

스마트 시티 인프라 관리를 위한 사물인터넷 식별체계 및 엣지 게이트웨이 아키텍처

황재영*, 손현서*, 성낙명**, 김재호**, 송재승^o

Internet of Things Identification and Edge Gateway Architecture for Smart City Infrastructure Management

JaeYoung Hwang*, HyeonSeo Son*, NakMyoung Sung**, JaeHo Kim**, JaeSeung Song^o

요 약

현재 도시에서 발생하는 다양한 문제를 해결하고 효율적으로 도시 인프라를 관리하기 위하여 사물인터넷, 인공지능, 5G와 같은 기술들을 결합하여 다양한 연구 및 검증이 진행되고 있다. 수십억 대의 디바이스가 연결되어 운영될 것으로 예상되는 스마트 시티 인프라 관리를 위하여 사용하는 기존의 중앙 집중형 클라우드에서 즉각적인 반응을 요구하는 서비스 및 대용량의 데이터를 모두 처리하는 것은 부담이 되므로 엣지 컴퓨팅 개념을 도입하여 효율적으로 스마트 시티 인프라를 관리하기 위한 방안이 연구되고 있다. 그러나 다양한 프로토콜 및 표준이 사용되는 사물인터넷 환경을 고려하지 않은 엣지 컴퓨팅 환경의 구조에서는 디바이스들이 사용하는 식별체계가 서로 다르기 때문에 엣지 게이트웨이에서 일괄적으로 디바이스를 관리하는 것이 어렵다. 따라서, 본 논문에서는 현재 스마트 시티에서 주로 사용되는 프로토콜 및 표준에 대한 식별체계 분석 및 이기종 환경 지원이 가능한 엣지 게이트웨이 구조를 제시하였으며, 또한, 엣지 게이트웨이를 통한 스마트 시티 서비스 제공을 하기 위한 상세 절차를 설명하였다. 본 논문에서 제안하는 식별체계 및 엣지 게이트웨이 기능을 통해 스마트시티 내의 새롭게 추가되는 시스템 및 다양한 프로토콜 뿐만아니라 기존 스마트 시티 인프라 요소들도 효율적으로 연결 및 관리할 수 있을 것으로 예상된다.

Key Words : Edge Computing, Identification, Internet of Things, Smart cities, Standards

ABSTRACT

In order to solve various problems occurring in the cities and to manage the urban infrastructure efficiently, various studies and verification are being conducted by combining technologies such as Internet of Things, Artificial Intelligence, and 5G. In the existing centralized cloud used for smart city infrastructure management, which is expected to be connected with billions of devices, it is burdensome to process both instantaneous response services and large amounts of data, so we introduce the concept of edge computing to manage smart city infrastructures is being studied. However, in the edge computing environment that does not take into consideration the Internet of Things environment where various protocols and standards are used, it is difficult to manage the devices effectively in the edge gateway because the identification schemes used by the devices are

※ 본 연구는 환경부의 재원으로 한국환경산업기술원의 지능형 도시수자원 관리사업의 지원을 받아 수행된 연구임 (RE201903069)

• First Author : Autonomous IoT Research Center, Korea Electronics Technology Institute & Sejong University, forest62590@sju.ac.kr, 정희원

^o Corresponding Author : Department of Information Security, Sejong University, jssong@sejong.ac.kr, 정희원

* Sejong University

** Korea Electronics Technology Institute

논문번호 : 201907-129-0-SE, Received June 14, 2019; Revised August 21, 2019; Accepted August 25, 2019

different. In this paper, we present the edge gateway architecture that can support the heterogeneous environment and analyse the identification of protocols and standards that are currently used in smart cities. By using this architecture, not only newly developed standards and protocols but also existing infrastructure elements are expected to be able to be efficiently connected and managed.

I. 서 론

현재 정부 및 지자체들은 도시에서 발생하는 다양한 문제해결 및 효과적으로 도시 인프라를 관리하기 위하여 스마트 시티에 대한 기술 및 운영 방안에 대해 연구 중이며¹⁻²⁾, 사물인터넷 (Internet of Things)은 5G, 인공지능과 같은 기술들과 결합하여 스마트 홈, 스마트 팩토리 뿐만 아니라 스마트 시티 등과 같은 4차 산업을 이끄는 주요 기술 중 하나로 자리매김하고 있다. 사물인터넷을 기반으로 하여 다양한 도시 인프라 서비스들이 제공되고 있으며, 스페인의 샌텐더 (Santander) 스마트 시티 및 한국의 부산은 이미 스마트 가로등, 버스 운행, 택시정보, 주차 공간 등의 정보를 실시간으로 수집하고 제공하는 서비스를 통해 효율적으로 도시 인프라를 관리함과 동시에 시민들에게 편의를 제공하고 있다³⁻⁵⁾.

현재 대부분의 기존의 도시에서 운영되는 사물인터넷 플랫폼은 도시의 데이터 축적 및 분석에 초점을 맞춘 클라우드 플랫폼 기반으로 구축 되어있다. 그러나, 많은 스마트 시티 서비스들의 경우 초저지연의 응답 속도를 요구사항으로 정의하고 있으며, 동영상 데이터 및 다양한 센서들로부터 정보를 클라우드에 전송하고 있어 고대역과 초대용량의 스토리지를 필요로 하고 있다⁶⁾. 예를 들어, 스마트 시티 자율주행 서비스는 일반적인 센서가 아닌 차량, 도시의 센서, 기지국 및 보행자들이 사용하는 수많은 데이터들에 (차량 및 사용자의 위치, 도시의 교통량, 교통신호 데이터, 통신상태 등) 대한 수집 및 종합적인 판단이 필요하다. 그러나 클라우드에서 모든 데이터를 수집하여 분석 및 처리하는 것은 예측 불가능한 지연상황을 발생시킬 수 있기 때문에 엣지 컴퓨팅을 도입하여 서비스 사용자와의 짧은 거리 (Proximity)를 확보하여 서비스 지연 속도 (Latency)를 줄일 수 있다⁷⁻⁸⁾. 또한, 다양한 프로토콜 및 표준을 사용하는 수십억 대 이상의 사물인터넷 디바이스가 배치 및 운영 될 것으로 예상되는 복잡한 스마트 시티 인프라 운영 환경에서⁹⁾, 기존에 사용하던 클라우드 기반 중앙 집중형 (Centric) 방식의 스마트 시티 인프라는 즉각적인 반응을 요구하는 서비스를 이용하는 시민에게 곧 바로 응답을 줄 수 없을 뿐

만 아니라 디바이스가 전달하는 대량을 데이터를 온전히 클라우드에서 처리하는 것은 상당한 부담이 된다¹⁰⁾. 또한 사람이 개입하여 모든 스마트 시티 인프라에 대한 직접 구성/관리/운영하는 서비스는 현실적으로 불가능하다. 따라서 이것은 효율적으로 스마트 시티 인프라를 관리하기 위한 새로운 아키텍처가 필요함을 의미하며, 현재 위와 같은 문제점들을 해결하기 위한 방법으로 엣지 컴퓨팅 (Edge Computing) 기술을 사용하여 효율적인 데이터 관리 및 데이터 전달의 실시간성을 증대시키고 또한 인공지능 기술과 결합하여 자동화 된 인프라 관리 기술을 적용을 하여 위 언급한 문제를 해결하기 위한 연구가 진행되고 있다¹¹⁾.

