(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2021-0077975
(43) 공개일자 2021년06월28일

(51) 국제특허분류(Int. Cl.)
G06F 16/22 (2019.01) *G06F 16/29* (2019.01)
(52) CPC특허분류
G06F 16/2264 (2019.01)
G06F 16/2246 (2019.01)
(21) 출원번호 10-2019-0169541
(22) 출원일자 2019년12월18일
심사청구일자 2019년12월18일

(71) 출원인
서강대학교산학협력단
서울특별시 마포구 백범로 35 (신수동, 서강대학교)
(72) 발명자
정성원
서울특별시 양천구 오목로 300, 204동 604호(목동, 현대하이페리온2)
최동근
서울특별시 마포구 백범로 35 서강대학교
(74) 대리인
정부연

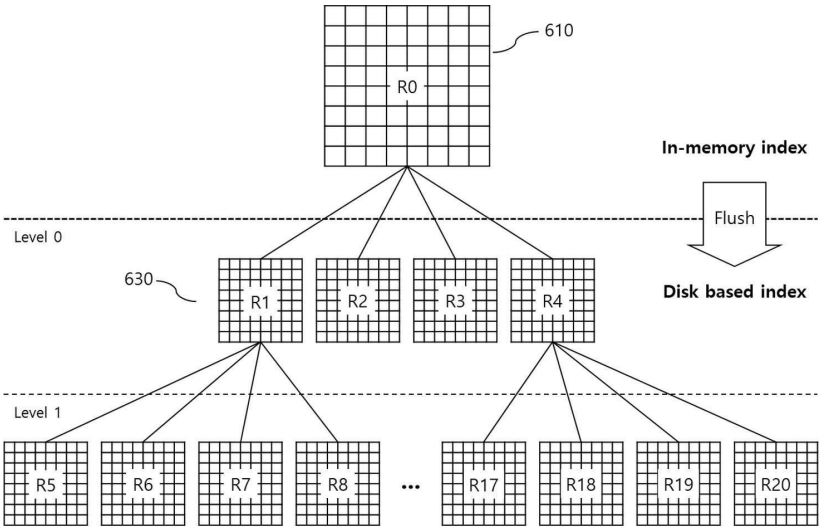
전체 청구항 수 : 총 10 항

(54) 발명의 명칭 **블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법 및 장치**

(57) 요 약

본 발명은 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법 및 장치에 관한 것으로, 상기 공간 데이터 인덱싱 방법은 전체 공간 영역을 고정 크기를 갖는 복수의 셀(cell) 영역들로 분할하고 공간 데이터 객체를 메모리 상에서 정의되는 메모리 인덱스 영역에 저장하는 단계 및 상기 메모리 인덱스 영역에 저장된 데이터의 양이 상기 메모리의 가용범위를 초과하는 경우 상기 메모리 인덱스 영역에 저장된 공간 데이터 객체를 상기 전체 공간 영역을 사분할 하고 디스크 상에서 정의되는 다계층 인덱스 영역에 플러싱(flushing)하는 단계를 포함한다.

대 표 도 - 도6



(52) CPC특허분류
G06F 16/29 (2019.01)

이 발명을 지원한 국가연구개발사업

과제고유번호	1711093140
부처명	과학기술정보통신부
과제관리(전문)기관명	정보통신기획평가원
연구사업명	정보통신기술인력양성(R&D)
연구과제명	적응형 블록체인 플랫폼 기술 개발 및 전문 인력 양성
기 여 율	1/1
과제수행기관명	서강대학교산학협력단
연구기간	2019.01.01 ~ 2019.12.31

명 세 서

청구범위

청구항 1

전체 공간 영역을 고정 크기를 갖는 복수의 셀(cell) 영역들로 분할하고 공간 데이터 객체를 메모리 상에서 정의되는 메모리 인덱스 영역에 저장하는 단계; 및

상기 메모리 인덱스 영역에 저장된 데이터의 양이 상기 메모리의 가용범위를 초과하는 경우 상기 메모리 인덱스 영역에 저장된 공간 데이터 객체를 상기 전체 공간 영역을 사분할 하고 디스크 상에서 정의되는 다계층 인덱스 영역에 플러싱(flushing)하는 단계를 포함하는 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법.

청구항 2

제1항에 있어서, 상기 메모리 인덱스 영역은

상기 복수의 셀 영역들 각각에 독립적으로 연결되는 복수의 버킷들로 구성되는 것을 특징으로 하는 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법.

청구항 3

제2항에 있어서, 상기 복수의 버킷들은

해당 셀 영역의 z-오더링 넘버(z-ordering number)를 구분자로 포함하고 상기 z-오더링 넘버에 따라 정렬되는 것을 특징으로 하는 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법.

청구항 4

제3항에 있어서, 상기 저장하는 단계는

상기 공간 데이터 객체의 위치정보를 기초로 상기 z-오더링 넘버를 산출하고 상기 z-오더링 넘버를 기초로 해당 공간 데이터 객체를 저장할 버킷을 결정하는 단계를 포함하는 것을 특징으로 하는 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법.

청구항 5

제1항에 있어서,

상기 다계층 인덱스 영역은 상기 사분할 된 각 사분면 영역마다 상기 복수의 셀 영역들로 분할되고 재귀적인 분할에 의해 계층적 구조를 형성하며,

상기 계층적 구조에 관한 정보는 상기 메모리 상에서 쿼드 트리(quad-tree) 형태로 관리되는 것을 특징으로 하는 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법.

청구항 6

제5항에 있어서, 상기 플러싱하는 단계는

상기 다계층 인덱스 영역에서 상기 메모리 인덱스 영역에 대응되는 셀 영역이 검출된 경우 해당 셀 영역에 연결

된 공간 데이터 객체를 병합하는 단계를 포함하는 것을 특징으로 하는 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법.

청구항 7

제6항에 있어서, 상기 플러싱하는 단계는

상기 병합에 따라 특정 사분면 영역에서 공간 데이터 객체의 개수가 임계값을 초과하는 경우 각 사분면 영역을 재-사분할 하고 각 재-사분면 영역으로 공간 데이터 객체를 재배치하는 단계를 포함하는 것을 특징으로 하는 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법.

청구항 8

제7항에 있어서, 상기 플러싱하는 단계는

각 재-사분면 영역을 고정 크기를 갖는 복수의 셀 영역들로 분할하고 특정 재-사분면 영역에서 공간 데이터 객체의 개수가 상기 임계값을 초과할 때마다 상기 재-사분할 과정을 반복 수행하는 단계를 포함하는 것을 특징으로 하는 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법.

청구항 9

제1항에 있어서,

질의 포인트와 질의 반경을 포함하는 공간 질의가 수신되면 상기 다계층 인덱스 영역과 상기 메모리 인덱스 영역에 대한 탐색을 순차적으로 수행하여 질의 결과를 생성하는 단계를 더 포함하는 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법.

