

UPnP 구조와 테스트 툴에 대한 고찰

정성원⁰ 장영숙

삼성전자 기술총괄 소프트웨어센터

sungwon_sw@samsung.com⁰, ys1113.jang@samsung.com

The Study on UPnP Architecture and Test Tool

Sungwon Jeong⁰, Youngsuk Jang

Software Center, Corporate Technology Operations, Samsung Electronics Co., Ltd.

요 약

본 논문은 홈 네트워크 상에서 존재 가능한 PC와 디지털 가전 기기(devices) 및 무선 기기들을 피어-투-피어(peer-to-peer) 방식으로 네트워크의 연결성을 배어둘게(pervasive) 하는 역할을 수행할 수 있는 UPnP(Universal Plug and Play, 이하 UPnP)의 구조를 분석하고 UPnP의 정의된 기능을 테스트하는 툴에 대한 연구로서, UPnP Device Architecture v1.0 표준 스펙(specification)에 따라 구현된 UPnP 미들웨어를 자동으로 테스트할 수 있도록 하는 방법과 UPnP 테스트 툴의 구조를 제안한다.

1. 서 론

최근 네트워크 장비의 고속화를 바탕으로 각종 디지털 가전 기기들이 널리 사용되면서, 이들을 하나의 네트워크로 연결시켜 통제하는 홈 네트워크 환경에 대한 관심이 점차 높아져가고 있다. 홈 네트워크 환경에서는 홈 네트워크에 연결된 여러 가지 디지털 가전 기기들의 상태를 모니터링하고 사용자의 요청에 따라 제어 명령을 전달 할 수 있도록 하는 홈 네트워크 기술이 필요한데, 이러한 기술로는 UPnP, Jini(Java Intelligent Network Infra-structure), Havi(Home Audio Video Interoperability), LonWorks 등의 기술이 제안되고 있다[1].

마이크로소프트사(MS)에서 제안한 UPnP Device Architecture v1.0 스펙(이하 UPnP 스펙)은 IP를 기반으로 제어 장치(Control Point)와 피제어 장치(Controlled Device)로 구분되어 둘 사이의 네트워킹을 정의하고 있다. 이러한 UPnP 스펙에 맞게 구현되어진 UPnP 미들웨어를 테스트할 수 있는 상용 테스트 툴은 현재 출시되고 있지 않으며 다만, UPnP 기구(www.upnp.org)에서 제공하고 있는 UPnP Reference 스펙(Stack)은 UPnP 미들웨어 개발을 위한 참조용 스택으로서 UPnP의 기본적인 기능만을 제한적으로 테스트할 수 있다. 이에 UPnP 미들웨어를 개발하고 있는 많은 기업에서는 UPnP 미들웨어를 적정 수준 이상으로 검사할 수 있는 테스트 툴을 필요로 하고 있는 실정이다.

본 연구에서는 UPnP 스펙에 정의된 UPnP 프로토콜과 네트워킹 방식을 분석하고 UPnP 기능을 가지도록 구현된 홈 네트워크 미들웨어를 테스트할 수 있는 방법과 UPnP 테스트 툴의 구조를 제시하고자 한다. 본 논문의 구성은 2장에서 UPnP 스펙에 정의된 UPnP 프로토콜과 네트워킹의 각 단계들을 분석하고, 3장에서 UPnP 테스트 툴의 구조 및 동작 방법과 UPnP 스펙(stack)을 구현한 미들웨어의 테스트 방법을 제시하며, 끝으로 4장에서는 본 연구에서 구현되어져 실제 적용된 UPnP 테스트 툴의 적용 효과와 결론을 제시한다.

2. UPnP의 구조 및 네트워킹 방식

UPnP는 IP, TCP, UDP, HTTP 및 XML 기반으로 구성되어 있

으며, 홈 네트워크 상의 장치 스스로가 동적으로 네트워크에 참여하고, IP 주소를 획득하고, 장치의 이름을 공표하고, 장치의 기능을 알리고, 다른 장치들의 기능을 발견하는 기능을 할 수 있는 자동 인식(Automatic Discovery) 기능을 제공하고 있다[2].

2.1 UPnP 프로토콜 스택(stack)

그림 1은 UPnP에서 사용되는 프로토콜 구조를 도시한 것으로 각 프로토콜에 대한 설명은 다음과 같다.



