-로그(update-log) 연결요구 또는 피어(peer) SLD 다운로드 연결요구가 있을 수 있다) 연결이 이루어지면 RECV_CONN 상태로 전이한다. CONN 상태에서 비정상 상황이 발생하면 항상 DISCONN 상태로 전이하여 다시 C-HSS로 수정-로그 연결요구를 수행하게 된다.

4.3 평가

스레딩 백업과 백업 이중화를 구현한 메모리 HSS 데이터베이스 시스템인 HSS-DBS의 최대 성능 시험을 위하여 다음 그림 4, 그림 5와 같은 시험 망을 구성하였다.

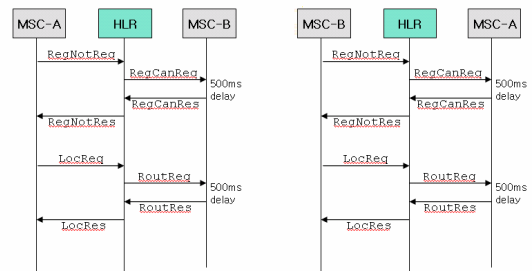


[그림 4] 시험 망 구성도



[그림 5] MGTS와 HSS간의 SS7 구성도(최대 성능 시험)

시험 망에서 MGTS(Message Generation Test System)는 1대, 링크 수는 32링크, MGTS 가용 최대 TPS(Transation Per Second) 예상치는 2,000 TPS(32링크 1,800 TPS)이며 호 처리 시나리오는 MGTS가 발착신 교환기 역할을 수행하며, 위치등록(위치취소 유발)과 위치요구(라우팅요구 유발)를 3 : 2의 비율로 HSS로 송출하도록 하였다. MGTS 노드에서는 위치 취소 및 라우팅 요구 수행 시, 500ms의 시간 지연을 두어 VLR 갱신 효과를 시뮬레이션하게 된다. 이와 관련한 상세 메시지 흐름은 그림 6과 같다.



[그림 6] 메시지 흐름도

시뮬레이션 시험 시 MSC1, MSC2와 MSC3, MSC4에서 각각 그림 6과 같은 메시지 흐름이 동시에 발생하게 된다. 위치 등록의 경우 실제 DB 변경을 위해서 대국당 1만 가입자를 순차적으로 반복해서 위치 등록을 하도록 하였다.

위치 등록 절차는 MSC-A가 RegNotReq(Registartion-NotificationRequest) 요구 메시지를 HLR로 전송하고 HLR은 MSC-B에게 RegCanReq (RegistrationCancelation Request) 요구 메시지를 전송한다. HLR은 MSC-B로부터 RegCanRes(Registration Cancelation Response) 응답 메시지를 받고 MSC-A에게 RegNotRes(Registartion Notification Response) 응답 메시지를 보낸다. 호처리 절차는 위치 등록 절차와 유사하며, RegCan Req 대신 LocReq(LocationRequest) 메시지가 사용되며, RegCanReq 메시지 대신에 RoutReq(RoutingRequest) 메시지를 사용한다. MSC-B에서는 메시지를 수신 후, 500ms를 지연시킨 후 응답 메시지를 전송한다.

시뮬레이션 수행 후의 최대 성능시험에 관한 평가 결과는 다음 표 1과 같다.

시험 환경은 다음과 같다. 첫 번째, 최대 성능시험은 600부터 최대 2000 TPS까지 200 단위로 증가시키며 15 분간의 유휴(IDLE) 데이터를 측정하는 방식으로 진행하였다. 두 번째, 장시간 시험은 1600 TPS로 24시간 진행시켜 MGTS에서 실패가 없음을 확인하고 유휴 데이터를 측정 하는 식으로 진행하였다.

