
OSGi 기반 홈게이트웨이 시스템의 설계 및 구현

김광현*

Design and Implementation of Home Gateway System based on OSGi

Gwang-hyun Kim*

요 약

임베디드 리눅스 시스템을 이용하여 OSGi 기반 원격 번들제어 서비스 및 방문자 확인을 위한 웹 카메라 제어를 통한 방문자의 신원을 확인할 수 있는 응용 시스템을 설계 및 구현하였다. 구현된 시스템은 임베디드 시스템에 리눅스와 자바를 포팅하여 오픈 Oscar 프레임워크를 설계하였고, 근거리 통신망을 통해 웹 카메라를 제어할 수 있도록 구현하였다. 또한, 구현 시스템에서 시리얼은 RFID 리더기의 데이터를 수신받아 원격지에서 웹을 통해 방문자의 정보를 확인할 수 있으며, 맥내에서는 무선으로 인터넷을 할 수 있는 시스템을 설계하였다. 본 논문에서는 기본적으로 각 장치의 관리 및 모니터링을 구현할 때 OSGi 기반의 통합된 서비스 제공을 통해 효율적인 홈네트워크 시스템 개발 및 다양한 응용 서비스의 개발 기간을 단축할 수 있음을 알 수 있었다.

ABSTRACT

We implemented an application system which is able to verify visitor identification for remote bundle control based on OSGi and web camera control using embedded linux system. This system are designed open Oscar framework by porting java on the embedded linux system and implemented web camera system in local area networks. Also This system can verify information of visitor from receiving the data of RFID reader and designed the wireless internet system in home. Basically, in this paper we expect that home network system development and the period of application service development are reduced with integrated service provision based on OSGi.

키워드

임베디드 리눅스, OSGi, RFID, 홈네트워크, 번들제어

I. 서론

OSGi(Open Service Gateway initiative)는 가전제품이나 보안시스템 및 모바일 등의 장치들을 인터넷에 접속하는 표준방식에 관한 산업체의 규격이다[1]. 이런 표준화를 통해 일반 가정의 사용자들은 시스템

을 설치할 때 새로운 배선이나 장치를 설치하지 않고서도 기존의 모니터링 서비스로부터 또 다른 시스템으로 변경할 수 있도록 하였다. 이러한 서비스 게이트웨이는 가정이나 기업용 주변기기 네트워크와 인터넷 간의 단순 게이트웨이를 벗어나 컴퓨터 내에 있는 하나의 애플리케이션 서버가 될 수 있다. 또한 OSGi는

* 광주대학교 정보통신학과(ghkim@gwangju.ac.kr)

접수일자 : 2009. 12. 15

심사완료일자 : 2010. 2. 10

서비스 제공업체와 가정 및 소규모 기업망내에 있는 장치들 사이에서 통신 및 제어를 할 수 있게 하기 위한 프로그래머용 API를 지정하였고, 이 API(Application Programming Interface)는 자바로 만들어져 있기 때문에 일반적으로 어떠한 운영체계를 가진 플랫폼에서도 동작이 가능하다[1,2]. OSGi는 개방형 표준 프로그래밍 인터페이스이므로 각종 변경사항들은 자바 커뮤니티 프로세스를 통해 진화될 것이다. 기존의 홈 네트워크에서 이미 사용 중 이거나 새로운 장치를 연결 및 제거할 때의 어려운 상황, 제한적인 시스템을 효율적으로 관리하기 위하여 OSGi의 기반으로 홈 게이트웨이 시스템을 제안하고자 한다.

본 논문에서는 원격지에서 게이트웨이의 상태를 관리하며, 방문자의 정보를 확인하고 제어할 수 있도록 구현하였다. 2장에서는 OSGi관련 연구에 관하여 기술하며, 3장에서는 임베디드 시스템에 리눅스와 자바 가상머신의 포팅 기술내용을 다루고, 오픈 프레임워크 Oscar를 이용한 서버구현과 웹 카메라 연동 부분을 설계하였다. 또한 RFID 태그를 이용하여 방문자의 신분을 확인하기 위한 시스템 구성환경과 그에 따른 번들 설계에 관하여 설명하고, 구현된 시스템의 동작과정을 기술한다. 마지막으로 4장에서는 결론 및 향후 연구 방향에 대하여 기술 한다.

