

# 사물웹 (Web of Things) 표준화 동향

전종홍 한국전자통신연구원  
인민교 한국전자통신연구원  
이승윤 한국전자통신연구원



## 1. 머리말

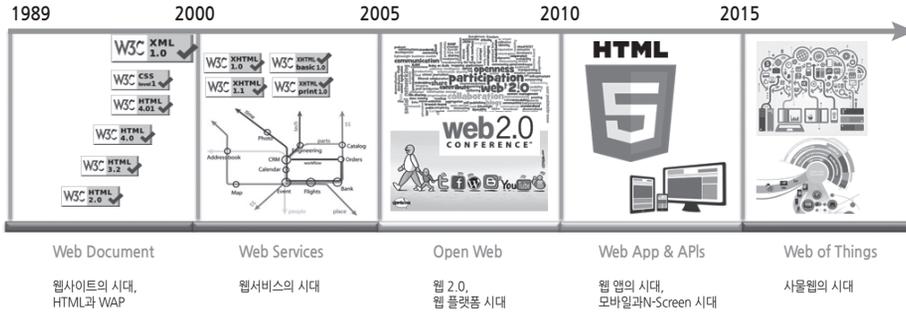
1989년 CERN의 팀 버너스 리에 의해 만들어진 월드 와이드 웹 기술은 HTML(HyperText Markup Language), URL(Unified Resource Locator), HTTP(HyperText Transfer Protocol)이라는 세 가지 기술을 기초로 간단한 문서와 자원들을 공유하기 위한 목적으로 출발하였다. 문서 공유를 목적으로 개발되었던 웹 기술은 뛰어난 확장성과 기술 활용성 덕분에 다양한 기술들과 융합되면서 인터넷 상의 모든 것들을 연결하는 대표적인 응용 기술로 발전해왔다.

웹 기술은 단순히 정적인 문서 형태로 정보를 공유하는 형식(Web of Document)에서, 서로 다른 서비스를 연결하는 기술(Web of Services)과 서로 다른 원격 데이터베이스를 연결하는 기술(Web of Data)로 진화하였고, 이제는 단말의 HW를 제어하고 연결하는 단계(Web of Devices)를 넘어, 클라

우드를 통해 정보를 공유 제공할 수 있도록 하며 모든 사물을 연결하는 단계(Web of Things)로까지 진화하고 있다.

특히 최근 단말의 소형화와 유무선 네트워킹 기술의 발전에 따른 사물인터넷 시대를 맞이하면서, 사람과 사물을 연결하고 사물과 사물 간을 연결할 수 있는 호환성 높고 개발이 용이한 대표적인 기술로 웹 기술이 재조명되면서 사물웹 기술과 표준에 대한 관심이 높아지고 있다.

사물웹의 관점에서 볼 때 웹 기술의 장점은 하부 네트워크에 상관없이 확장성이 뛰어난 HTTP 프로토콜을 활용해 통신할 수 있으며, 웹 표준을 준수하면 어떤 서비스와도 연결할 수 있고, 인터넷 상의 수많은 웹 서비스와 자원을 결합시킨 응용들을 용이하게 개발할 수 있으며, 손쉬운 개발언어와 개발환경을 제공한다는 큰 장점을 갖고 있다.



[그림 1] 웹 기술의 진화 발전 단계

본 고에서는 이러한 사물웹 기술과 관련 웹 응용 기술 관련 표준화 동향에 대해 살피고, 향후 사물웹 기술이 해결해야 할 이슈들과 나아갈 방향들에 대해 고찰해보고자 한다.

## 2. 인터넷과 웹의 진화

인터넷과 웹은 상호보완적인 관계로 진화 발전을 해왔다고 할 수 있다. 초창기 인터넷 기술은 물리적인 연결과 통신 프로토콜에 초점을 맞추고 진화를 해왔다면, 1989년 등장한 웹 기술은 인터넷 상의 다양한 자원들을 다시 웹이란 가상 정보 공간 속에서 연결될 수 있도록 하며 사람이 좀 더 편리하게 사용할 수 있도록 하는 기술로 진화해왔다. 이처럼 인터넷의 등장과 성장은 웹 기술의 등장과 성장을 이끌었고, 다시 웹 기술의 성장은 새로운 인터넷 기술의 진화 발전을 이끌었다.

