# Interoperability in IoT

Regel Gonzalez-Usach , Diana C. Yacchirema , Matilde Julian-Segui, Carlos Enrique Palau

regonus@doctor.upv.es, diayac1@doctor.upv.es majuse@upv.es, cpalau@dcom.upv.es

[a]*Universitàt Politècnica de Valencia,* [b]*Escuela Politécnica Nacional*

Interoperability refers to the ability of IoT systems and components to communicate and share information among them. This crucial feature is the key to unlock all of the IoT paradigm´s potential, including immense technological, economic and social benefits. Interoperability is currently a major challenge in IoT, mainly due to the lack of reference standards and the vast heterogeneity of IoT systems. In this chapter, we analyse the critical importance of interoperability, its different types, the problems encountered while trying to achieve it, diverse use cases and prospective interoperability solutions. Given that it is a complex concept that involves multiple aspects and elements of IoT, for a deeper insight, interoperability is analysed across different levels of IoT systems: Device, Network and Middleware. Additionally, in this chapter, interoperability is re-examined from a global approach, considering it among platforms. Finally, some conclusions regarding IoT interoperability and its future are drawn.

## 1. Introduction

Interoperability can be defined as the ability of different technology systems, system components or software applications to establish communication between them, exchange data, and interpret properly the received information for its use (ETSI 2013). This property applies to interactions within a system, regarding which comprises the internal communication of its different components, but also to the interaction between two or more systems.

There is a strong link between interoperability and IoT, as there is probably no other technology area in which interoperability becomes especially critical and relevant as in the case of IoT (World Economic Forum 2015). Interoperability is the key that allows any set of devices to exchange information and work together in concert, acting as an actual IoT system. For example, without interoperability lights would not respond to remote switches, sensors could not be read by smartphones, and devices in general would be unable to connect to accessible networks. Moreover, according to a study carried out by the McKinsey Global Institute in 2015 (Mckinsey Global Institute 2015), without interoperability, at least 40% of the potential benefits of IoT cannot be achieved. This is evident if we consider that a transparent integration and interconnection of different IoT systems and system components would critically simplify their implementation, maximize performance and facilitate their interconnection with other systems. This system's interconnection propitiates them to share relevant data and to establish significant synergies, improving the quality of the information, the quality of service and the experience provided to the user. These advantages can be better understood through some examples of interoperated IoT systems.

Specifically, consider an application of a bus company that calculates its optimal route. This application could benefit from interoperability with other transportation services, as it could, for instance, calculate links with trains using the real time information that they provide. This application could also benefit from the interoperation with the traffic monitoring service of the city, capable of indicating the less congested routes. Thus, the service provided by the bus application would be more precise, complete and useful for the user.

Let us also consider some examples of IoT systems in the e-Health domain. In this area, interoperability among sensors and medical devices permit the remote monitoring of different

bodily vital signals such as heart rate, blood pressure or breath rate. This can be even done without interfering with the quotidian life of the patient if those sensors are wearable. Then, it is possible for an IoT health system to detect any abnormality of vital signals remotely at any moment, and automatically alert the pertinent health services and caregivers.

To achieve a high degree of interoperability in an IoT system is therefore desirable, but regrettably it is still one of the most difficult and important challenges to solve in IoT. As a matter of fact, currently the different IoT systems are typically unable to correctly communicate with each other or to interoperate in general (Diallo et al. 2011). The main cause of this is directly related to the highly heterogeneous nature within and among IoT systems. The Internet of things covers a wide range of devices, protocols, technologies, networks, middleware, applications, systems and data that present a vast diversity. In this sense, the existence of a global reference standard for IoT would be helpful, as it would notably facilitate interoperability, by giving rules and certain homogeneity to this heterogeneous universe. However, currently we lack such a standard, posing a significant problem when designing new IoT systems (Ganzha and Paprzycki 2016). The heterogeneity of the underlying technologies can prevent the interoperability of smart objects that could be used to adapt a particular IoT environment to specific needs.

