

---

# LBSNS에서의 공간조인을 이용한 필터링 기법의 설계

이은식\* · 조대수\*\*

\*동서대학교

## A Design of Filtering Technique on LBSNS using Spatial Join

Eun-Sik Lee\* · Dae-Soo Cho\*\*

\*Dongseo University

E-mail : crubeido@nate.com , dscho@dongseo.ac.kr

### 요 약

최근 GPS를 내장한 디지털장치(스마트폰, 태블릿PC 등)가 등장함에 따라 수많은 LBSNS(Location-Based Social Networking Services) 애플리케이션들이 등장하고 있으며 기존의 많은 SNS 애플리케이션들 또한 위치기반 서비스를 추가하여 서비스 하고 있다. 트위터의 경우 관심지역에 대한 뉴스를 일반적인 구독의 형태가 아닌 연산자를 통한 검색의 형태로써 제공하고 있다. 본 논문에서는 트위터에서 사용자가 관심지역에 대한 뉴스를 간단히 구독을 할 수 있게 설계된 시스템을 설명한다.

이 시스템은 대량의 질의가 발생할 것을 고려하여 인덱스를 사용하지 않는 공간조인(Spatial Join) 기법 중 하나인 PBSM(Partition Based Spatial-Merge Join)을 통해 일괄처리 한다. 공간조인을 통해 만들어진 최종 결과들은 실체화 된 뷰(Materialized View)를 통해 저장되어 지고 사용자들에게 제공 된다.

### ABSTRACT

Owing to the advent of digital devices which equipped with GPS, such as smartphone and tablet pc, a number of LBSNS applications have been released and even SNS applications serve various Location-Based Services. In twitter's case, the news of interesting area is provided to user not by being subscribed them automatically, but by being searched on web-site. This paper describes the system designed for users want to subscribe the local news without procedure like searching using operators. This system uses PBSM(Partition Based Spatial-Merge Join) which has no index for batch processing and against a massive query. The results from Spatial Join are stored in Materialized View then provided to user.

### 키워드

트위터, LBSNS, 공간조인, PBSM, Materialized View

### 1. 서 론

무선통신이 가능한 디지털장치들이 등장하면서 SNS애플리케이션들이 등장하였고 최근에는 GPS가 함께 탑재된 디지털장치(스마트폰, 태블릿PC 등)도 개발되면서 LBSNS라는 새로운 분야의 애플리케이션들이 출시되고 있다. 이에 맞추어 기존의 일반 SNS애플리케이션 또한 위치기반 서비스를 추가하는 추세이다. 그 중 트위터는 자체적인 검색연산자(Operators)들을 사용하여 사용자의 관

심지역 뉴스를 구독의 형태가 아닌 검색의 형태로써 제공하고 있다. 예를 들어서 'near:pusan within:15mi' 다음과 같은 연산자를 사용한 검색어는 부산 인근 15마일 내에서 작성된 트윗들을 검색하는 것이다. 이러한 기능은 사용자가 간단히 관심지역의 정보를 얻기 위해서는 많은 연산자들을 외워야 하는 어려움이 따른다. 그리고 모든 사용자들의 트윗(Tweet)들을 대상으로 검색하기 때문에 자신이 팔로잉 하지 않은 사용자들의 글들도 함께 검색된다. 구독이라는 형태를 기본으

로 하고 있는 트위터의 위치기반 서비스라고 하기에는 아쉬운 점이 있다.

본 논문에서는 사용자가 자신의 관심지역 정보를 간편하게 구독 할 수 있게 하고 팔로잉하는 사용자들의 글들만을 필터링 할 수 있는 시스템을 소개한다. 먼저 각 트윗들과 사용자들에 위치 정보를 추가 할 수 있다. 이를 통해 나의 관심지역을 정적으로 설정 하여 정보를 구독 할 수도 있고 나의 위치가 이동함에 따라 동적으로 정보를 구독 할 수도 있다. 대량의 질의가 발생할 것을 고려하여 인덱스가 없는 공간조인[1][2][3]중 하나인 PBSM[2]을 사용하고 공간조인을 통한 정보는 실체화 된 뷰[4]를 통해 저장 되어 지고 사용자들에게 제공된다.

