편집위원 : 이승윤 (ETRI)

Web of things 기술 개요 및 분석

인민교, 이승윤
한국전자통신연구원

요 약

최근 통신 기술 및 IT기술을 발전이 급속히 이루어지면서 산업 및 사회 전반에 통신을 빼놓고는 일상 생활을 논할 수 없는 환경이 되었다. 특히 수만은 종류의 모바일 단말들이 널리 보급되면서 언제 어디서나 정보를 접근하고 획득할 수 있는 환경이 구축되어 있으며, 실시간 정보를 들을 접할 수 있는 환경이 조성되었다. 특히 통신 관련 장치가 소형화되고 저전력 기술이 적용되면서, 전에는 접할 수 없었던 특정 사물(컴퓨터, 프린터, 가전, 액츄에이터, 동영상 등)에 대한 네트워크적인 접근이 가능하며, 심지어 각종 센서 정보를 원격을 점검할 수 있게 되었다. 이러한 변화는 실 세계의 사물을 응용 단의 하나의 자원으로 활용할 수 있는 환경이 조성되었으며, 특히 웹 기술을 이용해서 자원을 검색하고 접속/제어하려는 작업이 즉, Web of Things(WoT)개발이 이루어지고 있다. 본 논문에서는 WoT에 대한 개념 및 기술적이슈/표준화적 이슈에 대해서 논한다.

Ⅰ. 서 론

웹의 근본적인 시작은 문서 즉 텍스트의 공유에서 시작되었다고 볼 수 있다. 하나의 문서를 작성하고 이를 웹 표준 기반의 기술로 무장된 브라우저를 통해서 언제든지 같은 내용을 보게 된 것이다. 자연히 웹 기술을 더욱 발전하여 텍스트뿐만 아니라, 동영상, 금융, 게임 등 무수한 자원들이 웹의 영역에 포함되고 활용되고 있다. 이러한 웹의 기술 발전과 동시에 주목해야 할 것은 실제 사물 세계의 변화이다. 최근에는 장비 자체 기술뿐만 아니라 통신 기술을 발달로 인하여 장비 자체의 기능 및 서비스가 네트워크의 일부로서 연결되고 접속 및 통신이 가능해지고 있다. 각각의 기기들에는 고유의 통신 프로토콜이 설치되고 이를 관리하는 전문 인력 및 장비를 투입하여 원격으로 장비를 접근 제어하고, 원하는 정보를 타 지역에서 접할 수 있는 것이다. 실제 이러한 욕구는 점차 강해져 모든 사물에 원격 검색/제어/감시 등의 원하는 사회로 진행하고 있다. 더불어 소비자들은 어떤 장비를 가지고 있던지 똑같은 서비스를 제공 받고자 하고, 언제 어디에 있던지 서비스를 이용하고자 하는 요구가 늘고 있다. 그러나 이를 위해서는 선결해야 할 중요한 여러 가지 과제가 발생한다. 우선 실제 사물을 이용해서 서비스를 개발하고자 할 때, 서비스에 활용될 사물에 대한 전반적이 이해가 필요할 것이다. 즉 사물의 통신 프로토콜, 동작 방식, 인터페이스 등에 대한 이해가 선결되어야 한다. 다시 말해 서비스 개발자는 어떤 곳이나 어떤 장비에서도 통신 및 제어가 가능한 서비스를 만들고자 할 때, 서비스에 활용되는 각종 장비 모두에 접근하고 활용할 수 있도록 프로그램을 작성해야 할 것이다. 때에 따라서 수십 수백 종의 장비가 활용 될 수도 있다. 각각의 장비는 고유의 시스템이 동작하고 고유의 프로그램과 인터페이스로 연결되어야 한다고 하면, 그러한 서비스를 오랜 시간과 많은 해당 전문 기술 인력이 투입되어야 할 것이다. 따라서 이러한 문제점을 해결할 수 있는 기술이 웹을 활용하는 것이다. 웹의 장점은 무수히 많지만 사물 웹의 관점에서 보면, 웹은 브라우징이 가능하며, 하부 네트워크에 상관없이 HTTP프로토콜을 활용하여 통신할 수 있으며, 각종 방화벽을 넘나들 수 있다. 또한 서비스의 검색 기능을 제공할 뿐만 아니라, 네트워크의 생명력이 뛰어나 어느 한 부분이 활용 불가능하더라도 우회 네트워크 (loosely coupled network) 할 수 있으며, 무엇보다도 요즘 개발자들에게 무척 친숙한 기술에 속한다. 따라서 모든 사물을 웹의 일부로 접근/제어 할 수 있도록 사물 웹을 구현한다면 실제 유비쿼터스 세상을 앞당기는데 한 역할을 할 수 있을 것이다 [1][2].

