

## 특집 스마트스크린 특집

원고

### N-스크린 서비스를 위한 원격 사용자 인터페이스 기술

구태연

[kutai@etri.re.kr](mailto:kutai@etri.re.kr)

(한국전자통신연구원 선임연구원)



구태연

[bjoh@etri.re.kr](mailto:bjoh@etri.re.kr)

(한국전자통신연구원 책임연구원)



오봉진

[baeys@etri.re.kr](mailto:baeys@etri.re.kr)

배유석

[baeys@etri.re.kr](mailto:baeys@etri.re.kr)

(한국전자통신연구원 선임연구원)



박종열

[jongyoul@etri.re.kr](mailto:jongyoul@etri.re.kr)

(한국전자통신연구원 선임연구원/팀장)



## 1. 서론

스마트폰의 폭발적인 보급에 따라 모바일 인터넷의 비약적 성장과 다양한 애플리케이션의 등장을 유발하였다. 더욱이 Morgan Stanley의 발표에 의하면 모바일 인터넷의 확산 속도는 유선 인터넷보다 5배 이상 빠르게 진행될 것으로 예상된다. 이에 스마트 폰에서 시작한 스마트 열풍은 스마트 패드, 스마트 TV로 발전하고 있다. 이와 더불어 보다 똑똑하며 이용이 편리한 콘텐츠 소비는 전자북(e-Book)을 넘어 스마트 미디어로 발전할 것으로 전망된다. 또한, 스마트 단말 보급 확산과 더불어 사용자는 장소와 시간에 구애 받지 않는 서비스를 제공 받기 원하게 되었다. 이는 스마트 콘텐츠 소비 기회가 증가하여 콘텐츠 및 서비스 시장의

확대를 가져올 것으로 예상된다.

N-스크린은 TV나 PC, 태블릿PC, 스마트폰 등 다양한 디바이스에서 하나의 콘텐츠를 끊임없이 이용할 수 있게 해주는 서비스를 말한다. 사용자가 구입한 콘텐츠가 단말기가 아니라 서버에 저장되어 있기 때문에 언제 어디서나 다양한 단말기로 불러와 이용할 수 있다는 장점이 있다. N-스크린 서비스를 가능하게 하는 핵심 기술은 다양한 디바이스들을 상호 연동하여 디바이스에 종속되지 않은 서비스 사용이 가능하도록 하는 디바이스 협업 오픈 미들웨어 기술과 원격 사용자 인터페이스 기술로 구분된다. N-스크린 기반 개방형 디바이스 협업 기술은 게임, 방송, 교육, 소셜 미디어 등의 다양한 콘텐츠를 효과적으로 제공하고 다양한

디바이스들 간에 데이터 공유가 가능하다. 또한 확장 가능한 원격 사용자 인터페이스 기술을 이용하면 방송, 동영상, 이미지, 웹 콘텐츠를 다양한 스마트 디바이스에서 동시에 사용할 수 있다.

N-스크린 서비스 기술은 다양한 스마트 기기의 대중화로 인해 KT, SKT, LGU+ 등 통신 사업자 뿐 아니라 삼성전자, LG전자 등 제조업자 및 NHN, 구글 등 포털업체 까지 관심이 급증하는 분야로, 언제 어디서나 원하는 단말로 서비스를 받을 수 있는 스마트 스페이스 구축의 핵심이 될 수 있다.

본 논문에서는 사용자가 다양한 스마트 단말과 TV, 가전을 사용하는 스마트 홈 환경에서, 콘텐츠, 응용 프로그램 및 단말 환경에 구애받지 않고, 다양한 기기들이 상호 협업하여 스마트TV 환경에 최적화되고 효과적인 다기능 융합 서비스를 제공하기 위한 N-스크린 서비스를 위한 원격 사용자 인터페이스 기술을 살펴보도록 한다.

