

LBS를 위한 실시간 GIS 엔진의 설계 및 구현†

윤재관⁰ 김동오 한기준
건국대학교 컴퓨터·정보통신공학과
(jkyun⁰, dokim, kjhan)⁰@db.konkuk.ac.kr

Design and Implementation of a Real-Time GIS Engine for LBS

Jae-Kwan Yun⁰ Dong-O Kim Ki-Joon Han
Dept. of Computer & Information Communication Engineering, Konkuk University

요 약

최근 무선 인터넷과 핸드폰, PDA, HPC 등과 같은 모바일 장치의 발전으로 인하여 이동체의 위치를 중심으로 하는 LBS가 점차적으로 많은 분야에서 사용되고 있다. 이동체의 위치 데이터는 GPS와 같은 위치 측정 시스템을 이용하여 대용량으로 발생하게 되는데 기존의 정적인 데이터를 처리하기 위한 GIS를 사용하여 이러한 데이터를 처리하는 것은 비효율적이다. 그러므로, 본 논문에서는 이러한 동적인 데이터를 효과적으로 처리할 수 있는 LBS를 위한 실시간 GIS 엔진을 설계 및 구현하였다. 본 논문에서의 실시간 GIS 엔진은 LBS에서 필요한 공간, 비공간, 위치 데이터를 효과적으로 관리할 수 있다. 실시간 GIS 엔진은 인터페이스 관리자, 질의 관리자, 인덱스 관리자, 데이터 로더, 레이어 관리자, 위치 데이터 관리자, 메타 데이터 관리자, 객체 관리자 등으로 구성되어 있으며 기존의 HP 실시간 데이터베이스 시스템에 공간 데이터 타입, 공간 연산자, 가변 길이 데이터 처리, 테이블 자동 확장 기능 등을 추가하여 LBS에서 사용되는 대용량 가변 길이 데이터를 효과적으로 처리할 수 있도록 하였다.

1. 서론

최근 컴퓨터의 대용량화 및 고성능화에 따라 GIS에 근거한 다양한 형태의 응용 시스템이 연구 및 개발되고 있다. 이전까지의 이러한 연구 및 개발은 주로 GIS에 대한 데이터베이스의 측면을 기준으로 많이 진행되었지만, 무선 인터넷과 핸드폰, PDA(Personal Digital Assistants), HPC(Hand Held PC) 등과 같은 모바일 장치의 발전으로 인하여 점차적으로 LBS(Location Based Service)를 중심으로 발전하고 있다[1, 8, 10]. 이러한 LBS에서는 지도 데이터와 같은 정적인 데이터의 처리보다는 차량의 움직임이나 개인의 위치 파악과 같은 동적인 데이터의 처리가 매우 중요하다. 이러한 동적인 데이터를 빠르게 처리하고 신속하게 대응하기 위해서는 디스크를 이용하는 기존의 GIS보다는 주기억장치를 이용하는 실시간 GIS 엔진의 사용이 보다 더 효과적이다[2, 9].

이를 위해서 본 논문에서는 기존의 실시간 데이터베이스 시스템인 HP 실시간 데이터베이스 시스템에 공간 데이터 타입, 공간 연산자, 가변 길이 데이터 처리, 자동 레코드 개수 확장 등의 기능을 추가하고, 주기억장치에 데이터를 적재하는 데이터 로더를 개발하여 LBS를 위한 실시간 GIS 엔진을 설계 및 구현하였다.

본 논문의 구성은 다음과 같다. 제 1장의 서론에 이어 제 2장의 관련 연구에서는 LBS와 HP 실시간 데이터베이스 시스템에 대하여 살펴본다. 제 3장에서는 시스템의 설계에 대하여 살펴보고, 제 4장에서는 시스템의 구현에 대하여 설명한다. 그리고, 5장에서 결론 및 향후 연구과제에 대하여 언급한다.

2. 관련 연구

본 장에서는 LBS에 대하여 소개하고, 본 논문에서 확장한 HP 실시간 데이터베이스 시스템에 대하여 설명한다.