## II. 관련 연구

### 2.1 공간조인(Spatial Join)

공간조인 연산은 두 가지 이상의 공간 데이터 집합들 중에서 공간술어(Spatial Predicate)에 만족하는 튜플들의 쌍들을 구한다. 예를 들어 아래의 그림 1에서 “모든 공원 내에 있는 건물들을 찾아주세요.” 라는 질의가 수행 되었을 때 최종 튜플들의 쌍은  $\{(A1,B1), (A2,B1), (A3,B2)\}$ 으로 나타낼 수 있고 이 중 공원 내 건물들은  $\{A1, A2, A3\}$ 임을 알 수 있다.

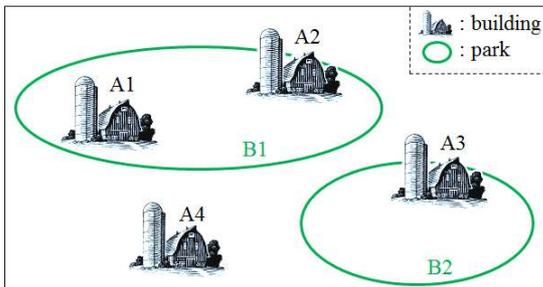


그림 1. 공간조인의 예

공간조인은 필터 단계(Filter Step)와 정제 단계(Refinement Step)로 나누어 수행한다. CPU에서 공간객체들이 서로 겹치는지 아닌지에 대한 판단을 하기 위해서는 많은 연산 비용이 필요하다. 따라서 그림 2와 같이 필터 단계에서는 먼저 공간객체에 MBR(Minimum Bounding Rectangle)을 씌우고 대략적인 연산을 통해 필터링 한다. 정제 단계에서는 필터 단계에서 나온 결과들이 실제로 맞는지에 대한 정확한 연산을 수행한다.

### 2.2 PBSM(Partition Based Spatial-Merge Join)

공간조인은 크게 내부 메모리기반과 외부 메모리기반으로 나눌 수 있다. 외부 메모리 기반에서는 다시 공간조인을 하게 될 두 개의 데이터 집합이 모두 인덱스(Index)를 가지고 있는 경우[1],

둘 중 하나만 가지고 있는 경우[1][5], 둘 다 없는 경우[1]로 나눌 수 있다. 본 논문의 시스템은 LBSNS의 특성상 대량의 질의를 처리하기 위해서 인덱스를 필요로 하지 않는 PBSM을 사용한다. PBSM은 인덱스가 없는 대신 공간 정렬 병합(Spatial Sort Merge) 알고리즘을 통해 필터 단계가 수행 된다.상세한 알고리즘 소개는 생략하도록 한다.

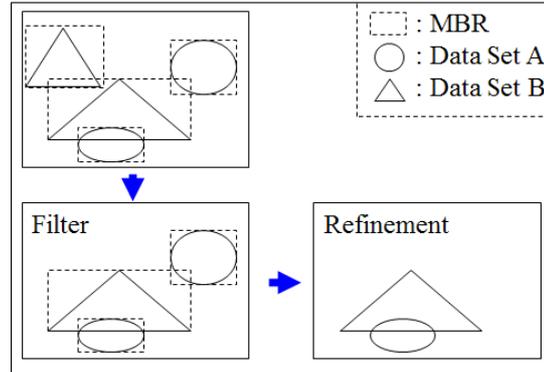


그림 2. 필터 단계 및 정제 단계

### 2.3 실체화 된 뷰(Materialized View)

실체화 된 뷰는 캐시와 같이 데이터로의 빠른 접근이 가능하다. 일반 뷰(View)의 경우 질의의 양이 매우 많거나 조인과 같은 복잡한 질의가 있는 시스템의 경우 모든 질의에 대해 갱신을 하는 것은 어렵다. 하지만 실체화 된 뷰는 물리적 디스크의 참조가 아닌 실제 메모리를 가짐으로써 빠른 갱신을 가능하게 한다.