〈그림 1〉에서 일반적인 사물 웹(Web of Things)의 개념을 살펴 볼 수 있다. 실제의 사물이 웹에 통합되고 이들은 단지 웹상에서 접근 가능한 하나의 서비스로 보여진다. 각각의 서비스를 이용하여 새로운 매쉬업 서비스의 지원도 가능해 지며, 응용개발의 입장에서 보면 단지 각 서비스의 URI를 이용하여 서비스 개발 및 이용 가능해서 지는 것이다. 물론 웹을 이용한 검색도 가능하다.

WoT에서 고려하는 사물의 범위는 그림1에서와 같이 특정한 물리적인 장비에 국한되지 않는다. 웹 상에서 구별 가능한, 즉

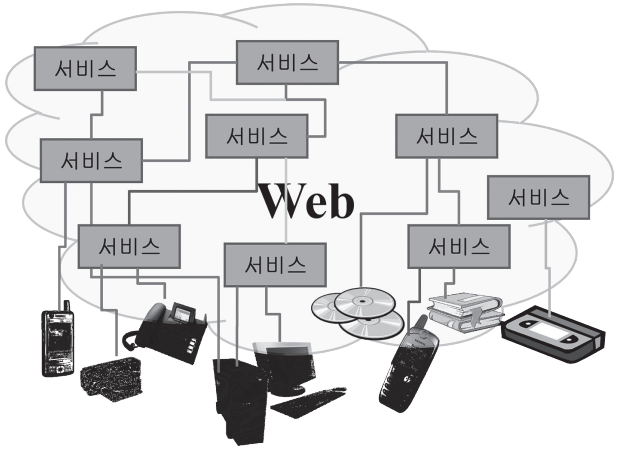

그림 1. 사물 웹 (WoT)의 개념

URI로 식별 가능한 리소스를 통칭한 것이라고 보면 된다. 실제 서비스 개발에 논리적인 컨텐츠(예, 동영상) 등도 하나의 자원화하여 사용할 수 있다.

이에 본 논문에서는 본론에서 WoT의 기술적 이해 관점에서 사물을 웹에 접속 시키는 방법, 사물을 기술하기 위해 고안된 메타데이터, 웹 상에서 사물을 검색할 수 방법, WoT매쉬업에 대해서 논하고, 마지막으로 WoT의 각 국제 표준화 이슈사항을 논할 것이다.

II. Web of things 관련 기술

1. WoT 접속 기술

사물을 웹에서 접속/제어 할 수 있도록 사물을 웹의 일부로 만들기 위한 방법에는 크게 2가지를 들 수 있다. 하나는 직접적으로 사물에 접근/제어하는 것이며, 다른 하나는 중간에 게이트웨이나 브로커를 두어 하는 방법이다. 첫 번째 방법은 사물에 웹 서버가 직접적으로 올라가 있는 경우로 사물웹 입장에서 보면 아주 바람직한 방법이다. 특별히 사물을 웹에서 제어하기 위한 특별한 장치 또는 프로토콜이 등이 필요 없으며, 사용자가 웹 기술만을 이용해 직접적으로 다룰 수 있기 때문이다. 이방법은 기술을 발전으로 인하여 경량의 임베디드 웹 서버가 더욱더 개선되고 이러한 장비가 늘어날 경우 향후에는 보편적인 방법이 될 수 있을 것이다. 그러나 이러한 방법이 모든 사물에는 적용될 수 없으며, 직접적으로 제어가 불가능한 경우에는 중간에 고유 사물과 웹 간의 기능을 중재하는 브로커나 게이트웨이 서버가 필요하며 두 번째 경우가 이에 해당된다.

웹을 통해 고유 프로토콜을 사용하는 센서나 화상 장치 등을 이용하고자 할 때 브로커는 웹 인터페이스로 요청을 받고, 이

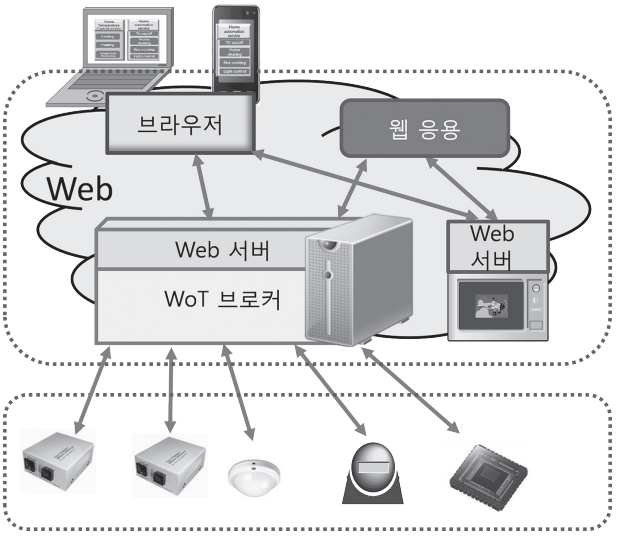

그림 2. WoT 장비 접근 방법

를 해당 장치에 사용되는 고유의 프로토콜로 변환하여 요청 사항을 송신/수신하게 된다. 또한 수신된 결과물을 웹 프로토콜로 전달 가능하도록 변환하여 요청 측에 보내게 된다.