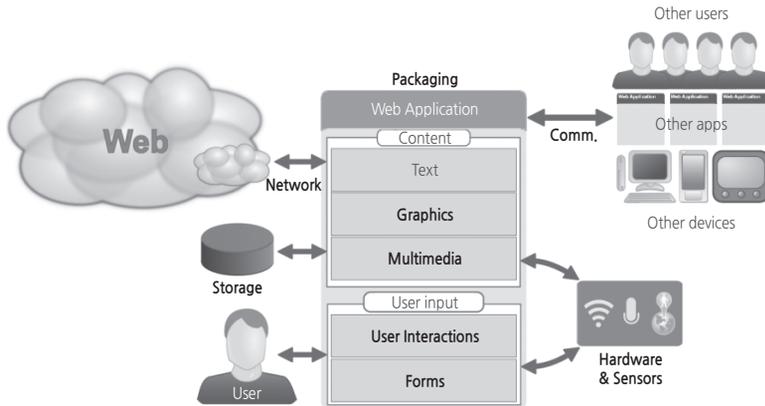
웹 기술의 진화 발전 과정을 살펴보면 다음과 같이 크게 4단계로 구분해볼 수 있다. 첫 번째 단계는 1989년부터 1999년까지의 기간으로 HTML, URL, HTTP라는 세 가지 기술에 기초 웹 기술이 제안되고 보다 나은 인간 중심의 정보처리 및 지식공유 등을 목표로 하는 단계였다. 두 번째 단계는 2000년부터 2004년까지의 기간으로 XML에 기반하며

인간 중심의 정보 처리뿐 아니라 다양한 디바이스와 서비스, 멀티미디어를 연결하는 것을 목표로 하는 단계였다. 세 번째 단계는 2005년부터 2009년까지로 구글, 아마존, 위키피디아 등의 성공과 함께 웹 산업을 제2의 전성기로 이끌며 다양한 신규 서비스가 등장할 수 있는 기반을 만들었다. 마지막 네 번째 단계는 2010년 이후부터 현재까지로 스마트폰 및 태블릿 등 다양한 모바일 기기를 포함해 다양한 단말과 스크린 환경에서 호환되어 동작하는 통합 응용 플랫폼으로써 웹이 자리 잡아가는 단계라 할 수 있다[1].

이처럼 웹 기술이 확산된 가장 큰 배경은 URI와 HTTP라는 확장성 있는 자원접근 방식과 프로토콜을 사용하였고, HTML이라는 장치 독립적 표현 언어를 이용하여 정보를 표현하였고, 이를 다양한 단말에서 사용자 클라이언트(브라우저)를 통해 통신하여 활용할 수 있게 함으로써, 언제 어디서든 표준 클라이언트만 있으면 웹을 사용할 수 있다는 단말 플랫폼 비종속적인 장점을 가질 수 있었기 때문이다.

### 2.1 웹 애플리케이션 기술의 진화

2005년 이후로 웹 기술은 또 한번의 큰 변화를 겪게 되는데, 그것은 ‘문서 위주의 웹 환경’에서 ‘응용 환경으로 웹’으로의 진화하는 것이다. 이 과정에서 AJAX와 같은 비동기식 처리 기술, 브라우저 및



[그림 2] 웹 애플리케이션 기술 표준 분류 및 현황[3]

JavaScript 가속화 기술, Open API와 RESTful 구조와 새로운 기술과 표준들이 등장하면서 웹 애플리케이션 기술 분야에서 많은 발전이 진행되었다[1].

일반적으로 ‘웹 애플리케이션’이라는 용어는 HTTP를 통해 전달되는 웹 페이지(XHTML 또는 그 변이형과 CSS, ECMAScript로 구성되는)의 집합체들이 웹 브라우저 내에서 애플리케이션 같은 환경을 제공하는 것을 말한다. 즉, 웹 애플리케이션은 여러 페이지를 거치는 대화형 처리 절차를 가지며, 이를 위한 상태 유지와 데이터 유지를 필요로 한다는 점에서 단순한 웹 콘텐츠와는 구분된다[2].

일반적으로 웹 애플리케이션의 경우 별도 프로그램의 설치없이도 표준 브라우저만 있으면 계속 업그레이드된 기능을 사용할 수 있고, Open API 등을 통해 손쉽게 매시업 할 수 있도록 기능을 제공하는 등 재활용할 수 있다는 장점을 갖는 반면, 오프라인 처리와 단말의 특성 정보를 활용할 수 없고, 브라우저의 성능에 좌우되며 대용량 처리 등에 한계를 갖는다는 단점을 갖고 있다. W3C에서는 이러한 웹 응용의 단점들을 극복할 수 있도록 다양한 API들을 만들고 있으며, 웹 응용 개발을 용이하게 만드는

XHR, Web IDL, Web Socket API, CORS 등과 같은 웹 응용 관련 신규 표준들을 개발하고 있다.