청구항 10

전체 공간 영역을 고정 크기를 갖는 복수의 셀(cell) 영역들로 분할하고 공간 데이터 객체를 메모리 상에서 정의되는 메모리 인덱스 영역에 저장하는 메모리 인덱싱부;

상기 메모리 인덱스 영역에 저장된 데이터의 양이 상기 메모리의 가용범위를 초과하는 경우 상기 메모리 인덱스 영역에 저장된 공간 데이터 객체를 상기 전체 공간 영역을 사분할 하고 디스크 상에서 정의되는 다계층 인덱스 영역에 플러싱(flushing)하는 디스크 인덱싱부; 및

질의 포인트와 질의 반경을 포함하는 공간 질의가 수신되면 상기 다계층 인덱스 영역과 상기 메모리 인덱스 영역에 대한 탐색을 순차적으로 수행하여 질의 결과를 생성하는 공간 질의 처리부를 포함하는 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 장치.

발명의 설명

기 술 분 야

[0001] 본 발명은 공간 데이터 인덱싱 기술에 관한 것으로, 더욱 상세하게는 블록 체인 환경에서 발생하는 공간 데이터에 대한 효과적인 인덱싱을 제공할 수 있는 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법 및 장치에 관한 것이다.

배 경 기 술

[0003] 블록 체인 시스템에서 지리 정보 데이터를 포함하는 트랜잭션 로그에 대한 대표적인 검색 기술로는 Ethereum에서 개발하고 있는 오픈 프로토콜인 FOAM이 있다. FOAM은 도 5과 같이 기존의 트랜잭션 해쉬코드와 함께 데이터 좌표를 기준으로 생성된 Geohash 코드를 삽입하여 객체의 위치 정보를 나타낼 수 있다. Geohash 코드는 2차원 좌표계를 1차원 공간에 매핑시키는 기술로, key-value 데이터베이스인 LevelDB와 LSM 트리 등을 사용하여 검색 속도를 향상시킬 수 있다.

[0004] 블록체인 시스템에서 발생하는 트랜잭션 로그의 특징은 주기적으로 대량의 업데이트가 발생한다는 것이다. 업데이트 속도만을 고려하면 LSM 트리를 활용한 검색은 충분히 좋은 선택이 될 수 있다. LSM 트리는 데이터 객체가 삽입될 때 이것을 메모리에 저장하며 메모리에 저장된 데이터에 대한 로컬 인덱스를 구축한다. 만약 메모리에 저장된 데이터가 일정 크기를 초과할 경우, 메모리에 저장된 데이터는 디스크로 옮겨지며 로컬 인덱스는 디스크 인덱스와 계층적으로 병합된다.

[0005] LSM 트리는 디스크에 저장된 데이터를 순차적으로 읽고 쓰기 때문에 업데이트 속도가 매우 빠르다. 그러나 geohash 기법은 데이터 객체의 공간적 인접성을 완전히 보장하지는 못하기 때문에, geohash 코드 기반의 1차원적인 공간 데이터 검색은 검색 과정에서 대량의 긍정 오류(false positive)에 의한 검색 성능 저하가 발생할 수 있다.

[0006] 기존에 공간 데이터를 다루는 대표적인 인덱스에는 트리 형태의 R-tree, Quad-tree 등의 기법이 있다. 그러나 이 기법들은 인덱스 노드의 분할 과정에서 많은 계산 시간을 소모하기 때문에 다수의 업데이트가 한번에 발생하는 블록체인 환경에서는 효과적이지 못하다. 이 밖에 빠른 업데이트 속도를 제공하는 기법으로는 Fixed grid가 있지만 이 기법은 전체 공간을 고정된 크기의 grid cell로 구분을 하여 각각의 cell에 대한 디렉토리에 데이터를 저장하는 방식이다. 고정된 크기의 격자식 분할로 인하여 데이터 객체는 삽입시 자신이 저장될 디렉토리의 주소를 빠르게 계산할 수 있지만, 이 기법은 객체의 분포가 편향될 경우 효과적인 인덱싱이 어려우며, 평면적인 구조의 인덱스이기 때문에 적절한 분할 크기를 결정하기 어렵다는 문제가 있다.

선행기술문헌

특허문헌

[0008] (특허문헌 0001) 한국공개특허 제10-2019-0018869(2019.02.26)호

(특허문헌 0002) 한국등록특허 제10-1784612(2017.09.27)호

발명의 내용

해결하려는 과제

[0009] 본 발명의 일 실시예는 블록 체인 환경에서 발생하는 공간 데이터에 대한 효과적인 인덱싱을 제공할 수 있는 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법 및 장치를 제공하고자 한다.

[0010] 본 발명의 일 실시예는 삽입된 공간 데이터 객체에 대한 인덱스를 메모리에 적재하고 주기적으로 디스크에 플러시하는 방법을 통해 디스크에 저장된 계층적 구조의 공간 인덱스와 병합할 수 있는 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법 및 장치를 제공하고자 한다.

[0011] 본 발명의 일 실시예는 전체 공간을 고정된 크기의 격자구조로 분할한 메모리 기반의 인덱스와 전체 공간을 사분할한 부분 공간에 대한 계층적 구조를 형성하는 디스크 기반의 인덱스를 통해 효과적인 업데이트와 데이터 탐색을 제공할 수 있는 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법 및 장치를 제공하고자 한다.

과제의 해결 수단

[0013] 실시예들 중에서, 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법은 전체 공간 영역을 고정 크기를 갖는 복수의 셀(cell) 영역들로 분할하고 공간 데이터 객체를 메모리 상에서 정의되는 메모리 인덱스 영

역에 저장하는 단계 및 상기 메모리 인덱스 영역에 저장된 데이터의 양이 상기 메모리의 가용범위를 초과하는 경우 상기 메모리 인덱스 영역에 저장된 공간 데이터 객체를 상기 전체 공간 영역을 사분할 하고 디스크 상에서 정의되는 다계층 인덱스 영역에 플러싱(flushing)하는 단계를 포함한다.

[0014] 상기 메모리 인덱스 영역은 상기 복수의 셀 영역들 각각에 독립적으로 연결되는 복수의 버킷들로 구성될 수 있다.

[0015] 상기 복수의 버킷들은 해당 셀 영역의 z-오더링 넘버(z-ordering number)를 구분자로 포함하고 상기 z-오더링 넘버에 따라 정렬될 수 있다.

[0016] 상기 저장하는 단계는 상기 공간 데이터 객체의 위치정보를 기초로 상기 z-오더링 넘버를 산출하고 상기 z-오더링 넘버를 기초로 해당 공간 데이터 객체를 저장할 버킷을 결정하는 단계를 포함할 수 있다.

[0017] 상기 다계층 인덱스 영역은 상기 사분할 된 각 사분면 영역마다 상기 복수의 셀 영역들로 분할되고 재귀적인 분할에 의해 계층적 구조를 형성하며, 상기 계층적 구조에 관한 정보는 상기 메모리 상에서 쿼드 트리(quad-tree) 형태로 관리될 수 있다.

[0018] 상기 플러싱하는 단계는 상기 다계층 인덱스 영역에서 상기 메모리 인덱스 영역에 대응되는 셀 영역이 검출된 경우 해당 셀 영역에 연결된 공간 데이터 객체를 병합할 수 있다.

