

# 웹 기반의 홈 네트워크 구현

## An Implementation of Web-based Home Network

이 경훈, 박 재형, 박 종백

삼성전자 소프트웨어 센터

*jbpark@swc.sec.samsung.co.kr*

### 요약

최근의 디지털 TV를 비롯한 디지털 가전 제품의 출현은 그간 IEEE-1394 등 네트워크 미디어 기술의 발전과 더불어 가전 기기간의 네트워크, 즉 홈 네트워크의 구성 요건을 완성 시키는 단계에 이르고 있으나, 일반 사용자에게 최선의 응용 서비스를 제공하기 위한 사용자 인터페이스 및 기기간의 제어문제도 주요 이슈로 부각되고 있다. 본 논문에서는 웹을 기반으로 하고, 기기간의 제어 및 인터넷 서비스를 홈 네트워크에까지 확대하기 위한 목적으로 구현된 홈 네트워크 구현 결과에 대해 고찰해 본다.

### 1. 서론

디지털 TV, 디지털 VCR 등의 디지털 가전기기의 등장으로 가정용 기기에도 네트워킹이 가능하게 되었으며, 종래의 컴퓨터를 중심으로 한 네트워크의 개념은 이러한 가전 기기간의 데이터 서비스로 확대 발전하게 되었다.

최근 IEEE-1394, HomePNA<sup>[1]</sup>, HomeRF<sup>[2]</sup> 등 홈 네트워크 구성을 위한 네트워크 미디어 기술이 발표되면서 홈 네트워크에 대한 관심은 계속 커지고 있으며, 이들 미디어 기술을 기반으로 홈 네트워크가 제공하게 될 응용 분야 및 서비스 안에 대한 연구 개발도 계속 되고 있다. 대표적인 예로서는 Microsoft를 중심으로 한 Universal Plug and Play, Sun의 Jini, Sony를 중심으로 한 HAVi 및 삼성을 중심으로 한 Home Wide Web (이하 HWW)을 들 수 있다.

홈 네트워크는 그 사용 방식에 있어서 사용자가 전문가가 아니라는 점에서 “plug and play” 또는 “easy to use”라고 표현되는 요구 조건은 중요하다. 이 요구 조건은 가전 분야에 만 있는 것은 아니지만 제품에 대하여 사용자가 기대하는 기대치의 차

이에서 오는 미묘한 것도 포함한다. 또 다른 요구 조건은 Quality of Service에 대한 것으로 컴퓨터 통신 분야는 전통적으로 Best effort service 중심이다. 하지만 가전 기기 특히 디지털 가전 기기는 멀티미디어 데이터 통신으로 고 화질이나 고 음질의 서비스를 저렴한 가격으로 원한다는 것이다.

인터넷 표준 역시 홈 네트워크 분야에 중요한 위치를 차지하고 있다. HWW은 인터넷 표준인 웹 표준(HTTP, HTML, ...)을 가전 기기, 특히 Digital TV와 디지털 가전 기기에 원격 제어와 사용자 인터페이스에 적용한 예이다. HWW은 사용자 인터페이스에서 프로그램 인터페이스까지 영역을 포함한다.

본 글에서는 홈 네트워크 미디어 기술을 기반으로 홈 네트워크 서비스를 제공하기 위한 Universal Plug and Play, Jini, 및 HAVi 등 미들웨어 기술들에 대해 간단히 살펴본 후 웹을 근간으로 홈네트워크를 구현한 HWW의 구현 사례에 대해 고찰해 본다.

## 2. HAVi, Jini, UPnP의 개요

### 2.1 HAVi<sup>[3]</sup>

HAVi는 "Home Audio/Video Interoperability"의 약자로 가전제품을 중심으로 하는 여러 기기들 간의 상호 연동성 (interoperability)에 중점을 두고 제안된, 시스템 소프트웨어의 구조와 기능 등을 규정한 표준화 안이다. HAVi는 IEEE Std 1394-1995 표준과 IEC-61883 표준을 지원하는 가전기기들을 기본적인 대상으로 상정하고 기술되어 있으며, HAVi를 지원하는 기기는 그림 2.1과 같은 소프트웨어 구성을 갖는다.

