



AN EMPIRICAL ANALYSIS OF WEB OF THINGS (WOT)

Dr. Firoj Parwej
Research Scholar,

Department of Computer Science & Engineering,
Singhania University, Rajasthan, India

Nikhath Akhtar
Research Scholar-Ph.D

Department of Computer Science & Engineering,
Babu Banarasi Das University, Lucknow, India

Dr. Yusuf Perwej

Assistant Professor, Department of Information Technology,
Al Baha University, Al Baha,
Kingdom of Saudi Arabia (KSA)

Abstract: The IoT has primarily focused on establishing connectivity in a diversify of constrained networking environments, and the next logical aim is to build on top of network connectivity by focusing on the application layer. In the Web of Things (WoT), we are thinking about smart things as first class citizens of the Web. We position the Web of Things as a purification of the Internet of Things by integrating smart things not only into the Internet, for instance the network, but into the Web (the application layer). The Web of Things is a computing concept that describes a future where day-to-day objects are fully integrated with the Web. The WoT is very homogeneous to the IoT in some ways and in others it is drastically different. The stipulation for WoT is for the "things" to have embedded computer systems that enable communication with the Web. This type of smart devices would then be able to communicate with each other using current Web standards. For instance, renowned Web languages PHP, HTML, Python, and JavaScript can be used to easily build applications involving smart things and users can leverage well-known Web mechanisms such as caching, browsing, searching, and bookmarking to communicate and share these devices. In this paper, aim to demonstrate a close-up view about Web of Things, including Web of things architecture, Open platform in Web of Thing, Web-enabling devices, Web of Thing security, use cases of Web of Things. The WoT concept, smart things and their services are fully integrated in the Web by reusing and conforming technologies and patterns commonly used for conventional Web content.

Keywords: Web of Things (WoT); Information Retrieval; Semantic Web; Web Search Engines; Web APIs; Internet of Things (IoT).

I. INTRODUCTION

Today scenario, it is a global network which connects people and we are use desktops, laptops, tablets, and phones to communicate with each other [1]. Generally the information we send around goes via servers which run websites, email software, etc. The Internet of Things (IoT) [2] is a term to describe how physical objects are being connected to the Internet so that they can be explored, monitored, controlled or interacted with. The IoT is more frequently used in the context of radio frequency identification (RFID) [3] and how physical objects are tied to the Internet and can communicate with each other [4]. IoT systems have applications across industries via their distinctive flexibility and ability to be appropriate in any environment. They ameliorate data collection, operations, automation and much more via smart devices and powerful enabling technology. As more "things" on planet Earth are converted to the inventory of digitally connected Internet devices, the roles and accountability of web developers and technology managers will need to develop in keeping pace with the ever expanding list of appliances and gadgets that need a web interface. This global tendency is known as "Internet of Things", and as a vision has inspired that same premise for "Web of Things", and incorporates similar features and

application models [5]. This piece will investigate the technical features that encapsulate WoT, and I will provide examples of current use cases applications in use at present, as well as offer

some hopeful prospects for the future of the web and things. Some many things trying to communicate with one another in various ways, it is challenging to seamlessly build a single communication platform in which all things speak together efficaciously. Many have attempted to create completely new platforms of communication, so all devices can communicate efficaciously, also known as an application layer protocol. However, this is time-consuming and may be nearly unfeasible. Consequently, many others believe we must use a platform that already exists. In this context, the WoT comes into play [6]. The web has been an already established system that can permit all things to communicate with each other. The Web may be used as a type of application [5] where all things can communicate together in the most dexterous manner. The Web of Things is intended to enable interoperability across IoT Platforms and application domains.

In the first instance, it endues mechanisms to formally describe IoT interfaces to permit IoT devices and services to communicate with each other, independent of their underlying implementation, and across multiple networking protocols [7]. In WoT, applications communicate with physical objects with the familiar HTTP protocol and RESTful API. This oversimplifies the access to physical objects, assent them to be used in web applications and merged with current web resources [8].

II. THE WEB OF THINGS (WoT)

If you take a moment and think about how we, humankind, function you will quest that we have a five senses, which are

felt by corresponding organs, then the stimuli are sent to the brain via the nerves, finally the brain processes these stimuli. The outcome is most often knowledge and perhaps also actions can be triggered the brain, then transmits commands via the nerves to the muscles which then trigger moving hands, legs, talking, etc. In this context, it is to see “things” as organs which detect the stimuli. These are then sent via wireless or wired technology, usually on an IP/HTTP network to processing and storage engines and these engines then crunch the obtained information and generate knowledge. Occasionally they can also trigger an action like as sending a tweet. The Web of Things, as a concept, [9] is discovery its way to day to day life and is gaining prominence with the development of the internet towards a network of interconnected objects, ranging from cars and transportation cargos to electrical appliances [5]. This development will endure new services and will enable new directions for communication and new types of agents involved in this process. The Web of Things offers to counter this fragmentation via metadata. That enables convenient integration across IoT platforms, and which make easy for application development via a common interaction model that is independent of the underlying protocols [7]. This is expected to drive down the costs and risks involved in developing services, and assistance realize the full potential for the IoT [2]. The WoT confer economical and rapid services with high-speed connectivity integrating miscellaneous wireless network technologies. The Web of Things is a refinement of the IoT [10] by integrating smart things not only into the Internet (network), but into the web architecture (an application). The Web of Things permits interoperability between devices from dissimilar manufacturers and fosters cross-device applications. It also makes it possible for a huge group of amateurs to buy all sorts of devices, build expeditiously their smart home systems [5], and especially reuse and customize these systems effortlessly for their peerless needs and incline.

