



From the Wireless Sensor Networks (WSNs) to the Web of Things (WoT): An Overview

Mina Younan^{1,3}, Sherif Khattab^{2,4}, and Reem Bahgat^{2,4}

¹ Faculty of Computers and Information, Minia University, Egypt.

² Faculty of Computers and Artificial Intelligence, Cairo University, Egypt.

³ mina.younan@mu.edu.eg

⁴ {s.khattab, r.bahgat}@fci-cu.edu.eg

* Correspondence: mina.younan@mu.edu.eg

Abstract

In the last two decades, Wireless Sensor Networks (WSNs) are gaining more popularity, where the concept of WSN always exists when everything connects. Almost of WSN applications cover wide area and large spaces for assessing and monitoring certain phenomenon. Moreover, WSN components have been integrated in daily life objects or things (object, place, and person), so that they could be monitored and controlled. As a result, a new paradigm called the Internet of Things (IoT) connects WSN components to the Internet to be globally monitored and controlled representing the surrounding environmental events and conditions. The future IoT is called the Web of Things (WoT), which visualizes the IoT data (sensory data) using current web tools and services (HTTP, RESTful services). This paper presents an overview of the WSNs, the IoT and its future paradigm (WoT) discussing key elements, main layers, main challenges, and annotation formats.

Keyword- Wireless Sensor Networks (WSNs), Internet of Things (IoT); Web of Things (WoT); Smart Things (SThs); Entities of Interest (EoIs).

1. Introduction

The proliferation of devices equipped with sensors and actuators that control and provide information about surrounding environments increases in the last decades to reach order of billions [1] [2]. Sensors allow states of things (e.g., places, devices, etc.) that they represent to be inferred. In a sense, sensors and actuators convert things to Smart Things (SThs) and Entities of Interest (EoIs), and environments to smart spaces. Converting things to SThs and EoIs is the key element in the Internet of Things (IoT) [3] [4]. The Internet of Every Thing (IoE) is another concept for connecting everything to the Internet. As a result, the IoT could be defined as a global network of connected computers, persons, and everything to reflect integration of daily life objects [5] [6], where application areas of the Wireless Sensor Networks (WSNs) expand to cover daily life spaces providing services such as monitoring, controlling and searching for SThs and EoIs [7]. Applications areas of the WSNs need professional and expert users, while users of IoT applications deal with ordinary users as well [8]. However, number of users in the IoT increases, but it is expected to be less than number of connected devices [2] [8].

The Web of Things (WoT) is a new paradigm that works on top of the IoT for visualizing sensory data in the IoT using current web tools and services. WSNs moves toward the IoT layer by controlling sensors and actuators through the Internet. Integration of sensors and actuators in ordinary things and places for representing their states in the IoT produces new challenges. For example, building IoT using different formats (Microformat and Microdata), this challenge decreases interoperability between IoT networks [9].

The WoT charts sensory data enabling users to deduce and extract summary and abstract knowledge about sensors and things they represent. Searching for STHs and EoIs is the key service in the WoT. Dynamicity of information is the main challenge in the WoT, where STHs' values and EoIs' states may change frequently in few seconds. As a result, crawling and indexing the WoT are hot topics for research and study. For more accurate results, WoT testbeds are used providing sense of the physical phenomena that STHs represent [10] [11] [12].

The remaining of this paper is organized as follows. Next section presents the most essential concepts in the state-of-the-art of IoT and the WoT. The main concepts in this section relate to the process of moving smart devices (e.g., sensors and actuators) from the WSN to be integrated into the WoT by connecting them to things and objects to monitor and control their states through the Internet using current web services and tools. Section 3 discusses challenges of the IoT and the WoT followed by state-of-the-art WoT testbeds in Section 4. Applications' areas of the embedded systems in the digital world are presented in Section 5. Section 6 presents paper conclusion.

2. From the WSN to the WoT

From the WSN to the IoT moving forward to the WoT, this trend has spread in the last two decades [1] [3] [4]. General purpose is sought for monitoring and controlling environmental conditions and events in the surrounding environment, such as temperature, humidity, pressure, sound, darkness level, power consumption, etc. In these sub-sections, we will discuss the process of moving embedded devices to everyday objects highlighting differences between WSN, IoT and WoT concepts, which are important as a prerequisite to a discussion of building WoT applications that is the core subject for the rest of this book.

2.1 Wireless Sensor Networks

Wireless Sensor Network (WSN) is a network of wide variety of sensors that are spatially distributed and connected wirelessly to cover certain area or environment. Main concepts that describe and relate to main features of the WSNs are: ubiquity, wireless, mobility, smart spaces, M2M, distributed, embedded, etc. these concepts exist where everything connects. WSNs are technological solutions for implementing low cost and self-configuring networks. WSN disseminates surrounding environment with smart controllers (i.e., sensor nodes with self-configuring capabilities)¹. Sensor nodes in the WSNs monitor physical or environmental conditions, they pass their data to a main location, called base-station or coordinator, to be stored and processed after that by users and machines. Passing data in modern WSN is done bi-directionally to control and manage sensors deployed in the network. WSNs components, sensors component, common WSNs topologies, and simulating WSNs using Cooja [13] are discussed in the following sub-sections.

2.1.1 WSN components

WSNs have multiple components to produce sensory data, manage data and other components in the network of the covered area. Such as shown in Figure 1, WSNs components [14] are: (1) gateway, which acts as an interface between the application platform that runs on a base-station (e.g., PC) and sensor nodes in the WSNs, it is considered as a central point for expanding WSNs to be integrated with other networks, (2) Relay Node, which is responsible for extending the covered area in the same network, it acts as a middle layer between sensor nodes and coordinators (base-stations), (3) Leaf Node, also called end-point device, which connects sensors and actuators directly, (4) Sensor and Actuator, are two types of smart devices that measure and control environmental events and things.

