

다중 홈 네트워크를 위한 미들웨어

Middleware for Heterogeneous Home Network

박흥성, 김형욱

강원대학교 전기전자정보통신공학부

1. 서론

최근 각 가정에 초고속 통신망 설치를 통해 인터넷 뿐만 아니라 상대적으로 큰 대역폭이 필요한 멀티미디어 서비스를 집에서 즐길 수 있는 시대가 되었다. 이에 발 맞추어 홈 네트워크에 관한 관심이 높아져 가고 있다. 홈 네트워크 기술이란 가정 내 냉장고나 TV와 같은 가전기기를 네트워크로 연결하여 서로의 정보를 공유하거나 한 쪽 기기에서 다른 쪽 기기를 제어하는 기능을 지원할 뿐만 아니라, 인터넷 접속을 통하여 원격지와의 정보 공유 및 동작제어를 가능하게 해주는 분산환경에서의 시스템 기술을 말한다. 가정 내 캠코더, Home Theater 장비, 디지털 TV, 냉장고, 방범 시스템 등의 기능과 사양이 높아지고 고급화되어 감에 따라 이들 디지털 기기들 서로 연동되어 새로운 서비스를 제공하는 것이 요구되고 있으며 이는 다양한 종류의 통신 인터페이스를 통해 기기들 연동이 가능하게 되었다. 이에 따라 디지털 가전 기기들은 상호 연동하여 기기 간에 정보를 교환하고 제어할 수 있을 뿐만 아니라 인터넷을 통한 원격 서비스도 제공할 수 있게 되었다. 이러한 홈 네트워크가 가져야 하는 특성을 살펴보면 다음과 같다: 특별한 기술없이 쉬운 연결, 쉬운 배선, 저 비용, 고 신뢰성, 비디오, 음성과 같은 다양한 데이터를 수용하는 전송 용량 등.

홈 네트워크 전송기술을 보면 가정에 배선되어 있는 전화선 또는 전력선과 같은 기존의 인프라를 그대로 활용하여 저렴한 비용으로 구축하기 위한 방안으로 HomePNA나 PLC, LonWorks기술이 있지만 이들 방식은 아직 지원되는 전송속도가 저속이므로 이를 홈 네트워크 전체의 백본망으로 적용하기에는 어려움이 있다. PLC 기술과 LonWorks 기술은 일반적으로 가스, 냉장고 등의 가전 기기들을 제어하는데 사용되는 것이 일반적이다. 이에 반해 Ethernet이나 IEEE1394 기술은 멀티미디어 통신과 같은 광대역 서비스를 제공할 수 있다는 장점을 갖고 있지만 최근 새롭게 지어지는 주택 및 아파트를 제외하고는 케이블을 새로 가설해야 한다는 문제가 있고 가정과 같은 환경에 도입하기에는 고가인 단점이 있다. 한편 배선이 필요 없고 단말의 이동성을 제공하는 HomeRF나 Bluetooth, IEEE 802.11/15와 같은 무선 기술은 전송거리, 접속 노드 수의 부족, 상호 간섭 등의 문제가 남아 있다. 이러한 다양한 홈 네트워크 기술은 기술의 개방을 무기로 하여 포럼과 같은 조직을 만들어 사용을

확대하고 있다. 이러한 다양한 홈 네트워크들은 앞에서 언급한 것들을 모두 만족하지는 못하고 있다.

홈 네트워크의 종류를 사용별로 살펴보면 1 Mbps 이하의 제어 네트워크, 1-100 Mbps의 데이터 네트워크와 100-400 Mbps의 A/V 네트워크로 나눌 수 있다. 제어 네트워크는 데이터들의 전송 신뢰성, 안정성 및 저가로 구축할 수 있어야 하며, 데이터 네트워크는 PC 및 일반 OA 기기들을 연결할 수 있어야 하며, A/V 네트워크는 고속의 멀티미디어 데이터를 처리할 수 있도록 A/V 기기들을 연결할 수 있어야 한다.

앞에서 언급한 내용들을 요약하면, 홈 네트워크는 어느 한가지 방식만으로 구현된다기 보다는 홈 네트워크의 환경과 단말기의 특성, 요구되는 서비스의 종류 및 비용측면을 고려하여 유무선 방식들을 혼합하여 여러 홈 네트워크들이 혼합된 구조로 발전할 것이다.

실제로 홈 네트워크에 사용 가능한 네트워크 인터페이스에는 IEEE1394, USB, Ethernet, LonWorks, PLC, HomePNA, Wireless LAN, Bluetooth, IEEE 802.11, IEEE 802.15, HomeRF, CAN 등 다양하고 많은 종류가 있다. 이와 같은 네트워크들은 전송 속도, 대역폭 등 특성이 종류별로 각각 다르다. 또한 네트워크 인터페이스를 이용하여 데이터를 외부 기기에 전송하기 위해서 사용되는 각 네트워크 인터페이스의 프로토콜 및 사용 방법 등도 모두 다르다. 이러한 이종 네트워크 기반 홈 네트워크에서 중단 디지털 가전기들 데이터를 전송하기 위해서는 여러 사항을 고려해야 한다. 예를 들어 TCP/IP기반의 네트워크를 사용하는 가전기기는 홈 네트워크상의 다른 가전기기에 데이터를 전송하기 위해 IP와 Port를 사용한다. 그리고 CAN이나 USB등과 같이 TCP/IP기반의 네트워크가 아닌 인터페이스를 사용하는 가전기기의 경우에는 Node ID, Offset 등 TCP/IP기반과는 다른 어드레싱 방법을 사용한다. 또한 이종 네트워크 기반에서 어드레싱 방법이 다른 네트워크간 데이터를 전송하기 위해서는 각 가전기기가 홈 네트워크상에 존재하는 가전기기가 사용하는 네트워크 종류에 따른 상대적인 주소를 알고 있어야 한다. 다시 말하자면 두 가지 이상의 종류가 다른 네트워크를 이용하여 데이터를 전송할 경우, 최종 목적지의 가전기기가 사용하는 인터페이스 종류에서 사용하는 어드레싱 방법과 그에 대한 주소를 전송측 가전기기에서 알고 있어야 한다는 것이다.