III. RELATED WORKS

The WoT integrates several research and application fields, among which embedded systems, wireless networks, software infrastructure, Web technologies and artificial intelligence. According to [11] and more recently, a Web of Things infrastructure should permit, explore objects without configuration, dynamically adapt to its environment, be secured, so that things and applications are innocuous and avoid privacy matter, permit manual or semi-automatic service composition and confer services that make sense for the users. From a more technical perspective, it should rely on Web standards to achieve interoperability [12] take into account several communication models, for instance request & response, message oriented, event based, publish subscribe [13]

[14], permit executing code on objects or delegating it to the cloud semantically conclude available functions and enrich data [15]. Preliminary surveys serve as road-maps [16] to realize IoT and WoT [17]. Their cover versions, definitions, enabling technologies and propose potential research directions [18]. Subsequent surveys focus on more specific usages. In the opinion of [19], WoT from the cloud computing perspective. In these surveys, WoTSE either receives a [20] concise discussion as a potential research topic or a short introduction presenting some representative works [21]. They either focus on one type of WoTSE or listing potential research issue without bearing in mind the state of the field. In these surveys [22], analyzes seven prototypes on nine dimensions that focus

on the ability to handle real-time, local sensor queries. In these approaches [23], WoTSE from perspective of EPC worldwide discovery service. They analyze five works on nine dimensions. Finally, evaluates the including both WoT-specific prototypes and outcome from other fields that are expected to be applicable to WoT [24].

IV. THE WEB OF THINGS ARCHITECTURE

Today scenario, web technology is ubiquitous and it has become part of our day-to-day lives and, moreover, many tech-savvy human beings without explicit training are able to create their own web applications [25]. Consequently, the Web of Things initiative goals for applying the well-known and proven patterns from the web to the demanding IoT domain [5].

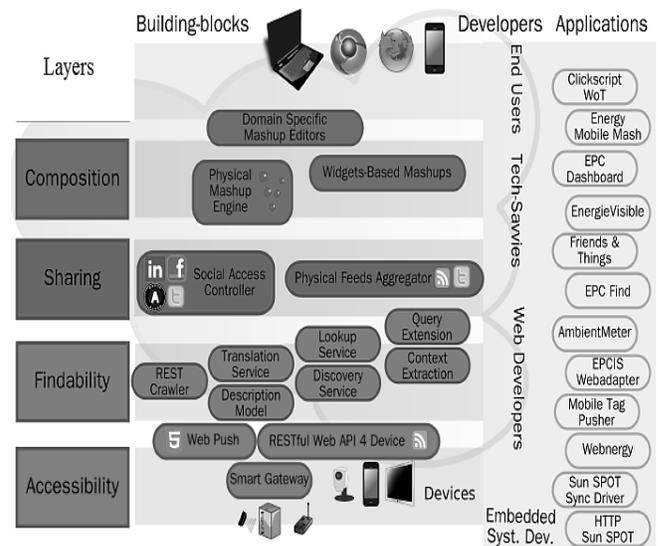


Figure 1. The Web of Things Architecture

As an outcome, devices can be browsed and bookmarked, Web pages can directly include real-time data from sensors, and users can build physical mashups that augment and control their day-to-day objects [26]. In this section, we are discussing the four layers of web of things architecture; it is based on, the device accessibility layer, findability layer, sharing layer and composition layer as shown in figure 1. The WoT architecture is an endeavor to structure the galaxy of web protocols and tools into a useful framework for connecting any device or object to the web. Users should be able to access and use smart things absentia the need for installing additional software [27]. From a web browser they should further have means to directly extract, share, and save smart things data and services.

This empowers creating applications [3] in which real-world data are directly consumed by resource coercible devices, namely mobile phones or wireless sensor nodes absentia need dedicated software on these devices.

A. Accessibility Layer

This accessibility layer is accountable for turning any Thing into a web thing that can be interacted with using HTTP requests just like any other resource on the Web. In other words, a web thing is a REST API that permits to communicate with something in the real world [24]. The accessibility layer in the WoT is built around two main

patterns firstly all things should be exposing their services via a RESTful API, either directly or via a gateway. REST is an architectural style at the root of the programmable web thanks to its implementation in HTTP 1.1. As an outcome, if things offer RESTful APIs over HTTP, they get a URL and become seamlessly integrated with the World Wide Web and its tools like as browsers, hyperlinked HTML pages and JavaScript applications. In this context, various designs describing how the services offered by things can be accessed via REST have been proposed. Secondly, the entreaty response nature of HTTP is often cited as one of the limitations for IoT use-cases as it does not match the event-driven nature of applications that are common in the wireless sensor networks. To get the better of this shortcoming while keeping a focus on fostering integration with the web, various authors have suggested the use of HTML5 web sockets either natively or via the use of translation brokers for instance translating from MQTT [3] or CoAP to web sockets.

B. Findability Layer

The web faced homogeneous challenges, when it moved from a hypertext of several thousands of documents to an application platform interconnecting an unprecedented number of documents, multimedia content and services. The concentrate of this layer is to provide a way to explore and locate things on the web and hence is strongly impression of the semantic web. This layer makes sure that your Thing can not only be easily used by other HTTP clients, but can also be findable and automatically usable by other WoT applications [25]. The approach here is to reuse web semantic standards to describe things and their services. This enables finding for things via search engines and other web indexes as well as the automatic generation of user interfaces or tools to communicate with Things. This enables finding for things via search engines and other web indexes as well as enabling machine to machine communicate based on a small set of well-defined standards and formats.

C. Sharing Layer

This accountable for the share layer, which specifies how the data generated by Things can be shared in an efficient and secure manner over the web. At this level, another batch of web protocols assists. Firstly, TLS, the protocol that makes transactions on the web securely. Then, techniques like as delegated web authentication mechanisms like an oath which can be integrated with our Things' APIs. The most basic needs for a WoT sharing platform is to be secure in order to make sure that access to smart things is not allowed to attackers. A WoT sharing platform should be uncomplicated and easy to use. Sharing the platform should also react mental models users are already familiar with [28]. In special, it should as

much as possible react the existing faith and social models of users. A WoT sharing platform should also support advertising the shared things directly on the web. In order to decrease the load of users and ameliorate security, sharing a smart thing and advertising the fact that it was shared should occur on the same channel without explicitly make known credentials.