[0019] 상기 플러싱하는 단계는 상기 병합에 따라 특정 사분면 영역에서 공간 데이터 객체의 개수가 임계값을 초과하는 경우 각 사분면 영역을 재-사분할하고 각 재-사분면 영역으로 공간 데이터 객체를 재배치하는 단계를 포함할 수 있다.

[0020] 상기 플러싱하는 단계는 각 재-사분면 영역을 고정 크기를 갖는 복수의 셀 영역들로 분할하고 특정 재-사분면 영역에서 공간 데이터 객체의 개수가 상기 임계값을 초과할 때마다 상기 재-사분할 과정을 반복 수행하는 단계를 포함할 수 있다.

[0021] 상기 공간 데이터 인덱싱 방법은 질의 포인트와 질의 반경을 포함하는 공간 질의가 수신되면 상기 인덱스 영역과 상기 복수의 버킷들에 대한 탐색을 순차적으로 수행하여 질의 결과를 생성하는 단계를 더 포함할 수 있다.

[0022] 실시예들 중에서, 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 장치는 전체 공간 영역을 고정 크기를 갖는 복수의 셀(cell) 영역들로 분할하고 공간 데이터 객체를 메모리 상에서 정의되는 메모리 인덱스 영역에 저장하는 메모리 인덱싱부, 상기 메모리 인덱스 영역에 저장된 데이터의 양이 상기 메모리의 가용범위를 초과하는 경우 상기 메모리 인덱스 영역에 저장된 공간 데이터 객체를 상기 전체 공간 영역을 사분할 하고 디스크 상에서 정의되는 다계층 인덱스 영역에 플러싱(flushing)하는 디스크 인덱싱부 및 질의 포인트와 질의 반경을 포함하는 공간 질의가 수신되면 상기 다계층 인덱스 영역과 상기 메모리 인덱스 영역에 대한 탐색을 순차적으로 수행하여 질의 결과를 생성하는 공간 질의 처리부를 포함한다.

발명의 효과

[0024] 개시된 기술은 다음의 효과를 가질 수 있다. 다만, 특정 실시예가 다음의 효과를 전부 포함하여야 한다거나 다음의 효과만을 포함하여야 한다는 의미는 아니므로, 개시된 기술의 권리범위는 이에 의하여 제한되는 것으로 이해되어서는 아니 될 것이다.

[0025] 본 발명의 일 실시예에 따른 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법 및 장치는 삽입된 공간 데이터 객체에 대한 인덱스를 메모리에 적재하고 주기적으로 디스크에 플러시하는 방법을 통해 디스크에 저장된 계층적 구조의 공간 인덱스와 병합할 수 있다.

[0026] 본 발명의 일 실시예에 따른 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 방법 및 장치는 전체 공간을 고정된 크기의 격자구조로 분할한 메모리 기반의 인덱스와 전체 공간을 사분할한 부분 공간에 대한 계층적 구조를 형성하는 디스크 기반의 인덱스를 통해 효과적인 업데이트와 데이터 탐색을 제공할 수 있다.

도면의 간단한 설명

[0028] 도 1은 본 발명의 일 실시예에 따른 공간 데이터 인덱싱 시스템을 설명하는 도면이다.

도 2는 도 1에 있는 공간 데이터 인덱싱 장치의 물리적 구성을 설명하는 도면이다.

도 3은 도 1에 있는 공간 데이터 인덱싱 장치의 기능적 구성을 설명하는 블록도이다.

도 4는 도 1에 있는 공간 데이터 인덱싱 장치에서 수행되는 공간 데이터 인덱싱 과정을 설명하는 순서도이다.

도 5는 geo-hash를 이용한 트랜잭션 해시 값을 설명하는 도면이다.

도 6은 본 발명의 일 실시예에 따른 인덱싱 구조를 설명하는 도면이다.

도 7은 공간 인덱스 간의 데이터 삽입 속도에 관한 실험 결과를 설명하는 도면이다.

도 8은 본 발명의 일 실시예에 따른 공간 질의 처리 알고리즘을 설명하는 도면이다.

발명을 실시하기 위한 구체적인 내용

[0029] 본 발명에 관한 설명은 구조적 내지 기능적 설명을 위한 실시예에 불과하므로, 본 발명의 권리범위는 본문에 설명된 실시예에 의하여 제한되는 것으로 해석되어서는 아니 된다. 즉, 실시예는 다양한 변경이 가능하고 여러 가지 형태를 가질 수 있으므로 본 발명의 권리범위는 기술적 사상을 실현할 수 있는 균등물들을 포함하는 것으로 이해되어야 한다. 또한, 본 발명에서 제시된 목적 또는 효과는 특정 실시예가 이를 전부 포함하여야 한다거나 그러한 효과만을 포함하여야 한다는 의미는 아니므로, 본 발명의 권리범위는 이에 의하여 제한되는 것으로 이해되어서는 아니 될 것이다.

[0030] 한편, 본 출원에서 서술되는 용어의 의미는 다음과 같이 이해되어야 할 것이다.

[0031] "제1", "제2" 등의 용어는 하나의 구성요소를 다른 구성요소로부터 구별하기 위한 것으로, 이들 용어들에 의해 권리범위가 한정되어서는 아니 된다. 예를 들어, 제1 구성요소는 제2 구성요소로 명명될 수 있고, 유사하게 제2 구성요소도 제1 구성요소로 명명될 수 있다.

[0032] 어떤 구성요소가 다른 구성요소에 "연결되어"있다고 언급된 때에는, 그 다른 구성요소에 직접적으로 연결될 수도 있지만, 중간에 다른 구성요소가 존재할 수도 있다고 이해되어야 할 것이다. 반면에, 어떤 구성요소가 다른 구성요소에 "직접 연결되어"있다고 언급된 때에는 중간에 다른 구성요소가 존재하지 않는 것으로 이해되어야 할 것이다. 한편, 구성요소들 간의 관계를 설명하는 다른 표현들, 즉 "~사이에"와 "바로 ~사이에" 또는 "~에 이웃하는"과 "~에 직접 이웃하는" 등도 마찬가지로 해석되어야 한다.

[0033] 단수의 표현은 문맥상 명백하게 다르게 뜻하지 않는 한 복수의 표현을 포함하는 것으로 이해되어야 하고, "포함하다"또는 "가지다" 등의 용어는 실시된 특징, 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것이 존재함을 지정하려는 것이며, 하나 또는 그 이상의 다른 특징이나 숫자, 단계, 동작, 구성요소, 부분품 또는 이들을 조합한 것들의 존재 또는 부가 가능성을 미리 배제하지 않는 것으로 이해되어야 한다.

[0034] 각 단계들에 있어 식별부호(예를 들어, a, b, c 등)는 설명의 편의를 위하여 사용되는 것으로 식별부호는 각 단계들의 순서를 설명하는 것이 아니며, 각 단계들은 문맥상 명백하게 특정 순서를 기재하지 않는 이상 명기된 순서와 다르게 일어날 수 있다. 즉, 각 단계들은 명기된 순서와 동일하게 일어날 수도 있고 실질적으로 동시에 수행될 수도 있으며 반대의 순서대로 수행될 수도 있다.

