

# Machine to Machine 기반의 N Screen 서비스 기술

양승률, 권영환, 이재호, 김경호, 이현재, 김진필  
LG전자

## 요약

N Screen 서비스는 하나의 콘텐츠를 스마트폰, PC, 태블릿, 자동차 등 다양한 디지털 정보기기에서 공유할 수 있는 차세대 컴퓨팅, 네트워크 서비스를 말한다. 이는 시간, 장소, 디지털 기기에 구애 없이 언제 어디서나, 어떠한 기기에서나 하나의 콘텐츠를 이어서 볼 수 있고, 사용자는 가장 편리한, 가장 좋은 환경의 기기에서 원하는 콘텐츠를 즐길 수 있다.

본고에서는 N Screen 서비스를 위한, 기기 간의 연동 기술을 홈 네트워크의 대표적인 UPnP, DLNA 표준과 특정 업체를 중심으로한 DIAL, AllJoyn 기술 및 스마트 TV 플랫폼에서 출발한 STA (Smart TV Alliance)의 N Screen 서비스 적용 그리고, Wi-Fi 통신망을 이용한 Wi-Fi Display 표준기술을 소개하고 서로 비교 함으로써, 기기간의 N Screen 서비스 기술 및 표준화 발전 방향을 살펴보고자 한다.

## I. 서론

인터넷 기술의 발달과 서비스 확대는 스마트 폰, 스마트 TV의 보급을 급격히 넓히게 되었고, 기존의 PC 및 Tablet을 포함한 N Screen 기기간의 Content를 공유하고, 사용자의 편의에 따라 서로 다른 기기에서 Service를 즐기고자 하는 Needs가 점차 증가되어가고 있다. 콘텐츠를 즐기는 소비 형태는 크게 거실에서 TV 및 대형 스크린을 통해, 고화질의 콘텐츠를 감상하는 형태와 이동을 하면서, 모바일이나 차량, Tablet 등을 이용하여 저용량의 콘텐츠를 감상하는 형태로 나눌 수가 있으며, 이와 같이 하나의 콘텐츠를 또는 서로 다른 콘텐츠를 각각 서로 다른 기기에서 연동하여 즐길 수 있는 것이 N Screen 서비스의 가장 큰 장점이라고 하겠다.

특히, 최근 방송의 소비 형태가 젊은 층을 대상으로 방송시간에 맞추어 실시간 시청하는 소비형태에서, 선호하는 프로그램에 대하여, 주문형 VoD 로 소비하는 형태로 점차 옮겨 가고 있

고, Content의 해상도, 화질, 용량 등을 고려하여 가장 적합한 기기에서 서비스를 이용하고, 또한 하나의 콘텐츠에 대하여 이어서 서비스를 받을 수 있는 N Screen 기술은 홈 네트워크의 중요한 기술로 요구가 되고 있다.

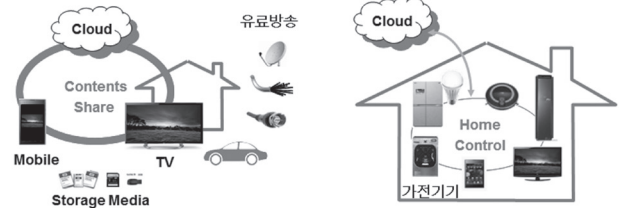


그림 1. 홈 네트워크의 콘텐츠 공유 및 기기 제어 서비스

홈 네트워크의 주요한 두 가지 기능은 <그림 1>과 같이, 기기 간의 콘텐츠의 공유와 기기간의 제어 기능이 있으며, N Screen 기능은 기기간의 콘텐츠 공유의 기술을 의미 한다. 한편, 기기간의 제어는 스마트 폰을 통하여, 맥내의 조명, 가전기기, 자동차 등을 제어한다든지, 스마트 TV를 통하여, 에어컨, 세탁기, 냉장고 및 기타 홈 가전의 상태를 점검한다든지 하는 기능들이 이에 해당된다. 본 고에서는 Machine to Machine 기반의 N 스크린 서비스 기술 홈 네트워크 안에서 상호 기기 간의 연결로 콘텐츠를 공유하는 기술에 대하여 대표적인 UPnP, DLNA, Wi-Fi Display 표준, 주요업체의 DIAL, AllJoyn 기술, 그리고 업체간의 Alliance 기술인 Smart TV Alliance에 대하여 살펴보고자 한다.

## II. 본론

### 1. UPnP (Universal Plug & Play) Forum / DLNA (Digital Living Network Alliance)

UPnP Forum은 Networking을 통한 연동을 기반으로 다양한 종류, 다양한 제조업체의 Device 간의 다양한 서비스를 제공하기 위해 90년대 후반부터 그 활동을 시작하였고, 현재는 제

