

<http://dx.doi.org/10.7236/JIIBC.2016.16.1.77>

JIIBC 2016-1-10

# 웹 기반의 IETF CoAP 프로토콜을 이용한 실내 IoT 자원 제어 서비스 설계 및 구현

## A Design and Implementation of Indoor IoT Resource Control Service using Web-based IETF CoAP Protocol

김문권\*, 김도현\*\*

Wenquan Jin, Do-Hyeun Kim

**요 약** 최근 모든 사물을 인터넷에 연결하는 IoT(Internet of Things)에서 사물 간의 데이터를 전송하는 통신 프로토콜에 대한 표준화가 진행되고 있다. 본 논문에서는 웹 기반의 IETF(Internet Engineering Task Force) CoAP(Constrained Application Protocol)을 이용한 실내 IoT 자원 제어 서비스를 설계하고 구현한다. 이를 위해 웹 서버에서 프록시(proxy)와 RD(Resource Directory) 기능을 갖는 웹 기반의 실내 IoT 자원 제어 구조를 제시한다. 실내 IoT 자원 제어 서비스에서는 CoAP 프로토콜을 통해 실내 IoT 자원을 웹 서버에 등록하고, 프록시의 HTTP과 CoAP메시지 변환 기능을 이용하여 인터넷 상에서 웹 클라이언트와 IoT 자원 간 통신을 지원하고, 사용자가 웹 브라우저를 이용하여 실내 환경을 모니터링하고 제어할 수 있도록 한다.

**Abstract** Recently, an IoT(Internet of Things) application communication protocol is standardizing for connectivity between every things on Internet. In this paper, we design and implement an indoor resource control service using IETF (Internet Engineering Task Force) CoAP (Constrained Application Protocol) based on Web . We present an indoor resource control architecture based on Web included functionalities of proxy and RD (Resource Directory) in a web server. Developed indoor resource control service supports to register low-powered and small-scale IoT nodes to web server using CoAP. This service allows users to control the indoor resources through a web browser using Web proxy with functionality of HTTP-CoAP converting.

**Key Words** : IoT, CoAP, Web Service, Control Service, Resource Directory,

### 1. 서 론

인터넷상에서 수많은 컴퓨터 간을 연결하는 링크와 웹(Web) 페이지 링크가 만들어지고, 웹을 통해 인간이 생성하고 가공한 데이터, 정보 및 지식을 공유할 수 있게 되었다. 이때 웹 페이지 접속을 위해 사용한 프로토콜이

HTTP(Hypertext Transfer Protocol)이다.

최근 국내외의 많은 연구 단체, 산업체 등에서 다양한 사물이 인터넷을 통해 연결하여 지능적인 서비스를 제공하는 IoT(Internet of Things) 연구를 활발히 진행하고 있다. IoT 기술을 이용하여 각 사물이 서비스를 제공할 수 있으며, 주변의 사물과 연동을 통하여 통합된 서비스

\*준회원, 제주대학교 컴퓨터공학과

\*\*중신회원, 제주대학교 컴퓨터공학과

접수일자: 2016년 1월 3일, 수정완료: 2016년 2월 3일

게재확정일자: 2016년 2월 5일

Received: 3 January, 2015 / Revised: 3 February, 2016 /

Accepted: 5 February, 2016

\*\*Corresponding Author: kimdh@jejunu.ac.kr

Dept. of Computer Engineering, Jeju National University, Korea

를 생성할 수 있다. 현재 IoT 기술은 스마트 홈, 스마트 공장, 스마트 시티 등에 응용되고 있다.

2010년에 IETF(Internet Engineering Task Force) CoRE(Constrained RESTful environments) 워킹그룹(Working Group)에서 CoAP 프로토콜에 대한 표준화를 진행하여 최근 RFC(Request for Comments) 7252로 채택하였다<sup>[1]</sup>. 저성능 CPU, 작은 저장장치, 저전력 등의 IoT 노드 제약 조건과, 높은 데이터 손실과 느린 데이터 전송 등의 네트워크 제약 환경에서 CoAP 프로토콜은 데이터 전송하는 응용 프로토콜이다. 현재 oneM2M, LWM2M, OIC IoTivity 등 IoT 플랫폼에 CoAP 프로토콜이 적용되고 있으며, 제한된 하드웨어 및 네트워크 자원을 갖고 있는 IoT 노드 등을 지원하고 있다.

