

SLP를 이용한 SDR 소프트웨어 검색 방안

장기현^o 이병호

한양대학교 정보통신대학원 차세대네트워크 연구실
{nuky99^o, bhrhee}@scann.hanyang.ac.kr

A Proposal for Searching SDR Software by SLP

Keehyun Jang^o Byungho Rhee

Soft Computing And Next generation Network Lab.
Division of Information and Communications, Hanyang University

요 약

SDR은 하드웨어에 의존적이던 신호처리기술을 소프트웨어로 구현함으로써 단말기를 교체하지 않고 소프트웨어의 변경만으로 다양한 통신 서비스를 이용할 수 있게 한다. SDR 기술이 본격적으로 쓰이게 되면 사용자들이 필요한 소프트웨어를 검색하고 다운로드 받는 네트워크 환경이 구축되어야 할 것이다. 소프트웨어를 하나의 서비스로서 검색하기 위해 Service Location Protocol(SLP)을 이용하는 것이 가능하다. 본 논문에서는 SDR 소프트웨어 검색을 위해 SLP 확장 헤더를 이용하는 방안을 제안한다.

1. 서론

SDR(Software Defined Radio)은 이동통신기술 간의 호환성 문제를 해결하여 단말기의 교체 없이 서로 다른 이동통신기술을 사용할 수 있게 한다. SDR 기술은 그 동안 하드웨어에 의존하던 신호처리를 소프트웨어가 맡게 함으로써 디지털 무선통신 기술이 디지털 무선에서 소프트웨어 무선으로 전환할 수 있는 기반을 마련하고 있다. 이동통신규격을 하드웨어가 아니라 소프트웨어로 구현하게 되면 소프트웨어의 변경만으로 서로 상이한 통신 서비스를 제공하는 지역을 하드웨어의 교체 없이 자유로이 오갈 수 있게 된다. 뿐만 아니라, 새로운 이동통신기술이 개발되었을 때 쉽게 적용시킬 수도 있게 된다.[1, 2]

현재의 이동통신 서비스는 각 지역마다 나라마다 규격화되고 고정된 단일 통신 시스템을 갖추고 있다. 지금의 이동통신 서비스라는 것은 이렇게 이미 설정되어 있어서 바꿀 수 없는 단말기를 사용하게 되어있고 그런 단말기와 기지국간의 상호 동작을 통해서 음성통신 및 데이터 통신을 제공하는 것을 말한다. 그러나, SDR 기술이 상용화되면 사용자들은 단순히 소프트웨어를 다운로드 받는 것만으로도 하나의 이동통신기술에 얽매이지 않고 다양한 형태의 통신기술을 사용할 수 있게 될 것이다. 이렇게 되면 이동통신 서비스라는 개념이 서비스 스펙에 맞는 단말기를 구입한다는 것에서 서비스를 구현하는 소프트웨어를 구입한다는 것으로 바뀌게 될 것이다. 즉, 소프트웨어는 하나의 서비스로 인식될 것이다.

소프트웨어를 하나의 서비스로 간주할 수 있다면 사용자가 적합한 소프트웨어를 찾는 과정에 Service Discovery Protocol(SDP)이 이용될 수 있다. SDP는 기존의 PC 나 핸드폰, PDA와 같은 기기들이 미리 사용될 수 있도록 구성된 설정 없이도 자유로이 프린터나 정보가전기 등 사용 가능하게 한다. 즉, 어떠한 기기 제공하는 서비스를 네트워크상에서 찾고 사용하는 시점에 이용 가능하도록 시스템을 설정하여 사용자가 기기 사용을 위해 이는 노력을 최소화 시킨다.[3]

SDR 시스템에 이용되어질 소프트웨어는 이동통신규격을 구현

한 것이 될 것이므로 SDP를 필요한 소프트웨어를 찾고 설치하는 과정에 이용할 수 있다. 본 논문에서는 앞서 기술했듯이 소프트웨어를 하나의 서비스로 간주하고 필요한 소프트웨어를 네트워크 상에서 검색하기 위해서 SDP의 한 종류인 Service Location Protocol(SLP)을 이용하는 방안에 대하여 논한다.