D. Composition Layer

At huge, the Web of Things materializes into an open ecosystem of digitally enlarge objects on top of which applications can be created using standard web languages and

tools. The earlier layers, we allowed developers to access and search for web enabled smart things and owners to have a straightforward and scalable mechanism to share them. In this endmost layer, [26] we would like to push further the boundaries of the WoT so that from getting close to developers, it also gets closer to end-users and enables them to create straightforward composite applications on top of smart things. The role of the endmost layer is to integrate the services and data offered by things into higher level web, making it even simpler to create applications include things and virtual web services.

V. OPEN PLATFORM IN WEB OF THING

The impenitent internet will confer a wide range of real world services supported by networked smart things setting up complex smart spaces on a worldwide scale. An open platform to provide developers with swift and convenient prototyping tools to plug smart things into the web [6]. This is called the Web of Things Open Platform (WoToP) and this model is many design principles. Firstly, the management of massive amounts of data generated by huge ecosystems of smart objects. Secondly, support for various communication modes to deliver information to clients according to their information consumption needs and most typical communication modes are supported on-demand and event-driven. Thirdly, usage of an information model and standardized protocol based on the representational state transfer (REST) in order to propose a public API accessible to as several clients as possible. Fourthly, finding, configuration and management [7] of miscellaneous ecosystems of smart things, i.e. Things equipped with sets of sensors and actuators as well as processing capabilities that are competent in performing inferences from its own contextual information. Fifthly, the integration process is supported by a set of synchronized smart gateways which implements both IP/TCP stack and particular protocols depending on the embedded devices plugged into it.

VI. DISCOVERY AND SEARCH IN THE WEB OF THINGS

The Web of Things search engines (WoTSE) is librarians of WoT. They explore and gather WoT resources in a specific scope and permit users to "search" on these resources. For succinctness and consistency, we use the term WoTSE for both systems designed specifically for WoT and IoT that can be adapted to WoT.

1. Detect Physical Objects: The WoTSE are commonly used to detect physical objects, which are tagged with passive RFID tags [29] or sensor nodes [30].
2. Sensor Explore: The use of WoTSE [31] for retrieving sensors based on their stable meta-data and contexts, like as the cost and trustworthiness.
3. Discovery Entity with Dynamic State: A WoTSE that discovery for [32] demonstrates physical objects, for instance meeting room, based on their real-time status for instance "empty" derived from their sensor readings.
4. Discovery Actuation Services: The use of WoTSE [33] as a middleware for retrieving services offered by physical objects for instance changing lamp intensity.
5. Retrieving Data Records: The use of WoTSE [34] to retrieve data records compatible to an individual physical object.

VII. OPEN PLATFORM ARCHITECTURE IN WEB OF THING

The open platform architecture [35] in Web of Thing is based on a layered architecture composed of three layers interconnected among them via well-defined interfaces shown in figure 2. In this architecture, every layer contains subsystems and components that fulfill particular aims.

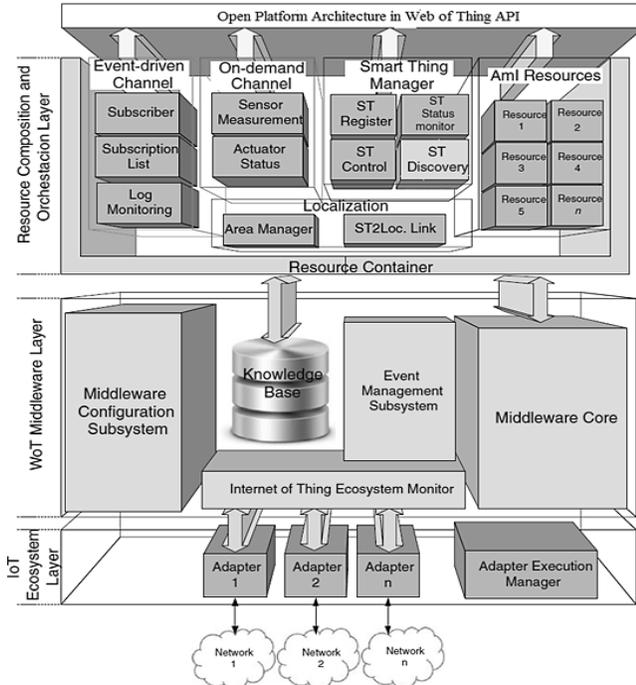


Figure 2. The Architecture of the Web of Things Open Platform

A. The IoT Ecosystem Layer

The IoT [2] ecosystem layer enables the WoToP link to things coexisting in the real-world. These things are usually smart, and they are characterized according to dissimilar natures and objectives within the smart space they belong to. A smart thing consists of one or more embedded devices equipped with sensors and actuators that enable it to play a role in the smart space, coexisting with other smart things to fulfill a given objective. The Internet of Things ecosystem

layer implements a mechanism based on the Plug & Play model. In this procedure, devices of very dissimilar technologies can be plugged to the WoToP. The components implementing drivers for any technology are called adapters, and they are stored in a repository. This procedure is supported by an OSGi framework that permits, scheduling [36] the lifecycle of adapters, i.e. load, unload, run and stop. To integrate the adapters with OSGi, they are implemented as bundles that are stored in an OSGi Bundle Repository (OBR). Whenever a new device is plugged to a WoToP gateway, be expected to bundle is retrieved and loaded automatically.

