

스냅샷을 가지는 다중 레벨 공간 DBMS를 기반으로 하는 센서 미들웨어 구조 설계

Design of Sensor Middleware Architecture on Multi Level Spatial DBMS with Snapshot

오은석* / Eun-Seog Oh

김재홍*** / Jae-Hong Kim

김호석** / Ho-Seok Kim

배해영**** / Hae-Young Bae

요약

최근 들어, 사용자가 주변 환경 및 요구 정보의 변화를 의식하지 않고 작업 환경과 수행하는 일에 집중하도록 배려하는 인간 중심 컴퓨팅 환경에 대한 연구 개발이 활발히 진행되고 있다. 그러나 이러한 컴퓨팅 환경에서 미들웨어는 사용자에게 RFID 센서로부터 들어오는 대량의 정보에 대한 처리 부하를 줄이기 위하여 분석이 끝난 스트림 데이터를 삭제한다. 따라서 사용자의 데이터 웨어하우징이나 데이터 마이닝에 필요한 확률, 통계 정보에 대한 요청, 또는 반복적이면서 동일한 데이터에 대한 요청을 처리할 수 없다는 문제점을 가진다.

본 논문에서는 기존의 미들웨어에서 문제가 되었던 과거 스트림 데이터 재사용 문제를 해결하기 위해, 사용자가 빈번하게 요구하는 데이터들을 스냅샷을 가지는 다중 레벨 공간 DBMS에서 관리하는 센서 미들웨어 구조를 설계하였다. 본 시스템은 사용자가 요구하는 데이터 마이닝이나 데이터 웨어하우징과 같은 과거 스트림 정보를 사용한 서비스 요청을 위해, 미들웨어에서 필터링된 과거 스트림 데이터를 디스크 데이터베이스에서 관리한다. 그리고 디스크 데이터베이스에 저장된 스트림 데이터 중에서 사용자에 대한 높은 재사용 빈도를 가지는 데이터들을 스냅샷의 형태로 메모리 데이터베이스에 저장하고 이를 관리한다. 또한, 본 시스템은 메모리 데이터베이스에 저장된 스냅샷 데이터의 높은 데이터 재사용성과 신속한 서비스를 유지하기 위해서 주기적인 메모리 데이터베이스 관리 정책을 수행한다. 본 논문은 기존의 미들웨어에서의 스트림 데이터에 대한 반복적인 요청, 또는 과거 스트림 데이터를 이용한 정책 결정 서비스 요청에 대한 서비스를 제공할 수 없는 문제들을 해결하였다. 그리고 메모리에 저장된 데이터에 대한 높은 데이터 재사용성을 유지함으로서 사용자에게 지속적으로 다양하고 신속한 데이터 서비스를 제공한다.

Abstract

Recently, human based computing environment for supporting users to concentrate only user task without sensing other changes from users is being progressively researched and developed. But middleware deletes stream data processed for reducing process load of massive information from RFID sensor in this computing. So, this kind of middleware have

■ 논문접수 : 2006. 3. 17 ■ 심사완료 : 2006. 4. 12

* 인하대학교 컴퓨터 · 정보공학과 석사과정(oes@dblab.inha.ac.kr)

** 인하대학교 대학원 컴퓨터 · 정보공학과 박사과정(hskim@dblab.inha.ac.kr)

*** 영동대학교 컴퓨터공학과 교수(jhong@youngdong.ac.kr)

**** 교신저자 인하대학교 컴퓨터공학부 교수(hybae@dblab.inha.ac.kr)

problems when user demands probability or statistics needed for data warehousing or data mining and when user demands very important stream data repeatedly but already discarded in the middleware every former time.

In this paper, we designs Sensor Middleware Architecture on Multi Level Spatial DBMS with Snapshot and manage repeatedly required stream datas to solve reusing problems of historical stream data in current middleware. This system uses disk database that manages historical stream datas filtered in middleware for requiring services using historical stream information as data mining or data warehousing from user, and uses memory database that manages highly reuseable data as a snapshot when stream data stored in disk database has high reuse frequency from user. For the more, this system processes memory database management policy in a cycle to maintain high reusement and rapid service for users.

Our paper system solves problems of repeated requirement of stream datas, or a policy decision service using historical stream data of current middleware. Also offers variant and rapid data services maintaining high data reusement of main memory snapshot datas.

주요어: RFID, 스트리밍, 센서 미들웨어, 다중 레벨 공간 DBMS

Keywords: RFID, Streaming, Sensor Middleware, Multi Level Spatial DBMS

1. 서 론

최근 5C(Computing, Communication, Connectivity, Contents, Calm) 또는 5Any(Any-time, Any-where, Any-network, Any-device, Any-service)를 지향하는 유비쿼터스 환경을 실현시키기 위한 많은 연구 개발이 진행되고 있고, RFID 관련 기술 및 데이터를 처리하는 소프트웨어인 RFID 미들웨어 기술이 주목을 받고 있다. RFID 미들웨어는 유비쿼터스 컴퓨팅 환경 실현을 위한 주요 시스템의 하나로서, RFID를 기반으로 유비쿼터스 환경에서 폭넓은 데이터들을 효율적으로 정제 및 관리하는 시스템이다. RFID 미들웨어는 사용자가 필요로 하는 다양한 응용 서비스에 대응하기 위하여 사용자가 관심있어 하는 정보를 적절한 위치에서, 적절한 시간에, 정확한 서비스를 제공하기 위한 목적을 가지며, 이를 위해서 처리해야 할 데이터의 양, 데이터 형식, 그리고 동시에 처리되는 필터링 조건의 수 등의 조건에 대응하여 다양한 필터링 과정을 실시간으로 수행한다.

그러나 기존의 미들웨어들은 미들웨어로 끊임없이 입력되는 스트림 데이터에 대한 처리 부하를 줄이기 위해서 스트림 데이터를 저장하는 정적 저장소를 가지고 있지 않다. 따라서 기존의 미들웨어는 다음과 같은 두 가지 문제점을 가진다. 첫 번째로, 사용자가 데이터 웨어 하우징이나 데이터 마이닝을 위한 확률 또는 통계 정보를 미들웨어에게 요구할 경우, 기존의 미들웨어는 과거에 처리된 스트림 데이터를 저장하지 않는 특성으로 인하여 이에 대한 서비스를 처리할 수 없다. 두 번째로, 사용자가 미들웨어에게 동일한 서비스를 빈번하게 요구할 경우, 미들웨어는 이에 대한 서비스를 요청받을 때마다 동일한 스트림 데이터에 대한 입력과 삭제를 반복하게 된다.

본 논문에서는 이러한 기존의 미들웨어에서 발생하는 문제점을 해결하고 사용자가 수행하는 일의 집중과 업무의 생산성을 향상시키기 위해서, 스냅샷을 가지는 다중 레벨 공간 DBMS를 기반으로 하는 센서 미들웨어 구조를 설계하였다[1, 7]. 본 시스템은 미들웨어 내에서 필터링된 스트림 데이터들을 스냅샷을

가지는 다중 레벨 공간 DBMS에서의 디스크 데이터베이스에 저장하고 관리함으로서, 사용자가 정책 결정을 위한 확률이나 통계 데이터 등을 요청할 경우 저장된 과거 스트림 데이터를 이용하여 적절한 서비스를 제공한다. 그리고 디스크 데이터베이스에 저장된 스트림 데이터 중에서 사용자의 높은 재사용 빈도를 가지는 일부의 데이터의 경우, 이를 메인 메모리에서 스냅샷의 형태로 관리함으로서, 사용자가 높은 빈도로 요청하는 스트림 데이터에 대한 신속한 데이터 서비스를 제공한다[2]. 또한, 시스템은 스냅샷 관리 정책을 이용하여 메인 메모리에 저장된 스냅샷 데이터의 높은 재사용성을 유지함으로서 사용자가 원하는 다양한 서비스 요청을 항상 정확하고 신속하게 제공한다.