그림 1. UPnP 프로토콜 구조

- TCP/IP : UPnP 장치간의 네트워크 연결성을 제공해주는 기본 프로토콜로 사용되고 있다.
- HTTPU/HTTPMU : HTTP의 변형으로 SSDP에서 사용되어지며 " HTTP (Unicast) over UDP" 와 " HTTP Multicast over UDP" 를 의미한다.
- SSDP(Simple Service Discovery Protocol) : 네트워크 상에서 어떻게 네트워크 서비스를 찾아낼 수 있는가를 정의하고 있다.
- GENA(General Event Notification Architecture) : " HTTP over TCP/IP" 또는 " Multicast over UDP" 를 통하여 경보를 받거나 보내는 기능을 한다.
- SOAP(Simple Object Access Protocol) : 원격 프로시저(procedure)를 실행하기 위한 XML과 HTTP 사용에 대한 정의를 하고 있다.

2.2 UPnP 네트워킹 방식

UPnP 네트워킹 방식은 6개의 단계(step)로 구성되어져 있는데, 각 단계에 대한 설명은 다음과 같다.

- Addressing Step : 모든 UPnP 장치들은 DHCP(Dynamic Host Configuration Protocol) 클라이언트를 보유하고 있어 네트워크에 연결되었을 때 DHCP 서버를 검색하고 DHCP 서버로

부터 자신의 IP 주소를 할당 받는다. 만일 DHCP 서버가 존재하지 않거나 DHCP 서버로부터 IP 주소를 할당 받는데 실패하면 Auto IP를 이용하여 자신의 IP 주소를 얻게 된다.

• **Discovery Step** : 피제어 장치가 네트워크에 접속되면 자신이 가지고 있는 장치의 정보와 서비스 정보를 SSDP에 의해 네트워크 상의 제어 장치에게 통보하며, 이와 비슷하게 제어 장치가 네트워크에 접속하게 되면 SSDP에 의해 네트워크 상의 관심 있는 피제어 장치들을 검색한다.

• **Description Step** : **Discovery Step** 이 후 제어 장치는 피제어 장치에 대하여 극히 일부분의 정보만을 가지게 되는데 피제어 장치와의 상호 작용을 위해서는 피제어 장치의 더 많은 정보를 습득해야 한다. 이를 위해 제어 장치는 XML로 표현된 피제어 장치의 자세한 정보(모델 번호, 장치 이름, 시리얼 번호, 제조사 등)를 수집하게 된다.

• **Control Step** : 제어 장치는 피제어 장치로부터 받은 장치 및 서비스에 대한 XML 정보(Device & Service Description)를 분석하여 SOAP를 이용한 제어 메시지를 송신하고, 피제어 장치는 XML로 표현된 제어 메시지에 대한 응답으로 서비스 기능 수행에 따른 특별한 값을 리턴(return)하거나 미리 정의되어진 error code를 리턴한다.

• **Eventing Step** : 피제어 장치의 서비스 정보(XML)에는 서비스 상태를 알리는 변수들이 존재하고 이러한 변수에 변화가 생겼을 경우에는 제어 장치가 피제어 장치의 서비스 상태 정보를 수신할 수 있어야 한다. 피제어 장치로부터 발생한 이벤트 메시지는 GENA를 이용한 XML 형식으로 상태 변수에 대한 이름과 현재의 값을 포함하고 있다.

• **Presentation Step** : 피제어 장치가 프리젠테이션(presentation)을 위한 URL을 가지고 있는 경우, 제어 장치는 이 URL로부터 페이지(page)를 가져와 피제어 장치의 상태를 보거나 홈 네트워크 사용자로 하여금 제어 장치를 제어할 수 있는 UI(User Interface)를 제공하는 기능을 가진다. 이 경우 페이지는 웹 브라우저(browser)를 사용하여 볼 수 있다.

3. UPnP 테스트 방법 및 테스트 툴

본 장에서는 UPnP 테스트 툴의 구조 및 동작 방법, 그리고 테스트 방법에 대하여 설명한다.