B. The WoT Middleware Layer

The functionalities of this layer permit designers and developers of WoT services to model, implement and deploy complicated smart spaces and disclose them according to the WoT model. In the IoT ecosystem monitor component objectives at listening data from the adapters deployed on the

IoT ecosystem layer. Principally, it registers every device connected to the WoToP as well as its own context information and events generated by them. The information temporally stored in that cache is periodically dumped into a knowledge base, which handle its persistency. In the event management subsystem was designed to enable WoToP to handle asynchronous communications for those clients that need to consume context information according to particular needs. This subsystem contains differing mechanisms to send notifications or events to the clients [35]. In the middleware configuration subsystem objectives in preparing and configuring the execution environment of the Web of Things middleware. An optimized configuration depends on the features of the device that is going to run the middleware. The standard parameters that can be configured via this subsystem are associated with the location and credentials to access the knowledge base, maximum number of clients that can access to WoToP services at the same time or maximum number of subscriptions that can be established in the subscription table. In the middleware core is to disclose the interfaces of the components, building the WoToP architecture, in order to facilitate the essential communication among them. This kind of communication is based on OSGi services.

C. Resource Composition and Orchestration Layer

As described in the earlier point, the resource composition and orchestration layer is a planned subsystem of the WoToP architecture since this layer enables WoToP to endow enriched RESTful services to handle smart spaces. The resources endowed by this layer are RESTful services which are implemented by components. Those components are deployed in the resource container which facilitates the handling of their lifecycle. The lifecycle of the components deployed in the resource container is planned in four phases running, stopping, destroying, and initializing. The initializing and the destroying phase allocate and free memory for the operation of the component. The running and the stopping phase are principally focused on opening and closing the REST endpoints that facilitates the access to the resources that are proposed by the components.

VIII. THE ARCHITECTURE FOR WOTSE

The query resolution process that we introduced is common in the existing WoTSE plan. Its implementation, transformation depending on the meta-path of every search

engine, but its activities and their arrangement are irreversible. Because, this model activities as standalone modules for building architecture for WoTSE. In figure 3 demonstrate our modular architecture. In these modules our architecture is organized into layers. The two lower layers handle discovery activities and the two upper layers handle search activities. The storage modules for resource collections and indexes link two set of layers. The whole system is kept safe by security, privacy, and trust assessment measures, which are grouped into a vertical layer. The discoverer module detects resource, identify in the meta-path of the search engine, in a certain physical scope. It can also be extended to discover the relationship between objects and resources.

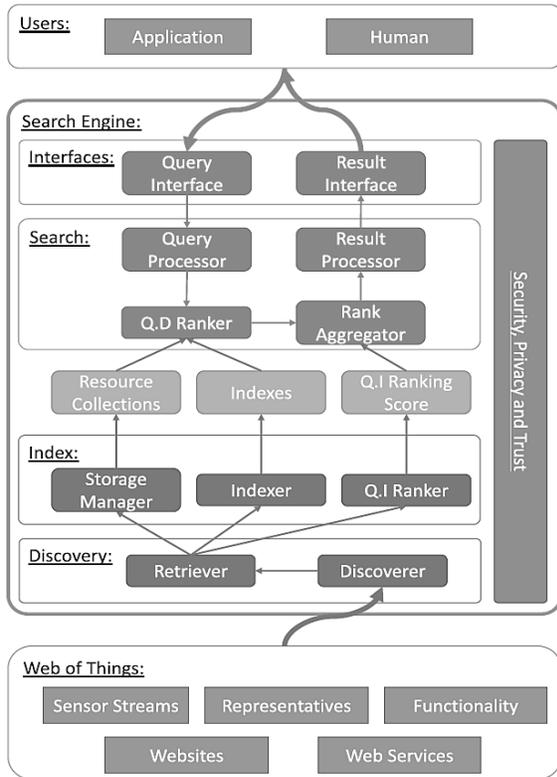


Figure 3. A Modular Architecture for Web of Things Search Engines

The retriever module collects the discovered resources and passes them to the upper layer and index layer stores and indexes resources with its collection manager and indexer modules. This layer also possesses query independent (Q.I) ranker to rank resources according to their inartificial order, independent from user queries. For example, page rank is a form of Q.I Ranking [37]. The depending on the timing between discovery and search activity, a WoTSE can push resources directly to the query resolution process, skipping the index and storage layer. For example, the MAX search engine discovers applicable objects in its vicinity during the query resolution process by broadcasting the query [38]. The set of responding objects, forms its query resource collection, which is blundering after the query is resolved. The search layer carries out the query resolution process. The query processor module metamorphoses raw user queries into the form possible by the system. The query dependent (Q.D) ranker scores discovered query resources with regard to the user query and utilizes the recorded links between resources to quest their corresponding outcome resources. The ranking

aggregator module is accountable for combining dissimilar Q.D and Q.I ranking outcome into a final score for each resource. Finally, the result processor extracts and combined the information from matching outcome resources and produces search outcome. The user interface layer interfaces WoTSE with users. It confers query interface and outcome interface to receive queries and return search outcome. Their forms and implementations vary depending on the meta-path of a WoTSE. It also depends on the type of users targeted by the WoTSE. Factually, a system designed for software applications needs a various interface than a system designed for human users. The modular architecture endues a reference framework assessing the diverse implementation of current

WoTSE. It evaluates the support that each module receives from the current works and how it is commonly implemented.

IX. THE WEB ENABLING DEVICES

In spite of the fact that, web servers are likely to be embedded into more and more devices, we cannot suppose that every smart device will directly offer a RESTful interface. In some instance, it makes sense to hide the platform dependent protocol to access the resources of a particular device, and to disclose them as RESTful service provided by a gateway. The real interactions behind that RESTful service are invisible and often will, contain specialized protocols for the specific implementation sequence of events. The REST elucidates the notion of intermediaries as a core part of the architectural style, and therefore such a design can comfortably be achieved by implementing the RESTful service on intermediaries. By using either proxies or reverse proxies, besides, it is possible to establish like an intermediary from the client or from the server side, effectively introducing a robust pattern for wrapping non-RESTful services in RESTful abstractions. In these circumstances, two solutions are possible first web connectivity directly on the smart things, and the second indirectly via a proxy. The prior work has shown that serving content using web servers for resource constrained devices is presumable [39].