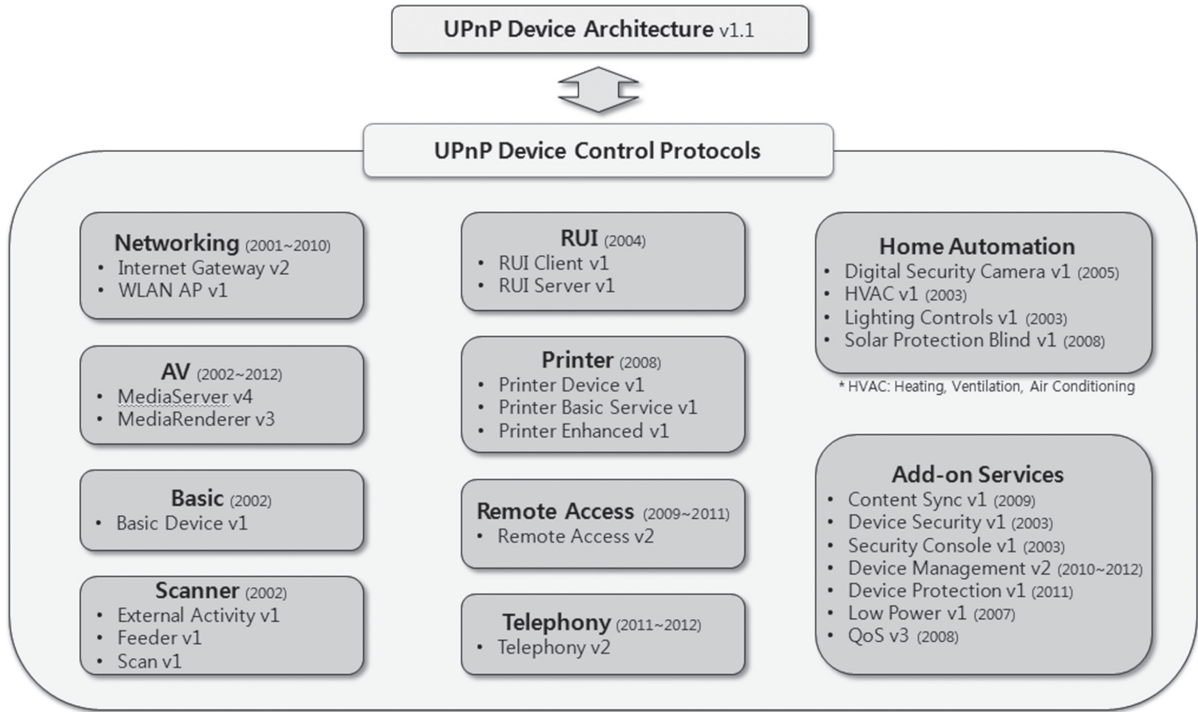


그림 2. UPnP 표준규격 제정 현황

조업체, 칩셋업체, 소프트웨어솔루션업체, 서비스공급업체 등의 IT분야의 Major 업체들이 대부분 참여하여 약 1000개 이상의 멤버가 활동 중에 있다. 본 Forum에서는 기기 간의 연동을 위한 Framework인 UPnP Device Architecture 및 세부적인 서비스 규격인 UPnP Device Control Protocol을 <그림 2>와 같이 정의하고, 해당 규격을 지원하는 기기의 테스트 및 인증프로그램을 제공하고있다.

UPnP Device Architecture 규격에서는 우선 연동 기기를 Device(제어 대상 기기)와 Control Point(제어 기기)로 분류한다. Device와 Control Point는 어느 기기에도 적용될 수 있는 기기의 역할에 따른 구분으로 이해할 수 있으며, UPnP기기 간의 Communication은 Device와 Control Point 간에만 발생할 수 있다. Device간 혹은 Control Point간의 Communication은 지원되지 않으나, 모든 기기는 중복된 역할(즉, Device이면서 Control Point)을 수행할 수 있으므로, Device/Control Point의 구분은 역할을 제약하기 위해서가 아니라, 다양한 시나리오로의 유연한 확장을 위한 가장 Simple한 논리적 구분이라고 볼 수 있다. 이러한 기기의 분류를 기반으로 UPnP Device Architecture 규격에서는 Device와 Control Point의 Addressing 및 Device와 Control Point간의 Discovery, Description, Control 및 Eventing 등의 Mechanism을 정의한다.

UPnP 기기는 우선 Address를 할당받을 필요가 있으며 이는

DHCP (Dynamic Host Configuration Protocol, RFC2131) 혹은 Auto-IP (RFC 3927)를 통해 보장된다.

IP Address가 할당된 UPnP 기기 간의 Discovery는 SSDP(Simple Service Discovery Protocol, RFC2616 Section4)로 정의되어있다. UPnP Forum에서 일반적으로 가정하고 있는 Home Network 환경에서는 얼마나 많은 개수의 기기가 Network에 연결되어있는지, Network에 연결된 기기의 Address가 무엇인지를 알 수 없다. 따라서 UPnP Device Architecture는 Multicast 방식을 통해서 불특정 다수의 기기 간의 Discovery를 지원한다. 또한 <그림 3>과 같이 Home Network환경에서는 기기가 수시로 Connected/Disconnected 될 수 있으므로, UPnP Device Architecture는 Advertisement와 Search의 두 가지 Mechanism으로 Discovery를 정의하여 불필요하게 반복적인 메시지 교환을 하지 않고도 기기간의 Discovery를 가능하도록 한다.

Discovery 과정 이후 Control Point는 Device의 Device



그림 3. UPnP Discovery Mechanism: Advertisement & Search

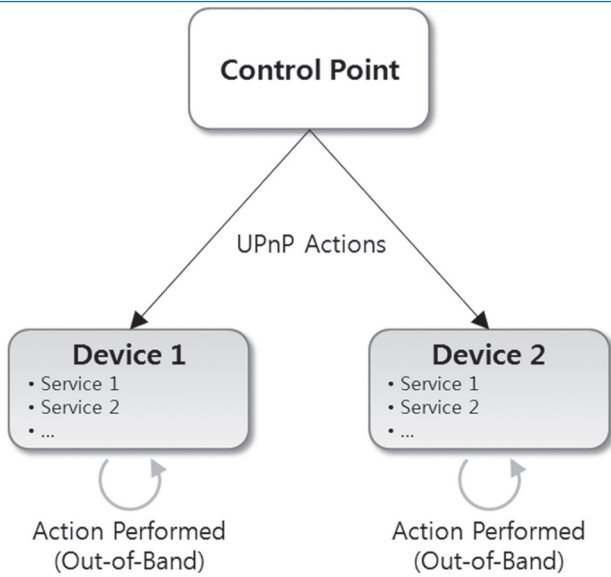


그림 4. General UPnP Interaction Model

Description 및 Service Description 문서를 수집할 수 있고 이를 통해 특정 Device가 어떤 역할을 하는 기기이고 어떤 서비스를 지원하는지 파악하여 원하는 서비스를 요청할 수 있다.

Description 정보의 수집 이후 Control Point를 통한 Device의 제어 및 상태확인인 SOAP(Simple Object Access Protocol)

을 통해 이루어진다. <그림 4>와 같이 Control Point는 XML로 표현된 SOAP Action을 Device에게 전달하여 Device를 제어하기도하고, 그 응답 메시지로 상태를 확인할 수 있다.

Device의 상태를 조금 더 효율적으로 확인하기 위해서 UPnP Device Architecture는 GENA (General Event Notification Architecture)를 채용하여, Device의 상태가 변경되면 Control Point의 SOAP Action없이도 Device가 Control Point에게 능동적으로 상태 변경에 대한 정보를 알려줄 수 있도록 <그림 5>와 같이 정의하고 있다.

서비스의 종류에 따라 세부적이고 부가적인 규격이 필요하고 따라서 각각의 서비스와 서비스 제공을 위한 Action 및 Variable은 UPnP Device Control Protocol로서 정의되어 있다. Multi-Screen에 적용될 수 있는 Content Sharing 관련 Device Control Protocol은 UPnP A/V Working Committee에서 정의하고 있다. A/V Working Committee에서는 Device를 Media Server와 Media Renderer로 분류하고 콘텐츠 관리 및 Server 제어 (Content Directory Service), 세션 관리 (Connection Manager Service), 스트리밍 제어 (AV Transport Service) 및 Renderer 제어 (Rendering Control Service)를 정의하고 있다.

