

모바일 사용자를 위한 Q+R 트리 기반 퍼브-서브 시스템

(Q+R Tree based Pub-Sub System for Mobile Users)

이명국*, 김경백**

(Myung-Guk Lee, Kyungbaek Kim)

요약

퍼브/서브 시스템(Pub/Sub System)은 시스템에서 발행되는 정보 중 사용자가 등록한 관심 정보만을 사용자에게 전달해주는 시스템이다. 기존의 퍼브/서브 시스템은 콘텐츠의 저장 및 전달을 담당하는 브로커들을 네트워크화 하여 구현되었다. 모바일 사용자가 급증함에 따라 사용자의 관심위치 정보와 같은 지속적으로 변하게 되는 관심정보를 다루기 위한 퍼브/서브 시스템에 대한 수요가 부각 되고 있다. 이 논문에서는 기존의 퍼브/서브 시스템에서 깊이 고려하지 않았던, 관심 위치 정보의 빈번한 변화를 효과적으로 처리하기 위한 브로커 네트워크 기반의 퍼브/서브 시스템을 제안한다. 사용자의 행동 패턴이나 지리적 특성을 고려해 퍼브/서브 시스템에서 관리하는 공간 데이터 영역을 Slow Moving Region과 Normal Moving Region의 두가지 타입으로 구분하고, 각 영역에 대한 사용자의 요청을 효과적으로 지원하기 위해 Q+R트리를 사용하여 브로커를 관리한다. 시뮬레이션을 사용한 실험 결과를 통해 제안하는 Q+R트리 기반의 브로커 네트워크가 불필요한 브로커의 로드와 네트워크 트래픽을 감소시킴으로써 보다 효과적으로 지속적인 사용자의 관심 위치 정보 변화를 지원할 수 있음을 확인하였다.

■ 중심어 : Q+R 트리, 퍼브서브 시스템, 브로커 네트워크, 모바일 사용자

Abstract

A pub(lish)/sub(scribe) system is a data forwarding system which forwards only interesting data among the whole published data, which is related to the subscriptions registered by end users. Classical pub/sub systems are realized by constructing a network of brokers which are responsible for storing or forwarding data. Along with the substantial increase of the population mobile users, it is required that the pub/sub system handles the subscriptions of user locations which changes continuously and frequently. In this paper, a new broker network based pub/sub system which efficiently handles the frequent changes of subscriptions related to user locations is proposed. In consideration of moving patterns of users and geographical property, the proposed pub/sub system categorize the entire data space into Slow Moving Region and Normal Moving Region, and manages the brokers which are responsible for these regions by using Q+R tree in order to handle user requests more efficiently. Through the extensive simulation, it is presented that the proposed Q+R tree based pub/sub system can reduce unnecessary needs of brokers and network traffic and can support the dynamic subscription related to user location.

■ keywords : Q+R tree, Pub/Sub System, Broker Network, Mobile Users

I. 서론

퍼브-서브(pub-sub) 시스템은 정보를 다수의 사용자들에게 전달하기 위한 가장 효과적인 모델 중 하나이다. 퍼브-서브 시스템에서 퍼블리셔(publisher)는 전달하고자 하는 정보를 생성하고, 서브스크라이버(subscriber)들은 관심있

는 정보들만을 시스템에서 받는다. 퍼브-서브 시스템은 메시징 기반 통신방식을 취함으로써, 퍼블리셔와 서브스크라이버간의, 시간적 그리고 공간적으로, 비 동기화된 데이터 통신을 제공한다[1]. 즉, 퍼블리셔는 제공하고자 하는 데이터에 관심 있어 하는 서브스크라이버들의 정보를 알지 못하더라도 데이터를 알맞은 서브스크라이버들에게 전송할 수 0 있다는 것이다. 이를 통해 퍼브-서브 시스템은 높은 확장성과 유연성을 가질 수 있다.

* Western Digital

** 정회원, 전남대학교 전자컴퓨터공학부

이 논문은 2015년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (NRF-2014R1A1A1007734).

접수일자 : 2015년 09월 10일

수정일자 : 2015년 09월 25일

계재확정일 : 2015년 09월 27일

교신저자 : 김경백, e-mail : kyungbaekkim@jnu.ac.kr

메시지 기반 통신방식을 효과적으로 구현하기 위해서, 퍼브-서브 시스템은 브로커(broker)를 사용한다[2][3]. 브로커는 퍼블리쉬나 서브스크라이버가 시스템에 접근할 때 사용되는 게이트웨이(gateway)의 역할을 할 뿐만 아니라, 서브스크라이버의 관심사항을 기반으로 관리되는 서브스크립션 테이블(subscription table)을 통해 퍼브-서브 시스템의 메시지 라우팅을 담당한다. 확장성과 유연성을 위하여, 다수의 브로커들은 하나의 네트워크로 구성되어 관리 된다[2][4][5].

