

애드 혹 퍼베이시브 환경에서 자원 모니터링을 위한 환경에 적응적인 광고 기법

곽경만⁰ Gonzalo Huerta-Canepa 이동만

한국정보통신대학교 분산협동시스템 및 네트워크 연구실

{kmkwak, ghuerta, dlee}@icu.ac.kr

Adaptive Advertisement for Resource Monitoring in Ad Hoc Pervasive Environment

Kyungman Kwak⁰ Gonzalo Huerta-Canepa Dongman Lee

Information and Communications University CDSN laboratory

요 약

최근의 개인 휴대 장치들의 급속한 발전에 힘입어 pervasive computing 환경은 인프라 시스템의 제약에서 벗어나 개인 장치들의 협력에 의해서 상황에 맞는 서비스를 제공 받을 수 있도록 변하고 있다. 하지만 이러한 ad hoc pervasive 환경은 기존의 스마트 스페이스에서 보다 동적인 변화가 더욱 많은 환경이기 때문에 pervasive computing을 위한 기능들 중에서도 자원 관리가 가장 중요한 역할을 해야 한다. Pervasive computing 환경에서 자원 관리는 자원 모니터링(Resource Monitoring), 자원 발견(Resource Discovery), 자원 할당(Resource Allocation), 자원 적응(Resource Adaptation)의 4가지의 주요 기능으로 구분 될 수 있고, 동적인 변화가 많은 환경에서는 무엇보다도 자원 모니터링이 가장 중요시 되어야 한다. 자원 모니터링에서 정보의 수집은 pull이나 push 방식으로 이루어질 수 있는데, pull 방식은 사용자 요구에 맞춰 요구 했을 때의 가장 최신의 정보를 모아 줄 수 있다. 따라서 이는 자원 발견에 가장 적합하지만 언제 정보 수집을 요청할지는 정보 소비자의 입장에서는 알 수 없기 때문에 push 방식이 다른 여러 기능들을 지원하기 위해서 보다 적절하다. 하지만 대부분의 push 방식은 주기적으로 자신의 정보를 광고하는 방식으로 이루어 진다. 하지만 상황 적응(adaptation)입장에서 봤을 때 특정 수준의 민첩성을 요구하기 위해서는 광고주기를 조절 해야 하고, 이때 필요 이상으로 네트워크의 사용량을 늘릴 수 있다. 뿐만 아니라 변화가 많은지 적은지 등의 각 단말의 상황은 무시된 채 모든 단말들이 동일한 주기로 정보를 광고한다. 이러한 문제점을 해결 하기 위해서 본 논문은 자원 정보 제공자의 상황을 고려한 자원 정보 광고 기법을 제안한다.

1. 서 론

최근 들어 개인들의 휴대 장치들의 급속한 발전에 힘입어 pervasive computing 환경은 특정한 장소의 인프라 시스템의 제약에서 벗어나서 점점 분산된 개인들에 의해서 통제 될 수 있는 ad hoc pervasive 환경으로 옮겨가고 있다. 기존의 pervasive computing 환경은 대부분 스마트 홈이나 스마트 오피스 등과 같은 특정 공간에서 pervasive한 서비스를 제공하기 위한 것이었다. 그렇기 때문에 각각의 스마트한 공간에는 서비스를 제공하기 위한 시스템을 갖추고 센서와 액추에이터를 통해서 사용자에게 서비스를 제공한다. 하지만 이러한 pervasive 서비스는 위와 같은 공간의 인프라 시스템에 제약을 받아서 시스템이 갖춰지지 않은 다른 공간에서 사용자는 서비스를 받을 수 없다. 따라서 pervasive computing 환경은 특정 공간의 인프라 시스템에 의해서가 아닌 분산되어 있는 각각의 단말들의 협력에 의해서 이루어 져야 한다. 이러한 환경에서는 어느 공간에서건 각각의 단말들의 협력에

의해서 서비스를 제공 받을 수 있다. 뿐만 아니라, 근래의 개인 휴대 단말들의 컴퓨팅이나 다른 자원들이 있어서 급속한 발전은 ad hoc pervasive computing 환경을 더욱 부추기고 있다.