## 2. N-스크린 서비스를 위한

### 원격 사용자 인터페이스 시스템

가정에 스마트 디바이스들이 증가하면서 원격에서 이들 스마트 디바이스를 제어할 수 있는 원격 사용자 인터페이스에 대한 연구가 확산되고 있으며, 이들은 Native UI를 전송하는 방식과 마크업(mark-up)언어를 사용하는 방법이 있다. VNC(Virtual Network Computing), X-윈도우즈 등에서 사용하는 Native UI 전송 방식은 서버에서 화면을 구성하여 전송하는 방식으로 쉽게 구현 가능하나, 전송 대역폭이 크고 실시간 보장이 어려우며 장치의 해상도나 특성을 반영하는데 한계가 있다. Mark-up 언어를 이용하는 방식은 UI 구성과 UI에 대한 동

작 방식을 기술하고 그에 따라 클라이언트에서 렌더링하는 방식으로, UI를 기술하는 방법으로 HTML, XML 또는 SVG(Scalable Vector Graphics)등이 사용되고 있다. UPnP RUI는 UI를 원격에서 실행시킬 수 있도록 디스플레이 장치에서 UI를 검색하고 제어하려는 디바이스와 세션을 설정해주는 프로토콜을 정의하고 있다.

N-스크린용 원격 사용자 인터페이스 기술이란 네트워크에 연결되어 있는 다양한 디스플레이에 적합하도록 인터페이스를 표현하고, 주변의 다양한 IO장치를 결합하여 가상의 IO를 제공할 수 있는 인터페이스 기술이다.

N-스크린 원격 사용자 인터페이스 시스템의 구조도는 그림 1과 같다.



그림 1. N-스크린 원격 사용자 인터페이스 시스템 구조도

N-스크린용 원격 사용자 인터페이스 시스템의 핵심 기술은 디스플레이에 적합하도록 UI 표현 및 동적 명령어 결합을 제공하는 Remote UI 엔진, IO 장치 성능에 따라 UI 생성 및 변환하는 UI 변환 엔진, Networked IO 장치를 결합하여 가상의 IO를 제공하는 Virtual IO 디바이스 관리자, 디바이스에 맞게 포맷 변환 및 경험 기반 필터링하는 미디어 트랜스코더로 구성된다.

#### 2.1 Remote UI 엔진

Remote UI 엔진은 다양한 단말에 “적응형 N-스크린 서비스 UI”를 제공하기 위한 것으로

로, N-스크린 원격 사용자 인터페이스 시스템에서 UI 변환 엔진과 연계하여 원격으로 서비스를 제어하기 위하여 단말 환경을 고려한 스케일러블 RUI (Remote User Interface)를 제공하는 기능을 수행한다.

RUI 엔진(이하 RUIE)은 시스템에서 RUI 서비스를 위한 엔진으로, N-스크린 서비스 환경에서 원격으로 접속하여 서버에서 실행되는 화면을 전송 받아 가상 뷰를 생성하고 이벤트를 원격 서버에 전달하여 서비스를 제어하는 기능과 서비스와 분리된 UI를 통하여 원격으로 서비스를 제어하는 기능을 포함하며, Remote UI 서비스를 위한 RUIE의 소프트웨어 구조도는 그림 2와 같다.

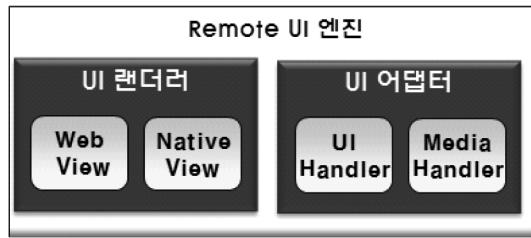


그림 2. Remote UI 구조

RUIE 기능은 크게 UI 렌더러 기능과 UI 어댑터로 구성되며, UI 렌더러 기능은 웹 뷰 기능과 네이티브 뷰 기능의 세부 기능을 제공하며, UI 어댑터 기능은 UI 핸들러 기능과 미디어 핸들러 기능의 세부 기능을 제공한다. 웹 뷰 UI 렌더링 기능이란 UI 변환 엔진에서 전송된 웹 UI를 해석하고, 이를 디스플레이 장치에 표시하기 위한 기능을 담당하는 모듈이다. 또한 RUI 환경에서 사용자 입력을 UIFM(User Interface ???, 문서에서 설명이 없음)에 전달하는 기능을 수행 한다. 네이티브 뷰(?? Native View ?? 한글과 영문이 혼용되어 하나로 통일 필요)는 UI를 해석하고, 이를 처리 디스플레이 장치(?? 단말 디스플레이 장치 ??)로 표시하는 기능과 로컬에서 발생하는 사용자의 이벤트를 UI 변환 엔진에 전달하는 기능을 담당하는 모듈이다. 로컬 서비스에서 UI에 발생되는 이벤트를 원격지 UI에 반영하기 위해서