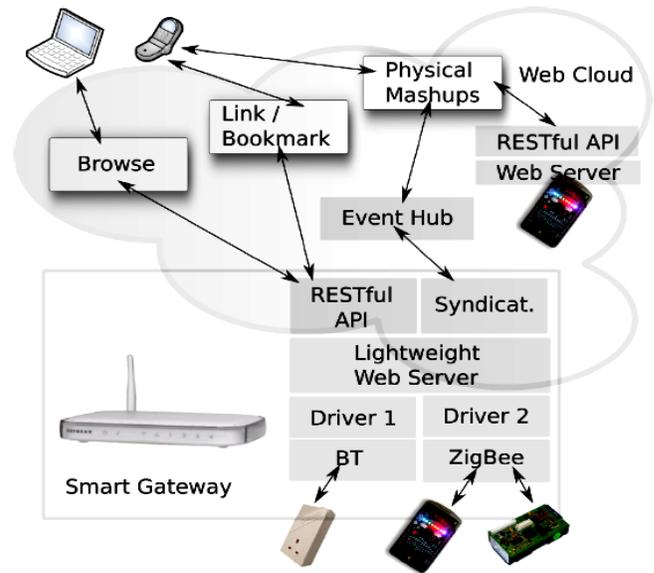


Figure 4. The Web Enabling Devices

In addition, the foreseeable future, most embedded platforms will have native support for TCP/IP connectivity in specific with 6LowPAN [40], consequently, a web server on

most devices is a suitable assumption. This approach is sometimes desirable, as there is no need to translate HTTP requests from web clients into the appropriate protocol for the various devices, and thus devices can be straight integrated and make them RESTful APIs straight accessible on the web. In spite of, when an on-board HTTP server is not feasible or not desirable, web integration takes place using a reverse proxy that bridges devices that are not directly accessible as web resources. We call like as proxy a smart gateway [41] to account for the fact that it is a network component that does more than only data forwarding. A smart gateway is a web server that hides the authentic communication between networked devices, for instance Bluetooth, Zigbee or

RedTacton [42] and the clients via the use of dedicated drivers behind a RESTful service. From the web subscriber viewpoint, the real web enabling process is fully transparent, as interactions are HTTP in both matters.

X. THE WEB OF THINGS SECURITY

Usually, openness and sharing in any ecosystem [10] come always with security and privacy problem, same applies to the WoT. Things' shared resources and data need to be secure against vulnerabilities raised from malicious intervention and inadvertent errors. In the last few decades, many web services based solutions have been proposed to address those privacy and security problem. Despite the fact that, those solutions in most of the time are not relevant to the constrained environment such as the WoT. Besides, it does introduce new dimensions of risk due to its miscellaneous nature. The controlling subscriber identity is a vital process in any application and system [43]. Such control contains authenticating users, identifying them and life cycle of such identities. This process is also applied on the subscriber data inside an ecosystem, since personal information like as identity, credentials, social security, etc., must be protected against unauthorized access. Securing the communication between the various components of the Web of Thing environment is compulsory, in order to preserve data confidentiality and integrity and to prevent a third party from eavesdropping and intercepting information exchanged between the various entities of the system. Conventional access control focuses on the protection of data based on the identity and attributes of the users [44]. Normally the access control is used to protect front-end and back-end data and system resources by adding restrictions on who can access the data, what users can do, which resource they have access to and what operations are allowed to be performed on the data. Ideally, access control prevents unauthorized users from viewing, copying or alteration the data.

XI. USE CASES OF WEB OF THINGS

The WoT extends the IoT in order to enable access and control of physical objects using web standards. The objects are expected to show up logical interfaces via web services, to describe web contents and services using semantic web languages and annotations, and to communicate together via standard protocols in order to provide software interoperability between objects. Such a WoT marketplace should permit developers and industrial companies to distribute their software applications and components, and should endue end-users with software pieces, allowing them to implement various functionalities into their objects in order to perform dissimilar tasks. There are several potential application domains for the Web of Things, stretching across many

industry sectors. In this section, we are attempting to structure these domains and link to further details for the associated use cases of WoT.

A. The WSNS and the Web of Things

It's understandable that the majority of WSN devices weren't designed for the public. Those platforms were intended to be mainly programmed by experts. Despite the fact that, web protocols are heavier than the optimized protocols used in embedded devices, there has been a lot of progress in optimized HTTP libraries that run on constrained devices. Besides, devices are becoming increasingly stronger and many

come with Wi-Fi connectivity on board. The ability to communicate with embedded sensors using standard web protocols makes the collection, storage, and analysis of data from miscellaneous sensors much simpler. In fact, integrating data across several cloud services is much faster thanks to the straightforwardness and the ubiquity of REST APIs.

B. The Web of Things and Marketing

The mobile applications can retrieve data about CPG and FMCG products communicate with them to attach digital content, and share information about them on social networks much rapidly and more comfortably over the web. If every product in the world had its own URL and web API, it would be comfortably in any application to recognize a product and access its data without much integration effort. At EVERYTHING4 firm, we used our Web of Thing platform to connect products to the web and deliver such Marketing 2.0 applications. As an instance, Diageo in Brazil printed distinctive QR codes on its whiskey bottles so that their customers could attach a personalized message to each bottle in that case, it was a video created on the customer's smartphone for Father's Day [45].

C. The Web of Things and Wearable

Integrating wearable and quantified self devices on the web, so that the data is directly accessible by other devices and applications will make it much easier to develop new classes of extensible applications for elder care, health and fitness, or fun and sports.