그러나 사물인터넷 환경에 대한 고려없이 엣지 컴퓨팅 기술을 도입하는 위와 같은 구조에서는 효율적인 스마트 시티 인프라 관리를 위한 해결책이 될 수 없다. 현재 사물인터넷 프로토콜 및 표준에는 LoRa (Long Range), BLE (Bluetooth Low Energy), oneM2M, NGSI (Next Generation Service Interface), OCF (Open Connectivity Foundation) 와 같은 다양한 기술들이 사용되고 있으며, 제조사에 따라 또는 스마트 시티 서비스 시나리오 환경에 따라 사용하는 프로토콜 또는 표준이 다르다. 즉, 해당 디바이스들이 사용하는 식별체계가 다를 뿐만 아니라 발생시키는 데이터 모델도 다르기 때문에 수많은 디바이스를 일괄적으로 관리하기가 어렵다. 따라서 앞서 언급한 문제점에 대해 고려하여 엣지 게이트웨이가 개발이 되어야 하며, 요약하면 다음과 같다.

- 스마트 시티 인프라 통합 운영을 위해서는 기존에 존재하는 이기종의 프로토콜 및 표준과의 상호운용성 확보를 통해 다양한 디바이스 지원을 위한 방법이 필요¹²⁾.
 - 이기종 사물인터넷 디바이스 관리를 위한 식별체계 및 해당 디바이스가 발생시키는 데이터를 관리하기 위한 데이터 모델링이 필요
 - 표준 기반의 엣지 게이트웨이 구조를 적용하여 하위계층에서 연결되는 디바이스 및 데이터를 표준화된 방식의 관리 및 통합이 필요
- 따라서, 본 논문에서는 위 요구사항 중에서 상호운용성을 지원하기 위한 엣지 아키텍처 및 디바이스 관

리를 위한 식별체계에 대해 제시하여 어떻게 다양한 프로토콜 및 표준과 연동 가능한 엣지 게이트웨이 운용이 가능한지에 대해 제시한다. 제시된 아키텍처를 사용하여 기존에 정의된 다양한 사물인터넷 표준뿐만 아니라 향후 새로운 요구에 따라 개발될 표준 및 프로토콜들도 손쉽게 구축된 스마트 시티 인프라에 결합하여 효율적으로 통합 스마트 시티 인프라 구축 및 운영이 가능할 것으로 예상된다.

본 논문의 구성은 다음과 같다. 먼저 스마트 시티에서 주로 사용되는 사물인터넷 표준 및 디바이스들의 식별체계에 대해 기술하며 (II장), 해당 식별체계를 통합하기 위한 표준화 된 엣지 게이트웨이 구조 및 식별체계를 제시한다 (III장). 엣지 컴퓨팅과 관련하여 진행 중인 프로젝트에 대해 알아보고 (IV장), 마지막으로 본 논문에 대한 요약 및 향후 연구 방향에 대해 기술한다 (V장).

II. 사물인터넷 디바이스 식별체계

스마트 시티에는 하나의 특정 프로토콜이나 표준만 사용되는 것이 아니라 다양한 기술들이 연결되어 특정 서비스를 제공하게 되며, 본 절에서는 스마트 시티 인프라 구축에 사용되는 사물인터넷 디바이스 프로토콜 및 표준의 종류에 대해 알아봄과 동시에 식별체계 분석을 하고자 한다.

2.1 디바이스 식별체계 분류

식별체계는 물리적 개체 또는 논리적 개체에 식별자 (Identifier)를 부여하여 고유하게 구별하기 위하여 사용된다. 또한, 해당 정보를 기반으로 개체들을 검색, 전송, 제어 및 관리 할 수 있으며, 아래와 같이 크게 3개의 범주로 나눌 수 있다¹³⁾.

- **Tangible Identifier (TI):** Tangible Identifier는 네트워크 연결 기능을 가지고 있지 않은 일반적인 사물들을 (방, 사람, 자동차 등) 구별할 때 사용하는 식별체계이다. Tangible Identifier은 보통 코드화 되어 물리적인 디바이스에 부착 되며, 1D barcode, 2D barcode, 또는 Radio-Frequency Identification (RFID)와 같은 타입이 있다.
- **Communication Identifier (CI):** Communication Identifier는 터미널, RFID 리더, 센서 및 네트워크 디바이스와 같이 통신 능력을 가지고 있는 (Communication capabilities) 노드들을 구별하기 위한 식별체계이며 Internet Protocol (IP)를 하나의 예로 들 수 있다.

- **Application Identifier (AI):** Application Identifier는 논리 객체, 애플리케이션 계층 또는 서비스 계층에서 제공하는 서비스들을 식별하기 위해 사용하는 식별체계이며, 그 예로 Uniform Resource Identifier (URI), Uniform Resource Locator (URL)을 들 수 있다.

위와 같이 총 3개의 큰 대분류로 나눌 수 있으며, 본 논문에서는 사물인터넷 특성에 맞게 CI 또는 AI와 같이 통신 기능을 가진 프로토콜 및 표준을 사용하는 디바이스들을 구별하기 위한 식별체계를 기반으로 기술한다.

2.2 사물인터넷 디바이스 프로토콜 및 표준

사물인터넷 아키텍처에서는 LoRa, Bluetooth Low Energy (BLE), oneM2M, FIWARE, IoTivity와 같이 여러 프로토콜 및 표준을 이용한 사물인터넷 디바이스들이 사용되며 다음 해당 프로토콜에 대한 설명 및 식별체계에 대해 기술하였다.

