

REST 기반 M2M 플랫폼 기술 연구

이창열^{1*}

¹동의대학교 컴퓨터공학과

A Study of M2M Platform Technologies based on REST

ChangYeol Lee^{1*}

¹Department of Computer Engineering, Dongeui University

요 약 M2M은 사물에 통신 모듈을 부착하여 양방향 서비스를 할 수 있는 체계로 사물과 플랫폼, 그리고 (웹 기반) 응용 서비스로 구성되었다. M2M의 장점은 통신이 가능한 곳에서는 어디서나 실시간으로 서비스가 가능하기 때문에 사물에 대한 이동 중 가시성과 자산 가시성을 제공할 수 있다. 그렇지만 개별적으로 구축하여 서비스하기 때문에 플랫폼의 호환성에 대한 연구는 부진한 상태이다.

본 논문에서는 이러한 M2M 서비스에 플랫폼의 호환성 기술에 대한 연구를 진행하였다. SOA보다는 매쉬업 서비스에 적합한 ROA 기술을, 식별에는 URL보다는 URN 기반의 Open API를 채택하였으며, 플랫폼과 응용 사이의 메시지 프로토콜로 MPM을 제시하였다.

Abstract M2M can provide the bi-directional services between applications and objects which have the attached telecommunication module. It consists of the objects, platform, and (web based) applications. M2M can realize In-Transit Visibility and Asset Visibility combined with GIS. Although many applications of M2M, such as metering, monitoring, controlling, tracking et al., are being serviced, the interoperability among the M2Ms is not considered.

In this paper, we described the interoperability. Specially, ROA, which is the state of art technology in mash-up, more than SOA, and URN, which is based on Open API in RESTful service, more than URL are studied. As the message protocol between platform and applications, MPM was defined.

Key Words : M2M, REST, mash-up, Open API, ROA, URN

1. 서론

사물지능통신(M2M; Machine to Machine)이란 사물에 통신 기능을 부착하여 사물의 정보를 언제나 실시간으로 전송할 수 있는 통신융합 서비스이다[1]. 2009년 기준 이동통신 가입자가 4,700만이며, 이중 134만개가 기계로 서비스를 하고 있으며, 기계의 모든 데이터를 서비스하는 통신 네트워크를 M2M이라 부른다[2]. 기존 음성 통신 가입자가 포화가 되면서 이동통신 사업자는 본격적인 M2M 사업의 확산을 수입모델로 하여 확충하고 있으며 Metering(검침, 환경), Monitoring(기계 설비, 시설물, 환경 오염), Control(가로등, 측사, 빌딩), Tracking(차량 관

제, 고가 자산, 대인 위치), Payment(VAN), Informing(도로전광판, Kiosk 등) 등에 대한 비즈니스 모델을 개발하고 있다[3]. 그러나 M2M 서비스의 확대는 M2M 서비스 시스템 사이 연계된 복합 서비스를 제공하기 하여 플랫폼 사이 호환성 있는 표준 체계를 제시하여야 하지만 아직은 미 정의된 상태이다. 이러한 체계를 구축하기 위하여 먼저 M2M 서비스에 대한 이해가 필요하다.

M2M 서비스는 다음과 같은 특징을 가지고 있다.

- 토탈 가시성(Total Visibility)

* In-Transit Visibility(ITV, 이동 중 가시성) : 기지국/GPS에 의하여 항상 오차 범위 내에서 이동 위치 파악이 가능하다.

본 논문은 교내연구과제로 수행되었음(2010AA178)

*교신저자 : 이창열(ley@deu.ac.kr)

접수일 11년 08월 03일

수정일 11년 08월 29일

게재확정일 11년 09월 08일

- * Asset Visibility(자산, 가시성) : 통신 모듈을 통하여 모든 자산(사물)에 대한 현황 파악이 가능하다.
- 매쉬업(mash up) 서비스
 - * 지도 또는 가시성 서비스와 응용 서비스가 결합된 서비스가 가능하다. 예를 들어, Metering, Monitoring, Tracking, Control, 등
- Resource Oriented Architecture(ROA)

M2M은 사물에 대한 지능 통신으로 사물 중심으로 서비스가 발생하고 있기 때문에 기업을 중심으로 비즈니스적 서비스의 핵심인 SOA (Service Oriented Architecture) 구조보다는 간단한 SOAP(Simple Object Access Protocol)을 사용하지 않는 ROA 기반 웹 서비스를 사용한다. 이는 매쉬업 서비스의 핵심 기술이다[4].