- XHR은 AJAX와 같은 비동기식 웹 애플리케이션 개발 기법의 핵심 요소, 서버와 클라이언트 사이의 데이터 전송을 위한 기능을 정의
- Web IDL은 브라우저에서 구현되어 웹상에서 인터페이스를 설명하기 위한 용도로 사용될 수 있는 IDL(인터페이스 정의 언어)을 정의
- 웹 소켓 API 규격에서는 원격 서버와의 양방향 통신을 가능하도록 하는 웹 소켓 API를 정의

## 2.2 웹 플랫폼으로의 진화

일반적으로 플랫폼은 협의(狹義)의 의미로는 API를 통해 애플리케이션에 필요한 서비스를 제공하는 특수한 SW 프로그램을 의미하며, 광의(廣義)의 의미로는 제품이나 서비스 개발의 기반이 되는 아키텍처로 다양한 응용과 서비스가 개발/제공되는 환경을 포괄적으로 의미한다.

초기의 웹 기술은 단순히 문서를 공유하고 정보를 표현/공유하기 위한 구조였지만, 지속적인 기술의

발전과 확장에 따라 2000년 이후부터 웹 기술은 점점 더 다양한 서비스와 데이터를 연동하고 서비스할 수 있도록 하는 응용 플랫폼의 형태로 발전을 해왔다. 이러한 웹 플랫폼 기술의 지속적인 확산 배경에는 다음과 같은 특징들이 있다.

- 1) 단순성: 웹 기술의 HTTP, URI, HTML이라는 단순한 요소들만을 활용함에 따라 기술 확장이 용이하고, 다른 응용 개발 환경보다는 손쉽게 개발자가 관련 기술을 익히고 활용하고 적용해 볼 수 있다는 장점을 갖고 있다.
- 2) 호환성: 웹 기술은 전 세계적으로 단일 표준화 되어 있고 표준과 기술이 공개되어 있어 관련 기술을 손쉽게 적용하여 활용할 수 있다는 장점을 갖고 있다.
- 3) 독립성: HW 및 운영체제 의존성이 적어서 웹 표준을 구현한 클라이언트(표준 브라우저 또는 웹 엔진)가 있는 환경이라면 어떤 장치에서든 일관되게 활용될 수 있다는 장점을 갖고 있다.
- 4) 개방성: 웹 응용은 서비스 재활용이 손쉬우며, 개방형 API 등을 통해 기존에 개발된 내용을 손쉽게 융합하여 활용할 수 있다는 장점을 갖고 있다.
- 5) 이식성: 웹 기술은 소형의 단말 환경에서도 손쉽게 적용할 수 있다는 특징을 갖고 있으며, 이를 통해 다양한 서비스와 데이터들을 손쉽게 연결할 수 있는 장점을 갖고 있다.

웹 플랫폼 기술은 웹 서버 플랫폼 기술과 웹 클라이언트 플랫폼 기술, 그리고 웹 서비스 플랫폼으로 구성되며, 최근 다양한 동향에서 볼 수 있듯이 웹 기술은 응용 프로그램 개발자 확보 용이성, 크로스 디바이스 적용 용이성, 응용 확보의 용이성, 웹의

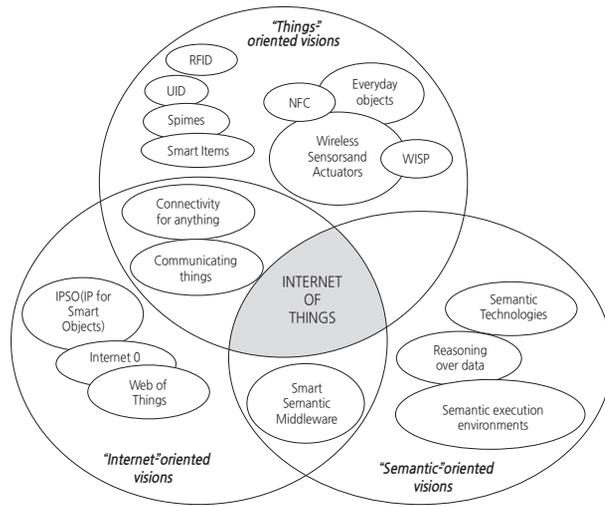
개방성, 글로벌 협력의 용이성, 웹 접속 디바이스의 급속한 증대 등의 장점을 갖고 있어, 점점 더 단말/플랫폼/스크린의 경계가 없도록 하며, 모든 정보와 서비스 및 사물을 묶는 인프라와 플랫폼이자 새로운 앱과 SW 환경으로서의 형태로 진화하고 있다[1].

### 3. WoT 표준화 동향 - W3C

인터넷이 네트워크를 통해 다양한 컴퓨터들을 연결하기 위해 노력하였던 것처럼, 사물인터넷도 다양한 사물들을 연결하기 위한 다양한 시도들을 진행하여 왔다. 이러한 사물인터넷의 연구들은 [그림 3]과 같이 어떤 부분에 좀 더 초점을 두느냐에 따라 크게 사물 중심, 인터넷 중심, 시맨틱 중심과 같은 세 가지 관점에서의 접근들로 구분할 수 있다. 이처럼 복잡하고 다양한 기술들이 등장하고 활용되다 보니, 반대로 정작 일관된 표준으로 사물들을 손쉽게 연결하고 활용하기는 점점 어려워진다는 문제가 등장하게 되었다[11]. 이에 주목받게 된 기술이 웹 기술이며, 웹 기반으로 사물들을 연결하고 제어/관리하며 다양한 응용들을 제공할 수 있도록 하기 위한 기술들을 통칭하여 ‘사물웹(Web of Things)’ 기술로 부르고 있다.