- **로라 (LoRa):** 롱 레인지 (Long Range)의 약자인 로라는 3G나 LTE (Long Term Evolution) 같은 기존의 통신망과는 다르게 저전력으로 장거리를 통신할 수 있는 통신망이다¹⁴⁾. 공원 내 미이방지, 공용자전거 및 스마트 파킹 서비스와 같이 실시간 현황이 필요한 서비스에 주로 사용되고 있다. 로라의 연결 방식에는 Over-The-Air-Activation (OTAA), Activation By Personalisation (ABP)와 같이 두 가지의 방식이 존재하며, 두 절차의 선택에 따라 사용하는 식별체계를 사용한다. 본 논문에서는 OTAA 방식의 식별체계인 End-device Identifier (DevEUI)를 사용한다. DevEUI는 64-bit의 주소로 이루어진 (IEEE EUI64) 장치 고유 하드웨어 식별자이며 TCP/IP의 MAC 주소와 비교될 수 있다¹⁵⁾.
- **BLE:** BLE는 Bluetooth SIG (Bluetooth Special Interest Group)가 헬스케어, 피트니스, 비컨과 같은 영역에서 개발되는 새로운 애플리케이션 지원을 위해 개발 중인 무선 네트워크 기술이다. 기존의 블루투스에 기술과 대비해 BLE는 비슷한 통신 범위를 유지하면서 전력 소비와 비용을 크게 줄일 수 있다. Bluetooth Device Address (또는 BD_ADDR)는 제조업체가 각 Bluetooth 장치에 할당 한 고유 한 48 비트 식별자이다. 블루투스 주소는 일반적으로 16진수로 작성되고 콜론으로 구분 된 6바이트로 표시 된다 (예: 00:11:22:33:FF:EE). 블루투스 주소의 상위 절반

(최상위 24 비트)을 조직 고유 식별자 (Organizationally Unique Identifier, OUI)라고 하며, 장치 제조업체를 결정하는 데 사용할 수 있으며 (Bluetooth MAC Address Lookup form). OUI 접두사는 Institute of Electrical and Electronics Engineers (IEEE) 에서 할당한다.

- oneM2M: 파편화 되는 사물인터넷 시장을 글로벌 단위의 수평적 서비스로 제공하기 위해서 시작된 서비스 레이어 표준 개발을 위한 파트너 쉽 프로젝트로 다양한 사물인터넷 프로토콜 및 표준과의 인터워킹 기능 제공, 데이터 및 사물에 대한 관리 기능을 제공하고 있는 구현 수준의 유일한 표준이다. 여러 도시에서 oneM2M 기반의 스마트 시티 인프라를 제공하고 있으며 특히 부산은 스마트 가로등 주차 및 빌딩 에너지 관리와 같은 서비스 제공을 위해 해당 표준을 기반으로 구축하여 운영 중이다. RESTful (REpresentational State Transfer) 아키텍처 기반의 oneM2M은 기본적으로 현실에 존재하는 실제 개체들을 리소스로 구분하고 관리하며, 해당 리소스들을 구분 및 관리하기 위하여 트리구조의 형태를 지닌다. 각 트리구조의 리소스들은 URI (Uniform Resource Identifier) 기반의 Resource Name과 같은 시스템 내의 유일한 식별자를 지니며 해당 식별자를 기반으로 리소스들을 구분한다.

- OCF: 14년 7월 삼성과 인텔의 주도로 설립된 OCF는 사물인터넷 플랫폼 표준 개발 및 오픈소스 프로젝트인 IoTivity를 개발 중에 있으며, 다양한 디바이스들이 해당 표준을 기반으로 개발되고 있다^[16]. OCF의 IoTivity 플랫폼은 REST 아키텍처 기반으로 실세계의 온도센서, 전구 또는 기타 디바이스를 리소스 형태로 구분한다. OCF의 식별자는 크게 Context-Specific과 Context-Independent로 나눌 수 있다. 먼저 Context-Specific은 특정 컨텍스트에서 고유하게 사용되는 도메인이며 예를 들어 IP (Internet Protocol)와 같은 주소는 집 안에서 특정 게이트웨이를 사용하는 디바이스의 유일한 식별자가 될 수 있다. Context-Independent같은 경우는 UUID (Universally Unique Identifier)와 같은 상대적으로 더 강력한 스키마를 사용해야 한다. IoTivity는 장치를 고유하게 식별하기 위하여 Device ID (DI)라는 애트리뷰트 (특성)을 제공하며 디바이스를 고유하게 식별하기 위하여 UUID 기반의 128비트의 숫자열 혹은 문자열을 사용한다^[17].

- FIWARE: 유럽의 미래 인터넷 플랫폼 (Future

Internet Platform)으로 Generic Enabler (GE)라고 하는 다양한 컴포넌트를 제공 및 조합하는 방법으로 다양한 서비스를 제공하고 있으며, FIWARE가 사용하는 표준은 Open Mobile Alliance (OMA)에서 정의한 NGSI 9, 10 표준을 사용한다. NGSI 표준의 핵심은 Entity이며 Entity를 통하여 실제 현실에 존재하는 자동차, 디바이스 등의 물리적 객체를 표현하며 해당 Entity는 Entity ID, Entity Type을 통해 구별하며 추가적인 정보 표현을 위해 Attribute같은 요소가 사용된다.

위와 같이 현재 각 디바이스마다 서로 다른 디바이스 식별체계를 사용하고 있기 때문에 이종 플랫폼 간 디바이스의 식별에 대한 어려움이 문제로 인식되고 있으며, 따라서 각 플랫폼 간 디바이스 식별체계에 대한 번역기 또는 해석기의 필요성이 제시되었다^[18-19].

III. 엣지 게이트웨이 아키텍처 및 식별체계

현재 대부분의 기존 도시가 클라우드 플랫폼을 기반으로 하여 도시의 데이터 축적과 분석에 초점을 맞추고 있는 반면, 인공지능과 같은 기술을 이용한 새로운 스마트 시티 서비스 개발 및 그에 따른 초저지연의 응답속도와 같은 새로운 요구사항에 대응하기 위하여 엣지 게이트웨이의 도입이 검토되고 있다. 스마트 시티에 엣지 게이트웨이를 도입할 때 고려해야 할 점으로 각 도시의 환경에 따라 다양한 프로토콜 및 표준을 사용하는 사물인터넷 디바이스가 배치되어 운영 될