이와 같이 다양한 종류의 네트워크로 구성된 홈 네트워크

환경에서 디지털 장비들간에 정보를 공유하고, 가정 내부의 장비들을 원격에서 제어하거나 정보를 검색하는 등 다양한 서비스를 홈 네트워크 사용자에게 제공하기 위해서는 장비간 데이터가 서로 다른 네트워크를 통해 전송될 수 있어야 한다. 그러나 홈 디지털 기기를 개발하거나 서비스를 제공하는 개발자가 이렇게 다양한 네트워크 환경을 일일이 고려하여 개발하는 것은 매우 어려운 일이다. 그러므로 이러한 다양한 종류의 홈 네트워크가 장착된 가전기기들을 서로 연결할 수 있는 미들웨어 기술이 필요하다. 미들웨어는 그림 1과 같이 사용되는 H/W와 네트워크 인터페이스의 상세 사항을 모르더라도 응용 프로그램을 개발할 수 있게 하는 기술을 말한다.

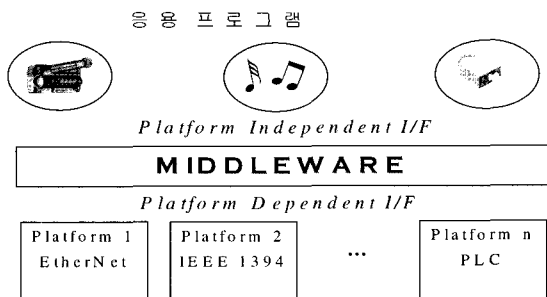


그림 1. 미들웨어의 위치

현재 홈 네트워크용으로 개발된 미들웨어는 HAVi[1], OSGi[2], UPnP[3], Jini[4], ECHONET[5] 등 다양한 미들웨어 표준들이 제시되어 있다. 이러한 미들웨어는 현재 IP 기반으로 네트워크 상에 존재하는 원격 디지털 가전 기기에 데이터를 전송하거나 수신한다. IP 기반이 아닌 디지털 가전기와 접속하기 위해서 ECHONET은 바로 직접 접속할 수 있지만 다른 미들웨어는 브리지 혹은 게이트웨이를 통하여 간접적으로 접속한다. 본 고에서는 이러한 미들웨어에 대해 동작과 장단점을 분석하고 홈 네트워크가 가져야 할 미들웨어의 성질을 살펴보고 이러한 미들웨어의 구현 가능성을 살펴본다.

2장에서는 기 제안된 홈 네트워크용 미들웨어에 대해 분석하고 3장에서는 2장에서 언급한 미들웨어를 비교하고 홈 네트워크가 가져야 할 미들웨어의 성질을 살펴본다. 마지막으로 4장에서 결론을 맺는다.

2. 홈 네트워크 미들웨어

2.1. HAVi (Home Audio Video Interoperability)

Home Entertainment 서비스에 초점을 맞춘 미들웨어로서 가정 내의 오디오 및 비디오 가전 기기간의 상호운용성(장비간의 통신 및 제어 기능 등)을 위한 홈 네트워크용 표준으로 정의되었으며 1996년 Sony와 Philips에 의해 시작되었으며 차후 오디오 및 비디오 장비를 생산하는 6개 주요

회사 (Grundig AG, Hitachi, Matsushita(Panasonic), Electrics, Sharp, Thomson Multimedia)가 참여하였다. 현재는 국내의 삼성, LG 전자를 포함하여 50여 개 이상의 업체들이 참여하고 있다.

HAVi를 사용한 네트워크 구성 예는 그림 2에 있으며, HAVi의 내부 구조는 그림 3에 표시되어 있다. HAVi를 구성하는 소프트웨어 요소들은 네트워크 관리, 디바이스 추상화(device abstraction), 디바이스간 통신, 디바이스 사용자 인터페이스 관리 등과 같은 서비스들을 제공한다. HAVi 장비들 사이의 상호 운용성을 위해 필요한 소프트웨어 요소에는 Messaging system, Event Manager, Registry, Resource Manager, DCM(Device Control Module) Manager 등이 있다. HAVi 장비에 있는 소프트웨어 요소들은 다른 소프트웨어 요소에게 메시지를 보냄으로써 다른 소프트웨어 요소의 API를 호출한다. 소프트웨어 요소가 다른 HAVi 장비에 존재할 경우 Messaging system이 네트워크 상으로 메시지를 보낸다. 사용자는 새로운 HAVi 장비들을 가지고 HAVi 네트워크를 확장시킬 수 있고 동시에 여러 가지 응용 프로그램을 실행할 수 있다. 응용 프로그램이 동일한 DCM을 제어하려고 하는 경우, Resource Manager를 이용하여 충돌을 해결함으로써 장비의 공유를 가능하게 한다.