WSN consists of sensor nodes, for sampling data from the environment, and gateways for managing sensor nodes. Each sensor node in the WSN has main components: (1) radio transceiver, (2) a microcontroller to interface with sensors and (3) power resource (battery). Sensor nodes in WSNs are equipped also with several sensors for measuring environmental events and conditions, local storage, CPUs and radio communication facilities, allowing them to both sense the local environment and communicate locally with other sensors in order to construct semantically rich

¹ Paolo Pagano (SSSA, Pisa), "Wireless Sensor Networks"

conclusions about the environment that they are sensing, forming what is called Internet of Things (IoT), as will be explained in the next sub-section.

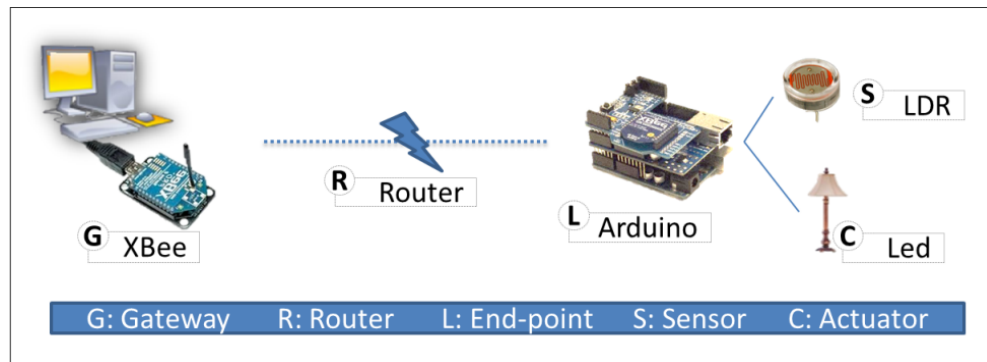


Figure 1: WSN components.

Due to constrained resources of sensor nodes in the WSNs, where limits of restricted power is the major issue in WSNs, a Periodic Data Prediction Algorithm (P-DPA) [15], is used for reducing number of transmissions in WSNs which increasing lifetime of sensor nodes. Predicting next sample value of a certain sensor node depends on correlation between consecutive observations from its historical values. Historical values of a sensor are utilized by a prediction algorithm as inputs for producing the predicted value as an output. Using a threshold and implementing aggregation functions on sensory data reduces number of transmissions. Dataset collector application in the WoT testbed in [12] used 'ChangeOnly' rule to receive changes only.

2.1.2 Network topologies

In a densely WSN, each sensor node can communicate wirelessly with adjacent nodes, which decreases its lifetime due to power consumed in transmission processes. Thus, for building a WSN that covers certain area, we have to cluster and group sensor nodes selecting the most appropriate technology and protocol (e.g., Zigbee) for communications between nodes. Network topology should be selected as well for configuring WSN nodes to cover the required environment. Number of nodes in the WSN and physical area are main factors that should be considered in network topology selection.

Cecilio and Furtado [14] discuss common network topologies and their pros and cons concluding that star topology consumes less power due to less number of edges between the coordinator and an end-point device, while mesh topology is more powerful for covering larger spaces. Implementing hybrid combination of network topologies for joining thousands of sensor nodes takes benefits of simplicity, lower power consumption, and covering larger spaces. Communication between sensor nodes could be done using single-hop or multi-hop data path [14] according to implemented topology in the WSN, i.e. information exchange occurs in WSNs between an application platform and one or more sensor nodes.

Christin et al. [16] presents three approaches for connecting the WSNs to the Internet forming the IoT paradigm that will be discussed in Section 2.2. First approach is to use single gateway for connecting a WSN to the Internet; this approach is called independent WSN. Second approach, is to compose independent WSNs using single gateway per the WSN, this approach is a hybrid of multiple independent networks. The third approach is to connect multiple sensor nodes using one hop in the WSN.

2.1.3 Simulating the WSN using Cooja

Several studies [11] [12] [17] summarize the differences between existing simulators according to a set of criteria. WSN could be built physically or virtually using simulators like Cooja [13]. Cooja simulator aids researchers and students to simulate WSN easily by adding different types of available nodes in Cooja, where TinyOS [18]

applications are built at first then attached to selected motes. TinyOS supports open source written in NesC² for different targets, for example ‘REST-Client-Server’ that simulates simple IoT. The steps for running this example are in two axes:

- Server side

1. Start *cooja* and *load* simulation “rest-server-example.csc”
2. **Make TARGET=cooja rest-server-example.csc**
3. open new terminal, go to the same directory and connect to cooja using tanslip6 : **make connect-router-cooja**
4. Coap or http is needed to interact with cooja
5. IP: `aaaa:0121:7402:0002:0202:8080`

- Client side

Use **curl** as a client to interact with the cooja motes running REST Codes.
Curl -H “User-Agent: curl” aaaa:0121:7402:0002:0202:8080/helloworld #get helloworld plain text

Building most of WSN systems needs experts in this field, where most of WSN devices communicate using low-level protocols. So in brief, the major WSN systems and devices were not built for the large public use, and its major goal is to monitor and control conditions and events in a certain environment by distributing sensors and actuators.