인프라 시스템에 의해 통제되던 pervasive computing 환경에서 단말들간의 협력에 의해 이루어지는 ad hoc pervasive 환경으로의 변화는 pervasive computing을 위한 미들웨어에도 많은 변화를 몰고 온다. 대표적으로 자원 관리를 들 수 있다. Ad hoc pervasive 환경에서 단말들의 잦은 이동과 자원의 제약성은 자원 관리를 기존의 환경에서 보다 더욱 부각시키는 요인이 된다. 뿐만 아니라 여러 사용자들의 공유된 자원에 대한 경쟁이 불가피 하기 때문에 이 환경에서 자원관리는 pervasive 서비스를 위한 시작과 끝이라고 해도 과언이 아닐 것이다.

기본적으로 pervasive computing 환경에서는 자원 관리는 독립적인 컴포넌트로 구분되어지지 않는 경우가 많다. 중앙의 시스템에 의해 모든 것이 관리되고 ad hoc pervasive 환경처럼 동적인 상황이 많지 않기

때문에 자원 관리에서 필요한 특정 기능들은 다른 컴포넌트에 포함되는 경우가 많다. 하지만 여러 시스템들을 종합해 볼 때 pervasive computing 환경에서 자원 관리는 크게 4가지 기능으로 나뉘 볼 수 있다. 첫 번째는 자원 모니터링(Resource Monitoring)이고, 다음은 자원 발견(Resource Discovery), 그리고 자원 할당(Resource Allocation), 마지막으로 자원 적응(Resource Adaptation)이다. 자원 모니터링은 자원 관리의 가장 기본이 되는 기능으로 주변의 단말들의 연결 상태부터 시작해서 각 단말의 자원 상태를 수시로 체크한다. 이를 통해 사용자에게 가장 적절한 자원을 발견 할 수 있다. 뿐만 아니라, 발견된 자원을 요청하고 요청된 자원을 할당한 후에도 할당된 자원에 대한 꾸준한 모니터링을 통해서 원하는 서비스가 계속 제공 될 수 있는지를 검사한다. 모니터링을 통해 할당된 자원의 상태 변화나 더 좋은 자원의 등장을 알게 되면 자원 적응 과정을 통해 새롭게 자원을 할당 받을 수 있다. 이렇듯 4가지의 자원관리의 기본기능들이 잘 어우러져 pervasive computing 환경에서 안정적인 자원관리가 가능하게 하고 있다.

하지만 ad hoc pervasive 환경으로 변화는 4가지의 기본 기능 중에서도 자원 모니터링을 가장 중요한 기능으로 만들고 있다. 자원 모니터링은 나머지 3가지 기능을 위한 밑바탕이 될 뿐만 아니라, ad hoc 환경의 동적인 상황들에 대처하기 위해서 꼭 필요한 기능이다. 기본적으로 Mobile ad hoc 환경은 여러 가지 특징을 가지지만 그 중에서도 이동성에 의한 잦은 토폴로지의 변화와 링크의 용량의 변동, 휴대용 단말이라는 제약에 의한 CPU나 메모리의 용량의 제약은 자원의 유효성(availability)를 자주 변화시켜 안정적인 서비스를 힘들게 한다. 이를 방지하기 위해서도 자원 모니터링은 자원 관리의 여러 기능 중에서도 가장 우선시 되어야 한다.

자원 모니터링은 과거에도 여러 시스템에서 광범위하게 활용되어 왔다. 하지만 대부분 네트워크 레벨의 특정 자원을 모니터링 하여 프로토콜에 활용되거나 특정 응용을 위해 사용되었다. 예를 들어 QoS 라우팅이나 Multimedia Streaming을 위한 모니터링들이 대표적이다. 특히 pervasive computing 환경에서의 여러 시스템들도 자원 모니터링을 위한 기능을 가지고 있다. 하지만 대부분 다른 기능들을 위한 내부 기능으로 사용되어 해당 기능에 종속적이다. 뿐만 아니라 기존의 이와 같은 자원 모니터링은 대부분 유선 환경에서 인프라의 서비스의 지원을 받거나 중앙의 특정 서버에 의해서 관리되었다. 하지만 이러한 것들은 ad hoc pervasive 환경에서는 불가능 한 것들이다. 그래서 몇몇 MANET 환경에서 자원 모니터링을 위한 시스템에서는 각각의 단말들에 단말들간의 협력에 의한 자원 모니터링을 위한 모듈들을 가지고 있다.