II. OSGi 개요 및 연구 동향

2.1. OSGi 개요

초기의 OSGi R1.0에서는 기초적인 정보기기의 연동과 상태 모니터링의 기능을 탑재했으며 표준화에 주력했고 2.0에서는 운영과 관리, 보안의 기능이 추가되었다. 본격적인 콘텐츠 서비스 플랫폼으로의 모습은 R3.0에 이르러 나타나게 된다. 외부의 오픈소스를 이용하여 처리하던 XML Parsing, Wire Admin, URL Handler 등의 기능이 표준 서비스로 채택되어 탑재되고, UPnP와 JINI 표준이 기본 시스템 서비스로 올라오면서 OSGi는 명시상부한 모바일, 임베디드, 데스크탑 애플리케이션, 클라이언트/서버 환경에서 미들웨어 프레임워크로 자리매김하게 된다. R4.0에서는 분야가 세분화되어 카테고리별로 디바이스 특성에 맞는 컨텐

츠, 시스템 서비스들이 발전하게 된다. 특히 휴대폰, 스마트홈 컨트롤러, 빌딩 자동화 컨트롤러, 자동차에 탑재되는 네비게이션과 텔레매틱스 솔루션 등의 폭발적인 성장에 힘입어 모바일, 임베디드 시스템에 대한 많은 요구 사항들이 직접 기본 서비스에 채택되고 탑재되게 되었다[3]. 여기서 나오는 시스템 기본 서비스는 OSGi Alliance에서 승인 및 채택되어 기본 OSGi Framework에 탑재되어 있다.

2.2. OSGi 주요 특징

OSGi는 자바기반의 컴포넌트 구조로 설계되어 있으며 자바 런타임 환경하에서(J2ME, J2SE, J2EE) 작동하게 만들어진 표준이다. 또한 OSGi는 서비스 기반의 구조를 지향하며 서비스는 모두 번들(Bundle)이라는 물리적 묶음에 포함된다. 복수개의 OSGi 서비스가 하나의 번들에 포함될 수도 있으며, 번들은 배포와 에 쟈 기본 단위를 형성한다. 이러한 번들을 에 해주는 것이 바로 프레임워크(Framework)이다. 그림 1은 OSGi의 플랫폼의 구성 정보이다.

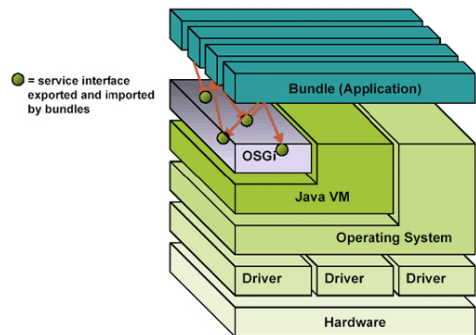


그림 1. OSGi 플랫폼 구성 정보
Fig. 1 OSGi Platform Configuration

OSGi는 원격관리를 지원하는데 OSGi는 번들 단위로 서비스를 형성하고 운영되며, 번들을 업데이트하거나 업그레이드를 원격에서 관리하고 제어할 수 있다. 이를 위해서 OSGi는 원격관리표준 프로토콜을 제정하였으며 OSGi가 탑재되고 운영되는 환경에 맞게 사용하면 된다. 많은 자바 애플리케이션 서버 환경에서 구동하는 자바 애플리케이션들은 독립성을 보장하기 위해서 극히 폐쇄적인 컨테이너 환경에서 작동된다.

OSGi구조는 다음과 같이 몇 개의 계층으로 구성되어 있다. OSGi는 우선 자바 런타임 환경에서 구동된다. 자바 런타임은 하드웨어 환경에 의해서 J2ME(CD C/CLDC), J2SE, J2EE로 구성되며, 하나의 VM에서 서로 다른 복수개의 클래스 로더들에 의해서 각각의 OSGi 애플리케이션이 작동된다. OSGi는 단 하나의 VM 인스턴스위에서 동작하고, 복수개의 클래스 로더를 수행하여 독립된 이름 공간을 갖게 된다. 또한 번들은 동적 라이프 사이클을(dynamic Install, Start, Stop, Delete, Update) 갖고 수행하며 성능을 위해서 여러 리소스와 인터페이스를 지원한다. 자바 애플릿처럼 서버에서 다운로드하는 형식이 아니라 로컬 디바이스에 상주하는 것도 하나의 특징이다. 그림 2는 번들의 동적 라이프 사이클을 나타내고 있다.