2.2 The Internet of Things (IoT)

More progress has been done on HTTP protocol to run on embedded devices [3]. As a result, augmentation of everyday (things or devices) objects to the internet has been increased. The notation of Internet of Things (IoT) comes from terms of “Ubiquitous Computing” or “Pervasive Computing” [3], where Ubiquitous computing means distributing sensors and actuators on persons and surrounded environment of interest to provide information about things those sensors and actuators represent. The IoT is defined in [3] as “a system of physical objects that can be discovered, monitored, controlled or interacted with by electronic devices which communicate over various networking interfaces, and eventually can be connected to the wider Internet”. Pfister [5] defines the IoT as “a global network of computers, sensors, and actuators connected through internet protocols”. Then, connecting more than WSN using the internet is called IoT (Figure 2), where the potential of the IoT is to connect the physical world with the virtual world for purposes of monitoring, controlling, and accessing objects (devices, places, etc.) [1] [19].

² NesC: <http://www.tinyos.net/tinyos-1.x/doc/nesc/ref.pdf>

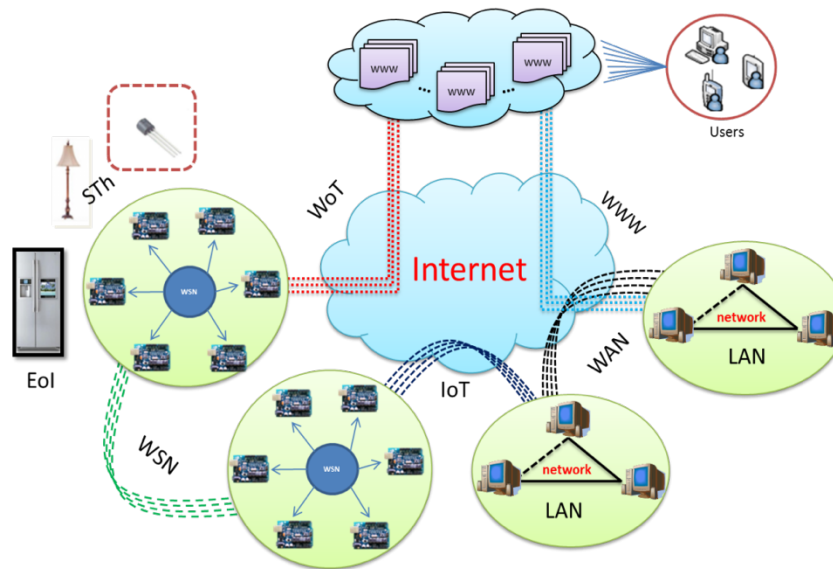


Figure 2: From the WSN to the WoT

In brief, The IoT has showed that internet is not only about network of computers and servers but also a network of smart objects. Due to the progress achieved in the manufacturing of computer hardware, almost all of the objects (things) are equipped with computational power and communicational capabilities, which made them smart (e.g. smart phones). In other words, the integration of smart objects to the internet expands the internet connection boundaries between web servers to be between smart objects such as the communication between sensor nodes in a wireless sensor network. Developers built IoT applications in different sights according to purposes and nature of the environment for which the IoT application is built.

Main layers of the IoT is shown in Figure 3; (a) sensing, perception, physical, or SThs' layer, (b) network layer, (c) cloud layer, and (d) application layer. WoT tools facilitate monitoring process for human user, thus it could be considered that it is resides the application layer. A new paradigm which concerns the financial trend in the IoT is called industrial IoT (IIoT) or industry 4.0 [11]. Due to increase dependence on IoT applications in various field especially in healthcare, smart cities, and security, fog layer is added as a middle layer. This layer enhances performance of IoT applications, where capabilities of the 3rd layer (cloud) become near from SThs to process sensory data streams [20]. SThs are key elements in the IoT which forms the perception layer (i.e., sensing layer). Thus Things, SThs, resources, and EoIs are main concepts in IoT and WoT. They have differences in meaning but the main goal is that they are used for integrating the physical world into the virtual world [1] [21]. The following sub-sections discuss the two concepts; SThs and EoIs.

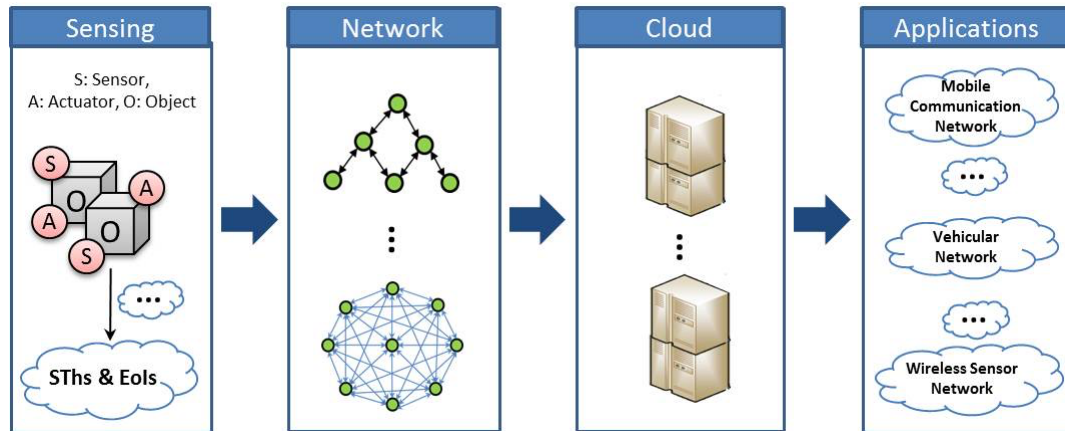


Figure 3: Main IoT layers [6].