이처럼 자원 모니터링은 환경이나 시스템의 특징에 의해 다양한 방법으로 지원된다. 하지만 ad hoc pervasive 환경도 pervasive computing을 기반으로 하기 때문에 무엇보다 가장 중요한 기능은 상황에 적응(adaptation) 이라고 할 수 있다. Pervasive computing의 특성상 상황의 변화를 감지하고 해당 상황에 가장 알맞은 서비스를 사용자에게 제공하는 것이 가장 중요한 기능이기 때문이다. 게다가 ad hoc pervasive 환경의 특성상 더욱 동적인 상황이 많이 생기기 때문에 더욱 상황 적응은 더욱 필수적이다. 이러한 적응을 위해서 자원 모니터링은 빠질 수 없는 요인이다. 자원 모니터링을 통해서 변화된 상황에 더욱 빠르게 적응을 할 수 있기 때문이다.

대부분의 자원 모니터링은 이를 위해 주변의 자원 정보를 수집하는 방법들을 가지고 있다. 대표적으로 크게 2가지로 나눌 수 있고, 이는 Pull과 Push 방식이다. 우선 pull 방식은 정보를 수집하는데 있어서 정보 소비자가 주변에 정보 제공을 요청하는 방식으로 사용자의 요구에 의해 이루어 질 수 있어서 가장 최신의 정보를 적시에 받아 볼 수 있다. 따라서 이는 자원 발견에 유용하게 활용될 수 있다. 하지만 상황 적응 기능 입장에서는 잘 맞지 않는 방식이다. 왜냐하면 정보의 소비자의 입장에서 pull방식을 환경 변화의 적응에 활용하려면 언제 요청을 보내야 할지를 알아야 하는데, 소비자 입장에서는 정보 제공자가 언제 제공자의 자원 상태가 변화되는지 알 수가 없기 때문이다. 뿐만 아니라 자원을 제공하는 것은 정보 제공자인데도 정보 제공자의 상황이 모니터링에 전혀 고려 되지 않고 있다. 따라서 자원 발견이나 상황 적응 모두에서 자원 모니터링이 이루어지기 위해서는 push 방식이 적절하다. 이러한 push 방식은 적절한 적응을 위한 적절한 민감성을 유지하는 것이 중요한데, 대부분의 push 방식이 주기적인 정보의 광고를 통해서 이를 해결 하고 있다. 하지만 이런 상황에서 더 높은 민감성을 요구하기 위해서는 더 짧은 주기를 요구하게 되고, 그렇게 되면 네트워크의 정보의 소통 량이 너무 많아지게 되어 mobile ad hoc 환경에 무리를 주게 된다. 뿐만 아니라 이러한 일괄적인 주기적인 정보의 광고는 정보를 제공하는 특정 단말에는 적절한 주기를 제공할 지 모르지만 다른 특정 단말에는 너무 민감하거나 너무 둔감한 주기를 제공할 수 도 있다. 본 논문에서는 이와 같은 상황을 해결 하기 위해서 자원 정보의 제공자의 상황을 고려한 자원 정보 광고 기법을 제안한다.

2. 관련 연구

자원 모니터링은 기존의 여러 환경에서 각자의 시스템의 특징에 맞게 활용되어 왔다. 그 중에서 대표적으로 전통적인 유선 인프라 환경과 거기에 최근의 pervasive computing이 적용된 환경, 그리고

MANET 환경에서 자원 모니터링 기법이 어떻게 사용되고 있는지 알아 보았다.

전통적인 유선 환경에서 대표적인 자원 모니터링 시스템인 Remos[5]는 네트워크 상황을 고려한 응용을 위한 자원 모니터링 시스템을 제공한다. Remos는 기본적으로 중앙 집중식으로 중앙의 서버에서 네트워크의 자원 정보들을 수집(Collector)하고 이 정보가 필요한 각 장치들이 정보를 가져간다(Modeler). 이 정보 모니터링 시스템은 다른 모듈들과 독립적으로 운영되며, 정보의 수집은 유선 인프라 환경이 제공하는 서비스인 SNMP 쿼리에 의해서 이루어진다. 즉 주기적인 pull 방식으로 자원 정보를 얻고 있는 것이다.