2. 기존 체계 연구

2.1 인터넷 식별 체계

인터넷 식별 체계는 URI(Uniform Resource Identifier) 기반으로 운영되며, 이는 URL(UR Locators)와 URN(UR Names)으로 구성된다.

URN 서비스는 다음과 같은 특징을 가진다 :

- 인터넷에서 자원을 위한 식별 체계는 URN에 기반하여 표현하여야 한다[5].
- URN은 궁극적으로 URL로 변환하여 서비스된다. 이때 1개의 URN이 여러 개의 URL로 변환할 수 있는 다중 변환(Multiple Resolution)이 제공될 수 있다.
- URN 서비스 시스템 구성[6]
 - * Root : URN 코드를 받아서 관련 RA(Registration Agency)를 찾아주는 서비스 기관
 - * RA : URN을 궁극적으로 URL로 변환 서비스하는 기관

대표적인 URN로 UCI(Universal Content Identifier)[7]가 있다. UCI 서비스는 NIA(한국정보화진흥원)가 2004년 개발하여 운영하다가 2009년 한국콘텐츠진흥원(KOCCA)로 이관되어 운영되고 있다(www.uci.or.kr).

2.2 M2M 서비스에서 사물의 식별 체계

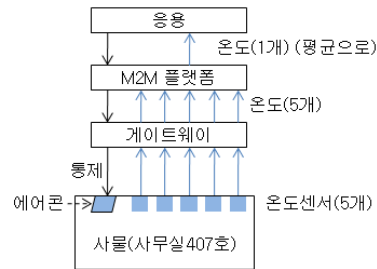
M2M에서 서비스되는 사물과 기기에 대한 식별은 다음의 3가지 특성을 지원해야 한다.

- 1 : 1 논리적 식별성 : 특정 사용자가 개통한 전화번호 "010-1234-5678"는 논리적 식별 번호이다. 왜냐

하면, 해당 사용자가 전화기를 바꾸어도 동일 전화번호 "010-1234-5678"을 가질 수 있기 때문이다. 이는 내부적으로는 USIM (Universal Subscriber Identity Module) 칩을 통하여 실현된다. 그러므로 응용 서비스에서 사물의 식별은 논리적으로 표현할 수 있다.

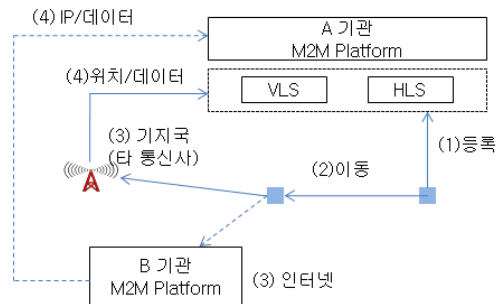
- 1 : N 논리적 식별성 : 그림 1에서 각 센서는 1개의 추상적 사물(예를 들어, 특정 건물, 시설하우스 등)에 부착된 센서이며, 이들 센서는 내부적으로는 물리적으로 식별되지만, 외부적으로는 논리적으로 식별될 수 있다. 다음과 같이 표현할 수 있다. "사무실 407호의 온도는 20도이다." 이때 20도는 사무실 407호에 물리적으로 부착된 센서 5개로부터 발생한 온도에 연산(예를 들어, 평균)을 하여 1개의 값으로 변환된 것이다. 그러므로 407호는 1개의 가상 온도 센서 A가 있는 것이며, 외부적으로 볼 때, A에 대한 식별만 필요하다. A는 내부적(M2M 플랫폼 내부)으로 5개의 센서에 매핑 될 수 있다.

만약에 온도 센서 5개 중에서 1개가 고장이 발생해도 가상 센서 A에 대한 식별에는 전혀 문제가 없다[8].



