

클러스터링 환경에서 효율적인 공간 질의 처리를 위한 로드 밸런싱 기법의 설계 및 구현

김종훈[†] · 이찬구^{**} · 정현민^{***} · 정미영^{****} · 배영호^{*****}

요 약

웹 GIS에서 인터넷 서비스 이용자의 집중 현상으로 발생하는 서버의 과부하 현상을 막기 위한 대표적인 방법으로 클라이언트와 서버가 모두 질의에 참여하는 하이브리드(Hybrid) 질의 처리 방식이 있다. 그러나 하이브리드 질의 처리 방식은 서버 확장에 제약이 존재하기 때문에 근본적인 해결책이 되지 못한다. 따라서 웹 GIS 서버의 안정적인 서비스 제공을 위해서는 웹 클러스터링 기술의 도입이 필요하다. 본 논문에서는 웹 GIS 클러스터링 시스템을 위한 질의 영역의 인접성을 이용한 로드 밸런싱 기법을 제안한다. 제안하는 기법은 공간 데이터를 관리하는 타일을 기반으로 인접한 타일 그룹을 생성하여 각 서버에 할당하며, 질의 영역 및 공간 연산을 고려하여 서버에서 질의가 처리되는 동안 버퍼 재사용율이 최대가 되도록 클라이언트의 질의 요청을 서버에 전달한다. 제안하는 기법은 서버의 버퍼를 공간 인덱스 탐색에 최적화함으로써 서버의 버퍼 재사용율을 높이고, 클러스터링 시스템에서 디스크의 접근 횟수를 낮추어 전체적인 서버 시스템의 처리 능력을 향상시킨다.

Design and Implementation of Load Balancing Method for Efficient Spatial Query Processing in Clustering Environment

Kim Jong Hoon[†], Lee Chan Gu^{**}, Junh Hyun Meen^{***},
Jeong Mi Young^{****} and Bae Young Ho^{*****}

ABSTRACT

Hybrid query processing method is used for preventing server overload that is created by heavy user connection in Web GIS. In Hybrid query processing method, both server and client participate in spatial query processing. But, Hybrid query processing method is restricted in scalability of server and it can't be fundamentally solution for server overload. So, it is necessary for Web GIS to be brought in web clustering technique. In this thesis, we propose load-balancing method that uses proximity of query region. In this paper, we create tile groups that have relation each tile in same group is very close, and forward client request to the server that can have maximum rate of buffer reuse with considering characteristic of spatial query. With our load balancing method, buffer in server is optimized for exploring spatial index tree and increase rate of buffer reuse, so it can be reduced amount of disk access and increase system performance.

Key words: 공간 데이터, GIS, 클라이언트-서버 환경

접수일 : 2002년 8월 12일, 완료일 : 2002년 12월 23일

[†] 정희원, (주)케이지아이 대표이사

^{**} (주)케이지아이 연구원

^{***} (주)케이티 서비스개발연구소 무선망설계연구실장

^{****} (주)케이티 서비스개발연구소 선임보안연구원

^{*****} (주)케이티 서비스개발연구소 무선망연구실

1. 서 론

인터넷 환경에서 사용자의 요구사항에 대하여 빠르게 접근하여 서비스를 하는 것은 매우 중요하다. 특히 사용자들이 과도하게 몰릴 경우, 서버에 발생하는 네트워크 트래픽 증가와 서버 리소스 사용량의 급격한 증가로 인한 서비스 중단 현상이 심각한 문제로 지적되었다. 특히 지리 정보 시스템(Geographic Information System:GIS) 등 대용량의 데이터를 다루는 분야에서 이러한 문제는 더욱 심각하다.

인터넷 환경에서 운영되는 지리 정보 시스템인 웹 GIS 분야에서 이러한 과부하 문제를 해결하기 위해 클라이언트와 서버 양 측에서 질의를 나누어 처리하는 연구가 진행되고 있다. 그러나 이러한 구조는 사용자 증가에 따른 서버의 과부하에 의한 서비스 중단 현상을 근본적으로 해결하지는 못한다. 따라서 웹 GIS의 서비스 중단 현상을 해결하기 위한 웹 클러스터링 시스템[1,2] 도입이 필요하지만 공간 질의의 특성으로 인하여 웹 클러스터링 시스템을 웹 GIS에 바로 적용할 수는 없다.

즉, 일반적인 웹 데이터는 클라이언트의 요구사항 간에 밀접한 연관성을 가지지 않으며 한개의 요청에 의해 서버에서 반환되는 리소스 역시 그리 많지 않게 된다. 즉 공간 질의는 인접한 영역에 대한 질의가 매우 빈번하게 발생하고, 공간 질의에 대해 결과값이 매우 크고 많기 때문에 각 서버에서 처리하는 질의에 대한 지역적 인접성이 낮아지게 된다. 이는 서버에서 공간 질의를 처리하기 위하여 탐색해야 하는 공간 인덱스의 범위가 넓어지게 되며 결과적으로 버퍼 재사용율을 낮추게 되는 원인이 된다. 서버의 버퍼 재사용율이 낮아지게 되면 디스크로 접근하여 페이지를 가져오는 빈도가 높아지게 되며, 많은 양의 공간 객체를 처리해야 하는 공간 질의에서는 질의 처리 비용이 증가하게 된다.

버퍼의 재사용율을 높여 디스크의 접근 횟수를 낮추기 위해 로컬 네트워크를 이용한 서버의 버퍼 공유 기법이 제안되었다[3,4]. 그러나 이러한 방법은 공간 질의 처리를 위한 디스크 접근 횟수는 낮아지지만 각 서버의 단일 버퍼 재사용율은 여전히 낮기 때문에 공유 버퍼의 참조 빈도가 높아지게 된다. 이는 잦은 버퍼 페이지 전송을 유발시켜 네트워크 트래픽을 증가시키는 원인이 된다.

공간 질의는 인접 영역에 대한 질의 처리가 잦은

특성을 갖기 때문에 질의가 처리되는 영역을 참조하여 처리 영역이 인접한 질의들을 그룹화하여 동일 서버에서 처리한다면 해당 서버가 탐색해야 하는 공간 인덱스의 범위가 좁아져 버퍼의 재사용율을 높일 수 있다[5,14].

본 논문에서는 로드 밸런싱을 이용하여 웹 GIS 클러스터링 시스템의 버퍼 재사용율을 높이고 디스크 접근 빈도를 낮추기 위한 기법을 제안한다.

제안하는 기법은 공간 인덱스를 구성하는 기법 중의 하나인 타일 기반 인덱스의 기본 단위인 타일을 기반으로 클러스터링에 참여한 서버의 구성에 따라 인접한 타일 집합을 생성한다. 생성된 타일 집합은 로드밸런싱을 제공하는 디스패처(Dispatcher)에 의해서 각각의 서버로 할당되며 디스패처는 클라이언트의 요청으로부터 공간 질의가 처리되는 영역에 대한 정보와 질의어에 포함되어 있는 공간 연산을 참조하여 해당 질의가 처리되는 동안 버퍼의 반환 빈도가 최소가 되도록 클라이언트 요청을 각 서버에 전달한다. 이를 위하여, 본 논문에서는 인접 영역 질의의 그룹핑 방법 및 질의 요청을 서버에 할당하는 방법을 사용한다.