2.1 LBS 소개

일반적으로 LBS는 이동 통신망을 기반으로 하여 사람이나 사물의 위치를 정확하게 파악하고 이를 활용하는 응용 시스템 및 서비스라고 정의된다. 3GPP(3rd Generation Partnership Project)의 TS22.071에서는 LBS를 위치 기반의 응용 제공이 가능한 네트워크를 이용한 표준화된 서비스라고 정의하고 있다[10]. 그리고, OGC(OpenGIS Consortium)에서는 LBS를 위치 정보에 접속, 제공

또는 위치 정보에 의해 작용하는 모든 응용 소프트웨어 서비스라고 정의하고 있으며[3, 4], FCC에서는 이동식 사용자가 그들의 지리적 위치, 소재, 또는 알려진 존재에 대한 서비스를 받도록 하는 것이라고 정의하고 있다.

일반적인 의미에서의 LBS는 이동통신 기술의 발달과 함께 휴대폰, PDA, HPC 등 휴대용 단말기를 이용하여 위치를 추적하고, 위치와 관련된 정보를 제공하는 유/무선 단말기의 진보된 서비스라고 할 수 있다. 이러한 LBS를 구축하기 위해서는 최첨단 위치 결정 기술, 위치 정확도 향상 기술, 무선 인터넷 위치 처리 기술, 공간 데이터 처리 기술, LBS 플랫폼 기술, LBS 응용 소프트웨어 개발 기술, 개방형 GIS 및 관련 표준화 기술, LBS 응용 서비스 개발 기술, LBS 데이터베이스 기술 등이 필요하다[7].

2.2 HP 실시간 데이터베이스 시스템

HP에서 개발한 주기억장치 데이터베이스 시스템인 HP 실시간 데이터베이스 시스템은 메모리내에서의 고성능 데이터 저장 및 검색을 지원함으로써 연산들에 대한 예측 가능한 빠른 응답시간을 보장한다[6]. 여러 프로세스들의 동시 접근이 가능하도록 데이터베이스가 공유 메모리에 상주하기 때문에 여러 개의 프로세스가 한 데이터베이스에 접근할 수 있고, 또한 한 프로세스가 여러 데이터베이스에 접근할 수 있다. 특히, 3단계의 패스워드에 의한 보안과 2가지의 잠금을 지원하며 유연성 및 확장 가능성이 뛰어난 특징이 있다.

HP 실시간 데이터베이스 시스템의 실시간 스케줄링은 운영체제의 우선 순위 선점 스케줄링에 기반하여 프로세스 차원에서 관리되며, 프로세스 메모리 잠금과 공유 메모리 잠금을 이용해 동시성 제어를 하고 있다. 또한, B-Tree 인덱스와 Hash 인덱스를 제공하고 있다. HP 실시간 데이터베이스 시스템에서의 데이터베이스는 다중 사용자가 데이터베이스를 접근하는 모드인 공유(shared) 모드와 한 프로세스만 데이터베이스 관리자 암호로 접근할 수 있는 배타(exclusive) 모드를 제공한다. 그리고, 시스템 정보가 공유 메모리에서 관리되기 때문에 여러 프로세스가 하나의 데이터베이스를 효과적으로 사용할 수 있다. 또한, 레코드나 열들 중의 하나의 값 또는 입력 영역(Input Area) 내에 값의 변화가 있으면 발생하는 이벤트를 지원하며, 메모리에 있는 데이터를 확일로 출력하는 언로드, 확일에 있는 데이터를 메모리에 적재하는 로드 유틸리티를 제공하고 있다.

본 논문에서는 현재 상업적인 용도로 사용되고 있는 HP 실시간 데이터베이스 시스템을 확장하여 LBS에서 사용되는 위치 데이터 및 공간 데이터를 효율적으로 관리할 수 있도록 하였다.

† 본 연구는 한국과학재단 목적기초연구(과제번호: R01-2001-00540) 지원으로 수행되었음.

3. 시스템 설계

본 장에서는 실시간 GIS 엔진의 구동 환경, 실시간 GIS 엔진의 구성, 그리고 지원하는 공간 데이터 타입에 대하여 설명한다.