[0035] 본 발명은 컴퓨터가 읽을 수 있는 기록매체에 컴퓨터가 읽을 수 있는 코드로서 구현될 수 있고, 컴퓨터가 읽을 수 있는 기록 매체는 컴퓨터 시스템에 의하여 읽혀질 수 있는 데이터가 저장되는 모든 종류의 기록 장치를 포함한다. 컴퓨터가 읽을 수 있는 기록 매체의 예로는 ROM, RAM, CD-ROM, 자기 테이프, 플로피 디스크, 광 데이터 저장 장치 등이 있다. 또한, 컴퓨터가 읽을 수 있는 기록 매체는 네트워크로 연결된 컴퓨터 시스템에 분산되어, 분산 방식으로 컴퓨터가 읽을 수 있는 코드가 저장되고 실행될 수 있다.

[0036] 여기서 사용되는 모든 용어들은 다르게 정의되지 않는 한, 본 발명이 속하는 분야에서 통상의 지식을 가진 자에 의해 일반적으로 이해되는 것과 동일한 의미를 가진다. 일반적으로 사용되는 사전에 정의되어 있는 용어들은 관련 기술의 문맥상 가지는 의미와 일치하는 것으로 해석되어야 하며, 본 출원에서 명백하게 정의하지 않는 한 이상적이거나 과도하게 형식적인 의미를 지니는 것으로 해석될 수 없다.

[0038] 도 1은 본 발명의 일 실시예에 따른 공간 데이터 인덱싱 시스템을 설명하는 도면이다.

[0039] 도 1을 참조하면, 공간 데이터 인덱싱 시스템(100)은 사용자 단말(110), 공간 데이터 인덱싱 장치(130) 및 데이

터베이스(150)를 포함할 수 있다.

[0040] 사용자 단말(110)은 공간 질의를 요청하고 그에 관한 질의 결과를 수신하여 확인할 수 있는 컴퓨팅 장치에 해당할 수 있고, 스마트폰, 노트북 또는 컴퓨터로 구현될 수 있으며, 반드시 이에 한정되지 않고, 태블릿 PC 등 다양한 디바이스로도 구현될 수 있다. 사용자 단말(110)은 공간 데이터 인덱싱 장치(130)와 네트워크를 통해 연결될 수 있고, 복수의 사용자 단말(110)들은 공간 데이터 인덱싱 장치(130)와 동시에 연결될 수 있다.

[0041] 공간 데이터 인덱싱 장치(130)는 공간 데이터에 관한 인덱싱을 통해 공간 데이터에 관한 데이터베이스를 구축할 수 있고 이를 기초로 공간 질의를 처리할 수 있는 컴퓨터 또는 프로그램에 해당하는 서버로 구현될 수 있다. 공간 데이터 인덱싱 장치(130)는 사용자 단말(110)과 블루투스, WiFi, 통신망 등을 통해 무선으로 연결될 수 있고, 네트워크를 통해 사용자 단말(110)과 데이터를 주고받을 수 있다.

[0042] 일 실시예에서, 공간 데이터 인덱싱 장치(130)는 데이터베이스(150)와 연동하여 공간 데이터 인덱싱 과정에서 필요한 정보를 저장할 수 있다. 한편, 공간 데이터 인덱싱 장치(130)는 도 1과 달리, 데이터베이스(150)를 내부에 포함하여 구현될 수 있고, 사용자 단말(110)과 하나의 장치로 통합되어 구현될 수 있다. 공간 데이터 인덱싱 장치(130)는 프로세서, 메모리, 사용자 입출력부 및 네트워크 입출력부를 포함하여 구현될 수 있으며, 이에 대해서는 도 2에서 보다 자세히 설명한다.

[0043] 데이터베이스(150)는 공간 데이터 인덱싱 과정에서 필요한 다양한 정보들을 저장하는 저장장치에 해당할 수 있다. 데이터베이스(150)는 공간 데이터에 관한 정보를 저장할 수 있고, 공간 데이터 인덱싱에 관한 정보를 저장할 수 있으며, 반드시 이에 한정되지 않고, 공간 데이터 인덱싱 장치(130)가 공간 데이터 인덱싱 과정에서 다양한 형태로 수집 또는 가공된 정보들을 저장할 수 있다.

[0045] 도 2는 도 1에 있는 공간 데이터 인덱싱 장치의 물리적 구성을 설명하는 도면이다.

[0046] 도 2를 참조하면, 공간 데이터 인덱싱 장치(130)는 프로세서(210), 메모리(230), 사용자 입출력부(250) 및 네트워크 입출력부(270)를 포함하여 구현될 수 있다.

[0047] 프로세서(210)는 블록체인 기반의 지리공간 데이터를 위한 공간 데이터 인덱싱 과정의 각 동작을 처리하는 프로시저를 실행할 수 있고, 그 과정 전반에서 읽혀지거나 작성되는 메모리(230)를 관리할 수 있으며, 메모리(230)에 있는 휘발성 메모리와 비휘발성 메모리 간의 동기화 시간을 스케줄할 수 있다. 프로세서(210)는 공간 데이터 인덱싱 장치(130)의 동작 전반을 제어할 수 있고, 메모리(230), 사용자 입출력부(250) 및 네트워크 입출력부(270)와 전기적으로 연결되어 이들 간의 데이터 흐름을 제어할 수 있다. 프로세서(210)는 공간 데이터 인덱싱 장치(130)의 CPU(Central Processing Unit)로 구현될 수 있다.

[0048] 메모리(230)는 SSD(Solid State Drive) 또는 HDD(Hard Disk Drive)와 같은 비휘발성 메모리로 구현되어 공간 데이터 인덱싱 장치(130)에 필요한 데이터 전반을 저장하는데 사용되는 보조기억장치를 포함할 수 있고, RAM(Random Access Memory)과 같은 휘발성 메모리로 구현된 주기억장치를 포함할 수 있다.

[0049] 사용자 입출력부(250)는 사용자 입력을 수신하기 위한 환경 및 사용자에게 특정 정보를 출력하기 위한 환경을 포함할 수 있다. 예를 들어, 사용자 입출력부(250)는 터치 패드, 터치 스크린, 화상 키보드 또는 포인팅 장치와 같은 어댑터를 포함하는 입력장치 및 모니터 또는 터치스크린과 같은 어댑터를 포함하는 출력장치를 포함할 수 있다. 일 실시예에서, 사용자 입출력부(250)는 원격 접속을 통해 접속되는 컴퓨팅 장치에 해당할 수 있고, 그러한 경우, 공간 데이터 인덱싱 장치(130)는 서버로서 수행될 수 있다.

[0050] 네트워크 입출력부(270)은 네트워크를 통해 외부 장치 또는 시스템과 연결하기 위한 환경을 포함하고, 예를 들어, LAN(Local Area Network), MAN(Metropolitan Area Network), WAN(Wide Area Network) 및 VAN(Value Added Network) 등의 통신을 위한 어댑터를 포함할 수 있다.

[0052] 도 3은 도 1에 있는 공간 데이터 인덱싱 장치의 기능적 구성을 설명하는 블록도이다.

[0053] 도 3을 참조하면, 공간 데이터 인덱싱 장치(130)는 메모리 인덱싱부(310), 디스크 인덱싱부(330), 공간 질의 처리부(350) 및 제어부(370)를 포함할 수 있다.