2. 관련 연구

2.1 버퍼 공유 관리 기법

기존의 클러스터링 시스템에서는 버퍼의 재사용율을 높이기 위한 방법으로 로컬 네트워크를 통하여 각 서버들이 버퍼를 공유하는 방법을 이용한다[4]. 이는 디스크의 접근 빈도를 낮출 수 있기 때문에 클러스터링 시스템에서 발생하는 디스크 병목 현상을 해결하는데 도움을 준다[3].

버퍼 공유 기법에서는 클라이언트의 요구에 대한 페이지가 서버 상에 존재하지 않을 경우 해당 페이지의 소유자인 서버에게 요청하게 된다. 소유자는 해당 페이지가 버퍼 상에 존재하는 다른 서버를 찾아 버퍼 포워딩(Forwarding)을 명령하는데, 만일 어떠한 서버에도 페이지가 존재하지 않을 경우 소유자가 디스크에서 읽어 들인다. 단, 이 페이지는 외부 서버의 요청에 의하여 읽은 페이지이기 때문에 버퍼 반환 시에 가장 먼저 선택되게 되는데 이를 Hated Page라 한다[7].

이와는 별도로 각 서버에서 읽어 들인 페이지인

단일 페이지와 공유 버퍼로부터 전송받은 페이지인 중복 페이지 중 반환할 버퍼를 선택하는 방법에 따라 Client-Server(CISv) 방식과 Duplicate Elimination (Dup-Elim) 방식으로 나눌 수 있다[3].

CISv 방식은 단일 페이지와 중복 페이지의 구별 없이 단지 LRU 정책에 의해서 페이지를 반환하기 때문에 구현이 간단하며 단일 서버가 Hot 상태가 될 경우 버퍼의 Hit율이 높아지는 특징이 있지만 전체적으로 서버간에 중복되는 페이지 때문에 버퍼의 활용율이 떨어진다.

반면 Dup-Elim 방식은 CISv와는 달리 디스크 액세스 비용과 원격 서버로의 액세스 비용 차이를 고려하는 방법이다. 원격 메모리로부터 중복 페이지를 복사하는 것이 디스크 접근 비용보다 높기 때문에 Dup-Elim 방식에서는 단일 페이지의 반환에 앞서 모든 중복 페이지를 삭제하고 이에 따라 버퍼에 많은 페이지를 저장한다.

그러나 이러한 방법들은 공통적으로 Hated Page에 대해서는 가장 높은 순위로 페이지 반환을 수행하기 때문에, 서버의 로컬 디스크로부터 읽어 들인 페이지가 아닐 경우 버퍼에 존재하는 확률이 낮아지게 된다. 이러한 특성 때문에 공간 연산을 처리할 경우 오히려 공유 버퍼의 참조 빈도가 높아져 빈번한 네트워크 전송으로 인한 네트워크 트래픽이 증가하게 된다.

2.2 타일 기반 공간 인덱스

기존 웹 GIS의 초기 전송 지연 문제를 해결하기 위하여 시스템 초기에 불필요한 데이터의 전송을 방지하고 클라이언트가 필요로 하는 영역의 데이터만 전송하기 위하여, 전체 영역을 일정한 크기의 작은 영역인 타일(Tile)로 분할하여 서버와 클라이언트간의 전송 단위로 타일을 사용하는 관리 구조가 제안되었다. 이 구조에서는 타일 단위로 분할된 공간 데이터에 대한 효과적인 질의 처리를 위하여 타일기반의 공간 인덱스를 구축하고 관리한다[15]. 타일 기반 공간 인덱스는 각각의 타일별로 구축된 공간 인덱스와 타일에 대해 구축된 공간인덱스로 구분된다[16].

공간 데이터를 타일 단위로 분할하면 클라이언트와 서버간의 효율적인 데이터의 전송이 이루어진다. 일반적인 공간 인덱스는 공간 객체를 검색하기 위해

각 노드의 동일 레벨 노드 검색이 필요한데 타일 기반 공간 인덱스에서는 이러한 문제점이 발생하지 않는다.

2.3 힐버트 곡선

그리드로 분할되어 있는 2차원 공간의 좌표를 1차원 공간의 좌표로 매핑하기 위하여 Z-Curve, Gray Coding, 힐버트 곡선(Hilbert Curve)과 같은 연구들이 진행되어 왔다[6,8].

Z-Curve와 Gray code 기법은 공간 탐색 과정에서 점프 현상이 발생하며 이로 인해 2차원 공간의 인접 좌표가 연속적인 선형 좌표로 매핑되지 않는 경우가 발생한다. 그러나 힐버트 곡선은 그림 1과 같이 좌표 공간 탐색에서 컬럼의 점프는 오직 바로 옆에 존재하는 컬럼으로만 이루어지기 때문에 2차원 공간의 인접 좌표가 선형 좌표에서 멀어지는 현상을 최소화한다[6].

힐버트 곡선의 특징은 힐버트 곡선의 각 노드에 할당되어 있는 힐버트 값에 따라서 정렬을 수행할 경우 2차원의 인접된 공간 좌표가 선형 좌표에서도 근사 영역으로 매핑된다는 것이다[9]. 힐버트 곡선의 이러한 특징은 기존의 공간 데이터베이스에 관한 연구에서 선형 공간으로 매핑된 공간 객체들을 그룹핑하거나 역 클러스터링(De-clustering)을 수행함으로써 R-Tree 인덱스나 공간 데이터 클러스터링 시스템에서 병렬성을 높이기 위한 방법으로 이용되어져 왔다[6,10,11].

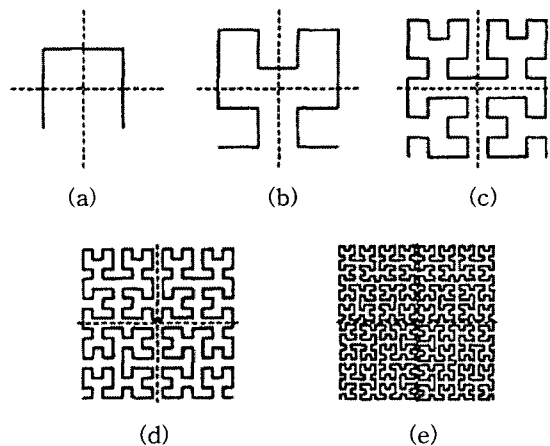


그림 1. 힐버트 곡선

3. 질의 영역의 인접성을 이용한 로드 밸런싱 기법의 설계

3.1 인접 영역의 그룹핑

2차원의 공간 좌표를 인접된 좌표의 관계를 유지한 채 1차원 좌표로 매핑하기 위해서는 공간이 그리드로 분할되어 있어야 한다. 클라이언트와 서버간의 전송 단위가 파일 전체이거나 레이어 단위로 전송을 하는 웹 GIS의 경우 디스패처는 서버로부터 공간 데이터의 영역에 관한 정보를 얻고 그리드 분할을 위한 기준을 정하여 공간 분할을 하여야 한다.