〈그림 2〉에서 보면 자체에 웹 기능이 없는 장치의 경우 브로커를 통해서 서비스가 제공되고 장치에 임베디드 웹 서버가 있는 경우 직접적인 접근 제어가 가능한 모델을 볼 수 있다.

1.1 사물 웹 브로커

현재 실시간 WoT 검색 엔진 등 다양한 사물웹 게이트웨이 기술이 제안되고 있다. 〈그림 3〉은 사물을 웹에 통합시키기 위한 사물웹 브로커의 구조를 보인 것이다. 사물 웹의 구조의 핵심은 WoT 브로커로서 WoT 브로커는 크게 2개의 계층 즉, 서비스 계층과 변환계층으로 분류되며, 총 6개의 기능 개체(functional entity: FE)와 다수의 에이전트로 구성된다. 서비스 계층은 물

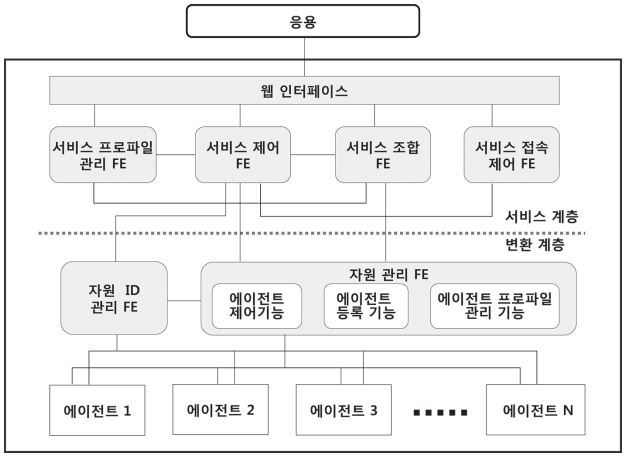

그림 3. 사물 웹 브로커 구조

리적인 서비스를 응용에서 정의된 웹 인터페이스를 통하여 접근 가능하도록 정의하고, 체계화하는 서비스로 제공하며, 변환 계층은 웹 접근이 불가능한 고유의 서비스를 웹 인터페이스를 통하여 접근/제거가 가능하도록 장비 고유의 인터페이스를 웹 인터페이스로 변화하는 기능을 담당한다.

각각의 기능에 대한 세부적인 내용을 아래와 같다.

- 서비스 프로파일 관리 FE: WoT브로커를 통해서 제공되는 모든 서비스의 프로파일 정보를 저장하고 있다. 주로 서비스의 유형(예, 감시 서비스, 출력 서비스, 조명 제어 서비스, 알람 서비스 등), 서비스의 이름 및 서비스 제공자 정보등이 저장되어 있다. 서비스 제어 FE와 서비스 조합 FE와 협력하여 서비스를 검색하거나 조합할 때 활용된다.
- 서비스 제어 FE: 실제적인 서비스 실행, 검색, 등록 및 삭제 등의 기능을 수행하며 WoT 브로커의 전체적인 서비스를 관리한다. 응용에서 브로커의 서비스를 요청할 때 제일 먼저 도달하는 FE이다. 서비스 제어 FE는 응용의 접속 권한과 접근을 관리할 수 있다.
- 서비스 조합 FE: 두 개 이상의 기존 서비스를 이용하여 새로운 서비스를 생성하는 기능을 가지고 있으며, 서비스 제공자에 의해서 제공된다. 새롭게 종합되는 서비스 또한 서비스 프로파일 관리 FE에 저장된다.
- 서비스 접근 제어 FE: WoT에 접속하거나, 서비스를 받고자 하는 응용이자 사용자 및 관리자에 대한 접근 허용 관련 정보를 가지고 있다.
- 자원 관리 FE: 실제적인 자원을 관리하는 기능들을 탑재하고 있다. 주요 기능은 에이전트 제어, 에이전트 등록/삭제 관리 및 메이전트 프로파일 정보 관리 3가지로 분류할 수 있으며, 해당 에이전트에 대한 접근/제어/실행 등에 관한 권한을 갖는다.
- 자원 ID 관리 FE: 물리적인 자원의 ID와 자원이 속해 있는 에이전트의 ID를 저장하고 있으며, 자원과 에이전트의 매핑 정보를 가지고 있다. 자원 ID관리 FE는 또한 하부 네트워크의 ID와 유형 및 서비스 위치 정보를 저장한다.
- 에이전트: WoT 서비스와 웹 접속 및 제어가 불가능한 장비 사이에서 브릿지 역할을 수행하며, 하부 네트워크와는 네트워크 고유의 통신 방식으로 정보를 주고받으며, 이러한 정보를 웹 인터페이스 형식으로 변환하는 기능을 담당한다 [7] [11].

