

효율적 공간 검색을 위한 새로운 색인 키 중복 기법

“이경모*, 이충호*, 김성희*, 배해영*
* 인하대학교 전자계산공학과

A New Index Key Duplication Method for Efficient Spatial Retrieval

Kyung-Mo Lee, Chung-Ho Lee*, Sung-Hee Kim*, and Hae-Young Bae*

* Dept. of Computer Science and Engineering Inha University

요 약

공간 DBMS는 공간 데이터와 비공간 데이터를 서로 다른 물리적인 페이지에 저장한다. 공간 질의의 효율적인 처리를 위해 공간 데이터에 대한 공간 색인을 생성하며, 성능 향상을 위하여 이를 클러스터링 색인으로 사용한다. 그러나, 공간 데이터에 대한 클러스터링 성질이 비공간 데이터 페이지에는 반영되지 않아 공간 데이터와 비공간 데이터는 서로 다른 물리적 순서를 갖는다. 이로 인해 공간 조건(Spatial Predicate)에 의해 선택된 공간 데이터는 물리적 인접성을 가지는 반면, 비공간 데이터는 물리적 인접성을 갖지 않는다. 즉, 공간 질의 처리 과정에서 비공간 데이터 페이지의 잦은 디스크 I/O를 유발한다.

본 논문에서는 효율적 공간 검색을 위한 색인 키 중복 기법을 제안한다. 제안한 기법은 공간 데이터의 색인 키 값을 비공간 데이터 내에 중복시키고 공간, 비공간 데이터에 대해 각각의 공간 클러스터링 색인을 생성한다. 두 클러스터링 색인에 의해 공간 데이터와 마찬가지로 비공간 데이터도 공간 조건에 대한 클러스터링 성질을 유지할 수 있다. 질의 처리 과정에서 공간 조건에 의해 선택된 공간 객체들은 공간, 비공간 데이터들 모두 물리적 인접성을 가지게 되며, 비공간 데이터 페이지에 대한 디스크 액세스 횟수를 줄여 공간 검색의 성능을 향상시킬 수 있다.

1 서론

공간 데이터베이스 시스템은 그래픽 형태의 공간 데이터와 이에 대한 속성값인 비공간 데이터로 이루어진 공간 객체를 체계적으로 데이터베이스화 하기 위한 도구로서 지리정보시스템, 도면 관리 시스템 등의 다양한 응용 분야에서 하부 구조로 사용된다. 공간 객체의 편리하게 관리할 수 있는 공간 DBMS는 공간 데이터 타입, 공간 연산자를 지원하고, 이들의 효율적 검색을 위해 공간 색인을 지원해야 한다[4].

공간 데이터베이스 시스템에서 객체는 주로 공간 데이터에 의하여 액세스 되므로 공간 질의의 성능 향상을 위해서 공간 색인을 사용하며 이를 클러스터링 색인으로 사용한다[5]. 공간 클러스터링 색인은 공간 내에서 인접한 객체들을 같은 페이지에 저장하므로, 공간 조건에 의해 선택된 객체들은 물리적으로 인접할 가능성이 높아진다[5].

공간 DBMS는 확장형 DBMS에 공간 특성을 내장한 통합형 구조가 일반적이다[4]. 통합형 공간 DBMS(Integrated Spatial DBMS)는 공간 데이터와 비공간 데이터를 서로 다른 물리적 페이지에 저장하고 양방향 링크(bi-directional link)로 두 데이터를 연결하여 데이터베이스에 저장한다[2]. 그리고, 공간 질의의 성능 향상을 위해 공간 데이터에 공간 클러스터링 색인을 생성한다. 그러나, 공간 클러스터링 색인이 공간 데이터에만 반영되지만, 비공간 데이터에는 반영되지 않는다. 따라서 공간 데이터의 물리적 저장 순서와 비공간 데이터의 물리적 저장 순서는 서로 상이하게 된다.

공간 조건을 가지는 검색 질의의 수행에 있어 선택된 공간 객체들의 공간 데이터들은 물리적 인접성을 가지는 반면, 비공간 데이터들은 물리적 인접성이 보장되지 않는다. 이는 액세스해야 할 비공간 데이터 페이지가 증가하여 잦은 디스크 I/O를 발생시켜 질의 처리 성능을 떨어뜨린다.