제안하는 시스템에서는 웹 GIS 서버에서 관리하는 공간 데이터에 관한 메타 데이터가 디스패처에 전달되며 디스패처는 이 정보를 바탕으로 전체 영역을 타일로 분할하고 인접 타일을 그룹핑한다.

힐버트 곡선은 $n \times n$ 차원의 정방형 공간에 대해 적용하는 기법이기 때문에 $n \times m$ 차원에 대해서는 인접 공간 정렬을 적용할 수 없다. 따라서 기존의 힐버트 곡선을 변형하여 $n \times m$ 차원에서도 힐버트 곡선을 이용한 인접 공간의 정렬이 필요하다[6].

그림 2는 기존의 힐버트 곡선과 변형된 힐버트 곡선을 보여주고 있다. 그림 2의 (a)는 Order가 4인 힐버트 곡선이며, (b)는 x차원이 4y인 공간에서 Order가 3인 힐버트 곡선을 x차원에 4번 반복한 변형된 힐버트 곡선이다.

힐버트 곡선을 이용하여 차원 매핑을 수행할 경우 힐버트 값을 계산하기 위한 기준점의 선정 방법에는 4-d Hilbert through corners(4D-xy), 4-d Hilbert through center and diameter(4D-cd), 2-d Hilbert through Centers Only(2D-c) 등이 있다.

읽기 연산만을 고려할 때 2D-c 방법을 이용하여 R-Tree를 정렬한 경우가 다른 방법을 이용하여 정렬한 경우보다 페이지의 접근 회수가 낮으며 R-tree의 응답 시간이 가장 짧다[10]. 따라서 본 논문에서는 그리드의 중심 좌표를 이용하여 힐버트 값을 계산한다.



(a)

(b)

그림 2. 변형된 힐버트 곡선

디스패처가 클러스터링에 참여하는 서버에 질의 처리 영역을 할당하는 과정은 다음과 같다.

1. 디스패처는 서버로부터 공간 데이터베이스의 메타 정보를 전달 받는다.
2. 서버의 공간 데이터의 타일 관리 구조와 일치하도록 전체 영역을 타일 크기로 분할한다.
3. 타일 개수에 맞추어 변형된 힐버트 곡선의 계산식을 보정한다.
4. 각 타일의 중심점을 이용하여 각 타일의 힐버트 값을 계산하고 오름차순으로 정렬한다.
5. 정렬한 타일을 클러스터링에 참여한 서버의 개수만큼 나누어 질의 처리 영역을 생성한다.
6. 질의 처리 영역을 각 서버에 할당하고 각 타일과 타일이 포함되어 있는 질의 처리 영역의 관계 테이블을 작성한다.

3.2 웹 GIS의 공간 질의어 분류

클라이언트의 질의 요청을 서버에 효과적으로 분배하기 위해서는 질의 영역에 대한 참조뿐만 아니라 질의 처리기에 의해 참조되는 영역에 대해서도 반드시 고려해야만 한다.

공간 질의는 지도 브라우징 질의, 레이어 제어 질의, 점 질의, 영역 질의로 분류되는데, 본 논문에서는 질의 처리기에서 참조하는 영역과 질의 영역과의 관계에 따라 다음과 같은 3가지 질의 클래스를 정의한다.

1. 영역 내부 질의 : 질의 처리 시 질의 영역을 경계로 영역 내부에 포함된 타일에 대해서만 참조가 일어나며 영역 외부의 타일에 대해서는 전혀 참조하지 않는다. 영역 내부의 타일에 존재하는 공간 객체에 대해서는 질의어의 조건을 만족하는지의 여부를 검사하기 위한 정제 과정이 필요하다.
2. 영역 외부 질의 : 타일 내부 질의와는 반대로 질의 영역 내부에 포함되지 않는 타일에 관한 연산으로 질의 영역 내부의 타일에 대한 참조는 전혀 일어나지 않는다.
3. 영역 경계 질의 : 질의 처리기는 질의 영역의 내부나 외부의 타일들에 대해서는 전혀 참조하지 않으며 오직 질의 영역 경계에 존재하는 타일만을 참조한다.

영역 내부 질의와 영역 외부 질의는 질의 영역이 동일하다더라도 실제로 질의를 처리하는 과정에서 참조하는 영역이 다르기 때문에 질의 영역만을 고려하여 서버에 전달할 경우 영역 외부 질의에 의해 버퍼 반환이 발생하기 때문에 버퍼의 효율이 떨어지게 된다. 그러므로 버퍼의 Hit 율을 높이기 위해서 영역 외부 질의는 질의 영역에 해당하는 영역의 질의 처리를 할당 받은 서버 이외의 서버에서 처리하는 것이 효과적이다.

영역 경계 질의는 일반적으로 영역 내부 질의와 동일한 규칙을 적용할 수 있다. 그러나 영역 내부 질의와는 달리 질의 영역 내부에 포함되는 타일은 전혀 참조하지 않기 때문에 질의 영역과 서버에 할당된 처리 영역과의 관계를 고려하여 서버에 할당해야 한다.

다음의 표 1은 공간 질의를 질의 영역에 포함되는 타일의 개수와 정의한 질의 클래스로 분류한 것이다.

3.3 로드 밸런싱을 위한 질의 처리

3.3.1 질의 처리 서버의 선정

단일 타일 질의 영역은 항상 질의 처리 영역에 포함된다. 그러나 복수 타일 질의 영역은 경우에 따라서 2개 이상의 질의 처리 영역에 걸쳐서 나타날 수 있다. 그림 3은 각 서버에 할당된 질의 처리 영역과 이 질의 처리 영역에 대해 요청된 질의 영역에 대한 예를 보이고 있다. 현재 공간 데이터베이스는 5개의 지리 처리 영역으로 분할되어 있으며, 각각의 영역은 1개의 서버에서 맡아 처리하고 있다.

회색의 사각형은 질의 영역을 의미하며 영역 C, D, F는 단일 타일 질의 영역이며, 영역 A, B, E, G는 복수 타일 질의 영역이다. 영역 A는 질의 처리 영역

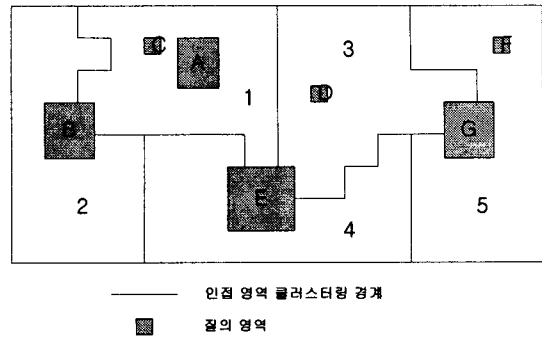


그림 3. 질의 영역과 질의 처리 영역

1에 완전히 포함되어 있으며, 영역 B는 질의 처리 영역 1과 2에 중복되어 있다.