이들 구성 요소중 DCM (Device Control Module)이 중심적인 개념으로, HAVi Network을 구성하는 각 기기를 HAVi 시스템 상에 소프트웨어 객체의 형태로 대표한다. HAVi 내의 소프트웨어 요소들은 사용할 기기의 DCM을 통해 해당 기기에 접근한다. 각 기기 별로 독립된 기능을 갖는 부분을 포함한 경우 FCM (Function Control Module)이 DCM보다 세분화된 제어를 지원할 수 있다. 한편, DCM의 관리를 위해, DCM Manager가 필요하며, DCM의 기능을 이용해 사용자가 필요로 하는 부가적인 기능을 구현한 Application Module이 있다. DCM은 DCM Code Unit이라는 형태로 저장장치에 저장되며, 각 기기에는 다른 기기와의 연계 동작을 위해, 필요한 정보를 보관, 제공하는 Registry와 Event 전달을 위한 Event Manager, Resource 관리를 위한 Resource Manager, Stream Data 관리를 위한 Stream Manager 등이 포함된다.

이와 함께, 실질적으로 HAVi 내의 시스템 요소들이 Remote / Local 구분 없이 다른 시스템 요소들과 통신하도록 도와주는 Messaging System과 Network Media에 의존적인 부분을 처리하는 Transport Adaptation Message와 Communication Media Manager가 포함된다.

HAVi 기기에는 자바(Java) 기술이 포함될 수 있는데, 자바로 DCM/FCM 및 응용 프로그램을 구현할 수 있다. 이때, 자바로 구현된 응용 프로그램은 Havlet이라 하고, 자바 실행환경 (Java Runtime Environment) 및 다른 시스템 요소를 모두 포함한 기기를 FAV (Full AV device), 자바 실행환경을 제외한 다른 시스템 요소를 모두 포함한 기기를 IAV (Intermediate AV device), 자바 코드(Java Code)로 구현된 DCM을 갖지만, 다른 시스템 요소가 없는 기기를 BAV (Base AV device),

그 밖의 HAVi를 지원하지 않는 기기를 LAV (Legacy AV device)라 하여 구분하고 있다. BAV는 내포하고 있는 DCM 자바코드를 네트워크상의 FAV로 전송하여 FAV 상에서 DCM을 동작하도록 한다. 한편, LAV는 네트워크 상의 FAV나 IAV에서 대응되는 DCM을 내장하였다가 실행시키면 이에 연동하여 동작하게 된다.

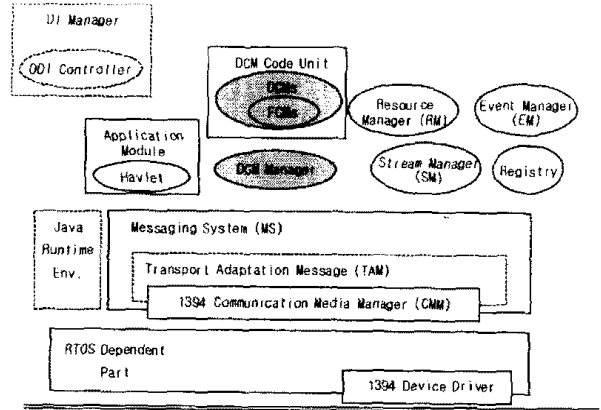


그림 2.1 HAVi FAV (Full AV device)의 구성

HAVi의 최근 버전인 1.0Beta는 아직 Security나 주요 제품의 소프트웨어 지원 범위와 같은 주요사항과, MS의 API, 초기화 과정의 정의 등 아직 보완해야 하는 사항들을 많이 내포하고 있는 상태이다.

### 2.2 Jini<sup>[4][5]</sup>

Jini는 자바 기술을 확장하고자 하는 Sun사의 R&D 프로젝트에서 시작되었다. 이 프로젝트의 기본 목적은 네트워크 접속과 이용과정을 단순화시키는 것이었는데, 이를 위해 네트워크를 통해 연결된 자바 가상 기계(Java Virtual Machine)들의 연합을 생성하기 위한 기술을 연구하게 되었다. 이 연구의 결과로 개발된 것이 Jini 기술이다.