이러한 문제점을 해결하기 위해서는 비공간 데이터도 공간 데이터와 마찬가지로 공간 질의에 의하여 함께 검색될 가능성이 높은 것들은 물리적으로 같은 장소에 위치시켜야 한다. 즉, 비공간 데이터도 공간 조건에 대해 클러스터링 성질을 가져야 하며 비공간 데이터에 공간 클러스터링 색인을 생성하여 해결할 수 있다.

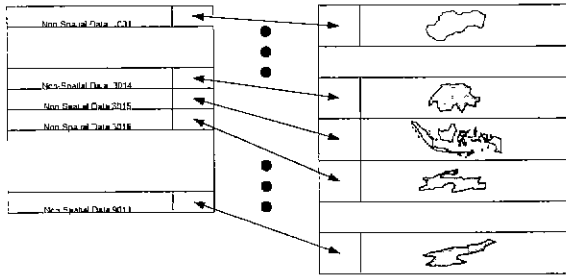
비공간 데이터 페이지의 액세스 빈도를 줄이기 위한 방법으로 본 논문에서는 공간 데이터의 색인 키를 비공간 데이터에 중복시키는 방법을 제안한다. 비공간 데이터의 색인 키에 공간 클러스터링 색인을 생성함으로써 비공간 데이터는 공간 데이터와 유사한 클러스터링 성질을 가질 수 있다. 이는 공간 검색 질의 처리 시 비공간 데이터 페이지에 대한 디스크 I/O를 줄여 처리 속도를 향상할 수 있다.

본 논문에서는 구성은 다음과 같다. 다음 절에서 데이터베이스 내에 저장되는 공간 객체의 저장 방식에 대해 설명하고, 3 절에서 간단한 공간 검색 질의를 통해 비공간 데이터의 페이지 액세스가 증가할 수 있음을 보인다. 이에 대한 해결 방법으로 색인 키 중복 기법은 4 절에서 제안하며, 마지막으로 5 절에서 결론을 맺는다.

2. 공간 객체의 저장 방식

공간 객체의 효율적인 관리를 위해 DBMS가 제공하는 잠정결과 검색속도 향상을 위한 노력은 공간 DBMS의 구조를 계층(Layered) 구조, 이중(Dual) 구조, 통합(Integrated) 구조의 형태로 발전되어 왔다[4]. 이 중 공간 DBMS의 구조는 디양용 응용을 지원할 수 있는 확장형 DBMS(extensible DBMS)를 기본으로 공간 기능을 내장하는 통합 구조가 가장 일반적이다[2][4].

공간 데이터베이스 내의 공간 객체는 공간 데이터와 비공간 데이터로 이루어지며, 통합형 공간 DBMS 는 두 데이터의 연결 정보를 [그림 1]과 같이 양방향 링크를 통해 유지한다. 양방향 링크는 비공간 데이터로부터 공간 데이터로의 연결 정보인 전방향(forward) 링크와 공간 데이터로부터 비공간 데이터로의 연결 정보인 후방향(backward) 링크로 구성된다. 전방향 또는 후방향 등의 단방향 링크만을 사용하여 공간 객체의 연결 정보를 관리하는 연구도 많았으나, 이들 시스템은 공간 또는 비공간 데이터의 한 쪽의 성능에 편중된 시스템이 대부분이다[2].



비공간 데이터 공간 데이터
[그림 1] 공간 데이터베이스의 구조

양방향 링크 정보는 공간 데이터와 비공간 데이터를 동일한 페이지 내에 연속하게 저장하여 간단히 표현할 수 있다. 그러나, 공간 데이터의 크기(extent)가 커지면 페이지 내에 저장할 수 있는 객체의 수가 감소하고, 이 수행 시 액세스해야 할 페이지 수가 증가하여 속성 조건에 의한 검색 성능을 크게 떨어뜨린다[2]. 일반적으로 공간 DBMS 는 공간 조건에 의한 검색과 속성 조건에 의한 검색 효율의 균등성을 위해 공간 데이터와 비공간 데이터를 서로 다른 페이지 내에 저장한다[2].