더불어 IETF CoRE 워킹 그룹에서 CoAP 프로토콜을 이용하여 IoT 노드를 등록하고 검색하는 RD에 대한 연구가 진행되고 있다<sup>[2]</sup>.

본 논문에서는 IETF CoAP 프로토콜을 이용하여 인터넷 웹 환경에서의 실내 IoT 임베디드 자원 제어 서비스를 설계하고 구현한다. 웹 기반 실내 IoT 자원 제어 서비스에서는 웹 서버에 프록시(proxy)와 RD(Resource Directory) 기능을 제공하고, 웹 서비스, IoT 자원 등록 서비스, HTTP와 CoAP 프로토콜 변환 서비스도 제공한다. 실내 IoT 자원 제어 서비스에서는 사용자가 CoAP 프로토콜을 이용하여 실내 환경을 모니터링하고, 웹 클라이언트를 이용하여 실내 환경을 제어할 수 있다.

## II. 관련연구

최근 IoT 기반의 스마트 홈에서는 실내에서의 유선 혹은 무선 통신을 이용하여 각 가전제품 혹은 가구들을 감시하고 제어하기 위하여 하나의 표준 프로토콜을 개발하고 있다.

IETF CoRE 워킹 그룹에서는 IoT 네트워크를 구축하기 위해 CoAP 프로토콜을 표준화하고 있다. CoAP 프로토콜은 사물간의 통신을 위한 응용층의 프로토콜이며, 작은 용량의 메모리와 저전력 등 제한된 환경에서 센서나 구동체 IoT 노드 간에 통신을 지원한다. CoAP 프로토콜은 HTTP와 쉽게 상호 변환할 수 있으며, IoT와 M2M 환경에서 저전력 센서와 구동체 네트워크를 통한 기반 시설을 감시하거나 관리할 수 있다.

현재 IETF의 CoAP 프로토콜은 oneM2M, LwM2M, OIC 등 국제 IoT 표준화 단계에서 사용하고 있다. CoAP 프로토콜은 작은 패킷크기를 갖고 있고 RESTful 구조로 메시지를 전송하고 있다. 그러나 아직 CoAP 프로토콜은 기존에 많은 인터넷 응용프로그램들은 HTTP를 적용하고 있어 완전히 HTTP를 대체할 수 없다. 따라서 현재는 CoAP 프로토콜과 HTTP의 연동이 필요하다<sup>[3]</sup>. 기존에 CoAP과 HTTP 매핑 방법에는 CoAP과 HTTP와 같은 URI를 사용하여 HTTP서버에서 받은 요청을 같은 URI를 가진 CoAP서버에 전송하는 방법과 URI에 다른 프로토콜을 지원하는 서버의 URI를 URI-Query 혹은 URI-Path로 추가하여 전송하는 방법이 있다<sup>[3,4]</sup>.

CoAP 프로토콜은 HTTP와 연동을 통하여 기존의 장비 모니터링 기능을 확장하거나 효율성을 향상할 수 있다. 따라서 CoAP 프로토콜은 현실 세계의 장비들의 통신에 적용하여 보다 적은 에너지 소비와 제한된 네트워크의 적용에 가능하며, 프록시를 통하여 기존의 HTTP 기반의 인터넷 프로그램과 연동하여 사이버 세계에 현실 세계의 데이터를 제공할 수 있다.