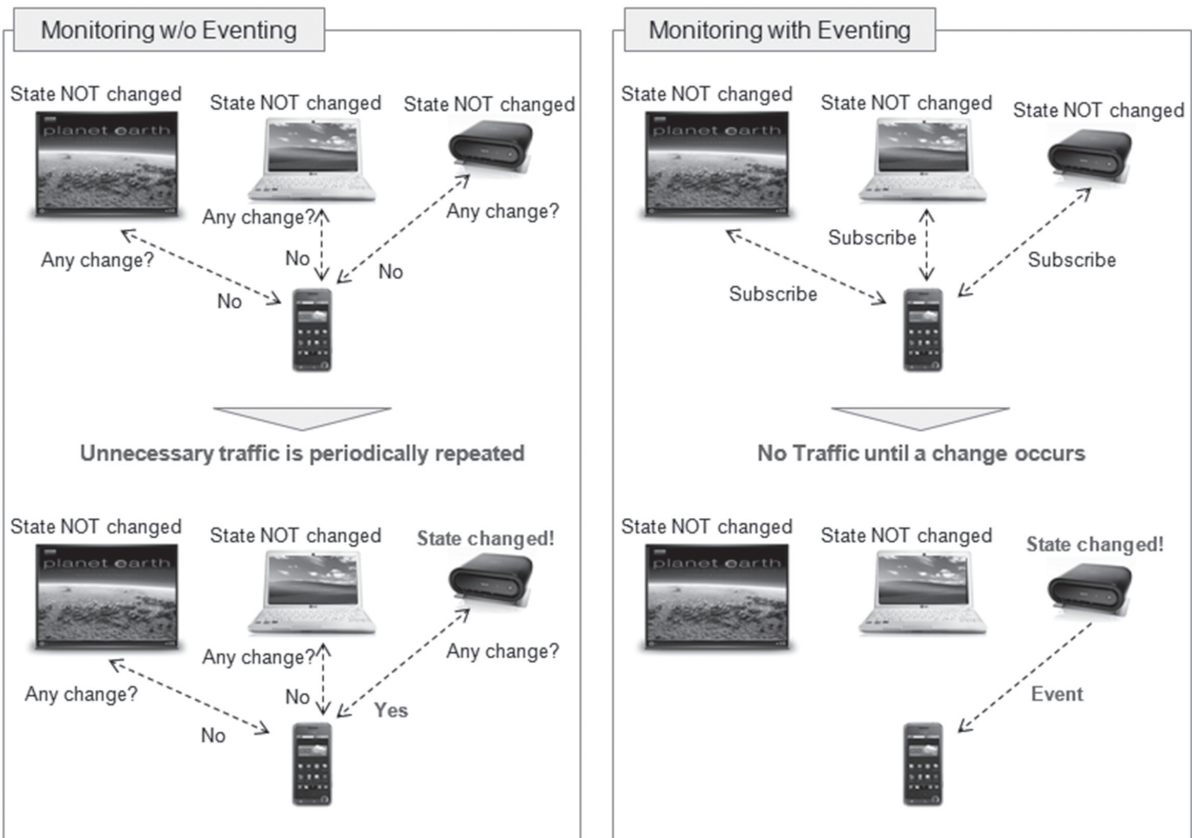


그림 5. UPnP Eventing Mechanism

UPnP AV표준기술은 서로다른 제조업체의 기기 간 Content Sharing 서비스를 제공하기 위한 Architecture와 Control Protocol을 제공하면서, 그 하위 계층 기술 (PHY/MAC, Transport) 및 상위 계층 기술 (Media Format, Link Protection)에 대해서는 제조업체가 자유롭게 선택할 수 있도록 제약을 두지 않고 있다. 그러나 호환성을 보장하기 위해서는 해당 계층 기술에 대한 제약이 필요하고 이러한 제약사항은 DLNA (Digital Living Network Alliance)에서 규격화하고 있다.

DLNA에서는 PHY/MAC 표준기술로 Ethernet, Wi-Fi, MoCA, Bluetooth를 채택하고 있고, Device Discovery & Control 표준기술로는 UPnP Device Architecture, Media Management 표준기술로는 UPnP AV Device Control Protocol, Transport 표준기술로는 HTTP, RTP, Media Format 표준기술로는 MPEG2, MPEG4, AVC/H.264, LPCM, MP3, AAC LC, JPEG 등을 채택하고 있으며, Link Protection 표준기술로는 DTCP-IP와 WMDRM-ND를 채택하고 있다.

최근 생산된 대부분의 PC, 스마트폰, 스마트TV 및 DVD/Blu-ray Player, Game Console, Photo Frame, NAS (Network Attached Server)는 이미 UPnP/DLNA 기술이 구현되어있으며, 그 수량은2012년 현재 3억대 이상으로 조사된 바있다 (ABI Research).

## 2. DIAL(Discovery And Launch) 과 AllJoyn

DIAL (Discovery And Launch) 기술은 Netflix 가 2nd 스크린 기기(스마트폰)에서 1st 스크린 (스마트TV) 기기를 찾아서 App을 실행시키기 위해 개발된 개방형 기술로써 무료 라이선스로 배포되고 있으며, LG, 삼성 등의 가전 업체 및 Youtube, Hulu, BBC 등의 콘텐츠 제공 업체들에서 지원하고 있다. UPnP 기술이 홈 네트워크 상에 존재하는 UPnP기기간에 저장된 콘텐츠를 이용하는 것이라고 하면, DIAL기술은 인터넷 상의 콘텐츠를 DIAL을 지원하는 기기간에 App을 통해 이용하는 기술이라 할 수 있다[1].

〈그림 6〉는 DIAL을 이용한 N 스크린 연동 방안으로 스마트폰에서 검색된 Youtube 콘텐츠를 TV에서 이용하기 위한 시나리오 절차를 보여주고 있다. 우선 2차 스크린인 스마트폰에서 Youtube 상의 콘텐츠를 검색한 후에 1차 스크린 기기의 Youtube App을 DIAL을 통해 실행시키고 검색된 콘텐츠 정보를 제공해 준다. 그러면 1차 스크린 기기가 인터넷을 통해서 해당 콘텐츠를 직접 play하여 사용자가 이용할 수 있다.

