

안드로이드 플랫폼상의 홈네트워크 미들웨어 연구

신근태*, 조은선*, 박성준**, 박상훈**

*충남대학교 전기전자통신 컴퓨터 공학부

**삼성 소프트웨어 멤버십

e-mail:shinkt_m@nate.com, eschough@cnu.ac.kr,

lovers0322@naver.com, hailuhanr@nate.com

A Research of Home Network Middleware on Android Platform

Keun-Tae Shin*, Eun-Sun Cho*, Sang-Hun Park**, Sung-Jun Park**

*Dept. of Computer Engineering, Chungnam National University

**Samsung Software Membership

요 약

2007년 구글에서 공개된 안드로이드는 개방형 모바일 플랫폼으로 모바일 장치를 목표로 한다. 안드로이드 플랫폼은 다양한 모바일 제품에 적용 할 수 있는 구조로 되어 있어 확장이 용이하며 향후 다양한 모바일 장치에 사용될 것으로 생각된다. 이러한 발전성을 고려하여 선행연구 차원에서 연구 될 가치가 충분하다고 판단이 된다. 본 논문에서는 안드로이드 플랫폼에 대표적 홈네트워크 미들웨어인 UPnP와 DLNA에서 명시하는 UPnP for A/V에 준하는 스택을 구현하여 UPnP를 통한 장치들을 검색 및 제어를 하고 DLNA에 준하여 디지털 콘텐츠를 공유, 전송 한다. 또한 안드로이드 플랫폼과 디지털 액자간의 네트워킹을 통해 홈네트워크 플랫폼으로서의 안드로이드 가능성을 확인한다.

1. 서 론

구글에서 공개된 안드로이드는 개방형 모바일 플랫폼으로 핸드폰을 타겟으로 한다. 안드로이드에 구성된 플랫폼 구조는 핸드폰 뿐만이 아니라 TV, SETTOP BOX D, DVD Player 등의 다양한 장치에 사용 되어질 것이라 판단이 된다.

다양한 장치간 네트워킹을 통해 콘텐츠에 대한 공유, 전송, 저장을 위해 홈네트워크 미들웨어 구현의 필요성이 요구 된다.

본 논문에서는 안드로이드 플랫폼에 대표적인 홈네트워크 미들웨어인 UPnP 를 모바일 장치에 적은 부하를 가지고도 홈네트워크 통신이 가능하도록 구현하는 방법에 대해 기술 한다.

2. 기술 경향

UPnP 홈네트워킹에 참여 하기 위해 안드로이드 플랫폼에 구현 되는 기술에 대한 간단한 설명을 한다.

· Universal Plug and Play (UPnP)

PC, 주변장치, 지능형 가전제품, 무선 장비 등과 같은 장치들을 네트워크에 접속시켰을 때, 인터넷과 웹 프로토콜을 사용하여 서로를 자동으로 인식할 수 있도록 해주는 표준이며, DLNA의 기반이 되는 홈네트워크 미들웨어 및 프로토콜이다.

· Digital Living Network Alliance (DLNA)

디지털 홈을 위한 다양한 PC 및 가전 장치들 간에 디지털 콘텐츠를 공유하기 위한 설계지침으로 UPnP 에 기반이다.

· HTTP1.1

이 규약은 기능을 신뢰할 수 있도록 구현하기 위해 HTTP/1.0보다 더 엄격한 필요조건을 포함 한다.

실제적인 정보 시스템은 단순한 조회보다 검색, 프린트-엔드(front-end) 갱신 및 주식 달기 등 많은 기능을 필요로 한다. HTTP는 요구의 목적을 표시하는 일련의 개방된 method를 (open-ended set of methods) 허용한다. 이 규약은 보편적 자원 식별자(URI), 자원 위치 (URL)또는 자원 이름(URN)이 제공하는 참고 방법에 따라 method를 적용할 자원을 지칭하는 데 사용한다. 메시지는 다용도 인터넷 메일 확장(MIME)에서 정의된 것처럼 인터넷 메일에서 사용되는 것과 유사한 형식으로 전송한다.

· Simple Service Discovery Protocol (SSDP)

네트워크 서비스를 네트워크 상에서 검색하는 방법을 정의한다. SSDP는 HTTPU 및 HTTPMU 기반 위에 구축되며, 제어 포인트가 네트워크 상에서 원하는 리소스를 검색하는 방법 및 장치들이 네트워크상에서 자신들이 가용상태에 있음을 알리는 방법을 정의한다.