브로커 네트워크는 퍼브-서브 시스템의 주요 목적에 따라 다양한 방식으로 관리 된다. 콘텐츠 기반의 퍼브-서브 시스템에서는, 그래프 기반의 라우팅 관리 방식을 사용하여 브로커 네트워크를 관리한다[2][3]. 토픽 기반의 퍼브-서브 시스템에서는, 브로커 네트워크는 사전에 정의된 주제들로 구별되는 트리형태의 논리적 체널을 유지하는 방식으로 관리된다[4][5].

토픽 기반의 퍼브-서브 시스템의 한 가지 예로 위치 기반 퍼브-서브 시스템을 생각할 수 있다. 최근 급격히 늘어난 스마트 모바일 기기의 수요에 발맞춰 다양한 형태의 위치기반 퍼브-서브 시스템들이 연구 되고 있다[6][7][8]. 이러한 위치기반 퍼브-서브 시스템들은 관심 있는 위치 정보를 하나의 토픽으로 설정하는 방식으로 브로커 네트워크를 관리한다. 하지만, 하지만 과거의 위치기반 퍼브-서브 시스템에 대한 연구에서는 서브스크라이버들의 장소이동에 대해서 크게 고려하지 않았다. 즉, 모바일 서브스크라이버들의 빈번한 이동에 따라 그 관심 정보(서브스크립션)또한 빈번하게 바뀔 수 있다는 점에 대해 크게 고려하지 않았다.

최근 퍼브-서브 시스템에서 사용자의 모빌리티를 지원하기 위한 연구들이 있었다[10][11]. 하지만, 이 연구들에서는 일시적인 서브스크립션 변화를 지원하기 위한 보다 효과적인 브로커간의 핸드오프에 대해 집중적으로 논하고 있다.

이 논문에서는 모바일 서브스크라이버들의 장소이동에 따른 빈번한 서브스크립션의 변화에 대해 효과적으로 대응하는 퍼브-서브 시스템을 위한 새로운 브로커 네트워크 관리 방법을 제안한다. 이를 위해 제안하는 퍼브-서브 시스템에서는 사용자의 모빌리티의 정도에 따라 관심 지역을 Slow Moving Region과 Normal Moving Region의 두 가지 타입으로 정의하고, 각 타입에 대한 서브스크립션들을 효과적으로 관리하기 위한 Q+R 트리기반의 브로커 네트워크 관리 방식을 제안한다.

이 논문의 구성은 다음과 같다. 2장에서는 Q+R트리 인덱싱에 대한 개념에 대한 이해를 위해 R트리와 Q트리의 특성 및 Q+R트리의 특성에 대해 소개한다. 3장에서는 Q+R트리 기반의 브로커 네트워크를 구성하기 위해 사용된 기본개념에 대한 소개와, 브로터 네트워크의 구성 방법, 서브스크립션 관리 기법 및 메시지 전송 방법을 기술한다. 제안된 기법들에 대한 성능 검증 결과 및 분석 등을 4장에서 기술하고, 5장에서 이 논문의 결론을 말한다.

II. Q+R 트리 인덱싱

1. R 트리

위치데이터를 이용하는 어플리케이션의 수요가 늘어남에 따라 다 차원의 공간의 데이터를 포함하는 공간적 질의(spatial query)의 빠른 처리를 위한 인덱싱 기법이 필요하게 되었다. 일반적인 B-트리 인덱싱을 사용할 경우 이와 같은 다 차원 공간의 데이터를 처리하기 힘들게 된다. 이러한 필요에 의해 R 트리기법이 제안되었다[12]. 그림 1에서는 R 트리 인덱싱에 대한 예를 나타내고 있다. R 트리에서는 데이터의 밀집 정도에 따라 공간 정보(Spatial Data)를 나타내는 MBR (Minimum Bounding Rectangle)을 구성한다. 각 MBR은 서로 겹쳐지는 것을 허용한다. 이는 보다 다이나믹한 인덱싱을 가능하게 하고, 이를 통해 임의의 공간적 질의에 대한 보다 빠른 응답을 가능하게 한다. 그러나 인덱싱 하는 데이터의 변화가 클 경우에는 R 트리의 성능이 급감한다.