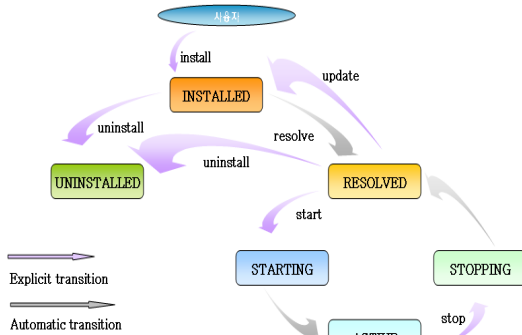


그림 2. 번들의 라이프 사이클
Fig. 2 Bundle Dynamic Life Cycle

OSGi 서비스는 자바 오브젝트로서 하나의 번들에 의해서 등록되며 다른 번들에 의해서 사용되거나 복수개의 번들이 연동/통합하여 독자적인 서비스를 만들어내기도 한다. 명시된 형태는 Java Interface 스타일로 표현되어, 서비스의 스펙과 구현이 완벽하게 분리되어 사용된다. 이렇게 되면, 동일한 서비스 기능을 갖고도 서로 다른 번들이 서로 다른 구현을 서비스 레지스트리에 등록할 수 있게 된다. OSGi의 기본적인 개념은 하루가 다르게 탄생하는 새로운 플랫폼과 표준 그리고 프레임워크를 볼 때에 1999년부터 꾸준히 발전해온 OSGi는 분명 독특한 장점이 있기에 계속해서 발전하고, 홈네트워크용 서비스 플랫폼이나 모바일/임베디드 프레임워크에서 점점 더 영향력을 확대

해 나갈 것으로 본다. 특히 이클립스 플랫폼에 기본 탑재되고 RCP로의 확장에 중요한 요소로 지목되는 것은 OSGi가 단순한 미들웨어나 프레임워크가 아니라 라는 것을 증명해주는 좋은 예가 될 것이다.

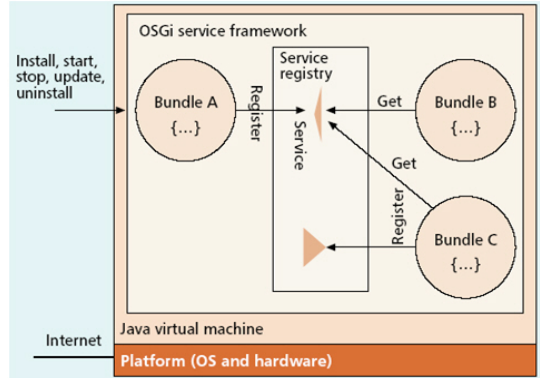


그림 3. 번들과 서비스 생성
Fig. 3 Bundle and Service Creation

2.3. OSGi 연구 동향

OSGi는 서비스가 작동하고 운영되는 서비스 환경에 관한 표준 기술이다. 과거 OSGi R1.0이 발표될 때만 하더라도 OSGi는 홈 네트워크의 네트워크 표준에 국한된 부분이 많았으나, 현재 4.0에 이르러서는 기존의 홈네트워크 외에 모바일, 임베디드, 텔레매틱스, PC 애플리케이션 (Eclipse 기반의 RCP)는 물론 엔터프라이즈 환경의 프레임워크에까지 확장하고 있다. 그만큼 기존의 미들웨어에 비해서 많은 장점을 가지고있다. 또한 과거 초기 단계의 홈 네트워크 및 빌딩 제어, 자동화 시장에서 디바이스 제어 기술이 주로 사용되고 있지만 현재는 그보다 훨씬 다양한 디바이스간의 상호작용을 요구하게 되고 더 나아가 컨텐츠 및 솔루션 서비스가 점차적으로 고도화되어 질수록 OSGi와 같은 서비스 플랫폼 환경이 꼭 필요하게 될 것이다. 현재 다양한 디바이스들 간의 상호운용성을 위해서 UPnP, HAVi, JINI같은 표준이 만들어지고 주로 장비의 제어나 데이터 전달 등을 처리하게 되므로 OSGi 기술과 상호 보완적인 위치에 있는 셈이다. OSGi를 사용하거나 구현하는 쪽에서는 이러한 미들웨어의 지원이 또 다른 고려 사항임에 유의해야 한다[4,7].