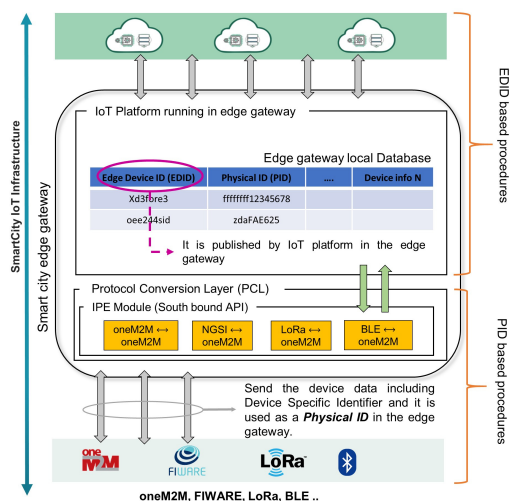


그림 1. 이기종 지원 엣지 게이트웨이 구조
Fig. 1. Edge gateway architecture for heterogeneous protocols

것으로 예측되고 있어 상호운용성에 대한 고려가 필요하다. 즉, 앞서 연구된 내용과 같이 디바이스들이 사용하는 프로토콜 및 표준에 따라 식별체계가 다르기 때문에 엣지 게이트웨이에 연결 될 수많은 디바이스를 일괄적으로 관리하기가 어렵다. 따라서 상호운용성을 고려하여 엣지 게이트웨이가 개발 되어야 하며, 본 절에서는 다양한 프로토콜 및 표준을 수용하여 스마트 시티의 엣지 게이트웨이를 운영하기 위한 방법으로 이중환경을 지원하는 엣지 게이트웨이의 아키텍처와 관련 절차에 대해 기술하였다.

3.1 이중환경 엣지 게이트웨이 아키텍처

앞 절에서 분석하였듯이 각 표준 및 프로토콜마다 디바이스를 구분하는 특별한 식별체계가 있으며 해당 식별체계를 통해서 유일하게 디바이스를 구별할 수 있다. 이러한 식별체계를 본 엣지 게이트웨이 구조에서는 물리 식별자 (Physical Identifier, PID) 라고 하여 구분하게 된다. 해당 PID는 엣지 게이트웨이의 내부에 존재하는 Protocol Conversion Layer (PCL)에 존재하는 프로토콜 변환 모듈인 Interworking Proxy Entity (IPE) module에^[20] 의하여 엣지 디바이스 식별자 (Edge Device Identifier, EDID)와 함께 관리 및 저장된다.

해당 아키텍처에서 EDID를 정의하여 식별체계를 관리하는 이유는 각 PID의 충돌 즉, 중복생성을 방지할 뿐만 아니라 좀 더 체계적이고 일괄적인 방식으로 식별체계를 운영하기 위함이다. 가령 10개의 디바이스 모두가 서로 다른 표준이나 프로토콜을 사용한다고 했을 때, 서로 다른 식별체계는 10개의 식별자를 생성하게 되고 서로 다른 식별체계가 엣지 게이트웨이에서 사용되게 되므로 각 디바이스들의 리소스들을 관리하기 위해 여러 개의 리소스 모델 정의 및 그에 따른 데이터베이스의 설계는 리소스 관리 측면에서 복잡성을 증가 시킬 수 밖에 없다. 따라서 표준화된 방법의 식별체계 관리를 통해서 하위 계층에서 등록되는 서로 다른 표준 및 통신 기술들은 표준화된 방식으로 엣지 게이트웨이 내부에서 통합 및 관리 될 수 있으며, 내부에서 사용되는 식별체계를 정리하면 다음과 같다.

- **Physical Identifier (PID):** 실제 디바이스들에게서 전달되고 외부에서도 식별될 수 있는 유일한 ID이며, 해당 ID를 이용하여 디바이스는 엣지 게이트웨이에 등록되어 운영된다. 엣지 게이트웨이 내에서 운영되는 사물인터넷 플랫폼과 등록하려는 디바이스가 다른 표준을 사용할 경우 PCL에 있는 IPE에

의해서 디바이스에 대한 데이터가 타겟 표준 형식으로 변환 된 다음 엣지 게이트웨이에 등록된다. 이때 사물인터넷 플랫폼은 엣지 게이트웨이와 클라우드 계층 내에서 사용되는 EDID를 발급한다.

- **Edge Device Identifier (EDID):** 엣지 게이트웨이 내부에서 동작하는 사물인터넷 플랫폼이 발급하는 유일한 ID이며, 해당 ID를 이용하여 엣지 게이트웨이는 디바이스의 정보를 외부의 사물인터넷 플랫폼에 등록하고 운영한다. 엣지 게이트웨이 내에서만 유일하게 인식되는 ID이기 때문에 다른 엣지 게이트웨이에서 동일한 ID가 생성 될 수 있다. 엣지 게이트웨이 또는 클라우드에서 엣지 게이트웨이를 거쳐 명령을 디바이스로 전달해야 할 경우에는 EDID를 사용하게 되면 PCL의 IPE에서 어떤 디바이스에 대한 명령인지 식별할 수 없기 때문에 해당 방법에서는 PID를 확인하여 디바이스에 명령을 전달하게 된다.

엣지 게이트웨이에서 관리하는 식별체계가 해당 도메인 내에서 유일한 것처럼, 클라우드 서버 계층에서도 유일하게 구별가능한 (Globally unique) EDID를 발급하고 관리가 되어져야 한다. 다음 절에서는 사물인터넷 표준을 활용하여 클라우드 서버 계층에서 유일하게 구별 가능한 EDID를 발급하고, 이를 엣지 게이트웨이 계층의 식별체계와 연동하는 방법에 대해 설명한다.

3.2 표준 기반 엣지 게이트웨이 통합 식별체계

스마트 시티 인프라를 관리하기 위한 구조는 그림 1과 같이 크게 다양한 디바이스들이 배치되어 있는 디바이스 계층 (Device Layer), 각 배치 된 디바이스들 연결되어 실시간성 증대시키기 위한 엣지 계층 (Edge Layer), 엣지 계층에서 전달된 반영구적인 데이터를 보관하거나 대용량의 데이터를 처리하는 클라우드 계층 (Cloud Layer)으로 분류 할 수 있다. 앞서 언급하였듯이 디바이스 계층에 있는 디바이스를 구분하기 위해서 엣지 계층으로 연결되는 디바이스들을 어떻게 식별할 수 있는지에 대해 초점을 두어 엣지 계층의 핵심인 엣지 게이트웨이의 구조 및 식별체계관리 방법에 대해 제시하였다.