[그림 1] 센서와 서비스 구조
[Fig. 1] Sensors and Service Architecture

- 사물의 이동성



[그림 2] 사물의 이동성 지원 체계
[Fig. 2] Mobility Supporting System

- * M2M 서비스 사이 호환이 이루어진다면, 모바일 서비스의 로밍(Roaming)처럼 정보의 이동성을 지원하여야 한다.
- * 이동성 문제는 사물이 초기 어느 망을 통하여 접속하는가에 따라 그림 2와 같이 2가지 유형으로 분리될 수 있다[9].
 - 이동통신망을 통한 접속 : 이동통신사업자의 VLS(Visiting Location Server)와 HLS(Home Location Server)를 통하여 현재 위치 서비스를 하기 때문에 응용 시스템에서 특별히 문제가 없음.
 - 인터넷 망을 통한 접속 : mobile IP처럼 등록 IP(DNS)와 이동 IP를 가지고 서비스해야 한다.

표 1에는 이동에 따른 위치 관리와 데이터 관리를 위한 개념과 사례를 보여주고 있다.

[표 1] M2M에서 이동성 유형과 사례
[Table 1] Mobility Types and Samples in M2M

measures mobility	in-situ	dynamic
fixed	교각, 시설하우스	버스정류장, 출입구
mobile	렌트카, 개인용 당노측정	산불탐사비행체, 소방차를 통한 소방 대상 정보

2.3 사례 분석(iTag-CS 플랫폼)

2.3.1 사물과 플랫폼 사이 프로토콜

iTag-CS는 메타라이즈(주)(www.metarights.com)에서 운영하는 M2M 플랫폼이다[9]. 사물과 플랫폼 사이 메시지 프로토콜은 다양한 업체 장치를 지원할 수 있는 플러그인 구조로 되어 있다. 대표적인 메시지 구조는 표 2와 같다.

[표 2] 메시지 구조 사례
[Table 2] Sample Message Structure

내용	헤드	NMS	센싱 정보		모뎀 번호	프레임 길이	TAIL	
Byte	1	24	24	...	24	11	3	1
Desc	'\$'		ID 1번 센서정보	...	ID N번 센서정보		전체길이	'#'

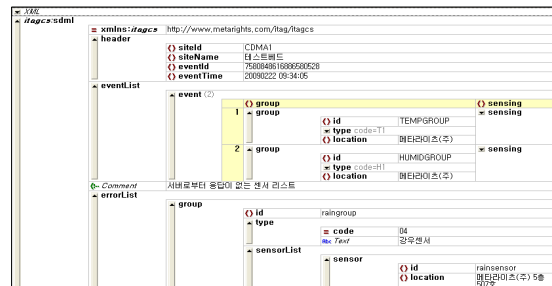
표 2의 내용을 설명하면 다음과 같다 :

- 헤드 : Frame Start Code (0x24)
- NMS : Network Management System (모뎀 상태정보 체크)

- 센싱정보 : MCU에 연결된 센서의 개수만큼 반복하여 전송한다. 한 개 센서정보는 24byte
- 모뎀 번호 : CDMA 또는 HSDPA 모뎀 번호
- 프레임 길이 : HEAD 부터 TAIL까지의 전체 바이트 수
- Tail : Frame End Code (0x23)

2.3.2 플랫폼과 응용 사이 프로토콜

메타라이즈(주)는 수신된 정보를 SDML (Sensing Data Markup Language) 구조에 맞추어서 서비스하고 있다 [10]. 그림 3과 같이 SDML은 header와 eventList, errorList의 3개의 엘리먼트로 구성되어 있다. header는 각 게이트웨이(또는 사물)를 식별할 수 있는 식별정보를 나타내며, eventList는 해당 게이트웨이에 등록된 모든 센서를 그룹별로 구분하여 해당 그룹의 센서정보와 센싱정보를 나타낸다. errorList의 경우는 해당 게이트웨이에 등록되었으나 정상적으로 동작하지 않는 무응답 센서 리스트를 나타낸다.



[그림 3] SDML 구조

[Fig. 3] SDML Structure

또한 응용(Users)에 정보를 제공하기 위한 방식으로 가입자 모드(subscription mode) 서비스를 하며, 4가지 프로토콜(HTTP, TCP/IP, FTP, SOAP)을 지원한다.