## III. 시스템 설계

### 3.1 위치정보의 추가

시스템 설계를 위해 가장 먼저 할 작업은 사용자들과 트윗들에 각각 MBR을 추가하는 것이다. 단, 트윗들의 MBR은 정적인 속성만 가능하며 사용자들의 MBR은 정적, 동적 속성 모두 가능하다. 다시 말해서 자신의 관심지역을 정적으로 설정하고 그 지역만의 정보만을 구독 할 수도 있고 동적으로 설정하여 자신의 위치가 이동함에 따라 달라지는 정보를 구독 할 수도 있다. 사용자가 관심지역의 속성을 동적으로 설정한다고 가정할 때 사용자는 MBR의 레벨을 선택해야 한다. 여기서 레벨이란 지도상에서 시스템이 지정한 구역(나라별, 도시별, 행정 구역별 등)을 VG(Virtual Grid)형태로 나눈 것을 말한다. 레벨이 올라갈수록 구역을 세분화 할 수 있다. 그림 3에서 사용자의 위치가 레벨 0에서 이동한 것과 레벨 1에서 위치가 이동한 것은 다른 의미를 갖는다. 레벨 0의 경우는 사용자의 위치가 이동되었다고 볼 수 없는 상태이다. 하지만 레벨 1의 경우는 경계선을

넘어 다른 구역으로 이동 되었기 때문에 사용자 위치가 이동 되었다고 본다.

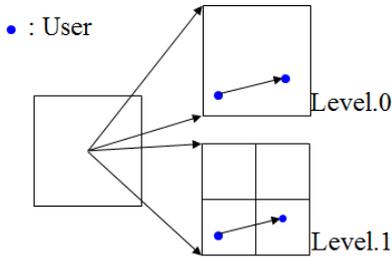


그림 3. 동적 속성 MBR 레벨

### 3.2 시스템 설계

전체적인 시스템의 구조는 그림 4와 같다.

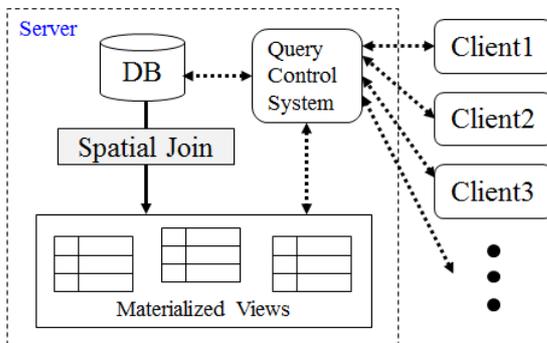


그림 4. 시스템 구조도

클라이언트들로부터 발생되는 모든 질의(트윗 작성, 트윗 삭제, 업데이트, 팔로잉 등)는 QCS(Query Control System)에 의해 처리가 된다.