[0054] 메모리 인덱싱부(310)는 전체 공간 영역을 고정 크기를 갖는 복수의 셀(cell) 영역들로 분할하고 공간 데이터

객체를 메모리 상에서 정의되는 메모리 인덱스 영역에 저장할 수 있다. 여기에서, 전체 공간 영역은 공간 질의 대상이 되는 공간 영역으로서 공간 데이터가 위치 가능한 전체 영역에 해당할 수 있다. 또한, 메모리 인덱스 영역은 공간 데이터 객체에 관한 인덱스 정보가 저장되는 영역으로서 메모리의 특정 영역에 해당할 수 있다. 메모리 인덱싱부(310)는 전체 공간 영역을 고정 크기를 갖는 2^n 개의 셀(cell) 영역들로 분할할 수 있고, 공간 데이터 객체를 각각의 셀 영역에 분산하여 저장할 수 있다.

[0055] 일 실시예에서, 메모리 인덱스 영역은 복수의 셀 영역들 각각에 독립적으로 연결되는 복수의 버킷들로 구성될 수 있다. 즉, 각 셀 영역은 특정 버킷과 직접 연결될 수 있고, 메모리 인덱싱부(310)에 의해 셀 영역으로 할당된 공간 데이터 객체는 해당 셀 영역과 연결된 버킷에 저장될 수 있다. 버킷은 메모리 상에서 정의되고 고정된 크기로 형성될 수 있다.

[0056] 일 실시예에서, 복수의 버킷들은 해당 셀 영역의 z-오더링 넘버(z-ordering number)를 구분자로 포함하고 z-오더링 넘버에 따라 정렬될 수 있다. z-오더링 넘버는 셀 영역의 z-오더(z-order)에 따라 부여될 수 있고, 각 버킷은 연결된 셀 영역의 z-오더링 넘버를 포함함으로써 서로 식별될 수 있으며, z-오더링 넘버는 버킷들의 정렬 순서를 결정하는 기준으로서 활용될 수 있다.

[0057] 일 실시예에서, 메모리 인덱싱부(310)는 공간 데이터 객체의 위치정보를 기초로 z-오더링 넘버를 산출하고 z-오더링 넘버를 기초로 해당 공간 데이터 객체를 저장할 버킷을 결정할 수 있다. 메모리 인덱싱부(310)는 전체 공간 영역에서의 위도(latitude)와 경도(longitude)를 기초로 z-오더링 넘버를 산출할 수 있고, z-오더링 넘버를 구분자로 가진 버킷을 결정하여 공간 데이터 객체를 저장할 수 있다.

[0058] 디스크 인덱싱부(330)는 메모리 인덱스 영역에 저장된 데이터의 양이 메모리의 가용범위를 초과하는 경우 메모리 인덱스 영역에 저장된 공간 데이터 객체를 전체 공간 영역을 사분할 하고 디스크 상에서 정의되는 다계층 인덱스 영역에 플러싱(flushing)할 수 있다. 여기에서, 다계층 인덱스 영역은 공간 데이터 객체에 관한 인덱스 정보가 저장되는 영역으로서 디스크의 특정 영역에 해당할 수 있다. 즉, 디스크 인덱싱부(330)는 메모리 인덱스 영역이 가득 찬 경우에만 디스크 접근을 시도하여 디스크 접근 횟수를 최소화할 수 있다.

[0059] 일 실시예에서, 다계층 인덱스 영역은 사분할 된 각 사분면 영역마다 복수의 셀 영역들로 분할되고 재귀적인 분할에 의해 계층적 구조를 형성할 수 있다. 여기에서, 계층적 구조에 관한 정보는 메모리 상에서 쿼드 트리(quad-tree) 형태로 관리될 수 있다. 다계층 인덱스 영역에 저장되는 사분면 영역은 쿼드 트리의 노드에 대응될 수 있고, 각 노드는 자신으로부터 분할된 사분면 영역에 대한 포인터를 포함할 수 있으며, 결과적으로 계층적인 구조를 갖는 쿼드 트리를 형성할 수 있다.

[0060] 일 실시예에서, 디스크 인덱싱부(330)는 다계층 인덱스 영역에서 메모리 인덱스 영역에 대응되는 셀 영역이 검출된 경우 해당 셀 영역에 연결된 공간 데이터 객체를 병합할 수 있다. 즉, 디스크 인덱싱부(330)는 동일 공간에 대한 인덱싱 정보가 중첩되는 경우 이를 하나로 병합하여 메모리의 내용을 디스크로 플러싱하는 과정에서 중복되는 셀 영역을 하나로 통합할 수 있다.

[0061] 일 실시예에서, 디스크 인덱싱부(330)는 병합에 따라 특정 사분면 영역에서 공간 데이터 객체의 개수가 임계값을 초과하는 경우 해당 각 사분면 영역을 재-사분할하고 각 재-사분면 영역으로 공간 데이터 객체를 재배치할 수 있다. 임계값은 하나의 사분면 영역에 저장되는 공간 데이터 객체의 최대 개수로서 사전에 공간 데이터 인덱싱 장치(130)에 의해 설정될 수 있다.

[0062] 즉, 디스크 인덱싱부(330)는 하나의 사분면 영역에 공간 데이터 객체가 과도하게 집중되는 것을 방지하기 위하여 적정 개수의 공간 데이터 객체가 모이면 각 사분면 영역을 4개의 재-사분면 영역으로 분할하는 재-사분할을 수행한 후 공간 데이터 객체를 분산 배치할 수 있다. 재-사분할 과정은 하나의 사분면 영역을 4개의 재-사분면 영역들로 분할하고 해당 사분면 영역에 저장되어 있던 공간 데이터 객체를 재-사분면 영역으로 재배치하는 과정에 해당할 수 있다.

[0063] 일 실시예에서, 디스크 인덱싱부(330)는 각 재-사분면 영역을 고정 크기를 갖는 복수의 셀 영역들로 분할하고 분할된 특정 재-사분면 영역에서 공간 데이터 객체의 개수가 임계값을 초과할 때마다 재-사분할 과정을 반복 수행할 수 있다. 전체 공간 영역이 4개의 사분면 영역들로 분할된 경우라면, 디스크 인덱싱부(330)는 사분할 또는 재-사분할 된 영역들 각각을 2^n 개의 셀 영역들로 분할할 수 있고, 특정 사분면 또는 재-사분면 영역에 대해 공간 데이터 객체의 재배치가 이루어지는 과정에서 다시 임계값을 초과하는 경우에는 각 사분면 또는 재-사분면 영역에 대한 재-사분할 동작이 반복적으로 수행될 수 있다.