다음으로 이러한 유선 인프라 환경에 pervasive computing을 제공하는 환경에서의 자원 모니터링 시스템에 대해 살펴보면, Odyssey[6]와 USS Monitor[7]를 들 수 있다. Odyssey는 Aura project에서 환경에 적응(adaptation)을 위한 내부 컴포넌트로서 적응을 위한 세부 기능 중에 하나로 자원 모니터링을 이야기하고 있다. 이 시스템에서는 우선 자원이 할당된 뒤에 할당된 자원에 대한 모니터링을 이야기하고 있고, 사용자가 자원 제공자에게 직접 언제 어떤 상황에서 자원 정보를 알려줄지에 대한 정보를 제공한다. 하지만 이는 환경에 적응이라는 기능에 너무 종속적으로 다른 여러 기능들을 지원하는 모니터링 시스템이 되기에는 우리가 있다. 뿐만 아니라 더 좋은 자원이 주변에 나타났을 때에 그 상황에 적응하는 것도 어렵다. USS Monitor는 이와는 달리 독립적인 자원 모니터링 시스템을 제공한다. 이 시스템도 기본적으로 Remos와 비슷한 구조로 중앙에서 정보를 수집해서 주변에서 활용하도록 하는 것은 같은 방식이지만, 자원 정보가 네트워크 자원에 국한 되거나 인프라가 제공하는 서비스를 이용하지 않는다. 직접 Request/Reply 방식으로 주변의 장치들로부터 직접 정보를 수집하고 자원 정보 또한 네트워크 자원 정보뿐만 아니라, CPU 나 메모리 내부 자원 정보 그리고 응용의 상태에 대해서 까지 정보를 수집한다.

마지막으로 MANET 환경에서는 Friend Relay[8]와 [9]에서 자원 모니터링 시스템의 예를 찾아 볼 수 있다. Friend Relay에서는 MANET 환경에서 자원의 분산된 주변의 단말들의 자원을 효율적으로 사용하기 위해서 자신의 자원 정보를 주기적으로 광고한다. 광고를 통해 정보를 수집한 각 단말들은 이를 바탕으로 주변의 자원 사용을 조절한다. [9]에서는 보다 다양하게 MANET 환경에서 자원 모니터링을 하기 위해 고려 되어야 할 사항들에 대해서 살펴보고 효율성은 유지하면서도 네트워크 사용량을 줄이는 것을 주요한 목표로 자원 모니터링 시스템을 설계하였다. 또한 여러 자원들의 각각의 특성을 반영할 수 있도록 기본 모니터링 인프라를 제공하고 내부에 각 자원들을 위한 서브 모듈을 둬으로써 시스템에 유연성을 제공하였다.

종합해 보면, ad hoc pervasive 환경에서는 Remos와 같이 중앙 집중식이나 인프라가 제공하는 서비스를 제공 받을 수 없으며, 마찬가지로 인프라가 갖춰진 pervasive 환경에서의 중앙 집중식의 시스템이나 특정 기능의 컴포넌트에 종속되는 것 또한 적절하지 않다. Ad hoc pervasive 환경에서 주변의 여러 자원의 상태를 효율적으로 모니터링하고 이를 여러 다른 컴포넌트들이 다양하게 활용할 수 있어야 하기 때문이다. 마지막으로 MANET 환경에서의 자원 모니터링 시스템들은 우리의 ad hoc pervasive 환경에서의 자원 모니터링이 가야 할 방향과 가장 유사한 모습을 보이고 있다. 하지만 이들 정보 수집에 있어서 MANET에 최적화 되어 있어서 pervasive 환경에서 가장 중요한 특징인 상황 적응(adaptation)을 위한 적절한 민감성을 제공하지 못하고 있다. 물론 정보 수집 주기를 최대한 짧게 함으로써 상황에 최대한 빨리 적응할 수 있도록 할 수 있지만 이는 낮은 네트워크 사용량 같은 자신들의 목적을 상실할 수 있다. 따라서 위의 여러 환경에서의 자원 모니터링 시스템의 특징과 ad hoc pervasive 환경의 특징을 고려한 자원 모니터링 시스템이 설계되어야 하며 가장 중요한 것은 상황적응의 민감성과 네트워크 사용량의 조절이 되어 할 것이다. 그리고 우리는 이를 위한 시스템을 제안한다.

3. 자원 정보 수집

여러 자원 모니터링 시스템을 종합해 볼 때 정보 수집 방법에 있어서 고려되어야 할 사항은 크게 두 가지이다. 첫 번째는 '누구의 관점에서 정보 수집이 이루어 지는가'이고 두 번째는 '정보 수집 방법 자체'이다.