이러한 WoT 기술은 접근 관점에 따라 다음과 같이 크게 네 가지로 구분할 수 있다.

- 1) 단순화된 연결 기술로 RESTful 구조와 HTTP를 활용하는 관점
- 2) API와 HTML 등을 이용하면서 기존의 다양한 웹 자원과 서비스들을 결합시키는 응용과 사용자 인터페이스 관점
- 3) 의미적인 연결과 데이터 통합을 위한 시맨틱 웹을 활용하는 관점



[그림 3] 사물인터넷 패러다임에 대한 3가지 관점[4]

4) 마지막으로 사물을 웹 공간으로 연결하고/관리하는 핵심 기술로 웹 기술을 활용하는 관점

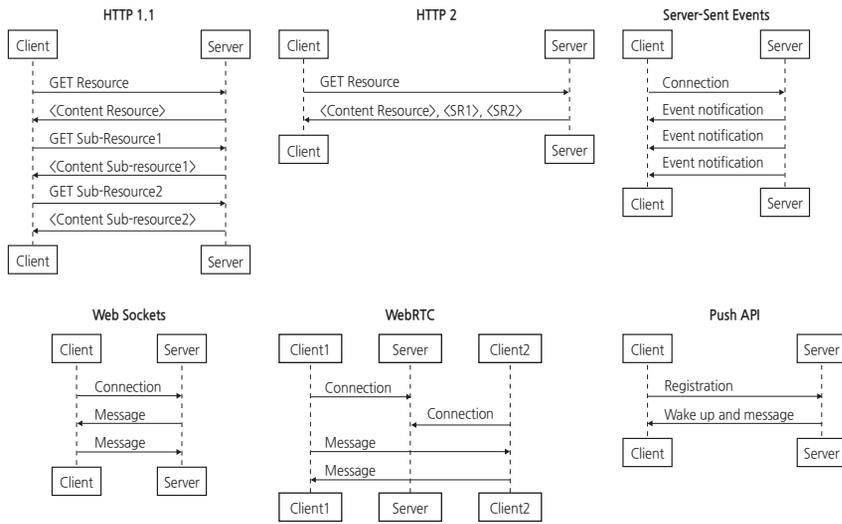
이러한 WoT 기술에 대한 표준화는 웹 핵심 기술을 다루는 W3C, 통신체계 전반의 표준화를 다루는 ITU-T, 그리고 인터넷 프로토콜 표준화를 다루는 IETF 등에서 진행되고 있다.

### 3.1 W3C의 표준화 동향 - REST 구조와 웹 프로토콜의 진화

W3C에서의 WoT 환경과 관련하여 다양한 시도가 진행되고 있는 분야는 프로토콜 분야이다. 그중에서 가장 많이 활용되는 것으로 HTTP를 활용한 RESTful 구조와 설계 방법을 꼽을 수 있다. 1993년 HTTP 규약을 설계한 로이 필딩(Roy Fielding)은 웹의 제약점들을 해소하고 확장하기 위해 웹의 제약점들을 여섯 가지 범주로 묶어 웹의 구조적 스타일(Web's architectural style)이라고 정의하였다 [10].

- ① 클라이언트/서버(Client/Server): 제공자와 이  
용자로 나누어 각각의 기술 사용 가능
- ② 일관된 인터페이스(Uniform Interface): 모든  
인터페이스는 표준에 기반 일관성 유지
- ③ 계층 시스템(Layered System): 시스템을 더  
계층화시키고 나눌 수 있음
- ④ 캐시(Cache): 클라이언트와 서버의 통신횟수와  
양을 감소시킨다
- ⑤ 상태 없음(stateless): 서버 측에서의 애플리케이션의 상태를 가지지 않는다
- ⑥ 주문형 코드(Code on demand): 클라이언트  
에 다운로드하여 해석하고 실행한다.

이러한 제약점과 특징들은 웹의 기본적인 구조로서, 이것이 REST(REpresentational State Transfer)라는 구조 설계 방식의 시작인 동시에 오늘날 다양한 응용 서비스 연동을 위한 보편적인 설계 방식으로 활용되고 있다.