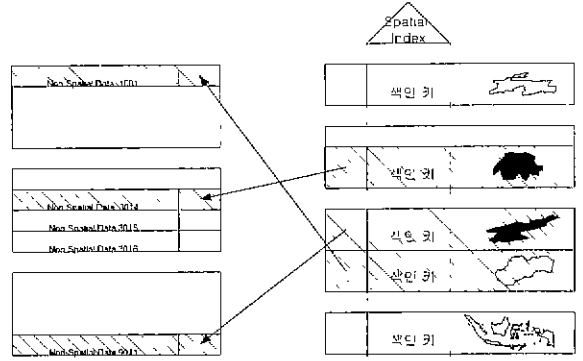
전방향 링크는 공간 데이터를 유일하게 식별할 수 있는 값을 가지며, 공간 객체의 비공간 데이터로부터 공간 데이터를 추출하기 위해 사용된다. 즉, 속성 조건에 의한 공간 데이터 검색에 사용된다. 후방향 링크는 객체의 비공간 데이터가 튜플 내에 저장되므로, 튜플 ID 등을 사용하여 표현하며 주어진 객체의 공간 데이터로부터 비공간 데이터를 얻는 데 사용된다. 즉, 공간 조건에 의한 비공간 데이터의 검색에 사용한다.

양방향 링크를 통해 서로 다른 페이지에 저장되어 있는 공간 데이터와 비공간 데이터는 공간 객체의 삽입, 삭제 등의 연산에서 공간 객체를 공간 데이터와 비공간 데이터를 서로 다른 페이지에 저장하고, 검색 연산을 위해 공간 데이터와 비공간 데이터를 조합해 공간 객체를 구성하는 기능이 필요하다[2].

3. 공간 검색의 페이지 액세스에 대한 고려

공간 데이터베이스 시스템에서는 공간 질의의 효율적 처리를 위해 공간 데이터에 공간 색인을 구축한다. 공간 객체는 그 형태와 크기가 다양하여 객체 자체를 색인의 키 값으로 표현할 수 없으므로 MBR(Minimum Bounding Rectangle), Grid 등의 공간 객체를 대표할 수 있는 근사(approximation)값을 공간 색인의 키로 사용한다[1][3]. 공간 색인은 동일한 질의에 의하여 검색될 가능성이 높은 객체들(즉, 위치와 크기가 유사한 객체들)을 물리적으로 같은 장소에 위치시킬 수 있도록 클러스터링 색인으로 사용된다.

통합형 공간 DBMS 는 공간 데이터의 비공간 데이터기 서로 다른 페이지에 위치하여 공간 색인에 의한 클러스터링 성질이 비공간 데이터 페이지에는 반영되지 않는다. 공간 데이터들을 공간 클러스터링 색인을 사용하면 [그림 2]와 같이 공간 데이터는 물리적 저장 순서기 바뀌는 반면, 비공간 데이터는 [그림 1]과 마찬가지로 물리적 위치는 변함이 없으며, 공간 데이터의 물리적 저장 순서와 상이하다.



비공간 데이터 페이지 공간 데이터 페이지
[그림 2] 공간 클러스터링 색인에 의한 데이터베이스 구조

공간 조건을 포함하는 검색 질의인 공간 선택선을 통해 공간, 비공간 데이터의 물리적 저장 순서의 차이가 비공간 데이터 페이지의 디스크 I/O를 증가시킴을 보인다.

공간 선택선(spatial selection)은 인접(neighbor), 연결(connected to) 등의 공간 조건(spatial predicate)에 기초하여 공간 객체를 검색하는 질의이다[3]. 그 예로서 다음과 같은 질의가 있다.

- A 리는 호수에서 10km 내의 모든 도로를 구하라
- 도로 B 와 교차하는 모든 도로를 구하라

공간 선택선이 질의 처리되는 과정은 간략히 다음과 같다. 공간 색인을 통해 공간 조건에 만족할 수 있는 후보 객체들을 선택하고 정제 과정을 통해 공간 조건에 부합하는 객체들을 결정한다. 공간 조건에 의해 선택되는 객체들은 공간 데이터만을 가지고 있으므로 비공간 데이터와 조합하여 질의 결과를 구성한다.

공간 조건에 부합하는 공간 데이터들은 클러스터링 성질에 의해 어느 정도 물리적으로 인접한다. 그러나, 비공간 데이터들은 그 물리적 인접성이 보장되지 않는다. [그림 2]에서 질의 결과(빗금 친 영역)는 공간 데이터의 경우 비교적 인접한 위치에 존재하지만, 비공간 데이터의 경우 여러 페이지에 걸쳐 존재하게 된다. 즉, 비공간 데이터는 공간 조건에 대해 클러스터링 성질을 갖지 않는다.