This chapter covers all these topics, starting with the explanation of the concept of interoperability in IoT, the different types of interoperability that exist, the problems that arise regarding their achievement, and also the considerable benefits that interoperability brings. As far as standards can simplify and ease interoperability, in what follows, we provide an overview of the existing standards, although so far none of them has been established as a de facto one. Interoperability is a complex concept that encompasses many aspects from all levels of IoT

systems. For the sake of clarity and to facilitate a deeper understanding of the concept of interoperability, in this chapter we will study it across the different layers of IoT systems: at the Device, Network and Middleware levels. Within each layer, the concept of interoperability is analysed alongside with the problems and obstacles found and the possible solutions at that specific level. Next, interoperability in IoT is analysed from a global perspective considering also interoperability between different IoT platforms. In the following section some representative IoT interoperability use cases are explained to illustrate its role in IoT systems. Finally, in the last section of the chapter some conclusions are drawn regarding the analysis of interoperability in IoT.

## 2. Motivation and State of the Art

In this section, the motivation for the study of interoperability in IoT is presented alongside with its definition, types and current standards.

### *2.1 Importance of Interoperability in IoT (Motivation)*

The connection of things or smart objects to the Internet generates unexpected insights and significant business value that will be positive for the citizenry and the industrial sector (Aloi et al. 2017a). However, as it has been mentioned before, according to (Mckinsey Global Institute 2015) without proper interoperability in IoT systems, it will not be possible to achieve on average 40% of the potential economic benefits of IoT.

Insufficient interoperability is the main obstacle for the development of IoT and its adoption by the market (Telecommunication Standarization Sector of ITU 2014). It is also the cause of major technological and business issues and setbacks (Aloi et al. 2017b). A typical issue is that some smart objects may not be compatible with certain IoT platforms. In addition, it causes an

increased difficulty in the development of IoT applications that exploit several platforms in diverse domains. This situation produces sluggishness in the large-scale IoT technology introduction. Some of its main drawbacks are frustration and discouragement when trying to adopt IoT technologies, increased costs, bad user experiences and the non-reusability of technical solutions.

Another important issue is the existence of isolated systems due to the general lack of interoperability among platforms. The IoT market is a highly fragmented ecosystem in which several vertical systems coexist. Due to the absence of interoperability among them, these systems stand as isolated vertical silos of information that are unable to inter-operate, collaborate or share specific information (Soursos et al. 2016) These vertical systems cannot benefit from synergies and opportunities that arise in a fast-paced business landscape as a fruit of system interoperability. This has significant market drawbacks, and affects to the quality of services offered to the user.

The envisioned future of IoT forecasts that all devices with communication and sensing capabilities will be able to interconnect and interact in a transparent way (Atzori, Iera, and Morabito 2010) (Gubbi et al. 2013a). According to this vision, interoperability plays a major role, as this seamless integration and interconnection of devices requires a very high degree of interoperation.

### *2.2 Definition of interoperability*

Different definitions of Interoperability have been established up to the date. One of the most relevant is the one by the Institute of Electrical and Electronics Engineers (IEEE), that defines interoperability as *"the ability of two or more systems or components to exchange data and use information"* (The Institute of Electrical and Electronics Engineers 1990). The Technical

Committee of the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) propose a more detailed definition of interoperability that includes user interaction: *"Interoperability is the capability to communicate, execute programs or transfer data among various functional units in a manner that requires the user to have little or no knowledge of the unique characteristics of those units"* (International Organization for Standarization 2003). There are also other definitions of interoperability that are adapted to specific use cases. For example, the Department of Defence of USA defined interoperability as *"the condition achieved among communications-electronics systems or items of communications-electronics systems equipment when information or services can be exchanged directly and satisfactorily between them and/or their users"* (Department of Defense of United States of America 2008). Although many definitions exist, they all agree on the same basic principles and highlight the necessary and sufficient conditions to achieve interoperability: "*Information exchange and usability of information"* (Diallo et al. 2011). Therefore, interoperability in the IoT ecosystem can be understood as the ability to exchange data and use the information across systems, applications, or system components.