D. The Web of Things and Industry

Using web standards to interconnect all the elements in a business process, such as the shop-floor machinery, enterprise software, employees in various departments, products, customer, and suppliers, will represent a significant change in how companies do business. Turning all the elements in a factory into easy-to-combine LEGO-like bricks will make it much easier and faster for companies to adapt to changing environments, get their products to market more quickly, optimize their business and manufacturing processes, and so on. When all the actors in those processes are able to automatically decide how best to perform their duty based on real-time data, there's no doubt that the way we design, manufacture, and distribute physical products will be profoundly changed.

E. The Web of Things and Smart Homes

The smart home environment is probably symptomatic of the (too) vast number of standards and protocols that exist for

connecting things to networks. Although all devices in your home should talk to each other, they can't because those protocols are incompatible and you end up with more apps and remote controls than ever before. The Web of Things offers an alternative approach where web languages are the baseline, the minimal API that devices should offer either directly or indirectly through gateways. In our own company "EVERYTHING19" we used the Web of Things approach to connect, at scale, a number of home automation devices from different manufacturers. The Web of Things allows interoperability between devices from different manufacturers and fosters cross-device applications. It also makes it possible

for a larger group of amateurs to buy all sorts of devices, build rapidly their smart home systems, and especially reuse and customize these systems easily for their unique needs and desires.

F. The Web of Things and Smart Logistics

Imagine a web-enabled supply chain that knows in real time the temperature of your strawberries and can send alerts as soon as the conditions change or even regulate automatically the temperature of trucks, ships, and warehouses according to the type of the products being stored and transported all of that information accessible over web APIs. Sharing historical data about devices using web standards will make it much easier for multiple applications to work together across the whole lifecycle of products. This means much lower integration costs and high data integrity across the different systems that will process and handle those products.

G. The Web of Things and Smart Cities

Using web standards in the context of smart cities is particularly interesting because they make it much easier to share sensor data with the public and make it easy for developers to consume real-time data about traffic, pollution, or public transportation in their own urban applications.

XII. BENEFIT OF WEB OF THINGS

In this section, we are discussing the benefit of Web of Things.

A. Convenient to Program Web of Things

The web protocols can easily be used to read and write data from to the devices, and are especially much simpler to use and faster to learn than the complex IoT protocols. In addition, if all devices could offer a Web API, developers could use the same programming model to interact with any of them. Once you get the basic skills needed to build simple web applications, you can rapidly talk to new devices with minimal effort.

B. Supporting Scalability Web of Things

The WoTSE must fit the search activity in local scale naturally, and at the same time, they must also be able to scale up to reach billion devices in the world. Scaling up a centralized search engine is not a preferable solution, because these systems are too far from the physical world, making them insensitive to changes. WoT makes them naturally fit for local search activity, while linking them together provides the coverage to address the upper ends of WoT scale.

C. Open and Extensible Standards Web of Things

The reason web standards have reached such popularity is that they're entirely open and free, so there's virtually zero risk that they would change overnight. They ensure that data can be rapidly and easily moved across systems, hence HTTP and REST is an obvious choice when one wants to offer public access to some data.

D. Rapidly and Effortless to Deploy, Maintain, and Integrate Web of Things

There's no risk that the web will suddenly stop working and require an upgrade. Yet, the limits of what can be done on the web have not ceased to be redefined in a decade, such as the ability to capture images from a camera or share one's location. In contrast, there are always new devices and protocols in the IoT world, and each time one of the many protocols change, all the other pieces of the puzzle that use the device need to be updated.

E. Validating the Discovered Web of Things Content

As real-world information in WoT is provided by exposing electronic tags and sensors that can be breached and forged, a malicious party can inject false information into WoT, which would be distributed by WoT Search Engines. A potential solution for this issue is validating the information received from sensors against past patterns and readings of their neighboring sensors. Another potential solution is building the audit-ability into WoTSE. Ensuring that one would be held accountable for his malicious activities is a powerful preventive mechanism.

F. Slack Coupling between Elements Web of Things

The HTTP is loosely coupled by design because the contract (API specification) between actors on the web is both simple and well defined, which leaves little room for ambiguity. This allows any actor to change and evolve independently from each other (as long as the contract doesn't change). That's why you can still visit a web page that hasn't been updated since the early '90s (we'll skip any comments about its visual design). The ability for devices on the Internet of Things to talk to new devices as they get added without requiring any firmware updates is essential for a global Web of Things.

G. Storing and Disinfect the Collected Data Web of Things

A WoT Search Engine must find a balance between the number of old readings stored for resolving historical queries and building prediction models, and the scale, the resources that it manages, because each set of past measurements duplicates the whole resource collection. As a result, mechanisms for ensuring the scalability of the data storage such as distribution, deployment and purging strategies must be investigated.

XIII. CONCLUSION

The Web of Things is an evolution of the Internet of Things that refers to the adoption of a technology that will enable this sort of global functionality among smart devices. The WoT was inspired by the IoT as in common day-to-day devices are connected to the Web and can communicate via different systems. The Web of Things is about the use of the

web technology for building communication competence for smart objects. The WoT mainly focuses on software standards and frameworks like as HTTP, REST and URIs to create applications and services that integrate and interact with heterogeneity of network devices. So, you could think of the WoT as day-to-day objects being able to access Web services. The Web of Things is not just another vertical IoT technology stack to compete with current platforms. It is intended as integrate horizontal application layer to bridge

together multiple underlying IoT protocols. The key point is that this doesn't involve the reinvention of the means of communication because current standards are used. The WoT is dissimilar because it doesn't care about underlying networking protocols or standards, only about how to weave different isolated systems and devices into a single, web-based ecosystem.