[0064] 공간 질의 처리부(350)는 질의 포인트와 질의 반경을 포함하는 공간 질의가 수신되면 다계층 인덱스 영역과 메모리 인덱스 영역에 대한 탐색을 순차적으로 수행하여 질의 결과를 생성할 수 있다. 공간 질의 처리부(350)는 공간 데이터 객체에 대한 인덱싱 정보를 이용하여 공간 질의를 처리할 수 있으며, 공간 질의에 대한 질의 결과는 일반적으로 질의 포인트를 중심으로 질의 반경 내에 존재하는 공간 데이터 객체에 관한 정보에 해당할 수 있다. 공간 질의 처리부(350)는 1차적으로 디스크에 저장된 정보를 검색할 수 있고, 디스크 검색 과정에서 메모리에 새롭게 저장된 최신 정보를 반영하기 위해 2차적으로 메모리에 저장된 정보를 검색할 수 있다.

[0065] 제어부(370)는 공간 데이터 인덱싱 장치(130)의 전체적인 동작을 제어하고, 메모리 인덱싱부(310), 디스크 인덱싱부(330) 및 공간 질의 처리부(350) 간의 제어 흐름 또는 데이터 흐름을 관리할 수 있다.

[0067] 도 4는 도 1에 있는 공간 데이터 인덱싱 장치에서 수행되는 공간 데이터 인덱싱 과정을 설명하는 순서도이다.

[0068] 도 4를 참조하면, 공간 데이터 인덱싱 장치(130)는 메모리 인덱싱부(310)를 통해 전체 공간 영역을 고정 크기를 갖는 복수의 셀(cell) 영역들로 분할하고 공간 데이터 객체를 메모리 상에서 정의되는 메모리 인덱스 영역에 저장할 수 있다(단계 S410).

[0069] 또한, 공간 데이터 인덱싱 장치(130)는 디스크 인덱싱부(330)를 통해 메모리 인덱스 영역에 저장된 데이터의 양이 메모리의 가용범위를 초과하는 경우 메모리 인덱스 영역에 저장된 공간 데이터 객체를 전체 공간 영역을 사분할 하고 디스크 상에서 정의되는 다계층 인덱스 영역에 플러싱(flushing)할 수 있다(단계 S430).

[0070] 또한, 공간 데이터 인덱싱 장치(130)는 공간 질의 처리부(350)를 통해 질의 포인트와 질의 반경을 포함하는 공간 질의가 수신되면 다계층 인덱스 영역과 메모리 인덱스 영역에 대한 탐색을 순차적으로 수행하여 질의 결과를 생성할 수 있다(단계 S450).

[0072] 도 5는 geo-hash를 이용한 트랜잭션 해시 값을 설명하는 도면이다.

[0073] 도 5를 참조하면, 지리 정보를 데이터를 포함하는 블록 체인 시스템에서 트랜잭션의 해시 값은 기존의 트랜잭션 해시 코드(530)와 함께 위치 정보(latitude, longitude)를 기초로 생성된 지오해시(geohash) 코드(510)를 기초로 생성될 수 있다. 보다 구체적으로, 트랜잭션 해시 값은 지오해시 코드(510)와 트랜잭션 해시 코드(530)를 기초로 각각 생성된 값을 서로 연결하여 생성될 수 있다.

[0075] 도 6은 본 발명의 일 실시예에 따른 인덱싱 구조를 설명하는 도면이다.

[0076] 도 6을 참조하면, 공간 데이터 인덱싱 장치(130)는 블록 체인 환경에서 발생하는 공간 데이터에 대한 효과적인 인덱싱을 제공할 수 있다. 블록체인 환경에서 발생하는 데이터의 주요 특징은 일정 간격으로 대량의 데이터 업데이트가 발생할 수 있다는 것이다. 공간 데이터 인덱싱 장치(130)는 공간 데이터 객체에 대한 인덱스를 메모리에 1차적으로 저장하고 주기적으로 디스크에 플러시(flush)하는 방법을 통해 디스크에 저장된 계층적 구조의 공간 인덱스와 병합할 수 있다.

[0077] 공간 데이터 인덱싱 장치(130)는 공간 데이터 객체가 삽입될 때마다 디스크에 저장된 대용량의 공간 인덱스를 업데이트하는 방식에 비해 디스크에 대한 임의 접근 횟수가 적어 업데이트 속도가 빠르고 계층적 구조의 인덱스를 탐색함으로써 질의 처리 과정에서의 긍정 오류(false positive)를 효과적으로 줄일 수 있다.

[0078] 도 6에서, 공간 데이터 인덱싱 장치(130)에 의해서 구축된 인덱스는 전체 공간을 고정 크기의 격자(grid)로 분할한 메모리 기반의 인덱스와 디스크 기반의 다계층 인덱스로 구성될 수 있다. 여기에서, 메모리 기반의 인덱스는 메모리 인덱스 영역(610), 디스크 기반의 다계층 인덱스는 다계층 인덱스 영역(630)에 각각 대응될 수 있다.

메모리 인덱스는 전체 공간을 2^n 개의 고정 크기 격자 셀로 분할한 후 각각의 영역에 속하는 데이터 객체들을 별도의 버킷에 저장할 수 있다. 데이터 객체가 삽입될 때, 삽입되는 데이터 객체의 위치정보를 바탕으로 z-ordering 넘버를 계산하여 저장될 버킷을 선택할 수 있다.

[0079] 또한, 반복적인 블록 삽입으로 인해 데이터의 양이 메모리의 가용범위를 초과하게 되면 공간 데이터 인덱싱 장치(130)는 모든 버킷에 축적된 정보를 디스크로 옮길 수 있다. 이때, 동일한 공간에 대한 인덱스 정보가 이미 디스크에 존재한다면 공간 데이터 인덱싱 장치(130)는 두 인덱스를 하나로 병합하는 작업을 수행할 수 있다.

[0080] 공간 데이터 인덱싱 장치(130)는 병합된 격자 인덱스가 포함하고 있는 데이터 객체의 수가 사전에 정의된 임계값 t를 초과한다면 격자 인덱스를 사분할하여 각각의 사분면에 대한 격자 인덱스를 생성할 수 있다. 이 때, 새로 생성된 사분면의 격자 인덱스는 분할된 사분면 공간을 다시 한번 2^n 개의 고정 크기의 격자로 분할한 것에 해당할 수 있다. 이 작업은 각각의 격자 인덱스가 커버하는 데이터 객체의 양이 임계값 t를 초과할 때마다 반복적으로 진행될 수 있고, 각각의 격자 인덱스는 자신으로부터 분할된 격자 인덱스에 대한 포인터를 가지고 있기 때문에 결과적으로 계층적인 구조를 갖는 인덱스가 생성될 수 있다.

[0082] 도 7은 공간 인덱스 간의 데이터 삽입 속도에 관한 실험 결과를 설명하는 도면이다.

[0083] 도 7을 참조하면, 디스크 상에서 구현된 각각의 공간 인덱스들, 예를 들어, Fixed grid, Grid file, R-tree 및 본 발명에 따른 인덱싱 방법(AMFG, Adaptive Multi-levels Fixed Grid) 각각에 따른 데이터 삽입 성능을 비교 평가한 결과를 확인할 수 있다.

[0084] 공간 데이터 객체는 geohash 코드에 의해 정렬되어 있기 때문에 격자 인덱스의 병합은 각 버킷의 병합 정렬을 통해 수행될 수 있다. 계층적 인덱스의 각 노드에 대한 병합 및 분할은 정렬된 버킷들에 대해 순차적으로 수행하기 때문에 순차적인 읽기 쓰기가 가능할 수 있다.