2.2.1 Smart Things (SThs)

Augmenting everyday objects with sensors and actuators convert objects or things into smart things and things' environment into smart spaces, so they could be monitored and controlled through the internet. Figure 4 (a) indicates the steps to convert things into smart things (SThs)³; (1) when things support IP connections, converting things to smart things and integrating them in the IoT is done directly, but (2) when things speak low level protocol (e.g., 6LoWPAN) then gateway is used as a bridge for integrating things in the IoT. For example, in Figure 4 (b), Lamp is a thing and is attached with smart devices (Arduino) to convert it to a smart lamp. Then STh is considered as a physical object that is attached with one or more embedded device (sensor, actuator, computation or communication interfaces) [3]. Examples of embedded components are as follows:

- Sensors: temperature, humidity, loudness, pressure, light, motion, etc.
- Actuators: displays, sound, motors, etc.
- Computational power for run programs and logic operations
- Communication interfaces: wired and wireless.

³ the web of things tutorial at <http://sensorlab.ijs.si/>

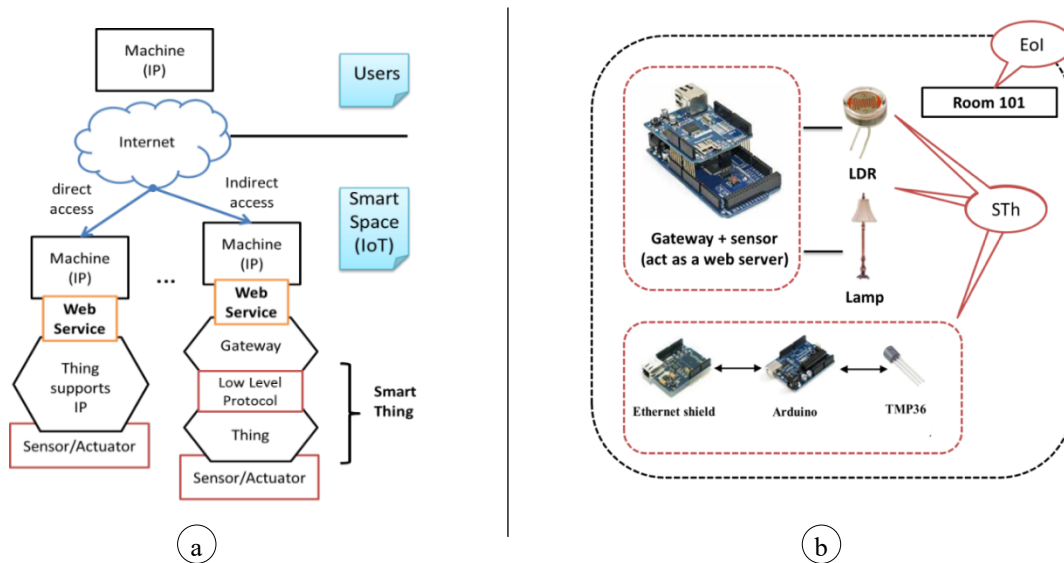


Figure 4: (a) Converting things into smart things (SThs): direct and indirect integration of SThs in the IoT. (b) Converting things to EoIs: LDR, Lamp, TM36 (SThs) attached to room 101 (EoI).

Guinard [4] defines a meta-data-model that hosts the most important SThs attributes, and Hong [22] discusses two scenarios for searching for and discovering SThs in the WoT, identifying two types of aspects; (a) unique aspects like serial number and URL, and (b) characteristic aspects like physical location.

2.2.2 Entities of Interest (EoIs)

Entity of interest (EoI) is an object (e.g., room, person, etc.) that is attached with or hosts a STh for representing its state, such as shown in Figure 4 (b), where room is called an EoI and has three SThs; LDR (photocell sensor) that represent the room illumination by measuring darkness level, a smart lamp is an actuator in the room where it adjusts its state depending on the LDR value, and LM35 (temperature sensor) is for representing the room temperature.

2.3 The Web of Things (WoT)

As been mentioned earlier, the IoT has become more popular in the last two decades, where everyday objects integrate the IoT to support real-time access, control, and information. In the previous sub-section, the IoT is defined as it is a global network of computers and physical objects that are attached with smart components like sensors and actuators to represent their states.

The web has been used not only for publishing static information about things but also dynamic information about them. This is due to sensors and smart devices that have been connected to the internet in existence of the web 2.0. In parallel with appearance of the IoT, the web 2.0 supports loosely coupled services. As a result, the Web of Things (WoT) is considered as a layer on top of the IoT, which uses current Web services and tools for building loosely coupled IoT applications, i.e., from web pages to web things, where web pages are founded upon: (1) IRIs, (2) HTTP, and (3) HTML for pages discovery. The WoT is founded upon the same elements, but using one protocol will not satisfy all connections between things, and HTML (Search engines follow links of SThs and EoIs in pages) has to embed additional formats for representing SThs and EoIs' properties and states [23]. So it is called the application layer of the IoT that converts physical objects to virtual objects (web pages). The HTTP protocol is used as a transfer protocol in the web but in the WoT it is used as an application protocol [22]. Enhancing aspects of the searching process (e.g., crawling, indexing, and searching) in the WoT increases its popularity.

2.3.1 Layers of the WoT

Because WoT is new fusion from the IoT, thus similarly to the IoT, The WoT consists of four layers [4] [24], namely as follows:

1. Accessibility Layer: integrate SThs directly or indirectly in the WoT.
2. Findability Layer: describing SThs and EoIs so that they could be crawled, indexed, and searched using a traditional search engine, like Google, or using special search engines, like Dyser [25] that allows human users and applications to search for SThs and EoIs in the WoT.
3. Sharing Layer: enabling the first two layers requires that SThs and EoIs are represented as Web resources. As a result they become available directly to the world from the Web.
4. Composition Layer: pushing boundaries to the WoT, so that simple applications could be created on top of SThs making the WoT closer to the end-users.