3.1 시스템의 구동 환경

일반적으로 디스크를 사용하는 기존의 GIS와는 달리 주기억장치 데이터베이스 시스템에서는 한번에 주기억장치에 적재할 수 있는 양이 제한되어 있기 때문에 대용량의 지도 데이터를 모두 적재하기에는 문제가 있다. 본 논문에서는 기존의 GIS를 백엔드(Back-end)로 사용하고 데이터 로더를 이용하여 실시간 GIS 엔진에 데이터를 적재한다.

실시간 GIS 엔진은 주기억장치의 용량에 저장할 수 있는 데이터의 양이 많은 영향을 받기 때문에 하나의 시스템이나 프로세스에 저장할 수 있는 데이터의 양을 작은 규모로 처리한다. 즉, 작은 규모의 실시간 GIS 엔진을 여러개 분산하여 구동한 후에 각 실시간 GIS 엔진과 데이터 통신을 수행하여 질의를 처리한다. 본 논문에서의 실시간 GIS 엔진의 구동 환경은 그림 1과 같다.

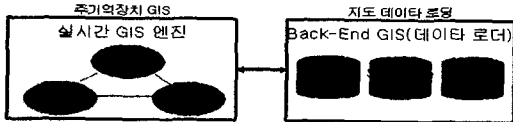


그림 1. 실시간 GIS 엔진의 구동 환경

3.2 시스템의 구성

본 논문에서 개발한 실시간 GIS 엔진은 그림 2와 같이 인터페이스 관리자, 질의 관리자, 인덱스 관리자, 데이터 로더, 레이어 관리자, 메타 데이터 관리자, 객체 관리자의 7개 관리자로 구성되어 있다.

인터페이스 관리자는 서로 다른 시스템에서 서로 다른 사용자들이 실시간 GIS 엔진에 접근할 수 있도록 API를 제공한다. 질의 관리자는 공간, 비공간, 위치 데이터를 처리하기 위한 공간 연산을 제공한다. 인덱스 관리자에서는 비공간, 위치 데이터의 검색을 위해서는 B-tree, Hash 인덱스를 사용하고, 공간 데이터에 대한 인덱스는 R-tree와 R*-tree를 사용한다. 메타 데이터 관리자는 테이블에 대하여 속성 정보, 객체에 대한 정보를 관리한다. 객체 관리자는 공간, 비공간, 위치 데이터를 삽입, 갱신, 삭제, 그리고 검색하는 기능을 수행한다. 생성되는 객체는 공유 메모리에 저장되며 메타 데이터 관리자에서 객체에 대한 offset, length, 데이터 타입 등을 관리하게 된다. 데이터 로더는 주기억장치에 공간, 비공간, 위치 데이터를 필요할 때마다 적재하여 주는 기능을 한다. 그리고, 레이어 관리자는 공간 및 위치 데이터에 중요도를 부여하고 관리하기 위하여 사용된다.

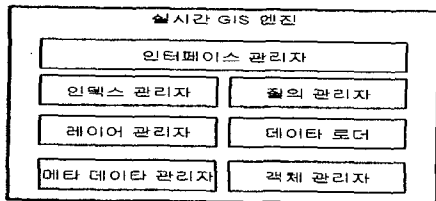


그림 2. 실시간 GIS 엔진의 시스템 구조도

3.3 공간 데이터의 타입

본 논문에서는 공간 데이터를 표현하기 위하여 그림 3과 같은 공간 데이터 타입을 제공하고 있다. 이중에서 RD_POINT, RD_LABEL, RD_LINE과 같은 공간 데이터 타입은 고정 길이를 가지며 주로 위치 데이터를 표현하기 위하여 많이 사용된다. 그리고, RD_LINestring, RD_POLYGON은 공간 데이터를 표현하기 위하여 사용되며, 가변 길이 데이터이기 때문에 가변 길이 처리 방법을 통하여 처리된다.

공간 데이터 타입	이름	타입	설명
RD_POINT	x	double	x 좌표
	y	double	y 좌표
RD_LABEL	p	RD_POINT	레이블의 위치
	str	char *	문자열
RD_LINE	v0l	RD_POINT	시작점
	v1l	RD_POINT	끝점
	m	double	길이
RD_LINestring	pts	RD_POINT *	(x0l, y0l ... x(npts-1), y(npts-1))
	npts	RD_POINT *	개수
RD_POLYGON	pts	RD_POINT *	(x0l, y0l ... x(npts-1), y(npts-1))
	npts	long	개수

그림 3. 공간 데이터 타입

4. 시스템 구현

본 장에서는 LBS를 위한 실시간 GIS 엔진의 구현 내용에 대하여 설명한다.