· Generic Event Notification Architecture (GENA)

TCP/IP를 통한 HTTP 및 멀티캐스트 UDP를 사용하여 통보(notifications)를 송수신하는 기능을 제공하기 위하여 정의. 또한 GENA은 이벤트 실행을 위하여 가입자 및 통보 발행자의 개념을 정의한다.

· Simple Object Access Protocol (SOAP)

XML 및 HTTP의 용법을 정의하여 원격 프로시저 호출을 실행한다.

· Digital Item Declaration (DIDL)

디지털 아이템은 멀티미디어 콘텐츠와 같은 개념으로 디지털 형태로 표현(혹은 코딩, 포매팅)되어 네트워크나 터미널에서 취급 거래, 이용되는 기본적인 콘텐츠 단위이다.

이는 일종의 구조적인 멀티미디어 객체로서 멀티미디어 리소스(정지/동영상, 오디오클립 등)뿐만 아니라, 이 디지털 아이템의 식별자와 기술자와 같은 메타데이터를 포함한다. 디지털 아이템은 디지털 아이템 선언에 의해 선언되고, 이를 식별하는 식별자는 디지털 아이템 식별 이라는 방법으로 기술 한다.

3. UPnP 홈네트워크 미들웨어 설계

안드로이드에 적용하기 위한 UPnP 미들웨어에 구성은 아래와 같은 재구성을 통해 적은 부하를 갖도록 설계 하였다.

	DLNA Guidelines	비고 (Android적용)
Media Formats	JPEG,LPCM,MPEG2	JPEG, PNG
Device Discovery, Control and Media Management	UPnP AV 1.0, UPnP Device Architecture 1.0	프레젠테이션, 단계 축소, SOAP 함수 최적화, DIDL 최적화
Media Transport	HTTP 1.0/1.1	HTTP 1.0/1.1
Network Stack	IPv4 Protocol Suite	IPv4 Protocol Suite
Network Connectivity	Wireless: 802.11a/b/g	Wireless: 802.11a/b/g

<표 1> DLNA 의 수정된 GuidLines

기존 DLNA에서 제공 하고 있는 가이드 라인에서 우측 비교에 나타나 있는 UPNP 의 프레젠테이션 단계를 축소 하였으며 실제 SOAP을 이용하여 자원을 요구 하는 부분을 디지털 이미지만을 요청 하도록 축소 하였다.

또한 DIDL 는 디지털 이미지의 레벨에 따른 인증 부분과 디지털 목록만을 가지고 있도록 하였다.

실제 구현은 통신부, 데이터 생성 및 처리부, 표현부로 나누어 진다.

3.1 통신부

데이터 교환을 위한 통신부 설계는 다음과 같다.

3.1.1 Multicast Packet

안드로이드 플랫폼이 적용 된 장치의 홈네트워킹 참여 의사 알리거나 홈네트워킹에 참여 하고 있는 장치 검색을 위하여 Multicast 패킷을 사용한다.

Internet Assigned Numbers Authority(IANA) 에 명시 되어 있는 1900 포트 번호를 사용 하며, SSDP 를 사용 하기위해 지정되어 있는 239.255.255.250 주소를 사용 한다.

3.1.2 UDP Packet

특정 장치가 안드로이드 플랫폼이 적용된 장치에 대한 명세를 요구하기 위하여 사용 한다.

이는 3.1.1 의 Multicast Packet 에 의해 안드로이드 플랫폼이 정용된 장치에 홈네트워킹 참여 여부를 확인 한 후 사용 한다.

3.1.2 TCP Packet

특정 장치가 안드로이드 플랫폼이 적용된 장치에 홈네트워킹 참여 의사를 알고, 그 장치에 명세를 획득 하였다면, 디지털 장치가 보유 하고 있는 이미지 자원 혹은 자원의 리스트를 요구하기 위해 사용 한다.

HTTP/1.1 방식으로 요청을 하며, 이에 대한 안드로이드 플랫폼이 적용된 장치에 응답은 SOAP, XML, Byte Coe 등의 형식 이다.

3.2 데이터 생성 및 처리부

3.1에 정의된 통신에 의해 전달되는 정보에 대한 생성. 디지털 콘텐츠 공유는 크게 두가지의 역할이 있다.