본 논문의 구성은 다음과 같다. 2장은 관련 연구로, 기존의 미들웨어 구조와 스냅샷을 가지는 다중 레벨 공간 DBMS에 대해 설명한다. 3장에서는 본 논문에서 제안하는 스냅샷을 가지는 다중 레벨 공간 DBMS를 기반으로 하는 센서 미들웨어 구조와 스트림 데이터의 내부적인 처리 과정에 대한 내용을 다룬며, 4장에서는 본 시스템을 기반으로 지역 정보를 다른 어플리케이션에 대하여 간략한 소개를 한다. 끝으로, 5장에서는 결론 및 향후 연구 방향을 제시한다.

2. 관련 연구

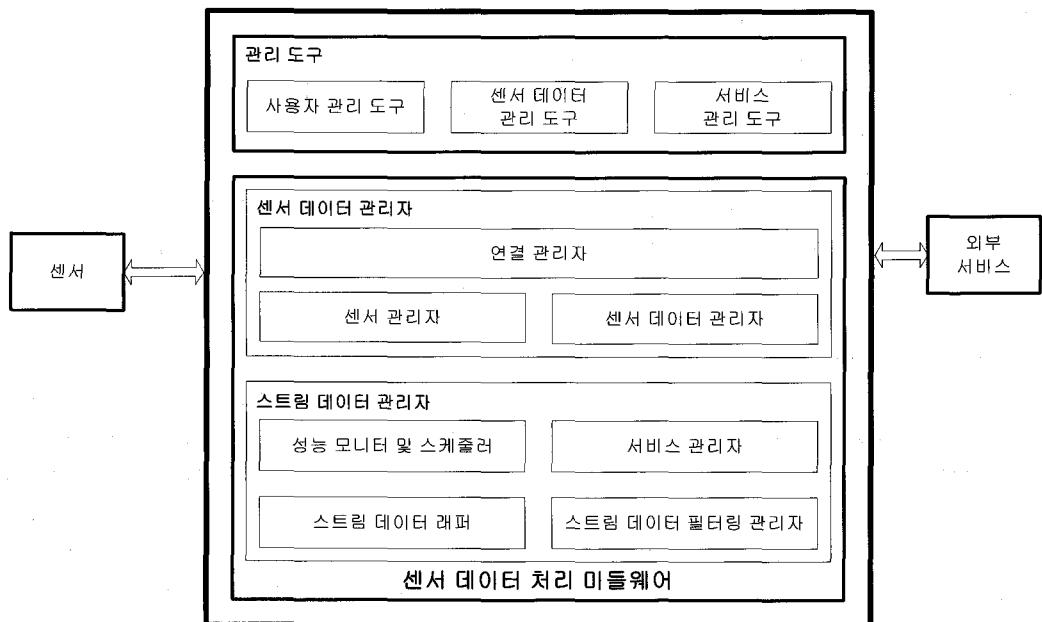
본 장에서는 본 논문에서의 기반이 되는 센서 데이터 처리 프레임 워크인 RFID 미들웨어, 그리고 저장된 데이터의 특성을 고려하면서 효율적으로 데이터의 재사용성을 관리하는 시스템인 스냅샷을 가지는 다중 레벨 공간 DBMS에 대한 시스템 구조에 대하여 알아본다. 그리고 기존의 미들웨어에서의 스트림 데이터 처리 과정과 문제점들을 알아보며, 이를 해결할 수 있는 효율적인 DBMS 시

스템의 필요성을 알아본다.

2.1 RFID 미들웨어

RFID 미들웨어는 다양한 응용 서비스에서 사용되는 RFID를 적절한 장소와 적절한 시간에 응용 서비스로 전달하는 구조를 가진다. RFID 미들웨어는 RFID 기반의 유비쿼터스 환경의 응용 서비스를 지원하며 끊임없이 지속적으로 입력되는 데이터를 실시간으로 정확하게 처리하고 응용 서비스에서 요구하는 결과를 획득해서 전달한다[3]. 이와 같이 지속적으로 입력되는 대량의 데이터 스트림을 처리하기 위해 RFID 미들웨어는 아래의 <그림 1>과 같은 시스템 구조를 가진다[4]. 그림의 미들웨어는 크게 3가지 구성 요소로 나누어 볼 수 있는데, RFID 센서 데이터 처리를 지원하기 위한 다양한 관리 도구, 센서로부터 들어온 데이터 서비스를 위한 센서 데이터 관리자, 입력된 센서 데이터를 분석하고 원하는 내용을 얻기 위한 매핑 과정을 지원하는 스트림 데이터 관리자로 구성된다.

<그림 1>에서 보듯이, 기존의 미들웨어는 막대한 스트림 데이터들을 신속하게 처리해야 하는 문제점과 스트림 데이터 처리 과정에 수반되는 부하 문제를 줄이기 위해서 정적 저장 공간을 가지지 않는다. 오직 미들웨어 내에 미리 지정된 트리거에 해당하는 정보를 분석하기 위해서 일시적으로 데이터를 저장하며, 저장된 데이터는 분석이 끝남과 동시에 데이터의 재사용성, 가치, 중요성에 관계없이 삭제된다. 기존의 이러한 미들웨어 시스템에서는 사용자가 동일한 데이터를 이용하여 처리되는 서비스를 빈번하게 요구할 경우, 또는 과거에 높은 중요성을 가졌지만 미들웨어 내에서 더 이상 존재하지 않는 데이터를 이용한 서비스를 요구할 경우, 이에 대한 서비스를 처리할 수 없다는 문제점을



<그림 1> RFID 미들웨어의 시스템 구조

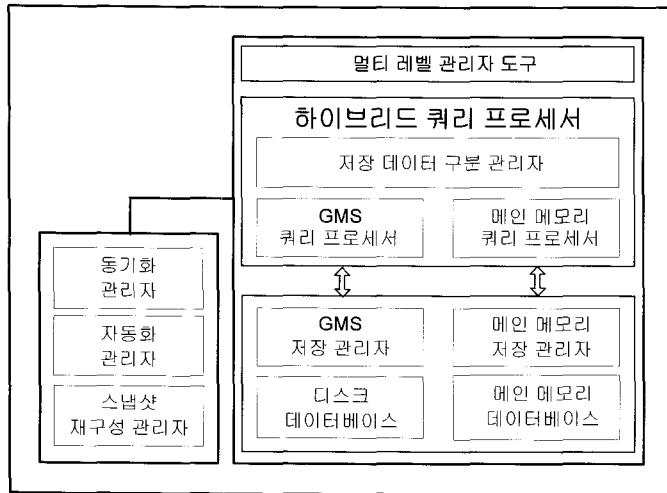
가진다. 따라서 미들웨어는 사용자의 특정 과거 데이터에 대한 요청, 과거 데이터 집단을 분석하는 확률 및 통계 서비스에 대한 요청, 그리고 특정 과거 데이터에 대한 높은 재사용 빈도를 고려하면서 이에 대한 효율적이고 신속한 서비스를 제공하는 데이터 저장 관리 시스템이 요구된다.