우선 첫 번째 고려사항을 살펴보면 기존의 시스템들은 대부분 자원의 소비자 입장에서 정보 수집이 이루어 지고 있다. 이는 자원의 소비자가 필요한 자원 정보와 자신의 상황에 맞는 주기(민감성)로 정보를 수집함을 의미한다. 혹은 전혀 정보의 제공자와 소비자는 상관없이 주기적으로 정보가 광고되기도 한다. 하지만 ad hoc pervasive 환경을 고려해 볼 때 자원 정보의 주체는 정보제공자 자신이고, 또한 자원 정보의 상태의 유동성도 자원 정보 제공자만이 알 수 있는 내용이기 때문에 자원 제공자의 관점에서 자원 정보 제공이 이루어 져야 한다. 기존의 방법으로 무조건 자원 정보 소비자 입장에서 일방적으로 일정 주기로 정보를 요구하거나 두 입장에 전혀 상관없이 정보를 주기적으로 광고하는 것은 특정 수준의 민감성에 맞추기 위해서 너무 많은 네트워크 사용량의 증가가 이루어 진다. 뿐만 아니라 적절한 수준의 주기를 찾았다 하더라도 각각의 정보 제공자의 상황이 무시 되어 있기 때문에 특정 제공자에게는 적절한 주기가 되지만 다소 정적인 제공자에게는 필요이상으로 데이터를 보내야

하고 너무 동적인 제공자에게는 충분한 민감성을 기대하기 어렵다.

두 번째 고려 사항은 정보 수집 방식인데, 이는 크게 Pull 방식과 Push 방식으로 나누어진다. Pull 방식은 정보의 소비자가 제공자에게 정보를 요구해서 자원 정보를 얻는 방식이고, Push 방식은 정보 제공자가 스스로 정보를 광고하는 방식이다. Ad hoc pervasive 환경을 고려했을 때, pull 방식은 적절한 민감성을 위해서 주기적인 정보제공 요청 메시지를 계속해서 보내고 이를 받은 정보 제공자는 다시 정보를 보내게 되어 단순 광고에 비해서 추가적인 네트워크 사용을 해야 한다. 뿐만 아니라 이 방식은 정보 소비자의 요구에 의해 요청을 보내게 되므로 앞서 말한 정보 제공자의 상황은 고려할 수 없게 된다. 하지만 자원 발견의 기능에 있어서는 발견을 위한 순간에는 가장 최신의 정보를 요청에 의해 얻을 수 있기 때문에 적절할 수 있다. 하지만 상황 적응의 입장에서는 적절한 선택이 아니다. 따라서 결국 남은 Push 방식이 보다 다양한 곳에 정보를 제공해 줄 수 있게 되는데, 이 방식도 대부분 정보 제공자의 상황에는 관계없이 모든 정보제공자가 모두 동일한 주기로 정보를 광고하도록 되어 있다.

따라서 우리는 정보 제공자 상황을 고려한 Push방식(광고)으로 적절한 민감성을 제공하면서도 무리한 네트워크 사용이 없도록 자원 모니터링 시스템을 구축하도록 할 것이다.

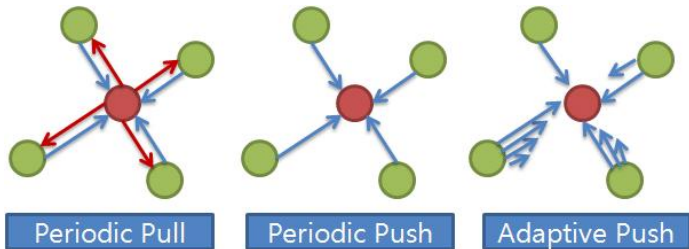


그림 1. 자원 정보 수집 방법

4. 상황에 적응적인 자원 정보 광고 기법

위에서 살펴본 것처럼 기존의 자원 모니터링 시스템들은 정보제공자의 상황을 고려하지 않고 고정된 주기로 정보를 수집하고 있다. 하지만 ad hoc pervasive 환경에서 네트워크 사용량을 크게 늘리지 않는 범위 내에서 최대한 상황 적응 민감성을 최대한 늘리기 위해서는 정보 제공자의 상황을 고려하여 상황에 맞는 주기로 자원 정보를 광고해야 한다.