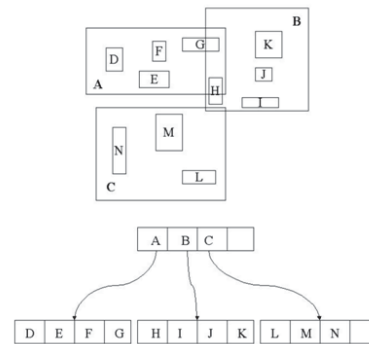


Fig 1. Example of R-Tree

2. Q 트리

Q 트리는 두 개의 차원에서 나타내는 공간 데이터를 임의의 구간을 4조각(Quadrant)으로 나누어 인덱싱하는 기법이다. 그림 2와 같이, Q 트리에서는 전체의 데이터 공간을 연속적으로 4조각으로 나누어감으로써 전체 데이터 공간을 작은 조각들로 인덱싱하는 Divide and conquer방식을 사용한다. 이때, 나누어지는 조각들은 모두 같은 크기를 가진다. 따라서 공간 데이터의 분포에 따라 Q 트리의 높이 및 질의 처리 성능이 달라진다. 만약 공간 데이터가 균등하게 퍼져 있다면 Q 트리의 높이는 모든 리프 노드에 대해서 거의 일정하게 유지 되고, 질의 처리 속도도 임의의 경우에 대해서 일정하게 된다. 반면, 공간 데이터가 가우시안 분포를 따라 임의의 구역에 몰려 있게 된다면, Q 트리의 높이가 불균형이 생기고 질의 처리 속도에 있어서도 큰 편차가 생긴다.

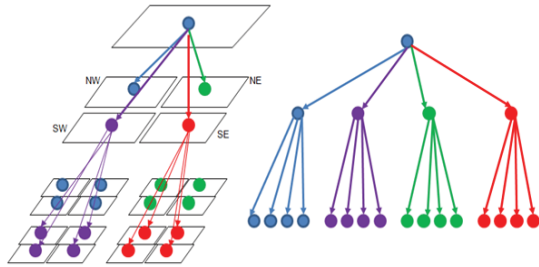


Fig 2. Example of Q-Tree

3. Q+R 트리

R 트리와 Q 트리는 각각 지리적 정보를 다루기 위한 대표적인 인덱싱 기법이다. 하지만 미리 언급한 것과 같이, R트리는 공간 정보가 좁은 지역에 밀집해있을 경우 보다 효과적인 데이터 관리가 가능해지는 반면, 빈번한 정보 변화에 대해서는 취약하고, Quad트리는 공간 정보가 넓게 퍼져 있는 경우에 보다 규격화된 정보 관리 방식을 통해 효과적인 데이터 관리가 가능하다. 이와 같은 장단점들을 보완한 데이터 구조로 Q+R 트리가 제안 되었다[9].

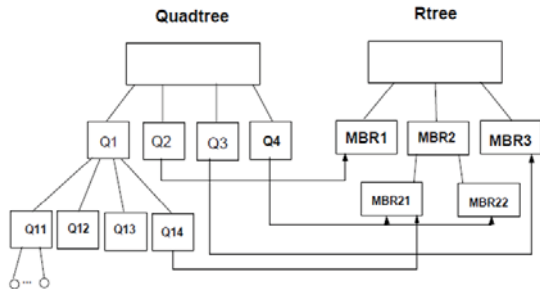


Fig 3. Example of Q+R-Tree

그림 3과 같이 Q+R트리에서는 Quad트리와 R트리가 공존한다. Quad트리와 R트리의 루트는 서로 같은 공간 데이터 영역을 관리 하고, 각 트리의 관리법에 따라 공간 데이터 영역을 분배하여 하위 트리 블록들을 생성 한다. Quad트리와 R트리의 하위 트리 블록들이 담당하는 공간 데이터 영역이 서로 겹치는 경우 각 블록간의 크로스 링크를 설정한다. 예를 들면 Quad트리의 Q2블록이 담당하는 공간 데이터 영역이 R트리의 MBR1블록이 담당하는 공간 데이터 영역과 겹치기 때문에 크로스 링크가 설정된다.

III. Q+R 트리 기반 퍼브/서브 시스템

1. 모빌리티 인지 공간 데이터 영역 분리

위치를 지속적으로 바꿀 수 있는 모바일 사용자를 위한 퍼브/서브 시스템에서는 서브스크립션의 지속적인 변화를 효율적으로 인덱싱하기 위한 기법이 필요하다. 이를 위해 우선 사용자들의 위치정보의 변화에 대한 특성을 살펴본다.

사용자들의 위치정보의 변화를 살펴보면, 그 변화가 자주 변하지 않는 영역과 그 변화가 빈번하게 변하는 영역으로 나누어 생각 할 수 있다. 빌딩들이 운집해 있는 다운타운 지역과 같이 사용자들의 위치정보가 비교적 긴 시간 동안 변하지 않는 Slow Moving Region에 해당하는 공간 데이터 영역을 생각할 수 있다. 반면 자동차 및 버스를 타고 빠르게 이동하는 사용자들은 Fast Moving Object로 생각할 수 있다. 일반적으로 Fast Moving Object는 Slow Moving Region의 다른 부분에서 주로 관찰할 수 있다. 따라서 전체 데이터 공간 중 Slow Moving Region의 다른 부분은 Normal Moving Region이라 할 수 있다. 이러한 공간 데이터 영역의 구분은 해당 공간에 해당하는 지리적 정보를 분석하거나, 해당 동간 존재하는 사용자들의 행동 패턴을 분석하여 얻어 낼 수 있다.