### *2.3 Types of interoperability*

There are different types of interoperability. The main sorts are:

- **Technical interoperability** refers to the ability of systems, system components or applications to establish communication and share messages, without necessarily understanding their content. Hence, it does not imply awareness of data format and meaning (Molina 2014). It typically requires the existence of network connectivity. Technical interoperability is strongly related with the elements that enable a machine-to-

machine (M2M) communication (e.g. required protocols, hardware and software) (Hans van der Veer 2008).

- **Syntactic interoperability** refers to the ability of systems of correctly interpreting the message structure of exchanged information and, thus, being capable to read its content, although they may not be aware of the meaning of this information (Gubbi et al. 2013b). An example of syntactic interoperability is a smart city system that receives information from a data center and is capable to properly recognise its specific data format (e.g. CSV) and thus correctly extract the data from the message (e.g. a set of values). Nevertheless, it may not be aware of what this data represents (for example the values could be temperatures), thus being unable to use the data within the correct context. Therefore, syntactic interoperability relies on data formats, as the messages exchanged among systems require a common data representation for the correct interpretation of the data structure and content. The use of standardized data formats avoids ambiguity in the interpretation of data. Examples of data formats are standards such as, XML, JSON or CSV, which provide a high-level syntax.
- **Semantic interoperability:** at this level, the systems are capable of interpreting the content and the meaning of the information exchanged. Ontologies, semantic technologies and knowledge management systems are means to facilitate semantic interoperability. In this regard, the Sensor Web Enablement (SWE) initiative of the Open Geospatial Consortium (OGC) defines data encodings and Web services to enable interoperability by SensorML and O&M ontologies (Ganzha and Paprzycki 2016). As an example, semantic interoperability allows a smart city system that has correctly extracted the data received from another system, to understand the meaning and context of the

information contained in this data. Then, this system can be aware that the set of values extracted actually represent temperatures of a city area. Thus, the system becomes capable of using this information in the proper context.

### *2.4 Standards for IoT*

An effective approach to tackle interoperability is the use of standards, reference architecture models and the application of best practices in IoT deployments. In this sense, the use of a global standard in IoT can potentially solve the interoperability problem and enable compatibility among IoT systems.

Next, the most relevant standards for IoT are mentioned:

- OneM2M: is the global standards initiative for IoT and Machine-to-Machine (M2M) communications. OneM2M is working on a service layer that includes technical requirements, Application Programming Interface (API) specifications, data semantics, and security solutions to enable IoT interoperability (Alaya et al. 2014).
- AllJoyn: is an open source framework driven by the AllSeen Alliance that allows devices to communicate with other machines regardless of the communication technology or manufacturer thanks to the use of a common protocol (Allseen Alliance 2017).
- IoTivity: is an initiative from the Open Connectivity Foundation. It provides an open source framework that enables seamless interconnection and management of wired and wireless devices, independently from the device manufacturer or the operating system used (Linux Foundation 2017).
- ARM: is an IoT Reference Architectural Model proposed by the European research project IoT-A. This standardization initiative consists of a set of building blocks that

represent basic concepts and components that enable the creation of interoperable IoT systems (Krco, Pokric, and Carrez 2014).

Furthermore, other working groups have also provided their own standardization initiatives, as is the case of the organizations ITU, ETSI, OpenIoT and IPSO Alliance. However, despite all these efforts, nowadays no global reference standard has been adopted for IoT. Moreover, none of the current standards is expected to become a referent in the medium-term future.

In this multi-standard context, the high fragmentation and development of vertical IoT systems is increasing, as systems operating on different standards are unable to communicate with each other. This produces a Babel's Tower-like situation that prevents interoperability among them.