2.2 스냅샷을 가지는 다중 레벨 공간 DBMS

스냅샷을 가지는 다중 레벨 공간 DBMS는 저장된 데이터들이 가진 대용량, 또는 빠른 트랜잭션 속도와 같은 고유의 특성을 고려하면서 데이터를 관리하는 DBMS이다 [5, 10]. 이 시스템은 상대적으로 낮은 사용 빈도를 가지는 대용량 공간 데이터를 관리하기 위한 디스크 기반 저장 계층을 기반으로, 높은 사용 빈도와 재사용 가능성을 가진다고 판단되는 일부의 디스크 데이터베이스

저장 데이터를 스냅샷의 형태로 메인 메모리 계층에서 관리한다 [6]. 그리고 계층 간의 데이터가 가지는 일관성을 유지시키기 위해서, 갱신된 디스크 계층의 데이터 값을 메모리 계층에 저장된 스냅샷 데이터를 주기적으로 갱신한다. 또한, 시스템은 메모리 데이터베이스 계층에 저장된 스냅샷 데이터가 가지는 데이터 재사용 빈도를 주기적으로 평가하고 사용 가치가 상대적으로 낮은 스냅샷 데이터를 주기적으로 삭제하는 스냅샷 갱신 연산을 이용하여 제한된 메모리 데이터베이스 공간에서의 빠른 스냅샷 데이터 응답 속도를 유지한다.

스냅샷을 가지는 다중 레벨 공간 DBMS의 구성은 <그림 2>와 같이 기존의 디스크 기반의 공간 DBMS인 GMS 시스템 [7]과 추가된 메인 메모리 DBMS 및 관련 컴포넌트들로 구성되어 있다. 스냅샷을 가지는 다중 레벨 공간 DBMS는 기존의 디스크 데이터베이스 기반의 DBMS를 기반으로 데이터를 저장하고 관리하



<그림 2> 스냅샷을 가지는 다중 레벨 공간 DBMS의 시스템 구조

며, 일부의 높은 접근 빈도, 사용 횟수, 그리고 빠른 응답 속도 등을 요구하는 데이터에 대해서는 데이터의 전체 또는 일부를 스냅샷으로 생성하고 이를 메인 메모리 데이터베이스에서 관리한다. 그리고 이렇게 두 개의 서로 다른 데이터베이스 계층을 유지 관리하기 위해서 시스템은 관리 시스템에서 중요하다고 판단되는 데이터를 스냅샷으로 관리하는 메인 메모리 기반 데이터베이스, 데이터의 저장 위치를 판단하는 하이브리드 쿼리 프로세서[9], 시스템의 효율적인 데이터 관리를 위한 동기화 관리자, 자동화 관리자, 스냅샷 재구성 관리자, 그리고 관리자에 의한 직접적인 스냅샷의 생성 및 삭제를 위한 다중 레벨 관리자들과 같은 다양한 컴포넌트들을 사용한다. 이러한 시스템 특성을 이용하여, 스냅샷을 가지는 다중 레벨 공간 DBMS는 기존의 미들웨어에서 발생하는 스트림 데이터 재사용 문제들을 해결한다. 스냅샷을 사용하는 다중 레벨 공간 DBMS는 미들웨어에서 필터링된 스트림 데이터를 디스크 데이터베이스에 저장하고 관리함으로서, 저장된 과거 데이터를 필요로 하는 사용자의 다양한 서비스를 처리할 수 있다. 그리고 일부의 높은 재사용 빈도를 가지는 데이터

에 대해서는 메인 메모리 스냅샷의 형태로 저장하고 이를 이용함으로서, 동일한 서비스에 대한 빈번한 요청이 있을 경우 사용자에게 신속한 데이터 서비스를 제공한다.

3. 스냅샷을 가지는 다중 레벨 공간 DBMS를 기반으로 하는 센서 미들웨어

본 장에서는 스냅샷을 가지는 다중 레벨 공간 DBMS와 센서 미들웨어를 연동하여, 수집된 센서 스트림 데이터를 분석하는 시스템 구조와 처리 과정을 알아본다. 그리고 본 논문의 시스템을 사용함으로서 기존의 미들웨어에서 문제가 되는 과거 스트림 데이터를 이용한 데이터 서비스 요청, 동일한 스트림 데이터에 대한 빈번한 서비스 요청, 그리고 대용량 스트림 데이터에 대한 효율적인 스트림 데이터 처리 과정에 대해 알아본다.

3.1 시스템 구조

스냅샷을 가지는 다중 레벨 공간 DBMS 센서 데이터 처리 미들웨어는 기존의 미들

웨어가 가지는 기능에 DBMS를 이용한 의미 있는 과거 스트림 데이터의 관리 기법을 활용함으로서 사용자에게 보다 확장된 데이터 서비스를 제공하는 시스템이다. 본 시스템은 미들웨어에서 사용자가 원하는 스트림 데이터를 지원하기 위하여 스트림 데이터를 관리하기 위한 각종 관리 도구, 센서 데이터 관리자, 스트림 데이터 관리자, 저장 데이터 관리자, 그리고 스냅샷을 가지는 다중 레벨 공간 DBMS로 구성되어 있다.

센서 데이터 관리자는 스트림 데이터를 수집하는 이기종 센서들의 연결 상태를 관리하는 센서 관리자, 센서들과 미들웨어 간의 연결 상태를 관리하는 센서 연결 관리자, 그리고 센서에서 미들웨어로 입력된 데이터를 저장하는 센서 데이터 관리자로 구성된다. 우선, 데이터 스트림 관리자는 센서 데이터 관리자로부터 얻은 센서 데이터를 미들웨어에서 분석 가능한 스트림 데이터의 형태로 변환하는 스트림 데이터 래퍼, 그리고 변환된 스트림 데이터가 미들웨어 내에 지정된 필터링 조건과 트리거를 만족하는지를 분석하는 스트림 데이터 필터링 관리자와 같은 기존의 미들웨어에서 스트림 데이터를 처리하는 콤포넌트를 가지고 있다. 또한, 스트림 데이터 관리자는 스냅샷을 가지는 다중 레벨 공간 DBMS에 저장될 스트림 데이터에 대한 처리 연산을 위한 확장된 사용자 서비스를 지원하는 관리자, 그리고 시스템 내에서 스트림 데이터에 대한 전송 기능을 담당하는 스트림 데이터 전송 관리자를 가진다. 마지막으로, 확장 서비스 지원 관리자는 사용자가 과거 스트림 데이터를 이용하는 확률이나 통계 등에 관한 서비스 요청, 또는 동일한 스트림 데이터에 대한 반복적인 서비스 요청 등과 같은, 기존의 미들웨어에서 처리할 수 없었던 서비스를 지원한다.