[그림 4] iTag-CS에서 전송관리 인터페이스

[Fig. 4] Message Protocol Interface

2.4 표준 프로토콜

TTA는 데이터에 대한 표준으로 “USN 서비스 표현 언어”(SSDL; Sensor Service Description Language)[11]만 제공하며, 상위 수준의 서비스 함수는 정의하지 않고 있다.

OGC(Open Geospatial Consortium)의 경우, SWE (Sensor Web Enablement) 표준 구조에서 데이터는 정보는 SensorML(센서 플랫폼 정보)[12], O&M(Observations and Measurements) (센싱 데이터 정보 표준)[13], GML(Geography Markup Language), SDL(Style Layer Description Language) 등이 있고, 웹서비스 표준으로는 SOS (Sensor Observation Service), SAS (Sensor Alerts Service), SPS (Sensor Planning Service), WNS (Web Notification Service) 를 제공하고 있다.

2.5 RESTfull 웹 서비스

□ 개념

REST 웹 서비스는 SOAP을 사용하지 않는 ROA 기반 웹서비스이다[14]. 구글은 2006년 12월 이후 SOAP 기반 검색 API를 중단하고 REST 기반 Ajax Search API를 제공하고 있으며, 2003년 Amazon 서비스의 2가지 인터페이스(REST vs SOAP)에 대한 점유율이 85% vs 15%로 REST 기반 웹 서비스가 대중성을 표현하고 있다. 즉, 기존의 OPEN API가 RPC 기반 SOAP에서 REST 기반 서비스를 바뀌어가고 있는 것이다[15]. REST는 웹의 리소스를 URI로 표현하고, 리소스를 구조적이며 유기적으로 연결하고, 비 상태 지형적 방법으로, 정해진 메소드만 사용하는 구조를 제공한다. 이러한 REST 원칙을 따르는 시스템을 RESTful이라고 표현한다[4].

□ 특징

- Addressability(주소 URI로 표현) : 모든 정보를 URI (실제적으로 URL로 표기하고 있음)로 표현한다. 정보 서비스가 특정 파일로 연계되고 해당 파일 내용을 통하여 서비스되면 RESTful하지 않는다.
- Connectedness(연결됨) : 예를 들어, 다음과 같이 표현한다.

```
<user>HongGilDong</name>
  <home>HongGilDong/home</home>
  <office>HongGilDong/office</office>
</user>
```

 HongGilDong은 자신과 관련된 리소스(home, office)를 연계하여 표현하고 있다.
- Statelessness (상태 없음) : 클라이언트 상태를 서버에서 관리하지 않는다. 모든 요청은 1회성 성격으로 세션을 유지하지 않는다.
- Homogeneous Interface(동일한 인터페이스) : 제공되는 4개 메소드와 추가적 2개 메소드를 이용하여 모든 동작을 정의한다.

- * 리소스 조회 : GET
- * 새로운 리소스 생성 : POST
- * 리소스 갱신 : PUT
- * 리소스 삭제 : DELETE
- * 리소스 메타데이터 보기 : HEAD
- * 리소스 지원 메소드 체크 : OPTION

□ RESTful Open API 표현 및 사례

- 표현 : 서버주소 + 서비스이름 + 자원
- 사용 사례 :
 요청 : `http://wisefree.com/employee/123`
 결과 :

```
<employee-ID>123</employee-ID>
<Name>Hong, GilDong</Name>
```

```
<Resume>http://wisefree.com/employee/123/resume
</Resume>
```

3. REST 기반 센서 서비스 연구

3.1 M2M 식별자 정의

M2M 서비스를 위한 명명 규칙과 표현을 URN 표준에 맞추어 정의하였다.

- 이름 : MRI (M2M Resource Identifier)
- 구조 : urn:mri:플랫폼코드.서비스기관코드.사물코드 (자원코드)

3.2 RESTful 웹 서비스 정의

전형적인 RESTful Open API는 URL로 서비스를 제공한다. 예를 들어, 렌트카에 대한 모니터링 M2M 서비스를 위한 Open API 표현은 다음과 같다.

- (1) <http://m2m.skt.com/rentcar> : 렌트카 업체의 정보 전체를 얻어온다.
- (2) <http://m2m.skt.com/rentcar>/서초지점 : 보유 차 중에서 서초지점의 정보를 얻어온다.
- (3) <http://m2m.skt.com/rentcar>/대여차 : 보유 차 중에서 대여된 차 정보를 얻어온다.