이 과정에서 DIAL은 2차 스크린에서 1차 스크린을 발견하

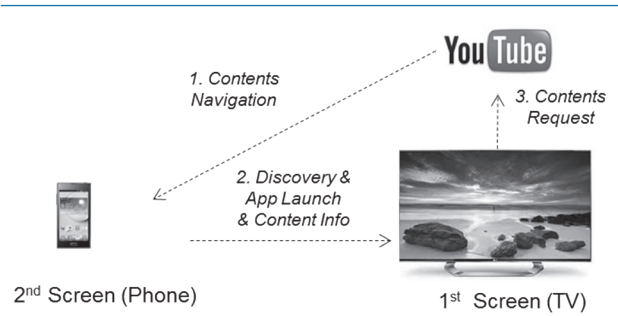


그림 6. DIAL을 이용한 N스크린 연동 방안

는 과정과 1차 스크린의 App을 실행시키고 콘텐츠 정보를 제공해 주는 역할을 수행하게 된다. 1차 스크린을 발견하는 과정은 UPnP와 동일하며, SSDP를 사용하여 1차 스크린의 DIAL지원 여부를 알 수 있다. DIAL은 이를 위하여 새로운 ST(Service Type: urn:dial-multiscreen-org:service:dial:1)을 정의하였다. 그리고, 발견된 1차 스크린의 App을 실행하는 방법은 UPnP에서 사용하는 제어 프로토콜인 SOAP을 사용하지 않고 REST API 기술을 사용하며, 이를 위해서 Application-URL 정보를 제공해야 한다. 이 Application-URL 정보를 활용해서 2차 스크린 기기는 1차 스크린 기기의 App을 제어(실행, 중지, 애플리케이션 정보 요청) 할 수 있다.

한편, AllJoyn은 가정 및 자동차의 근접 네트워크 환경에서 연결된 기기 및 S/W App 간의 상호 연동성을 제공하기 위한 S/W 프레임워크를 개발하는 오픈 소스 프로젝트이다. 이 기술은 웹캠에서 주도하고 있으며, 〈그림 7〉과 같이 기기 및 서비스 (App) 발견, 네트워킹 관리 및 메시지 라우팅, 다양한 OS 및 네트워킹 지원, 기기 및 서비스간 연결성 제공 등의 기능을 제공하고 있으며 추가적으로 메시지 인증 및 암호화를 제공하여 안전하게 기기간 통신을 가능하게 하는 환경을 제공하고 있다. 특히 S/W 개발자가 네트워크 계층이나 OS 계층의 동작에 대하여 신경을 쓰지 않고도 다양한 기기간 연동 S/W를 개발할 수 있는 환경을 제공하며 Linux, Windows, Android, OS X등을 지원하는 SDK들을 배포하고 있으며 적은 메모리를 사용하는 임베디드 기기를 지원하기 위한 AllJoyn Thin Client, 오디오 스트



그림 7. AllJoyn 프레임워크 [2]

리밍을 위한 Audio SDK, 짧은 메시지 전송을 위한 Notification SDK 등을 추가적으로 배포하고 있다.

〈그림 8〉은 AllJoyn의 핵심인 기기간 메시지를 교환할 수 있는 구조를 보여준다. 각 기기 (Smartphone & Linux Host)에는 독립된 Daemon (D) 이 존재하고 각각의 기기에 있는 Client (C) 들과 Service (S) 들과 이 Daemon과 연결되어 있으며 Daemon 간의 연결은 AllJoyn Bus라는 논리적 통로로 연결되어 있다. 따라서 기기 내부에서 메시지를 주고 받을 경우에는 Daemon이 기기 내부에서 메시지를 중계하는 역할을 수행하고, 기기 외부와 메시지를 주고 받을 경우에는 기기 내부의

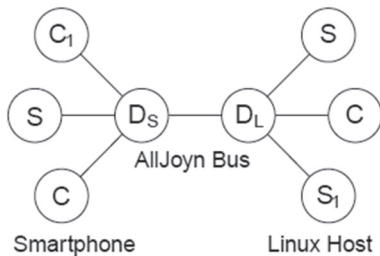


그림 8. AllJoyn의 메시지 교환 구조 [2]

Daemon이 다른 기기의 Daemon에게 메시지를 전달하여 다른 기기의 Client 및 Service에게 전달되고, 다른 기기로부터 받는 메시지들도 이 Deamon을 통해서 받게 된다. 기기 발견 및 서비스 발견을 위해서 각 Daemon들은 다른 기기의 Daemon을 발견하고 기기 내부의 Client 및 Service 들의 정보를 다른 기기의 Daemon에게 제공하여야 한다. AllJoyn은 기기 내부의 App간 통신을 하기 위한 프레임워크를 제공하고 있다.

DIAL과 AllJoyn은 쉽게 기기간 연동을 제공하기 위해서 특정 업체 주도적으로 개발된 기술로서, 다른 기기의 서비스를 간편하게 발견하고 이를 제어할 수 있는 기술이다. 하지만, 이 기술들은 UPnP/DLNA에서 제공하는 콘텐츠 관리, 서버 제어, 스트리밍 제어와 같은 기술들에 대해서는 정의하지 않고 있다. 아직까지는 다양한 사용자의 요구를 만족하기 위한 기능들을 제공하지 않지만, 단순하고 개발자가 접근하기 편한 장점을 가지고 있기 때문에 주목을 받고 있으며, 때로는 UPnP/DLNA와는 경쟁하면서 또는 공존을 통한 기술 개발이 이루어질 것으로 예상된다.

### 3. Smart TV Alliance

LG전자와 TP Vision은 2013년 6월에 20일에 공식 Press Release를 내고 글로벌 Smart TV 시장을 선호하기 위하여 Smart TV Alliance(<http://www.smarttv-alliance.org>)를 만들었고, 현재는 Toshiba, Panasonic 과 같은 가전사들을 포함하여 VESTEL, XUMO, Qualcomm, IBM, Opera, Technisat, ABox42, Dolby, IBM, OBIGO, FPT Software, Realtek, Warner Brothers등의 19개의 멤버사들로 구성되어 있다. Smart TV Alliance 목표는 스마트TV App 개발자들에게 매력적이며 플랫폼 독립적인 서비스를 만들 수 있는 통합된 ecosystem을 제공함으로써 고객에게 확장된 UX(User Experience)를 제공하는 것이다. 통합된 ecosystem 및 플랫폼 독립적인 환경은 개발자들에게 각 제조사들의 플랫폼에 따른 불필요한 리소스의 낭비를 최소화 하고 하나의 개발 환경으로 통합된 테스트를 통하여 창조적인 애플리케이션을 만들 수