자신의 상황을 파악하고 자원 정보의 상태를 상황에 맞게 광고하기 위해서는 우선 해당 자원의 특징에 대해 보다 정확히 알아야 한다. 어떤 상황에 광고 주기를 늘리고 줄여야 할지를 명확히 알아야 되기 때문이다. 이를 바탕으로 생각해 봤을 때 우리의 환경에서 생각해

볼 수 있는 자원은 네트워크 관련 자원과 장치 내부의 자원으로 크게 나눌 수 있다.

4.1. 네트워크 관련 자원 정보

네트워크 관련 자원은 대역폭이나 토폴로지 혹은 단말간 연결성 등을 들 수 있으며, 이들은 단말 내부의 특정 자원 정보가 아닌 단말의 존재 자체가 의미가 부여된다. 예를 들어 연결성에 있어서 살펴보면 새로운 단말이 나타나면 이를 알리기 위해서 메시지들이 많이 전달되어야 하며, 새로운 단말이 더욱 많이 등장하거나 사라지는 경우에는 그에 따라 더욱 신속하고 많은 광고 메시지들이 전달되어야 한다. 즉 새로 수집되는 정보에 의지해서 정보 전달 주기를 조절해 네트워크 사용량도 조절하면서 원하는 상황 적응 민감성을 유지할 수 있도록 할 수 있다.

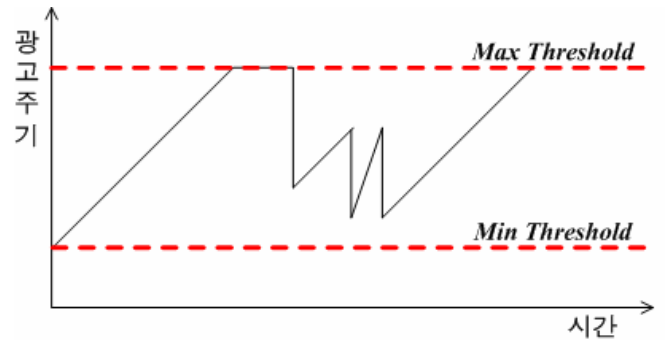


그림 2. 네트워크와 연관된 자원정보를 위한 광고 주기

주변 단말의 연결성의 예를 좀 더 자세히 살펴보자. 주변 단말들의 연결성을 파악하기 위한 정보 광고는 우선 기본적으로 초기 광고주기는 주어진 광고 주기중 가장 짧은 주기를 초기 값으로 갖는다. 그리고 일정 시간이 흐르도록 외부로부터 정보가 들어오지 않으면 한 단위의 주기를 늘린다. 주변에서 정보가 들어오지 않는다는 의미는 그만큼 주변의 상태가 정적인 상태임을 의미하기 때문이다. 시간에 따라 계속적으로 광고 주기는 한 주기기 씩 증가하게 된다. 하지만 어느 순간 정해진 최장의 주기에 이르면 해당 주기를 유지한다. 이 상태에서 주변에서 새로운 단말이 등장한다. 해당 단말은 특정 주기의 광고를 할 것이고 이 광고를 받은 단말은 자신이 가지고 있던 주변 정보와 비교해서 새로운 정보이면 광고 주기를 반으로 줄인다. 그리고 계속해서 해당 단말이 주기적으로 같은 광고를 보내면 받았을 때 이전 정보와 동일한 정보임을 파악하고 시간에 관계없이 광고주기를 한 단위 증가시킨다. 따라서 일정 시간이 지나기 전에 여러 번의 같은 광고를 반복적으로 받으면 광고 주기가 여러 단위까지 늘어날 수 있다. 이는 네트워크 상에 필요 없는 광고 메시지를 조기에 제거하기 위한 목적이다.

이러한 일련의 과정은 위의 그래프를 통해서 알 수 있다.

4.2. 단말 내부 자원 정보

단말 내부의 자원 정보가 외부에 알려져야 하는 경우는 크게 두 가지로 나누어 생각해 볼 수 있다. 내부의 자원 상태 자체가 변했을 때와 주변 단말들의 변화되었을 때, 내부의 자원 정보는 외부로 광고 되어야 한다. 우선 내부의 자원 상태의 변화에 의한 자원 정보 광고는 주기적으로 내부의 자원 정보를 모니터링하다가 변화가 생기면 광고를 하면 된다. 이때 모니터링의 주기와 광고를 내보낼지 말지에 대한 결정은 자원의 특징에 따라 달라 질 수 있다.