는 우선 서비스에서 발생하여 UI에 변경을 주는 이벤트를 모니터링 하고, 이를 통해 UI를 변환하여야 할 필요가 있는 경우 이를 UI 변환 엔진에 전달하여 원격지 UI에 반영될 수 있도록 하여야 한다. UI 핸들러는 해당 단말에서 동작중인 UI를 확장형 원격 사용자 인터페이스를 지원하기 위해 UI 변환 엔진과의 통신을 담당하는 모듈로 원격 UI 전달 기능과 원격 UI 수신 기능으로 나뉜다. 이는 동작중인 UI정보와 수신 단말 속성 정보, 그리고 전달해야 하는 UI Handler의 네트워크 정보 등을 전달하여 UI 변환 엔진이 해당 단말 속성을 기반으로 UI를 변환하여 수신 UI 핸들러에 전달하도록 한다. 수신 UI 핸들러에서는 UI 변환엔진으로부터 변환된 원격 UI를 수신하여 해당 렌더러에 전달하는 기능을 담당한다. 미디어 핸들러 기능은 미디어 전달 기능과 수신 기능을 제공하기 위하여 미디어 핸들러로 구성된다. 미디어 핸들러는 미디어 콘텐츠를 미디어 트랜스코더로 전달하거나 수신하는 기능을 수행한다.

## 2.2 UI 변환 엔진

UI 변환 엔진은 단말의 성능 및 프로파일 정보, 네트워크 상황 등을 고려하여 자동적으로 UI를 단말의 상황에 맞게 변화하는 역할을 수행하며, 하부 컴포넌트로 프로파일 관리자(Profile Manager), 변환 정책 관리자(Transformation Policy Manager)를 가진다. UI 변환 엔진은 프로파일 관리자와 변환 정책 관리자를 통하여 단말 프로파일, 사용자 환경, 상황에 따라 적응적으로 UI를 변환하고 변환된 UI를 전달하는 기능을 제공한다. UI Repository에 등록된 UI와 UI가 수행될 단말의 특성을 프로파일 관리자로부터 제공 받아 변환 정책 관리자를 통해 전달된 단말의 변환 정책을 기반으로 타겟 단말에 맞도록 UI를 변환하는 역할을 수행한다. 그림 3은 단말의 특성에 맞게 적응적으로 UI 변

환 기능을 담당하는 UI 변환 엔진의 소프트웨어 구조도이다.

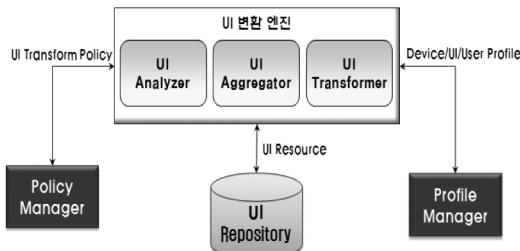


그림 3. UI 변환 엔진 구조

UIFM(??) 소프트웨어의 기능 구성은 UI 분석 기능, UI Aggregator 기능, UI 변환 기능으로 구성된다. UI 분석 기능에서는 디바이스 성능 요소와 외부 환경적 요소를 UI에 포함된 메타데이터와 비교하여 UI를 분석한다. 디바이스 성능 요소로는 출력장치 요소, 입력 장치 요소, 프로세싱능력, 단말기의 네트워크 요소 등이 고려될 수 있으며, 외부 환경적 요소로는 단말 위치, 시청환경, 사용자의 취향 등이 고려될 수 있다. UI 분석 기능에서는 프로파일 관리자, IODM(Input/Output Device Manager ??)과 연계하여 단말의 성능 요소 및 상태 정보를 획득하게 된다. UI Aggregator 기능은 UI Repository에 저장되어 있는 UI Resource들의 경로에 대해서 UI 변환 기능에서 접근할 수 있는 동적 패스(Dynamic Path)를 제공하는 기능을 담당한다. 또한, UI Repository는 원격 서버 또는 택내 서버에 위치할 수 있다. UI 변환 기능은 UI Aggregator를 통해서 전달 받은 동적 패스를 통해 UI Resource를 로드하고 이를 변환하여 타겟 단말의 UI Renderer에게 전달하는 역할을 수행한다. 따라서 UI 변환 기능은 변환 정책 관리자와 연계하여 대상 단말의 특성에 맞게 UI를 변환하고 전달하는 기능을 수행한다.