- Media Renderer - 디지털 콘텐츠를 보여주게 되는 장치 예) 디지털 액자
- Media Server - 디지털 콘텐츠 소스를 제공하게 되는 장치 예) 모바일 장치, TV, PC

3.2.1 홈네트워킹 참여 의사 데이터 생성

```
M-SEARCH * HTTP/1.1/r/n
HOST: 239.255.255.250:1900/r/n
MAN: "ssdp:discover"/r/n
MX: seconds to delay response/r/n
ST: search target/r/n
/r/n
```

<표 2> Media Renderer 의 요구

Media Renderer 는 홈네트워킹 참여시 자신이 제어 혹은 콘텐츠를 공유 할 수 있는 장치를 찾기 위해 표1 와 같이 데이터를 Multicast 로 전달한다.

```
HTTP/1.1 200 OK/r/n
CACHE-CONTROL: max-age = /r/n
DATE: /r/n
EXT: /r/n
LOCATION: /r/n
SERVER: /r/n
ST: /r/n
USN: /r/n
/r/n
```

<표 3> Media Server 의 응답

표1과 같은 정보를 Multicast 로 전달 받은 Media Server 는 자신이 홈네트워킹에 참여 하고 있으며 자신의 명세를 요구 받기 위해서 Media Renderer 가 요구 할 주소를 전달한다.

3.2.2 장치 정보 요구에 대한 응답 생성

```
GET/upnp/hst/udhisapi.dll?content=uuid:24c0ebef-5670-4ec2-9695-ec0420324c6b HTTP/1.1/r/n
Connection: Keep-Alive/r/n
Host: 192.168.10.5:3000/r/n
/r/n
```

<표 4> Media Renderer 의 요구

Media Server 의 홈네트워킹 참여 의사를 받는 Media Renderer 는 Unicast 로 다음과 같은 표 3와 같은 요구한다.

```
HTTP/1.1 200 OK/r/n
Content-Length: 5353/r/n
Cache-Control: max-age=1800/r/n
Connection: Keep-Alive/r/n
Content-Type: text/xml/r/n
/r/n
<?xml version="1.0"?>
<root xmlns="urn:schemas-upnp-org:device-1-0">
<specVersion>
<major>1</major>
<minor>0</minor>
</specVersion>
<device>
<modelName>Google Phone</modelName>
<----- 생략 ----->
```

<표 5> Media Server 의 응답

Media Server 는 표3와 같은 요구에 표4와 같은 응답한다

다. 이는 Media Server 의 장치 제작자, 장치 이름, 서비를 호출하기 위한 주소로 구성 한다.

3.2.4 인증 정보 생성

```
POST/upnphost/udhisapi.dll?control=uuid:ba7327f9-3573-4ae2-a90b-3434f3b9f278+urn:microsoft.com:serviceId:X_MS_MediaReceiverRegistrar HTTP/1.1/r/n
SOAPACTION:"urn:microsoft.com:service:X_MS_MediaReceiverRegistrar:1#IsAuthorized"/r/n
Content-Type: text/xml ; charset="utf-8"/r/n
Content-Length: 316/r/n
Connection: Keep-Alive/r/n
Expect: 100-continue/r/n
Host: 192.168.10.5:3000/r/n
/r/n
<?xml version="1.0" encoding="utf-8"?>
<s:Envelope s:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/" xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
<s:Body>
<u:IsAuthorized xmlns:u="urn:microsoft.com:service:X_MS_MediaReceiverRegistrar:1">
<----- 생략 ----->
```

<표 6> Media Renderer 의 요구

Media Server 의 장치 정보를 획득한 Media Renderer 는 Media Server 에 대한 제어권을 획득 하기 위해 SOAP 를 사용 하여 표 5와 같은 요구 한다.

```
HTTP/1.1 200 OK
Content-Length: 418
Cache-Control: max-age=1800
Connection: Keep-Alive
Content-Type: text/xml

<?xml version="1.0"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<SOAP-ENV:Body>
<m:IsAuthorizedResponse xmlns:m="urn:microsoft.com:service:X_MS_MediaReceiverRegistrar:1">
<----- 생략 ----->
```

<표 7> Media Server 의 응답

Media Renderer 의 장치 제어 요구를 받은 Media Server 는 내부적으로 Media Renderer 에 대한 정보를 생성하여 장치 제어 수락을 하며 그에 대한 결과를 표 6 와 같은 형식으로 응답한다.