On top of each layer, a WoT application can be built. The WoT applications apply Representational State Transfer services (RESTful services) [26] for visualizing SThs readings and EoIs states and making them more accessible, such as shown in Figure 5, where SThs are various and need to integrate with each other [22]. REST APIs are executed over the network on remote hosts [22] [26] for increasing interoperability between loosely coupled services in the web. Due to the simplicity of RESTful Application Programming Interface (REST APIs), which supports the integration between web services (WoT applications) [22] by encapsulating the functionality into a reusable component, REST APIs depend on HTTP verbs (GET, PUT, UPDATE, and DELETE) and therefore the integration of SThs data across several cloud services will be much faster [3].

The Web Services Description Language (WSDL) [27] is used in somewhere (SOAP) for describing functionalities of SThs but it is not sufficient for describing functional behaviors of the SThs. Leveraging the WoT as an application platform for integrating resources (e.g., SThs pages) as services using RESTful services is better than generating codes from WSDL of resources. Web services are considered as the main method for accessing WoT devices [28]. Mayer et al. [19] propose a hierarchical infrastructure for building WoT to enhance the performance of the searching service. Nodes receive queries then pass them to the right nodes in the network to answer the queries. The searching scenario starts by getting a list of sensors that can answer a query according to their static properties and predicted values. After that, the identified sensors are queried to check their current values, which are used for ranking the search results. The searching scenario is integrated into the testbed in [12].

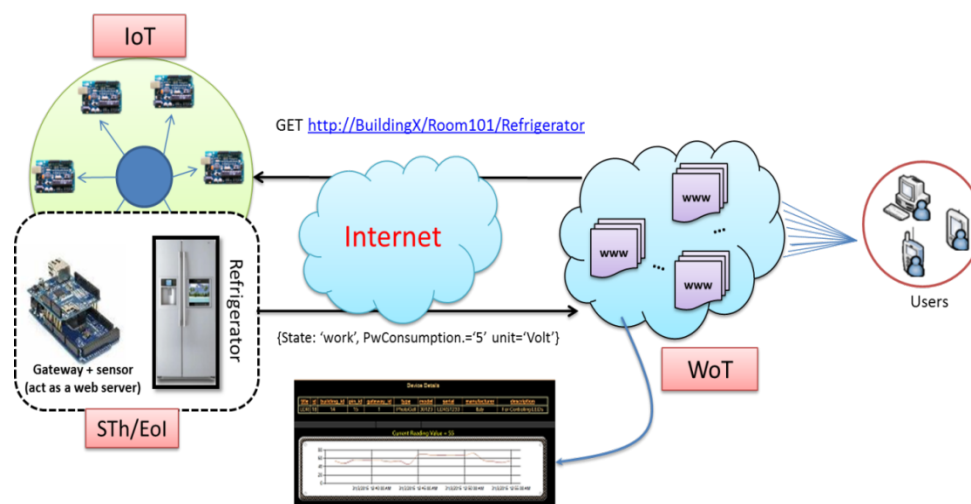


Figure 5: Querying status of a smart refrigerator in the WoT by calling GET request (RESTful Service).

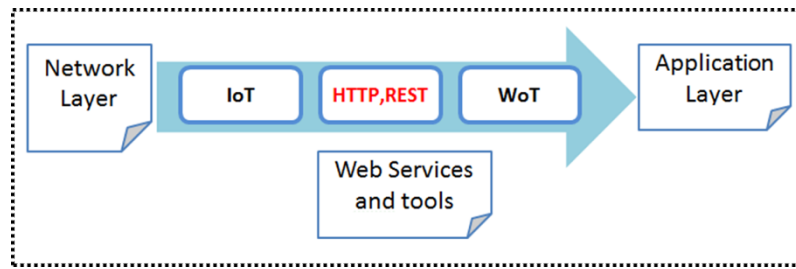


Figure 6: Summary for moving from the IoT to the WoT.

In brief, Figure 6 summarizes the difference between the IoT and the WoT. As mentioned earlier, the usability of embedded devices has been growing from the ordinary use in the WSN to the digital augmentation in everyday objects, which called the IoT. For accessing STHs through the internet in the IoT, web applications, tools and services should be used. Embedding web servers on STHs converts them to RESTful resources. The idea of expanding usability of the IoT by making its devices prime citizens in the web depending on existing web tools and services is called the WoT [3] [28].

2.4 Semantic Annotation Formats

Different formats have been used in the field of the WoT for representing STHs and EoIs metadata (i.e., information about sensory data), which help web search engines to discover and analyze web pages in order to create indices [28]. The web spiders search for *<meta>* tag in HTML pages. The most popular formats that are used for marking up metadata of STHs and EoIs pages (i.e., semantic representation) in the WoT are: Microformat, Microdata, Resource Description Framework (RDF), Resource Description Framework in attributes (RDFa) [22], and JavaScript Object Notation (JSON) [29].

Hong in [22] compares between Microformat and RDFa concluding that using Microformat is better than using RDF, because extracting RDF triples needs parsing due to the freedom of adding additional vocabularies by RDF publishers, this in addition to its complicated syntax. A few details about Microformat and Microdata are presented in the following sub-sections.