### 2.3 Virtual IO 디바이스 관리자

IO 디바이스 관리자(이하 IODM)는 홈 네트워크에 연결되어 있는 스마트 폰, 태블릿

PC, 스마트 TV 등을 에이전트 역할을 하는 안드로이드 플랫폼 기반의 애플리케이션(앱)을 설치하여 홈서버에서 IO 디바이스 속성 정보, 검색, 제어 등의 기능을 처리할 수 있게 한다.

홈 네트워크에 연결되어 있는 홈서버는 스마트 폰, 태블릿 PC, 스마트 TV 등에 설치되어 있는 애플리케이션(앱)을 통하여 IO 디바이스 속성 정보 추출, IO 디바이스 탐색, IO 디바이스 검색 등의 기능을 처리하며 IO 디바이스 속성 정보와 분류 정보를 관리하는 저장소를 가지고 있는 구조이다. 그림 4는 IO 디바이스를 관리하는 IODM (Input/Output Device Manager ??)의 소프트웨어 구조도이다.

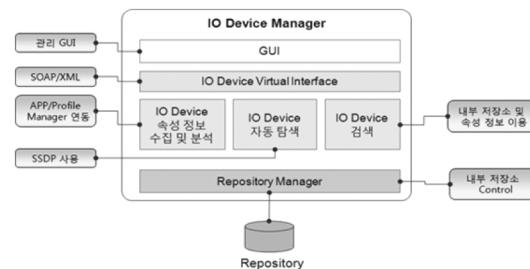


그림 4. IODM 소프트웨어 구조도

IODM 소프트웨어의 기능 구성은 IO 디바이스 저장소 관리 기능, IO 디바이스 검색 기능, IO 디바이스 속성 표현 및 정보 수집 기능, IO 디바이스 자동 탐색 기능, IO 디바이스 추상화 인터페이스 기능으로 구성된다.

- IO 디바이스 저장소 관리 기능은 IO 디바이스의 분류 및 기능(Function) 정보와 IO 디바이스 속성 정보를 저장, 조회, 수정, 삭제 등의 정보 관리 역할을 담당한다. IO 디바이스 검색 기능은 IO 디바이스 관리 서버로부터 스마트홈에 연결된 IO 디바이스에 대하여 IO 디바이스의 분류 정보 및 기능과 속성 정보를 검색하는 역할을 담당한다. IO 디바이스 속성 표현 및 정보 수집 기능은 스마트홈에 연결된 IO 디바이스는 IO 디바이스 관리 서버에 IO 디바이스 속성 정보 수집 명령을 받으면 IO 디바이스내에 에이

전트 애플리케이션이 응답하여 해당 IO 디바이스 속성 정보를 추출하여 IO 디바이스 관리 서버에 전송하는 기능을 수행한다. IO 디바이스 자동 탐색 기능은 IO 디바이스가 스마트홈에 연결되면 IO 디바이스 관리 서버에게 멀티캐스트 통신으로 자신을 알리고 IO 디바이스 관리 서버는 홈네트워크에 연결된 IO 디바이스에 유니캐스트 응답을 하여 정보를 주고받는 자동 탐색 기능을 수행한다. IO 디바이스 접근 추상화 인터페이스는 IO 디바이스의 속성 정보 수집, 디바이스 제어 등을 위해 XML형식으로 정의된 메시지이다. 다양한 IO 디바이스에 적용하기 위해 표준화된 XML 기반으로 인터페이스를 정의한다.

#### 2.4 미디어 트랜스코더

미디어 트랜스코더는 트랜스코딩을 담당하는 미디어 변환 기능과 미디어 스트리밍을 처리하는 미디어 세션 관리 기능으로 구성된다. 미디어 변환 기능은 외부 연동을 담당하는 부분과 다양한 입력 스트림을 처리하는 인터페이스 부분, 그리고 트랜스코딩을 수행하는 부분으로 구성된다. 또한, 미디어 세션 관리 기능은 서비스 요청의 연동 부분과 트랜스코딩된 스트림을 담당하는 미디어 처리 부분, 그리고 스트림을 처리하는 부분으로 구성된다. 그림 5는 미디어 변환과 세션 관리 기능을 담당하는 미디어 트랜스코더의 소프트웨어 구조를 보여준다.