[그림 4] web connectivity 기술과 프로토콜의 진화[5]

HTTP는 1991년에 0.9 버전이 처음 발표되었고, 1996년에 1.0 버전, 1999년에 1.1 버전이 나온 뒤로 10년이 넘는 시간 동안 큰 변화가 없었다. 그러나 오늘날의 웹 서비스 환경은 90년대에 비해 HTTP 요청 개수가 20배 가량 증가했다. 이에 따라 오버헤드를 줄이고 보다 빠르게 데이터를 교환할 수 있도록 하는 SSE(Server-Sent Events), Web Sockets, Web RTC, Web Push API, XHR(XMLHttpRequest)과 같은 다양한 웹 확장 프로토콜들에 대한 기술 개발과 표준화를 진행하고 있다.

또한, 데이터의 기본적인 교환 형식도 XML 기반이 아닌 JSON과 같이 자바스크립트 기반의 교환 형식과 같이 빠르고 간단한 형태가 확산하고 있다. 향후 보다 저사양의 다양한 단말들과 사물들이 등장하고 이를 WoT 응용과 서비스로 연결하기 위해 좀 더 빠르고 효과적이며 lightweight한 웹 프로토콜들에 대한 보다 많은 연구개발과 표준화 노력이 진행될 것으로 보인다.

### 3.2 W3C의 표준화 동향 - 유비쿼터스 웹부터 사물웹까지

모바일 등과 같은 소형 단말이 증가하고 다양한 스크린 환경이 등장함에 따라 W3C도 이러한 단말 환경의 변화를 수용하기 위한 새로운 표준화 노력을 기울여 왔다. W3C는 소형 단말들을 중심으로 이루어지는 디바이스 웹 표준화를 위해 2006년부터 유비쿼터스 웹 도메인을 신설하고, 산하 활동으로 기기독립 활동(Device Independence Activity), 모바일 웹 이니셔티브 활동(Mobile Web Initiative Activity), 멀티모달 활동(Multimodal Activity), 그리고 보이스 브라우저 활동(Voice Browser Activity) 등을 진행하여 왔다.

최근에는 단말 환경이 좀 더 소형화됨에 따라 2010년부터는 WoT(Web of Things)에 대한 논의를 시작하였는데, 2012년 초까지는 별다른 활동으로 이어지지 않았다. 최근 소형 단말들의 확산과 IoT 확산에 따라 2013년 초부터 논의를 시작하여 WoT 표준을 논의하기 위한 WoT CG(Web of Things Community Group)를 발족하게 되었다.

WoT 커뮤니티 그룹의 기본적인 취지는 WoT와 관련된 다양한 표준화 이슈들을 수집하고 이를 정리하는 데 있다. 이를 위해 기존 표준들의 검토, 유즈케이스 수집 및 분석, 응용에 필요한 새로운 표준 이슈 검토, Gap 분석, 다양한 멤버들과의 논의 등과 같은 활동들을 추진하는데 목적을 두고 있다.

WoT CG에서는 2014년 6월에 표준화 이슈 발굴을 위해 독일 베를린에서 WoT 표준화 워크숍을 개최하였다. W3C 워크숍의 기본 목표는 업계로부터 표준화 이슈와 요구사항, 핵심 표준화 항목 등을 도출하고 이를 기반으로 표준화 활동을 추진하기 위해 있는 것처럼, 이번 워크숍에서도 WoT 표준화 이슈 발굴을 위해 주요 사례발표, 기술 이슈 발표, 패널토의, 소분과 토의 등을 진행하였다[8].

1일차(6/25)	2일차(6/26)
<ul style="list-style-type: none"> <li>• 사례, 유즈케이스 및 요구사항 주제발표</li> <li>• 사물에서 WoT로의 주제발표 및 토론</li> <li>• Security, Trust, Privacy 주제 발표</li> <li>• Breakout session 구성 및 토론(5개)</li> <li>• core technology - 패널토의</li> <li>• 데모 발표</li> </ul>	<ul style="list-style-type: none"> <li>• Semantic, Linked Data, vocabulary 주제발표</li> <li>• IoT 표준화로부터의 경험 - 패널토의</li> <li>• Breakout session 구성 및 토론(5개)</li> <li>• 이후 나아갈 방향 논의 - 패널토의</li> <li>• 데모 발표</li> </ul>

이번 워크숍은 최초로 WoT에 대한 전반적인 표준화 이슈들을 함께 논의하는 워크숍이었다는데 큰 의의가 있었으며, 다음과 같이 그 결과를 요약할 수 있다.