[표 1] 최대 성능 시험 결과

TPS	MSC1		MSC2		MSC3		MSC4		TOT INV	Idle
	REG	LCR	REG	LCR	REG	LCR	REG	LCR		
600	6	4	6	4	6	4	5	3	302	80
800	8	5	7	5	8	5	7	5	400	75
1000	9	6	9	6	9	6	10	7	496	71
1200	11	7	11	7	11	8	12	8	600	67
1400	13	9	13	9	13	9	13	8	696	62
1600	15	10	15	10	15	10	15	10	800	57
1800	17	11	17	11	17	11	17	12	904	51
2000	19	12	18	12	19	13	19	13	1000	47

표 1에서 REG는 RegNotReq(Regisrartion NotificationRequest) 요구 메시지를, LCR은 LocReq(LocationRequest) 메시지를 의미한다. REG/LCR 각각 1회 발생 시, RegCanReq(RegistrationCancelation Request), RoutReq(RoutingRequest) 요구 메시지가 각각 1회 발생한다. TOT INV(Total INVoke)는 MSC1/2/3/4에서 유발하는 총 메시지 회수를 의미한다. 302개의 메시지

가 유발될 경우, 그림 6의 메시지 흐름도에 따라 총 TPS는 대략 600(302*2)이 된다. 각 TPS마다의 숫자(600 TPS에서 6,4,6,4,6,4,5,3)의 의미는 302개에 대해 각 MSC1/2/3/4에서 유발되는 상대적인 메시지 비율이다. 1400 TPS에서 CPU 사용율은 38%이고 유휴율은 62%임을 알 수 있다. 최대 성능 시험 결과로 2,000 TPS에서 CPU 사용율이 53%임을 알 수 있다. 이 의미는 초당 2,000개의 메시지를 안정적으로 처리할 수 있음을 의미한다.

표 2의 24시간 장시간 시험에서는 1,600 TPS 메시지를 24시간 동안 메시지 유실없이 처리됨을 확인할 수 있었다. 이 실험에서는 CPU 사용율이 44%임을 알 수 있었다. 두 성능 실험을 통해 일반적으로 요구되는 1,600 TPS 이상, CPU 유휴율 40%이하를 충족함으로써 HSS 시스템이 매우 안정적이며 상용 서비스를 제공하는데 문제가 없음을 확인할 수 있었다.

[표 2] 24시간 장시간 시험 결과

TPS	MSC1		MSC2		MSC3		MSC4		TOT INV	Idle
	REG	LCR	REG	LCR	REG	LCR	REG	LCR		
1600	15	10	15	10	15	10	15	10	800	56

5. 결론

본 논문에서는 HSS 시스템의 안전성과 성능을 고려한 효율적인 백업 이중화 기법을 제시하였다. 제한한 쉘도잉 백업 기법을 통해 위치 등록으로 인한 변경된 로그 정보를 일정 시간 간격을 두고 지연 회복함으로써 과부하를 최소화하면서 실시간 백업이 가능하도록 하였다. 또한, 로그를 재 수행하여 메모리상의 복제된 데이터베이스를 유지함으로써 오류 발생 시 메모리 혹은 디스크 데이터베이스로부터 신속하게 복구할 수 있으며 원격 서버에 실시간으로 백업하는 백업 이중화 기법을 통해 보다 안정적인 시스템 운영을 보장하였다.

최대 성능 시험과 24시간 장시간 시험을 포함한 시뮬레이션 결과를 통해서 제한한 쉘도잉 백업 이중화 기법을 적용한 HSS-DBS 시스템이 매우 안정적이며 상용서비스를 제공하는데 적합한 성능을 제공함을 확인하였다.

참고 문헌

- [1] Baulier, J. et al, "DataBlitz Storage Manager : Main

Memory Database Performance for Critical Applications," Proc. of the ACM SIGMOD Intl. of Conf. on Management of Data, pp. 519-520, USA, 1999.

[2] Bernstein, P. A. et al., "Concurrency Control and Recovery in Database Systems," Addison Wesley, 1987.

[3] Cha, S. K. et al, "Main-Memory Database for Location Management in Telecommunication," Proc. of Workshop on Databases in Telecommunications, pp. 174-190, 1999.