이 Slow Moving Region에서는 사용자들의 이동속도가 낮고 많은 사용자들이 밀집해 있기 때문에, R트리 기반의 데이터 관리방식이 알맞다. 반면 Normal Moving Region에서는 사용자들의 이동이 빈번하게 일어날 가능성이 높고 사용자의 밀도도 Slow Moving Region에 비해서 상대적으로 낮기 때문에, 규격화된 트리 관리를 통한 효과적인 정보 관리를 지원하는 Q트리 기반의 데이터 관리 방식이 알맞다.

2. Q+R 트리 기반 브로커 네트워크

앞서 언급한 것과 같이, Slow Moving Region과 Normal Moving Region은 각각 R 트리와 Q 트리에 의해서 관리하는 것이 효과적이다. 이에 따라 이 논문에서 제안하는 모바일 사용자를 위한 퍼브/서브 시스템에서는 Q+R 트리를 사용한 공간 데이터 인덱싱을 응용한 브로커 네트워크 관리 기법을 제안한다. Q+R트리 기반의 브로커 네트워크를 구성하기 위해서는 R트리 기반의 브로커 네트워크와 Q트리 기반의 브로커 네트워크를 우선적으로 구성한 후 각 트리의 MBR을 담당하는 브로커간의 링크를 연결 한다.

Q+R트리 기반 브로커의 구성은 R트리의 구성으로부터 시작한다. R트리의 루트(Root) MBR은 전체 데이터 공간을 대표하고, 말단(Leaf) MBR은 임의의 Slow Moving Region을 대표한다. R트리의 임의의 MBR에 포함되는 Slow Moving Region의 개수가 정해진 임계값을 넘어가는 경우, MBR은 다수의 중간(Intermediate) MBR로 분리된다. 그림4의 경우 전체 데이터 영역은 R1, R2의 중간 MBR로 분리되고 그 하부에 말

단 MBR들이 존재하는 R트리로 표현이 가능하다. 이 R트리에서는 말단 MBR에 해당하는 브로커들만 실제적으로 사용들과 연결되어 있고 중간 MBR들은 트리의 구성을 위해 사용된다.

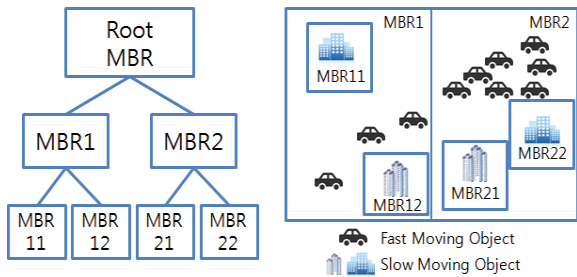


Fig 4. R Tree Construction with Slow Moving Object

R트리가 구성된 이후 Q트리의 구성이 진행 된다. R트리의 루트 MBR은 Q트리의 루트 MBR로 사용된다. Q트리의 MBR은 각 MBR에 포함되는 사용자의 수가 정해진 임계값을 넘어가게 되면 4개의 하부 MBR들로 분리 된다. 이때, Q트리에서 계산하는 사용자의 수에는 R트리 구성 시 사용된 Slow Moving Region에 포함된 사용자들은 고려되지 않는다. R트리와 마찬가지로, Q트리에서도 말단 MBR에 해당하는 브로커들만 사용자들과 실제적으로 연결되어 있고, 다른 MBR들은 트리의 구성을 위해 사용된다.

R트리와 Q트리의 구성이 완료되면, Q+R트리의 구성을 위해 R트리의 말단 MBR과 이에 해당하는 Q트리의 말단 MBR들을 연결한다. 즉, 그림 5와 같이 R트리의 말단 MBR에 해당하는 데이터 공간과 겹치는 Q트리의 말단 MBR들을 모두 찾아 서로 연결한다. 이에 따라, R트리 말단 MBR과 Q트리 말단 MBR간엔 1:N 맵핑 관계가 생긴다.