III. OSGi 기반 홈게이트웨이 시스템 구현

3.1. 홈 게이트웨이 시스템

게이트웨이의 운영체제는 리눅스기반 운영체제인 OpenWrt를 사용하고 있다. OpenWrt는 GPL라이선스로 이루어지고 있는 프로젝트이며, 커널 2.4.x버전을 사용하며 라우터, 게이트웨이 기능을 하기 위해 개발된 리눅스이다. 전세계에서 많은 개발자들의 참여로 상용 운영체제 수준의 기능을 가지고 있으며 GPL 라이선스를 따르고 있기에 라이선스 비용을 지불하지 않아도 된다. 또한 OpenWrt는 상용 운영체제와 달리 기능의 추가/삭제가 편리하다. 상용 운영체제는 게이트웨이의 기능을 업그레이드 또는 추가를 위해 별도의 펌웨어(Firmware)를 해야한다. 업체에서 패키지를 해서 배포하기 때문에 설치의 간편하지만 최신기술 및 사용자 운별도로 원하는 기능은 사용할 수 없다. OpenWrt는 초기 설치의 별도의 펌웨어를 거치지 않아도 된다. 이는 IPKG(Itsy Package Management System)을 통한 패키지 관리를 하기 때문이며 IPKG로 인해 패키지의 추가/삭제가 매우 편리하다. 많은 개발자들이 최신기술을 적용한 패키지를 배포하고 있으며 이용자는 빠르게 최신기술을 게이트웨이에 접목시킬 수가 있다. 게이트웨이의 하드웨어 구성은 Broadcom사의 MIPS 프로세서 기반 Broadcom 4704 Chip을 사용하였다. Broadcom 4704는 264Mhz Clock Speed를 사용하며 메모리 구성은 Nand Flash 8M 바이트와 SDRam 32M바이트로 구성되어 있다. Ethernet은 10/100Mbps를 지원하며 Wireless는 802.11g와 802.11b 두가지를 지원하여 폭넓은 무선인터넷 환경을 제공한다. 사용가능한 인터넷 포트는 1개의 WAN와 4개의 LAN으로 구성되어 있다.

그림 4는 홈게이트웨이의 구조를 나타내는 Block Diagram이다. 현재 테스트용으로 개발된 제품이며 추가적인 기능(웹서버, 데이터베이스)을 수행하기엔 다소 부족한 시스템 사양을 가지고 있다. BCM4704 Chip 중심으로 각종컨트롤러와 실제 사용가능한 Ethernet Port, WIFI용 무선안테나 연결 포트가 있는 것을 볼 수 있다. 테스트를 위한 리셋 스위치, 아래쪽에 위치한 Serial Port, USB Port을 사용가능하도록 소켓홀을 열어 두었다.

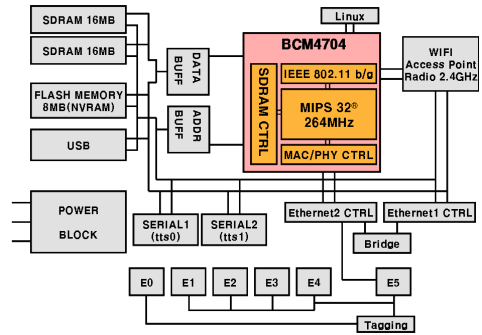


그림 4. 게이트웨이 블록 다이어그램
Fig. 4 Gateway Block Diagram

3.2. OSGi 기반 홈게이트웨이 프레임워크 개발

OSGi Framework가 동작하기 위해서는 JAVA의 설치가 필요하다. JAVA는 운영체제에 상관없이 가상머신이 설치가 되어있는 환경에서 클래스 실행을 할 수 있다. 게이트웨이 프로세서(MIPS) 환경에 맞게 포팅 해야 한다. 게이트웨이 환경(MIPS)에서는 컴파일 이 불가능하기 때문에 호스트 PC에서 빌드를 실시하였고, Mika 플랫폼을 이용해 포팅 가능 여부를 확인하였다. 그림 5는 Mika(JAVA)의 구성도이다. 게이트웨이의 Flash Memory가 8M 바이트로 한정되어 있기 때문에 확장된 USB를 이용하여 /usr/bin, /usr/lib의 경로에 심볼릭 링크를 걸었다. 또한 게이트웨이에서 구축하기 위해서 텍스트 모드로 구성했으며, 호스트 PC에 설치 후 Oscar구성 파일들을 USB에 복사하였다.