2. WoT 검색 기술

어떻게 웹을 통해 사물을 접근/제어하는가의 문제와는 별개로 향후에는 이렇게 생산되고 웹 상에서 접근 가능한 사물 기능을 어떻게 검색할 것인가 또한 하나의 이슈이다. 궁극적으로는 현재의 웹과 같이 기존은 검색 엔진을 활용하는 것이 바람직할 것이다. 그러나 웹 문서와는 분명이 차이점이 존재할 수 있다. 웹 문서의 경우에는 인간이 식별할 수 있는 텍스트 등의 컨텐츠 등이 주를 이루지만, 사물의 경우에는 위도/경도, 고도 또는 소유주 등 상황정보가 중요한 요소이며, 또한 웹 문서와는 다르게 사물을 대상으로 함으로 상황에 따른 빈번한 상태 정보(예, 모바일 장치의 위도/경도) 같이 변화될 수 있기 때문이다.

2.1 사물의 메타 데이터 모델

사물을 웹에서 검색하기 위해서는 사물 자체와 그들의 서비스를 기술 할 수 있는 방법이 먼저 논의 되어야 한다. 현재 이러한 관점에서 관련 메타 데이터 모델이 제안되고/논의되고 있으며, 크게 정적인 속성과 동적 속성으로 구분하여 다음과 같이 분류하고 있다 [3].

- 정적 속성
  - 제품: 사물이 자체의 관점에서의 속성
  - 서비스: 사물이 제공하는 서비스를 관점에서 속성
- 동적 속성
  - 위치: 사물이 위치한 절재 위치에 대한 속성
  - QoS: 얼마나 잘 수행되었고, 수행되는가에 대한 속성

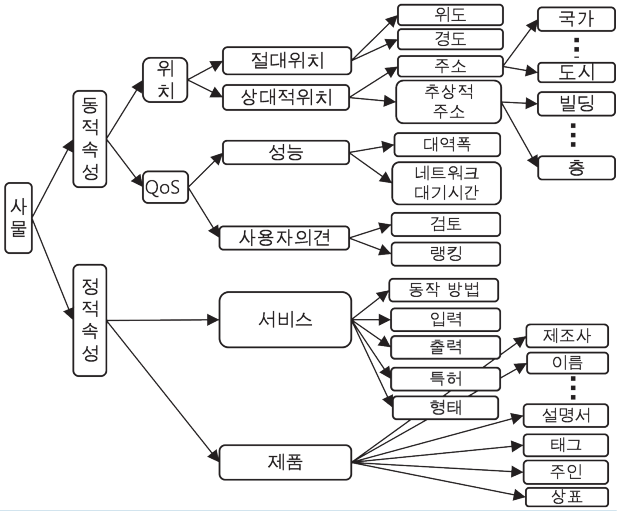

그림 4. 사물 웹 메타 데이터 모델 예제

〈그림 4〉는 사물웹 메타데이터 모델에 해서 세부적인 분류에 제를 볼 수 있다. 각각에 대해서 좀더 세밀이 개발되고 제안되는 속성을 살펴보면 다음과 같다.

- 제품 속성: 제품에 대한 속성을 나타내는 메타데이터로는 hProduct 마이크로포맷이 주로 논의 되고 있으며, 이를 이용하여 웹 브라우저와 검색 엔진 및 웹 응용은 적합한 제품 선택을 할 수 있다. 또한 관련된 제품의 소유주 및 제조사에 관련된 것은 hCard 마이크로포맷이 주로 논의 된다.
- 서비스 속성: 서비스 속성은 WoT에서 중점적으로 고려되는 RESTfull API특성에 기인하여 크롤러(crawler)와 HTTP프로토콜을 이용해서 서비스 정보를 획득 할 수 있다. 그러나 크롤러는 REST 자체의 제약조건으로 인하여 모두를 파악할 수는 없으며, 많은 HTTP 콜(call) 유발 시키는 단점이 있다. 따라서 이를 보완하기 많은 추가적인 연구가 이루어지고 있으며, hREST 마이크로포맷이 이에 대한 해답을 제공한다. hRESTs는 크롤러 없이도 서비스 메타데이터를 제공할 수 있으며, 결과적으로 HTTP 콜을 감소시킬 수 있다.
- 위치 속성: 위치에 대한 속성을 표현하기 위해서는 hCard와 geo마이크로포맷이 논의된다.

사물 요소	마이크로포맷	속성	의미
위도	geo	위도	사물의 위도
경도	geo	경도	현재 경도
주소	hCard	거리 주소, 지역, 우편번호, 국가명	우편 주소

- QoS: 무수히 많은 사물이 웹 환경에서 검색 및 활용 되기 위해서 사물의 QoS 정보 즉, 대역폭, 업로드 시간, 평균 응답 시간 등은 매우 중요한 선택기준의 하나이다. 더불어 UX-(User Experience)의 고려 또한 중요한 기준이다. 현재 이를 고려한 마이크로포맷은 hReview 제안 되고 있다.