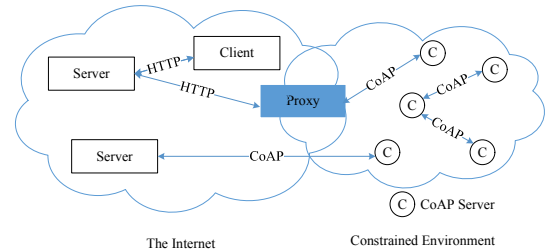


그림 1. IETF CoAP 프로토콜 환경  
Fig. 1. IETF CoAP environment

기본적으로 CoAP 프로토콜은 서버와 클라이언트 모델이다. CoAP 프로토콜은 그림 1과 같이 서버, 프록시, 클라이언트로 구성된다. IoT 노드는 기존의 인터넷 상의 컴퓨터 서버가 아니라 주위 상황 정보를 획득하는 IoT 노드를 의미한다. 그리고 CoAP 클라이언트는 일반 인터넷 상에서 상황 정보를 요청하는 사용자 단말을 말한다. IoT 노드와 클라이언트 사이에서 존재할 수 있는 CoAP 프록시는 데이터를 수신하여 메시지를 전달하는 중개 역할을 수행한다. CoAP 프록시는 HTTP 메시지를 CoAP 메시지로 상호 전환하여 사용자의 웹 클라이언트에 전송하여 상황 정보를 제공한다<sup>[5]</sup>.

IoT 네트워크에서 IoT 노드의 정보를 하나 이상의 서

버에 저장하여 기타 IoT 노드들이 서버를 통하여 IoT 노드들의 상태 및 정보를 검색한다. 검색 원리에는 집중형 검색과 분산형 검색이 있을 수 있다<sup>[6]</sup>. 집중형 검색에서는 하나의 RD가 있으며 하나의 네트워크에서 IoT 노드들의 정보는 RD에 저장된다. CoAP 클라이언트가 IoT 노드를 검색하려면 이 RD의 URI를 통하여 RD에 접근하여 검색하려는 IoT 노드의 정보를 요청할 수 있다. 분산형 검색에서는 각 IoT 노드가 자신의 정보를 갖고 있으며, CoAP 클라이언트는 반드시 검색하려는 IoT 노드의 URI 정보를 갖고 있어야 한다.

### III. 실내 IoT 자원 제어 서비스 설계

최근 소용량, 저성능, 저전력의 제약된 IoT 노드와 손실이 높고 전송율도 낮은 IoT 네트워크 환경에서 IoT 자원을 등록하고 상태를 감시하고 제어 관리하는 중요하다.

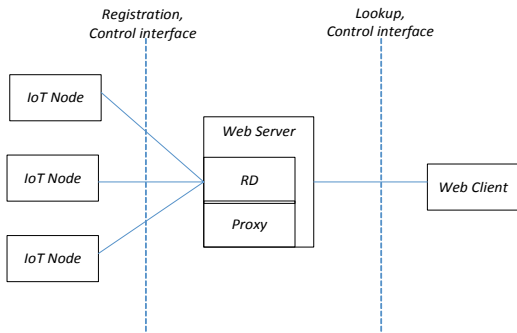


그림 2. CoAP 프로토콜을 이용한 웹 기반 실내 IoT 자원 제어 구조  
 Fig. 2. Indoor resource control architecture based on web using CoAP

CoAP 프로토콜을 이용한 웹 기반 실내 IoT 자원 제어 구조를 제시하고, 그림 2와 같이 보여주고 있다. 여기서 웹 서버는 웹 링크의 저장소와 기본적인 웹 서비스 역할뿐만 아니라 프록시와 RD 기능을 수행한다. 프록시 기능은, HTTP-CoAP 메시지를 상호 변환하여 웹 클라이언트에게 센싱 데이터 및 제어 결과를 보여준다. RD 기능은 IoT 노드의 정보 등록과 사용자에게 IoT 노드 검색을 제공한다. 웹 클라이언트가 RD에 요청하면 RD에서 해당 IoT 노드의 URI를 조회하여 요청을 재전송하는 원리를 사용한다. 여기서는 집중형 검색을 이용하여 각 IoT 노드는 실행하기 전에 먼저 RD에 IoT 자원 정보를 등록

혹은 갱신 작업을 수행하는 검색 방법을 제안한다.

그림 3은 실내 IoT 자원을 웹 서버 내 RD에 등록하는 흐름도이다. IoT 노드가 작동 될 때 IoT 노드의 정보를 포함한 등록 메시지를 웹 서버로 전송한다. 데이터베이스에 해당 식별자를 가진 IoT 노드 정보가 존재하지 않으면 이 정보를 데이터베이스에 추가하며 존재하면 IoT 노드의 정보를 갱신한다.