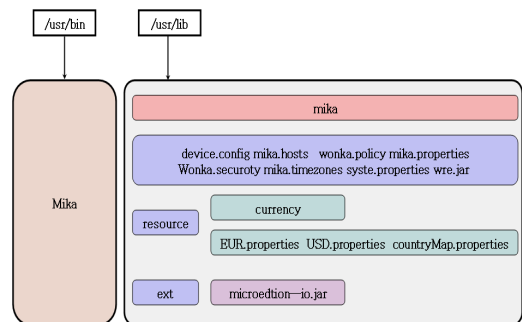


그림 5. 자바(Mika) Diagram
Fig. 5 Java Mika Diagram

3.3. 웹기반 카메라 시스템 개발

그림 6은 게이트웨이와 카메라간 연결 제어 및 호스트에게 어떻게 제공을 하는지에 대한 전달과정을 표현한 흐름도 이다. 카메라 또한 하나의 임베디드 장치로 상호간 연결 매체는 RJ-45로 유선 LAN 환경이다. 카메라는 자신의 카메라 데이터를 게이트웨이에 전달하고 게이트웨이는 외부 호스트들에게 화면을 제공하는 단계를 거친다. 또한 카메라는 자신의 카메라의 제어를 API로 제공하여 웹 환경에서 컨트롤 할 수 있는 환경을 제공한다

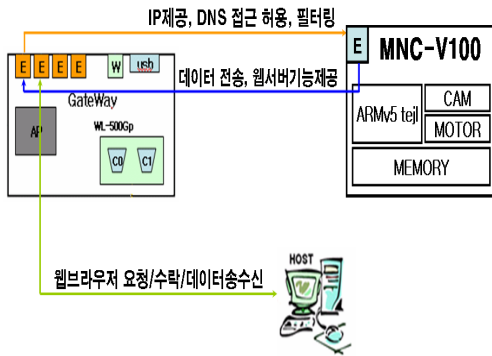


그림 6. 게이트웨이와 카메라 간 흐름도
Fig. 6 Flow Chart between gateway and Camera

게이트웨이는 연결된 카메라에게 IPv4 주소를 부여 해주며 DNS 및 외부 접근제한을 위한 방화벽 및 연결을 위한 카메라 고유 포트를 포워딩한다. 외부에서 카메라 사용 요구에 대해 확인을 하고 카메라와 연결을 하며 카메라에 대한 관리 서버를 제공하고 있다. 구현된 시스템 카메라는 단순 인코딩 기능을 가진 카메라가 아닌 네트워크로 데이터를 주고 받을 수 있는 장치이다. 실험에 사용된 카메라는 Arm5 기반의 비디오 인코딩 기능을 가졌으며 메모리는 SDram 32M 바이트와 Flash 32M 바이트로 구성되어 있다. 최대 사용가능 해상도는 640X480 pixel이며 스텝 모터를 이용하여 상하좌우 회전이 가능하며 저조도 방지를 위한 자체 라이트가 내장 되어 있다. 카메라는 게이트웨이보다 많은 메모리를 보유하고 있다. 이는 인코딩 단계에서 사용되는 메모리가 상당히 크기 때문이다. MPEG 기반으로 동영상 촬영하며 간략한 동영상

및 스크린 캡처를 위해 저장 공간이 필요하기도 하며 스트림 기반으로 데이터가 전달되는 것이 아니기 때문에 인코딩이 끝나면 데이터를 가지고 있어야할 공간이 필요한 것이다. 게이트웨이는 자체 인코딩 기능이 없기 때문에 카메라에서 인코딩이 된 데이터를 LAN을 통해 받아오게 된다. 받은 데이터는 요청이 있는 호스트들에게 재전송이 된다. 호스트는 게이트웨이 외부 네트워크를 통해 접근한 호스트가 될 수 있으며 내부에서 접근하는 호스트 또는 무선을 통해 접근하는 호스트들도 게이트웨이를 통해 카메라를 제어할 수 있으며 화면 데이터를 받을 수 있다.