[0085] 기존의 Fixed grid 기법은 분할된 각 셀마다 고정 크기의 디렉토리를 유지하고 있기 때문에 비어 있는 셀의 디렉토리로 인한 공간적 낭비가 심할 뿐 아니라, 객체가 많이 포함된 디렉토리는 추가적인 디렉토리 체인을 갖게 되어 해당 셀에 대한 검색 및 업데이트시 디렉토리 체인에 대한 탐색을 필요로 하게 된다.

[0086] 반면, 공간 데이터 인덱싱 장치(130)에 의해 수행되는 인덱싱 방법은 부분 공간에 대한 정렬만을 필요로 하기 때문에 디스크 공간을 절약할 수 있을 뿐 아니라 삽입 속도 또한 빠를 수 있다.

[0088] 도 8은 본 발명의 일 실시예에 따른 공간 질의 처리 알고리즘을 설명하는 도면이다.

[0089] 도 8을 참조하면, 공간 데이터 인덱싱 장치(130)는 공간 질의 처리부(350)를 통해 질의 포인트와 질의 반경을 포함하는 공간 질의가 수신되면 디스크 상의 다계층 인덱스 영역과 메모리 상의 메모리 인덱스 영역에 대한 탐색을 순차적으로 수행하여 공간 질의에 대한 질의 결과를 생성할 수 있다.

[0090] 보다 구체적으로, 범위 질의 처리를 위한 인덱스 탐색은 루트 노드로부터 시작될 수 있다. 공간 데이터 인덱싱 장치(130)는 먼저 인덱스 노드가 커버하는 사분면과 질의 범위가 겹치는 부분이 있는지 검사할 수 있다. 겹치는 부분이 있다면 공간 데이터 인덱싱 장치(130)는 노드가 가진 격자 셀 중 질의 범위와 관련 있는 셀들을 탐색하여 해당 격자 셀에 위치한 오브젝트들을 검사할 수 있다.

[0091] 이 때, 노드에 포함된 격자 셀은 고정된 크기를 갖기 때문에 공간 데이터 인덱싱 장치(130)는 모든 셀들을 검사할 필요없이 질의 영역과 격자 범위를 비교 계산하여 필요한 셀들을 사전에 검출할 수 있다. 만약 탐색한 노드가 자식 노드를 가지고 있다면, 공간 데이터 인덱싱 장치(130)는 자식 노드에 대한 탐색을 같은 방법으로 수행할 수 있다.

[0092] 공간 데이터 인덱싱 장치(130)는 디스크 상의 데이터에 대한 검색이 끝나면 메모리에서 아직 디스크로 플러시(Flush)되지 못한 최근 입력 데이터에 대한 탐색을 진행할 수 있다. 이 경우, 메모리에 저장된 격자 인덱스에 대한 탐색은 디스크 상에서의 탐색과 동일한 방법으로 수행될 수 있다.

[0094] 상기에서는 본 발명의 바람직한 실시예를 참조하여 설명하였지만, 해당 기술 분야의 숙련된 당업자는 하기의 특허 청구의 범위에 기재된 본 발명의 사상 및 영역으로부터 벗어나지 않는 범위 내에서 본 발명을 다양하게 수정 및 변경시킬 수 있음을 이해할 수 있을 것이다.