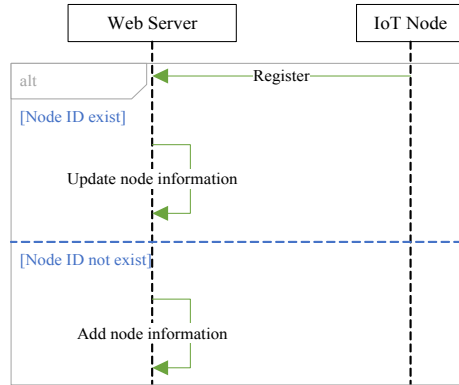


그림 3. 실내 IoT 자원 등록 흐름도  
 Fig. 3. Sequence diagram for IoT node registration

그림 4는 웹 서버를 통하여 실내 IoT 자원을 제어하는 흐름도이다. 사용자는 웹 client에 키워드를 입력하여 웹 서버를 통하여 등록된 IoT 노드의 정보를 수신한다. 다음 정보를 확인하고 해당 IoT 노드의 센서나 구동체를 제어한다. 웹 클라이언트에서 전송한 HTTP 제어 메시지는 웹 서버의 프록시 기능을 통해 CoAP 메시지로 전환하여 실내 IoT 자원에 전송된다. IoT 노드는 해당 명령을 실행하고 다시 그 결과를 사용자에게 전달한다.

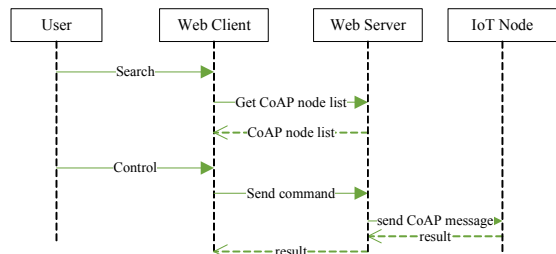


그림 4. IoT 노드 제어 흐름도  
 Fig. 4. Sequence diagram for controlling IoT node

사용자가 IoT 노드를 검색한 결과목록에서 특정 링크를 선택하여 해당 IoT 노드의 유닛에 대하여 제어할 수

있다. 그림 5에서는 IoT 노드의 온도 센서 유닛을 통하여 온도정보를 수집하는 흐름과 IoT 노드의 LED를 동작하는 흐름을 보여주고 있다.

사용자는 웹 클라이언트를 이용하여 온도센서를 포함한 IoT 노드를 검색하여 실시간 온도 값을 요구한다. 웹 클라이언트에서 요청명령을 HTTP를 통하여 웹 서버에 전송할 경우 CoAP 메시지를 생성하여 CoAP을 통하여 IoT 노드에 전송된다. 이때 IoT 노드는 여러 개의 센서 혹은 구동체를 갖고 있으며, 실내 IoT 자원이다. IoT 노드는 웹 서버를 거쳐 받은 웹 클라이언트의 요청을 처리하여 결과를 전송한다. 웹 서버는 이 결과를 다시 웹 클라이언트에 응답메시지로 전달한다.

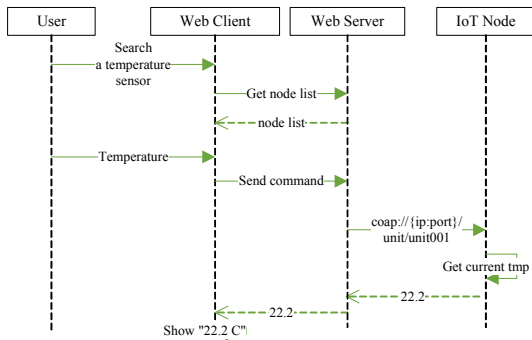


그림 5. 온도 센서 데이터를 요청하는 흐름도  
Fig. 5. Sequence diagram for requesting a temperature sensor data

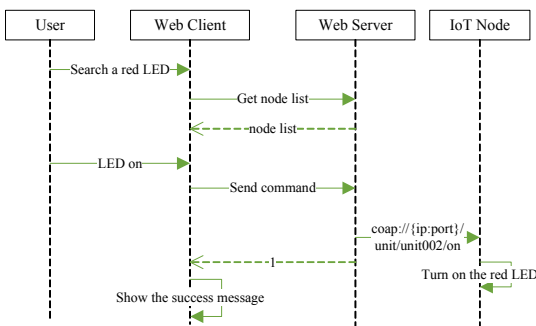


그림 6. 실내 IoT 자원의 LED를 제어하는 흐름도  
Fig. 6. Sequence diagram for controlling a LED-red