2.4.1 Microformat

It is designed for embedding semantics in STHs and EoIs resources by adding information directly in HTML tags in their pages. Microformat could be used by human users and machines [29]. Search engines crawl and parse pages for building indices. Microformat is built upon standards, all types of Microformat are small HTML patterns [22]. For example, *hCard* is used for contact information including sub-patterns *Geo* and *adr*, *hProduct* for describing product attributes (details), *hRESTS* is used for annotating STHs and EoIs resources with information about services they offer by declaring inputs, outputs, and services formats, so it limits people entry. Figure 7 shows an example of geo Microformat. The best method for describing STHs is to use compound types of Microformat such as *hREST*, *hCard*, *hProduct*, and *Geo* [22] [30].

```
<div>
  </span>
  <span class="geo">
    <abbr class="latitude" title="48.816667">N 48° 81.6667</abbr>
    <abbr class="longitude" title="2.366667">E 2° 36.6667</abbr>
  </span>
</div>
```

Figure 7: Example of geo Microformat.

2.4.2 Microdata

Microdata is used like Microformat for embedding semantic information about STHs and EoIs resources in HTML tags. Microdata is part of the HTML5 specification [30]. The main difference between Microdata and Microformat is that Microformat overloads HTML class tags, while Microdata uses new tag attributes, like *itemscope*, *itemprop*, etc., where semantics of Microdata depend on vocabularies' definitions.

Mayer in [28] compares between Microformat and Microdata and concludes that creating RESTful services using Microdata is desirable. Microdata closes the gap between simplicity of Microformat and more abstract concepts based on RDF [29]. Authors in [28] and [29] provide a method for solving the problem of using multiple formats (e.g., Microformat and Microdata) in the WoT. They propose to add multiple strategies for parsing and producing information in the intended format. However, their work does not result in a dataset. They implemented an algorithm [29], called extensible discovery service, as a Web application that asks users about sensor page URL and retrieves information about devices if and only if the page is written in one of a set of pre-defined formats.

To sum up main concepts could be summarized as follows:

- STHs = Things + Smart Device (e.g., Microcontroller, Sensor, Actuator, etc.).
- EoIs = STHs + physical object (e.g., car, room, person, etc.).
- IoT = STHs + EoIs + Internet.
- WoT = IoT + Web Services and Tools (HTTP protocol and RESTful Services).

3. Challenges of the IoT and the WoT.

Due to the great interest in converting things into STHs, more challenges have been found [4] [6] [7] [8] [23], where IoT systems are built in different manners. In general, challenges naturally associated to the IoT, are:

- Huge number and heterogeneity of connected devices
- There is no standardized naming for STHs' attributes.
- It is difficult for developers to track and learn different platforms.
- Dynamic states (readings) and dynamic attributes (locations for movable objects) [9].
- The logical path isn't considered as an attribute of the STH, however it is used in the search process. For example, Dyser [2] [5] answers queries in the form of name: value (e.g., 'find room X occupancy: empty').

Challenges that are naturally associated to the WoT in brief are:

- Most of WoT pages host dynamic parts that are coded using Java Script Object Notation (AJAX) for monitoring STHs and EoIs live. Therefore, most of search engines' spiders cannot crawl them.
- Single STH's state could be represented using different words.

Challenges related to search service in the IoT and the WoT [14] could be summarized as follows:

- Real-time queries: due to the huge number of STHs/EoIs and their dynamic states.
- Searching in multiple domains: due to non-standardized naming of STHs properties and using different formats for writing STHs and EoIs data [31].

4. WoT Testbeds

Several studies [10] [11] [12] [17] [32] discuss and compare between existing simulators and testbeds using general criteria, such as the number of nodes, heterogeneity of hardware, and mobility. However, none of them discuss WoT features, such as STH's logical path, supported formats in which EoIs' states are presented, and accuracy of the datasets generated by the testbeds (frameworks). The comparison of testbeds in this section is divided into two main categories according to the testbed architecture: (1) infrastructure layer (IoT environment) and (2) software layer.

- Elements of the infrastructure layer are:
 1. Number of nodes and possibility of discovering new nodes (scalability). Number of nodes in WSN testbeds should be more than 10, but in IoT should be more than 1000.
 2. Environmental properties: relate to certain physical phenomenon (e.g., temperature, light, and humidity).

3. Availability and portability: the ability to deploy and re-deploy the testbed in different places.
 4. The hardware heterogeneity: the ability to include different types of sensors.
- Elements of the software layer are:
 1. Software heterogeneity: the ability to run the testbed on multiple operating systems.
 2. Re-usability (repeatability and reproducibility): the ability to save an experiment to re-run it later.
 3. WoT features: include data sharing, user involvement, GUI, RESTful APIs, STh's attributes and formats. These features identify the tasks that the users can do on testbeds, the possibility to support data visualization (high-level knowledge), and the possibility to share datasets.
 4. Real-time information: measures if the testbeds produce imitative information (as in simulators) or real information in the real-time (as in physical testbeds).

Younan et al. [12] argue that WoT research needs a special type of testbeds, such the WoT testbed in [1], to support absent WoT features that are needed to leverage existing testbeds to the WoT level.