3.2.5 디지털 콘텐츠 리스트 생성

```
POST /upnphost/udhisapi.dll?control=uuid:ba7327f9-3573-4ae2-a90b-3434f3b9f278+urn:upnp-org:serviceId:ContentDirectory HTTP/1.1/r/n
SOAPACTION: "urn:schemas-upnp-org:service:ContentDirectory:1#Browse"/r/n
Content-Type: text/xml ; charset="utf-8"/r/n
Content-Length: 462/r/n
Connection: Keep-Alive/r/n
Host: 192.168.10.5:3000/r/n
/r/n
<----- 생략 ----->
```

<표 8> Media Renderer 의 요구

Media Server 로부터 제어권을 획득한 Media Renderer 는 Media Server 에게 디지털 콘텐츠 목록을 요구한다. Media Renderer 의 콘텐츠 리스트 요구를 받은 Media S

```
HTTP/1.1 200 OK/r/n
Content-Length: 3052/r/n
Cache-Control: max-age=1800/r/n
Connection: Keep-Alive/r/n
Content-Type: text/xml/r/n
/r/n
<?xml version="1.0"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
<SOAP-ENV:Body>
<m:BrowseResponse xmlns:m="urn:schemas-upnp-org:service:ContentDirectory:1">
<Result xmlns:dt="urn:schemas-microsoft-com:datatypes" dt:dt="string">
<DIDL-Lite xmlns:dc="http://purl.org/dc/elements/1.1/" xmlns:upnp="urn:schemas-upnp-org:metadata-1-0/upnp/" xmlns="urn:schemas-upnp-org:metadata-1-0/DIDL-Lite/">
<----- 생략 ----->
```

<표 9> Media Server 의 응답

erver 는 해당 Media Renderer 에게 제공한 제어권을 확인 후 표 10 와 같은 형식의 콘텐츠 리스트를 생성하여 전달. 각각의 콘텐츠는 해당 콘텐츠를 요구 할수 있는 주소 정보를 포함한다.

3.2.6 디지털 콘텐츠 바이트코드 생성

```
GET/images/3.jpg?format=JPEG,width=800,height=800
HTTP/1.1/r/n
Connection: Keep-Alive/r/n
Host: 192.168.10.5:3000/r/n
/r/n
```

<표 10> Media Renderer 의 요구

콘텐츠 리스트를 전달 받은 Media Renderer 는 해당 콘텐츠에 대한 요구 을 표 8 와 같이 요구한다.

```
HTTP/1.1 200 OK/r/n
Content-Length: 153252/r/n
Content-Type: image/jpeg/r/n
Server: Microsoft-HTTPAPI/1.0/r/n
Accept-Ranges: bytes/r/n
TransferMode.DLNA.ORG: Interactive/r/n
Date: Fri, 25 Jul 2008 19:53:10 GMT/r/n
/r/n
<-- 콘텐츠의 바이트 코드 -->
```

<표 11> Media Server 의 응답

콘텐츠 요구를 한 Media Renderer 에게 Media Server 는 표 9와 같은 응답와 함께 해당 콘텐츠의 바이트 코드를 전달한다.

3.3 표현부

3.2의 데이터 생성 및 처리부에서 생성된 데이터 들을 3.1의 통신부를 통해 보내기 위해서는 제어 방법와 그 결과를 표시해주기 위해 표현부 사용한다.

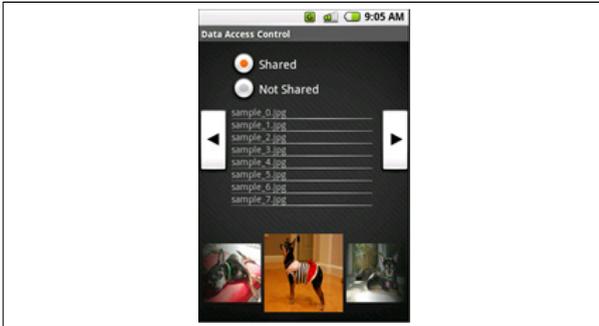
3.3.1 환경 설정



<그림 1> 환경설정

자동적으로 할당 받은 IP 에 대한 설정을 이루며, 원할시 포트 번호를 설정 할수 있고 해당 장치에 대한 간단한 이름을 설정할수 있도록 구성한다. 또한 장치정보 유효시간을 설정한다.