3.1 UPnP 테스트 툴의 구조

그림 2는 UPnP 테스트 툴의 구조를 나타낸 것으로 네트워크 통신 채널을 통한 데이터 송/수신을 처리하는 NCL(Network Connection Logic), 테스트 케이스들을 저장하고 있는 Test Suite DB와 테스트 결과 및 테스트에 대한 모든 기록을 저장하고 있는 테스트 로그 DB의 파일 입/출력을 관리하는 FIOL(File In/Out Logic), 테스터(tester)가 UPnP 테스트 툴의 UI(User Interface)에서 입력한 명령을 해석하고 이를 다시 DL(Display Logic)과 TMCL(Test Manager Control Logic)에 전달하는 UIL(User Input Logic), 테스터의 입력(테스트 케이스 선택 및 테스트 실행/중지 등) 및 테스트 결과, 그리고 선택한 테스트 케이스의 설명 등을 사용자에게 보여주는 DL, 마지막으로 UIL에서 전달되어온 명령을 해석하여 테스트의 케이스 하나 하나의 실행을 수행하고 수행된 테스트 케이스의 결과와 그 결과를 분석하여 DL에 전달하는 기능과 각 로직(logic)간에 상호 작용을 관리하는 TMCL로 구성되어 있다.

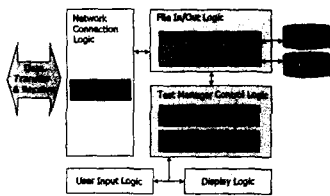


그림 2. UPnP 테스트 툴 구조

3.2 UPnP 테스트 툴의 동작 방법

UPnP 테스트 툴의 동작 방법을 설명하기 위하여 그림 3과 같이 Host B에 UPnP CD(Controlled Device) 스택이 구현되어진 피제어 장치가 존재하고 UPnP 테스트 툴이 Host A에서 실행된다고 가정하면 먼저 테스트 툴의 UI에서 테스트 하고자 하는 테스트 케이스들을 선택하고 "test" 버튼을 클릭한다.



그림 3. UPnP CD 테스트를 위한 네트워크 구성

테스터의 테스트 실행 명령에 의하여 UPnP 테스트 툴은 그림 2의 UI로부터 받아들인 명령을 DL에 전달하여 테스터가 선택한 테스트 항목들과 테스트 진행 사항들을 디스플레이 할 수 있도록 하며, 테스터의 명령 해석 및 실제 테스트가 이루어질 수 있도록 TMCL로 테스터의 명령을 전달하게 된다. TMCL은 테스터가 선택한 테스트 케이스 파일들을 읽어오기 위하여 FIOL로 선택 테스트 케이스들에 대한 이벤트를 보내게 되고 FIOL에서는 Test Suite DB를 통하여 테스터가 선택한 테스트 케이스 파일들 중 맨 처음의 테스트 케이스 파일을 다시 TMCL로 보내게 된다. 이 과정에서 받은 테스트 케이스 파일을 TMCL은 분석하고 UPnP 테스트에 필요한 환경 구축 작업으로 송신할 메시지를 만든 후 필요한 프로토콜을 설정하여 NCL에 메시지를 전달한다. TMCL로부터 받은 메시지를 UPnP 네트워킹 프로토콜에 맞게 타겟(to be tested) Host B에 송신하고 타겟 Host B의 응답 메시지를 기다리게 된다. UPnP 테스트 툴이 Host B의 응답 메시지를 수신하게 되면 NCL은 이를 TMCL로 전달하게 되고, TMCL은 이를 분석하여 Host B의 첫 번째 테스트 케이스의 "Pass/Fail" 판정을 내린 다음, 이에 대한 테스트 로그를 FIOL을 통하여 Test Log DB에 저장하며 테스트 결과와 로그를 DL에게 넘겨 테스터에게 디스플레이 하도록 한다. 이 같은 방식으로 두 번째 테스트 케이스 및 그 이후의 다른 테스트 케이스들도 동작하여 모든 테스트 케이스들에 대한 테스트가 끝나게 되면 UPnP 테스트 툴은 UI를 통하여 테스터에게 테스트의 전체 결과 및 로그를 디스플레이하고 테스터의 다음 명령을 기다리게 된다.

3.3 UPnP 테스트 방법

UPnP 스택이 구현된 네트워크 미들웨어의 테스트 방법은 제어 장치와 피제어 장치의 테스트 방법으로 나눌 수 있으나, UPnP 네트워킹은 두 장치간에 상호 네트워킹을 하는 구조로 되어 있으므로 제어 장치에 대한 테스트 방법은 피제어 장치에 대한 테스트 방법과 대칭적이다. 따라서 본 논문에서는 피제어 장치에 대한 테스트 방법만을 UPnP 네트워킹 단계별로 언급하도록 하며 피제어 장치의 IP 주소를 얻는 Addressing Step과 프리젠테이션 페이지를 제공하여 웹 브라우저(browser)에서 피제어 장치를 제어할 수 있도록 기능을 제공하는 Presentation Step의 테스트 방법은 본 논문에서 설명하지 않는다.