- Semantic, Security, Linked data, Home network, Energy, IoT application 등의 다양한 영역의 산학연 멤버들이 참여
- 총 105편의 position paper들이 제출되었고 EU 프로젝트 관련 논문 15편, 시맨틱웹 관련 8편,

IoT 표준화 관련 6편, IoT 응용 표준화 관련 6편, 임베디드 시스템 관련 5편, 유즈케이스 관련 7편, 가상화 플랫폼 6편 등이 제출됨

- 제출된 position paper 중 use case와 보안, 응용 사례, Semantic 및 Linked data 관련 19편을 추려 발표 세션을 구성
- WoT를 위한 스크립팅, 응용 런타임, 서비스 표현, 인터랙션 모델, 비즈니스 모델, 응용 레벨 프로토콜과 데이터 인코딩, 프라이버시, 로봇, Web of Data 등의 주제로 11개 소그룹 논의를 진행함
- WoT의 핵심 기술이 무엇인지와 IoT 표준화 경험 등에 대한 이야기를 듣는 2개의 패널토의
- Siemense, Everything, Plantronics, Yaler.net, KIT, CNI 등의 10건의 데모/시연

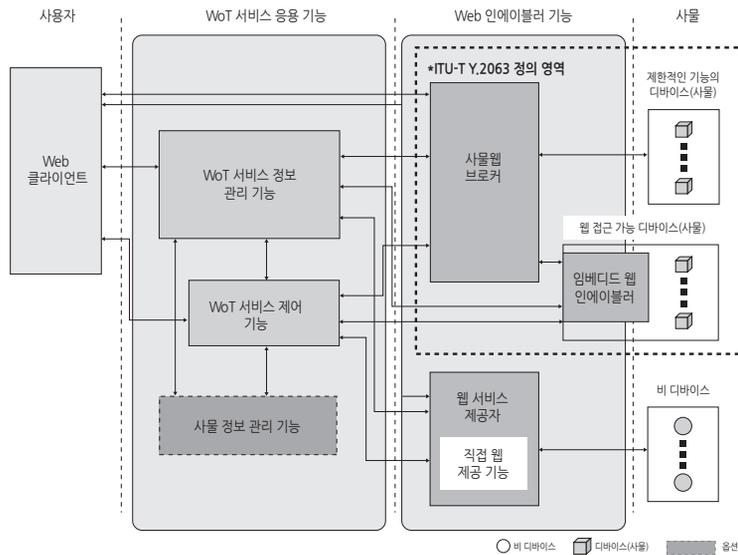
이번 워크숍 논의 결과를 기초로 금년 11월에는 실질적인 표준화 추진을 위한 WG를 구성할 것으로 예상된다. WoT WG 생성 후에는 주요하게 4가지 영역(application runtime, service description, security/privacy, web of data)에 대한 TF를 나누어 구성한 다음, 4가지 영역별로 표준화 이슈와 요구사항들을 정리하면서 구체적인 표준문서 작업 계획을 수립하는 형태가 될 것으로 예상된다.

#### 4. WoT 표준화 동향 - ITU-T

ITU-T에서의 WoT에 대한 표준화는 미래 네트워크 환경에서의 응용 관점에서 이슈 발굴과 접근을 시작하였다.

##### 4.1 ITU-T SG13 Q.12

ITU-T에서는 2010년부터 네트워크 분야를 다루는SG13에서 사물웹(WoT)에 관련된 표준



[그림 5] 사물웹 서비스 구조

권고안 작업을 시작했으며, 2012년 7월에 SG13 산하의 Q.12(Question 12)에서 표준화 권고안 ‘Y.2063(Web of Things framework)’을 제정하였다. 주요 내용은 사물웹의 개념 정립, 기본적 요구사항 및 기능적 요구 사항을 정의했으며, 이를 기반으로 사물웹의 프레임워크를 완성하였다. 세부적으로는 사물의 종류를 웹 기능이 내장된 장치와 웹 기능을 갖추지 못한 제약을 지닌 2가지로 분류했으며, 각각에 대해 사물웹에서 활용할 방법을 명시하였다[6].

Y.2063에서 정의한 전체적인 사물웹 프레임워크에서 응용은 웹 기술을 이용해 사물의 기능을 접근·활용할 수 있으며, WoT 브로커는 장치(사물)를 웹 응용에서 활용할 수 있도록 돕는다. WoT 브로커는 서비스 계층과 적용 계층으로 분리된다. 적용 계층은 물리 계층에 해당하는 각각의 장치를 웹 인터페이스를 활용해 접근/제어/활용이 가능하도록 하며, 서비스 계층은 적용 계층을 통해 입력된 각각의 장치를 웹 자원화하는 기능을 담당한다. 웹 자원화 방법은 각 순수 장치 기능을 웹 응용에서 바로 접속

하도록 할 수도 있지만, 내부적인 조합(사업자별로 다를 수 있음)을 통해 새로운 서비스를 만들 수 있게 된다.

응용의 관점에서 보면, 각각의 사물은 단지 웹의 자원 또는 서비스로 보이며 접근 및 활용이 가능하게 된다. 이러한 모델 하에 각 장비와 서비스들은 HTTP, URI, REST 등의 웹 기술을 이용해 쉽게 통신하고 기능을 활용할 수 있다. 예를 들어 모든 사물은 URI에 의해 식별 가능하며 HTTP를 통해 상호 통신할 수 있으며, 일부 센서와 같은 장비들은 REST나 RSS를 이용해 정보를 교환할 수 있게 된다[7].