두 번째로 내부 자원 상태 정보가 광고되는 경우는 주변의 단말들이 바뀌었을 경우이다. 이 경우는 4.1에서 살펴본 네트워크 정보 광고에 의해서 확인된다. 따라서 네트워크 자원 정보 광고 주기가 짧아 지면 주변 단말들이 새로 등장하는 경우이다. 따라서 이때 내부 자원 상태 정보를 광고하면 된다. 즉 네트워크 관련 자원 정보 광고 주기에 맞춰서 내부 자원 상태 정보도 같이 광고를 하면 주변의 단말들의 상황에 맞게 자신의 내부 자원 정보를 알릴 수 있게 된다.

대표적인 예로 해당 단말에서 수행되고 있는 프로세스의 수를 예로 들어보자. 내부적으로 최적의 모니터링 주기로 프로세스의 수를 확인하고 있다가 프로세스의 수가 갑자기 증가했다. 자원의 정책에 의해 바로 이 자원 상태 정보를 광고한다. 하지만, 일회성으로 광고가 끝나는 것이 아니라, 4.1에서처럼 네트워크 관련 자원 정보의 광고주기에 따라 주기적으로 계속 해당 자원 정보를 광고하게 된다. 즉 새로운 단말이 주변에 발견되면, 네트워크 정보 광고 주기가 짧아지고, 이와 함께 내부 자원 정보도 짧은 주기로 광고 되는 것이다.

5. 자원 모니터링 모듈 아키텍처

이처럼 각자의 자원 정보 제공자가 자신의 상황에 맞춰서 알맞은 주기로 주변에 광고를 할 수 있도록 [9]에 나오는 자원 모니터링 시스템을 수정해 보았다. [9]에서는 질의를 기반으로 한 Pull 방식의 모니터링 시스템이지만 앞서 언급한 것처럼 ad hoc pervasive 환경에서는 Push 방식이 적절하다. 따라서 해당 시스템에서 Pull 방식은 Push 방식으로 변경되고, 또한 추가적으로 우리의 상황에 적응적인 광고를 위해서 Advertisement Interval Calculator 라는 서브 모듈이 추가 되었다. 이 모듈은 4.1에서의 알고리즘에서와 같이 aggregator에서 받은 광고를 바탕으로 광고 주기를 계산해 낸다. 그리고 이 주기를 네트워크 자원 센서 모듈과 다른 모든 내부 자원 센서 모듈들이 같이

사용한다. 그리고 Sensor 서브 모듈들과 Aggregator 서브 모듈들에 의해 모아진 정보는 interaction을 통해서 누구나 얻을 수 있다. 대표적인 예로 내부 특정 모듈이 특정 자원 정보를 원한다면 내부의 interaction module들을 통해서 Subscribe 함으로써 정보를 Publish 받을 수 있을 것이다.