표준화 된 방법으로 디바이스를 관리하기 위해서 사물인터넷 표준인 oneM2M을 사용하여 엣지 게이트웨이의 식별체계를 구성하였다. oneM2M 표준은 모든 실제 개체를 리소스들로 구성하여 하나의 구조화된 트리가 되며 일반적으로 데이터를 관리하기 위한 구조는 그림 2와 같으며 해당 리소스들의 정보는 다

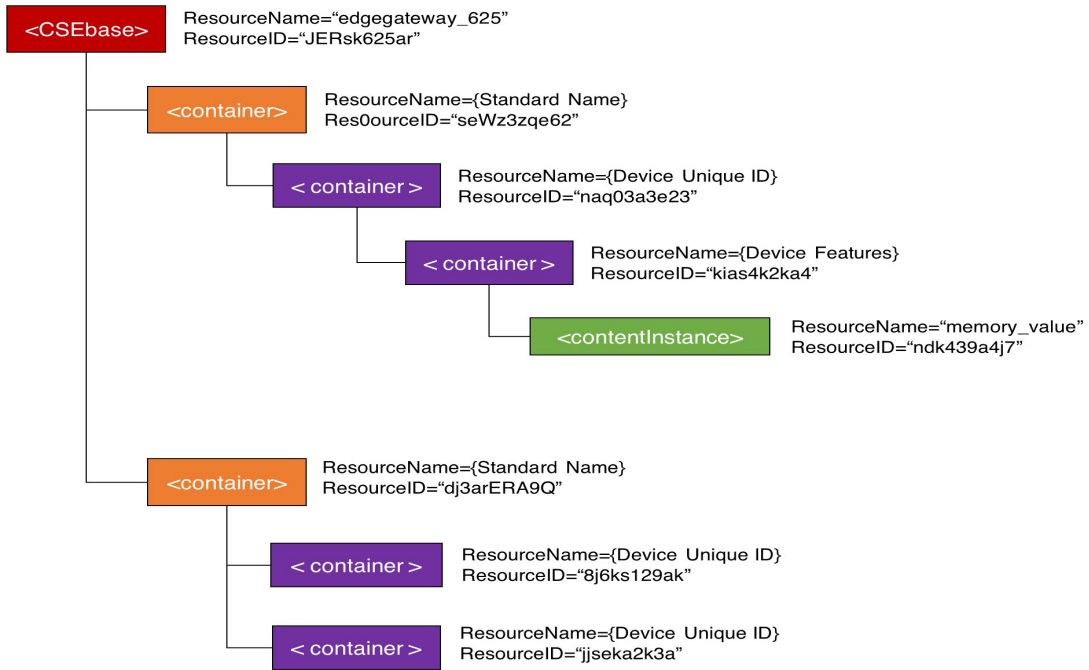


그림 2. oneM2M 리소스 구조
Fig. 2. oneM2M Resource structure

음과 같다^[21]. <CSE>는 oneM2M에서 생성되는 모든 리소스의 루트가 되며, <AE>는 <CSE>에 등록되는 하나의 서비스 단위이다. <container>는 특정 개체의 데이터 인스턴스의 컨테이너 역할을 하며, <contentInstance>는 실제 측정되는 데이터를 저장하는 인스턴스 역할을 한다.

oneM2M은 각 디바이스를 구분하기 위하여 RETSful 아키텍처 기반의 리소스 (Resource) 형태로 저장한다. 해당 아키텍처에서의 식별 방법은 oneM2M에는 Hierarchical/Non-Hierarchical 두 가지의 리소스 구조가 있으며, Hierarchical에서는 Resource Name (RN)을 사용하고 Non-Hierarchical에서는 Resource Identifier (RI)라는 oneM2M 시스템에서 자체에서 발급하는 유일하게 식별 가능한 식별자를 사용하게 된다. oneM2M의 표준에 따르면^[21], RI는 IETF RFC 3986: “Uniform Resource Identifier (URI): General Syntax”의 Unreserved Character에 정의된 문자들만 사용가능 하도록 되어있다 (Unreserved Characters = ALPHA / DIGIT / “-” / “.” / “_” / “~”)^[22]. 따라서, 그 외에 문자열 길이와 같은 제한은 없기 때문에 Unreserved Characters를 조합하여 서비스 내에서 유일한 RI를 생성해 내는 방법은 플랫폼을 구성하는 개

발자의 구현 방식에 따라 달라 질 수 있다. RN으로 oneM2M 시스템에 등록하게 되면 리소스가 등록 될 때 해당 시스템에서 유일한 랜덤한 RI라는 것을 발급 하여 운영하는 형식이다. 이 때 해당 RI는 앞 절에서 제시한 EDID가 되며 디바이스 등록에 사용 된 RN은 PI로 사용된다. 예를 들어, oneM2M 리소스 구조에서 엣지 게이트웨이에 등록 된 디바이스는 자신이 사용 중인 메모리의 상태 정보를 지속적으로 올리고 있는 상태이며 해당 정보는 다음과 같이 <contentInstance>에 저장된다. 해당 메모리 정보에 접근할 수 있는 방법으로는 위와 같이 Hierarchical/Non-Hierarchical 두 가지의 방법이 있으며 먼저 Hierarchical 방법은 edgeway_625/{Standard Name}/{Device Unique ID}/{Device Features}/memory_value와 같이 접근할 수 있으며 Non-Hierarchical에서는 시스템 내에서 유일하게 할당된 식별자이기 때문에 ndk439a4j7 와 같은 방법으로 접근 할 수 있다.

3.3 디바이스 등록 및 데이터 획득 절차

일반적으로 디바이스 관리 측면에서 보았을 때 Create, Retrieve, Update, Delete (CRUD)에 해당하는 4개의 명령에 대해 고려해 볼 수 있으며, 본 절에서는 디바이스 등록 및 데이터 획득 과정의 이해를 돕