스냅샷을 가지는 다중 레벨 공간 DBMS

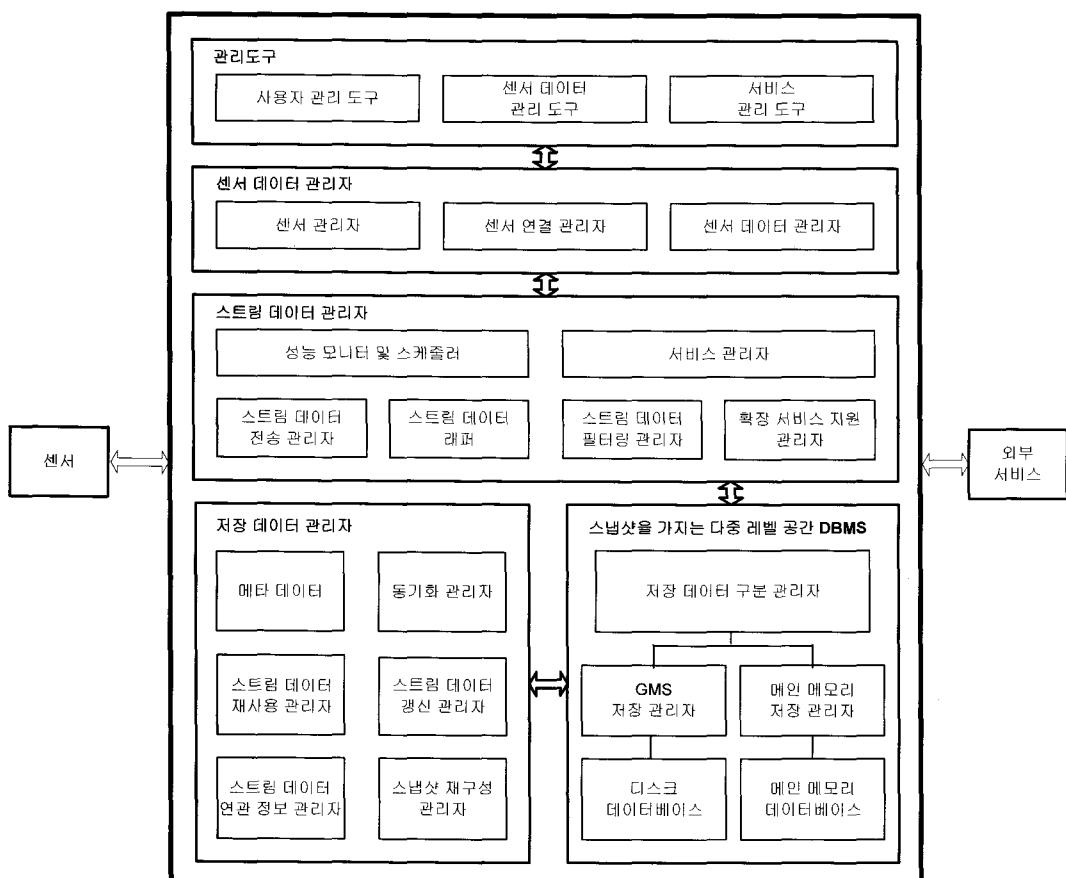
는 사용자가 요구하는 다양한 서비스를 지원하기 위해서 스트림 데이터를 저장하고 관리하는 시스템이다. 따라서 스트림 데이터를 효율적으로 관리하고, 사용자가 필요로 하는 데이터를 신속하게 서비스하기 위한 메인 메모리 스냅샷 관리 기능이 크게 요구된다. 본 논문에서는 이러한 기능을 강화하기 위해서 기존의 스냅샷을 가지는 다중 레벨 공간 DBMS의 기본적인 콤포넌트 이외에, 스트림 관리를 위한 메타 데이터, 스트림 데이터 재사용 관리자, 스트림 데이터 연관 정보 관리자, 그리고 스트림 데이터 간접 관리자를 추가하였고, 미들웨어에서 필요로 하지 않는 일부의 기능을 수정하였다. 스냅샷을 가지는 다중 레벨 공간 DBMS는 저장 데이터 구분 관리자와 각각의 데이터베이스, 그리고 이러한 데이터베이스들에 저장된 데이터들을 관리하기 위한 저장 관리자로 구성된다. 우선, 저장 데이터 구분 관리자는 메타 데이터를 이용하여 전달받은 스트림 데이터를 분석하고, 해당 데이터가 가지는 데이터 재사용성에 따라 디스크 데이터베이스 또는 메인 메모리 데이터베이스로 저장 위치를 결정한다. 그리고 디스크 데이터베이스는 대용량, 또는 상대적으로 낮은 사용빈도를 가지는 스트림 데이터들을 관리하며, 메인 메모리 데이터베이스는 디스크 데이터베이스에 저장된 스트림 데이터들 중에서 사용자가 빈번하게 요청하는 일부의 스트림 데이터들을 스냅샷의 형태로 저장하고 이를 관리한다.

마지막으로, 저장 데이터 관리자는 스냅샷을 가지는 다중 레벨 공간 DBMS에 저장된 스트림 데이터가 가지는 재사용 횟수, 사용비율, 최근 참조 시간 등과 같은 다양한 데이터 재사용 기준들을 이용하여 저장 가치가 높은 데이터를 분석하고 관리하는 기능을 가진다. 메타 데이터는 저장된 스트림 데이터를 분석하는 데 필요한 정보들을 기록

한다. 동기화 관리자는 스트림 데이터의 저장과 갱신 연산에서 발생하는 데이터베이스 간의 데이터 일관성 문제를 관리하며 [8], 스트림 데이터 재사용 관리자는 저장된 데이터에 대한 높은 재사용 우선 순위를 관리하는 데 사용된다. 스트림 데이터 갱신 관리자는 스트림 데이터 관리 인덱스를 이용하여 메인 메모리에 저장된 스냅샷에 대한 저장 가치를 판단하고 더 이상의 높은 재사용 우선 순위를 가지지 않는 스냅샷을 삭제한다. 그리고 필요에 따라, 디스크 데이터베이스에 저장된 스트림 데이터를 분석하고 이에 대한 삭제 작업을 수행할 수도 있다. 또한, 스

트립 데이터 연관 정보 관리자는 가장 높은 재사용 빈도를 가지는 데이터의 패턴이나 주기와 같은 연관 정보를 관리한다. <그림 3>은 본 시스템의 이러한 스트림 데이터 관리 구조를 보여준다.

본 논문에서는 이와 같은 확장된 미들웨어 관리 시스템을 이용하여, 기존의 센서 미들웨어에서 문제가 되었던 과거 스트림 데이터에 대한 요청, 과거 스트림 데이터들에 대한 분석으로 얻는 확률 및 통계 데이터에 대한 요청 등에 대하여 효율적인 데이터 서비스를 제공한다. 본 시스템은 사용자로부터의 서비스 요청이 있을 경우 미들웨어 고유



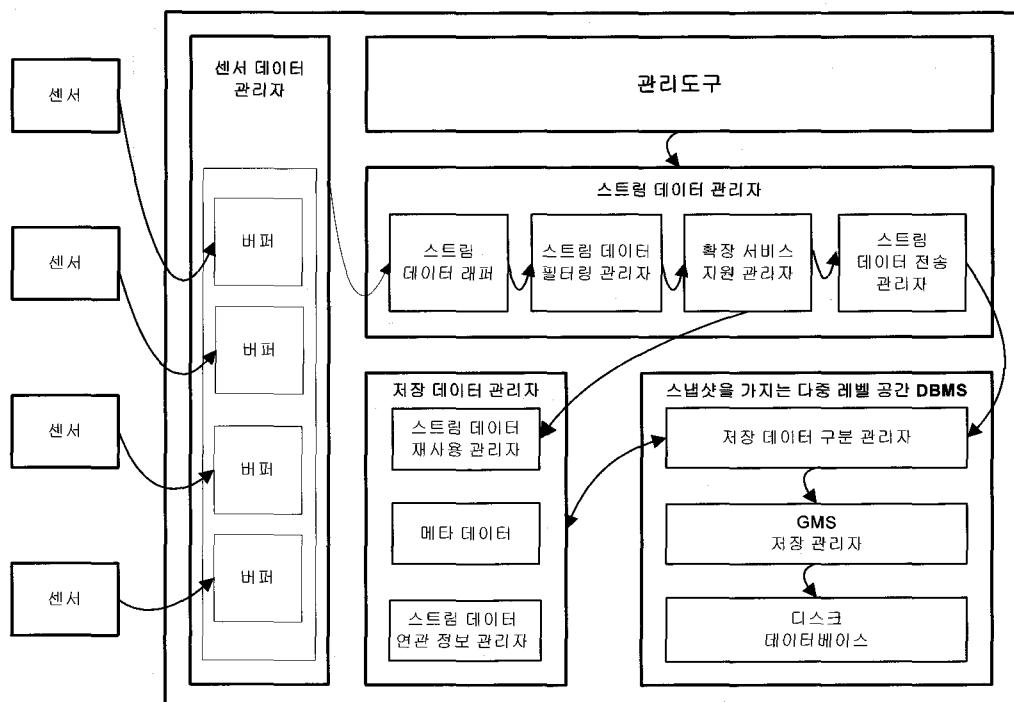
<그림 3> 스냅샷을 가지는 다중 레벨 공간 DBMS를 기반으로 하는 센서 미들웨어 구조