영역 C, D, F에 대한 영역 내부 질의는 각각 질의 영역을 포함하고 있는 질의 처리 영역 1, 3, 5로 전달되어 처리된다. 동일한 질의 영역에 대해 영역 외부 질의는 질의 영역 외부의 타일에 대한 접근 빈도가 높기 때문에 질의 처리 영역의 버퍼 효율을 위해서 질의 영역을 포함하지 않는 질의 처리 영역을 선택하는 것이 좋으나 단일 타일 질의 영역은 질의 영역의 크기가 1개의 타일에 불과하기 때문에 어떤 질의 처리 영역에서 처리하여도 버퍼에 대한 영향은 동일하다. 때문에 단일 타일에 대한 영역 외부 질의는 영역 내부 질의와 같이 질의 영역이 포함되어 있는 질의 처리 영역을 선택하여 처리하도록 한다.

복수 타일 영역인 A, B, E, G는 질의 처리기가 버퍼를 가장 효율적으로 사용할 수 있는 질의 처리 영역을 선택해야 한다. 즉, 질의 영역의 중심점을 구한 후 중심점에 대한 힐버트 값을 구하여 이 값을 포함하는 질의 처리 영역에서 질의를 처리하도록 할당

표 1. 공간 질의 분류 테이블

질의영역	질의 클래스	공간 질의	세부 질의
단일 타일	영역 내부 질의	점 질의	점 질의
		영역 질의	Meet, Overlap, Covers, Covered_by, Inside, Contains, Equal
	영역 외부 질의	영역 질의	Disjoint
복수 타일	영역 내부 질의	지도 브라우징	지도 확대, 축소, 이동, 영역 확대, 끌기
		레이어 제어	레이어 추가
	영역 질의	Covers, Contain, Equal	
	영역 경계 질의	영역 질의	Meet, Overlap, Covered_by, *Inside
	영역 외부 질의	영역 질의	Disjoint

한다.

그림 4의 (a)와 같이 질의 처리 영역의 경계 (1)과 (2)가 직선으로만 질의 영역을 가로지른다면 중심점을 포함하고 있는 질의 처리 영역이 질의 영역의 많은 부분을 할당 받을 수 있다. 그러나 (b)와 같이 직선이 아닌 형태로 질의 영역을 분할하게 되면 중심점을 포함하고 있음에도 중심점을 포함하지 않는 영역이 질의 영역의 많은 공간을 차지하게 된다. 또한, 영역 경계 질의의 경우 영역 내부 질의와는 다른 특성을 갖기 때문에 단순히 중심점을 이용하여 질의 영역을 질의 처리 영역에 할당하는 것은 효과적이지 못하다.

따라서 본 논문에서는 질의 영역을 구성하고 있는 타일들을 계산하여 타일들을 가장 많이 포함하고 있는 질의 처리 영역을 선택하는 방법을 취한다. 동일한 개수의 타일을 포함하는 질의 처리 영역이 2개 이상 존재할 경우는 임의의 영역을 선택한다. 그림 5의 (a), (b), (c)의 음영 부분으로 처리된 타일이 각각 영역 내부 질의, 영역 경계 질의, 영역 외부의 경우 질의 처리 영역을 결정하기 위한 타일이다.

영역 내부 질의의 경우 영역 내부 타일을 가장 많이 포함하고 있는 질의 처리 영역을 구하기 위해 질의 영역을 포함하고 있는 모든 타일을 검색한다. 영역 경계 질의의 경우 질의 영역 경계 이외의 타일에

대해서는 접근할 필요가 없기 때문에 질의 영역 경계를 포함하고 있는 타일에 대해서만 검색한다. 영역 외부 질의의 경우 힐버트 값을 계산해야 할 타일은 영역 내부 질의와 같다. 그러나 영역 내부 질의와는 반대로 질의 영역이 포함하고 있는 타일을 전혀 포함하지 않거나 포함하는 타일의 개수가 최소인 질의 처리 영역을 선택한다.

영역 외부 질의는 다른 클래스의 질의보다 질의 처리기가 참조하는 영역의 크기가 매우 크기 때문에 버퍼에 미치는 영향 또한 매우 크다. 제안하는 기법에서는 영역 외부 질의를 처리하기 위한 질의 처리 영역의 선정에 의하여 질의 영역에 포함되는 타일의 개수뿐만 아니라 각 서버의 버퍼 사용량을 이용한다.

버퍼 사용량의 정확한 측정은 힘들기 때문에 질의 처리 영역과 해당 영역에서 처리되었던 질의 영역의 크기를 통해서 추정하도록 한다. 버퍼 사용량의 추정은 디스패처는 질의 영역을 서버로 할당할 때 각 질의 처리 영역에서 처리된 질의 영역의 타일 개수를 기록한다. 이 정보로부터 각 서버에서 처리된 질의 영역과 질의 처리 영역의 비율을 계산할 수 있으며 이 비율이 가장 낮은 서버를 버퍼 사용량이 가장 낮은 서버로 가정하며 이 서버에 영역 외부 질의를 전달한다.

3.3.2 질의 처리 영역 분할

제안하는 기법은 전체 영역을 타일로 분할한 후 인접 타일들을 그룹핑하여 질의 영역과 질의 처리 영역의 포함관계를 이용하여 할당하여 클라이언트의 질의 요구를 처리하기 때문에 특정 영역에 대한 클라이언트의 질의가 폭증할 경우 해당 질의 처리 영역을 담당하는 서버에 과부하가 걸려 시스템의 성능이 저하될 수 있다.