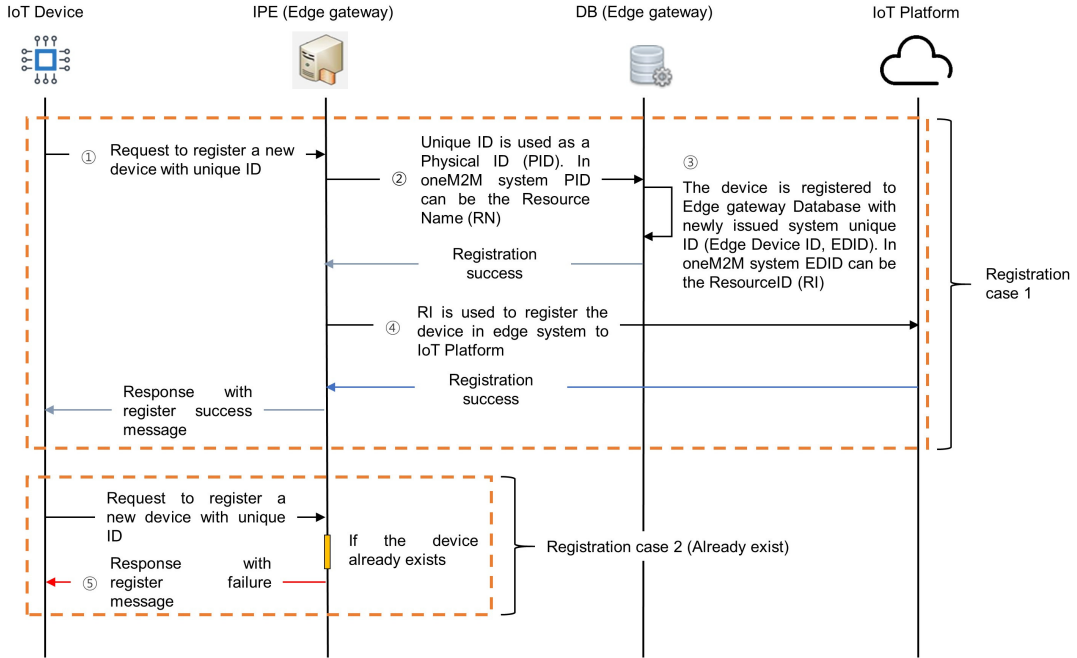


그림 3. 사물인터넷 디바이스 등록 절차
Fig. 3. Procedures of IoT device registration

기 위하여 그림 3, 4와 같이 기술하였다.

- 디바이스 등록 과정: 우선 디바이스는 1) 자신이 연결할 엣지 게이트웨이에 등록 요청을 보낸다. 2) 등록 요청을 받은 게이트웨이는 해당 등록을 요청한 디바이스가 이미 자신에게 등록이 되어 있는 디바이스의 PI인지 확인 후 그렇지 않으면 해당 PI를 이용하여 엣지 게이트웨이의 oneM2M system에 등록한다. 3) 디바이스 등록 시에 시스템 내에서 유일하게 구별 가능한 SI를 발급받는다. 4) 클라우드에 등록 시 엣지 게이트웨이는 SI를 이용하여 가상의 디바이스 개체를 생성하고 클라우드에서는 해당 식별자를 기반으로 디바이스를 관리한다. 5) 만약 디바이스가 이미 등록되어 있는 상태라면 엣지 게이트웨이는 이미 등록이 되어 있다는 에러 메시지와 함께 디바이스에게 응답하게 되며, 해당 메커니즘으로 인해 한번 엣지 게이트웨이에 등록 된 디바이스는 해지 과정을 거치지 않으면 등록이 될 수 없다.
- 디바이스 데이터 획득 과정: 디바이스가 이미 엣지 게이트웨이와 클라우드에 등록이 되어 있다고 가정하였을 때 디바이스의 데이터를 사용자가 사용할 수 있는 과정은 다음과 같다. 1) 디바이스는 측정된 데이터를 엣지 게이트웨이에 전송한다. 이 때

디바이스의 유일한 ID 정보도 함께 전달된다. 2~4) IPE는 디바이스가 전달한 PID를 확인하여 해당 측정 데이터를 저장한다. 5~6) 측정 데이터가 엣지 게이트웨이에 성공적으로 저장된 후, 엣지 게이트웨이는 PID와 연계된 EDID 사용하여 클라우드 IoT 플랫폼의 해당 리소스에 측정 데이터를 저장한다. 7) 사용자는 EDID를 사용하여 클라우드에 접근하여 사물인터넷 관련 서비스를 이용할 수 있다.

3.4 시사점 및 향후 연구 방향

엣지 게이트웨이의 기반의 인프라 구축을 위해서는 식별체계와 관련된 또 다른 해결해야 할 사항에 존재한다. 사물인터넷 시스템 개발에 있어 사물인터넷 디바이스의 라이프 사이클에 대한 연구 및 개발은 효율적인 디바이스의 자원관리를 통해 최적화된 시스템을 제공하기 위해서 필수적으로 고려되어야 할 사항이다. 잘 관리되지 않은 디바이스 라이프 사이클은 좀비 장치 (Zombie devices) 수의 증가로 이어지며, 네트워크 상에서 사용되지 않는 다양한 자원을 소유하고 있는 좀비장치는 비효율적인 네트워크 자원 사용을 초래한다²³⁾. 따라서 디바이스의 라이프 사이클 관리를 통하여 펌웨어 업데이트 및 환경 재구성을 통해 하드웨어의 최적의 상태를 유지함과 동시에 원활한 네트워크

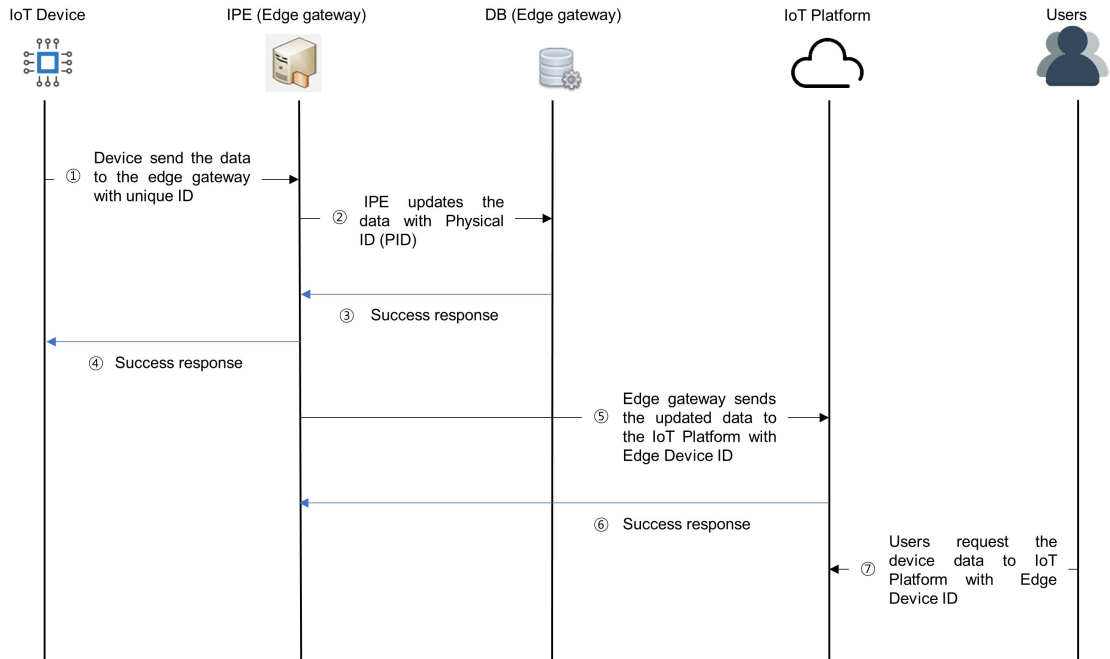


그림 4. 사물인터넷 디바이스 데이터 획득 절차
Fig. 4. Procedures of IoT device data retrieve

환경 제공을 해야 한다. 이에 대하여 ISO/IEC (International Organization for Standardization/International Electrotechnical Commission)는 식별체계 시스템을 효율적이고 효과적으로 구현해야하는 필요성을 해결하기 위해 식별체계 라이프사이클 및 프레임워크를 개발 하였다^[24-25].