이러한 마이크로 포맷을 이용하는 것은 웹 환경에서 많은 장점이 있다. 우선 기존의 검색엔진 즉, Google, Yahoo 등에서 직접적인 검색이 가능하며, 이러한 검색 엔진은 메타데이터를 콘텐츠 검색 결과의 제공에도 활용할 수 있다. 또한 검색 엔진은 또한 사물의 메타데이터 값을 이용해서 전혀 다른 종류의 결과를 도출해 낼 수 있다 [3].

2.2 WoT 서비스 검색 및 룩업서비스

웹을 통한 검색 서비스는 이미 보편화 되어 있고 많은 기술들이 개발되어 있다. 따라서 WoT장치(사물) 및 그 서비스를 기존 검색엔진에 기반하여 검색한다면 편리성이 높아질 것이다. 그러나 기존의 검색엔진은 문서 검색 등에 초점이 맞춰져 있는 것으로 WoT 에서 고려하는 사물로 영역을 넓히면 사물의 특수성으로 인하여 많은 새로운 문제점이 도출된다. 특히 위치 정보가 중요한 사물의 경우에는 컨텍스트가 빈번히 변경되어 웹 문서와는 다른 문제점을 도출 될 것이다. 현재 이에 대한 많은 기술들이 제안되고 있다. 그 중 Sensor ML 기술은 센서와 네트워크 응용과 장치에 대한 기술하는 표준 XML 언어이며, DPWS는 실 사물의 의미 정보를 기술하는 디바이스 메타데이터 언어를 제공한다.

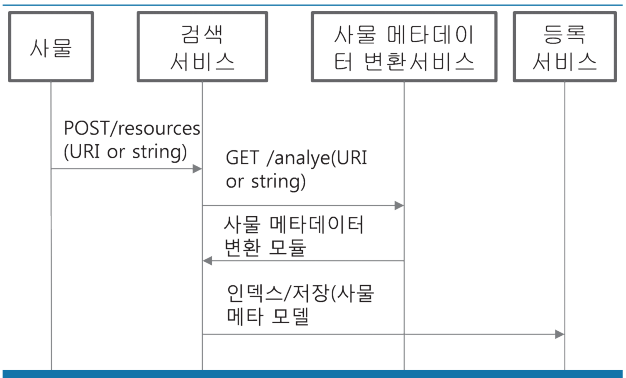

그림 5. 사물 웹 검색 절차

사물의 검색 프로세스는 사물이 WoT의 구조에 참여로 시작된다. 사물이 로컬네트워크에 접속하며 사물 또는 사물 브로커는 POST 요청을 전송한다. 사물은 자신의 루트 URL 또는 리소스를 기술한 페이로드를 보낼 수 있으며, 검색 서비스는 이를 사물 메타데이터 변화서비스로 보낸다. 사물메타데이터 변환서비스에서는 시멘틱 메타데이터를 추출하며 결과를 검색서비스 측에 전달한다. 검색 서비스는 이를 해당 URI와 바인딩 시키며, 최종적으로 등록 서비스에 등록 요청한다. 또한 WoT룩업 서비스는 사용자 즉, 개발자에게 사물을 검색하고 이를 활용하여 응용을 개발할 수 있도록 인터페이스를 제공한다.

2.3 WoT 매쉬업

웹 기능을 통해 접근 가능하도록 구현된 다양한 사물들은 개별적으로 활용될 수 있을 뿐만 아니라 여러 개의 조합을 통해 다양한 응용으로 재생산이 가능하다. 예를 들어 지도 정보를 제공하는 웹 API와 GPS 정보를 제공하는 웹 API을 이용하여 이동 경로 등을 제공하는 서비스가 한 예이다. 현재 웹에서 매쉬업는 많은 기술에 개발되고 보편화되어 있는 분야로 웹에서는 주로 다음과 같은 특징으로 동작된다 [3].

- 웹 사이트에서 데이터 추출
- 또 다른 의미 있는 데이터로 변환
- 재결합(repackage) 된 데이터의 전송

사물의 이용한 물리적인 매쉬업은 가상 서비스 및 물리적 서비스의 조합으로 생성될 수 있으며, 사용자는 홈 가전 또는 센서와 같은 실 세계 장비를 이용하여 응용을 개발할 수 있다. 이

때 고려할 수 있는 매쉬업 방법은 다음과 같다 [3][4].

- 매뉴얼 매쉬업 개발: 특별한 매쉬업 툴이 존재하지 않고 기본적인 웹 기술 즉, HTML, HTTP, Atom, Javascript를 이용한 매쉬업
- 위젯 기반 매쉬업 개발: 데이터를 지속적으로 메모리의 변수에 기록하고, 개발자는 메모리 속의 데이터를 읽을 수 있는 위젯을 개발하여 이용하는 것
- 매쉬업 에디터를 이용한 사용자 개발: 최종 사용자가 직접 자신 만의 매쉬업 응용을 만들 수 있는 기술, 웹 2.0의 경우에 이는 일반적으로 매쉬업 에디터를 통해서 이루어진다.(예, Yahoo Pipes)