Jini의 핵심 개념으로 다음을 포함한다.

- Services : 사람이나 프로그램 혹은 다른 Service에 의해 사용가능한 entity를 의미
- Lookup Services : 필요한 Service의 일부 정보를 갖고 정보의 Reference를 얻도록 해주는 Service
- Java RMI (Remote Method Invocation) : Jini 기반기술로, Jini의 service들간의 통신을 담당

- **Leasing** : 특정 entity에 대한 접근, 사용 가능한 시간을 조절하여 자원을 관리
- **Transaction** : 일련의 연속된 연산들을 묶어 전체 결과에 따라 각 연산의 성공여부 처리
- **Events** : Java 의 Event 개념을 분산환경까지 확장, 다양한 구조의 분산시스템 구성 지원

이런 개념들을 구현하기 위해 Jini는 Services, Infrastructure, Programming Model 라는 세 가지 요소로 구성된다.

Jini Service 는 Service 를 제공하는 측이 Service 를 이용하기 위한 Proxy Code 를 Lookup Server 에 등록하고, 등록된 Proxy Code 를 Service Client 가 얻어 수행시키면서 Service 를 이용하는 형태로 동작한다.

이를 지원하기 위해 몇 가지 요소를 포함하는 Infrastructure 가 있어야 하는데, 여기에는 Service 의 등록 및 관리를 위한 Lookup Service, Jini 시스템 요소간의 기본 통신 기법으로 사용되는 Extended Java RMI, Service 연합내의 다른 member 들과 기능제공/이용을 위한 Discovery Protocol, Java 기반의 보안요소를 분산환경으로 확대한 Distributed Security System 이 포함된다.

Programming Model 은 분산 시스템을 구성하는 각 Service 들을 구현하는데 필요한 기능을 제공하는 방법으로, Resource 의 할당/해제 방식을 정의하는 Leasing Interface, Jini 소프트웨어 서비스간의 Event 통신을 위해 JavaBeans 의 Event 모델을 확대한 Event & Notification Interface, 및 일련의 operation 을 묶어 이들 전체의 동작 결과를 실행/비실행의 상태로 유지하도록 하는 Two-phase Commit (Transaction) Interface 로 구성된다.

이들 세 가지 요소를 통해 형성된 연합은 전체적으로 하나의 분산 시스템으로 통합되며, 기본 구조가 단순하기 때문에 관리가 쉬우며 가변적으로 운용된다. 이를 Home Network 에 응용하는 경우, Plug-and-Play, 다양한 환경에 대한 적응성, IP Network 에 대한 확장성, Java 연관제품, 시스템과의 호환성 확보 등의 장점이 있으나, JVM 을 실행시키기 위한 연산능력과 저장능력을 갖는 시스템 요구사항, 전반적인 수행속도 저하, 외부 Network 연결 시 보안문제 등의 문제점들도 내포하고 있다.

### 2.3 UPnP<sup>[6][7]</sup>

UPnP 는 Universal Plug-and-Play 의 약자로,

PC 기반의 Plug-and-Play 개념과 기술을 Network 기반으로 확장한 것이다. 이를 위해 UPnP 는 Network 에 연결된 기기나 Service 들도 찾을 수 있도록 탐색 및 점검 방법을 확장하였으며, PC 가 없더라도 다른 기기와 일대일 통신을 위한 탐색, 수행이 가능하도록 하였고, 통신 프로토콜을 비롯한 각 기기의 고유 특성을 탐색, 점검할 수 있도록 하였다. 이와 함께, TCP/IP, HTML, XML, HTTP, DNS, LDAP 등과 같은 기존의 중요 표준들을 지원할 수 있으며, 기존의 네트워크 미디어 외에도 PLC, RF, cable 등의 새로운 표준들을 지원한다.