5. Embedded System Applications for a Digital World

WSN, IoT and WoT have variety of applications in different areas and fields, and this made them more popular. Christin et al. [16] categorize WSN applications into four categories: (1) monitoring objects or things, (2) monitoring environmental events and conditions in spaces, (3) monitoring interaction between objects and spaces, and (4) monitoring human interaction with things and spaces. Figure 8 shows statistics about the popularity of embedded systems applications in different fields. Statistics are done on Google, Twitter, and LinkedIn [33]. For example but not exhaustively, most popular areas of embedded systems applications [3] [7] are: smart home (e.g., creating smart lighting systems in homes), wearable (e.g., smart watch for measuring body temperature and blood pressure), smart cities (e.g., smart parking), etc. When we get an advanced warning, we can understand and better manage what we currently have, this could be done by monitoring and detecting some events such as follows:

- a. Monitoring (seas, air pollution, water quality, waste water, machine health, health care, large areas especially in military for detecting enemy intrusion, etc.).
- b. Detection (forest fire, landslide, seismic areas, swimming conditions, etc.).
- c. Prevention (firing, contamination, natural disaster, etc.).
- d. Measuring quality of drinking water (pollution levels), and water level in rivers and dams.

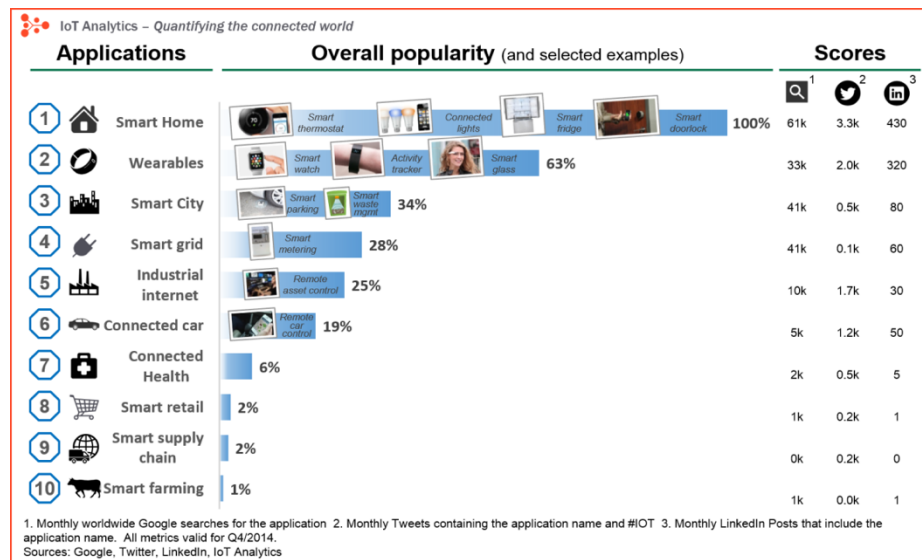


Figure 8: Quantifying connected world (IoT) on Google, LinkedIn, and Twitter websites [33].

6. Conclusion

This paper presents the main concepts of in the field of the IoT and discusses the process of moving embedded devices from the WSN to the WoT and concluding that the WSN is a network in which sensors are connected wirelessly and are distributed to cover a certain place to give information about certain phenomena. The IoT focuses on the network layer for interconnecting devices, while the WoT virtualizes the IoT focusing on the application layer. The WoT uses existing web tools and services (RESTful services) for building applications on top of the IoT layer. SThs such as indicated earlier are the new versions of the traditional things after converting them into smart things by attaching them with sensors, actuators or other smart devices, to speak about themselves representing things that they are attached with. EoI is the object, which is considered smart after attaching a STh with it.

Sensors can provide great benefits if their readings are presented in a meaningful and friendly way to users and machines. Searching for SThs and EoIs is one of the most important services in the WoT. The potential of the WoT lies in interconnecting and integrating services with human users in different WoT networks. Human users prefer searching in high-level states of SThs and EoIs in real-time and in the future. Every day augmentation of SThs in the IoT and the WoT makes sizes of the WoT search engine indices increase with the same degree of SThs integration, and the interval between periodical indexing processes decreases with the same degree of SThs changes.

References

- [1] M. Younan, S. Khattab, and R. Bahgat, "An Integrated Testbed Environment for the Web of Things," in *ICNS 2015 : The Eleventh International Conference on Networking and Services*, ISBN: 978-1-61208-404-6, Rome, Italy, May, 2015.
- [2] M. Blockstrand, T. Holm, L.-Ö. Kling, R. Skog and B. Wallin, 11 February 2011. [Online]. Available: http://www.ericsson.com/news/110211_edcp_244188811_c. [Accessed 25 January 2015].
- [3] D. Guinard and V. Trifa, "From the Internet of Things to Web of Things," in *MEAP - Building the Web of Things*, 8 ed., Manning, 2015, pp. 1-28.
- [4] D. Guinard, "A Web of Things Application Architecture - Integrating the Real-World into the Web," ETH Zurich, Zürich, 2011.
- [5] C. Pfister, Getting Started with the Internet of Things, First Edition ed., B. Jepson, Ed., O'Reilly Media, Inc., May 2011.
- [6] M. Younan, E. H. Houssein, M. Elhoseny, and A. A. Ali, "Challenges and recommended technologies for the industrial internet of things: A comprehensive review," *Measurement*, vol. 151, p. 107198, 2020..
- [7] Libelium Comunicaciones Distribuidas S.L, December 2015. [Online]. Available: http://www.libelium.com/top_50_iiot_sensor_applications_ranking/.
- [8] M. Younan, S. Khattab, and R. Bahgat, "WoTSF: A Framework for Searching in the Web of Things," in *INFOS 2016 : The 10th International Conference on Informatics and Systems*, ACM, Egypt, Cairo, May, 2016.
- [9] M. Younan, E. H. Houssein, M. Elhoseny, and A. A. Ali, "Performance analysis for similarity data fusion model for enabling time series indexing in internet of things applications," *PeerJ Computer Science*, vol. 7, p. e500, 2021.
- [10] A. Gluhak, S. Krco, M. Nati, D. Pfisterer, N. Mitton, and T. Razafindralambo, "A Survey on Facilities for Experimental Internet of Things Research.," *IEEE Communications Magazine, Institute of Electrical and Electronics Engineers (IEEE)*, no. <10.1109/MCOM.2011.6069710>. <inria-00630092>, pp. 58-67, 2011, 49 (11).
- [11] J. Miloš, N. Zogović and G. Dimić, "Evaluation of Wireless Sensor Network Simulators," in *the 17th Telecommunications Forum (TELFOR 2009)*, Belgrade, Serbia, 2009.
- [12] M. Younan, S. Khattab, and R. Bahgat, "Evaluation of an Integrated Testbed Environment for The Web of Things," in *IntSys15v8n34 : International Journal On Advances in Intelligent Systems*, vol. 8, no. 3&4, pp. 467-482, November 2015.
- [13] h. b. T. AS IS. [Online]. Available: <http://www.contiki-os.org/>. [Accessed January 2015].
- [14] J. Cecilio and P. Furtado, "Wireless Sensor Networks: Concepts and Components," in *Wireless Sensors in Heterogeneous Networked Systems*, Springer International Publishing, 2014, pp. 5-25.
- [15] J. Zhao, H. Liu, Z. Li, and W. Li, "Periodic Data Prediction Algorithm in Wireless Sensor Networks," in *CWSN 2012: Advances in Wireless Sensor Networks*, Huangshan, China, 2013.