4.1 가변 길이 처리

본 논문에서 제공하는 공간 데이터 타입 중 고정 길이 데이터인 경우에는 테이블이 생성될 때 그 열의 크기가 결정된다. 그러나, 가변 길이 데이터인 경우에는 포함된 점의 개수에 따라서 열의 길이가 결정된다. 즉, 데이터의 입력이 발생된 후에야 열의 길이를 알 수 있게 된다. 실시간 GIS 엔진은 HP 실시간 데이터베이스 시스템에서 지원되지 않았던 이와 같은 가변 길이 데이터의 처리를 위해 다음과 같은 방법을 사용한다.

우선 실시간 GIS 엔진에서 공간 데이터 타입이 있는 테이블이 생성될 경우 입력 영역이라고 하는 공유 메모리 영역이 새로이 생성된다. 그리고, 메타 데이터 관리자에서는 입력되는 가변 길이 데이터에 대한 offset과 length 정보를 유지 관리한다. 이와 같은 방법을 사용하게 되면 실제 가변 길이 데이터는 별도로 저장되기 때문에 비공간 데이터에 대한 접근만이 일어날 경우에는 실제 공간 데이터에 대한 접근이 필요하지 않으므로 좀 더 빠른 데이터 검색을 수행할 수 있게 된다. 그림 4는 가변 길이 처리에 대한 그림을 보여준다.

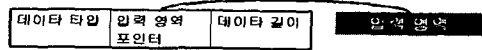


그림 4. 가변 길이 데이터의 처리

4.2 자동 레코드 개수 확장

기존의 HP 실시간 데이터베이스 시스템에서는 테이블을 생성할 경우에 테이블에 저장될 수 있는 레코드의 개수를 미리 정해 놓은 다음 그 이상의 데이터가 저장될 경우에 오류를 발생시켰다. 그러나, 위치 데이터와 같이 주기적으로 생성되는 데이터는 최대 레코드의 개수를 미리 알 수 없기 때문에 레코드의 개수가 지정된 값을 초과할 경우에 자동으로 할당된 개수가 증가하도록 처리하는 것이 필요하다. 본 논문에서는 이를 지원하기 위해 다음과 같은 자동 레코드 개수 확장 방법을 사용하였다.

4.2.1 테이블 재생성 방법

테이블 재생성 방법은 데이터의 삽입 시 지정된 레코드의 개수를 넘어갈 경우에 확장된 크기의 새로운 테이블을 생성하고 기존 테이블에 있는 모든 데이터를 새로운 테이블에 입력하는 방법이다. 이 방법을 사용할 경우 실제 실시간 GIS 엔진의 성능이 떨어지는 경우는 지정된 레코드 개수를 넘는 한 개의 레코드가 추가되는 시점에만 발생하게 된다. 처음 테이블을 생성할 때 레코드 개수 지정을 크게 하면 많은 데이터가 복사되기 때문에 새로운 테이블의 생성이 오래 걸린다는 단점이 있지만 발생하는 간격이 적기 때문에 큰 데이터의 삽입에 효과적이다.

4.2.2 체이닝(Chaining) 방법

체이닝 방법은 데이터의 삽입 시 지정된 레코드의 개수만큼 새로운 메모리 공간을 할당하여 새로운 데이터를 추가한 후 링크로 두 개의 테이블을 연결하는 방법이다. 이 방법은 지정된 레코드의 개수를 초과하는 시점에 데이터를 복사하지 않기 때문에 삽입에

걸리는 시간이 짧다는 장점을 가지고 있다. 그러나, 많은 체인으로 연결된 경우 검색 시간이 오래 걸리고, 인덱스를 구성하기가 어려운 단점이 있다.