클러스터링 시스템의 부하 불균형 문제를 해결하기 위한 방법은 대표적으로 라운드-로빈(Round-Robin) 스케줄링을 이용하여 클라이언트의 요청을 임의의 서버에 전달하는 방법과 각 서버간의 메시지 전달을 통해서 서버의 디스크와 CPU의 부하 정보를 파악하거나 서버의 부하를 계산하여 부하가 높은 서버의 프로세스를 부하가 낮은 서버로 이전시키는 방법 등이 있다[12,13]. 그러나 제안하는 기법은 서버에서 처리하게 되는 질의 처리 영역이 구분되어 있기 때문에 기존의 방법으로 부하 균등화를 수행할 경우

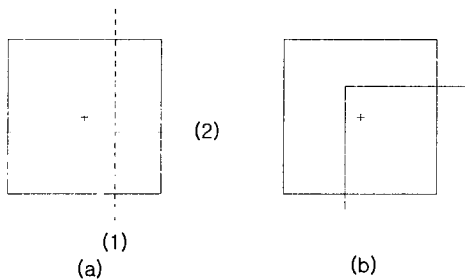


그림 4. 질의 영역

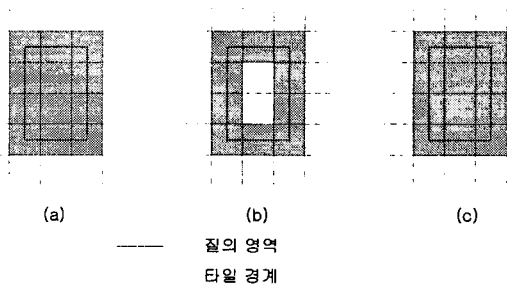


그림 5. 질의 영역과 타일 공간

버퍼에 존재하는 페이지의 영역 인접성이 떨어져 버퍼 효율이 낮아지는 문제가 발생할 수 있다. 따라서 본 논문에서는 질의 요청 빈도에 따라 각 타일에 가중치를 두어 각 서버에 할당하는 질의 처리 영역에 포함되는 타일 개수를 다르게 함으로써 특정 서버로 질의가 집중되는 것을 분산시킨다.

시스템의 초기 구축 시에 각 타일은 가중치 1의 값을 가지며, 각 질의 처리 영역은 동일한 개수의 타일을 배분 받는다. 디스패처는 질의 영역에 해당하는 타일의 개수를 기록할 뿐만 아니라 각 타일의 질의 처리 횟수와 질의 처리 영역의 질의 처리 횟수를 기록한다. 질의 처리 영역의 질의 처리 비율이 65%를 넘었을 경우 질의 요청이 과도하게 집중된 것으로 규정하여 재분배 작업을 수행하는 것이 가장 효율적이다[2].

다음은 질의 처리 영역 A의 부하율 임계점을 구하는 계산식이다. 식의 우 항은 영역 A에 할당된 타일의 비율이며, 좌 항은 영역 A가 처리하는 질의의 비율이다. 타일의 할당 비율에 비해 질의 처리 비율이 65%를 넘어섰을 경우 영역 A는 과부하 임계점에 도달한 것이다.

$$\frac{\text{영역 A의 질의 처리 횟수}}{\text{전체 질의 처리 횟수}} \geq \frac{\text{질의 처리 영역의 타일 개수}}{\text{전체 타일 개수}} \times \frac{3}{2} \quad (\text{식3-1})$$

질의 처리 영역 재분배 수행은 첫째로 각 타일의 가중치 값을 (타일의 질의 처리 횟수/전체 질의 처리 횟수)로 초기화 하며 전체 타일 개수 Tottile을 타일 i의 가중치로 초기화 한다. 이때 각 질의 처리 영역에 할당되는 타일의 개수 Numtile은 (Tottile 클러스터링 서버의 개수)이며, 질의 처리 영역의 타일 할당은 각 타일의 가중치 합이 Numtile이 될 때까지 반복 수행한다.

그림 6의 (a)는 4개의 서버로 구성된 클러스터링 시스템의 초기 구축 시에 각 타일에 가중치 1을 부여하고 영역을 분할한 것이다. 디스패처는 표 2와 같은 각 타일에 대한 질의 처리 정보를 기록한다. 이 경우 타일 m, n, o, p가 포함된 질의 처리 영역은 전체 질의 처리 비율이 40.4%로 서버 부하율 임계점인 37.5%를 넘었다. 따라서 질의 처리 영역 재분배를 위하여 각 타일에 대한 가중치를 계산하고 새로운 타일 할당 개수인 25에 근접하도록 그림 6의 (b)와 같이 질의 처리 영역을 분할한다. 타일 가중치에 의한 재분배 방

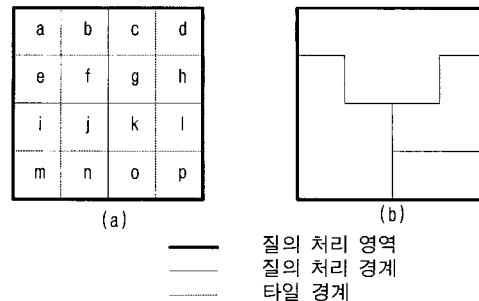


그림 6. 타일 가중치 재분배

표 2. 타일 메타 정보

영역	질의 횟수	질의 처리 비율(%)	타일 가중치
a	8	8.5	9
b	1	1.1	1
c	5	5.3	5
d	3	3.2	3
e	6	6.4	6
f	3	3.2	3
g	3	3.2	3
h	4	4.3	4
i	7	7.4	7
j	6	6.4	6
k	3	3.2	3
l	7	7.4	7
m	9	9.6	10
n	10	10.6	10
o	9	9.6	10
p	10	10.6	11

법에 의하여 질의 처리 빈도가 높은 영역은 서로 다른 서버로 재분배되며 질의 처리 빈도가 낮은 서버는 하나의 질의 처리 서버로 통합된다.

4. 로드 밸런싱을 지원하는 디스패처의 구현

본 장에서는 질의 영역의 인접성을 고려한 디스패처 구현에 대해 기술한다. 다음 그림 7은 전체 시스템 구조도이다.

먼저 클라이언트의 질의 영역을 소켓 접속 시점에 파악하기 위하여 URL을 이용한 질의 영역을 전달한다. 클라이언트의 URL을 바탕으로 질의 영역을 추출하고, 클라이언트의 요청을 그림 8과 같이 클러스터링 서버로 전달하기 위한 통신 관리기, 공간 질의어

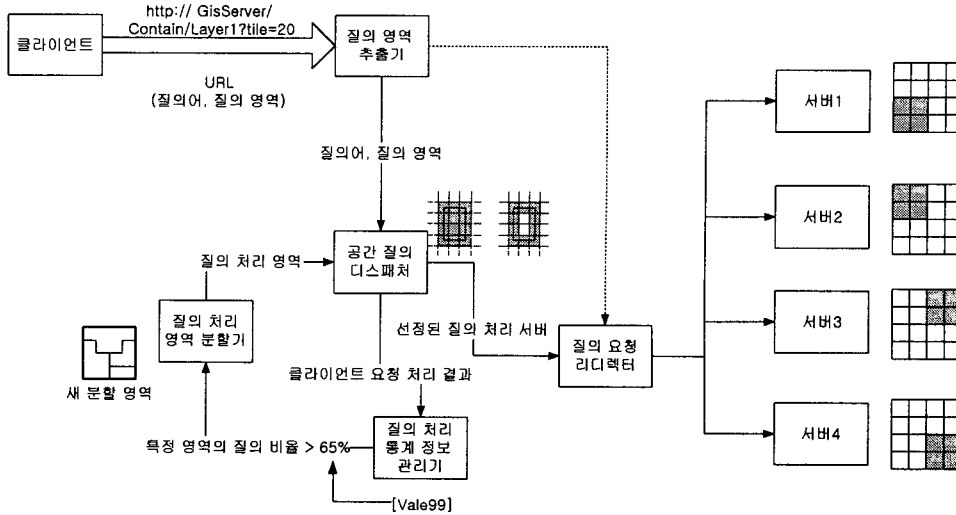


그림 7. 시스템 구조도

및 질의 영역을 바탕으로 질의 요청을 처리하기 위한 서버를 선택하는 공간 질의 디스패처, 클러스터링 서버의 효과적인 로드 밸런싱을 위한 질의 처리 통계 정보 관리기, 질의 처리 영역의 타일을 인접 공간으로 분할하는 질의 처리 영역 분할기에 대해 다룬다.