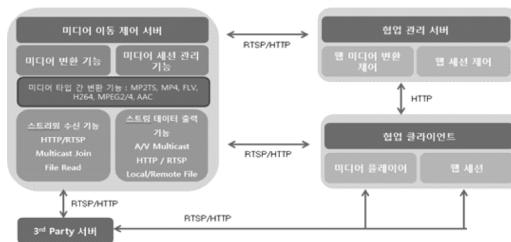


그림 5. 미디어 트랜스코더 구조도

미디어 트랜스코더의 기능 구성은 다양한

입력 소스로부터 미디어 데이터를 수신하여 지정된 미디어 포맷으로 변환하는 미디어 변환 기능, 변환된 포맷을 다양한 출력 프로토콜에 맞추어 출력하고 다양한 단말에서 미디어 서비스 수행을 관리하는 미디어 세션 관리 기능으로 구성된다. 미디어 변환 기능은 변환 하고자 하는 미디어와 서비스에 사용될 목적 미디어에 대한 정보를 이용하여 미디어를 변환한다. 소스로부터 입력된 미디어 콘텐츠를 타겟 단말 환경(해상도, 코덱, 네트워크 상황 등)에 적합한 형태로 변형과 가공을 하는 트랜스코딩 기능을 수행한다. 목적 미디어는 지원을 요하는 단말 기 종류와 서비스 종류에 따라 미디어의 프로파일을 미리 정해 둔 것을 이용할 수 있다. 미디어 세션 관리 기능은 변환 중이거나 변환이 완료된 미디어를 포함하여 미디어 스트리밍 서비스의 세션을 관리하기 위한 것이다. 다양한 단말기를 수용하기 위하여, 다양한 미디어 타입과 미디어 정보 그리고, 미디어 서비스를 받기 위한 프로토콜의 지원이 필요하다. 미디어 서비스 세션은 단말기 변화에 따른 파일 혹은 프로토콜이 변경될 수 있다.

### 3. 결론

본 논문에서는 다양한 스마트 디바이스 간에 동일한 Look-and-Feel을 유지함으로써 서비스의 단일성을 추구하기 위한 핵심 기술로 N-스크린 서비스를 위한 원격 사용자 인터페이스를 소개하였다. 이는 사용자 인터페이스 생성 과정을 자동화 혹은 간소화 시킬 수 있으며, 스마트TV 서비스의 복합형 서비스로서 기존 리모컨의 제약을 극복하면서도 사용하기 편리한 사용자 인터페이스를 제공한다.

이러한 목적을 달성하기 위해, N-스크린 서비스를 위한 원격 사용자 인터페이스 기술은 단말 성능에 최적화된 동적 UI 표현을

위해 Remote UI 프레임워크와 UI 변환 엔진을 핵심기술로 가지며 주변의 IO 장치로 입출력 수단을 제공하기 위한 Virtual IO 디바이스 관리자와 미디어 변환 기능과 미디어 스트리밍을 처리하는 미디어 세션 관리 기능을 가지는 미디어 트랜스코더로 구성된다. 이를 통해 사용자가 어떤 단말을 사용하던지 ‘동일한 체험’(homogeneous experience)을 제공하며, 동기화된 방송 콘텐츠와 웹 콘텐츠가 단말기, 시간, 장소와 무관하게 소비될 수 있다.

N-스크린 서비스를 위한 원격 사용자 인터페이스 기술은 플랫폼 기술과 함께 스마트TV 기반의 새로운 시장을 창출해 낼 수 있을 것이다. 또한 N-스크린 서비스용 스마트 콘텐츠 개발과 보급화를 통해서 관련 콘텐츠 개발자, 관련 기업 육성 및 일자리 창출효과 등을 기대할 수 있는 선순환적 생태계 구성을 기여할 수 있을 것이다.

## 연구지원

.※ 본 연구는 지식경제부의 산업융합원천기술개발사업의 일환으로 수행하였습니다. [10039202, N-스크린 서비스를 위한 스마트TV용 디바이스 협업 오픈 미들웨어 및 원격 사용자 인터페이스 기술 개발]

## 참고문헌

1. <http://www.naver.com>
2. <http://www.realvnc.com/docs/rfbproto.pdf>
3. Juha Leppilahti, “Remote UI protocols for home environment”, TKK T0119.5190 Seminar on Internetworking
4. User Interface Markup Language (UIML), <http://docs.oasis-open.org/uiml/v4.0/cd01/uiml-4.0-cd01.pdf>
5. Scalable Vector Graphics, [http://en.wikipedia.org/wiki/Scalable\\_Vector\\_Graphics](http://en.wikipedia.org/wiki/Scalable_Vector_Graphics)