먼저 새로운 트윗 작성이나 삭제와 같은 질의에 대한 처리는 DB로 접근하여 새롭게 삽입, 삭제, 갱신 등의 처리를 수행하도록 한다. 반면 타임라인의 업데이트와 같이 질의가 복잡한 경우 질의 발생 시 마다 DB로 접근하여 처리하는 것은 매우 어렵다. 이를 위해 2.3절에 소개한 실체화 된 뷰를 사용함으로써 문제를 해결하고자 한다. 실체화 된 뷰가 데이터로의 빠른 접근을 보장한다고는 하지만 여전히 대량의 질의가 발생할 때 마다 실체화 된 뷰를 갱신 하는 것은 무리가 있다. 따라서 주기적인 공간조인을 통하여 실체화 된 뷰의 갱신 비율을 낮출 수 있도록 하였다. 반면 공간조인을 통해 만들어진 실체화 된 뷰는 항상 최신의 정보를 가질 수 없기 때문에 QCS를 통해 현재 사용자의 정보가 최신상태인지를 검사 되어진 후 수행한다. 만약 최신의 정보가 아니라면 QCS는 DB로 접근하여 처리한다. 자세히 말하면 그림 5와 같이 실체화 된 뷰가 1분에 한번 씩 공간조인으로 인해 업데이트 된다고 가정 할 때 현재 30초가 지난 정보임에도 불구하고 가장 최신의 실체화 된 뷰이며 현재 DB에 저장되어진

정보와 다를 수 있다. C가 새로운 트윗 C.2를 작성하였으며 동적 속성을 사용하였기 때문에 현재 위치가 이동이 되었기 때문에 실체화 된 뷰 로의 접근이 불가능 하다. 따라서 QCS는 이것을 검사와고 처리하는 역할을 하는 것이다.

| User     | A(static) | B(static) | C(dynamic) | D(static) |
|----------|-----------|-----------|------------|-----------|
| Follower | B, C      | A, C      | B, D       | C         |
| Tweet    | A.1       | B.1       | C.1        |           |
|          | A.2       |           | C.2(new)   |           |
|          | A.3       |           |            |           |

| Up-to-date Materialized View | DB            | QCS           | Client1       | Client2       | Client3       |
|------------------------------|---------------|---------------|---------------|---------------|---------------|
| A.1, A.2, A.3                | A.1, A.2, A.3 | A.1, A.2, A.3 | A.1, A.2, A.3 | A.1, A.2, A.3 | A.1, A.2, A.3 |

| Up-to-date Database | DB                 | QCS                | Client1            | Client2            | Client3            |
|---------------------|--------------------|--------------------|--------------------|--------------------|--------------------|
| A.1, A.2, A.3, C.2  | A.1, A.2, A.3, C.2 | A.1, A.2, A.3, C.2 | A.1, A.2, A.3, C.2 | A.1, A.2, A.3, C.2 | A.1, A.2, A.3, C.2 |

그림 5. 최신정보가 아닌 실체화 된 뷰

## IV. 결 론

본 논문에서는 트위터가 가지고 있는 위치기반 서비스 면의 문제점을 언급하였고 이를 보완하기 위한 시스템을 제안하였다.

트위터의 현재 사용자의 수와 이용률을 고려하여 새로운 시스템에서는 대량의 공간 질의들을 처리할 기법이 필요하여 인덱스가 없는 공간조인 PBSM을 사용하고 그 결과는 실체화 된 뷰를 통해 사용자에게 제공되었다.

향후 과제로는 공간조인을 사용함으로써 최신의 정보를 실체화 된 뷰를 가지지 못하는 단점을 보완하는 기법이 필요하다. 그리고 본 논문의 시스템을 바탕으로 실제 공간조인을 이용한 방법이 공간조인을 사용하지 않았을 때보다 빠르지에 대한 성능평가가 필요하다.

## 참고문헌

- [1] Edwin H. Jacox , Hanan Samet "Spatial Join technique" ACM Transactions on Database Systems, 2007
- [2] Jignesh M. Patel , David J. Dewitt "Partition Based Spatial-Merge Join" ACM SIGMOD Record, 1996
- [3] Ming-Ling Lo , China V. Ravishankar "Spatial hash-joins" ACM SIGMOD Record, 1996
- [4] [http://ihelpers.x2soft.co.kr/programming/reference/01spring\\_materialized\\_view.pdf](http://ihelpers.x2soft.co.kr/programming/reference/01spring_materialized_view.pdf)
- [5] Ming-Ling Lo , China V. Ravishankar "Spatial Joins using Seeded Trees" ACM SIGMOD Record, 1994