UPnP 는 크게 2 가지 형태의 네트워크 환경을 고려하여 설계되었는데, 하나는 주요 정보를 제공할 수 있는 서버를 포함하는 환경이고, 다른 하나는 특정 서버없는 기타 일대일 연결 위주의 환경이다. 서버를 포함한 환경에서는 DHCP 를 이용해 각 기기들의 IP 주소를 할당하고, DNS 를 통해 Naming 서비스를 제공하며, Directory Service 를 이용해 탐색을 지원한다. 이에 반해, 일대일 연결 위주의 환경에서는 AutoNet 방법을 통해 임시로 IP 주소를 할당하고, Multicast DNS 를 이용해 Naming 서비스를 구현하며, Simple Discovery 방법을 통해 탐색을 지원한다.

UPnP 에서는 Service 를 제공하는 측에서 제공 가능한 기능과 특징들을 XML 로 기술하여 Client 측에서 이를 이용해 필요한 기능을 선택, 활용하는 형태로 동작하는데, 각 기기 별로 갖는 특징을 일관성있게 기술할 수 있는 별도의 표준화 작업이 필요한 상황이다.

### 3. Home Wide Web <sup>[8][9]</sup>

HWW 은 인터넷 응용 구조를 홈 네트워크 까지 확장한 것이다. 홈 네트워크는 디지털 가전기기를 연결하거나 홈 오토메이션 기기를 연결하는 네트워크로 일반적인 컴퓨터 네트워크와 다른 요구조건을 가지고 이를 해결하는 구조이다. 그림 3.1 에 보인 바와 같이 HWW 의 기본구조는 controller device 가 web browser 를 통해 web server 가 설치된 controlled device 를 제어하기 위한 구조로 되어 있다.

사용자 인터페이스로 웹의 이용은 여러 가지 장점이 있다. 웹이 사용자에게 친숙하며 쉽게 배우고 사용할 수 있다. 웹은 클라이언트 서버 모델이므로 제어 대상이 되는 장치는 웹서버를 통하여 자신의 제어 요소를 사용자에게 전달한다. 사용자는 브라우저를 통하여 이 제어요소를 제어한

나. 물론 이런 제어 방법은 프로그램들 간의 통신으로도 확장할 수 있다.

그림 3.2는 웹 브라우저로 구현한 홈네트워크 관리자 응용 그림을 보여 주고 있다 (삼성 Digital DTV에 내장된 웹 브라우저에서 IEEE 1394 네트워크에 연결된 디지털 기기를 나열한 그림으로 HWW의 주요 사용자 인터페이스이다). 그림 3.3은 Digital TV에 웹 서버로부터 Digital TV를 제어하기 위한 장치 베이스 화면으로 Digital TV는 자체내에 웹 서버도 포함하고 있다. 그림 3.4는 Digital TV에서 Digital VCR을 제어하기 위한 브라우저 화면을 보이고 있다.

HWW는 가정 내의 디지털 기기를 제어하는 목적에서 출발하였지만 궁극적 목적은 디지털 기기들의 분산 환경을 제공하는 것이다. 개방형 분산 환경에서 표준화의 위치는 대단히 중요하다. 특히 새로운 시장으로 홈 네트워크는 여러 영역의 기술들이 수렴하거나 경쟁하는 성격이 강하여 표준화 경쟁 역시 치열하다. 분산 환경에 대한 연구는 컴퓨터 업계에서 많이 되어 왔지만 가정이라는 특별한 환경이 가지는 요구조건 때문에 가전 업계가 새로운 표준을 제시하는 것이다. 기존의 컴퓨터가 사용된 사무 환경과 다른 요구 조건은 경제적인 면과 사용 방식에 관한 것 등 다양하다.

HWW는 하부 구조는 Protocol들과 표현 계층(6계층)의 것들과 다양한 플랫폼에 적용 가능한 API에 대한 것이다.