이를 URN 기반 체계로 다음과 같은 구조로 각각 대응하여 제안한다(렌트카 서비스 사례).

- (5) urn:mri:vrn2m.skt01.rentcar
- (6) urn:mri:vrn2m.skt01.rentcar.서초지점
- (7) urn:mri:vrn2m.skt01.rentcar.대여차

(6)번을 MRI 구조에 맞게 해석하면 다음과 같다. M2M 서비스하는 Root(vrn2m; 플랫폼 코드)를 통하여 RA(서비스 기관(SKT01, 서비스기관코드))에 MRI가 전달되면 SKT01에서는 해당 ID의 사물정보(rentcar, 사물 코드)를 확인하고, 서비스 사항(“서초지점”, 선택적으로 자원 코드)을 확인후 저장된 URL 형태 정보를 돌려준다.

위와 같이 URN 형식의 표기는 다음과 같은 장점을 가진다.

- 신뢰성 : Open API가 "서버주소"를 기반으로 이루어졌기 때문에, "서버주소"가 변경되면 기존의 Open API 체계를 변경해야하나 URN 기반의 체계는 변경할 필요가 없다.
- 호환성 : 사물을 식별하는 체계와 서비스하는 Open API가 동일 개념의 다른 표현으로 처리됨으로써 정보 표현과 관리가 단순하다.

렌트카 적용 사례로써 차량에 대하여 다음과 같이 ID(MRI)를 부여할 수 있다.

- (8) urn:mri:vrn2m.skt01.rentcar.서울1234

위의 ID는 RA(skt01) 시스템 내부적으로 해당 정보를 얻을 수 있는 URL로 매핑되는데, 다음과 같이 4가지 유형으로 다중 변환되어 필요한 정보를 서비스 할 수 있다 :

- 서울 1234 차량 전체 정보 URL
- 서울 1234 차량 위치 정보 URL
- 서울 1234 차량 대여자 정보 URL
- 서울 1234 차량 운행 정보 URL

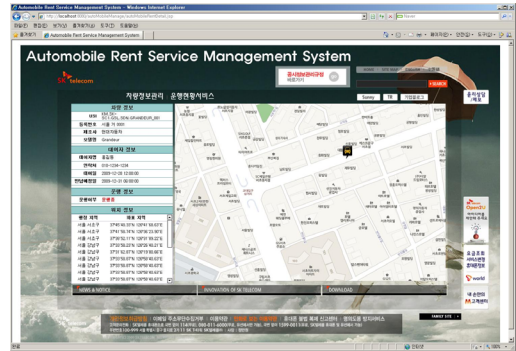
위의 변환 4가지 종류는 각각 Open API로 다음과 같이 표현될 수 있다.

- (9) http://m2m.skt.com/rentcar/서울1234
- (10) http://m2m.skt.com/rentcar/서울1234/위치
- (11) http://m2m.skt.com/rentcar/서울1234/대여차
- (12) http://m2m.skt.com/rentcar/서울1234/운행

만약에 Open API 서비스를 제공하는 서버의 위치가 바뀌어도 URN 체계는 바뀐 형태를 제공하기 때문에 항상 Open API를 제공받을 수 있다(이는 URN의 특징이기

도 하다). 예를 들어, (9)의 서버 위치가 http://abc.skt.com/으로 바뀌면 MRI의 변환 결과가 (9)인 경우 (9)로 바뀌어 제공된다.

- (9) http://abc.skt.com/rentcar/서울1234



[그림 5] 차량 서울1234의 매쉬업 서비스
[Fig. 5] Mash-up Service for "서울1234" car

Open API를 이용한 서비스 결과는 그림 5와 같이 제공될 수 있다. 그림 5에는 차량 정보, 대여자 정보, 운행 정보가 보여지고 있다.

3.3 메시지 구조

기존 메시지 구조[9][11][13]를 비교하면 표 3과 같이 분석될 수 있다.