## 3. Interoperability layered-approach

IoT interoperability is a complex concept that encompasses many different aspects and elements from each layer of an IoT system. Instead of only providing a conventional global and holistic approach regarding interoperability, this section offers an analysis of interoperability across specific layers or levels of IoT systems: Device, Network and Middleware. This perspective offers a better comprehension of the IoT interoperability concept and its associated challenges.

The Device layer represents the set of sensors, actuators and smart objects that compose the lowest level of an IoT system. The Network layer is the level of networking and communication that encompasses networks and communication protocols. Finally, the Middleware layer represents the software infrastructure of an IT system that enables communication among the different components of that IT system. Together, all these layers constitute the main part of a standard IoT system.

Next, these three layers are analysed, noticing the existing problems regarding interoperability and potential solutions.

## 3.1 Device Layer

The Device Layer in the context of IoT, refers to the collection of sensing devices or actuators connected to an IoT system. These are commonly known as the ‘Things’ in the Internet of Things. This layer is composed by smart objects (sensors, actuators and virtual devices) that are connected to a network and quite often have limited CPU, energy resources and memory. Devices that present these limitations are called "constrained devices".

A classification of these constrained devices can be found in (Bormann 2014). They are sorted out into three different categories:

- Class 0 devices, which are very constrained in memory and processing capabilities. Thus, they need the help of other devices, such as gateways, to communicate with other Internet nodes.
- Class 1 devices, which can communicate with other Internet nodes making use of a protocol stack specifically designed for constrained devices.
- Class 2 devices, which are the less constrained ones and, thus, support most of the protocol stack implemented in other Internet nodes. These devices can also make use of the protocols defined for class 1 devices in order to reduce their energy and bandwidth consumption, as well as the use of the resources needed for the applications.

### *3.1.1 Problems regarding interoperability*

Interoperability at the device level refers to the ability of heterogeneous IoT devices to interact with other devices or other elements of an IoT system. It also means that they could be integrated into an IoT platform.

Interoperability at this layer is mainly hindered by the large heterogeneity of devices regarding the protocols they use, their communication technologies, hardware specifications, providers,

etc. Besides, the IoT device software is never platform-independent, since companies produce proprietary and closed solutions motivated by economic reasons. These facts make interoperability much harder to achieve.

### *3.1.2 Communication Models at the Device layer*

In order to analyze interoperability at the device layer, it is necessary to understand first the communication models that smart devices employ. Smart objects introduce a new communication paradigm and interoperability issues that cannot be solved by the existing patterns for traditional Internet architectures. For this reason, new communication models for smart objects have been recently defined (H. Tschofenig, J. Arkko, D. Thaler 2015): device-to-device (D2D), device-to-gateway (D2G) and device-to-cloud (D2C).

*Device-to-Device Communication Pattern (D2D)*

D2D refers to direct communication between two devices. This includes M2M (e.g. direct communication between smartphones). Many specific communication aspects of the two devices need to be specifically defined and addressed in order to make D2D communication possible. Such aspects comprise for instance a common protocol stack and protocol design (e.g. supported physical layer, network technology, IP addressing, architecture, data rate constraints, transport protocol, and other aspects).

This type of communication requires a very specific solution design for each case and usually is only possible between devices from the same vendor that support a common network technology.

*Device-to-Cloud Communication Pattern (D2C)*

In this case, the device sends directly information to a cloud platform, application or service. The cloud service provider is in charge of guaranteeing the interoperability with a wide range of devices.

*Device-to-Gateway Communication (D2G)*

This model refers to the communication between a smart device and a gateway. A gateway is a node that links two networks that employ different protocols. Whereas the function of a bridge is to conjoin two similar types of networks, a gateway connects two dissimilar networks. The main functionality of the gateway is the protocol conversion, as it converts the protocols from an entering communication flow, before transmitting the flow outside the gateway. This conversion is done at all levels (device, network, physical and application), allowing the interoperability among the endpoints of a communication process. For example, at the network level a gateway can convert between the IPv4 and IPv6 protocols; at the physical level, between 802.15.4 and 802.11, and at the application level between MQTT and CoAP.