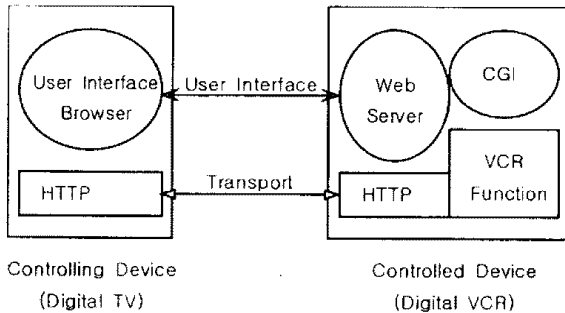


그림 3.1 HWW의 구성 예

HWW의 구조는 웹을 기반으로 하는 응용 구조 (Web-based Application infra-structure)와 Plug-and-Play를 위한 네트워크 구조로 구성되어 있다.

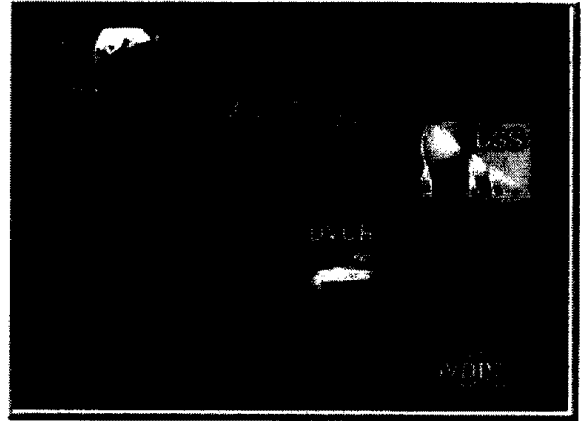


그림 3.2 웹 브라우저로 구현한 홈 네트워크 관리자 응용 프로그램

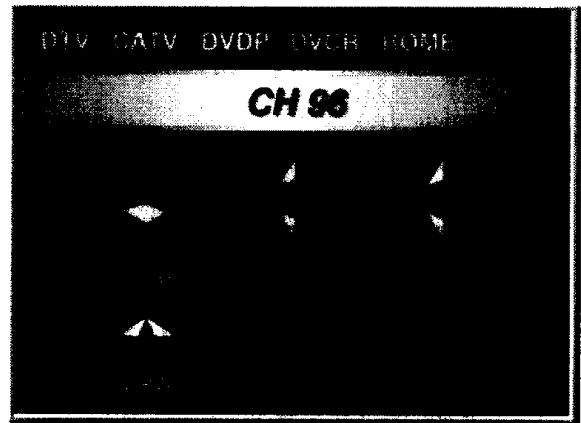


그림 3.3 Digital TV의 제어 화면



그림 3.4 Digital VCR 제어 화면

### 3.1 웹을 기반으로 하는 응용 구조 (Web-based Application infrastructure)

일정 정도의 메모리와 계산능력이 있는 모든 기기들은 웹 서버를 가지고 있고 자신을 제어할 웹 페이지를 가지고 있어서 브라우저를 통하여 사용자가 그 기기를 제어할 수 있다는 것이 HWW의 기본 전제이며, 이런 관점에서 HWW는 사용자 인터페이스(User Interface)에 대한 표준이다. 이 표준의 상호 연동성은 웹 페이지에서 사용하는 태그들을 규정하는 정도에서 해결된다. 물론 기본적으로 HTTP와 같은 표준 전송 프로토콜을 사용하는 규정도 필요하다. 유저 인터페이스는 웹을 근간으로 하는 방식이 가장 효과적인 대안이다.

한편, 기기들간의 프로그램 인터페이스는 다른 차원의 표준화가 필요하다. Remote Procedure Call (RPC)과 같은 전송과 데이터 표현에 대한 표준이 필요하다. HAV나 Jini나 HomeAPI등에서도 방안을 제시하고 있으나, 각각의 방안을 수용하려면 제약점이 있다. 예를 들면 Jini는 Java의 프로그램 모델만을 제공하고, HomeAPI는 MS Windows의 프로그램 환경 중심이며 HAV는 AV 네트워크 중심이다. 이런 점들 때문에 보다 통합적이고 기술 중립적인 관점을 필요로 한다. WBEM (Web Based Enterprise Management)은 이런 관점의 전형적인 예이다.