IV. 관련연구

사물인터넷 환경의 엣지 게이트웨이에서 다양한 프로토콜 및 표준을 지원하고 그에 따른 디바이스 식별 및 앞 절에서 제시한 엣지 게이트웨이에서 발생할 수 있는 여러 문제점들을 해결하기 위하여 현재 여러 기업에서 컨소시엄을 구성해 유사한 연구들이 진행되고 있으며 가장 활발하게 연구하고 있는 컨소시엄들은 다음과 같다.

4.1 ETSI MEC

ETSI MEC (Multi-access Edge Computing) ISG (Industry Specification Group)는 2014년 9월에 설립되어 애플리케이션 및 콘텐츠 제공업체와 같은 네트워크 업체들이 엣지 컴퓨팅 기술을 활용하여 더욱 더 향상된 서비스를 사용자들에게 제공할 수 있도록 엣

지 컴퓨팅 표준화 작업 및 표준화 된 API (Application Programming Interface)를 제공하는 역할을 하고 있다^[26]. 기존의 Mobile-access Edge Computing에서 시작하여 주로 3G/4G와 같은 모바일 환경에서의 엣지 컴퓨팅 개념을 고려하여 작업이 진행되었으나, 사물인터넷과 같이 다양한 네트워크 환경에 따라 현재와 같이 Multi-access Edge Computing으로 프로젝트 이름을 변경하여 사물인터넷 환경을 포함한 다양한 네트워크 환경에서의 작업을 진행 중에 있으며, 엣지 컴퓨팅 환경에서의 요구사항, 유스케이스, 프레임워크에 대한 정의를 시작으로 현재 18개의 표준화 문서가 작업 중에 있다.

4.2 EdgeX

EdgeX는 리눅스 파운데이션 (Linux Foundation)에서 관리되고 델, 노키아, 인텔, 에릭슨, 레드햇 등과 같은 기업들이 해당 프로젝트에 참여하고 있는 엣지 컴퓨팅 관련 오픈소스 단체이다. 본 논문에서 언급한 문제점인 다양한 프로토콜 및 표준이 사용되는 사물인터넷 환경에서 해당 기술을 수용하기 위한 벤더 중립적 사물인터넷 엣지 공통 프레임워크를 만드는 것을 목표로 하고 있다^[27]. 해당 프로젝트의 핵심은 하드웨어 및 운영체제에 독립적으로 운용되는 계층에서

이기종 디바이스들의 데이터들을 수집하는 방법으로 파편화 된 시장을 통합하고 사물인터넷 솔루션의 배포를 가속화하는 것이다.

4.3 OpenFog

오픈포그 컨소시엄 (OpenFog Consortium)은 시스코, ARM, 델, 인텔, 마이크로소프트, 프린스턴 대학 등 현재 북미, 아시아, 유럽 전역에 57개 기관이 참여하고 있으며, 엣지 시스템과 연결된 클라우드 및 다른 엣지 시스템과의 충돌 없는 상호운용성 환경을 제공하기 위한 가이드라인 개발이 진행되고 있다. 따라서 오픈포그 컨소시엄은 서로 다른 환경의 (컴퓨팅, 저장, 네트워크) 디바이스 계층, 포그 컴퓨팅 계층, 클라우드 계층 간의 발생하는 요구사항 및 문제점을 정의하고 그에 따른 해결방안을 제시하여 투명성 있는 애플리케이션 개발을 지원하는데 초점을 두고 있다²⁸⁾.

V. 결 론

오늘날 도시의 다양한 문제를 해결하고 효율적인 인프라 관리를 위하여 사물인터넷 기술을 중심으로 한 다양한 서비스들이 시행되고 있으며, 기존의 클라우드 방식의 인프라 운영방식에서 발생된 문제점들을 해결하기 위하여 엣지 컴퓨팅 개념을 도입한 새로운 아키텍처를 기반으로 스마트 시티 인프라 관리 방안이 연구 및 개발되고 있다. 그러나 다양한 프로토콜 및 표준을 사용하는 사물인터넷 환경의 디바이스에 대한 고려 없는 운영환경에서는 이기종의 디바이스를 효율적으로 관리 할 수 없다. 따라서 본 논문에서는 이기종 환경의 스마트 시티 인프라 관리를 위하여 각 디바이스에 대한 식별체계를 분석 후 엣지 게이트웨이의 상호운용성 환경을 제공하기 위한 oneM2M 기반의 표준화 된 엣지 게이트웨이 아키텍처 및 식별체계를 제시하였으며, 또한 추가적인 연구사항을 제시하여 좀 더 발전된 환경의 인프라 관리를 위한 가능성을 제공하였다. 따라서, 해당 표준화 된 방법을 통하여 이기종 디바이스들을 일괄적으로 관리하여 효율적으로 스마트 시티에 연결된 디바이스들을 관리할 수 있을 것으로 예상된다.

향후 연구방향으로 본 논문의 추가 연구사항에서 언급한 보안 및 식별체계 재사용성에 대해 연구함과 동시에 실제 스마트 시티 환경에 적용하여 추가적인 검증 및 보안 계획에 있다.