2.4 물리적 매쉬업 에디터(Physical Mashup Editor)

웹의 편리한 매쉬업 기능을 WoT에 적용하면 보다 쉽게 매쉬업 구현이 가능하다. 그 중에 Clickscript mashup editor을 확장한 방법이 제안되어 있다 [4]. Clickscript는 자원 및 연산자를 빙딜 블록화하여 이를 사람들이 눈으로 인지하면서 매쉬업 기능을 수행할 수 있다. 에디터는 단지 클라이언트 쪽 웹 기술을 이용해서 응요을 생성할 수 있으며, 보급 및 확장도 매우 간단하다. Clickscript는 Javascript로 구현되고 웹 브라우져에서 동작한다. 이 자체가 Bluetooth, Zigbee 같은 하위 수준 프로토콜에 기반한 자원을 직접적으로 사용하지는 못하지만, HTTP를 지원하므로 쉽게 RESTful 서비스를 액세스할 수 있다. 각각의 사물은 웹 기술을 접속이 가능하며, 편집기 상에서 하나의 기능 블록으로 표시되며, 연산자들과 더불어 새로운 응용 개방을 손쉽게 할 수 있다.

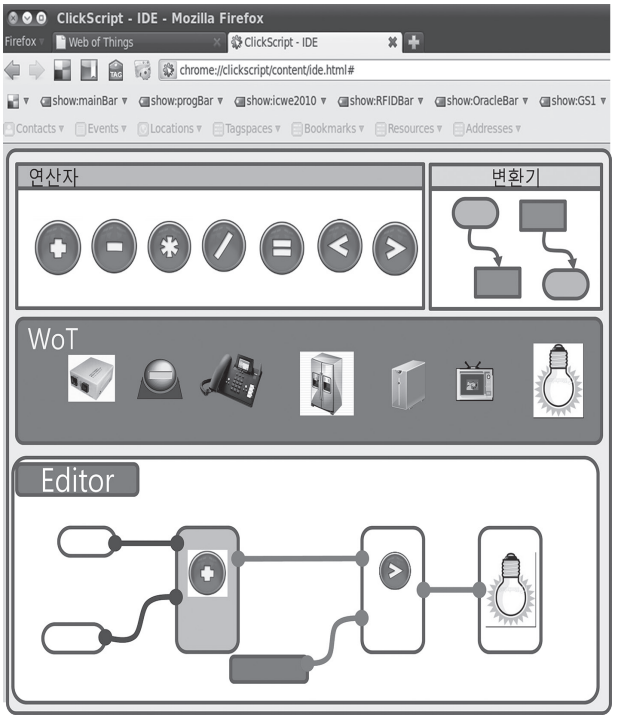

그림 6. Clickscript Mashup 편집기을 활용한 WoT물리적인 매쉬업 방법

Ⅲ. Web of things 표준화 이슈

WoT관련해서 표준화 작업도 국제표준화 기구를 중심을 활발히 논의 되고 있으며 본 장에서는 이에 대해서 간단히 논의 하도록 하겠다.

1. ITU-T SG13 Q.12

사물 웹(WoT) 관련하여 표준 개발은 ITU-T SG13산하의 Q.12(Question12)에서 "Y.2063"(Web of Things framework)로 개발 완료 하였다. 사물웹 프레임워크 표준에는 사물을 웹에 접속/제어할 수 있도록 해주는 WoT브로커를 중심으로, 사물웹 적용 모델, 기능적 구조 〈그림 7〉들을 명시하고 있다. 실세계 사물의 관점에서 보면 IP기반의 장비, RFID 디바이스(device), 무선 센서 디바이스 및 Zigbee 디바이스 등 무수히 많은 물리적인 사물이 다양한 네트워크를 기반으로 존재한다. 이들 중에는 인터넷에 연결할 수 없는 디바이스도 있고, 웹으로의 접근이 가능한 사물도 있을 수 있다. Y.2063에서는 이를 웹 접근 불가능 장비와 웹 접근/제어 가능 장비로 구분하여 나타내고 있다.