이러한 질의 결과의 구성 과정에서 비공간 데이터 페이지의 잦은 디스크 I/O 를 요구하여 공간 검색의 성능을 저하시킨다. 질의 결과에 해당하는 공간 객체들을 구성하기 위해서는 공간 데이터의 후방향 링크를 사용하여 비공간 데이터를 얻는다. 하나의 공간 객체를 구성하면, 선택된 비공간 데이터가 포함된 페이지는 버퍼 내에 존재한다. 그러나, 다음 질의 결과를 구성할 때 원하는 비공간 데이터가 이전에 로드된 버퍼 페이지 내에 있을 확률이 적다. 즉, page fault 의 확률이 높으며 질의 결과가 많을수록 심하다.

공간 데이터베이스 시스템에서는 공간 조건을 사용하는 검색 질의가 많이 발생하게 되며, 이를 위해 비공간 데이터들도 공간 조건에 대해 클러스터링 성질을 갖도록 해야 한다

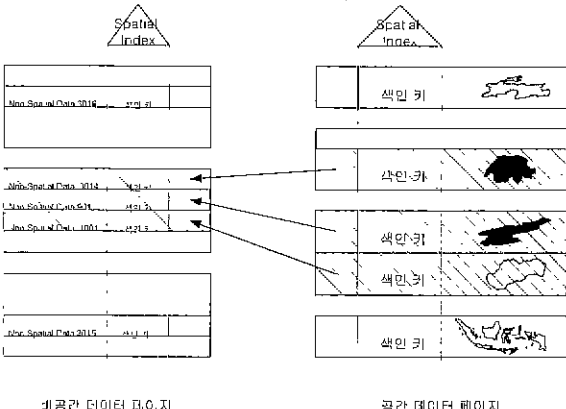
4. 색인 키 중복 기법의 제안 및 공간 검색 처리 과정

공간 선택성 등의 공간 조건을 사용하는 검색 질의의 효율을 위해서는 비공간 데이터들도 공간 데이터들과 유사한 물리적 저장 순서를 가질 수 있어야 한다 즉, 비공간 데이터도 공간 데이터들과 마찬가지로 공간 조건에 의한 클러스터링 성질을 가져야 한다

이를 위해 다음과 같은 방법들을 고려해 볼 수 있다. 먼저, 비공간 데이터의 공간 내이티브를 동일한 페이지에 저장하여 이러한 문제를 해결할 수 있다 이러한 방법은 공간 검색에 대한 효율은 보장할 수 있으나, 앞에서 언급했듯이 공간 객체는 그 크기와 형태가 다양하여 공간 데이터의 크기(extent)가 거대한 페이지 내에 저장되는 객체 수가 감소해 속성 조건에 의한 검색 효율이 떨어지므로 바람직하지 못하다

둘째, 데이터베이스 구축 시 공간 객체들을 공간 조건에 대한 클러스터링 정도를 고려한 순서대로 일괄 저장하는 방법이다 즉, 물리적으로 저장될 순서대로 객체들을 정렬하여 저장함으로써 비공간 데이터는 자연스럽게 공간 데이터와 물리적 순서를 동등하게 유지할 수 있다 그러나 공간 객체의 삽입, 삭제에 의해 비공간 데이터의 물리적 순서는 공간 데이터의 저장 순서와 다르게 되어 클러스터링 성질을 상실하게 된다

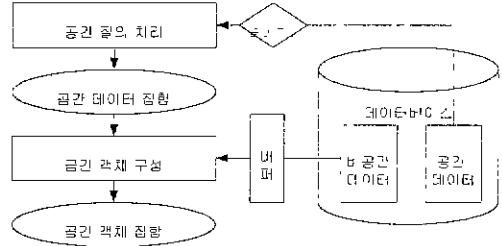
본 논문에서 제안하는 방법은 [그림 3]과 같이 비공간 데이터에도 공간 데이터의 동일한 공간 클러스터링 색인을 유지하는 것이다. 공간 데이터의 비공간 데이터는 서로 다른 페이지에 저장되므로 비공간 데이터를 위한 공간 색인도 클러스터링 색인으로 사용할 수 있다. 공간 데이터와 비공간 데이터에 색인 키를 중복시키고 이들에 대해 공간 클러스터링 색인을 사용하면 두 데이터의 물리적 저장 순서가 유사하게 되며 비공간 데이터도 공간 조건에 대한 클러스터링 성질을 갖게 된다 [그림 3]의 빗금 친 영역은 공간 검색 질의에서 공간 조건에 의해 선택된 공간 객체를 보여준다 [그림 2]와 달리 비공간 데이터 페이지에서 선택된 데이터들도 물리적으로 인접함을 알 수 있다