2. SLP Overview

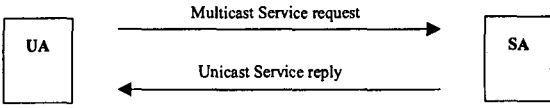
SLP는 인터넷 사용자들이 네트워크상에서 공유할 수 있는 장치들의 설정을 단순화시키기 위한 목적으로 IETF(Internet Engineering Task Force)의 SVRLOC(Service Location Protocol) working group 이 발표한 프로토콜이다. 무선 네트워크의 출현과 모바일 기기 사용의 확산으로 자동설정에 대한 중요성이 점차 증대되면서 SLP에 대한 관심도 많아졌다. SLP는 기존의 버전 1에서 문제가 되었던 점들을 개선한 버전 2가 발표되었고 현재는 IPv4를 대체하는 IPv6에 대한 확장 SLP가 논의 중에 있다.[4, 5, 6]

SLP는 네트워크상에서 서비스를 찾기 위해서 세 가지 agent가 유기적으로 동작하는 구조를 가지고 있다.

- **User Agent(UA)** 사용자의 기기상에서 동작하면서 필요한 서비스 객체와 데이터를 주고 받는다. UA는 서비스 정보를 얻기 위해서 Service Agent(SA)나 Directory Agent(DA)와 통신한다.
- **Service Agent(SA)** 하나 이상의 서비스들의 정보를 관리하면서 UA의 요청에 응답하거나 DA에 서비스를 등록한다.
- **Directory Agent(DA)** SA의 service advertisement를 수집하여 관리하고 UA의 요청에 응답한다. DA는 네트워크상의 여러 SA들에 대한 cache server의 기능을 한다.

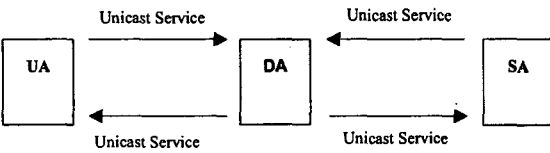
소규모 네트워크에서는 UA가 직접 SA를 찾아서 서비스를 요청한다. UA는 필요한 서비스의 정보를 가지고 있는 SA를 찾아내기 위해 응답을 받을 때까지 반복적으로 multicast 메시지를 보내

고 SA는 unicast 메시지로 UA의 요청에 응답한다.<그림1>



<그림 1> Direct Service Request

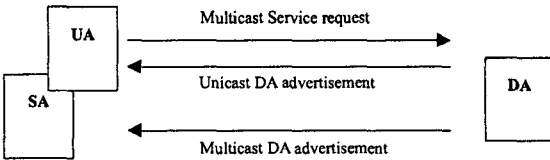
대규모 네트워크에서는 DA가 서비스 검색과정에 참여해서 반복적인 UA의 multicast 메시지로 인한 네트워크 부하를 줄이고 UA의 서비스 요청에 더욱 빠른 응답시간을 가지게 한다.



<그림 2> DA를 통한 Service Request

SA는 DA에 unicast 서비스 등록 메시지를 보내고 등록이 완료되면 DA로부터 unicast 서비스 완료 메시지를 받는다.<그림 2>

대규모 네트워크에서 UA와 SA가 DA에 서비스를 요청하거나 등록하기 위해서는 DA의 위치를 미리 알고 있어야 한다. DA discovery는 active DA discovery와 passive DA discovery로 구분된다.



<그림 3> DA Discovery

Active DA discovery는 UA와 SA가 multicast 메시지를 네트워크 상에 전파하고 이를 수신한 DA가 unicast 메시지를 보내는 방식이며, Passive DA discovery는 DA가 네트워크 상에 multicast 메시지를 전파함으로써 UA와 SA가 DA의 위치를 알 수 있게 하는 방식이다.[4]

3. SLP Message Header와 Extension Header

모든 SLP 메시지는 <그림 4>와 같은 메시지 헤더를 가지고 있다. 메시지 헤더의 Function ID 값에 따라 다음에 오는 데이터의 쓰임이 결정된다.