리할 서버를 선정하여 통신 관리기로 넘겨준다. 공간 질의 디스패처로부터 질의 처리 서버를 넘겨받으면 클라이언트의 요청을 해당 질의 처리 서버로 전달한다. 그림 9는 이러한 통신 관리기에 대한 전반적인 구조도이며 알고리즘 1은 통신 관리기에 대한 구현 알고리즘이다.

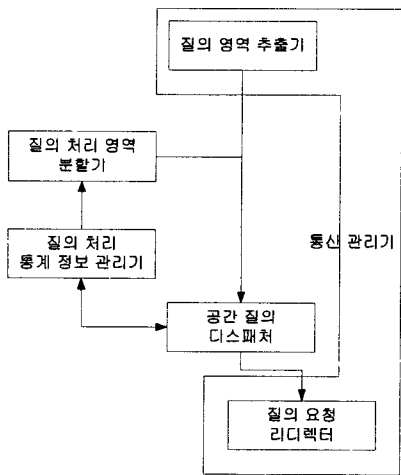


그림 8. 디스패처 컴포넌트

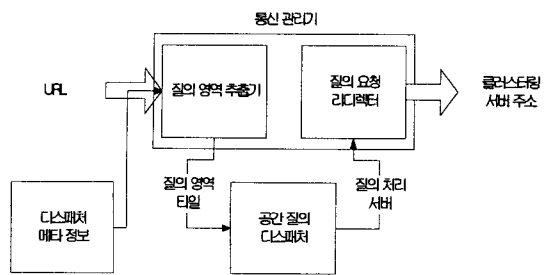


그림 9. 통신 관리기

4.1 통신 관리기

통신 관리기는 클라이언트에서 요청한 URL로부터 질의 영역 정보를 추출하여 질의 영역에 대응되는 타일을 계산한다. 계산된 타일은 공간 질의 디스패처로 전달되며, 공간 질의 디스패처는 질의 요청을 처

알고리즘 1 : 통신 관리기

Input : Web GIS 클라이언트의 URL
Output : 클러스터링 서버의 주소

```
String Redirector(String URL)
{
    URL로부터 레이어 정보를 추출
    URL로부터 공간 질의어를 추출
    URL의 노드 정보를 추출하여 Nodename 변수와 Nodevalue 변수에 저장
```

타일 배열을 초기화


```

Switch ( Nodename ) {
    숫자 :
    질의 영역에 해당하는 타일 인덱스를 계산하여 타일 배열에
    추가
    "Tile" :
        Nodevalue를 타일 배열에 추가
    }
return SpatialQueryDispatch(Layer, Query, array of Tile)
}

```

4.2 공간 질의 디스패처

공간 질의 디스패처는 통신 관리기의 질의 영역 추출기로부터 전달되는 공간 질의를 질의 분류 테이블을 이용하여 분류하고 질의 영역인 타일들로부터 질의 요청을 처리하기 위한 클러스터링 서버를 결정하여 통신 관리기의 질의 요청 리디렉터(Redirector)로 넘겨준다. 공간 질의 디스패처는 질의 요청을 서버에 할당한 결과를 질의 처리 통계 정보 관리기로 넘기어 질의 처리 영역 및 각 타일에서 처리된 질의의 통계 정보를 기록하도록 한다. 그림 10은 이러한 공간 질의 디스패처에 대한 전반적인 구조도이며 알고리즘 1은 공간 질의 디스패처에 대한 구현 알고리즘이다.

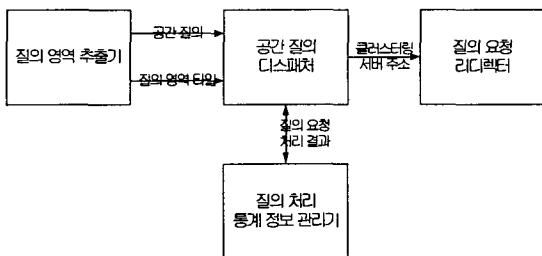


그림 10. 공간 질의 디스패처

알고리즘 2 : 공간 질의 디스패처

Input : 레이어, 공간 질의어, 질의 영역 타일
 Output : 클러스터링 서버 주소

```

String SpatialQueryDispatch (Layer, Spatial Query,
Tiles of Query Area)
{
    switch(Spatial Query) {
    영역 내부 질의 :
        질의 영역 내부에 포함된 타일을 가장 많
        이 포함하는 질의 처리 영역을 찾음
        해당 질의 처리 영역을 QPA 변수에 저장
    영역 경계 질의 :
        질의 영역 경계에 걸친 타일을 가장 많이
        포함하는 질의 처리 영역을 찾음
        해당 질의 처리 영역을 QPA 변수에 저장
    }
}

```

```

영역 외부 질의 :
    버퍼 사용량이 최소인 질의 처리 영역을
    찾음
    이들 질의 처리 영역 중 질의 영역에 포함
    되는 타일을 최소로 포함하는 질의 처리
    영역을 찾음
    해당 질의 처리 영역을 QPA 변수에 저장
}
QPA를 할당 받은 서버의 주소를 반환.
}
}

```

4.3 질의 처리 통계 정보 관리기

질의 처리 통계 정보 관리기는 질의 처리 영역의 재분할이나 최소량의 버퍼를 사용하는 질의 처리 영역을 찾기 위한 통계 정보를 관리한다. 질의 처리 영역의 재분할을 서버의 부하율을 고려하여 이루어지기 때문에 각 질의 처리 영역에서 처리되는 질의에 대한 통계 정보를 기록해야 하며, 영역 외부 질의를 할당하기 위한 질의 처리 영역을 찾기 위하여 각 질의 처리 영역에서 질의 처리에 참여한 타일의 개수를 기록해야 한다.

다음의 표 3은 최소량의 버퍼를 사용하는 질의 처리 영역을 찾기 위한 통계 데이터의 테이블 구조이다.