#### 4.2 ITU-T SG16 Q.25

Y.2063에서는 사물이 웹에서 접근 가능하도록 하는 프레임워크를 정의하였으며, 다양한 사용예를 들고 있다. 이를 기반으로 ITU-T SG16 Q.25에서는 2012년 서비스 관점에서의 사물웹 서비스 구조에 대한 개발을 시작(H.WoT-SA: Web of things service architecture)하였으며, 현재 기본적인 서

비스 구조를 개발한 상태이다. [그림 5]에서는 현재까지 개발 완료된 사물웹의 서비스 구조를 보인다.

H.WoT-SA 표준 권고안에서 정의하는 사물웹의 서비스 구조는 크게 사용자 기능, WoT 서비스 응용 기능, Web 인에이블러 기능 및 사물 자체로 분류된다. 사용자는 웹 클라이언트나 관련 앱을 통해서 사물을 활용한 서비스를 제공받거나, 혹은 직접적으로 사물에 접근하여 읽거나, 동작을 제어할 수도 있다. 또한, 사물은 웹서버가 내장되어 있는 사물과 웹 기능이 없는 사물로 분류되며, 웹 기능이 없는 사물은 웹 인에이블러 기능을 통해서 웹 접근이 가능하게 된다. [그림 5]에서 정의된 사물웹 브로커는 ITU-T Y.2063에서 정의되어 있는 기능이다. 웹 인에이블러 기능에서는 사물웹 브로커 기능 이외에도 웹 서비스 제공자를 통해서 장치 이외의 것 즉, 동영상, 사진, 텍스트 등을 비 장치(디바이스)에 대한 웹 접근을 제공하고 있다. 사물웹의 기본적인 개념은 모든 웹 접근이 가능한 것이므로, 장치에 한정되지 않으며 웹 콘텐츠 등도 포함된다. WoT 서비스 응용 기능은 서비스 구조에서 핵심적인 역할을 하는 부분으로 사물을 이용한 서비스의 제공, 서비스 검색, 서비스 조합 등 다양한 기능을 제공하며 사용자에게 직접 제공한다. 사용자는 자신이 원하는 기능을 WoT 서비스 응용 기능을 제공하는 서비스 사업자를 통하여 검색/제공받을 수 있게 된다. H.WoT-SA 권고안은 현재 활발히 개발 중에 있으며, 사물에 대한 기술(description), 접근 방법, 흐름도 기능 등이 추가로 개발될 예정으로 2015년 상반기 권고안 승인을 목표로 진행되고 있다[9].

## 5. WoT 표준화 동향 - IETF

IETF에서는 2010년 6월 기존 HTTP에 비해 보다

제약이 많고, 낮은 사양의 장치에서 활용 가능한 전송 프로토콜인 CoAP(Constrained Application Protocol)를 만드는 작업을 시작했다. 이는 저사양의 하드웨어 장치, 즉 제한된 램과 롬 등을 가진 저성능의 장치에서 활용 가능한 프로토콜로써 Core(Constrained RESTful Environment) WG에서 개발을 진행하였다.

2013년 RFC 표준으로 채택된 CoAP 프로토콜은 제한된 기능을 가진 장치를 대상으로 하는 것으로, 이더넷(Ethernet), 와이파이라와 같은 강력한 네트워크 기능을 탑재하고 활용하는 데스크톱 컴퓨터, 노트북 및 스마트폰과 달리 신뢰성이 낮거나, 전송 범위가 좁은 네트워킹 기능을 탑재한 장치를 대상으로 한다. 따라서 불안정한 네트워크 환경에 있는 경우가 많아 TCP나 HTTP 같은 무거운 프로토콜은 사용이 어렵다. 이에 CoAP는 UDP를 기반으로 개발되었으며, 대상으로 REST 아키텍처 기반의 자원 검색, 멀티캐스트 지원, 비동기 트랜잭션 요청 및 응답 등을 지원하기 위한 프로토콜이다. IPv6를 지원하는 6LoWPAN을 하위 프로토콜로 간주하며, 주요 목표는 전달하는 메시지 자체를 가능한 한 작게 해 메시지 Payload가 단편화되는 현상을 막고, 필요로 하는 이벤트 요청 응답, 자원검색을 가능하게 하는 것이다[6].