그림 6에서는 LED 구동체를 제어하는 흐름을 보여주고 있다. 여기서의 LED는 켜기/끄기 두 가지 기능을 가지고 있는 LED이며 사용자의 요청에 따라 그 결과가 실행여부를 사용자에게 반환한다. IoT 노드에서 정상적으로

로 실행하였으며 “1”값을 반환하는데 이 값이 웹 클라이언트를 통해 사용자에게 성공한다는 메시지를 도신한다.

#### IV. CoAP 프로토콜을 이용한 웹 기반 실내 IoT 자원 제어 서비스 구현

웹 서버는 IoT 노드의 정보 등록과 사용자에게 IoT 노드를 검색하는 기능을 제공한다. IoT 노드가 작동 될 때 IoT 노드의 정보를 포함한 등록 메시지를 웹 서버로 전송한다. 이 질의는 웹 서버의 데이터베이스에서 해당 IoT 노드를 조회하고 새로운 IoT 노드를 추가하거나 기존의 IoT 노드 정보를 갱신할 수 있다. CoAP과 HTTP를 연결하는 프록시를 제안하여 인터넷 상에서 웹 클라이언트와 IoT 노드 간의 상호 호환성을 갖는 통신 서비스를 구현한다.

사용자에서 웹 클라이언트를 이용하여 등록된 IoT 노드의 조회하고 해당 IoT 노드를 제어할 수 있다. 웹 클라이언트와 웹 서버 사이의 통신은 HTTP를 이용하고, 웹 서버와 IoT 노드 사이의 통신은 CoAP을 이용한다. 표 1과 같은 개발 환경에서 실내 IoT 자원 제어 서비스를 개발한다.

표 1. 실내 IoT 자원 제어 서비스 개발환경

Table 1. Development environment for indoor IoT resource control service

운영체제	Windows 7 64bit
프로그래밍언어	C, Java, HTML, CSS, Javascript, SQL
라이브러리	Californium CoAP, Libcoap, MRAA, Spring MVC, MyBatis, etc.
DBMS	SQL server 2014
개발도구	STS, Intel IoT tool kit
IoT 노드	Intel edison board

IoT 노드와 웹 서버 간의 CoAP 기반 통신을 구현하기 위하여 IoT 노드 측에서 Libcoap을 사용하고 웹 서버 측에서 Californium CoAP을 사용한다<sup>[7]</sup>. Libcoap과 Californium CoAP은 모두 RFC 7252의 표준을 따르며 CoAP 기반의 통신을 지원한다<sup>[8]</sup>. IoT 노드의 센서와 구동체 유닛을 제어하기 위하여 인텔에서 제공하는 MRAA을 이용하여 입출력 기능을 구현한다. 웹 서버의 HTTP 기반의 웹 서비스는 Spring MVC 프레임워크를 이용하여 개발한다.

실내 IoT 자원 정보 수집 서비스를 제공하기 위해 사용자가 웹 클라이언트를 통해 실내 IoT 노드에게 웹 서버를 통해 현재 온도값을 요청할 수 있다. 그림 7은 그 결과를 보여주는 웹 클라이언트의 화면이다. 여기서 요청과 더불어 현재 온도값을 확인할 수 있다.

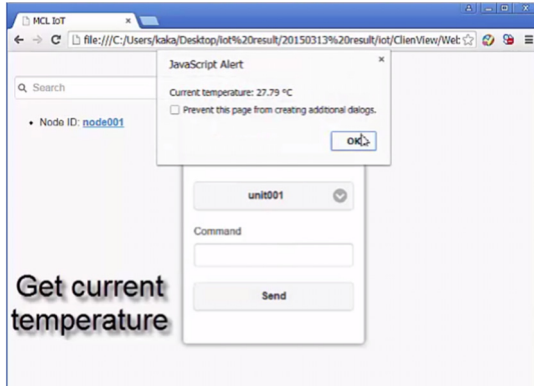


그림 7. 온도값을 요청하고 받은 웹 클라이언트  
 Fig. 7. Web client for requesting and response of a temperature value

실내 IoT 자원 제어 서비스를 제공하기 위해 먼저 실내 IoT 노드는 자신의 정보를 웹 서버의 RD에 전송하여 기존의 정보를 수정하거나 새로이 추가한다. 그림 8은 웹 서버의 RD에서 IoT 노드가 실행하면서 전송하는 CoAP 메시지를 보여주고 있다.