HWW은 WBEM의 구조를 홈 네트워크에 적용한 것이다. WBEM의 장점은 응용 프로그램에 개발에 있어서 일관된 관점을 제공하여 준다라는 점이다. 하위의 네트워크 관리 프로토콜이 SNMP이건 DMI이건 프로그램에 입장에서 이들 Agent가 제공하는 데이터가 단일하게 보이기 때문이다. 이렇게 단일한 데이터를 제공하는 계층을 제공하는 것이 WBEM이기도 하다. 홈 네트워크는 다양한 목적과 다양한 기술의 네트워크가 상호 연동된 형태일 것이다. 따라서 이를 통합적으로 보는 관점이 있어야 하고 이것의 중심은 웹을 근간으로 하는 제어가 될 것이다.

### 3.2 Plug-and Play를 위한 HWW 하부 구조

네트워크 기기를 표현하는 표준적인 정보 모델이 필요하며 이런 것에 대한 표준이 우선 존재야 한다. 예를들어 IEEE 1394는 고속 직렬

버스에 대한 표준이며, 컴퓨터 데이터와 같은 비동기 데이터 처리 뿐만 아니라 오디오 비디오 같은 동시성을 요하는 데이터를 처리까지 가능한 구조를 갖는다 또 버스 구조이므로 네트워크에 연결된 기기에 대한 레지스터 정보를 공유하여 "Plug-and Play"를 지원한다. 네트워크 구성 정보를 모든 노드들이 쉽게 얻는 것이다. 한편, IEEE 802 계열의 근거리 네트워크 표준들은 네트워크 구성에 대한 정보 얻는 방법을 따로 정의하지 않는다. 상위의 응용 계층에서 해결하여야 한다. 또, SNMP (Simple Network Management Protocol)는 네트워크 관리를 위한 표준으로 응용 계층에서 네트워크 구성에 대한 정보를 얻는 방법 중에 하나이다. SNMP는 인터넷과 같은 대규모의 네트워크를 관리하기 위한 것으로 홈 네트워크와 같은 작은 규모나 CPU/메모리의 용량에 제한이 있는 기기에는 부적합하다. 뿐만 아니라 기본적으로 전문가의 도구로서는 적합하나 일반적인 사용자가 사용하기에는 사전 지식을 요구하는 기술들이다. 따라서, 자동적으로 서비스를 찾는 프로토콜이 필요하며, 필요에 따라 사람의 개입이 없이 작동하는 네트워크 홈 네트워크의 요구 조건인 것이다.

이것은 네트워크에 기기가 접속하면 이 기기가 추상화 된 형태로 정보 모델을 제공하여 이것을 사용하는 응용이나 사용자에게 알려져야 된다. 또 필요에 따라서 네트워크의 서비스를 이용하는데 서비스를 찾는 방법은 중요하다. 네트워크에 어떤 서비스가 존재하는지 찾는 것은 사람에게나 응용 소프트웨어 모두에게 중요하다. 네트워크의 구성과 서비스에 대한 정보는 사람 뿐만 아니라 응용 프로그램에게 매우 중요하다. 서비스를 구분하는 구분자와 이 서비스가 위치하는 장치의 주소 즉 위치 정보와 서비스에 어떻게 접근할 것인지에 대한 사용 방법에 대한 정보에 표현이나 전달 방법 등이 필요하다. 네트워크의 구조는 이런 서비스를 찾는 서비스 즉 서비스에 대한 서비스를 위한 시스템의 구조이다. 이런 구조의 중심에서는 Service Discovery Protocol (SDP) 이 필요하다.

Simple Service Discovery Protocol (SSDP)은 MS가 제안한 표준의 초안이다. 이 표준은 계속 개발되고 있다. 작은 규모의 네트워크에서 관리자나 계획 없이 "Plug and Play"가 가능하도록 설계되어 있다. 이것과 AutoIP, Multicast DNS 등을 단말에 표준적으로 설치하여야 한다. 이 프로토콜은 전송 방법에

대한 표준들은 비슷하지만 전송 데이터의 표현에 대한 표준은 다양하다. 이것은 표현 계층 표준으로 응용 프로그램 입장에서 중요한 요소이다. HWW는 SSDP 상위에서 이를 수용 가능하다.