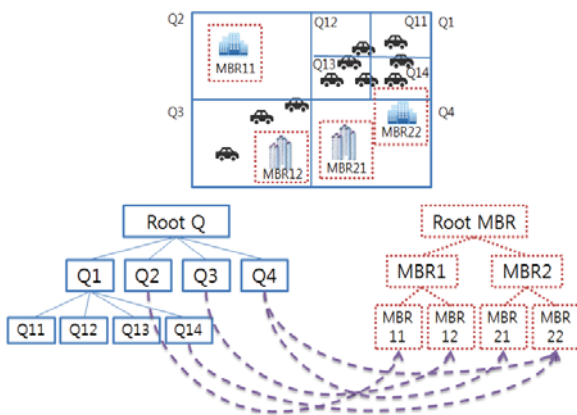


Fig 5. Example of Q+R Tree Construction

3. 서브스크립션 관리

Q+R트리의 각 트리 블록들에는 각 트리의 관리법에 따라 지정된 공간 데이터 영역을 담당하기 위한 브로커가 할당된다. 이렇게 할당된 브로커는 해당 공간 데이터 영역에 관심을 두고 있는 서브스크라이버들을 관리한다. 즉, 해당하는 서브스크립션 테이블을 관리한다. 만약, 한 브로커에서 담당하는 서브스크라이버의 개수가 너무 많아지게 되면, 하위 트리 블록들을 만들어 담당하던 서브스크라이버를 각 하위 트리 블록에 분배한다. 새로 생성된 트리 블록에 대해서도 브로커가 할당 된다. 이 논문에서 명시된 브로커는 하나의 독립적인 머신 뿐만 아니라 대규모 클러스터 시스템에서 제공되는 하나의 워커(Worker)로 볼 수 있다. 이와 같은 트리 관리를 통해, 한 브로커에 큰 로드가 몰리는 상황을 막고, 전체 브로커의 로드를 평균화 할 수 있다.

서브스크라이버가 모빌리티 인지 퍼브-서브 시스템에 관심 지역을 등록할 때, 우선 Q+R트리의 R트리를 사용하여 관심 지역의 Slow Moving Region로의 포함 여부를 확인한다. 즉, 관심 지역이 R트리의 임의의 하위 MBR이 담당하는 영역 내에 존재할 경우, 이 관심 지역은 Slow Moving Region에 포함된다고 하고, 해당 MBR을 담당하는 브로커가 이 관심 지역에 대한 서브스크립션을 저장한다. 만약 관심 지역이 어떤 MBR에 관련된 영역에 대해서도 포함되지 않는다면 이 관심 지역은 Normal Moving Region에 포함된다고 한다. 이 경우에는, 관심 지역을 포함하는 Quad트리의 말단 MBR을 찾고 해당 브로커에 이 관심 지역에 대한 서브스크립션을 저장한다.

4. 데이터 전송

구성된 Q+R트리 기반의 브로커에서 임의의 위치 정보와 관련된 데이터가 퍼블리싱 되면, 이 데이터는 해당 위치정보에 관심 있는 사용자에게 전달되어야 한다. Q+R 트리에서의 데이터 전송은 기본적으로 Q트리를 사용하여 이루어진다. Q+R트리 기반의 브로커 네트워크에서 데이터 전송은 임의의 위치정보를 담은 데이터를 Q트리의 루트 MBR에 전송하며 시작한다. Q트리의 루트 MBR은 위치정보 데이터를 받으면 우선 자신과 연결된 R트리의 MBR이 있는지를 확인하고, 만약 연결된 R트리의 MBR이 해당 위치정보를 포함한다면 R트리의 MBR로 데이터를 전송한다. 만약 연결된 R트리 MBR이 없고, Q트리의 하부의 MBR이 해당 위치정보를 포함한다면, 데이터는 하부의 MBR로 전송된다. 이와 비슷한 방식으로 Q트리의 루트 MBR에서 시작한 데이터는 Q트리 또는 R트리의 말단 MBR을 만날 때까지 중간 MBR들을 통해 포워딩 된다

Q+R트리의 메시지는 Q트리에서 R트리 쪽으로 전달된다. Q+R트리의 메시지 전송을 R트리에서 Q트리 쪽으로 선택하지 않은 이유는 최악의 경우에 발생하는 비용이 클 수 있기 때문이다. 만약 구성된 R트리의 높이가 N일 경우, R트리에서 시작된 데이터 전송은 최악의 경우 $O(N)$ 의 비교를 통해 해당 위치정보가 R트리에 존재 하는지를 확인하게 된다. 만약 R트리에 해당 위치정보를 포함하는 MBR이 없을 경우 $O(N)$ 의 비교를 마치고 다시 Q트리의 루트 MBR부터 데이터 전송을 다시 시작해야 한다. 반면 높이가 M인 Q트리에서 데이터 전송을 시작할 경우, 최악의 경우 $O(M)$ 의 비교를 통해 해당 위치정보를 포함하는 Q트리의 말단 MBR을 찾게 된다. 따라서, 메시지 전송이 R트리에서 시작될 경우의 최악의 코스트는 $O(N)+O(M)$ 이 되어, Q트리에서 시작되는 메시지 전송의 최악의 코스트 $O(M)$ 보다 추가적인 오버헤드를 가지게 된다. 또한 Q트리에서 시작되는 메시지 전송은 R트리와의 연결을 통해 최악의 코스트 $O(M)$ 이 발생할 확률을 줄여주는 점을 감안한다면, 데이터 전송을 R트리에서 시작하는 것이 더욱 비효율적이라고 할 수 있다.