3.3.2 콘텐츠별 공유 설정



<그림 2> 공유 설정

각 콘텐츠마다 공유 혹은 공유하지 않음에 대한 설정을 할 수 있도록 제공.

3.3.2 공유 제어



<그림 3> 공유 제어

공유, 공유 하지 않음에 대한 인터페이스를 제공한다.

4. 구현 환경

개발환경은 Window XP에서 구글에서 제공하고 있는 Android Development Tools 을 사용 Java 로 구현 하였으며, 테스트 환경은 Android Development Tools에서 제공하는 Android Emulator 를 사용하여 삼성 SPH-83V 디지털 액자와 디지털 콘텐츠 교환을 통해 테스트 하였다.

5. 관련 연구

안드로이드 플랫폼과 같은 형태의 플랫폼으로는 인텔사의 Moblin 이 있다.

Moblin 은 오픈 소스 기반의 모바일 리눅스 소프트웨어로 x86 아키텍처에 최적화가 되어 있으며 아직은 대중성을 확보 하고 있지는 않다.

인텔사에서 제공하고 있는 x86 아키텍처 기반으로 제작 되어진 소프트웨어를 Mobile Internet Device(MID) 에 손쉽게 포팅 할 수 있고, 어플리케이션 개발 툴 체인을 제공과 리눅스를 기반으로 하고 있다는 것을 장점으로 꼽을 수 있다.

MID 와 휴대폰의 경계가 무너지고 있는 시점에서 향후 안드로이드, Moblin 와 같은 개발 툴 체인을 공급 하는 것이 시장 확보에 핵심 이 될 것이다.

6. 결론

안드로이드 플랫폼에 대표적 홈네트워크 미들웨어인 UPnP 를 DLNA에서 제안하는 권고안에 최소한의 부하를 갖는 구현을 하였다.

안드로이드 플랫폼에 구현한 미들웨어를 통해 디지털 콘텐츠를 공유 할 수 있는 제품은 Sony 사의 Playstation3 , Microsoft 사의 Xbox 360 , 삼성전자의 보르도 TV 등이 있으며, DLNA 인증을 받은 전자 제품은 대부분 가능 하다.

이동성이 좋은 모바일 장치는 디지털 콘텐츠의 생성이 빈번하게 일어난다. 하지만 소형 이라는 점에 있어서 에너지가 고려된다. 이러한 점을 고려해 보았을 때 타 장치와의 디지털 콘텐츠 공유를 적은 오버헤드를 갖도록 하는 것이 매우 중요하다.

본 연구를 통해 디지털 콘텐츠 공유 서버로서의 안드로이드 플랫폼의 그 가능성을 알아보았다.

[참고 문헌]

- [1] 손지연(Jiyeon Son·孫知延), 박준석(Junseok Park·朴俊錫) 저, "Universal Plug and Play 기술 개요 및 동향", 한국정보기술학회, 한국정보기술학회지 韓國情報技術學會誌 제1권 제1호, 2003. 12, pp. 89 ~ 94 (6pages)
- [2] 임승옥, 정광모, "UPnP(Universal Plug and Play) 기술 분석", 전자정보센터(EIC) 원고, 2003. 3.
- [3] 정성원(Sungwon Jeong), 장영숙(Youngsuk Jang) 저, "UPnP 구조와 테스트 툴에 대한 고찰", 한국정보과학회, 한국정보과학회 학술발표논문집 한국정보과학회 2004년도 봄 학술발표논문집 제31권 제1호(B), 2004. 4, pp. 421 ~ 423 (3pages)
- [4] UPnP Forum, "UPnP Device Architecture 1.0", http://www.upnp-ic.org/resources/UPnP_device_architecture_docs/UPnP-DeviceArchitecture-v1_0-20060720.pdf, 2006
- [5] Yeon-Joo Oh Hoon-Ki Lee Jung-Tae Kim Eu i-Hyun Paik Kwang-Roh Park "The DLNA Proxy System Architecture for Sharing In-Home Media Contents via Internet" Advanced Communication Technology, 2006. ICACT 2006. The 8th International Conference 2006
- [6] 김철승 (Chul Seung Kim), 김구수 (Gu Su Kim), 임영익 (Young Ik Eom) "IEEE1394 기반 AV 콘텐츠 공유를 위한 DLNA DMS 설계 및 구현" 한국정보처리학회 정보처리학회논문지D, Vol.13, No.7 2006