의 기능을 이용하여 서비스를 제공하며, 과거 정보를 필요로 하는 서비스를 요청할 경우 스냅샷을 가지는 다중 레벨 공간 DBMS에 저장된 스트림 데이터를 이용한다. 또한 저장 데이터 관리자 계층을 이용한 주기적인 저장 데이터 관리 알고리즘을 이용하여 저장된 스트림 데이터에 대한 재사용성을 유지한다. 시스템 내에서의 스트림 데이터 관리 과정은 크게 스트림 데이터의 데이터 베이스 저장 연산과 저장된 스트림 데이터에 대한 재사용성을 유지하기 위한 스트림 데이터 관리 연산으로 이루어진다. 다음 장에서는 시스템 내에서 처리되는 스트림 데이터의 저장과 관리 연산에 대해 알아보도록 한다.

3.2 스트림 데이터의 저장

RFID를 기반으로 유비쿼터스 환경의 응용 서비스를 지원하는 미들웨어는 지속적으로 끊임없이 입력되는 데이터를 정확하게 실시간으로 처리하고 응용 서비스에서 요구하는 결과를 획득해서 전달해야 한다. 본 논문의 다중 레벨 공간 DBMS를 기반으로 하는 센서 미들웨어는 끊임없이 입력되는 스트림 데이터를 처리하면서 사용자에게 확장된 서비스를 위한 일부의 스트림 데이터를 다중 레벨 공간 DBMS로 저장하는 연산을 수행한다. 이에 대한 스트림 데이터 저장 구조는 <그림 4>와 같다.

위의 그림은 미들웨어로 들어오는 스트림 데이터를 분석하고 일부의 저장 가치를 가지는 데이터를 스냅샷을 가지는 다중 레벨 공간 DBMS에 저장하는 시스템 구조를 보여준다. 시스템은 센서 데이터 관리자를 통하여 센서 데이터들을 입력 받는다. 입력 받은



<그림 4> 스트림 데이터 저장 구조

센서 데이터들은 스트림 데이터 래퍼를 통하여 미들웨어 내에서 분석 가능한 스트림 데이터 형태로 변환한 후, 스트림 데이터 필터링 관리자에서 사용자를 위한 데이터 필터링 과정을 거친다. 필터링 과정을 거친 데이터를 이용하여 확장 서비스 지원 관리자는 일부의 필터링된 스트림 데이터를 제공하여 사용자에게 서비스한다. 이렇게 사용자를 위한 서비스에 사용된 스트림 데이터들은 데이터에 대한 재사용성을 고려하여 스트림 데이터 전송 관리자에게 스트림 데이터를 전달한다. 그리고 해당 데이터의 재사용성에 관한 정보를 스트림 데이터 재사용 관리자에게 전달한 후, 다음 트리거 연산을 위해서 스트림 데이터를 삭제한다. 스트림 데이터를 전달받은 스트림 데이터 전송 관리자는 이 데이터를 스냅샷을 가지는 다중 레벨 공간 DBMS 계층의 저장 데이터 구분 관리자에게 전달한다. 스트림 데이터를 전달 받은 저장 데이터 구분 관리자는 디스크 데이터베이스에 이미 존재하는 데이터인지를 분석하고, 데이터베이스에 존재하지 않는다고 판단되면 해당하는 데이터를 디스크 데이터베이스에 저장한다. 그리고 이후의 해당 데이터에 대한 데이터 관리 연산을 위해서, 관련된 메타 데이터 정보를 저장 데이터 관리자로 전송한다.

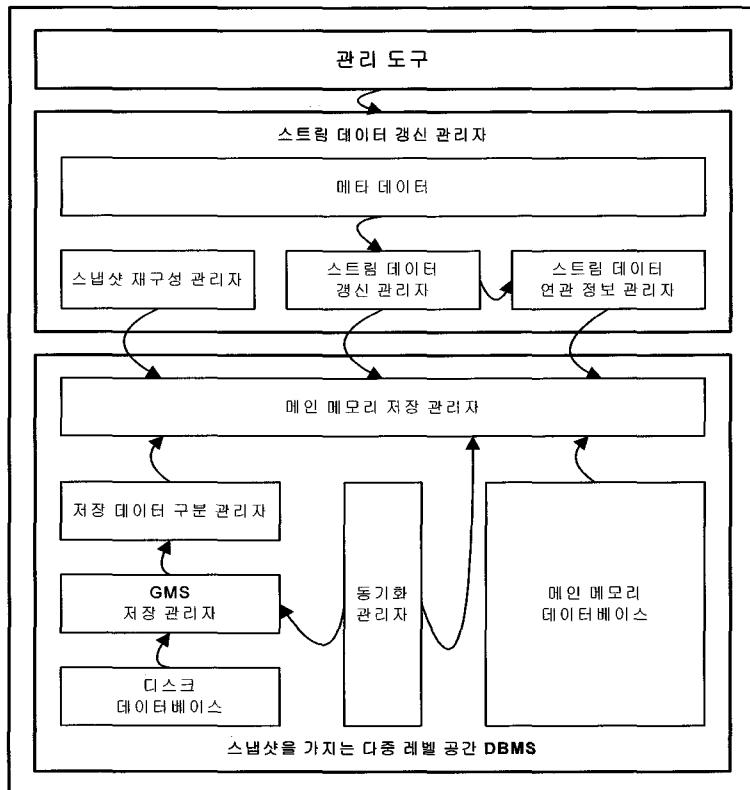
이러한 저장 데이터 구분 관리자의 저장 과정은 기존의 스냅샷을 가지는 다중 레벨 공간 DBMS에서의 저장 데이터 구분 관리자와 차이점을 가진다. 기존의 스냅샷을 가지는 다중 레벨 공간 DBMS는 특정 데이터에 대한 빠른 응답 속도를 사용자가 요구할 경우, 디스크 데이터베이스에서 저장된 해당 데이터를 메인 메모리 데이터베이스에서의 스냅샷으로 생성하고 이에 대한 서비스를 제공한다. 그러나 미들웨어는 사용자로부터 별도의 빠른 응답 속도나 빠른 처리 속도에 대한 요청을 받지 않으며, 오직 입력되는 스

트림 데이터를 가공하고 이를 사용자에게 서비스하는 기능을 가진다. 따라서 저장 데이터 구분 관리자는 해당 스트림 데이터에 대한 재사용성을 기준으로 저장된 스트림 데이터의 높은 저장 가치를 파악하고, 파악된 저장 순위를 이용하여 메인 메모리 데이터베이스의 스냅샷을 생성한다. 이러한 저장 스트림 데이터에 대한 관리 알고리즘은 다음 장에서 설명한다.