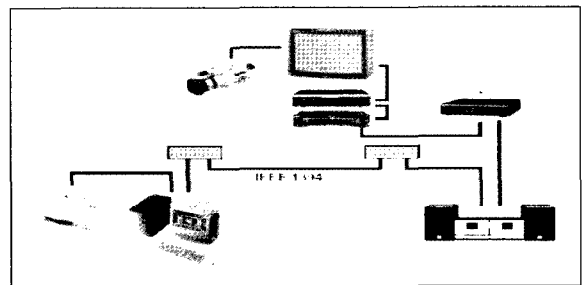


그림 2. HAVi의 네트워크 구성 예

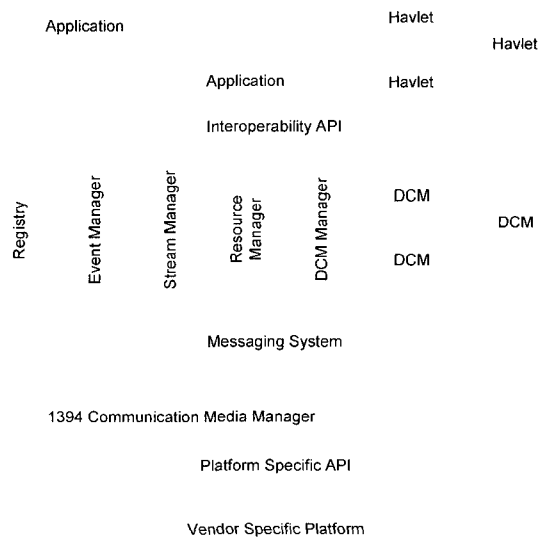


그림 3. HAVi의 내부 구조

DCM은 HAVi 네트워크에서 하나의 단일 디바이스를 나타내는 소프트웨어 요소로서 해당 디바이스에 대한 HAVi defined APIs 들을 제공한다. 이러한 API는 모든 HAVi 장비에 있는 응용 프로그램에 의해 호출되며, DCM은 HAVi로 정의된 API를 장비에 정의된 제어 인터페이스상의 명령어로 전환해준다. 디바이스가 네트워크에 추가되거나 제거될 때 해당 DCM이 설치되거나 제거된다. Havlet은 가전 기기들을 제어하기 위한 사용자 인터페이스를 제공하는 응용 프로그램이다. Communication Media Manager는 가전 기기들이 1394 기반 홈 네트워크를 통해 통신하게 하는 요소이다. Messaging System은 네트워크로 연결된 다른 가전기기들간 메시지를 전달한다. Event Manager는 이벤트를 관리하고 알맞은 프로그램에게 해당 내용을 알려준다. Registry는 홈 네트워크상에 있는 소프트웨어 객체의 위치를 등록하는데 사용되며 directory service로써 동작한다. Resource Manager는 네트워크상의 자원들을 배분하고 할당한다.

HAVi 장비들은 각각의 역할과 복잡도에 따라서 제어기와 피제어기로 구분된다. 제어기는 피제어기를 제어하는 장비로 기능과 복잡도에 따라서 다시 FAV(Full AV)와 IAV(Intermediate AV)로 나뉘고, 피제어기는 BAV(Basic AV)와 LAV(Legacy AV)로 구분된다. FAV는 Java Runtime 환경을 가지고 있어서 다른 장비로부터 프로그램(Bytecode)을 넘겨받을 수 있고 피제어기들에 향상된 기능을 제공할 수 있다. DTV(Digital TV), STB(Set-Top Box), 홈 서버 등이 FAV의 예이다. IAV는 FAV에 비해서 가격도 저렴하고 제한된 자원을 갖는 가전기기를 말하며 Java Run-time 환경을 가지지 못하여 임의 장비에 대한 제어기 역할을 못한다. 그러나, 가전기기들에 대하여 기본 기능의 제어기능 보유한다. BAV는 ROM에 자신의 제어 정보를 보유하여 FAV 혹은 IAV에 의해 제어되는 가전기기를 말한다. LAV는 HAVi SDD(Self-Description Data)가 없는 장비 혹은 HAVi를 지원하지 않는 기존 장비를 말하며, FAV나 IAV에서 Legacy 장비의 프로토콜로 변환하여 제어해야 한다.

제품이 연결되면 자신의 존재와 기능을 자동으로 알려서 네트워크 주소나 기기 드라이버 등의 설치 없이 간편하게 장비를 설치하고 바로 사용할 수 있다. 중앙에 서버형태의 기능을 두지 않고 기기간 peer-to-peer 통신을 통해 각 기기들을 제어할 수 있는 장점이 있는 반면 AV 기기를 제외한 다른 가전기와 연동하기 위해서는 다른 미들웨어와 연동이 필요하고 하부 네트워크로 IEEE1394만을 사용하는 것이 단점이다.

2.2. OSGi(Open Service Gateway initiative)

OSGi 는 통신 가전 컴퓨터 등 각 분야의 세계적인 업체들이 1999년 3월 설립한 비영리 민간단체로 주로 홈 게이

트웨이 서비스를 정의하는 산업표준화에 참여하고 있다. 도이치텔레콤, 프랑스텔레콤, 노키아 등 통신 업체와 삼성 전자, 지멘스, 샤프, 도시바 등 가전업체, 그리고 컴팩, HP, 썬 마이크로시스템스 같은 컴퓨터 관련업체 등 80여 개 회원사를 갖고 있다. 각 산업분야별 선두업체들로 구성된 OSGi는 개방형 구조인 '자바 임베디드 서버' (JES) 기반의 게이트웨이 아키텍처용 소프트웨어를 만드는 것을 주요 목적으로 하고 있다. 각 업체는 자신의 기술을 제공, 이를 공유해 모든 업체의 제품에서 공통으로 사용할 수 있는 프로토콜이나 아키텍처를 개발하고 있다. OSGi는 그림 5와 같은 구조를 이용하여 그림 4의 다양한 표준들을 지원하고 있다. 이러한 표준들은 드라이버로 제공되어야 한다.

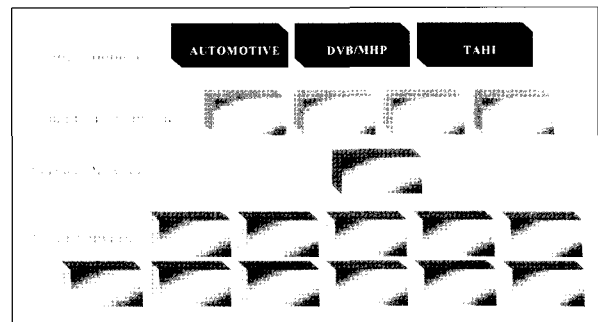


그림 4. OSGi와 관련 표준들

OSGi는 Service Gateway를 정의하는 것으로서 Service Gateway는 이종 디바이스나 이종 프로토콜들이 서로 알아들을 수 있는 데이터로 변환하는 역할을 하며 Service, Bundle, Framework의 세가지 컴포넌트로 구성되어 있다. OSGi의 Application은 Bundle이라고 한다.