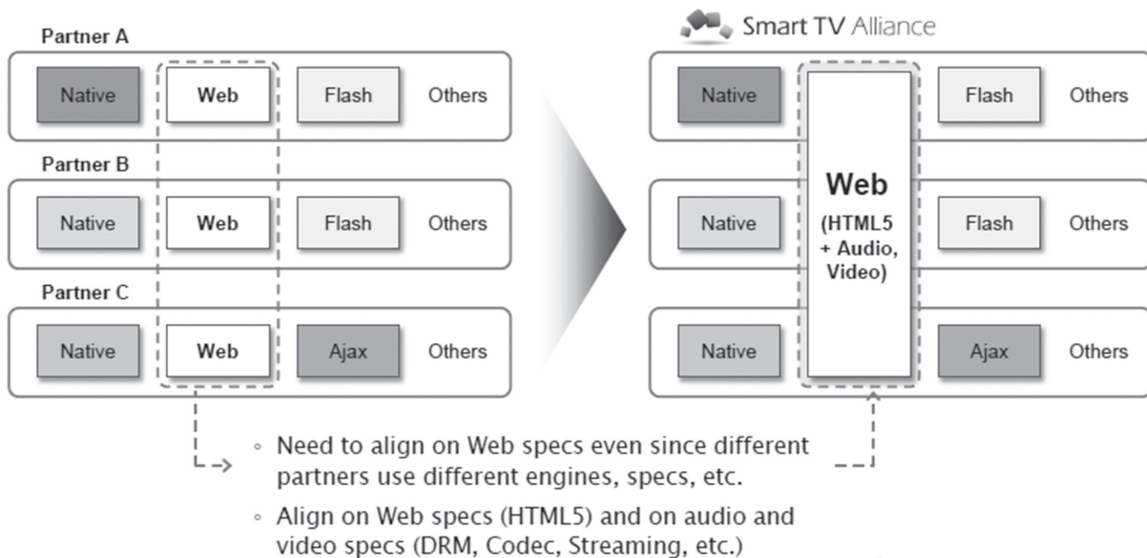


그림 9. Smart TV Alliance Invitation to Join (May 2014)

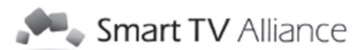
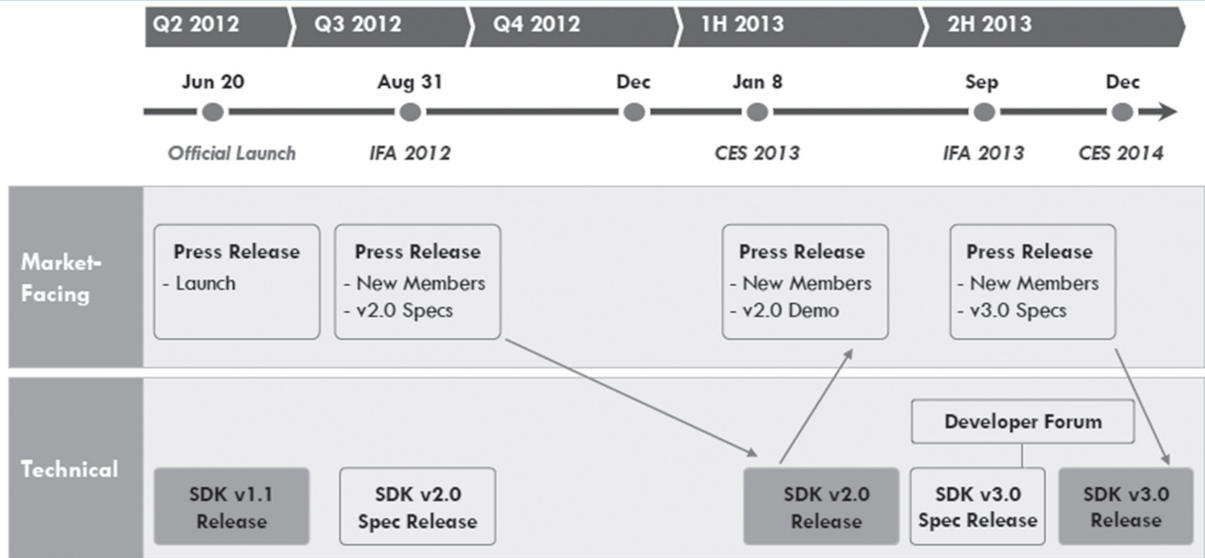


그림 10. Schedule (Smart TV Alliance Invitation to Join (May 2014))

있도록 하는 것[5]으로 Smart TV Alliance의 모토인 “Build once, Run Everywhere”에서도 이러한 강력한 의지를 알 수 있다. Smart TV Alliance의 핵심 기술은 Open Standard인 W3C HTML5 기술이며, 이는 제조사들이 공통적으로 가지고 있는 애플리케이션 실행 환경이다.

W3C HTML5 표준 기술 외에도 WebApp이 동작하기 위해서는 필요한 부가 기술들이 필요하다. WebApp에서 비디오 전송을 위한 Adaptive Streaming, DRM 기술, 코덱, 멀티스크린 등이 포함된다. 다음은 2012년부터 2013년까지 Smart TV Alliance의 표준화 일정으로 현재 Smart TV Alliance Specification v2.5가 공식 홈페이지에서 배포 되었으며, 이에 따르는 개발 환경인 Eclipse 기반의 SDK도 곧 배포될 예정이다. 현재는 v3.0에 대한 표준화가 진행중에 있으며, IFA2013에 발표될 예정이다.

멀티스크린은 산업계에서 많은 표준화가 진행이 되고 있으나, 실제로 사용하는 기술들을 채택하여 사용된 통합 표준은 Smart TV Alliance가 처음일 것이다.

Smart TV Alliance Technical Specification v2.5는 위와 같은 구조를 가지며, Multiscreen 기능을 지원하기 위하여 DIAL과 W3C WebSocket을 포함하고 있다. DIAL은 앞 장에서 소개한 바와 같이 Smart TV 제조사들이 Netflix와 Youtube 등과 같은 App들을 Smart TV 에서 원격으로 실행하기 위하여 채용하고 있는 기능으로, Smart TV Alliance에서는 이를 이용하여 2nd 스크린에서 Smart TV용 WebApp을 실행할 수 있도록 하였다. Smart TV에서 실행한 WebApp과 2nd 스크린 간의 App to App 통신은 W3C WebSocket을 이용하여 통신을 할 수 있으며, 2nd 스크린의 모바일 App은 WebSocket