3.4. RFID 시스템 개발

RFID 리더기는 주파수는 13.56MHz의 대역을 이용하며 시리얼(RS-232) 인터페이스 및 태그로 구성되어 있고, RFID는 게이트웨이 간 시리얼로 연결되어 있다. 태그접촉을 통해 태그정보 데이터를 게이트웨이 및 웹으로 전송하기 위해서 서버/클라이언트 형태로 제작되었으며, 원격지에서 방문자의 정보를 확인하기 위해서는 웹을 이용하기 때문에 태그의 정보는 공유메모리를 통해 애플릿으로 전달된다. 그림 7은 시스템의 전체 구성도를 나타내고 있다.

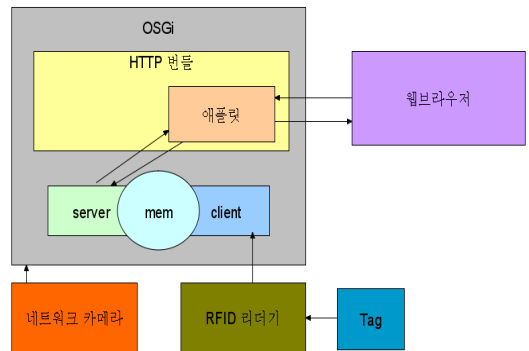


그림 7. RFID 시스템 구성도
Fig. 7 RFID System Configuration

3-5 번들(Bundle) 제작

OBR(Oscar Bundle Repository)에서 제공하는 Servlet와 HTTP Service(HTTP Server), HTTP Admin

(Oscar Bundle Control)을 이용하여 웹페이지를 구성하였다. OSGi Framework 의 HTTP Service는 경량의 HTTP Server 이다. 그림 8과 같이 이러한 서비스의 제공은 현재 가장 많이 쓰이고 있는 front-end user interface인 웹 브라우저를 통해서 편의성을 제공하고 있고, OSGi 서비스 플랫폼은 보통 인터넷이나 다른 네트워크 상에서 서비스에 접근하는 것을 사용자에게 제공하고 Http Service는 Servlet과 Resources의 구성을 통해 제공된다.

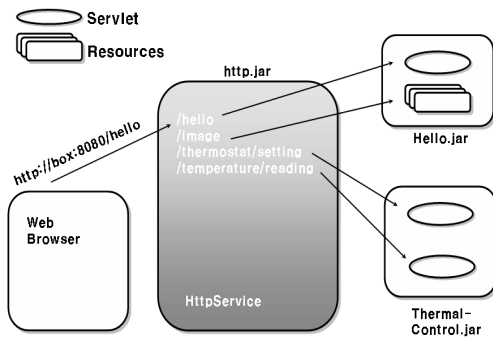


그림 8. 웹서비스 의한 서블릿 등록과 매핑
Fig. 8 Servlet/resource registration and mapping by HttpService

그림 8과 같은 HTTP Service 구조를 이용하여 그림 9와 같은 실제 번들이 작성된다.

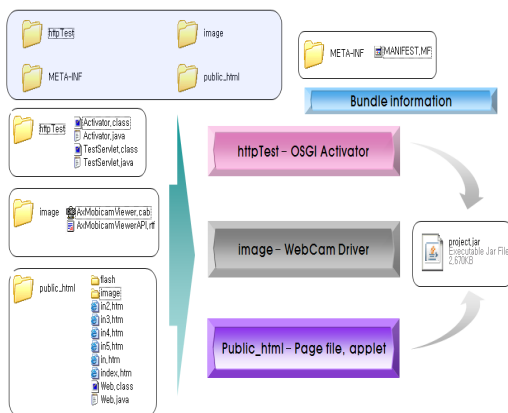


그림 9. 번들의 구조
Fig. 9 Bundle Structure

3.6. 구현 시스템의 동작 실험

평상시 방문자가 없을 때 그림 10에서 ①의 화면처럼 태그정보가 없으면 그에 따라 정보가 NULL로 표시가 되고, ②는 방문자가 태그를 접촉했을 때 방문자의 정보를 애플릿으로 구성된 화면에 나타내주고 있다.