OSGi는 자바기반의 Application Layer Framework로서 Device Access manager, Log service, HTTP service등의 API로 구성된다.

Framework가 하는 기능은 응용 프로그램들을 Java VM에 공유시키고, 라이프 사이클, 보안, 어플리케이션간 독립성 등을 관리하며 Bundle은 배달 가능한 어플리케이션으로 윈도우에 EXE파일과 비슷하다. 여러 개의 번들이 동시에 수행될 수 있으며, 다른 번들에 의해서 서비스를 찾아가기도 한다. Framework는 그 자체가 system 번들로서 표현된다. 번들은 java Archive (JAR) file 로써 전달되는데 동작은 라이브러리 형태(ex DLL)와 비슷하게 동작하며 Framework에서는 다이내믹하게 번들을 Install/update/uninstall 할 수 있다. Service는 번들에 의해서 Framework에 등록된 객체이며 유일한 아이디를 갖는다. Service의 포맷은 Java interface에 의해서 정의 되어 있다. 여러 이종 네트워크에 Plug & Play가 지원되는데 연결을 하면 Device Manager가 번들의 형태로 적합한 드라이버를 다운로드 한다. 이것은 Driver selector, Driver locator라는 것이 있어 드라이버를 찾아온다.

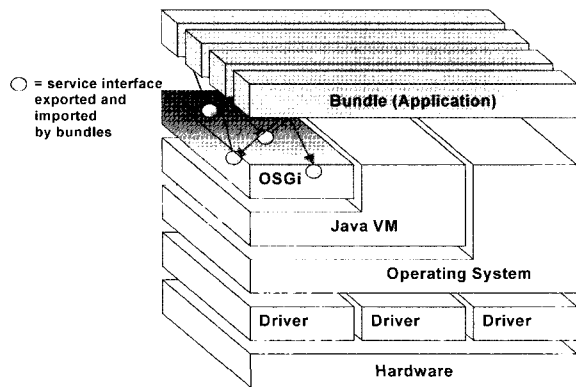


그림 5. OSGi 구조

OSGi의 표준안은 특히 블루투스, HomePNA, HomeRF, IEEE1394, LonWorks, USB 등 다양한 유무선 네트워크 기술을 수용할 수 있어 포괄적인 개방형 네트워크 기술로 인정 받고 있으며 특히 개방형 서비스 게이트웨이는 Jini, HAVi, UPnP 등도 수용한다. 그러나 이러한 내용을 지원하기 위해서는 bundle을 만들어야 한다. OSGi는 일종의 gateway 기술이기 때문에 다른 종류의 홈 네트워크들 간의 통신, 예를 들면 블루투스와 IEEE 1394간의 통신을 하기 위해서는 블루투스 번들, IEEE 1394 번들 외에 블루투스와 IEEE 1394간의 게이트웨이 기능을 하는 번들을 만들어야 한다. 따라서 이종 네트워크간에는 peer-to-peer 통신을 못하고 OSGi를 거쳐서 통신을 하게 되므로, 이종 네트워크간 통신 량이 많아지면 OSGi 역할을 하는 시스템의 부하가 많아져 처리 속도가 늦어지는 단점이 생길 수 있다. 또한 Java 만을 지원하는 단점도 있다.

2.3. UPnP (Universal Plug and Play)

홈 네트워킹의 확산으로 다양한 하위 네트워크 기술이 사용되는 환경으로 바뀌면서, 가전제품 들은 IP 프로토콜을 사용하여 독립적이며 통일된 방식으로 네트워크화 할 수 있는 기술에 대한 필요성이 제기되었고, 마이크로소프트사(MS)에 의해 UPnP가 제안되었다. 현재 500개 이상의 업체가 UPnP 포럼에 가입하고 있으며, 현재 활동중인 working committee로는 AV(Audio Video), IGD(Internet Gateway Device), Home Automation, image/printing, Security가 있으며, AV와 IGD가 가장 활발하게 활동 중이다. MS, Intel, Allegro 에 의해 SDK가 개발되고 있으며, Metro Link 가 Java용 Embedded System을 위한 UPnP SDK 개발되었다. 이미 국내의 가전사들도 UPnP 포럼에 참여하고 있다. 2002년초부터 UPnP를 지원하는 제품들이 나오기 시작하였다.

UPnP의 소프트웨어 구조가 그림 6에 있고, UPnP 기기와 Legacy 기기와의 연결하는 구성의 예가 그림 7에 있다. 그림 6과 그림 7로부터 UPnP는 IP 기반 프로토콜을 사용하기 때문에 이더넷, LonWorks, X10, IEEE 1394

등의 다양한 네트워크들이 IP를 지원하면 UPnP를 사용할 수 있다는 것을 알 수 있다. 또한 IP가 없을 경우에는 UPnP 프로토콜을 legacy protocol로 변환하는 브리지를 사용하는 것을 그림 7에서 보여 주고 있다. 따라서 UPnP를 사용하여 모든 홈 네트워크를 연결할 수 있다. 제어 포인트는 피제어 정보 가전기기(controlled information appliance)들을 제어하기 위한 소프트웨어 모듈의 집합으로 표준 PC, 셋톱 박스등이 예이다. 피제어 정보 가전기기들은 제어 포인트와 통신을 하는 소프트웨어 모듈의 집합을 말하며 항상 수동적으로 제어 포인트의 명령을 기다린다. 이러한 기기의 예에는 VCR, DVD 플레이어, 조명제어기 등이 있다.