4.3 실시간 GIS 엔진의 주요 함수

실시간 GIS 엔진에서 제공하고 있는 함수는 HP 실시간 데이터베이스 시스템의 API를 확장하여 구현되었다. 본 논문에서는 ANSI C를 기반으로 하여 구현되었으며 주요 함수는 그림 5와 같다.

```

int RdConnectDatabase(), int RdDisconnectDatabase()
int RdCreateTable(), int RdDropTable()
int RdCreateSpatialIndex(), RdCreateSpatialIndex()
int RdASpatialInsert(), RdASpatialDelete()
int RdASpatialUpdate(), RdASpatialUpdate()
int RdASpatialDelete(), RdASpatialDelete()
int RdASpatialUpdate(), RdASpatialUpdate()
void *RdGetFieldNum(), int RdGetRecordNum()
void *RdGetASpatialValue(), RD_POINT *RdGetSpatialValue()
RdIterator *GetRecord()
    
```

그림 5. 실시간 GIS 엔진의 주요 함수

RdConnectDatabase, RdDisconnectDatabase 함수는 외부 인터넷에서 실시간 GIS 엔진의 데이터베이스에 연결하기 위하여 사용된다. 데이터베이스에 연결하기 위해서는 데이터베이스 이름, 관리자 암호가 필요하며 성공적으로 연결되었을 경우에는 세션 번호와 공유 메모리 ID를 얻게 된다. 세션 번호는 데이터베이스의 모든 객체에 접근하기 위해서 반드시 필요하다. RdCreateTable, RdDropTable 함수는 실시간 GIS 엔진에서 새로운 테이블을 생성하거나 기존 테이블을 제거하기 위해서 사용된다. 새로운 테이블을 생성하기 위해서는 테이블 이름, 열 정보, 테이블 ID가 필요하다.

RdCreateASpatialIndex, RdCreateSpatialIndex 함수는 실시간 GIS 엔진에서 생성된 테이블에 대하여 비공간, 공간 인덱스를 생성하기 위해서 사용된다. RdASpatialInsert, RdASpatialDelete, RdASpatialUpdate 함수는 비공간 데이터를 한 테이블의 특정 레코드에 대하여 삽입, 삭제, 갱신을 하기 위해서 사용된다. 그리고, RdSpatialInsert, RdSpatialDelete, RdSpatialUpdate 함수는 공간 데이터에 대한 삽입, 삭제, 갱신을 하는 함수이다. RdGetFieldNum 함수는 테이블의 열 항목의 개수 정보를 알려주며, RdGetRecordNum은 검색된 데이터의 레코드 개수를 반환한다. RdGetASpatialValue는 현재 선택된 레코드의 비공간 데이터에 대한 값을 얻을 수 있고, RdGetSpatialValue는 공간 데이터에 대한 값을 얻을 수 있다. GetSpatialRecord 함수를 통하여 반환된 반복자를 통하여 선택된 공간 레코드의 위치를 얻을 수 있다.

4.4 성능 평가

본 논문에서는 LBS를 위한 실시간 GIS 엔진을 개발한 후 이에 대한 성능 평가를 실시하였다. 성능 평가는 1GHz 듀얼 CPU, 1GB의 메모리, 512MB의 공유 메모리를 설정한 시스템에서 공간 데이터의 삽입, 검색, 삭제, 그리고 갱신 작업에 대하여 수행하였으며 기존의 디스크 기반 시스템(ORACLE, ZEUS, PostgreSQL)과 전체 테이블 검색(Full Table Scan)을 수행하는 경우에 소요되는 시간을 측정하였다. 성능 평가를 하기 위해서 사용된 데이터는 200개의 점을 포함하고 있는 폴리곤을 사용하였다. 그림 6과 7은 폴리곤의 수가 증가함에 따라 공간 데이터의 삽입과 검색에 소요된 시간을 보여준다.

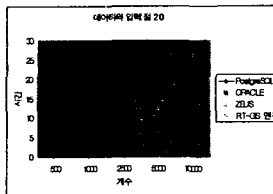


그림 6. 공간 데이터의 삽입

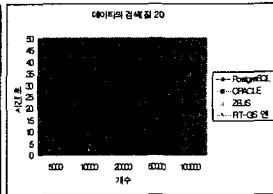


그림 7. 공간 데이터의 검색

위의 결과에서도 볼 수 있듯이 본 논문에서 개발한 실시간 GIS 엔진을 사용할 경우 기존의 GIS에 비하여 공간 데이터의 처리 속도가 매우 신속하게 이루어짐을 알 수 있다. 즉, 이러한 실시간 GIS 엔진은 공간 데이터의 요청이 많은 시스템이나 LBS와 같이 공간 및 위치 데이터의 삽입, 검색, 삭제, 갱신이 빈번하게 일어나는 시스템에 적합하다.