IV. 성능 평가

Q+R트리를 사용한 브로커 및 서브스크립션 관리가 사용자의 모빌리티를 인지하는 위치기반 퍼브-서브 시스템에 미치는 영향을 확인하기 위해서, 시뮬레이션 기반의 실험을 수행 하였다. 실험을 수행한 위치 기반 퍼브-서브 시스템은 임의의 크기의 사각형 모양의 공간 데이터 영역을 지원하는 시스템이다. 퍼블리쉬들은 시스템이 제공하는 공간 데이터 영역의 임의의 위치에 위치하고, 임의의 크기의 사각형 공간 데이터 영역에 대한 이벤트를 발생시킨다. 서브스크라이버들은 시스템이 제공하는 공간 데이터 영역 내에서 임의의 주어진 속도로 지속적으로 이동하면서, 주기적으로 (20 ms) 현재의 위치에 관련된 모든 이벤트를 수신한다. 이때 각 서브스크라이버의 이동속도는 전체의 절반정도는 느리게 움직이고 나머지 절반은 약 세배의 속도로 움직인다고 가정하였다.

이러한 퍼브-서브 시스템에서 R트리, Quad트리, 그리고 Q+R트리를 사용하여 브로커 네트워크를 구성 하였을 때, 시스템에 미치는 영향을 관찰하였다. 이를 위해 Number of Required Brokers, Retransmission Ratio, Number of Handover 그리고 Number of Multicast Messages를 확인하였다. Number of Required Brokers는 각 트리 기법을 사용하였을 경우 시스템에서 필요한 브로커의 개수이다. Retransmission Ratio는 전체 이벤트 메시지 전송 트래픽에 대한 서브스크라이버의 이동에 따른 이벤트 재전송시 사용된 트래픽

의 비율이다. 서브스크라이버가 해당 위치를 바꾸게 되면 서브스크립션이 저장되는 브로커의 위치를 바꾸게 되는데, 이때 브로커 간에 발생하는 이벤트 포워딩 또는 루트로부터 다시 전송되는 이벤트 메시지들을 재전송 트래픽으로 간주한다. Number of Handover는 사용자가 연결되어있는 게이트웨이를 바꾸는 횟수를 나타낸다. 이 값이 클수록 서브스크립션 관리가 복잡해짐을 의미한다. 구성된 브로커 네트워크가 보다 사용자의 행동양식을 잘 이해 할수록 이 값은 작아진다. Number of Multicast Message들은 데이터 퍼블리싱 시 사용된 총 데이터 메시지의 개수를 나타낸다. 이 인자를 통해 구성된 브로커가 전체 네트워크 트래픽에 미치는 영향을 알 수 있다. 인자들이 각 트리 기법을 사용한 브로커 네트워크가 퍼브-서브 시스템의 확장성과 유연성을 얼마나 지원할 수 있는지를 관찰하기 위해 서브스크라이버의 수 및 시스템 런 타임을 변경시켜가며 각 인자들을 측정하였다.

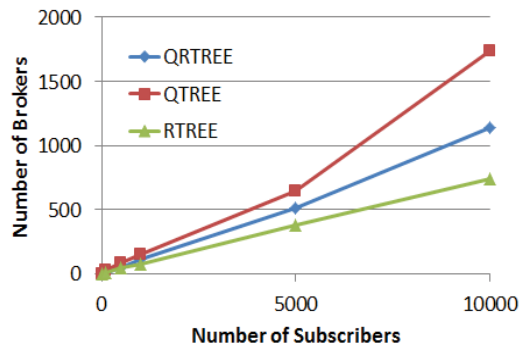


Fig 6. Number of Brokers as a function of Number of Subscribers

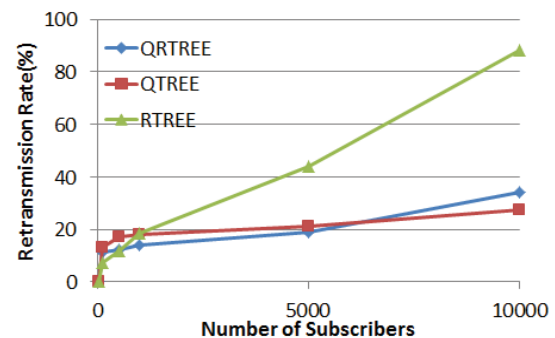


Fig 7. Retransmission Rate as a function of Number of Subscribers

그림 6에서는 서브스크라이버 수의 변화에 따른 필요한 브로커 수의 변화에 대한 결과를 나타내고 있다. 즉 QTREE를 사용할 경우 가장 많은 브로커를 사용하게 되고 RTREE를 사용할 경우 가장 적은 브로커를 사

용하게 된다. Q+R트리를 사용할 경우 Slow Moving Region에 포함되는 다수의 서브스크라이버들을 R트리를 사용해 관리하게 된다. 이를 통해, Q+R트리 사용시, 필요한 브로커의 개수는 Q트리를 사용했을 경우에 비해 35%정도 적음을 확인 하였다.