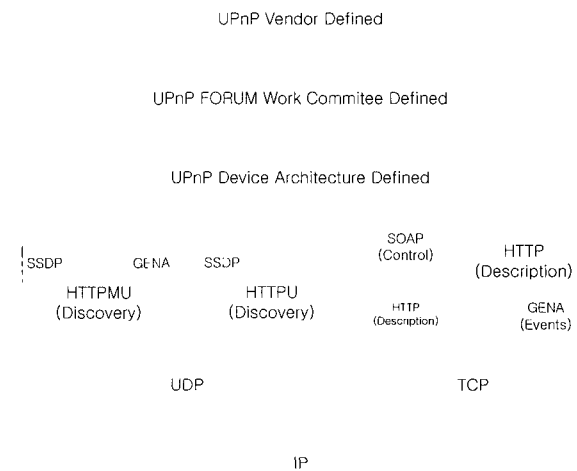


그림 6. UPnP 구조

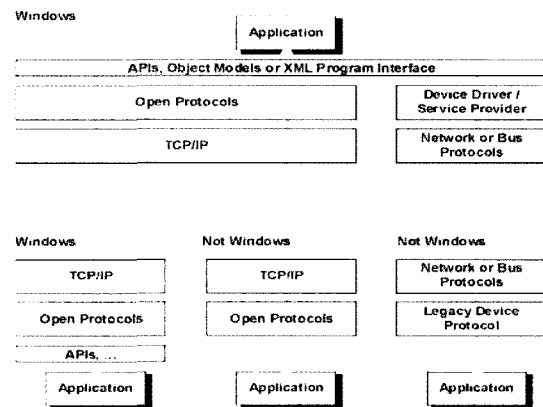


그림 7. UPnP 기기와 Legacy 기기와의 연결

UPnP 프로토콜의 핵심은 주소 지정(addressing)이다. 각 기기는 DHCP 클라이언트를 가지고 있어야 하며, 기기가 맨 처음 네트워크에 연결되면 DHCP 서버를 검색하거나 Auto IP를 사용하여 IP 주소를 확보해야 한다. 일단 기기가 네트워크에 연결되고 적절한 IP 주소가 지정

되면 검색 작업이 진행된다. 검색 작업은 SSDP(Simple Service Discovery Protocol)가 처리한다. 기기가 네트워크에 추가되면 SSDP는 이 기기가 제공하는 서비스를 네트워크 상에 있는 제어 포인트에 알리도록 한다. SSDP는 HTTPU(검색요청의 응답시) 및 HTTPMU(검색요청시) 기반 위에 구축되며, 제어 포인트가 네트워크 상에서 원하는 리소스를 검색하는 방법 및 기기들이 네트워크 상에서 자신들이 가용상태에 있음을 알리는 방법을 정의한 프로토콜이다.

제어 포인트가 기기를 검색한 후 검색된 기기 및 기기의 기능에 대한 정보를 자세하게 파악하기 위해 기기가 제공하는 URL로부터 기기 설명 내용을 확인해야 한다. 기기들은 다른 논리적 기기 및 서비스를 포함하고 있을 수 있으며, 기기에 대한 UPnP 설명은 XML로 표현되어 있으며, 공급 업체 고유의 제조 정보(모델명, 일련번호, 제조업체 이름, 제조업체 URL 등)를 포함하고 있다. 이 설명은 또한 제어, 이벤트 및 프리젠테이션을 위한 URL 뿐만 아니라 많은 내장된 기기 및 서비스에 관한 목록도 포함하고 있어, 제어 포인트가 기기를 제어할 때 이러한 정보를 이용한다.

GENA(Generic Event Notification Architecture)는 TCP/IP를 통한 HTTP 및 멀티캐스트 UDP를 사용하여 통보(notifications)를 송수신하는 기능을 제공하기 위하여 정의되었다. 또한 GENA는 이벤트 실행을 위하여 가입자 및 통보 발행자의 개념을 정의한다. UPnP는 GENA 포맷을 사용하여 존재 발표 내용을 생성한 후에 SSDP 프로토콜을 통하여 전송하고 UPnP 이벤트 작업의 서비스 상태 변화를 신호로 알려주는 기능을 한다. 이벤트 통보를 수신 받고자 하는 제어 포인트는 원하는 서비스, 이벤트 송신 위치 및 이벤트 통보 구독 시간을 포함하는 요청을 송신함으로써 이벤트 소스에 가입하게 된다. 가입 내용은 주기적으로 갱신되어 지속적으로 통보되며, GENA를 사용하여 가입을 취소하는 것도 가능하다.

SOAP(Simple Object Access Protocol)는 XML 및 HTTP의 용법을 정의하여 원격 프로시저 호출을 실행한다. 이것은 인터넷을 통한 RPC 기반 통신의 표준이 되어가는 추세이며, 기존의 인터넷 인프라를 통하여 방화벽 및 프록시에서 아주 효율적인 기능을 수행한다. SOAP는 또한 보안용으로 SSL(Secure Sockets Layer)을 활용하고 HTTP의 연결 관리 기능을 활용함으로써 인터넷을 통한 분산형 통신을 웹 페이지에 접속하는 것 만큼이나 쉽게 만든다. 원격 프로시저 호출과 유사하게, UPnP는 SOAP를 사용하여 제어 메시지를 기기들로 전송하며 그 결과 및 오류 내역을 제어 포인트로 반환한다. 각 UPnP 제어 요청은 실행할 동작 및 일련의 매개 변수를 포함하는 SOAP 메시지이며, 그 응답도 또한 SOAP 메시지이며 상태, 반환 값(return value) 및 모든 반환 매개 변수를 포함하고 있다.

기기가 프리젠테이션용 URL을 가지고 있다면 제어 포인트는 브라우저를 사용하여 그 URL을 표시하여 해당 기기를 제어하거나 상태를 볼 수 있게 한다.

UPnP는 IP 기반하에서 동작하는 프로토콜이므로 IP를 지원하는 홈 네트워크이면 모두 수용 가능하며 유연성이 많은 미들웨어 구조이다. 그러나, IP 기반을 지원하지 않는 홈 네트워크인 경우에는 UPnP를 올릴 수 없는 제약이 존재한다. 전송되는 데이터 양이 많기 때문에 PLC와 같은 저속인 네트워크에서 사용할 경우에는 시간이 많이 소요되어 접속이 끊어질 수 있는 문제도 생길 수 있다. 또한 네트워크 종류가 다른 경우에는 모두 응용 레벨의 소프트웨어에 의해 변환이 되기 때문에 브리지를 통과하면서 지연 시간이 많이 생기며 데이터 투명성에 문제가 생길 수 있다.