5. 결론

최근 PDA, HPC와 같은 모바일 장치의 급속한 발전과 무선 인터넷의 사용이 증가함에 따라 GPS를 이용한 LBS에 대한 관심이 점차적으로 확대되고 있다. 이러한 추세에 따라 국내외적으로 이동 통신 업체를 중심으로 하여 위치 추적, 경로 안내, 지역 안내 등과 같은 실생활에 밀접한 서비스를 제공하고 있다. 이와 같은 LBS에서 사용되는 위치 데이터는 기존의 GIS에서 사용되었던 변화가 적은 대용량의 정적인 데이터보다는 특정 시간에 갱신이 빈번한 동적인 데이터가 주로 사용되므로 이를 반영할 수 있는 시스템이 필요하다. 이를 위해서 본 논문에서는 동적인 데이터를 효과적으로 처리할 수 있는 LBS를 위한 실시간 GIS 엔진을 설계 및 구현하였다.

본 논문에서의 실시간 GIS 엔진은 LBS에서 필요한 공간, 비공간, 위치 데이터를 효과적으로 관리한다. 실시간 GIS 엔진은 인터넷 관리자, 질의 관리자, 인덱스 관리자, 데이터 로더, 레이어 관리자, 위치 데이터 관리자, 메타 데이터 관리자, 객체 관리자로 구성되어 있으며 기존의 HP 실시간 데이터베이스 시스템의 기능에 공간 데이터 타입의 추가, 공간 연산자 추가, 가변 길이 데이터 처리 기능 추가, 테이블 자동 확장 기능 등을 추가하여 대용량 가변 길이 데이터를 효과적으로 처리할 수 있도록 구현되었다. 본 논문의 향후 연구 과제로는 실시간 GIS에서 처리되는 위치 데이터의 검색 속도를 더욱 향상시키기 위한 새로운 클러스터링 알고리즘의 개발이다.

참고 문헌

- [1] Barbara, D., "Mobile Computing and Database-A Survey," IEEE Trans. on Knowledge and Data Eng., Vol.11, No.1, 1999, pp. 108-117.
- [2] Gueting, R.H., "An Introduction to Spatial Database Systems," The VLDB Journal, Vol.3, No.4, 1994, pp. 357-399.
- [3] Niedzwiedek, H., "OpenLS-1 Interoperability Project: An Overview," LBS 포럼 창립총회 및 기념세미나, 2002, pp.30-51.
- [4] OpenLS Initiative, A Request for Technology In Support of an Open Location Services(OpenLS™) Testbed, <http://www.openls.org>, 2000.
- [5] Prakash, R., and Singhal, M., A Dynamic Approach to Location Management in Mobile Computing Systems, Dept. of Computer and Information Science, The Ohio State Univ., Technical Report, OSU- CISRC-4/96-TR22, 1996.
- [6] 신명철, 박인하, 이강준, 한기준, "HP 실시간 데이터베이스 시스템을 위한 회복 시스템," 1999년도 한국 데이터베이스 학술대회 논문집, 15권1호, 1999.2. pp.74-81.
- [7] 오승, 이창진, 김창호, "위치기반서비스 분야의 표준화 동향 분석," 2001 개방형 지리정보시스템 학회 학술회의 논문집, 4권1호, 2001, pp.123-141.
- [8] 윤재관, 이근호, 한기준, "Shareware DBMS를 이용한 모바일 GIS의 개발," 개방형GIS연구회 논문지, 2권2호, 2001, pp.25-37.
- [9] 진희재, 박상미, 안병익, "위치기반정보서비스를 지원하는 시스템 구조 및 소프트웨어 기술동향 분석," 2001 개방형 지리정보시스템 학회 학술회의 논문집, 4권1호, 2001, pp.145-160.
- [10] 최혜옥, "위치기반서비스 (LBS, Location-Based Services)," 제3회 공간정보 워크샵, 2002년, pp.5-22.