## 6. 맺음말

지금까지 사물웹 기술과 관련하여 주요 표준화기구에서 진행하고 있는 표준화 동향과 함께 빠르게 변화하고 있는 여러 사물웹 관련 동향에 대해 살펴보았다. 웹 기술은 편의성, 응용성, 호환성, 보편성, 확장성을 확보하기 위해 끊임없이 진화를 해오고 있고, 특히 최근에는 단말과 네트워킹 기술이 발전하고

사물인터넷 시대가 다가옴에 따라 새로운 기술개발 및 표준화 노력이 진행되고 있다는 사실을 확인할 수 있었다. 웹 기술은 범용적으로 잘 알려진 기술로서, 다양한 통신 방식을 사용하는 장치들을 단일한 방식, 즉 웹 인터페이스를 이용해 제어가 가능한 하나의 방법으로 인식되고 있으며, 이에 따라 주요 표준화 기구(IETF, ITU-T, W3C)에서도 관련 표준화 작업을 개발해왔으며 현재도 지속적인 개발이 이뤄지고 있다는 점도 확인할 수 있었다.

지금까지 살펴본 것과 같이 웹 기술이 환경에 변화에 따라 5년 주기로 진화하는 경향을 확인할 수 있었다. 지난 2010년부터 모바일 환경과 다중 단말 환경의 성장에 따라 웹 앱 기술에 대한 다양한 발전이 있었고, 다가오는 2015년에는 보다 소형화된 다양한 단말과 사물들을 연결하는 사물웹 시대가 열릴 것으로 전망되고 있다.

이러한 사물웹 시대를 위해서는 또다른 기술적인 진화가 필요할 것으로 보인다. 지금까지의 웹 클라이언트가 데스크톱이나 모바일 단말과 같은 고사양의 단말을 대상으로 했다면, 사물웹 시대에는 낮은 사양의 단말 클라이언트 환경에 적합한 경량화된 웹 프로토콜과 제어들이 필요할 것이며, 사물들을 연결하는 웹 클라우드와 웹 관리 기술들도 필요할 것으로 보인다.

지금까지의 웹 기술이 사람에게 정보를 제공하기 위해 표현(representation) 기술에 초점을 맞추었다면, 사물웹 시대에는 제어(control)에 초점을 맞춘 기술들이 필요할 것으로 보인다. 오픈소스 하드웨어와 같은 저사양 단말들과 다양한 액세서리들을 웹 환경에 통합하는 노력이 필요할 것으로 보이며, 그러기 위해서는 마크업보다는 스크립트와 API에 초점을 맞추어야 할 것으로 보인다.

사물웹의 확산을 위해서는 결국 사용자 가치를

제공할 수 있는 응용과 서비스가 확산되어야 하며, 것처럼 사물웹 응용과 서비스의 확대시키기 위해서는 개발자들이 손쉽게 익힐 수 있고, 호환될 수 있으며, 개발할 수 있는 쉬운 기술들로 개발되어야 할 것이다.

그동안 웹이 많은 것들을 연결할 수 있었던 배경이 단순함과 호환성, 그리고 확장성에 있었던 것처럼, 세상 모든 것을 연결하겠다는 크고 거창한 모델보다는 손쉽게 연결할 수 있는 쉬운 기술 개발과 표준화가 더욱 중요할 것이다. 

---

본 연구는 미래창조과학부의 지원을 받는 정보통신표준화 및 인종 지원사업의 연구결과로 수행되었음

#### [참고문헌]

- [1] 전종홍, 이승윤, 'HTML5 기반의 웹 플랫폼 기술 표준화 동향', 전자통신동향분석 제27권 제4호, 2012년 8월
- [2] 전종홍, 이승윤, '차세대 모바일 웹 어플리케이션 표준화 동향', 전자통신동향분석, 제25권 제1호, 2010년 2월
- [3] EU FP7 MobiWebApp Project report, 'Standards for Web Applications on Mobile: May 2011 current state and roadmap', <http://www.w3.org/2011/11/mobile-web-app-state>
- [4] Luigi Atzori, Antonio Iera, Giacomo Morabito, 'The Internet of Things: A survey', Computer Networks Volume 54, Issue 15, 28 October 2010, Pages 2787-2805
- [5] Dominique Hazael-Massieux, 'Web of Things', <http://www.w3.org/2013/Talks/dhm-wot/#/connectivity>
- [6] 인민교, 이승윤, '사물웹 표준화 동향', TTA 저널, Vol. 148, 2013
- [7] ITU-T Recommendation Y.2063, Framework of Web of Things, ITU-T 2012.7.
- [8] W3C Workshop Report on the Web of Things, <http://www.w3.org/2014/02/wot/report.html>
- [9] H.WoT-SA 'Web of Things Service Architecture'(New): Output draft(Sapporo, 30 June - 11 July 2014)
- [10] Roy T. Fielding, Richard N. Taylor, 'Principled Design of the Modern Web Architecture', ACM Transactions on Internet Technology, Vol. 2, No. 2, May 2002
- [11] Rich Quinnell, 'The Many Standards of the IoT', IoT World, 1/15/2014, <http://bit.ly/1ygRZYL>