References

- [1] B. Ahlgren, M. Hidell, and E. C-H. Ngai, "Internet of things for smart cities: Interoperability and open data," *IEEE Internet Computing*, vol. 20, no. 6, pp. 52-56, 2016.
- [2] A. Zanella, et al., "Internet of things for smart cities," *IEEE Internet of Things J.*, vol. 1, no. 1, pp. 22-32, 2014.
- [3] L. Sánchez, et al., "SmartSantander: Experimentation and service provision in the smart city," *2013 16th IEEE Int. Symp. WPMC*, Atlantic City, NJ, USA, Jun. 2013.
- [4] P. Sotres, et al., "Practical lessons from the deployment and management of a smart city Internet-of-Things infrastructure: The smart santander testbed case," *IEEE Access*, vol. 5, pp. 14309-14322, 2017.
- [5] J. Y. Hwang, et al., "Development and application of interoperability techniques with semantics for global internet of things (GIoTs)," *J. KICS*, vol. 42, no. 11, pp. 2208-2216, 2017.
- [6] B. Cheng, et al., "FogFlow: Easy programming of IoT services over cloud and edges for smart cities," *IEEE Internet of Things J.*, vol. 5, no. 2, pp. 696-707, 2017.
- [7] oneM2M Specification TR-0052, *Study on Edge and Fog Computing in oneM2M system*, <http://www.onem2m.org/technical/published-drafts/release-4>
- [8] 클라우드 씨앗, 엣지 컴퓨팅과 인공지능, https://www.nia.or.kr/site/nia_kor/ex/bbs/View.do?cbIdx=99863&bcIdx=20969&parentSeq=20969
- [9] H. Rajab and T. Cinkelnr, "IoT based Smart Cities," *IEEE 2018 ISNCC*, Rome, Italy, Jun. 2018.
- [10] N. Chen, et al., "Enabling smart urban surveillance at the edge," *IEEE SmartCloud*, New York, NY, USA, Nov. 2017.
- [11] Y. Wang, et al., "Network management and orchestration using artificial intelligence: Overview of ETSI ENI," *IEEE Commun. Standards Mag.*, vol. 2, no. 4, pp. 58-65, 2018.

- [12] *FogFlow: Orchestrating IoT Services over Cloud and Edges*, 11 June, Available at: <https://www.nec.com/en/global/techrep/journal/g18/n01/180110.html>
- [13] *TR-0044 Physical object heterogeneous identification and tracking in oneM2M*, V0.5.0, 7 Jun., Available at: <http://www.onem2m.org/technical/published-drafts/release-4>
- [14] S. E. Seong and J. W. Jang, "Trend and comparative analysis of LoRa technology for efficient sensing," in *Proc. KICS Winter Conf.*, pp. 1384-1385, 2019.
- [15] *LoRaWAN® Specification v1.0.3*, 10 Jun., <https://lora-alliance.org/resource-hub/lorawan-specification-v103>
- [16] W. Lee, et al., "사물인터넷 오픈소스 기술 -IoTivity," 정보와 통신 열린강좌 32.12 (별책 2호) pp. 27-35, 2015.
- [17] H. Aftab, et al., *Analysis of identifiers in IoT platforms*, Digital Communications and Networks, 2019.
- [18] J. Koo and Y.-G. Kim, "Interoperability of device identification in heterogeneous IoT platforms," *2017 13th IEEE ICENCO*, Cairo, Egypt, Dec. 2017.
- [19] J. Koo, S.-R. Oh, and Y.-G. Kim, "Device identification interoperability in heterogeneous IoT platforms," *Sensors*, vol. 19, no. 6, 2019.
- [20] J. Yun, et al., "Interworking of Onem2m-Based IoT systems and legacy systems for consumer products," in *Proc. ICTC*, pp. 423-428, 2016.
- [21] *oneM2M Specification TS-0001*, Functional Architecture V3.15.1, 13 Jun., <http://www.onem2m.org/technical/partner-transpositions>
- [22] "Unreserved Characters", "Uniform Resource Identifier (URI): General Syntax", IETF RFC 3986, <https://tools.ietf.org/html/rfc3986#section-3.2>
- [23] G. Soós, et al., "IoT device lifecycle - A generic model and a use case for cellular mobile networks," *2018 IEEE 6th Int. Conf. FiCloud*, Barcelona, Spain, Aug. 2018.
- [24] ISO/IEC 24760-1, *Information technology - Security techniques - A framework for identity management*, Part1: Terminology and concepts
- [25] Ericsson, *Cross domain identity of things*, 10 Jun., Available at: <https://www.ericsson.com/en/ericsson-technology-review/archive/2016/cross-domain-identity-of-things>
- [26] D. Sabella, et al., *Developing software for multi-access edge computing*, ETSI White Paper, vol. 20, 2019.
- [27] *EdgeX Project Introduction*, 10 Jun., <https://www.edgexfoundry.org/wp-content/uploads/sites/25/2018/09/EdgeX-Overview-091018.pdf>
- [28] Open Fog Reference Architecture for Fog Computing, 10 Jun., <https://www.openfogconsortium.org/>

황재영 (JaeYoung Hwang)



2015년 : 세종대학교 공학사
 2015년~현재 : 세종대학교 석박
 사통합과정
 2016년~현재 : 전자부품연구원
 자율지능 IoT 연구센터 위
 축연구원
 <관심분야> 사물인터넷, 5G,
 엣지 컴퓨팅, 머신러닝, 스마트 시티

손현서 (HyeonSeo Son)



2019년 : 대구대학교 공학사
 2019년~현재 : 세종대학교 석사
 과정
 <관심분야> 사물인터넷, 머신
 러닝, PID 제어

성 낙 명 (NakMyoung Sung)



2010년 : 한국외국어대학교 공학사

2015년 : 한국외국어대학교 공학석사

2011년~현재 : 전자부품연구원 자율지능 IoT 연구센터 선임연구원

<관심분야> 사물인터넷, 사물인터넷 플랫폼, 임베디드 시스템

김 재 호 (JaeHo Kim)



2017년 : 연세대학교 공학박사

2014년~현재 : TTA STC1/사물인터넷 융합 서비스 PG 의장

2014년~현재 : 사물인터넷포럼 기술분과 단말 WG 의장

2015년~현재 : 한국항행학회, 한국인터넷정보학회 이사

2019년~현재 : 전자부품연구원 자율지능 IoT 연구센터장

<관심분야> 사물인터넷 플랫폼, 센서네트워크, 무인이동체 네트워크

송 재 승 (JaeSeung Song)



1996년 : 연세대학교 공학사

2002년 : 서강대학교 공학석사

2013년 : Imperial College London 공학박사

2002년~2008년 : LG전자 이동통신연구소 선임연구원

2012년~2013년 : NEC Europe, 네트워크 연구소 선임연구원

2013년~현재 : 세종대학교 정보보호학부 부교수

2019년~현재 : The University of British Columbia Visiting Professor

<관심분야> 사물인터넷, 소프트웨어 테스트, 미래인터넷