3.3 스트림 데이터의 관리

스냅샷을 가지는 다중 레벨 공간 DBMS는 디스크 데이터베이스에 저장된 데이터들을 기반으로, 사용자의 데이터 서비스 요청에 대한 높은 데이터 재사용성과 빠른 데이터 응답 속도를 유지하기 위해서 사용 빈도가 높은 데이터를 메모리 데이터베이스에 유지하며, 만약 스냅샷 데이터가 사용자의 서비스 요청에 대한 더 이상의 저장 가치를 가지지 않을 경우 해당 스냅샷을 효율적으로 교체한다. 그리고 디스크 데이터베이스에 저장된 스트림 데이터가 사용자에게 높은 재사용율을 보일 경우, 해당 데이터를 스냅샷의 형태로 메인 메모리에 저장한다. 본 시스템에서는 이러한 다중 레벨 공간 DBMS에서의 효율적인 스냅샷 관리 기법을 활용한 확장된 미들웨어 스트림 데이터 관리 알고리즘을 설계하였다.

본 시스템은 디스크 데이터베이스와 메모리 데이터베이스에서 스트림 데이터를 저장하고 관리한다. 디스크 데이터베이스는 미들웨어에서 필터링되고 이후의 재사용 가치가 고려되는 스트림 데이터들을 관리한다. 따라서 디스크 데이터베이스에 저장된 데이터는 메모리 데이터베이스에 저장된 데이터들보다 상대적으로 낮은 재사용 빈도를 가지며, 단지 확장된 데이터 서비스를 제공하기 위한 목적을 가지므로 잊은 스트림 데이터 재



<그림 5> 스냅샷 관리 구조

사용성을 분석할 필요가 없다. 이러한 이유로, 본 시스템에서는 기존의 다중 레벨 공간 DBMS에서와 같은 스트림 데이터 관리 알고리즘을 가지므로 설명을 생략한다[7, 11].

디스크 데이터베이스에 저장된 스트림 데이터들 중에서 사용자에 대한 높은 재사용 빈도를 가지는 메모리 스냅샷 데이터들은 그 빠른 서비스 데이터 응답 속도를 유지하기 위한 관리 알고리즘을 가진다. 이에 대한 스냅샷 관리 구조는 <그림 5>와 같다.

위 그림은 스냅샷을 가지는 다중 레벨 공간 DBMS에서 메인 메모리 데이터베이스에 저장된 스냅샷을 관리하는 구조를 나타낸다. 메타 데이터는 미들웨어 내에서 사용자에게 제공되는 서비스에 대한 정보와 서비스에

사용되는 스트림 데이터 정보를 유지한다. 스트림 데이터 간접 관리자는 주기적으로 메타 데이터에 기록된 스트림 데이터 정보를 분석하고, 사용자 서비스에 사용되는 스트림 데이터들의 재사용 가치에 따른 메인 메모리 스냅샷 관리 우선 순위를 결정한다. 만약 높은 데이터 재사용 순위를 가지는 스트림 데이터가 디스크 데이터베이스에 있을 경우, 해당 데이터를 스냅샷의 형태로 메인 메모리 데이터베이스에 저장하고 관리한다. 그리고 메타 데이터를 기반으로 한 분석에서 상대적으로 낮은 저장 순위를 가진다고 판단된 데이터가 메인 메모리에 스냅샷의 형태로 존재할 경우, 스트림 데이터 간접 관리자는 메인 메모리 저장 관리자에게 해당

서비스 요청 횟수	= 해당 스트림 데이터를 사용하여 사용자에게 제공된 서비스들의 횟수
최근 요청 횟수	= 요청 횟수 / (최근 참조 시간 - 데이터베이스에 저장된 시간)
서비스 연관 정보	= 서비스에 대한 해당 스트림 데이터의 기여도 (%)
서비스 변화폭	= 해당 스트림 데이터를 사용하는 서비스들의 분산

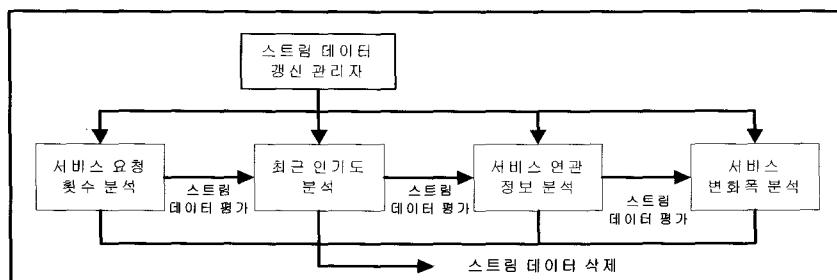
<그림 6> 스트림 데이터 관리를 위한 메타 데이터

스냅샷에 대한 삭제를 요청한다. 그 이외에 다중 레벨 공간 DBMS는 시스템 내에서의 지정된 스냅샷 생성 요청이나 관리자로부터의 특정에 대한 신속한 서비스 지원 요청과 같은 상황을 지원하기 위한 목적에 사용되는 관리도구, 그리고 시스템 셋다운과 같은 메인 메모리 데이터베이스에 저장된 스냅샷의 파손에 대한 복구 과정을 담당하는 스냅샷 재구성 관리자를 가진다. <그림 6>은 메모리 데이터베이스에서 스냅샷을 관리할 때, 스냅샷을 가지는 다중 레벨 공간 DBMS의 스트림 데이터 간접 관리자에서 데이터 재사용성을 분석하고 저장할 데이터베이스 계층을 결정할 때 필요한 메타 데이터 정보들을 나타낸다.

스냅샷을 가지는 다중 레벨 공간 DBMS는 디스크 데이터베이스와 메모리 데이터베이스에 저장된 각각의 데이터들을 관리하고 서로 다른 데이터베이스 간에 발생할 수 있는 데이터 일관성 문제를 해결하기 위하여

다양한 메타 데이터를 유지한다. 메타 데이터는 이러한 다중 레벨 공간 DBMS의 기본적인 데이터 관리 연산 이외에 미들웨어에서 요구하는 사용자에 대한 확장된 서비스를 제공하기 위하여, 미들웨어에서 사용자가 빈번하게 요구하는 서비스에 대한 요청 횟수, 요청된 서비스에 대한 최근성, 특정 서비스를 분석하는 데 필요되는 스트림 데이터의 개수, 그리고 단위 스트림 데이터를 필요로 하는 서비스의 분산 정도와 같은 스트림 데이터 정보를 추가적으로 관리한다. <그림 7>은 이러한 스트림 데이터 정보를 이용하여 스트림 데이터를 분석하는 알고리즘을 보여준다.

스트림 데이터 간접 관리자는 DBMS에 저장된 스트림 데이터의 사용자에 대한 효율적인 서비스를 유지하기 위해서, 미들웨어의 필터링 조건을 만족하는 스트림 데이터들에 대한 서비스 요청 횟수, 최근 인기도, 서비스의 변화폭, 그리고 서비스 연관 정보



<그림 7> 스트림 데이터 분석 알고리즘

```

while(스트림 데이터의 갱신 주기)
{
    do(스트림 데이터에 대한 메타 데이터 분석 & 해당 스트림 데이터에 대한 재사용 우선 순위를 분석);
    if(사용자의 서비스 요청에 사용되는 데이터)
    {
        do(메타 데이터 정보를 이용한 스트림 서비스의 재사용 우선 순위를 분석);
        do(메인 메모리 내에서 가장 낮은 재사용 우선 순위를 가지는 스냅샷과 비교);
        if(스냅샷보다 높은 저장 순위를 가지는 경우)
        {
            do(낮은 재사용 우선 순위를 가지는 스냅샷을 삭제);
            do(메인 메모리 데이터베이스에 해당 스트림 데이터에 대한 스냅샷을 생성);
            do(스냅샷 교체 정보를 메타 데이터에 저장);
        }
    }
}
}