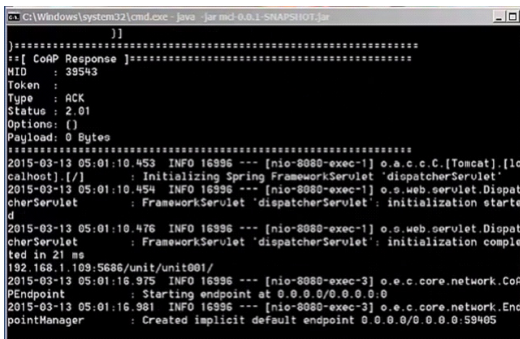


그림 8. 웹 서버의 RD 실행하면(IoT 노드 검색)  
 Fig. 8. RD result of Web server for node retrieving

그리고 사용자는 IoT 노드를 검색하여 원하는 실내 IoT 자원을 제어할 수 있으며, 이를 위해 제어 가능한 IoT 노드들의 정보를 확인한다. 그림 9는 IoT 노드를 검색한 다음 얻은 결과를 보여주고 있다.

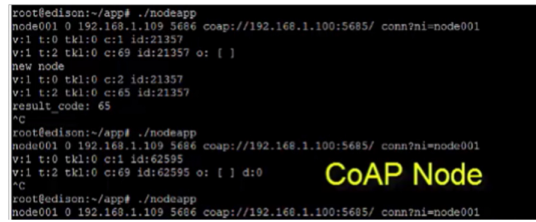


그림 9. 실내 IoT 자원 개발 결과  
 Fig. 9. Development result of indoor IoT resource

그림 10에서 LED 구동체를 제어하는 웹 클라이언트를 보여주고 있다. 여기서 사용자가 LED 구동체를 제어하기 위해 명령 메시지를 IoT 노드에게 전달하고, IoT 노드는 명령을 수행하여 성공여부를 웹 클라이언트에게 전달한다.

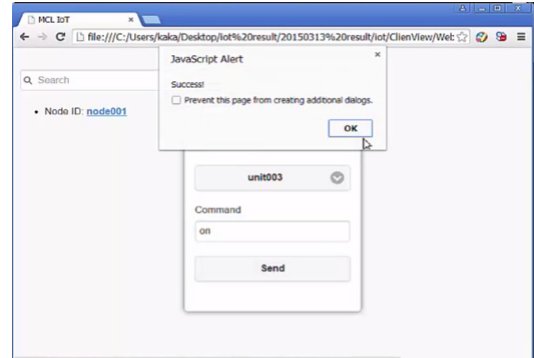


그림 10. LED를 제어하는 웹 클라이언트  
 Fig. 10. Web client for controlling a LED

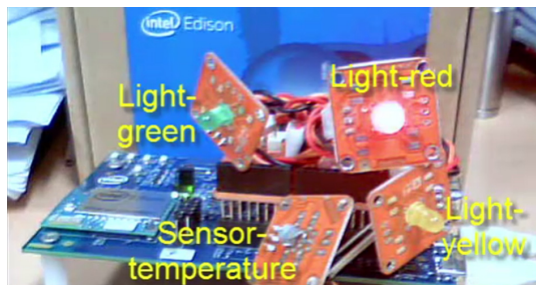


그림 11. LED 제어를 위한 IoT 노드 하드웨어  
 Fig. 11. IoT node hardware of controlling a LED

그림 11에서는 그림 9에서 실행한 구동체 제어 서비스를 위한 IoT 노드 하드웨어를 보여주고 있다. IoT 노드 하드웨어는 인텔 에디슨 보드 상에 3개의 LED와 1개의 온도센서로 구성된다. 그림 10에서 요청한 unit003에 대한 “ON” 명령을 처리하여 LED를 켜고, 그림 10과 같이

성공 메시지를 웹 클라이언트에 도신한다.