Version	Function ID			Length
Length, contd.	O	F	R	reserved
Next Ext Offset, contd.	Next Ext Offset			
Language Tag Length	XID			
Language Tag		Language Tag		
Data				

<그림 4> SLP Message Header

- **Version** 현재 사용되고 있는 SLP의 버전, 1이나 2가 된다.
- **Function ID** 메시지 헤더 다음에 올 데이터가 어떤 목적을 위한 것인지 나타낸다.
- **Length** 헤더를 포함한 SLP 메시지의 전체 길이를 나타낸다.
- **Flags Overflow(0x80)**는 메시지가 너무 커서 하나의 데이터 그램으로 보내질 수 없을 때 설정된다. **Fresh(0x40)**는 모든 새로운 service registration 메시지에 설정된다. **Request(0x20)**는 multicast나 broadcast가 필요할 때 설정된다.
- **Next Ext Offset** 메시지가 가지고 있는 데이터 외에 더 많은 정보를 필요로 할 때 더해지는 데이터의 시작 위치를 나타낸다.
- **XID** 요청 메시지들의 중복을 피하고 유일하게 구분하기 위해 설정한다.
- **Language Tag Length** Language Tag 필드의 길이를 나타낸다.
- **Language Tag** 데이터가 해석되는 언어를 설정한다.

SLP는 Function ID로 구분되는 11개의 메시지를 가지고 있고 상황에 따라 메시지를 확장할 수 있다. SLP Extension을 사용하기 위해서는 Next Ext Offset의 값을 SLP Extension 헤더의 시작 위치로 설정하면 된다. SLP Extension이 사용되지 않으면 이 값은 0이 된다. <그림 5>는 SLP Extension 헤더의 구조를 보여준다.

Extension ID	Next Ext Offset
Next Ext Offset, contd.	Extension Data

<그림 5> SLP Extension Header

SLP Extension은 Extension ID에 따라 세 가지로 구분된다.

- **0x0000-0x3FFF** 선택적으로 구현 가능하며 메시지 헤더 분석 시 식별이 안되면 무시한다.
- **0x4000-0x7FFF**의 목적으로 구현해야 하며 UA나 SA가 응답 메시지에 이 확장 메시지를 수신 후 식별하지 못하면 응답을 하지 않는다.
- **0x8000-0x8FFF** 비 표준 형식의 확장 메시지에 사용한다.

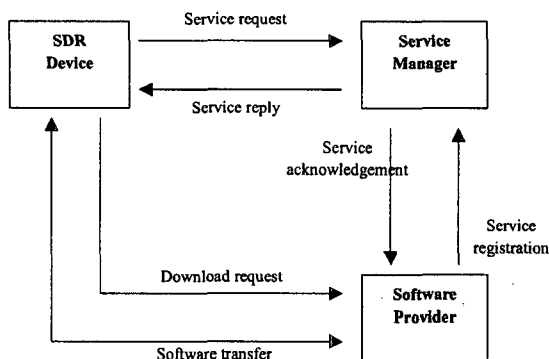
0x9000-0xFFFF의 값은 Extension ID로 쓰이지 않는다. Next Ext Offset은 여러 개의 확장 메시지가 존재할 경우 다음 메시지의 시작 위치를 나타낸다.[7]

4. SLP를 이용한 SDR Software Discovery Process

SDR Software Download Process에 SLP Service Discovery 모델을 적용해보면 <그림 6>과 같은 형태가 될 것이다. Service Manager(SM)와 Software Provider(SP)는 각각 DA와 SA의 역할을, 그리고 SDR Device(SD)는 UA의 역할을 하게 된다. Discovery Process는 소프트웨어의 정보를 등록하는 Registration, 필요한 소프트웨어를 검색하는 Lookup, 검색된 소프트웨어를 다운로드 하는 Transfer의 세 가지 단계로 구성된다.

- **Registration SM**은 SD가 소프트웨어를 찾기 위한 Service request 메시지를 보내기 전에 소프트웨어를 보유한 SP의 정보를 가지고 있어야 한다. SP는 Service registration 메시지를 통해서 자신의 위치와 소유한 소프트웨어의 종류와 속성을 SM에 등록한다.

- **Lookup SD**는 시스템의 현재 설정과 구성 정보를 Service request 메시지를 통해 SM에게 전송해야 한다. SM은 SD의 시스템 구성 정보를 토대로 SD가 다운로드 받아서 실행할 소프트웨어가 SD의 현재 시스템 구성에서 정상적으로 동작할 지를 판단하여 실제 소프트웨어를 다운로드 할 SP의 정보를 Service reply 메시지를 통해 넘겨준다.
- **Transfer SD**는 SM을 통해서 필요한 소프트웨어를 보유하고 있는 SP의 위치를 알아낸 후 대상 SP에 download request 메시지를 보낸다. 요청이 승낙되고 나면 소프트웨어 전송이 시작된다.