3.3.1 Discovery Step

그림 4는 UPnP 테스트 툴과 피제어 장치 간의 Discovery Step 네트워킹을 나타낸 것으로서, 피제어 장치에서는 할당된 IP 주소를 통해 네트워크 상의 제어 장치들에게 자신의 장치와 서비스 정보를 알리는(advertise) 메시지(NOTIFY)를 멀티캐스트(multicast)하여 자신의 장치를 통해 제공되는 서비스를 알리면 피제어 장치의 제어기 가능하도록 네트워크 상의 역할(제어 장치의 역할)이 결정된 UPnP 테스트 툴에서는 멀티캐스트로 제공되는 메시지를 받아 제어 대상으로 등록함과 동시에 메시지의 포맷(format)이 UPnP 스택에서 정의한 포맷과 같은지 비교하게 된다. 반면에, 네트워크에 새로 연결되는 UPnP 테스트 툴의 경우는 할당된 IP 주소를 이용하여 네트워크 상에서 동작하는 피제어 장치들을 확인하기 위해 SSDP와 HTTPMU를 통하여 검색

메시지(M-SEARCH)를 멀티캐스트 한다. 이때 UPnP 테스트 툴로부터 멀티캐스트 되는 M-SEARCH 메시지를 수신하는 피제어 장치에서는 수신된 M-SEARCH 메시지에 대한 응답으로 UPnP 테스트 툴에게 응답 메시지(M-SEARCH Response)를 유니캐스트(unicast) 한다.

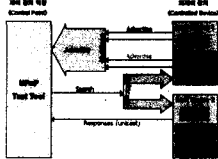


그림 4. Discovery Step 네트워크

이러한 네트워킹 과정에서 일어날 수 있는 모든 경우의 수를 여러 테스트 케이스들로 구현하고, 구현된 테스트 케이스들은 테스트 실행 명령을 통하여 피제어 장치로부터 수신된 메시지와 UPnP 스택에 정의된 메시지 포맷을 비교한다. 또한 비정상적인(invalid) 메시지를 피제어 장치에게 전송하거나, 같은 메시지를 계속적으로 피제어 장치에게 전송하는 등의 비기능적인 테스트 방법을 통하여 피제어 장치의 안정성(Stability)과 강건성(Robustness) 등을 테스트한다.

Description, Control, Eventing Step의 테스트 케이스들을 구현하는 방법도 이와 유사하며 테스트 케이스들의 통과(Pass)/실패(Fail) 판정을 내리는 방법의 경우는 Discovery Step과 동일하다.

3.3.2 Description Step

UPnP 테스트 툴은 그림 5와 같이 Discovery Step에서 알게 된 피제어 장치의 IP 주소를 이용하여 피제어 장치에 대한 정보(Device Description, XML로 작성됨)를 HTTP GET 방식으로 요청한다. 이에 따라, 피제어 장치에서는 요청된 XML 파일 형태의 Root Device Description을 UPnP 테스트 툴에 전송한다. 또한, UPnP 테스트 툴에서는 피제어 장치에서 제공되는 서비스에 대한 정보(Service Description)를 정상적으로 제공하는지 여부를 파악하기 위해 피제어 장치의 Service Description을 요청하고, 피제어 장치에서는 XML 파일 형태의 Service Description을 UPnP 테스트 툴로 전송한다.

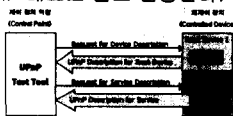


그림 5. Description Step 네트워크

3.3.3 Control Step

그림 6은 UPnP 테스트 툴과 피제어 장치간의 Control Step 네트워킹 과정을 나타내는 것으로서, UPnP 테스트 툴에서는 Description Step에서 얻은 Device & Service Description을 참조하여 피제어 장치에게 서비스 제어 메시지(Action Invoke)를 전송하면, 제어 메시지를 수신한 피제어 장치에서는 제어 메시지에 따라 요구되는 서비스 기능을 실행시키고 그 결과 메시지를(Action Response)를 UPnP 테스트 툴로 전송한다. 또한, UPnP 테스트 툴에서는 제어 장치의 서비스 변수에 대한 값을 알아내기 위하여 쿼리 메시지(Query Invoke)를 피제어 장치에게 전송하고 피제어 장치는 요청 받은 서비스 변수 값을 UPnP 테스트 툴에게 알려주게(Query Response) 된다.