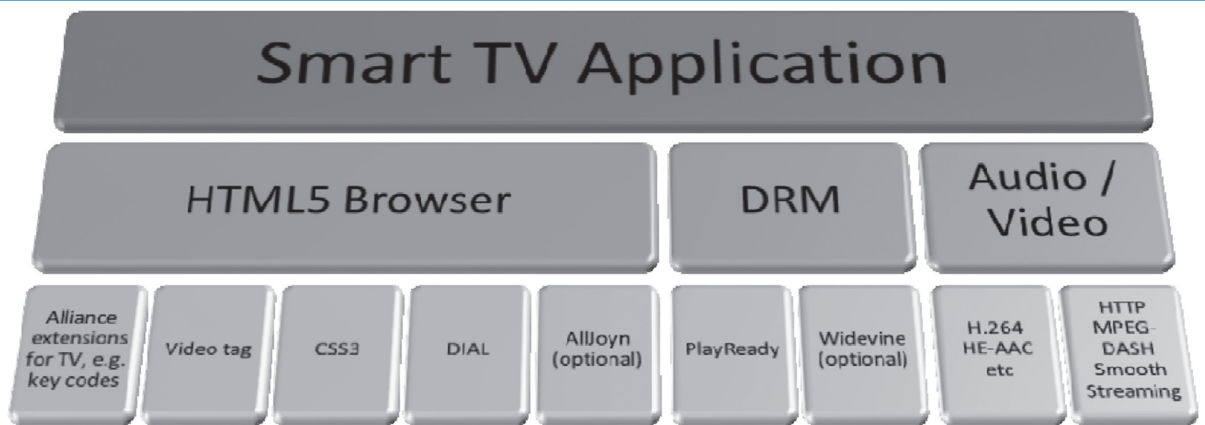


그림 11. Smart TV Alliance Architecture v2.5

Server를 내장하여 Smart TV가 2nd 스크린의 모바일 App에 WebSocket 접속하여 양방향 통신을 할 수 있다. 그러나 App to App 통신 프로토콜은 서비스 제공자가 개발해야 하는 부담을 줄여 주기 위하여 WebApp 개발자들에게는 보다 손쉽게 확장된 Multiscreen 기능을 WebApp에서 구현할 수 있도록 AllJoyn (<https://www.alljoyn.org/>) 기술을 선택적으로 지원하고 있다. AllJoyn은 Open Standard로 누구나 쉽게 Smart TV Alliance SDK 및 AllJoyn SDK를 다운받아 개발환경을 설정할 수 있다. AllJoyn은 다양한 플랫폼을 지원하므로 원하는 플랫폼의 모바일 App을 구현할 수 있다[7].

#### 4. Wi-Fi Display (Miracast)

Home Network 환경에서 멀티 스크린을 구성하는 또 다른 기술로서 Wi-Fi통신망을 응용하는 Wi-Fi Display 표준 기술이 있다. 이 표준 기술은 Wi-Fi Alliance 에서 제안하고 2012년 8월 v1.0 표준이 발표되었으며, 최근 출시되고 있는 많은 모바일 장치(스마트폰 등)와 대내 화상 장치(TV, 모니터 등)가 이 기술을 지원하고 있다. 본 기술은 Phone이나 노트북 등의 화면과 음성을 TV나 모니터 등의 대화면 기기로 실시간 전송하여 재생시키는 응용 계층의 전송 기술로서 기존 Wi-Fi 무선 통신 기술을 활용하여 실시간 스크린 미러링 기능을 지원하기 위하여 정의되었으며, 표준 기술로서 WFD(Wi-Fi Display)라 명칭하며 마케팅 용어로서 Miracast라고 부른다.

Wi-Fi는 노트북, 넷북, 태블릿 및 스마트폰의 필수 기능일 뿐만 아니라, 이들 장치에서 발생하는 트래픽 전송 방안에서 있어서도 가장 지배적인 기술이다. 미국 comScore에 따르면 태블릿 트래픽의 90% 이상이 Wi-Fi를 통해 전송되고 있으며[3], 이러한 트래픽은 대부분 가정 내에서 발생되고 있다. 오늘날 가정은 특정한 사용 용도나 contents 유형에 따라 최적화된 온갖 연결된 장치들이 범람하고 있는 최첨단 환경이다. 이러한 contents는 가정용 데스크탑 등의 중앙화된 위치에 저장되어 있기 보다는 모바일 등의 다양한 장치에 걸쳐 분산화되어 있다. 가정 내에 여러 모바일 장치에 저장되어 있는 contents를 사용자가 제조사나 유형에 관계없이 대화면 장치에서 감상할 수 있

도록, WFD 기술은 보다 편리한 사용성과 끊임없는 영상 재생을 제공하기 위하여 표준화 차원에서 제안되었다.

이러한 WFD 기술은 <그림 12>와 같이 WFD Source 장치와 WFD Sink 장치로 구분되며, 각각 영상 정보를 공유하려는 모바일 장치와 이를 화면에 표출하기 위한 화상 장치가 이에 해당된다. 먼저 WFD Source 장치는 자신의 영상과 음성 정보를 실시간으로 Sink 장치에 송신하여 영상 공유를 수행하며, 이때, Sink 장치에서는 마우스 등의 주변장치를 통하여 화면 제어가 가능하다. 이러한 주변장치 이벤트는 WFD Sink 에서 정보화하여 WFD Source로 송신되며, WFD Source 장치는 이 정보를 기반으로 자신의 화면정보를 제어한 후, 변경된 화면 정보를 WFD Sink로 다시 송신하는 동작을 통하여 WFD Source장치와 WFD Sink의 주변장치간 동기화를 수행한다.

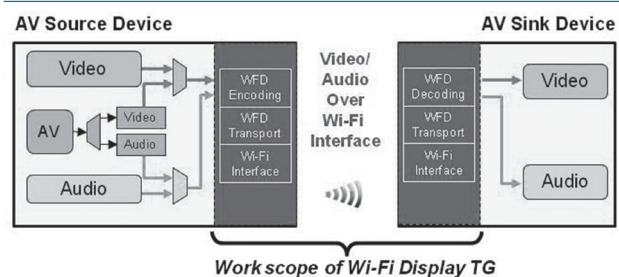


그림 13. Source 및 Sink 장치의 기능 구성

WFD Source로서의 모바일 장치는 자신의 영상 및 음성 정보를 <그림 13>과 같이 실시간으로 각각 인코딩하고 이를 Wi-Fi 기술을 매체로 무선으로 전송하며, WFD Sink로서의 화상 장치에서는 수신된 인코딩 데이터를 디코딩 한 후, 이 정보를 화면에 표출하여 실시간 스크린 미러링 기능을 수행한다. 이때, 데이터 전송 매체로서 사용되는 Wi-Fi 무선 통신기술은 기존의 다양한 Wi-Fi 기술을 모두 지원하지는 않으며, 장치간 단대단 통신을 위한 Wi-Fi Direct 표준 기술과 AP 접속 기반의 TDLS (Tunneled Direct Link Setup) 기술을 기반으로 한다.