그림7에서는 서브스크라이버 수의 변화에 따른 Retransmission Ratio의 변화에 대한 결과를 나타내고 있다. 필요한 브로커의 개수에 대한 결과와는 반대로 R트리를 사용하였을 경우 Retransmission이 더욱 빈번히 일어나게 되고 서브스크라이버의 수가 늘어날수록 재전송시 필요한 트래픽의 양이 급격히 늘어나는 것을 확인 할 수 있었다. R트리는 사용자들이 비교적 좁은 구역에서 자주 움직이지 않는 경우를 위해 사용되기 때문에, 이와 같이 지속적으로 서브스크립션이 변하는 상황에는 R트리가 적합하지 않다는 것을 확인 할 수 있다. 반면, Q+R트리를 사용할 경우 Q트리의 구조적 특징을 사용해서 지속적인 서브스크립션 변화에 필요한 Retransmission Ratio를 낮출 수 있었다.

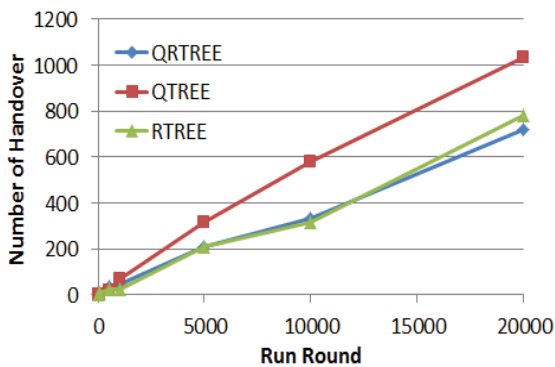


Fig 8. Number of Handover as a function of Run Round

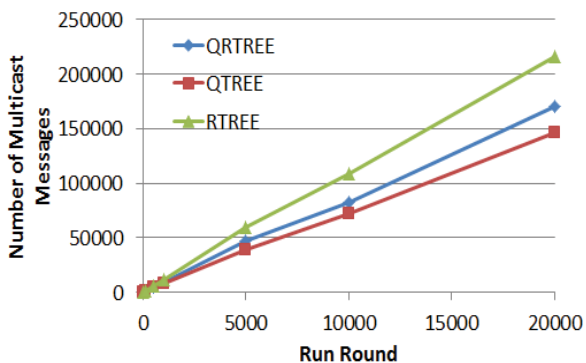


Fig 9. Number of Multicast Messages as a function of Run Round

그림 8에서는 시스템의 런 타임에 따른 사용자의

핸드오버의 수를 측정된 결과이다. 이 결과 Q트리에서 사용자의 핸드오버가 가장 많이 발생하고, R트리와 Q+R트리는 서로 비슷 한 정도의 값을 가지는 것을 확인 하였다. 이를 통해 사용자의 이동 패턴을 이해하는 R트리방식의 인덱싱을 사용할 경우 핸드오버의 수를 줄일 수 있음을 확인 할 수 있었다. 하지만 이러한 다이나믹한 방식의 인덱싱방식은 메시지의 수를 증가시킬 수 있다. 그림 9에서는 시스템의 런타임에 따른 메시지 사용량을 나타내고 있다. 이 결과 R트리에서 발생하는 메시지가 가장 많음을 알 수 있고, Q트리에서 발생하는 메시지의 양이 가장 작음을 알 수 있다. 즉 고정된 구조인 Q트리에서는 보다 일정한 양의 메시지를 통해 데이터 전송이 가능한 반면 다이나믹하게 인덱싱이 변하는 R트리의 경우 예기치 않은 메시지를 사용하게 되는 경우가 많아져 네트워크 트래픽을 증가시킬 수 있다는 점을 확인 하였다.

이러한 결과를 통해 Q+R트리는 Q트리와 R트리의 장점들을 수용해, 모바일 사용자를 위한 브로커 네트워크를 구성했음을 확인 할 수 있었다.

V. 결론

모바일 스마트 기기가 위치기반 퍼브-서브 시스템에서 사용될 경우, 사용자의 이동에 따라 장소 정보와 관련된 서브스크립션이 지속적으로 변화하게 된다. 이러한 지속적으로 변화 하는 서브스크립션을 다루는 모빌리티 인지 퍼브-서브 시스템에서 브로커 네트워크를 사용할 경우, 사용자의 이동성을 고려하여 브로커 네트워크를 효과적으로 구성함으로써 시스템에서 필요한 리소스를 줄일 수 있음을 확인 하였다.