표 3. 버퍼 사용량 통계 테이블

QPA	질의 처리 영역 분할기에 의해 분할된 질의 처리 영역의 인덱스	
Tile	Flag0	질의 처리 영역에 포함되어 있는 각 타일에 대해 존재하는 Flag으로 질의 영역에 대응하는 타일의 Flag이 F이면 T로 전환한다.
	Flag1	
	Flag2	
	Flag3	
	Flag4	
Count	타일 Flag이 T인 타일의 개수	

4.4 질의 처리 영역 분할기

질의 처리 영역 분할기는 각 타일의 힐버트 값을 계산하여 오름차순으로 타일을 정렬한다. 또한 질의 처리 통계 정보 관리기에 의해 계산된 통계 정보를 바탕으로 각 타일에 가중치를 부여하여 각 클러스터링 서버에 할당될 타일의 가중치 값이 평균화되도록 분할함으로써 특정 지역에 대한 질의의 과중으로 인한 서버의 과부하 현상을 해결한다.

알고리즘3 : 질의 처리 영역 분할기

Input : 기존에 분할된 타일 영역
 Output : 새로 분할되는 타일 영역

```

void QPTDivide(){
    if ( 시스템이 처음 실행 or 임의의 QPA의 처리
        비율 >= 65% )
    {
        타일의 질의 처리 가중치를 계산
        질의 처리 가중치의 평균값을 계산
        변형된 힐버트 곡선에 따른 타일의 힐버트 값을 계산
        힐버트 값의 오름차순으로 타일을 정렬

        각 타일 그룹이 동일한 타일 가중치의 합을 갖도록 분할
    }
    else return
}
    
```

5. 성능 평가

본 장에서는 서버에서 처리하는 질의 영역을 인접 영역으로 그룹화한 경우와 라운드 로빈 기법을 이용한 경우 그리고 버퍼 공유 기법을 이용한 경우를 각각 비교 평가한다. 또한 질의 영역의 인접 정도에 따른 디스크 접근 횟수를 측정한다.

5.1 평가 환경

제안하는 기법과 기존의 기법과의 비교 평가를 위하여 각 기법을 적용한 시스템의 시뮬레이터를 작성하였으며 시뮬레이터의 동작 환경은 표 4와 같다.

표 4. 시스템 환경

CPU	Pentium III 700MHZ (FSB 100 MHZ)
Main Memory Size	256MB (PC100)
디스크	WD 20GB HDD (7200 RPM)
OS	Windows 2000

성능 비교에 사용된 데이터는 표 5와 같이 타일의 크기를 256개로 고정하였다.

표 5. 시뮬레이터에서 사용된 데이터

지도 데이터 크기	1.47 MB
공간 객체 수	2004 개
타일 개수	256 개

5.2 실험 평가

실험 평가는 로드 밸런싱 기법에 따른 디스크 접근 횟수와 클라이언트의 질의들의 인접 정도에 따른 디스크 접근 횟수의 두 가지로 수행하였다.

로드 밸런싱 기법에 따른 디스크 접근 횟수의 측정은 라운드 로빈 기법과 본 논문에서 제안한 기법을 적용하였다. 질의 영역의 인접 정도에 따른 디스크 접근 횟수의 측정은 클라이언트의 질의 영역이 전체 영역의 10%, 20%, 30%, 40%, 50% 이내로 인접할 경우 노드 수를 증가시킴에 따라 1000개의 인접 영역에 대한 질의를 처리하는 동안 디스크에 접근한 횟수를 측정하였다. 로드 밸런싱 기법에 따른 디스크 접근 횟수의 그래프는 다음 그림 11과 같다.

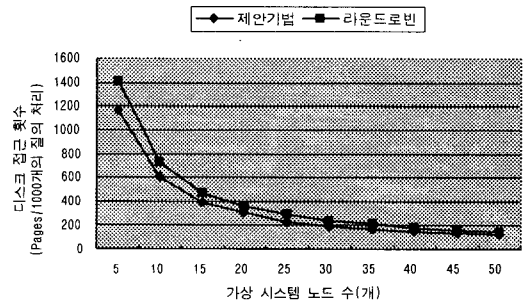


그림 11. 로드 밸런싱 기법에 따른 디스크 접근 횟수

실험 결과 제안한 기법이 라운드 로빈 기법에 비하여 향상된 성능을 보이나 노드수가 증가함에 따라 성능의 차이는 점차 감소하고 있다. 질의 영역의 인접도에 따른 디스크 접근 횟수의 그래프는 다음 그림 12와 같다.

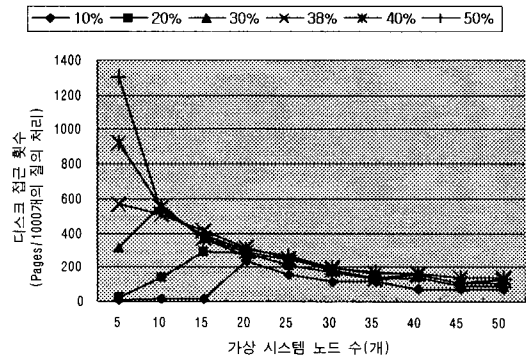


그림 12. 질의 영역의 인접도에 따른 디스크 접근 횟수

실험 결과 질의 영역의 인접도가 38%가 되기 전까지는 전반적으로 노드 수가 충분히 증가하지 않으면 오히려 디스크 접근 횟수가 증가하지만, 질의 영역의 인접도가 38%를 넘은 후에는 노드 수에 관계없이 디스크 접근 횟수가 감소한다.

6. 결 론

인터넷 이용자의 수가 급속하게 증가하자 이용자의 집중 현상으로 인한 인터넷 서비스의 중단 현상이 심각한 문제로 지적되었다. 웹 GIS에서 발생할 수 있는 서버의 과부하 문제를 해결하기 위한 기존의 연구는 근본적으로 이러한 문제를 해결하지 못한다. 웹 GIS에 웹 클러스터링 기술을 적용함으로써 안정적인 웹 GIS 서비스를 제공할 수 있다. 그러나 웹 데이터의 요청과는 달리 공간 질의는 인접 영역에 대한 질의가 매우 잦고 하나의 질의에 대한 결과 객체가 많은 특성을 가지기 때문에, 이러한 공간 질의의 특성을 고려하지 않을 경우 낮은 버퍼 재사용율로 인하여 질의 처리 비용이 증가하게 된다. 따라서 공간 질의 처리의 특성을 고려하여 서버의 버퍼를 공간 인덱스 탐색에 최적화하기 위한 방법이 필요하며, 본 논문에서는 질의 처리 영역의 인접성을 이용하여 질의 처리 영역이 인접된 질의들을 하나의 서버에서 처리하는 디스패처를 설계하고 구현하였다.

제한한 디스패처는 변형된 힐버트 곡선을 이용하여 공간 데이터를 관리하는 타일을 기반으로 인접한 타일들을 그룹핑하여 서버에 할당하였다. 또한, 공간 질의가 처리될 때 질의 처리기에 의해 참조되는 영역들의 특징을 기반으로 영역 내부 질의, 영역 경계 질의, 영역 외부 질의로 분류한 공간 질의어와 질의 영역을 참조하여, 버퍼가 공간 인덱스 참조에 최적화 될 수 있는 영역에서 질의가 처리되도록 질의 요청을 전달하였다.