표현 계층은 전통적으로 ASN.1 (Abstract Syntax Notation one)과 같은 국제 표준과 XDR, CDR (Common Data Representation)과 같은 CORBA 표준이 있다. 최근에는 DMTF와 같은 단체에서 XML (extensible markup language)을 데이터 표현의 표준으로 많이 사용하고 있다. Microsoft가 제안한 SSDP (Simple Service Discovery Protocol)에서 장치에 관한 정보를 XML 표준을 이용한다. 이런 방식의 장점은 SNMP에서 사용하는 ASN.1 보다 풍부하게 정보를 표현할 수 있다는 점이다.

#### 4. 요약 및 HWW의 향후 계획

앞에서 언급한 것과 같이 HWW의 기본적인 개념과 접근방식에 대한 표준화 작업은 진행 중에 있으며, 실제 시스템의 구현 및 적용을 통해 지적되는 문제점들에 대한 보완작업도 진행 중이다. 현재는 주로, Device와 Device간의 제어와 정보교환이 가능하도록 사양을 보완하는 작업을 진행하고 있다.

한편, 이외에도 HWW의 확산을 위한 시도는 크게 2가지 방향이 있다. 그 중 하나는 HWW의 개념과 기능을 최대한 활용할 수 있는 응용분야의 개발이며, 다른 하나는 HAVi나 Jini 혹은 UPnP처럼 다른 Home Network 관련 시스템들과의 상호연동성을 구현하는 것이다.

HWW의 기본적인 개념을 효과적으로 상품에 적용할 수 있기 위해서는 기존의 가전 제품들을 단순히 네트워크로 묶어 놓은 것 이상의 부가적인 효과를 창출하고, 보여줄 수 있는 응용분야가 개발되어야 할 필요가 있다. HWW를 적용한 가전 제품들을 설치한 가정의 조명이나 난방 등을 사용자가 외부에서 네트워크를 통해 제어하거나, 다양한 멀티미디어의 정보를 PC가 아닌, TV를 통해 사용자가 접하도록 하는 것 등이 이런 예가 될 것이다.

세계시장의 주요 부분을 차지하는 선진 가전, 정보통신 회사들이 제안하고 진행하는 다양한 Home Network 관련 시스템들과 HWW는 상호 경쟁적인 입장이지만, 사용자의 입장을 고려할 때 이들 시스템들과 HWW는 상호 호환성이 지원되

어야 할 필요가 있다. 이에 따라, HWW에서는 이들 시스템들과 HWW를 연결하기 위한 전용 기기를 Gateway 방식으로 구현하는 방안에 대해 연구하고 있으며, 연관 회사들과 긴밀한 협력을 진행하고 있다.

#### 참고자료

- [1] HomePNA Spec., <http://www.homeapi.org>
- [2] HomeRF Working Group., <http://www.homerf.org>,
- [3] HAVi Consortium, "The HAVi Specification - Specification of the Home Audio/Video Interoperability (HAVi) Architecture" ver 1.0 Beta, <http://www.havi.org>,
- [4] Sun Microsystems Inc., "Jini Architecture Specification" Revision 1.0DC, <http://www.javasoft.com>
- [5] Sun Microsystems Inc., "Jini Device Architecture Specification" Revision 1.0DC, <http://www.javasoft.com>
- [6] Bengt Christensson, Olof Larsson, "Universal Plug and Play Connects Smart Devices", WinHEC 99 White Paper, April 1, 1999
- [7] Universal Plug and Play, <http://www.upnp.org>
- [8] "Home Network, Client/Server/Server, Control Model", Contribution to VESA Home Network Committee by Samsung Information Systems America, October 24, 1998, <http://www.vesa.org>
- [9] Samsung Home Wide Web Specification to be released