2.4. Jini

1998년 Sun Microsystems에서 발표한 분산 환경의 홈 네트워크 자원 공유 플랫폼으로서 현재 도시바, 소니, 미츠비시, 샤프 등을 포함하는 회원사들이 기술을 개발 추진이며 HAVi와 상호 연동할 수 있는 기술 개발하고 있다. 기본적으로 JVM이 올라가 있는 기기에서 모두 사용이 가능하므로 프로그램의 이식성이 매우 좋다.

Jini는 그림 8과 같은 구조를 가지고 있으며, 각각의 소프트웨어 구성 요소들은 자바 가상기계 상에서 실행되고 서비스된다. Jini 기기들은 상호 운용성을 가능하게 해주기 위해서 각각 Jini 기술을 탑재하고 있어야 하며 각 장비들은 서비스 주체에게 자신의 능력이나 기능에 대한 자세한 정보를 전송한다. Jini 장비는 Jini Service 컴포넌트에 메시지를 보냄으로써 다른 기기를 사용할 수 있다. 사용자는 새로운 Jini 기기를 가지고 Jini 네트워크를 확장시킬 수 있다.

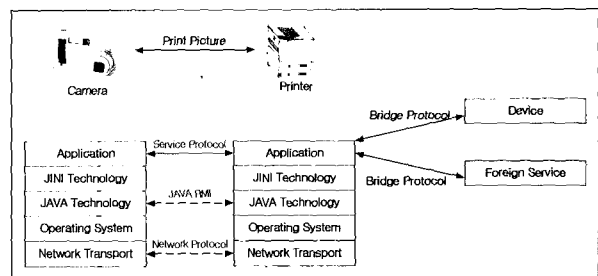


그림 8. Jini의 구조

UPnP와 비슷하게 Jini가 장착된 가전기기가 홈 네트워크에 연결될 때, lookup 서비스를 찾기 위해 discovery 절차를 먼저 수행하고, 찾았다면 lookup 서비스에 해당 서비스를 등록한다. 정보 가전기기는 먼저 lookup 서비스를 찾고, lookup 서비스를 찾았다면 정보 가전기기가 가지고 있던 모든 서비스 인터페이스를 구현한 객체를 업로드한다(join). Discovery는 Jini 서비스나 Jini 클라이언트가 Jini 룩업 서비스를 수행하는 과정이다. Jini 서비스나 Jini

클라이언트는 Jini가 가능한 기기가 될 수도 있으며, 소프트웨어로만 동작하는 Jini 프로그램일 수도 있다. 룩업 서비스는 Jini 클라이언트(사용자 포함)가 Jini 룩업서비스를 이용하여 자신이 필요로 하는 Jini 서비스를 찾고자 할 때 발생한다. Jini 클라이언트는 자신이 찾고자 하는 인터페이스를 이용하여 Jini 룩업서비스에게 요청한다. Join 서비스는 Jini 서비스가 지니 룩업 서비스에 등록되어 지니 공동체에 등록되는 과정으로서 Jini 서비스는 Jini 룩업 서비스에 자신의 프록시를 등록함으로써 Jini 클라이언트가 Jini 룩업서비스에 자신이 찾고자하는 Jini 서비스를 요청할 경우 이용된다. 이와 관련 동작이 그림 9와 그림 10에 있다.

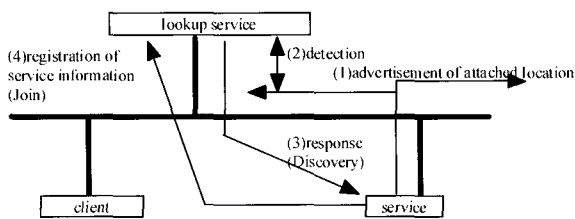


그림 9. 서비스 등록

Java VM을 사용하기 때문에 운영 체제와 하드웨어의 종류에 구애받지 않고 구현된 응용 프로그램을 사용할 수 있다. Jini는 UPnP와 같이 IP 기반 네트워크하에서 동작 되도록 되어 있고 Java VM상에서 동작하여야 하기 때문에 제약사항이 있다. 사실 Jini는 Java VM상에서 동작되기 때문에 물리층의 전송방식에 의존하지는 않지만 사양은 IP(internet protocol)를 전제로 기술된 점이 많다.

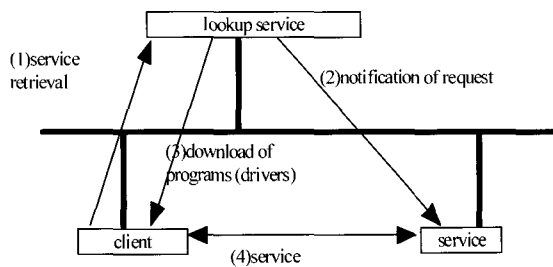


그림 10. 클라이언트의 서비스 사용

2.5. ECHONET(Energy Conservation and Homecare Network)

일본에서 Hitachi, Matsushita, Mitsubishi, Sharp, Toshiba 등의 가전업체가 주도하고 일본정부가 지원하는 백색 가전용 네트워크 규격으로서 PLC, Low power RF, HBS, IrDA, LonTalk, 블루투스, 이더넷을 지원한다. 특히 블루투스를 사용할 때는 블루투스의 PAN(Personal Area Profile)을 사용함으로써 IP 기반으로 동작할 수 있는 방법을 제공하고 있다.

ECHONET의 네트워크 구성 모델의 예시가 그림 11에 표시되어 있다. 그림 11에서 보듯이 집 밖의 인터넷 등과는 ECHONET gateway를 사용하여 연결하고 집내의 다른 프로토콜과는 ECHONET router를 사용한다. 특히 ECHONET router는 같은 프로토콜을 사용하는 경우에서도 사용할 수 있는데 이 경우는 각 네트워크를 서버네트로 구분하기 위해 사용된다. ECHONET 기기(device)가 실제로 가전기에 장착되어 동작한다. 각 노드는 1 바이트의 서브넷 식별자와 1바이트의 노드 식별자를 사용하여 ECHONET 주소를 사용한다.