WFD에서 사용되는 통신 프로토콜을 계층적으로 분류하여 도식화하면 <그림 14>와 같다. Wi-Fi Display를 사용하기 위한 전체 시스템 Architecture는 크게 Interface를 담당하는 Wi-Fi Direct 및 TDLS, 상위 WFD 구동을 위한 TCP/IP, 마지막으로 영상 전송을 위한 RTSP 및 RTP 등의 응용계층 프로토콜들로 구성된다.

일반적으로 AP가 존재하는 환경에서는 동일 네트워크를 이용하기 위하여 Wi-Fi Connection에 WFD Source와 WFD Sink 장치가 모두 참가하며, 이때 영상 데이터의 Low Latency를 위하여 WFD는 TDLS 방식을 통하여 수행된다. 또한, AP가 존재하지 않는 환경에서 두 기기 간 직접 영상 전송을 지원하기 위

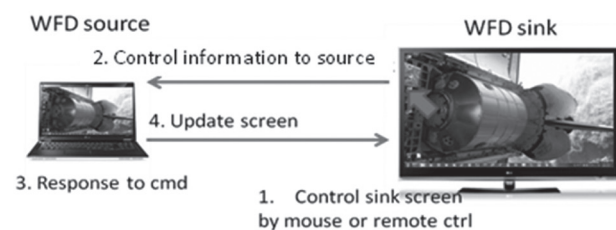


그림 12. Wi-Fi Display 각 구성 장치와 동작 흐름

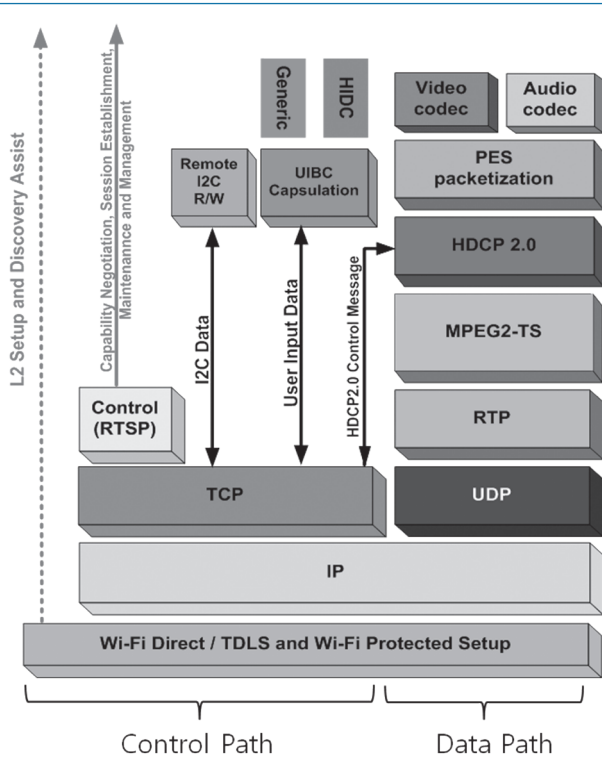


그림 14. Wi-Fi Display Architecture [4]

하여 WFD는 Wi-Fi Direct 방식을 추가적으로 지원한다.

WFD는 이러한 두 가지 방식을 통하여 선택적으로 이루어진 무선 접속을 기반으로, 상위 응용계층의 프로토콜 수행을 위한 IP설정을 TV 등의 WFD Sink를 중심으로 진행한다. 이때, 만약 Wi-Fi Direct를 통하여 무선 접속이 이루어진 경우에는 두 기기간의 일대일 접속을 통하여 네트워크가 단발적으로 형성되었기 때문에, WFD Source 장치는 접속된 WFD Sink 장치와의 데이터 통신을 위한 IP 설정이 이루어지지 않을 가능성이 크다. 이 경우 SFD Sink 장치는 자신을 위한 임의의 IP 주소를 설정한 후, DHCP (Dynamic Host Configuration Protocol) 등의 기술을 이용하여 동일 네트워크 내의 또 다른 임의의 IP 주소를 WFD Source 장치에게 할당해 준다.

WFD 관련 응용 프로토콜은 RTSP(Real Time Streaming Protocol) 와 RTP(Real-time Transport Protocol) 로 구분된다. RTSP는 WFD Session을 생성하고 영상 데이터 전송을 위한 초기 Pairing 관련 부분을 담당하는 프로토콜이며, WFD Source와 WFD Sink의 Capability Negotiation (지원되는 동영상 Format, 해상도, Codec 정보 등), Session Management 역할을 주로 수행한다. 또한 WFD Sink에서의 마우스, 키보드 등의 입력장치를 통하여 발생된 사용자 제어정보는 RTSP의 Back-Channel 기능을 통하여 WFD Source로 송신된 후 Source 장치에서 명령 처리가 수행된다.

RTSP 설정이 완료된 후, WFD Source의 영상 및 음성 정보를 포함한 Streaming 데이터는 RTP를 통하여 실시간 전송된다. 이러한 RTP 프로토콜 관련 설정 정보는 RTSP 협상 절차에서 확정되며, RTP port 정보 등이 포함된다.

참고적으로 WFD에서 지원되는 영상 및 음성 정보는 아래 <그림 15>와 같다.

Miracast: Supported formats	
Display Resolution	<ul style="list-style-type: none"> <li>17 Consumer Electronics Association (CEA) formats, from 640 x 480 up to 1920 x 1080 pixels, and from 24 to 60 frames per second<sup>2</sup> (fps)</li> <li>29 Video Electronics Standards Association (VESA) formats, from 800 x 600 up to 1920 x 1200 pixels, and from 30 to 60 fps</li> <li>12 handheld formats, from 640 x 360 up to 960 x 540 pixels, and from 30 to 60 fps</li> </ul>
Video	<ul style="list-style-type: none"> <li>ITU-T H.264 (also known as Advanced Video Coding [AVC]) for high-definition (HD) video, supporting the Constrained Baseline Profile (CBP) and the Miracast-specific Constrained High Profile (CHP), at levels ranging from 3.1 to 4.2.</li> </ul>
Audio	<ul style="list-style-type: none"> <li>Mandated codec: Linear Pulse-Code Modulation (LPCM) 16 bits, 48 kHz sampling, 2 channels</li> <li>Optional audio codecs, based on:                             <ul style="list-style-type: none"> <li>LPCM mode 16 bits, 44.1 kHz sampling, 2 channels</li> <li>Advanced Audio Coding (AAC) modes</li> <li>Dolby Advanced Codec 3 (AC3) modes</li> </ul> </li> </ul>

그림 15. WFD에서 지원하는 영상 및 음성 format 정보 [3]

### III. 결론

지금까지 Machine to Machine 통신을 통한 N-Screen 서비스의 근간이 되는 다양한 표준 기술에 대해 살펴 보았다.