[표 3] 기존 메시지 정보 비교
[Table 3] Message Data Formats

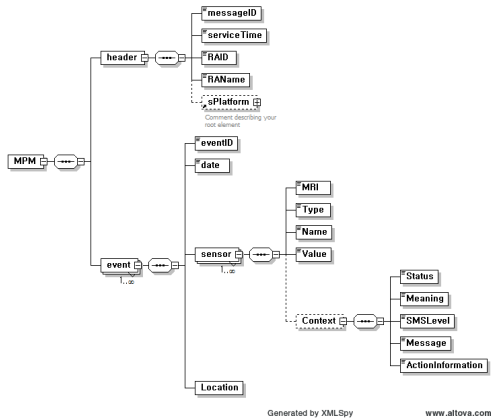
메시지 항목	O&M (OGC)	SSDL (TTA)	SDML (메타라이프)
기본 조건	SAS	USN 메타데이터	CDML
주 목적	센싱 정보 교환	센싱 관련 환경 기술	센싱 정보 교환
특징	물리적 정보 표현	메시지 수신, 요청자 포함 정보	센서 그룹 기반 추상화
추상화	물리적 정보	물리적 정보	물리적/논리적 전부 지원
데이터 요청 방법	별도 SOS 사용	메시지에 포함	가입자 방식, 별도 정의.

- OGC의 O&M 은 센서 시스템에서 Platform으로 제공하는 정보에 가까워서 M2M 플랫폼 메시지와는 차이가 있으며

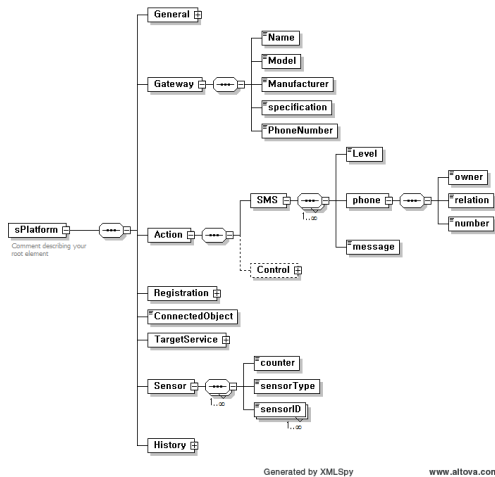
- TTA 표준으로 제시된 SSDL은 데이터를 주고 받는 정보에 가깝고 물리적, 논리적 센서 환경이 부족하며
- 메타라이즈(주)의 SDML은 서비스 플랫폼 개념이 약한 상태이다.

본 연구에서는 메시지 프로토콜로 MPM(M2M Platform Message)을 제시한다. MPM 구조는 [그림 6]처럼 'header' 정보와 'event' 정보로 구성된다. 'event'는 사물에서 발생하는 데이터를 기술하는 부분으로 한번 XML로 데이터를 보낼 때 여러 개의 'event'를 보낼 수 있게 설계되었다.

각 'event'는 데이터 유형(센서 유형)에 따라 반복되면, 발생한 데이터 값에 추가적으로 "Context"가 부가된다. 이는 발생한 데이터의 분석된 값을 의미하는 것으로 예를 들어, "정상", "오류", "비상" 등의 값을 가지는 것이다.



[그림 6] MPM 메시지 구조
[Fig. 6] MPM Message Structure



[그림 7] [그림 6]의 서비스 플랫폼 부분
[Table 7] Service Platform Part of [Table 6]

[그림 7]은 "서비스 플랫폼"에 대한 정보이다. 이는 MRI 구조에서 언급한 "플랫폼 코드"에 해당되며, 사물 자체 정보, 사물에 부착된 게이트웨이 정보, 센서 리스트(온도, 습도, 위치, ...), 통제 리스트(에어콘, 창문 개폐, 스프링클러, ..) 등에 대한 정보를 기술하는 것으로 SensorML[12]에서 언급하는 플랫폼과 구별하여, "서비스 플랫폼(sPlatform)"으로 정의하였다. 이는 필요 시 MPM에서 선택적으로 참조하여 사용할 수 있게 구성되었다.

4. 결론

M2M 서비스는 전통적인 수동적 사물을 능동형 사물로 바꾸어 서비스하는 획기적인 서비스 체계이다. 전통적 사물에 통신 모듈과 선택적인 센서 모듈을 부착하여 사물의 상태 정보를 전송하여 조치를 취할 수 있게 하는 것으로 이동통신사업자의 새로운 BM(Business Model)이며, 누구나 M2M 플랫폼을 구축하여 서비스할 수 있다.