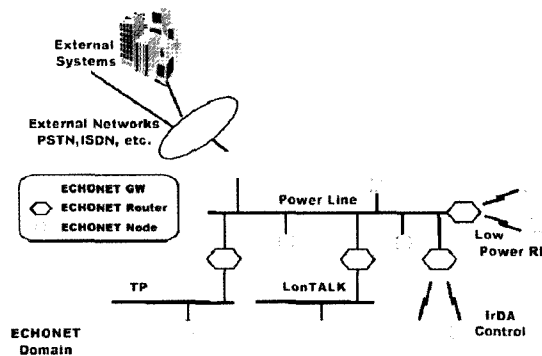


그림 11. ECHONET 네트워크 구성 모델 예

ECHONET 통신 구조가 그림 12에 있다. ECHONET 통신 구조는 크게 응용 소프트웨어, ECHONET 통신 미들웨어와 저 계층(Lower-Layer) 통신 소프트웨어의 3부분으로 구성되어 있다. 응용 소프트웨어는 시스템에 연결된 기기의 원격 제어기능을 제공하는 소프트웨어와 에어컨이나 냉장고와 같은 개별 기기의 기능을 수행하는 소프트웨어로 나눌 수 있다. 응용 소프트웨어는 Basic API와 Service API를 사용하여 만든다. 전자는 ECHONET 통신을 관리하고 데이터의 송수신을 하기 위해 사용된다. Service API는 서비스 미들웨어 기능을 동작시키기 위해 사용된다. 참고로 서비스 미들웨어는 응용프로그램을 위해 공유되는 기본적인 프로세스들을 정의하는 소프트웨어로서 이러한 기능들을 사용할 수 있도록 API들을 응용프로그램에 제공한다.

ECHONET 통신 미들웨어는 응용 소프트웨어와 하위 계층 통신 소프트웨어 사이에 위치하여 ECHONET 프로토콜에 따른 통신기능을 수행한다. ECHONET 통신 미들웨어는 ECHONET 통신 부분과 프로토콜 차이 흡수부(Absorption of Protocol Difference)로 나눌 수 있다. 전자는 응용 소프트웨어가 가전기기를 원격제어하거나 모니터링할 때 필요한 통신 프로토콜을 처리하는데 사용된다. 다시 말하자면, 기기 객체(device object)와 같은 객체들을 액세스할 때 통신 부분에서 처리해야 되는 부분을 수행한다.

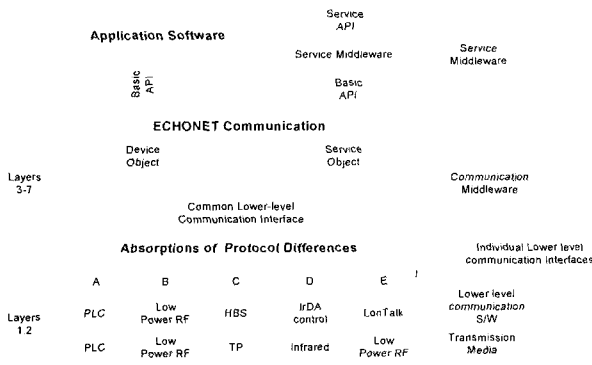


그림 12. ECHONET 통신 구조

프로토콜 차이 흡수부는 다양한 저 계층 통신 프로토콜로 구성된 네트워크들을 통합하고 그들을 사용할 때 하나의 네트워크처럼 보이게 한다. 이 부분에서 주소 변환, 프로토콜 변환, 메시지 분할 및 조합들을 수행한다. 참고로 기기 객체는 센서, 에어컨과 냉장고와 같은 가전기기를 표현하는 혹은 저장되는 데이터들을 논리적 모델이다.

하위계층 통신 소프트웨어는 각 전송 매체에 대응되는 통신 프로토콜, 즉 PLC, RF, HBS 등의 프로토콜을 수행한다.

ECHONET은 앞에서 언급한 미들웨어와는 달리 MAC 및 물리 계층을 구분하여 이들을 통합하는 방법을 제공하고 있다. 그러나 저 계층 네트워크들을 통합하는 통신 미들웨어를 강조하여 응용 프로그램의 이식성이 뛰어나지 못하다. 또한 IP 계층과의 연결은 블루투스만 현재 지원되는 단점이 있다. peer-to-peer 통신이 가능하다.

3. 미들웨어의 성질

2장에서 살펴본 미들웨어를 비교한 내용이 표 1에 표시되어 있다. 이러한 내용을 살펴보면 다양한 홈 네트워크들과 응용 프로그램들에 공통적으로 사용할 수 있는 미들웨어의 성질을 추정할 수 있다.

미들웨어가 가져야 하는 성질을 정리하면 다음과 같다.

- 다양한 홈 네트워크와 연동
- 비 표준 장비와 연동
- P2P 통신이 지원
- PnP 지원
- 이종 네트워크간의 데이터들의 라우팅
- 플랫폼은 자바 혹은 SDK 제공
- 개발 언어는 자바 혹은 C/C++
- addressing, eventing, control, monitor, self-description 기능 제공
- 인터넷과의 연동하는 게이트웨이 기능 제공
- legacy 미들웨어와 연동