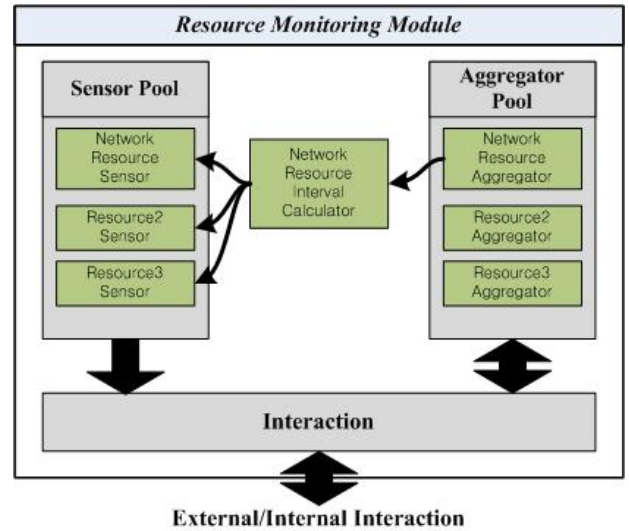


그림 3. 자원 모니터링 모듈 아키텍처

6. 결론 및 향후 연구

지금까지 살펴 본 것처럼 기존의 자원 모니터링 시스템을 살펴보면 유선의 인프라에서는 중앙 집중적이고 인프라가 제공하는 서비스를 이용하는 형식을 가지고 있었다. 이 환경에 pervasive computing이 추가된 환경에서도 마찬가지로 중앙 집중식의 정보 수집이 이루어 지고, 게다가 다른 기능의 서브 기능으로 특정 기능에 종속적인 성격을 띄고 있었다. MANET 환경에서 자원 모니터링 시스템이 우리의 목표 환경인 ad hoc pervasive 환경에서 필요한 자원 모니터링 시스템에 가장 적절한 모습을 제공하는 시스템들이 있었다. 하지만 ad hoc pervasive 환경의 특징인 보다 동적인 환경과 이에 따른 민감한 상황 적응이라는 측면에서 특정 기능들의 수정과 추가가 필요했다. 그 중에서 정보 수집 방법을 살펴보게 되었고 기존의 방식들은 자원 정보 소비자 중심의 Pull 방식이거나 정보 소비자/제공자에 관계없이 일정 주기의 광고로 정보 수집이 이루어 지고 있었다. 하지만 자원 정보의 소비자 중심의 정보 수집은 정보 제공자 당사자의 상황을 알 수 없으므로 의미 없는 메시지를 늘리거나 원하는 수준의 상황 적응 민감성을 얻을 수 없게 된다.

이에 따라 본 논문은 각 정보 제공자가 자신의 상황에 맞추어 자신의 자원 정보 광고 주기를 조절함으로써 네트워크에 필요 없는 메시지 수를

줄이고 모두가 원하는 적절한 수준의 상황 적응 민감성을 제공해 줄 수 있다.

추가적으로 이 연구를 통해서 정보 소비자에 의한 pull 방식의 정보 수집 방법과 정보 제공자에 의한 주기적인 Push(광고)에 의한 정보 수집 방법에 대한 네트워크 사용량과 민감성의 관계를 살펴 보아야 할 것이다. 뿐만 아니라 우리가 제시한 방법의 적절성을 살펴보기 위해 정보 제공자 측면의 상황에 적응적인 광고 방식에서의 네트워크 사용량과 민감성에 대해서도 simulation을 통한 검증이 필요하다.

7. 참고 문헌

- [1] M. Roman, C. K. Hess, R. Cerqueira, R. H. Campbell, and K. Narhstedt, "Gaia: A Middleware Infrastructure to Enable Active Spaces," *IEEE Pervasive Computing Magazine*, vol. 1, pp. 74-83, 2002
- [2] S. Chetan, J. Al-Muhtadi, R. Campbell, M.D. Mickunas, "Mobile Gaia: a middleware for ad-hoc pervasive computing," *Consumer Communications and Networking Conference, 2005. CCNC. 2005 Second IEEE*, pp. 223-228, Jan. 2005.
- [3] D. Garlan, D.P. Siewiorek, A. Smailagic, P. Steenkiste, "Project Aura: Toward Distraction-Free Pervasive Computing," *IEEE Pervasive Comp.*, vol.1, Issue.2, pp. 22-31, 2002.
- [4] Marko Jurmu, "Resource management in smart spaces using context-based leases," M.Sc. thesis, Department of Electrical and Information Engineering, University of Oulu, Finland. 2007
- [5] P. Dinda, T. Gross, R. Karrer, B. Lowekamp, N. Miller, P. Steenkiste, and D. Sutherland, "The Architecture of the Remos System," in *Proc. 10th IEEE Symp. on High-Perf. Dist. Comp. (HPDC'01)*, (San Francisco), pp. 252-65, Aug. 2001.
- [6] B. Noble, M. Satyanarayanan, D. Narayanan, J. Tilton, J. Flinn, and K. Walker, "Agile application-aware adaptation for mobility," *Proc. of 16th Symposium on Operating Systems Principles*, SaintMalo, pp. 276-287, France, Dec. 1997.
- [7] K Kang, J Song, J Kim, H Park, WD Cho, "USS Monitor: A Monitoring System for Collaborative Ubiquitous Computing Environment," *Consumer Electronics, IEEE Transactions on*, Vol. 53, Issue. 3, pp. 911-916, Aug. 2007.
- [8] H Caituiro-Monge, K Almeroth, M del Mar Alvarez, "Friend relay: a resource sharing framework for mobile wireless devices," *Proceedings of the 4th international workshop on Wireless mobile applications and*

services on WLAN hotspots, pp. 20-29, 2006.

[9] C. Tudece, T. Gross, "Resource monitoring issues in ad hoc networks," *Wireless Ad-Hoc Networks, 2004 International Workshop on*, pp. 259-264, 2004.