제한한 기법을 적용하기 위하여 클라이언트 시스템은 질의 영역에 대한 정보를 포함하고 있는 URL을 생성하여 웹 GIS 클러스터링 시스템에 전달하였으며, 특정 영역으로의 질의가 집중에 의해 발생할 수 있는 특정 서버로 질의가 집중되는 문제를 각 타일에서 처리된 질의의 통계 데이터를 바탕으로 계산된 타일의 질의 처리 가중치를 계산하여, 각 질의 처리 영역에 포함되는 타일의 개수를 다르게 할당함으로써 해결하였다.

본 논문은 기존 클러스터링 시스템의 버퍼 관리 방법을 이용한 시뮬레이터와 제안한 기법을 이용한 시뮬레이터를 구현하여 실험 평가를 함으로써 질의 처리 영역의 인접성을 이용한 로드 밸런싱 기법이 디스크 접근 비용이 향상됨을 보였으며 클라이언트의 질의 영역이 전체 영역의 38%이상에 해당하는 영역으로 인접될 경우 클러스터링 시스템의 효율은 노드 수에 영향을 받지 않는다.

향후 연구로는 공간 질의어뿐만 아니라 비공간 질의어에 대해서도 효과적으로 버퍼를 관리하기 위한 방법이 고려되어야 하며, 제안한 기법에 버퍼 공유 기법을 적용하여 디스크의 접근 빈도를 더욱 낮추기 위한 연구가 진행되어야 할 것이다.

참 고 문 헌

- [1] S.M. Baker, B. Moon, Scalable Web Server Design for Distributed Data Management, Proceedings of Fifteenth International Conference on Data Engineering, 1999.
- [2] Valeria Cardellini, Michele Colajanni, Philip S. Yu, Redirection Algorithms for Load Sharing in Distributed Web-server Systems, Proceedings of The 19th IEEE International Conference on Distributed Computing Systems, 1999.
- [3] Shivakumar Venkataraman, Miron Livny, Jeffrey F. Naughton, Memory Management for scalable Web Data Servers, Proceedings of Thirteenth International Conference on Data Engineering, 1997.
- [4] M. Dahlin, R. Wang, T. Anderson, D. Patterson, "Cooperative Caching : Using Remote Client Memory to Improve File System Performance," Proceedings of the First Conference on Oper. Sys. Design and Implementation, 1994.
- [5] Ibrahim Kamel, Christos Faloutsos, "Parallel R-trees," SIGMOD Conference 1992.
- [6] H. V. Jagadish. "Linear Clustering of Objects with Multiple Attributes," In Proceedings of the 1990 ACM SIGMOD International Conference on Management of Data, 1990.
- [7] M. Franklin, M. Carey, and M. Livny, "Global

Memory Management in Client-Server DBMS Architectures," Proceedings of the eighteenth International Conference on VLDB, 1992.

[8] The Hilbert curve, <http://ourworld.compuserve.com/homepages/compuphase/hilbert.htm#VOORHIES>, 1999.

[9] Greg Breinholt, Christoph Schierz, "Algorithm 781: Generating Hilbert's Space-Filling Curve by Recursion," ACM Transactions on Mathematical Software, Vol. 24, No. 2, 1998.

[10] Ibrahim Kamel, Christos Faloutsos, "On Packing R-trees," CIKM 1993, 1993.

[11] Shashi Shekhar, Sivakumar Ravada, Vipin Kumar, Douglas Chubb, Greg Turner, "Declustering and Load-Balancing Methods for Parallelizing Geographic Information Systems," IEEE Transaction on Knowledge and Data Engineering, Vol. 10, No. 4, 1998.

[12] Linda Wilson, Wei Shen, "Experiments in Load Migration and Dynamic Load Balancing in SPEEDES," Proceedings of the 1998 Western Multiconference, 1998.

[13] Mounir Hamdi, Chi-kin Lee, "Dynamic Load Balancing of Data Parallel Applications on a Distributed Network," Electronic Edition (ACM DL), 1995.

[14] 이춘근, 김정원, 정기동, "스트라이핑 기반 병렬 입출력 시스템에서의 Enhanced Parallel R-Tree," 정보과학회 학술발표논문집, Vol.25, No.2, p.117-119, 1998.

[15] 조영섭, "타일 저장구조와 하이브리드 공간 질의 처리기법을 이용하는 클라이언트 중심 웹 지리정보시스템," 인하대학교 전자계산공학과 박사학위 논문, 1999.



김 중 훈

1991년 인하대학교 전자계산공학과(공학사)
 1993년 인하대학교 대학원 전자계산공학과(공학석사)
 1998년 인하대학교 대학원 전자계산공학과(공학박사)
 1996년 인하대학교 전자계산공학과 전임대우

1998년~현재 (주)케이지아이 대표이사
 관심분야 : 데이터베이스, 멀티미디어 데이터베이스 시스템, 지리 정보 시스템, SAN



이 찬 구

1999년 인하대학교 전자계산공학과(공학사)
 2001년 인하대학교 대학원 전자계산공학과(공학석사)
 2001년~현재 (주)케이지아이부설연구소 연구원

관심분야 : 데이터베이스, 멀티미디어 데이터베이스 시스템, 지리 정보 시스템, SAN



정 현 민

1984년 연세대학교전자공학과 졸업(공학사)
 1986년 연세대학교 본대학원 전자공학과(공학석사)
 1986년~1992년 한국통신 연구개발본부
 1996년 연세대학교 본대학원 전자공학과(공학박사)

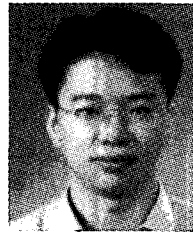
1996년~현재 (주)케이티 서비스개발연구소 무선망설계 연구실장
 관심분야 : 무선망설계 시스템 개발/엔지니어링, 영상부호화



정 미 영

1993년 덕성여자대학교 전산학과 졸업(이학사)
1996년 연세대학교 전산학과 대학원 졸업(이학석사)
1996년~현재 (주)케이티 서비스 개발연구소 무선망설계연구실 선임보연구원

관심분야 : 무선망 설계 시스템, 지형데이터 처리 시스템, 데이터베이스시스템, 데이터마이닝



배 영 호

1993년 포항공과대학 전자계산학과 (공학사)
1995년 포항공과대학 본대학원 전자계산학과 (공학석사)
1995년~현재 (주) 케이티 서비스 개발연구소 무선망연구실

관심분야 : 무선망 설계 시스템 개발, 통신 시스템 성능 분석

교신저자

김 종 훈 406-130 인천 연수구 동춘동 923-5 연수구청 6층 (주)케이아이