이러한 서비스가 융합, 확대되려면 M2M 플랫폼 사이에 호환성이 기반이 되어야 하고, 이는 기본적인 사물에 대한 식별, 그리고 가벼운 구조의 Open API 체계, 그리고 데이터 구조의 표준에 기반한다.

본 논문에서는 사물에 대한 추상적(논리적) 표현과 이를 기반으로 URN 형태의 사물의 식별 체계(MRI)를 제시하였다.

또한 M2M 플랫폼 사이 호환과 활용을 위한 웹 서비스는 REST 기반의 고정된 4개(필요시 2개 추가) 메소드에 기반한 체계를 제시하였다(이는 RESTful 웹 서비스에서 정의된 것임). SOA가 기관 사이 비즈니스에서 필요한 기능에 대한 웹 서비스를 정의하여 운용되는 것인 반면, M2M은 메소드보다는 자원 중심의 서비스이기 때문에 메소드를 HTTP에서 제공하는 4개 기능을 기반으로 한정함으로써 웹 서비스를 간략화하는 방식으로 RESTful 서비스를 채택하였다.

RESTful 서비스의 Open API를 URN 체계의 MRI 기반 Open API를 제시하였다. 즉 MRI -> Open API로 변환할 수 있는 체계가 마련된 것이다. 이는 Open API 구문에서 서버의 위치가 바뀌어도 서비스가 변화하지 않게 정의하는 것이며, 또한 궁극적으로 사물의 식별이 사물의 질의와 일치(예를 들어 "차량 1234"의 식별은 "차량 1234"에 대한 질의와 개념적으로 동일한 Input이다)한다는 개념에 기반 하였다.

마지막으로 M2M 플랫폼의 데이터 표준으로 MPM을 제시하였는데, 이는 기존의 TTA 자료, OGC 표준, 메타라이즈(주)의 SDML을 고려하여 M2M 서비스에 적합하

게 제시한 것이다.

References

- [1] ChangYeol Lee, "Identification and Management Framework for M2M", M2M Technologies and Prospect Seminar, Korea Communication Commission, June, 2009.
- [2] JungSu Song, "General Plan for M2M", M2M Forum Kick-Off Seminar, November, 2009
- [3] WooYong Kim, "M2M service examples and plan", M2M Forum Kick-Off Seminar, November, 2009
- [4] Jonathan Douglas Latherm, SA-REST: Adding Semantics to Rest-Based Web Services, Master Thesis, University of Georgia, 2007
- [5] Moats, Ryan, "URN Syntax", RFC-2141, May 1997.
- [6] K. Sollins, Architectural Principles of Uniform Resource Name Resolution, IETF RFC 2276, January 1998
- [7] NIA, UCI Specifications 2.0, 2005
- [8] ChangYeol Lee, "Global Identification for IP-USN application", *KOREN Workshop*, may 2009
- [9] NIA, "A study on the identification framework and security/reliability in IP-UNS", Research Report, December 2009
- [9] HanChae Jung, "The implementation of USN Context-Aware Middleware Engine" Research Report No. 7002369, KIAT, Ministry of Knowledge Economy November, 2008
- [11] TTA, "USN Sensor Service Description Language", TTAK.KO-06.0199, December, 2008.
- [12] OGC, "OpenGIS Sensor Model Language(SensorML) Implementation Specification", Version 1.0, 2007-07-17
- [13] OGC, "Observations and Measurements (O&M)", Version 1.0, 2007-12-08
- [14] Roy Thomas Fielding, "Architecture Styles and the Design of Network-based Software Architectures, University of California, Irvine, 2000
- [15] ChangHwan Han, "The design and implementation of the efficient REST web service", Master Thesis, Korea University of Technology and Education, 2009

이 창 열(ChangYeol Lee)

[정회원]



- 1985년 2월 : 고려대학교 수학과 (이학사)
- 1991년 2월 : 고려대학교 전산과 학과 (이학 석사)
- 1997년 6월 : 파리 7대학교 전산과 박사
- 2000년 3월 ~ 현재 : 동의대학교 컴퓨터공학과 교수

<관심분야>

RFID, USN, URI, 메타데이터