이 논문에서는 사용자의 이동성을 두가지 타입으로 구분한 후, 서브스크립션의 변화가 더딜 것으로 예상되는 Slow Moving Region을 R트리방식으로 관리하고 그 외의 Normal Moving Region을 Quad트리 방식으로 관리하도록 하는 Q+R트리 방식의 브로커 네트워크 관리 방식을 제안 하였다.

향후 연구 방향으로는 보다 다양한 사용자의 이동성을 고려한 브로커 네트워크에 대한 연구, 제한된 브로커 리소스를 사용하였을 때 지속적 서브스크립션 변화에 대응하기 위한 브로커 네트워크 관리 방법에 대한 연구 등이 있다.

REFERENCES

- [1] Andrew S. Tanenbaum and Marrteen Van Steen, "Distributed systems principles and paradigms", 2006
- [2] R. Baldoni, R. Beraldi, L. Querzoni, and A. Virgillito. Efficient publish/subscribe through a self-organizing broker overlay and its applica

- tion to SIENA. *The Computer Journal*, 2007.
- [3] Pietzuch, P. and Bacon, J. (2002) Hermes: A distributed event-based middleware architecture. *Proceedings of the 22nd International Conference on Distributed Computing Systems Workshops (DEBS)*, Vienna, Austria, 2-5 July, pp. 611- 618. IEEE Computer Society, Washington.
- [4] Terpstra, W. W., Behnel, S., Fiege, L., Zeidler, A., and Buchmann, A. P. (2003) A peer-to-peer approach to content-based publish/subscribe. *Proceedings of the second International Workshop on Distributed Event-Based Systems*, San Diego, California, USA, 8 June, pp. 1-8. ACM, New York.
- [5] Rowston, A., Kermarrec, A., Castro, M., and Druschel, P. (2001) SCRIBE: The Design of a Large-Scale Notification Infrastructure. *Proceedings of the third International Workshop on Networked Group Communication*, London, UK, 7-9 November, pp. 30-43. Springer-Verlag, Berlin.
- [6] Eugster, P.T., Garbinato, B., Holzer, A.: Location-based publish/subscribe. In: *NCA '05: Proceedings of the Fourth IEEE International Symposium on Network Computing and Applications*, pp. 279-282. IEEE Computer Society Press, Washington (2005)
- [7] J. H. Ahn, Uichin Lee, and H. J. Moon, Geo Serv: A Distributed Urban Sensing Platform, *Proc. CCGrid 2011*.
- [8] K. Kim, Y. Zhao, and N. Venkatasubramanian, GSFord: Towards a Reliable Geo-Social Notification System, *Proc. SRDS 2012*
- [9] Yuni Xia, Sunil Prabhakar, Q + R tree: efficient indexing for moving object databases, in: *Proceedings of Eighth International Conference on Database Systems for Advanced Applications DASFAA, 2003*, pp. 175-182
- [10] L. Li, A. Gaddah, and T. Kunz, "Mobility Support in a Tactical P2P Publish/Subscribe Overlay", *Proceedings of the 27th International Conference for Military Communication, (MILCOM2008)*, San Diego, CA, USA, November 2008.
- [11] J. Wang, J. Cao, J. Li, and J. Wu, "MHH: A Novel Protocol for Mobility Management in Publish/Subscribe Systems", *Proceedings of the 2007 International Conference on Parallel Processing (ICPP'07)*, IEEE Computer Society, September 2007, Washington, DC, pp. 54-61
- [12] Sebastian kabisch, R-Tree Algorithms and Datastructures for Database Systems *SS 2003*
- [13] Myung-guk Lee, Kyungbaek Kim, Q+R tree based broker network for efficient mobility aware pub-sub system, *Proceedings of 38th KIPS 2012 fall conference*, November 2012, Jeju, Republic of Korea.

 저 자 소 개



이 명 국

2005년 한신대학교 정보통신학과 학사 졸업

2005년~2008년 Dasan Network L2/L3 Switch Firmware Engineer

2012년 University of California Irvine 석사 졸업

2012년~2014년 AnyData Firmware Engineer

2014년~현재 Western Digital Firmware Engineer
<주관심분야 : IoT, BigData, WebRTC>

김 경 백 (정회원)

1999년 한국과학기술원 전기 및 전자공학과 학사 졸업

2001년 한국과학기술원 전기 및 전자공학과 석사 졸업

2007년 한국과학기술원 전기 및 전자공학과 박사 졸업

2007년~2011년 University of California Irvine, 박사후 연구원

2012년~현재 전남대학교 전자컴퓨터공학부 조교수
<주관심분야 : 분산시스템, 미들웨어, 피어투피어/오버레이 네트워크, 소셜 네트워크, SDN>