사용자가 가전기기를 홈 네트워크에 쉽게 연결하여 사용하기 위해서는 PnP 지원이 필수적이며, 입의의 홈 네트워크가 장착된 가전기기를 자신의 홈 네트워크에 연결하기 위해서는 여러 형태의 홈 네트워크와 연동이 되어야 하며 동시에 P2P 통신이 가능해야 한다. 브리지 혹은 게이트웨이를 사용하면 비용이 증가할 뿐만 아니라 고장의 확률도 많아지게 되므로 이들 기기들의 사용은 가정에서 많이 지양하여야 할 것으로 보인다. 또한 P2P 통신이 되기 위해서는 이종 네트워크간의 데이터 교환이 필수적이다. 그러나 이러한 데이터 교환 방법은 UPnP에서 사용하는 방식과 ECHONET에서 사용하는 방식이 있다. 전자는 응용 계층에서 데이터를 교환하기 때문에 투명성에서 문제가 생긴다. UPnP 등에서 사용하는 IP 기반 네트워크에서는 실제로 IP 계층이 라우팅 기능을 제공한다. 그러나 라우팅 기능을 넣기 위해서는 프로그램의 크기가 매우 늘어나게 되므로 응용이 매우 제한이 된다. 후자는 통신 미들웨어 부분에서 데이터를 다른 네트워크로 전환시켜 준다. 이를 위해 새로운 가전기기의 식별 번호를 도입하여 이를 기반으로 라우팅 기능을 수행한다. 일반적인 라우팅 기능이 후자의 방식이다. 그러나 이 경우도 IP 기반 네트워크와 연동하기 위해서는 새로운 방식이 필요하다. 따라서 홈 네트워크에서 흐르는 데이터들의 라우팅을 위하여 새로운 방식이 필요하다. 또한 인터넷과 연동하는 방식도 필요하다. 인터넷과 연동하기 위해서는 게이트웨이 이를 통해서 가능하기 때문에 여기서는 고려하지 않는다.

사용자가 사용하는 서비스는 진화하기 때문에 사용자가 필요할 경우 서비스 프로그램을 다운로드 받아 사용할 수 있어야 한다. 이를 위해 플랫폼은 자바 기반만으로 하는 것이 좋을 것으로 생각된다. 그러나 가전기기의 크기와 비용에 따라서는 사용하는 제어기의 크기와 비용이 제한이 되므로 가능한 C/C++ 기반의 SDK가 제공되는 것이 필요하다. 지금까지 미들웨어가 홈 네트워크와 연동하기 위한 성질과 응용 프로그램들과 연동하기 위한 성질들을 살펴보았다. 지금부터는 미들웨어가 직접 가져야 하는 기능을 살펴보겠다. 미들웨어는 자신의 가전기기에 대한 주소와 성질들을 홈 네트워크에 연결되어 있는 가전기기에 알려 필요할 경우 가전기기를 제어할 수 있게 하도록 해야 한다. 이러한 기능이 어드레싱과 self-description 기능이다. 또한 가전기기의 이상 동작 등을 할 시 이벤트를 통지하게 하는 기능을 설정해야 하며, 다른 기기에서 제어를 하거나 다른 기기를 제어할 수 있게 해야 한다. 또한 다른 기기의 상태 등을 모니터링 할 수 있어야 한다.

4. 결론

본 고에서는 현재 사용하고 있는 홈 네트워크용 미들웨어 중 HAVi, OSCi, UPnP, Jini, ECHONET에 대한 동작과 특징

표 1. 홈 네트워크용 미들웨어 비교

	HAVi	OSGi	UPnP	Jini	ECHONET
회원사	Sony, Philips, Hitachi, Grunding AG, Matsushita, Sharp 등 15개 사	Ericsson, Nokia, Cisco, Siemens, Philips, Oracle, Motorola 등 44개 사	MS, IBM, Mitsubishi, Philips, HP, Siemens, Sony 등 524 개사	Sun Microsystems	Hitachi, Matsushita, Mitsubishi, Sharp, Toshiba 등 107개사
지원 네트워크	IEEE1394	GW를 통해 다양한 네트워크와 연동	IP기반 네트워크	IP기반 네트워크	PLC, Low power RF, HBS, IrDA, LonTalk 지원
플랫폼	Java 기반	JES 기반	UPnP SDK 기반	Java 기반	-
개발 언어	Java	Java	C,C++, JAVA	Java	-
타겟 서비스	홈 오디오/비디오	모든분야	모든분야	모든분야	백색가전
P2P 통신	O	X	O	O	O
PnP 지원	O	O	O	O	O
IP와 연동	X	O	O	O	일부
라우팅	X	X	X	X	O
비표준 장비 연동	브리지를 통해 연동	GW를 통해 연동	브리지를 통해 연동	GW를 통해 연동	GW를 통해 연동

을 살펴보고 이들 비교하였다. 이러한 미들웨어를 비교함으로써 홈 네트워크용 미들웨어가 가져야 하는 성질들을 정리하였다. 특히 데이터들이 이더넷상의 네트워크의 기기로부터 IEEE 1394의네트워크를 거쳐 PLC 네트워크에 있는 기기를 제어할 수 있어야 하므로 데이터의 라우팅 기능이 중요하다. 물론 이더넷, IEEE 1394 및 PLC에 IP 계층과 UPnP를 구현하면 되지만 이러한 프로토콜을 구현 못하는 가전기기가 있을 수 있기 때문에 새로운 미들웨어가 필요하다. 홈 네트워크 종류가 다양하고 앞으로도 새로운 홈 네트워크가 나타날 수 있기 때문에 이를 수용할 수 있는 구조가 되어야 한다. 여러 legacy 미들웨어와도 연동할 수 있는 구조가 되어야 한다.

참고문헌

1. HAVi v1.1 Specification, <http://www.havi.org>
2. OSGi Alliance, <http://www.osgi.org>
3. UPnP Forum, <http://www.upnp.org>
4. Jini Specification 2.0, <http://java.sun.com/Jini/specs/index.html>
5. ECHONET Consortium, ECHONET Specification Ver2.11, <http://www.echonet.gr.jp>
6. 박현, Home Network 기술동향과 표준화방향, TTA 저널, 제84호, pp. 28-36, 2002.

저 자 소 개



《박홍성(朴洪聖)》

- 1961년 3월 16일생.
- 1983년 서울대 제어계측공학과 졸업.
- 동 대학원 석사(1986).
- 동 대학원 박사(1992).
- 1992년~현재 강원대 전기전자정보

통신공학부 정교수



《김형욱(金炯喆)》

- 1973년 7월 16일생.
- 1999년 강원대 제어계측공학과 졸업.
- 동 대학원 석사(2001).
- 2001년~현재 강원대 제어계측공학과 박사과정