XIV. REFERENCES

- [1] Yusuf Perwej, "An Experiential Study of the Big Data," International Transaction of Electrical and Computer Engineers System (ITECES), USA, ISSN (Print): 2373-1273 ISSN (Online): 2373-1281, Vol. 4, No. 1, page 14-25, March 2017, DOI:10.12691/iteces-4-1-3.
- [2] [2] Yusuf Perwej, Mahmoud Ahmed Abou Ghaly, Bedine Kerim and Hani Ali Mahmoud Harb, "An Extended Review on Internet of Things (IoT) and its Promising Applications", Communications on Applied Electronics (CAE), ISSN : 2394-4714, Foundation of Computer Science FCS, New York, USA, Volume 9, Number 26, Pages 8- 22, February 2019, DOI: 10.5120/cae2019652812
- [3] Yusuf Perwej, Majzoob K. Omer, Osama E. Sheta, Hani Ali Mahmoud Harb, Mohamed S. Adrees, "The Future of Internet of Things (IoT) and Its Empowering Technology", International Journal of Engineering Science and Computing (IJESC), ISSN : 2321- 3361, Volume 9, Issue No.3, Pages 20192- 20203, March 2019
- [4] Nikhat Akhtar, Firoj Parwej, Yusuf Perwej, "A Perusal of Big Data Classification and Hadoop Technology," International Transaction of Electrical and Computer Engineers System (ITECES), USA, ISSN (Print): 2373-1273 ISSN (Online): 2373-1281, Vol. 4, No. 1, page 26-38, May 2017, DOI: 10.12691/iteces-4-1-4.
- [5] Yusuf Perwej , Kashiful Haq, Firoj Parwej, Mumdouh Mirghani Mohamed Hassan , " The Internet of Things (IoT) and its Application Domains" , International Journal of Computer Applications (IJCA) ,USA , ISSN 0975 – 8887, Vol. 182, No.49, Pages 36- 49, April 2019, DOI : 10.5120/ijca2019918763
- [6] Sujith Samuel Mathew, Yacine Atif, Quan Z. Sheng, Zakaria Maamar , " Web of Things: Description, Discovery and Integration " , International Conference on Internet of Things and 4th International Conference on Cyber, Physical and Social Computing, IEEE, Dalian, China, Oct. 2011
- [7] B. C. Blessing Solomon ; Sharmila Mary Aral," WOT based routing protocol for the home diagnostic instruments" , Third International Conference on Science Technology Engineering & Management (ICONSTEM), IEEE, Chennai, India, March 2017
- [8] Simon Duquennoy, Gilles Grimaud, and Jean-Jacques Vandewalle,"The web of things: Interconnecting devices with high usability and performance", In Proceedings of the International Conference on Embedded Software and Systems (ICCESS'09), IEEE, 323–330, 2009
- [9] Heath, T., Bizer, C, "Linked Data: Evolving the Web into a Global Data Space", Morgan & Claypool, ISBN 978-1608454303, February 2011
- [10] Yusuf Perwej, Firoj Parwej, Mumdouh Mirghani Mohamed Hassan, Nikhat Akhtar, "The Internet-of-Things (IoT) Security: A Technological Perspective and Review" , International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT), ISSN : 2456-3307, Volume 5, Issue 1, Pages 462-482, February 2019, DOI: 10.32628/CSEIT195193
- [11] Dominique Guinard, Vlad Trifa, Friedemann Mattern, and Erik Wilde, "From the Internet of Things to the Web of Things: Resource Oriented Architecture and Best Practices", chapter 5, pp. 97–129, Springer, New York Dordrecht Heidelberg London, 2011
- [12] Dominique Guinard, Vlad Trifa, and Erik Wilde, "A resource oriented architecture for the web of things," in Proceedings of Internet of Things 2010 International Conference (IoT 2010), Tokyo, Japan, Nov. 2010
- [13] Yusuf Perwej, "The Next Generation of Wireless Communication Using Li-Fi (Light Fidelity) Technology," Journal of Computer Networks (JCN), USA, ISSN (Print): 2372-4749 ISSN (Online): 2372-4757, Science and Education Publishing Vol. 4, No. 1, page 20-29, June 2017, DOI: 10.12691/jcn-4-1-3
- [14] Yusuf Perwej, Kashiful Haq, U. Jaleel, S. Saxena, "Some drastic improvements found in the analysis of routing protocol for the Bluetooth technology using scatternet" , International Conference on Computing, Communications and Information Technology Applications (CCITA-2010) ,Ubiquitous Computing and Communication Journal (UBICC) Seoul, South Korea, Volume CCITA-2010, Number 5 , pages 86-95, ISSN Online 1992-8424, ISSN Print 1994-4608, 2010
- [15] Amelie Gyrard, "A machine-to-machine architecture to merge semantic sensor measurements," in Proceedings of the 22nd international conference on World Wide Web companion. International World Wide Web Conferences Steering Committee, 2013, pp. 371–376.
- [16] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey", Comput. Netw. 54, 15, 2787–2805, 2010, DOI: 10.1016/j.comnet.2010.05.010
- [17] Alessandro Bassi and Geir Horn," Internet of things in 2020: A roadmap for the future", In Proceedings of the European Commission: Information Society and Media, 2008
- [18] Daniele Miorandi, Sabrina Sicari, Francesco De Pellegrini, and Imrich Chlamtac," Internet of things: Vision, applications and research challenges", Ad Hoc Netw. 10, 7,1497–1516, 2012
- [19] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. 2013. Internet of things (IoT):
- [20] A vision, architectural elements, and future directions. Fut. Gen. Comput. Syst. 29, 7 (2013), 1645–1660.
- [21] Friedemann Mattern and Christian Floerkemeier,"From the Internet of Computers to the Internet of Things", Springer, 242–259, 2010
- [22] Yongrui Qin, Quan Z. Sheng, Nickolas J. G. Falkner, Shahram Dustdar, Hua Wang, Athanasios V. Vasilakos,"When things matter: A survey on data-centric internet of things", J. Netw. Comput. Appl. 64, 137–153, 2016
- [23] [22] Kay Romer, Benedikt Ostermaier, Friedemann Mattern, Michael Fahrmaier, and Wolfgang Kellerer,"Real-time search for real-world entities: A survey", Proc. IEEE 98, 11, 1887–1902, 2010
- [24] Sergei Evdokimov, Benjamin Fabian, Steffen Kunz, and Nina Schoenemann, "Comparison of discovery service architectures for the internet of things", In Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC'10), IEEE, 237–244, 2010
- [25] Yuchao Zhou, Suparna De,Wei Wang, and Klaus Moessner," Search techniques for the web of things : A taxonomy and survey", Sensors 16, 5, 600, 2016
- [26] Li Xiao-Hong," Research and Development of Web of Things System Based on Rest Architecture", Fifth International Conference on Intelligent Systems Design and Engineering Applications, IEEE, Hunan, China, June 2014
- [27] Guinard, D., Trifa, V., Pham, T., Liechti O, "Towards Physical Mishap's in the Web of Things", In Proc. INSS Pittsburgh, PA, USA, 2009
- [28] S. S. Mathew, Y. Atif, Q. Z. Sheng, Z. Maamar, "Web of things: Description discovery and integration", Internet of