[그림 3] 색인 키 중복에 의한 데이터베이스 구조

[그림 4]의 공간 객체 구성 과정에서 하나의 공간 객체를 구성하던 해당하는 비공간 데이터의 페이지는 여러 내에 존재하게 된다 비공간

데이터들도 물리적으로 인접해 있어 다음 객체의 구성에 필요한 비공간 데이터도 동일한 페이지 내에 존재할 확률이 높다 즉, 원하는 페이지가 여러 내에 존재할 확률이 높아 디스크 I/O 를 줄일 수 있으며 공간 검색 질의의 성능이 향상된다



[그림 4] 공간 검색 질의 처리 과정

동일한 색인 키를 유지하기 위해서는 공간 데이터의 색인 키를 비공간 데이터에 중복시켜야 한다 색인 키를 중복 유지하기 위해서는 공간 객체의 삽입, 삭제, 갱신 등에 대한 고려를 하여야 한다. 공간 객체의 삽입 시 공간 객체 판독자는 공간 데이터의 색인 키(예를 들어, MBR)를 추종하고 이를 공간 데이터 뿐만 아니라 비공간 데이터에도 적용한다 공간 객체의 삭제는 공간 데이터와 비공간 데이터를 모두 삭제하므로 이는 문제되지 않는다. 공간 객체의 갱신 시 공간 데이터의 색인 키는 변경될 수 있으며, 변경이 일어난 경우엔 비공간 데이터의 중복 키를 변경시킨다

제안한 기법은 색인 키를 중복시키고 공간 클러스터링 색인에 의해 자동적으로 비공간 데이터의 클러스터링 성질을 유지시키기 때문에 별도의 공간 색인 구조의 변경을 필요로 하지 않는다

5. 결론

본 논문에서는 공간 DBMS 에서 공간 조건에 의한 공간 검색 질의가 질의 결과 구성 과정에서 비공간 데이터의 페이지 액세스를 빈번하게 발생함을 보이고, 이를 위해 비공간 데이터도 공간 조건에 대해 클러스터링 성질을 가지는 힘을 설명하였다 이를 위한 해결 방안으로 비공간 데이터에도 공간 데이터의 색인 키를 중복하여 비공간 데이터와 공간 데이터에 각각의 공간 클러스터링 색인을 적용시키는 방법을 제안하였다 제안한 방법은 공간 검색의 질의 처리 과정에서, 공간 데이터로부터 비공간 데이터로의 참조 과정에서 액세스해야 할 페이지를 줄여 디스크 I/O 를 줄인다

제안한 방법은 비피 관리, 질의 최적화 등에도 영향을 미치며, 이에 대한 향후 연구가 필요하다.

[참고 문헌]

- [1] H. Lu and B.-C. Ooi, "Spatial Indexing: Past and Future," IEEE Data Engineering Bulletin, Vol. 16, No. 3, pp. 16-21, Sept. 1993.
- [2] H. Samet and W. G. Aref, "Spatial Data Models and Query Processing," Modern Database Systems, W. Kim (Eds.), ACM Press, pp. 338-360, 1995.
- [3] M. J. Egenhofer and A. U. Frank, "Towards a Spatial Query Language: User Interface Considerations," Proc. 14th International Conference on VLDB, edited by D. DeWitt and F. Bancillon, 124-133, Los Angeles, CA, Aug. 1988.
- [4] R. Güting, "An Introduction to Spatial Database Systems," VLDB Journal, Vol. 3, pp. 357-399, 1994.
- [5] T. Brinkhoff and H. P. Kriegel, "The Impact of Global Clustering on Spatial Database Systems," Proc. 20th International Conference on VLDB, pp. 168-179, 1994.