```

<그림 8> 스트림 데이터 관리 알고리즘

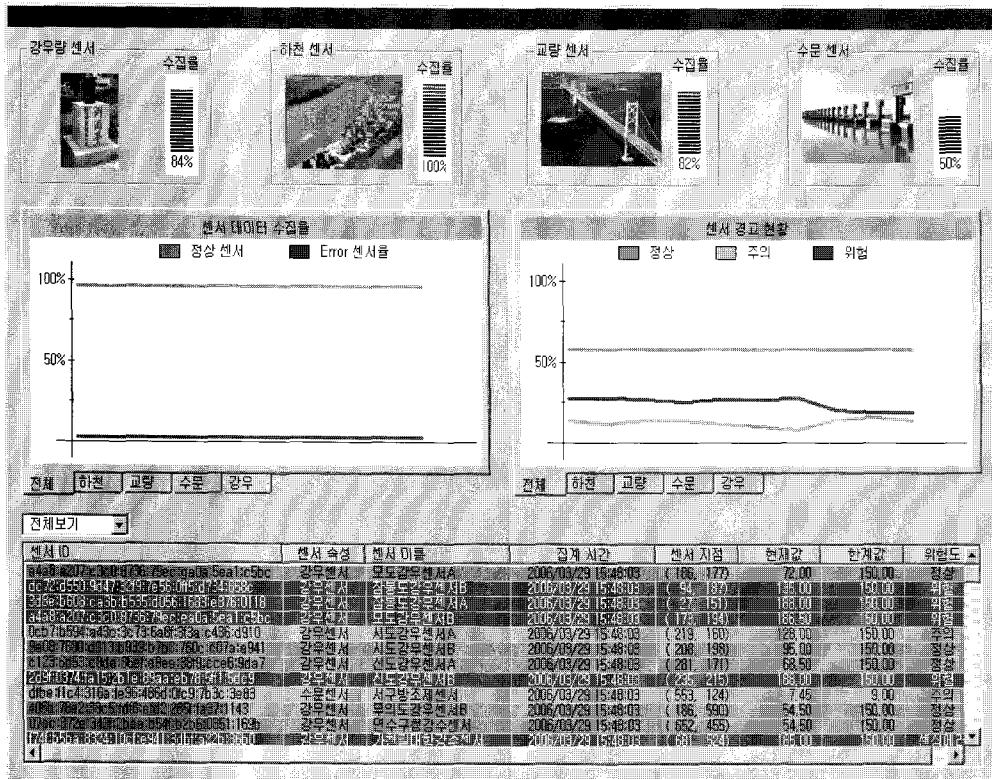
와 같은 데이터 특성을 순차적으로 분석한다. 우선, 미들웨어에서 빈번하게 서비스가 요청되거나 최근에 사용된 스트림 데이터는 가장 높은 재사용 가치를 가지므로 이에 대한 스트림 데이터를 평가한다. 그리고 사용자로부터 빈번하게 요청되는 서비스를 생성하는 데 특정 스트림 데이터가 기여하는 정도를 나타내는 서비스 연관 정보와, 특정 스트림 데이터가 사용되는 서비스들에 대한 변화 정도를 순차적으로 분석한다. 또한 스트림 데이터가 가지는 각각의 기준들에 대한 분석을 기반으로 메모리 데이터베이스에서의 높은 데이터 재사용성을 유지시킨다. 이러한 스트림 데이터 분석 알고리즘을 기반으로 본 시스템은 다음의 <그림 8>과 같은 스트림 데이터 관리 알고리즘을 가진다.

본 시스템은 메인 메모리 데이터베이스에 저장된 스트림 데이터에 대한 데이터 재사용성을 유지하기 위한 일정한 스트림 데이

터 갱신 주기를 가진다. 일정한 갱신 주기마다 스트림 데이터 갱신 관리자는 메타 데이터에 저장된 스트림 데이터들에 대한 정보 중에서, 필터링 조건을 만족하고 서비스 요청에 사용되는 데이터들에 대한 데이터 재사용성을 분석한다. 재사용성에 대한 분석 과정은 <그림 6>의 메타 데이터 목록을 <그림 7>의 분석 알고리즘을 거쳐 얻어지며, 이러한 분석 결과를 메인 메모리 데이터베이스에 저장되어 있는 스냅샷들과 비교하여 데이터 재사용 우선 순위가 높은 데이터를 메인 메모리에 스냅샷의 형태로 유지하는 알고리즘을 가진다.

4. 센서 미들웨어 구현

본 논문에서는 제안하는 시스템 구조를 사용하여 가상 센서로부터 다양한 스트림 데이터를 수집하고, 수집된 스트림 데이터를



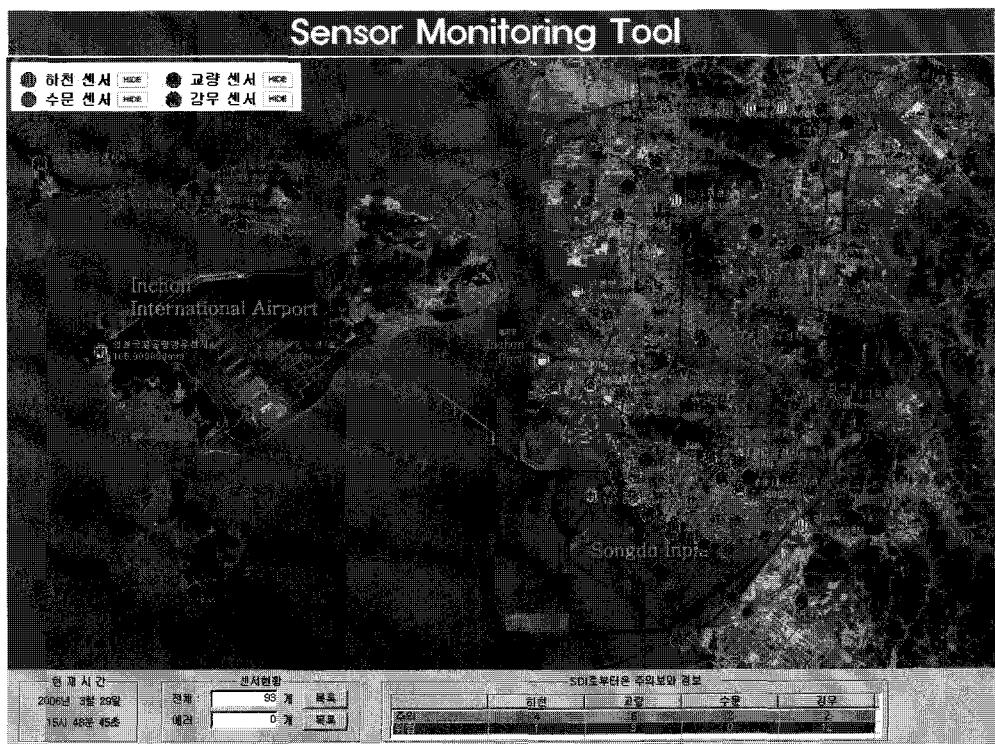
〈그림 9〉 구현된 시스템에서의 센서 데이터 정보

기반으로 확장된 사용자 서비스를 제공하는 센서 미들웨어 시스템을 구현하였다. <그림 9>는 본 시스템에서 관리하는 사용자 요청 데이터 정보를 보여준다. 시스템은 센서 ID, 센서 종류, 센서 이름, 집계 시간, 센서 지점, 한계값, 그리고 위험 정도와 같은 각각의 스트림 데이터 정보를 사용자에게 실시간으로 제공하며, 센서 데이터 수집율이나 센서 경고 현황과 같은 누적된 과거 정보가 필요한 서비스 요청에 대해서는 데이터베이스에 저장된 과거 강우량 정보들을 이용하여 통계 및 확률과 같은 다양한 형태로 사용자에게 서비스한다.