- Things (iThings/CPSCoM) 2011 International Conference on and 4th International Conference on Cyber Physical and Social Computing, pp. 9-15, Oct 2011.
- [29] Michelle L Mazurek, J. P Arsenault, Joanna Bresee, Nitin Gupta, Iulia Ion, Christina Johns, Daniel Lee, Yuan Liang, Jenny Olsen, Brandon Salmon, Richard Shay, Kami Vaniea, Lujo Bauer, Lorrie Faith Cranor, Gregory R Ganger, and Michael K Reiter, "Access Control for Home Data Sharing: Attitudes, Needs and Practices", In Proc. of the 28th international conference on Human factors in computing systems (CHI 2010), CHI '10, pages 645-654, New York, NY, USA, ACM, 2010
- [30] Mizuho Komatsuzaki, Koji Tsukada, Itiro Siio, Pertti Verronen, Mika Luimula, and Sakari Piesk, "IteMinder: Finding items in a room using passive RFID tags and an autonomous robot", In Proceedings of the 13th International Conference on Ubiquitous Computing. ACM, 599-600, 2011
- [31] Simon Mayer, Dominique Guinard, and Vlad Trifa, "Searching in a web-based infrastructure for smart things", In Proceedings of the 3rd International Conference on the Internet of Things (IOT'12). IEEE, 119-126, 2012
- [32] Charith Perera, Arkady Zaslavsky, Peter Christen, Michael Compton, and Dimitrios Georgakopoulos, "Contextaware sensor search, selection and ranking model for internet of things middleware", In Proceedings of the IEEE 14th International Conference on Mobile Data Management (MDM'13), Vol. 1. IEEE, 314-322, 2013
- [33] Benedikt Ostermaier, K. Romer, Friedemann Mattern, Michael Fahrmaier, and Wolfgang Kellerer, "A real-time search engine for the web of things", In Proceedings of the 1st International Conference on the Internet of Things (IOT'10). IEEE, 1-8, 2010
- [34] Benoit Christophe, Vincent Verdot, and Vincent Toubiana, "Searching the web of things", in Proceedings of the 5th IEEE International Conference on Semantic Computing (ICSC'11). IEEE, 308-315, 2011
- [35] Sergei Evdokimov, Benjamin Fabian, Steffen Kunz, and Nina Schoenemann, "Comparison of discovery service architectures for the internet of things", In Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC'10). IEEE, 237-244, 2010
- [36] YangQun Li, "An Integrated Platform for the Internet of Things Based on an Open Source Ecosystem", Future Internet, vol. 10, pp. 105, 2018
- [37] Yusuf Perwej, "The Ambient Scrutinize of Scheduling Algorithms in Big Data Territory", International Journal of Advanced Research (IJAR), ISSN 2320-5407, Volume 6, Issue 3, PP 241-258, March 2018, DOI: 10.21474/IJAR01/6672
- [38] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd, "The pagerank citation ranking: Bringing order to the Web", Stanford InfoLab, 1999
- [39] Kok-Kiong Yap, Vikram Srinivasan, and Mehul Motani, "MAX: Human-centric search of the physical world", In Proceedings of the 3rd International Conference on Embedded Networked Sensor Systems. ACM, 166-179, 2005
- [40] Duquenois S, Grimaud G, Vandewalle J, "The Web of Things: interconnecting devices with high usability and performance," Proceedings of the 6th IEEE International Conference on Embedded Software and Systems (ICCESS'09). HangZhou, Zhejiang, China, 2009
- [41] Hui J, Culler D, "Extending IP to low-power, wireless personal area networks", IEEE Inter-net Comput 12:37-45, 2008
- [42] Trifa V, Wieland S, Guinard D, Bohnert TM, "Design and Implementation of a Gateway for Web-based Interaction and Management of Embedded Devices.", Proceedings of the 2nd International Workshop on Sensor Network Engineering (IWSNE 09). Marina del Rey, CA, USA, 2009
- [43] Yusuf Perwej, "A Literature Review of the Human Body as a Communication Medium using RedTacton" Communications on Applied Electronics (CAE), ISSN : 2394-4714, Foundation of Computer Science FCS, New York, USA, Volume 9, No.4, Pages 7 - 17, April 2016, DOI: 10.5120/cae2016652161
- [44] X Yang, Z Li, Z Geng, H Zhang, "A Multi-layer Security Model for Internet of Things" in Internet of Things, Berlin Heidelberg:Springer, pp. 388-393, 2012
- [45] Ezedine Barka, Sujith Samuel Mathew, Yacine Atif, "Securing the web of things with role-based access control", International Conference on Codes Cryptology and Information Security, pp. 14-26, 2015
- [46] <http://adage.com/article/global-news/diageo-personalizes-whiskey-videos-gift-givers/238015/>.