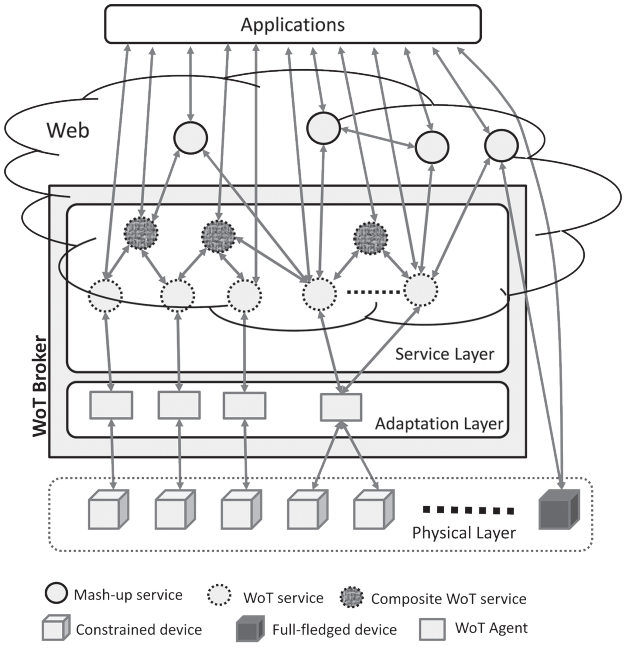

그림 7. WoT 프레임워크 기능적 구조

특히 웹 접근 불가능한 장비는 직접적으로 인터넷에 연결될 수 없으며, 웹에서의 접근이 불가능한 것으로, 표준에서 제시하는 WoT 브로커에 자원을 등록할 수 있으며, 읽기 전용의 디바이스의 경우 자원 정보를 수집하는 데만 활용될 수 있다 [7].

응용의 관점에서 보면, 각각의 사물은 단지 웹의 자원 또는 서비스로 보이며 접근 및 활용이 가능하게 된다. 이러한 모델 하에서 각각의 장비와 서비스들은 HTTP, URI, REST등의 웹 기술을 이용하여 쉽게 통신을 할 수 있으며, 기능을 활용 할 수 있다. 예를 들어 모든 사물들은 URI에 의해서 식별 가능하며 HTTP를 통하여 상호 통신할 수 있으며, 일부 센서와 같은 장비들은 REST나 RSS를 이용하여 정보를 교환할 수 있게 된다.

2. ITU-T SG16 Q.25

ITU-T SG16 Q.25에서는 서비스 관점에 표준화 작업을 진행하고 있다. 응용 개발자 및 사용 자 관점에서 사물웹 관련한 서비스 구조에 대해서 2012년 초에 개발을 시작하였다. 전체적으로 사물웹을 접근적 측면, 검색 측면, 공유 측면, 조합 측면으로 고려하고 있으며, 현재는 기본적인 구조만 완성되어 있는 단계이다 [9].

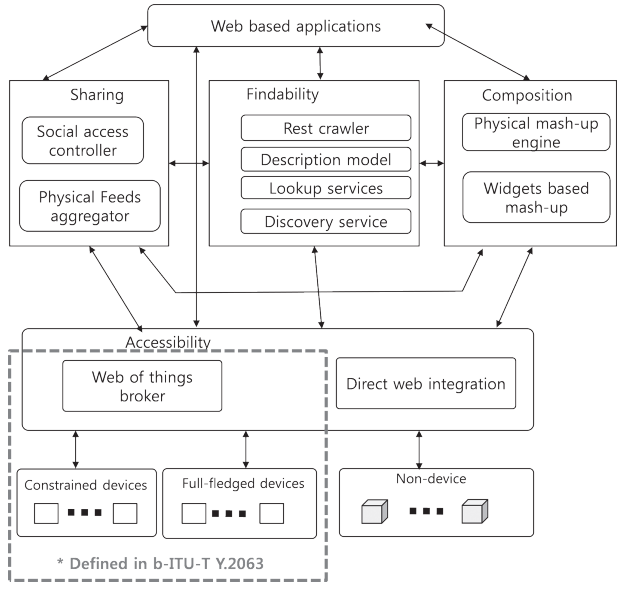

그림 8. WoT 서비스 구조

3. IETF Core WG(Constrained RESTful Environments WG)

IETF에서는 사물웹에서 활용 가능한 센서 및 작은 장비에서 웹 접근이 가능하도록 HTTP프로토콜보다 경량의 CoAP (Constrained Application Protocol) 프로토콜을 개발되고 있다.

이는 저성능의 프로세스를 가질 경우와 더불어 제한된 램(RAM)과 롬(ROM)을 가진 저사양의 Constrained 노드를 대상으로 하고 있다. 또한 REST 아키텍처를 기반으로 자원 검색, 멀티캐스트 지원, 비동기 트랜잭션 요청 및 응답 등을 지원하기 위한 프로토콜을 말한다. 기본적으로 IPv6를 지원하는 6LoWPAN을 하위 프로토콜로 생각하고 있으므로 주요 목표는 전달하는 메시지 자체를 가능한 작게 해서 메시지 Payload가 단편화되는 현상을 막으면서 필요로 하는 이벤트 요청 응답, Resource Discovery를 가능하게 하는 것이 표준의 목적이다. CoAP는 UDP기반에서 작동하는 프로토콜로 HTTP 보다 가벼운 경량 프로토콜의 개념이다. 현재 진행 중인 주요 CoAP 표준화 제안 현황은 다음과 같다. Sensinode에서 제안 중인 CoAP 프로토콜, InterDigital Communications에서 제안 중인 CoAP에서의 Sleep 노드처리와 멀티캐스트 고려 부분, 브레멘 대학에서 제안 중인 이벤트 리소스 동작절차와 각 헤더옵션에 대한 부분, 노키아에서 제안 중인 CoAP에서의 혼잡제어 부분으로 나누어진다 [10].