[4] Dewit, D.J. et al, "Implementation Techniques for Main Memory Database Systems," Proc. Intl. Conf. on Management of Data, ACM SIGMOD, Boston, USA, pp. 1-8, 1984.

[5] Eich, M. H., "Main Memory Database Research Directions," In Database Machines, 6th Intl. Workshop, pp. 251-268, France, 1989.

[6] Eich, M. H., "A Classification and Comparison of Main Memory Database Recovery Techniques," In Proc. of IEEE Intl. of Conf. on Data Engineering, pp. 332-339, USA, 1987.

[7] Garcia-Molina, H, "Main Memory Database Systems An Overview," IEEE Trans on Knowledge and Data Engineering, Vol. 4, No. 6, pp. 509-516, 1992.

[8] Jagadish, H. V. et al, "Dali : A High Performance Main Memory Storage Manager," Proc. 20th Intl. Conf. on Very Large Databases, Santiago, Chille, pp. 48-59, 1994.

[9] Jagadish, H. V. et al, "Recovering from Main-Memory Lapses," Proc. 19th Intl. Conf. on Very Large Databases, Ireland, pp. 391-404, 1993.

[10] Kemme, B. and Alonso, G., "A New Approach to Developing and Implementing Eager Database Replication Protocols," ACM Transactions on Database Systems, Vol. 25, No. 3, pp. 333-379, 2000.

[11] Lehman, T. J. and Carey, M. J., "A High Performance Memory-Resident Database System," Proc. 12th Intl. Conf. on Management of Data, USA, pp. 104-117, 1987.

[12] Peter, M. G et al, "PRISMA/DB A Parallel, Main Memory Relational DBMS," IEEE Trans. on Knowledge and Data Engineering, Vol. 4, No. 6, pp. 541-554, 1992.

[13] Ronstrom, M., "Database Requirement Analysis for a Third Generation Mobile Telecom System," Proc. of Workshop on Databases in Telecommunications, pp. 90-105, 1999.

[14] Yingyuan Xiao, Jun Miyazaki, "A memory subsystem with comparator arrays for main memory

database operations," SAC 2006, pp. 511-512, 2006.

[15] 강은호, 윤석우, 김경창, "실시간 모바일 GIS 응용 구축을 위한 주기억장치 데이터베이스 시스템 설계 및 구현," 정보처리학회 논문지, 11-D권 1호, pp. 11-22, 2004.

[16] 김장환, "분산 모바일 임베디드 시스템 기반의 새로운 위치정보 관리 시스템," 한국통신학회논문지, Vol. 29, No. 12B, pp. 1022-1036, 2004.

[17] 윤용익 외, "디지털 이동통신에서 가입자 정보관리를 위한 데이터베이스 시스템 개발," 한국정보과학회 가을학술발표논문집, Vol. 20, No. 2, pp. 655-658, 1993.

[18] 임선배, 최숙희, 정태의, "이동통신 시스템에서의 고속메모리 상주 객체지향 HLR DB 엔진," 한국정보과학회 가을학술발표논문집, Vol. 20, No. 2, pp. 693-696, 1997.

박 성 진(Seong-Jin Park)

[정회원]



- 1993년 2월 : 고려대학교 일반대학원 전산학과(이학석사)
- 1998년 2월 : 고려대학교 일반대학원 전산학과(이학박사)
- 1998년 3월 ~ 2000년 2월 : 한국전자통신연구원 선임연구원
- 2000년 3월 ~ 현재 : 한신대학교 컴퓨터공학부 부교수

<관심분야>

데이터웨어하우징, 모바일데이터베이스, OLAP 등

박 형 수(Hyung-Soo Park)

[정회원]



- 1995년 2월 : 고려대학교 일반대학원 전산학과 석사
- 2008년 2월 : 고려대학교 정보경영공학전문대학원 박사
- 1995년 3월 ~ 2002년 2월 : LG 전자 선임연구원
- 2003년 3월 ~ 현재 : (주)엔텔리아 개발 팀장

<관심분야>

정보보호, 데이터베이스, 무선이동통신, 네트워크보안 등