부호의 설명

[0096] 100: 공간 데이터 인덱싱 시스템

110: 사용자 단말 130: 공간 데이터 인덱싱 장치

150: 데이터베이스

210: 프로세서 230: 메모리

250: 사용자 입출력부 270: 네트워크 입출력부

310: 메모리 인덱싱부 330: 디스크 인덱싱부

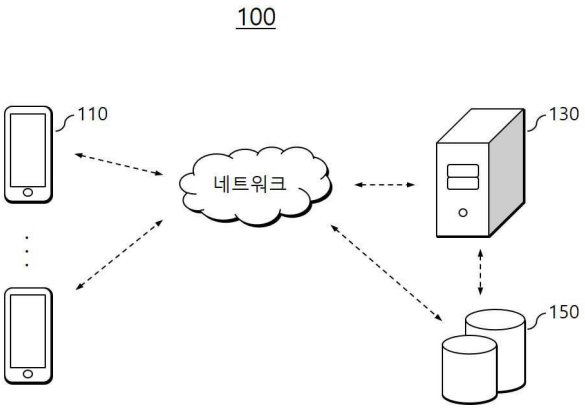
350: 공간 질의 처리부 370: 제어부

510: 지오해시 코드 530: 트랜잭션 해시 코드

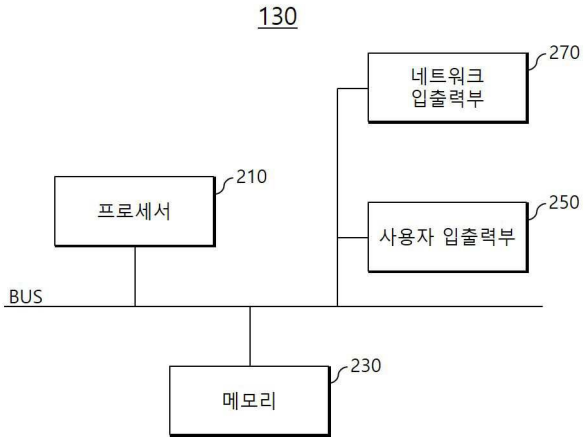
610: 메모리 인덱스 영역 630: 다계층 인덱스 영역

도면

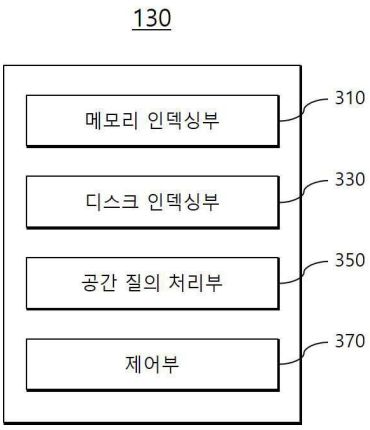
도면1



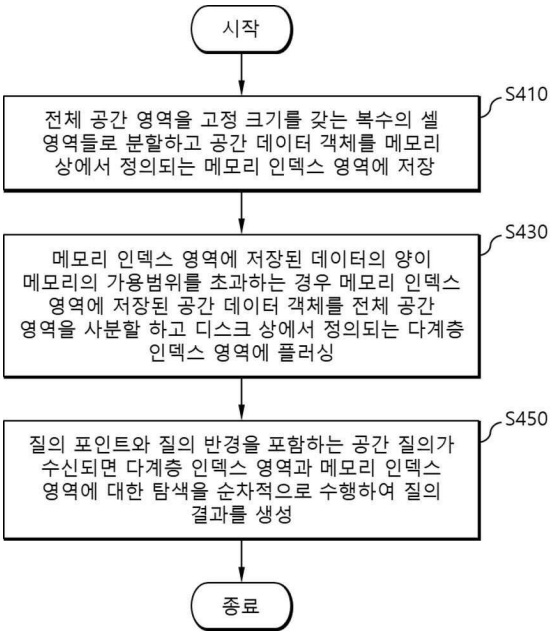
도면2



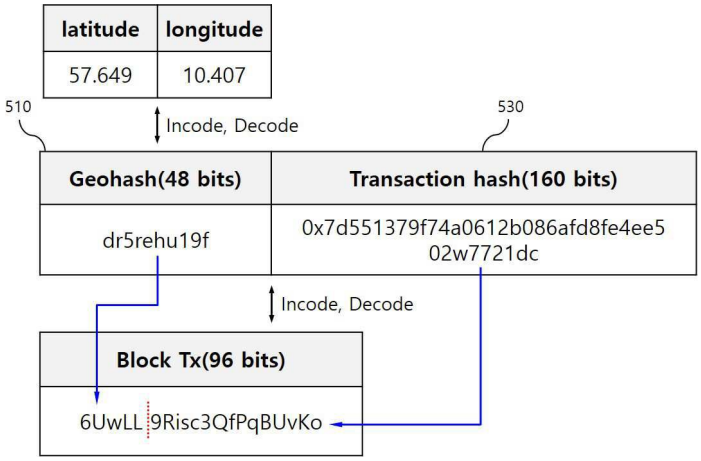
도면3


130
메모리 인덱싱부
310
디스크 인덱싱부
330
공간 질의 처리부
350
제어부
370

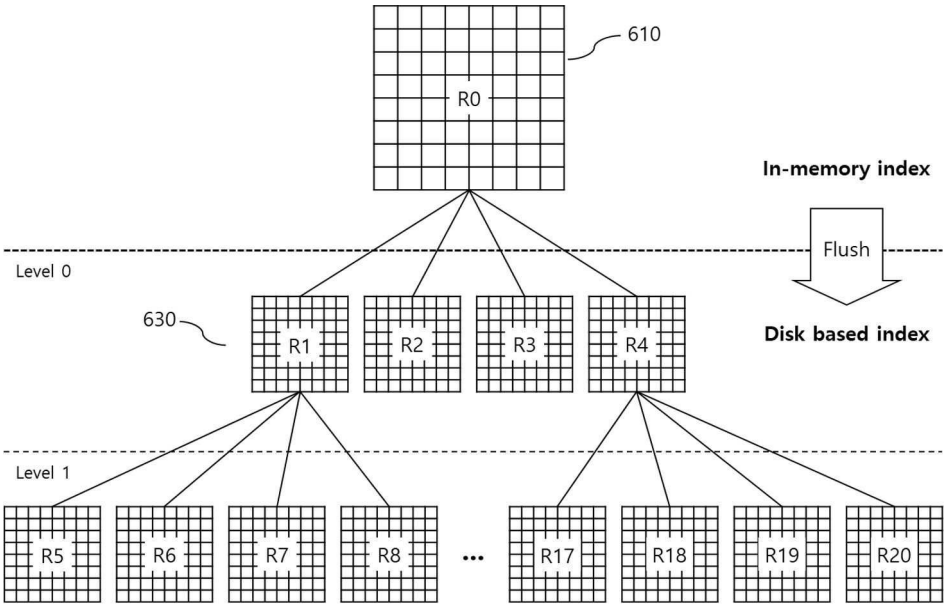
도면4


시작
전체 공간 영역을 고정 크기를 갖는 복수의 셀 영역들로 분할하고 공간 데이터 객체를 메모리 상에서 정의되는 메모리 인덱스 영역에 저장
S410
메모리 인덱스 영역에 저장된 데이터의 양이 메모리의 가용범위를 초과하는 경우 메모리 인덱스 영역에 저장된 공간 데이터 객체를 전체 공간 영역을 사분할 하고 디스크 상에서 정의되는 다계층 인덱스 영역에 플러싱
S430
질의 포인트와 질의 반경을 포함하는 공간 질의가 수신되면 다계층 인덱스 영역과 메모리 인덱스 영역에 대한 탐색을 순차적으로 수행하여 질의 결과를 생성
S450
종료

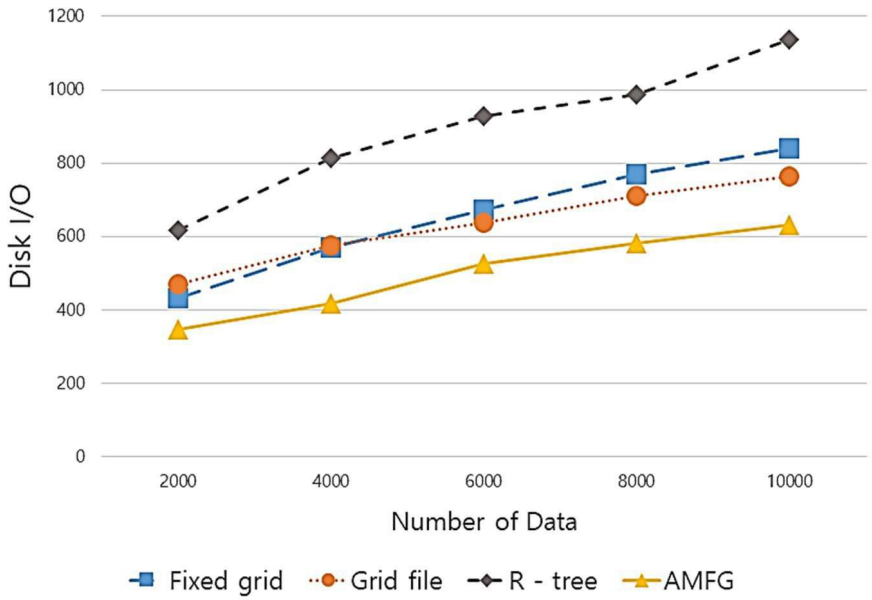
도면5


latitude
longitude
57.649
10.407
510
Incode, Decode
530
Geohash(48 bits)
Transaction hash(160 bits)
dr5rehu19f
0x7d551379f74a0612b086afd8fe4ee5
02w7721dc
Incode, Decode
Block Tx(96 bits)
6UwLL
9Risc3QfPqBUvKo

도면6


610
R0
In-memory index
Flush
Disk based index
Level 0
630
R1
R2
R3
R4
Level 1
R5
R6
R7
R8
R17
R18
R19
R20

도면7


1200
1000
800
600
400
200
0
Disk I/O
2000
4000
6000
8000
10000
Number of Data
Fixed grid
Grid file
R - tree
AMFG

도면8

Algorithm: Range query
Input: **Root** = root node of AMFG index
M = fixed grid data in memory
P = Geospatial query point
R = query range
Output: numbers_block

```
initialize node stack N, result set S=∅
push(N, Root)
while N is not empty do
  ε = pop(N)
  if Min_distance(P, ε)< R then
    for each grid cell g_i covered with query range do
      for each object o_i in cells do
        if Min_distance(P, o_i)< R then
          S=S∪{o_i}
      done
    done
  if ε have child pointer then
    for each child c_i of ε do
      push(N, c_i)
    done
done
for each grid cell g_i in M do
  if Min_distance(P, gi)< R then
    for each object o_i in cells do
      if Min_distance(P, o_i)< R then
        S=S∪{o_i}
    done
done
return S
```