## V. 결론

본 논문에서는 웹 기반의 CoAP 프로토콜을 이용한 실내 IoT 자원 제어 서비스를 설계하고 구현한다. 이 서비스를 제공하기 위해 웹 서버는 RD와 프록시를 갖고 있으며, 웹 서비스, IoT 자원 등록 서비스, HTTP와 CoAP 프로토콜 변환 서비스를 제공한다. 인터넷 상에서 실내 IoT 자원을 모니터링하고 제어할 수 있다. 향후 이 서비스는 스마트 홈, 스마트 공장 등에 적용되어 표준 프로토콜 기반의 다양한 지능적인 IoT 서비스를 개발하는 데 기여할 것이다.

## References

[1] Z. Shelby, B. Frank, D. Sturek, "Constrained Application Protocol (CoAP)", RFC 7252, June, 2014.

[2] Z. Shelby, M. Koster, C. Bormann, P. van der Stok, "CoRE Resource Directory", draft-ietf-core-resource-directory-04, July, 2015.

[3] A. Castellani, S. Loreto, A. Rahman, T. Fossati, E. Dijk, "Guidelines for HTTP-CoAP Mapping Implementations", draft-ietf-core-http-mapping-07, July, 2015.

[3] Castellani, A., Fossati, T., Loreto, S., "HTTP-CoAP cross protocol proxy: an implementation viewpoint", Mobile Adhoc and Sensor Systems (MASS), 2012 IEEE 9th International Conference on, Oct, pp.1-6, 2012.

[5] Mingozzi, E., Tanganelli, G., Vallati, C., "CoAP Proxy Virtualization for the Web of Things", Cloud Computing Technology and Science (CloudCom), 2014 IEEE 6th International Conference on, pp.577-582, Dec, 2014.

[6] Yoona Kim, Ki-Hyung Kim, Taeshik Shon,

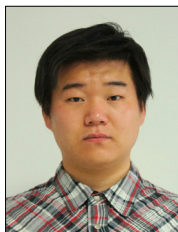
Jai-Hoon Kim, "A hybrid service discovery approach to mitigate overhead concentration on resource directory", Information Networking (ICOIN), 2015 International Conference on, pp.407-409, Jan, 2015.

[7] Jian Wang, June SaKong, Ho-Young Kwak, Do-Hyeun Kim. "Design and Implementation of IoT Middleware Using Data Refinement Scheme based on IETF CoAP", The Journal of The Institute of Internet, Broadcasting and Communication (JIIBC), Vol. 15 No. 6, Dec. 2015.

[8] Wen-Quan JIN, Do-Hyeun Kim, "Implementation and Experiment of CoAP Protocol based on IoT for Verification of Interperability", The Journal of The Institute of Internet, Broadcasting and Communication (IIBC), Vol.14, No.4, pp.7-12, Aug.31.2014.

## 저자 소개

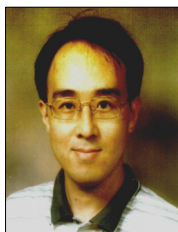
### 김 문 권(준회원)



- 2008년 9월 ~ 2013년 8월 : 연변과학기술대학 전산학과 학사
- 2013년 9월 ~ 2015년 2월 : 제주대학교 대학원 컴퓨터공학과 석사
- 2015년 3월 ~ 현재 : 제주대학교 대학원 컴퓨터공학과 박사과정

<주관심 분야> : 사물인터넷, 임베디드 시스템

### 김 도 현(종신회원)



- 1990년 3월 ~ 1995년 3월 : 국방과학연구소 전산망연구실 연구원
- 2004년 9월 ~ 현재 : 제주대학교 공과대학 컴퓨터공학과 교수
- 2013년 1월 ~ 현재 : 대한전자공학회 M2M/IoT 연구회 회장

<주관심 분야> : 서비스 컴퓨팅, 사물인터넷, 모바일 컴퓨팅

※ "이 논문은 2015년도 제주대학교 학술연구지원 사업에 의하여 연구되었음"