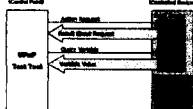


그림 6. Control Step 네트워크

3.3.4 Eventing Step

UPnP 테스트 툴은 Control Step을 통해 수행되는 네트워킹 동작에 따라 피제어 장치의 서비스 정보 변경 상태를 확인하기 위해 그림 7과 같이 Subscription을 요청하는 메시지(SUBSCRIBE)를 피제어 장치로 전송한다. 그러면, 피제어 장치에서는 UPnP 테스트 툴의 Subscription 요청에 따라 유일한 Subscription ID를 할당하고 초기 이벤트 메시지를 전송하게 된다. 이 때 UPnP 테스트 툴과 피제어 장치의 관계를 subscriber와 publisher의 관계에 있다고 말한다.

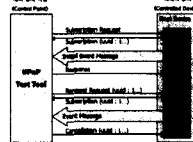


그림 7. Eventing Step 네트워크

UPnP 테스트 툴이 Subscription 상태에 있는 동안 피제어 장치의 서비스 상태 변수가 변경되면, 피제어 장치에서는 변경된 정보를 알리기 위하여 GENA를 통해 XML 형태의 이벤트 메시지를 UPnP 테스트 툴로 전송한다. UPnP 테스트 툴에서는 이벤트 메시지가 수신되면 이벤트 메시지에 기록된 서비스 변경 정보를 분석하여 피제어 장치에 대한 서비스 정보를 업데이트하고, 피제어 장치로 응답 메시지(Response)를 전송한다. 만일 Subscription TIMEOUT 시간 안에 Subscription 재갱신이 필요하면 UPnP 테스트 툴에서는 할당 받은 Subscription ID로 피제어 장치에 Subscription 연장을 요청하는 메시지(Renewal Subscription)를 전송한다. 피제어 장치는 UPnP 테스트 툴의 Renewal 요청에 초기 이벤트 메시지를 전송하여 응답한다. 또한, 더 이상 피제어 장치의 이벤트 메시지 전송이 요구되지 않으면, UPnP 테스트 툴은 Subscription을 취소하는 메시지(UNSUBSCRIBE)를 피제어 장치로 전송한다.

4. 결론

본 논문에서는 표준 기술로서 제안된 마이크로소프트사의 UPnP 프로토콜 구조와 네트워킹 방식에 대하여 살펴보고, UPnP 스택이 구현된 UPnP 미들웨어 테스트 방법과 테스트 툴의 구조를 제시하였다.

본 연구의 UPnP 테스트 방법 및 테스트 툴은 실제 기업에서 개발되어 사용하였던 테스트 사례로서 홈 네트워크 임베디드 시스템에 탑재되는 UPnP 미들웨어를 테스트하였으며, UPnP 미들웨어 테스트에 대한 Manual과 Automated 테스트에 대한 사례 연구를 통하여 Automated 테스트의 효율성과 재사용성, 그리고 그 중요성을 입증한 바 있다[3].

본 연구에서 제시된 UPnP 테스트 방법 및 자동화 테스트 툴의 구조는 UPnP 표준 네트워킹 동작에 따른 다양한 테스트 케이스들을 생성 가능하게 하고 홈 네트워크 상의 피제어 장치 및 제어 장치를 대상으로 UPnP 네트워킹을 자동으로 테스트 할 수 있는 테스트 툴 개발에 많은 기여를 할 것으로 기대된다.

[참고 문헌]

[1] 김동희, 이화영, 임경식, "정보가전기기간 상호운용성 지원을 위한 OSGi 기반 UPnP 기술", 대한전자공학회 전자공학회지, 제29권 제6호, pp.60~67, 2002.
 [2] 임승욱, 정광모, "UPnP(Universal Plug and Play) 기술 분석", 전자정보센터(EIC) 원고, 2003. 3.
 [3] 장영숙, 여기동, 이현동, "Manual과 Automated 테스트에 대한 사례 연구", 2003 한국정보과학회 가을학술발표논문집(II), 제30권 2호, pp.343~345, 2003.
 [4] Microsoft, " Universal Plug and Play Device Architecture version 1.0", http://www.upnp.org/download/UPnPDA10_20000613.htm, Jun. 2000