- [16] D. Christin, A. Reinhardt, P. Mogre, and R. Steinmetz, "Wireless Sensor Networks and the Internet of Things: Selected Challenges," in *Proceedings of the 8th GI/ITG KuVS Fachgespräch Drahtlose Sensornetze*, 2009.
- [17] H. Sundani, H. Li, V. K. Devabhaktuni, M. Alam and P. Bhattacharya, "Wireless Sensor Network Simulators - A Survey and Comparisons," *International Journal Of Computer Networks (IJCN)*, vol. 2, no. 6, pp. 249 - 265 , February 2011.
- [18] P. Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, and A. Woo, "TinyOS: An Operating System for Sensor Networks," in *Ambient Intelligence*, W. Weber, J. M. Rabaey and E. Aarts, Eds., Springer Berlin Heidelberg, 2005, pp. 115-148.
- [19] S. Mayer, D. Guinard, and V. Trifa, "Searching in a Web-based Infrastructure for Smart Things," in *Proceedings of the 3rd International Conference on .the Internet of Things (IoT 2012),IEEE*, Wuxi, China, October 2012.
- [20] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, "Industrial Internet of Things: Challenges, Opportunities, and Directions," *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS*, VOL. 14, NO. 11, NOVEMBER 2018, pp. 4724-4734.
- [21] B. Negash, T. Westerlund, and H. Tenhunen, "Towards an interoperable Internet of Things through a web of virtual things at the Fog layer," *Future Generation Computer Systems*, www.elsevier.com/locate/fgcs, p. 96–107, 2018.
- [22] S. Haller, "The Things in the Internet of Things," *Poster at the (IoT 2010). Tokyo, Japan*, vol. 5, no. 26, p. 4, November 2010.
- [23] S. Hong, "Mobile Discovery in a Web of Things," Zurich, April 22. 2010.
- [24] Dave Raggett, W3C, "An introduction to the Web of Things Framework," 2015.
- [25] L. Mainetti, V. Mighali, and L. Patrono, "A Software Architecture Enabling the Web of Things," *IEEE Internet of Things Journal*, DOI: 10.1109/JIOT.2015.2477467, vol. 2, no. 6, pp. 445 - 454, 2015.
- [26] B. Ostermaier, M. Elahi, K. Römer, M. Fahrmaier, and W. Kellerer, "A Real-Time Search Engine for the Web of Things," in *The 2nd IEEE International Conference on the Internet of Things (IoT)*, Tokyo, Japan, Nov. 2010.
- [27] Dr. M. Elkstein, "Learn REST: A Tutorial," November 2014. [Online]. Available: <http://rest.elkstein.org/2008/02/what-is-rest.html>. [Accessed 26 2 2013].
- [28] 11 2014. [Online]. Available: http://en.wikipedia.org/wiki/Web_Services_Description_Language.
- [29] S. Mayer, "Service Integration - A Web of Things Perspective," in *W3C Workshop on Data and Services Integration*, Citeseer, Bedford, MA, USA, October 2011.
- [30] S. Mayer and D. Guinard, "An Extensible Discovery Service for Smart Things," in *Proceedings of the 2nd International Workshop on the Web of Things (WoT 2011)*, ACM, San Francisco, CA, USA, June 2011.
- [31] D. Guinard, V. Trifa, F. Mattern, and E. Wilde, "From the Internet of Things to the Web of Things: Resource Oriented Architecture and Best Practices," in *Architecting the Internet of Things*. F. M. a. D. U. M. Harrison, Ed.
- [32] M. Younan, E. H. Houssein, M. Elhoseny, and A. A. Ali, "Data Reduction Model for Balancing Indexing and Securing Resources in the Internet-of-Things Applications," *IEEE Internet of Things Journal*, Vol. 8, No. 7, vol. 8, pp. 5953-5972., 2020.
- [33] I. Muhammad, A. Md Said, and H. Hasbulla, "A Survey of Simulators, Emulators and Testbeds for Wireless Sensor Networks," in *Information Technology (ITSim), 2010 International Symposium in*, Kuala Lumpur, June 2010.
- [34] IoT Analytics GmbH, December 2015. [Online]. Available: <http://iot-analytics.com/10-internet-of-things-applications/>.