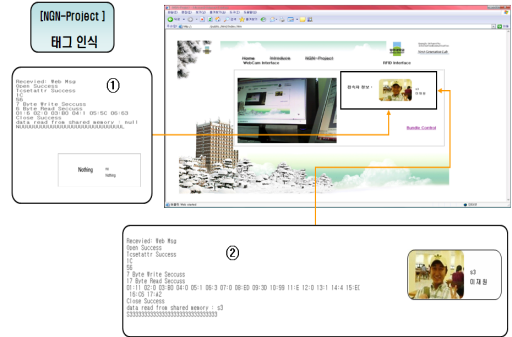


그림 10. 태그 인식 화면
Fig. 10 Tag Recognition Screen

방문자가 있을 경우 그림 11에서 처럼 모션을 체크하여 동영상으로 저장하거나 사진으로 저장할 수 있고, 주위가 어두우면 플래쉬를 작동시켜 신원을 확인할 수 있으며 마이크를 통해 대화를 할 수 있도록 되어 있다. 원격지에서 접속하여 현재 현관(카메라)주변 상황에 대하여 관찰할 수 있도록 좌/우, 상/하로 카메라의 조사가 가능하다.



그림 11. 카메라 제어 화면
Fig. 11 Camera Control Screen

그림 12는 웹을 통해 OSGi 번들의 상태를 확인하고 시작/중지/업데이트/제거를 할 수 있도록 구성된 상태를 볼 수 있으며 손쉽게 번들의 제어가 가능하다.



그림 12. 번들 제어 화면
Fig. 12 Bundle Control Screen

IV. 결론

최근 홈 네트워크 시장은 광대역 서비스의 확대 및 정보단말의 보급을 통해 서로 다른 디지털 정보기기 및 정보가전들이 가지고 있는 데이터 및 대역폭의 공유를 통해 시장이 급격하게 확대되고 있다. 이는 사용자에게 음성·영상·데이터 등의 통합 서비스를 제공하고, 다른 장치들 간에 통신이 가능하도록 하는 기능이 요구되고 있다. 본 논문에서는 임베디드 리눅스 시스템을 이용하여 OSGi 기반 원격 번들제어 서비스 및 방문자 확인을 위한 웹 카메라 제어를 통한 방문자의 신원을 확인할 수 있는 응용 시스템을 설계 및 구현하였다. 이는 홈네트워크 장치의 관리 및 모니터링을 구현할 때 효율적인 홈네트워크 시스템 개발 및 다양한 응용 서비스의 개발 기간을 단축할 수 있음을 알 수 있었다.

향후에는 가입자망의 고도화됨에 따라 그에 적합한 각종 콘텐츠와 응용서비스 개발이 동시에 진행되어야 할 것이다. 또한 향후 연구과제로는 OSGi 번들을 이용해 센서(온도, 습도, 조도 등)장치들을 활용하여 맥내 환경에서 통합 서비스 환경 구축이 필요하다.

감사의 글

이 연구는 2009년도 광주대학교 대학 연구비의 지원을 받아 수행되었음

참고 문헌

- [1] <http://www.osgi.org/Main/HomePage>
- [2] http://mobicam.co.kr/home/mobicam_product_v100.html
- [3] 김석우, “유니버설 미들웨어 OSGi 최신기술 동향”, [IITA] 정보통신연구진흥원, 1320호, 지능형 홈네트워크, pp.24-37, 2007.
- [4] 안중현, 김대영, 이종연, 조국현, “Oscar bundle 을 이용한 OSGi 기반 홈 네트워크 관리 플랫폼 설계”, 한국정보과학회 가을학술발표논문집, Vol.31, No.2, pp.157-159, 2004.
- [5] 최병욱, 고경철, 문전일, 임계영, “임베디드 리눅스”, 홍릉과학출판사, pp.151-191, 2002.
- [6] 진준근, 김현수, “모바일 환경용 권피규레이션 API 구현 및 테스트”, 한국정보과학회 가을 학술발표논문집, Vol.30, No.2, pp.394-396. 2003.
- [7] 서대영, 이해준, 강동현, OSGi Programming 실무, 토비스출판, 제13장, pp.177-187, 2006.
- [8] 전호인, 신용섭, “홈 네트워킹 기술 및 표준화 동향”, 전자공학회지, 제29권, 6호, pp.638-660, 2002.

저자 소개



김광현(Gwang-hyun Kim)

1989년 2월 광주대학교

전자계산학과 학사

1991년 2월 광주대학교

전자계산학과 석사

1997년 2월 광주대학교 전자계산학과 박사

2001년 8월 ~ 2002년 7월

Pennsylvania State Univ. Post-Doc

1997년 3월 ~ 현재 광주대학교 정보통신학과 교수

※ 주 관심분야 : 차세대인터넷, 인터넷 QoS, 네트워크 관리, 센서네트워크