Ⅲ. 결 론

본고에서는 WoT에 대한 기본적인 개요와 기술 분석 및 표준화 이슈에 대해서 살펴 보았다. WoT는 각기 다른 통신 방식을 사용하는 이 기종의 사물(컴퓨터, 모바일장치, 센서, 데이터, 기능, 사람 등)를 웹 기술을 이용하여, 해당 사물을 자유롭게 원격 제어(접속, 감시, 제어 등)할 수 있는 미래형 시스템 통합 기술이라 할 수 있으며, 특히 범용적인 웹 기술을 이용하여 사물에게 표준화된 웹 인터페이스를 부여함으로써, 언제 어디서든 (웹 브라우저를 통해) 자유롭게 접속 및 제어가 가능하다. 또한 웹 기술을 특성상 방화벽의 문제가 자유롭고, 기존 다양한 인터넷 서비스와 연동이 매우 용이한 특성을 가지고 있다. 이는 기존의 사물(PC, 모바일, 센서 등) 제어 방식이 제조사/서비스 제공자에 종속되어, 이를 이용하기 위해서는 제조사/서비스 제공자의 장비에 특화된 플랫폼(프로토콜과 인터페이스)를 활용해야 만 가능 방식에서 벗어나, 웹 방식을 이용함으로써 사물에 대한 브라우징, 검색 및 북마킹 등이 기능이 지원되어 기존 인터넷(웹) 서비스와 동일한 방법으로 사물을 제어할 수 있는 특징을 가지고 있다. 향후 WoT는 유비쿼터스 세상을 실현하는데 중요한 방법이 될 수 있을 것을 기대된다.

Acknowledgement

"본 연구는 지식경제부의 지원을 받는 정보통신표준화 및 인증지원사업의 연구결과로 수행되었음"

참 고 문 헌

[1] Vlad Trifa and Dominique Guinard, “Towards the Web of Things- Whitepaper 1.0", http://www.webofthings.com

[2] Dominique Guinard, “A Resource Oriented Architecture for the Web of Things”

[3] Dominique Guinard, A Web of Things Application Architecture – Integrating the Real-World into the Web, ETH Zurich, 2011

[4] Jin Yu, Boualem Benatallah, Fabio Casati, and Florian Daniel. Understanding Mashup Development. IEEE Internet Computing, 12(5):44{52, 2008.

[5] Benedikt Ostermaier, Matthias Kovatsch, Silvia Santini, Connecting Things to the Web using Programmable Low-power WiFi Modules, Pervasive2011.Workshop09, 2011

[6] Dominique Guinard, Vlad Trifa, Stamatis Karnouskos, Patrik Spiess, and Domnic Savio. Interacting with the SOA-Based Internet of Things: Discovery, Query, Selection, and On-Demand Provisioning of Web Services. IEEE Transactions on Services Computing, 3(3):223{235, February 2010.

[7] ITU-T Recommendation Y.2063, Framework of Web of Things, ITU-T 2012.7.

[8] Z. Shelby, K. Hartke, C. Bormann, “Constrained Application Protocol (CoAP)”, http://tools.ietf.org/id/draft-ietf-core-coap-08.txt, 2011

[9] Web of things service architecture, Draft Recommendation H.WoT-SA, ITU-T 2012.9.

[10] 송성학, “[M2M] CoAP(Constrained Application Protocol) 표준화 동향”, TTA ICT Standard Weekly 2010.

[11] 인민교, “Web of Things(WoT) 표준화 동향 “ TTA Journal Vol.138, 2011

[12] Benedikt Ostermaier_, Kay R¨omery_, Friedemann Mattern_, Michael Fahrmairz and Wolfgang Kellererz, A Real-Time Search Engine for the Web of Things, 2010

약 력

인 민 교

1998년 충남대학교 정보통신공학과(학사)
2000년 충남대학교 정보통신공학과(석사)
2000년~현재 한국전자통신연구원 표준연구센터 선임연구원
2010년~현재 TTA 웹프로젝트그룹 간사
2006년~현재 ITU-T SG13 Editor
2012년~현재 ITU-T SG16 Editor
2008년~현재 TTA ICT 국제표준화전문가
관심분야: 차세대 웹, Web of things, IPv6, 클라우드 컴퓨팅

이 승 윤

1999년~현재 ETRI 표준연구센터, 책임연구원
2003년~현재 ETRI 표준연구센터 서비스융합표준연구 팀장
2004년~현재 TTA ICT 국제표준화전문가
2006년~현재 TTA IT응용기술위원회(TC06) 부의장
2006년~현재 TTA 웹프로젝트 그룹(PG605) 의장
2006년~현재 ITU-T SG13 Editor
2010년~현재 ISO/IEC JTC 1 SC38 WG3 의장 (Convener)
2006년~현재 APT ASTAP IRT EG Rapporteur
관심분야: 차세대 웹 표준, 유비쿼터스 웹서비스 (UWS) 표준, 모바일웹 표준, 웹 2.0 표준