<그림 6> SDR Software Discovery Process

5. SDR을 위한 SLP Extension Message의 활용

기존의 SLP 메시지들은 SDR Software Download Process에서 요구하는 사항을 만족시킬 수 없다. 예를 들어, SLP Service request 메시지에는 SD의 시스템 구성 정보를 담아야 하는데 현재 SLP 메시지 구조에서는 불가능하다. 이런 문제를 해결하기 위해서 SLP Extension을 이용해야 한다.

SLP Extension을 이용하기 위해서는 Extension ID를 선택해야 한다. Extension ID는 앞서 기술한 대로 세 가지 범위에서 선택이 가능하다. <표 1>에서와 같이 기존의 SLP Extension을 위해 설정된 값들을 제외하고 이미 구현된 SLP 기반의 Application들과의 오동작을 피하기 위하여 0x0007에서 0x3FFF 사이의 값으로 Extension ID를 설정한다.

Extension ID	Description
0x0001	Attribute missing
0x0002	Attribute list
0x0003	Vendor opaque
0x0004	Subscribe
0x0005	NotifyAt
0x0006	Mesh-enhancement
0x4001	SLP Local Service
0x4002	Selection
0x4003	Sort

<표 1> Extension IDs

SLP Extension을 사용할 SLP 메시지들에는 Service request와 Service registration이 있다. <그림 7>은 SDR을 위한 SLP Service request 메시지를 보여준다.

0x0002	0x0001	0x0057
0x0057	0 0 0	0x0000
0x0025		0x0025
0x0010		cn-GB-ocd
Data		
0x0010		0x0000
0x0000		Extension Data

<그림 7> Extended SLP Service request message

Service request의 Function ID는 1(0x0001)이고, Extension ID는 0x0007과 0x3FFF 사이의 값인 16(0x0010)으로 설정하였다. Service registration도 동일한 적용이 가능하다. Service registration의 Function ID는 3(0x0003)이고, Extension ID는 17(0x0011)로 설정할 수 있다.

6. 결론

디지털 무선통신 기술이 디지털 무선에서 소프트웨어 무선으로 바뀌어 가면서 신호처리 프로세스를 소프트웨어로 구현하는 방안이 연구 중이다. 소프트웨어 무선의 특징은 신호처리에 대한 하드웨어 의존도를 줄여서 단일 단말기에서 다중의 통신시스템을 이용할 수 있게 한다. 소프트웨어 다운로드를 통한 시스템의 재구성은 이러한 특징을 구현 가능하게 할 수 있다. 본 논문에서는 소프트웨어를 하나의 서비스로 간주하고 소프트웨어 검색 과정을 SLP Service Discovery 모델에 적용하는 방안에 대하여 기술하였다. SLP Extension을 이용하면 기존의 SLP 메시지가 담을 수 없었던 부가 정보를 전송할 수 있다는 것을 보이고, SLP Extension을 사용하기 위한 Extension ID에 대하여 논하였다. 향후 실제 구현을 통하여 발생하는 문제점을 파악하고 개선 방안에 대한 연구가 필요하다.

참고문헌

- [1] 김지연, 김진업, "SDR 기술의 현재와 발전방향", 한국통신학회지, Vol. 19, No. 11, pp.14-23, 2002. 11.
- [2] 조권도, 김지연, 이규대, 이승환, 신민정, 김진업, "SDR 기술개발 동향", 전자통신동향분석 Vol. 19, No. 3, pp.63-72, 2004. 6.
- [3] Sumi Helal, "Standards for Service Discovery and Delivery", IEEE, 2002
- [4] E. Guttman, "Service Location Protocol : Automatic Discovery of IP Network Services", IEEE, 1999
- [5] E. Guttman and J. Veizades, "Service Location Protocol Modifications for IPv6", May 2001, RFC Proposed.
- [6] C. Perkins, "Service Location Protocol for Mobile Users", IEEE, 1998
- [7] E. Guttman, C. Perkins, J. Veizades, and M. Day, "Service Location Protocol, Version 2", RFC 2608, June 1999

* This work was supported by the HY-SDR research center at Hanyang University, Seoul, Korea, under the ITRC program for MIC, Korea.