본고에서 살펴본 표준들은 모두 TCP/IP 프로토콜을 기본으로 하여, Application을 구현하는 기술이다. 각각의 표준들은 약간의 정도 차이는 있겠지만 대부분 다음에 설명하는 개념을 포함하고 있다고 간주하면 이해하는 데 큰 무리가 없을 것이다. 우선 Physical layer는 TCP/IP 패킷을 전달하기 위한 Interface로 유선은 Ethernet, 무선은 Wi-Fi를 기본으로 한다. IP layer에서는 자동 Address할당과 홈 네트워크 범위 내 통신을 위한 Multicast domain 개념을 적극 활용하고 있다. Transport layer에서는 traffic의 신뢰성이 필요한 경우에는 TCP, 신뢰성보다 신속성과 다양한 peer와의 통신이 필요한 경우에는 UDP를 적절히 사용하고, 상위 Application layer에는 기기 상호간 Capability를 파악하고 상호간 능력에 따라 Session을 협상하는 과정이 포함하고 있다. 이에 덧붙여 일반적인 Network Layer에서 정의하지 않고 있는 Content format, Link Protection, User Account, Authentication 등의 개념이 기기간 호환성을 증진시키기 위해 정의되어 있다.

M2M을 위한 근거리 Peer to Peer 통신의 가장 기본이 되는 표준 기술은 HTTP와 XML 기술을 바탕으로 SSDP, SOAP, GENA같은 IETF 기술들을 채용한 UPnP라 볼 수 있으며, 이를



영상 등 AV 서비스에 특화 시킨 표준이 DLNA이다. DLNA는 UPnP AV를 상용화하기 위한 Alliance개념의 표준단체이다.

Netflix가 제안한 DIAL은 UPnP에서 채택한 SSDP와 Web기술인 RESTful API를 사용해 상대방 기기의 App을 launch하는 매우 simple한 기술이고, AllJoyn은 Qualcomm이 제안한 Peer to Peer간 SW Network Bus구조라 볼 수 있다. 이 두 기술은 표준은 아니지만 제안사가 무료로 Industry에 Open하였고, 차츰 채용 서비스가 늘어나는 추세이기 때문에, 본고에 포함하였다.

SmartTV Alliance는 HTML을 기반으로 SmartTV platform을 구현하는 목적으로 시작되어, v2.5에 이르러서는 기기간 통신 시나리오도 포함하여 AllJoyn과 W3C Websocket을 규격에 채용하고 있다.

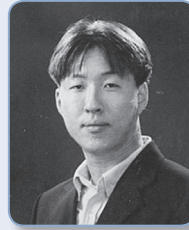
Wi-Fi Display는 Mirroring 기술로 Mira(Mirror) Cast(던지다, 전송하다)로부터 별명인 Miracast가 탄생하였다. 다른 표준 기술과 달리 AP가 필요 없는 1:1 무선기술인 Wi-Fi Direct 위에서 동작하며, 특히 Mirroring을 위한 Video content format을 상세하게 정의하였다.

Machine to Machine 통신은 본고에서 살펴본 근거리 Peer to Peer 기반의 통신으로부터 점차 Internet으로 범위를 확장하여 최근에는 Internet of Thing(IoT)라는 용어로도 회자되고 있으며, Internet에 연결된 모든 기기들이 상호간을 자동으로 인식하고, 주어진 context에서 최적화된 방식으로 통신하는 방식으로 진화 중이다.

## 참고 문헌

- [1] 유재홍, "DIAL: Netflix의 개방형 N-Screen 프로토콜 기술 규격", 정보통신산업진흥원, pp. 1-13,
- [2] AllJoyn 홈페이지, <http://www.alljoyn.org/>
- [3] "Wi-Fi CERTIFIED Miracast: Extending the Wi-Fi experience to seamless video display", Wi-Fi Alliance White-Paper, Wi-Fi Alliance, pp. 1-18, Sep. 2012.
- [4] "Wi-Fi Display Technical Specification Version 1.0.0", Wi-Fi Alliance, pp. 1-152, Aug. 2012.
- [5] "Smart TV Alliance Invitation to Join May 2013", Smart TV Alliance, Inc
- [6] "Smart TV Alliance Technical Specification v2.5", Smart TV Alliance, Inc. pp 1-37, 2nd, May, 2013
- [7] "<https://www.alljoyn.org/about/core-services>", AllJoyn.

## 약 력



양 승 루

LG전자 Convergence 연구소 선임연구원  
2011년~현재 UPnP Steering Committee member, AV WG chair  
2011년~현재 Wireless HD Board member  
2012년~현재 HDMI Board member  
관심분야: Device Control Protocol, 유무선 Communication Interface



권 영 환

LG전자 Convergence연구소 선임연구원  
2005년~2006년 TTA PG219 (IPTV) Editor  
2006년~2008년 ITU-T SG13 Editor  
2012년 DLNA Low Power WG vice chair  
2013년~현재 Bluetooth Architecture Review Board  
관심분야: Home network and cloud, Bluetooth.



이 재 호

LG전자 Convergence연구소 선임연구원  
2012년 Reviewer of Journal 'Wireless Personal Communications'  
관심분야: Sensor Network, Body Area Network, Personal Area Network, Mobile Ad-hoc Networks



김 경 호

LG전자 컨버전스 연구소 책임연구원  
2011년~현재 TTA PG804 (Interactive Broadcasting) vice chair  
2012년~현재 Smart TV Alliance Specification WG vice chair  
관심분야: Broadcasting, Network and Cloud standard



이 현 재

LG전자 CTO Convergence 연구소, 수석연구원  
2011년~현재 TTA PG219 (IPTV) vice Chair  
2012년~현재 UPnP Board member  
2012년~현재 HDBaseT Board member  
2011년~2013년 W3C 'Web and TV' IG Chair  
관심분야: 홈네트워크 및 유무선 Connectivity 표준

## 약 력



김진필

LG전자 Convergence 연구소, STD 팀장, 연구위원  
1997년 ATSC-TG, DVB-TM, ATIS-III, TTA TC8  
Representative of LGE  
관심분야: 방송 및 홈 네트워크 표준 설계