본 시스템은 폭우 주위보나 폭우 경보와
같이, 사용자에게 되도록 신속하게 제공되어
야 하는 중요 데이터가 있을 경우, 다양한

어플리케이션을 이용하여 관리하는 서비스 정보를 디스플레이한다. <그림 10>은 이러한 중요 정보들을 사용자에게 시각적으로 제공하는 센서 모니터링 툴을 보여준다. 센서 모니터링 툴은 스트림 데이터에서 제공하는 센서 ID 정보를 이용하여 맵상의 위치 정보를, 그리고 위험도와 측정 수치 등을 이용하여 해당 위치에서 가지는 위험 정도를 파악하고 이를 지도상에 표시한다.

본 시스템은 강우량에 대한 현재 정보를 서비스하는 기능과 함께, 디스크 데이터베이스에 축적된 강우 정보들을 분석하여 시간의 변화에 따른 센서 데이터 수집율, 그리고 센서 경고 상황 등을 사용자에게 시각적으로 제공하였다. 또한 빈번하게 위협이 예측되는 지역에 대한 정보를 메인 메모리 테이



〈그림 10〉 구현된 시스템에서의 센서 모니터링 툴

터베이스에 저장함으로서, 사용자가 알아야 하는 중요 정보를 보다 신속하게 서비스하였다.

5. 결론 및 향후 연구

본 시스템은 기존의 미들웨어에서 문제가 되었던 과거 스트림 데이터를 이용하여 얻을 수 있는 확률 및 통계 데이터에 대한 서비스 요청, 그리고 과거 정보에 대한 빈번한 서비스 요청을 해결하였으며, 사용자에게 확장된 데이터 서비스를 제공하기 위해서 다중 레벨 공간 DBMS를 이용하여 스트림 데이터를 저장하고 분석하는 미들웨어 구조를 제안하였다. 본 시스템은 저장 가치가 높은 데이터를 우선적으로 저장하여 외부로부터의 다양한 서비스 요청에 대응하고, 사용자

로부터 빈번하게 요구되는 일부의 스트림 데이터에 대한 재사용 우선 순위를 분석하여 재사용 순위가 높은 스트림 데이터를 메인 메모리 데이터베이스에 스냅샷으로 관리함으로서 사용자에게 신속한 데이터 서비스를 제공한다. 또한 메인 메모리 데이터베이스에 저장된 스냅샷 형태의 스트림 데이터가 가지는 데이터 재사용성을 주기적으로 분석하고 더 이상의 저장 가치를 가지지 않는 스냅샷 데이터를 간신함으로서, DBMS에서 관리하는 데이터에 대한 높은 일관성과 재사용 가치를 유지하였다. 그리고 이러한 본 논문의 시스템 구조를 가상 센서 데이터를 이용한 어플리케이션으로 실험한 결과, 기존의 다른 미들웨어들보다 좋은 기능을 보여주었다.

향후 연구 과제로는 미들웨어의 외부 환

경 변화에 효율적으로 대응하면서 부하를 관리하는 시스템 부하 관리자, 미들웨어로 입력되는 스트림 데이터의 패턴을 효율적으로 분석할 수 있는 패턴 인식 알고리즘, 그리고 스냅샷을 보다 효율적으로 관리하기 위한 메모리 데이터베이스에서의 스냅샷 관리 기법에 대한 연구가 필요하다[12, 13, 14].

참고문헌

1. Arvind Arasu, Brian Babcock, "STREAM: The Stanford Data Stream Management System", IEEE Data Engineering Bulletin, Stanford, 2003.
2. Tobin J. Lehman, Eugene J. Shekita, Luis-Felipe Cabrera, "An Evaluation of Starburst's Memory Resident Storage Component", Transactions on Knowledge and Data Engineering, IEEE, Vol.4, No.6, 1992, pp. 555-556.
3. Christian Floerkemeier, Matthias Lampe, "Middleware: RFID middleware design : addressing application requirements and RFID constraints", International Conference Proceeding Series. ACM, Vol.121, 2005, pp. 219-224.
4. 정원일, "유비쿼터스 컴퓨팅을 위한 센서 데이터 스트림 처리 기술", 한국전자통신연구원, 대전, 2005.
5. 천종현, 어상훈, 김호석, 배해영, "스냅샷 데이터를 갖는 다중레벨 공간 DBMS 설계 및 구현", 추계학술발표회 논문집, 제32권, 제2호, 2005, pp. 217-219.
6. James J. Lu, Guido Moerkotte, Joachim Schue, V. S. Subrahmanian, "Efficient maintenance of materialized mediated views", International Conference on Management of Data.
7. 박상근, 박순영, 정원일, "GMS: 공간 데이터베이스 관리 시스템", 2003 공동 춘계학술대회, 2003, pp. 217-224.
8. 주봉, 어상훈, 김명근, "스냅샷 데이터를 갖는 다중 레벨 저장 시스템에서의 효율적인 리프레시 기법", 한국정보처리학회, 제12권, 제1호, 2005, pp. 55-58.
9. 장석규, 어상훈, 김명근, "스냅샷 데이터를 갖는 다중레벨 저장 DBMS에서 성능 향상을 위한 질의 분류 방법", 데이터베이스연구회, 2005, pp. 121-126.
10. Ralf Hartmut Guting, "An Introduction to Spatial Database Systems", 1994, VLDB, pp. 357-399.
11. Michel J. Garey, David J. Dewitt, Joel E. Richardron, "Object and File Management in the EXODUS Extensible Database System", VLDB, 1986, pp. 357-383.
12. Ossama Othman, Douglas C. Schmidt, "Issues in the Design of Adaptive Middleware Load Balancing", ACM SIGPLAN, 2001, pp. 205-213.
13. Manuel Roman, Nayeem Islam, "Dynamically programmable and reconfigurable middleware services", Middleware Conference", Vol.78, 2004, pp. 372-396.
14. María Jesús Garzaran, Milos Prvulovic, José María Llabería, "Tradeoffs in buffering speculative memory state for thread-level speculation in multi-processors", Transactions on Architecture and Code Optimization (TACO). ACM, Vol.2, 2005, pp. 247-279.

오은석

2002년 인하대학교 컴퓨터공학부(공학사)

2004년~현재 인하대학교 대학원

컴퓨터 · 정보공학과(석사과정)

관심분야: 다중 레벨 공간 DBMS, RFID 미들웨어

김호석

2001년 인하대학교 전자계산공학과(공학사)

2003년 인하대학교 대학원

전자계산공학과(공학석사)

2003년~현재 인하대학교 대학원

컴퓨터 · 정보공학과(박사과정)

관심분야: 공간데이터베이스, GIS, LBS

김재홍

1985년 인하대학교 전자계산공학과(공학사)

1990년 인하대학교 전자계산공학과(공학석사)

1994년 인하대학교 전자계산공학과(공학박사)

1995년~현재 영동대학교 컴퓨터공학과 교수

관심분야: 멀티미디어 데이터베이스, 이동 객체

데이터베이스, 지리정보 시스템

배해영

1974년 인하대학교 응용물리학과(공학사)

1978년 연세대학교 대학원 전자계산학과(공학석사)

1989년 숭실대학교 대학원 전자계산학과(공학박사)

1985년 Univ. of Houston 객원교수

1992년~1994년 인하대학교 전자계산소 소장

1982년~현재 인하대학교 컴퓨터공학부 교수

1999년~현재 지능형GIS연구센터 센터장

2000년~현재 중국 중경우전대학교 대학원 명예교수

2004년~현재 인하대학교 정보통신대학원 원장

관심분야: 분산 데이터베이스, 공간 데이터베이스,

지리정보 시스템, 멀티미디어

데이터베이스 등