The D2G communication model is generally employed to allow long-distance communication for constrained devices. This communication pattern is also implemented in IoT systems to enable remote interactions with smart devices in real time. In that case, the gateway is permanently connected to the Internet.

Another case study of D2G is the use of a mobile gateway (e.g. a smartphone), where connectivity between the device and the Internet may be intermittent.

### *3.1.3 Interoperability solutions in the Device Layer*

Interoperability solutions for heterogeneous devices are typically gateway-oriented. This kind of approach allows the establishment of D2D and D2C communication when it is not possible

directly due to technical limitations. Regarding D2D, a gateway allows communication between devices that are not capable to communicate among them directly, which is the most frequent case. Both devices must be connected to the gateway. For D2C communications, a gateway becomes necessary when dealing with constrained devices, as they do not have enough resources to manage a protocol stack for the interconnection with the cloud. In this case, the gateway acts also as an intermediary element.

*Gateway-oriented interoperability solutions*

A gateway is a key element for providing interoperability in many IoT systems. It allows for interoperability among heterogeneous devices and between heterogeneous networks at many levels (i.e. device, application and physical) (Yacchirema, Palau, and Esteve 2016).

Gateways for IoT, also called smart gateways, offer additional functionality to traditional gateways. A smart gateway adds a data processing stage before the information is sent to its destination. Another feature of a smart gateway is that it usually provides low-power connectivity through typical IoT technologies such as ZigBee or Bluetooth, in addition to the regular gateway connectivity (through WiFi or Ethernet). As many smart devices only use low-power technologies this gateway feature enables their interoperation.

Gateways provide technical interoperability, allowing basic communication between the two endpoints (i.e. the smart device and the external destination endpoint). The data processing stage of the smart gateway facilitates syntactic and semantic interoperability. On one hand, the gateway can process the information received from a sensor, and convert it into the appropriate syntactic data format for the receiver. On the other hand, semantic metadata can be added to the sensor data using a data aggregation functionality to support semantic interoperability. This

additional semantic metadata consists of information about the meaning and context of the data, and can be interpreted using the proper ontology.

An example of smart gateways are the software-defined IoT gateways. These can be implemented in any hardware featuring the necessary minimal requirements in terms of processing power (for instance, they can be installed in a low-power processor, such as a Raspberry). Some relevant examples of these gateways are:

- **Eclipse Kura:** is an open-source Eclipse project that provides a platform for building IoT gateways. Kura offers a service API and is capable of handling events. It enables the remote management the IoT gateways. As it is Java-based, Kura is platform independent (it runs on any platform). As a disadvantage, it cannot be installed in devices with limited memory and processing power because Java requires considerable resources.
- **OneM2M middle node:** The middle node of a OneM2M platform acts as a smart IoT gateway. OneM2M middle node has a common service layer that enables interoperability and data exchange. This is done through the functions of device discovery, connectivity management and establishment of secure connections. This architecture can be easily extended by developing specific modules for new devices and protocols.
- **Mihini:** is another Eclipse open source project that allows device interoperability and the development of M2M applications. This framework permits to build lightweight and portable smart IoT gateways, which require few processing power.
- **AGILE IoT:** is a modular hardware and software gateway specifically conceived for the Internet of Things. It features support for protocol interoperability, device management, device data and IoT apps.

- **Intel IoT Gateway:** Intel offers a proprietary IoT gateway and a platform that allows its remote management. In addition to the software-defined gateway, Intel also provides the physical device. Intel IoT Gateway can connect both legacy industrial devices and modern smart objects to an IoT system.
- **Bodycloud:** A smart mobile gateway that can be installed on a smartphone. It was designed for medical purposes, to allow for the monitoring of a set of medical sensors on the body of a patient that carries the smartphone.

### 3.2 Network Layer

The network level of an IoT deployment refers to the set of protocols, systems, and devices that work on the Network layer of the OSI protocol stack (Whitmore, Agarwal, and Da Xu 2015), This layer contains hardware elements such as switches, firewalls, routers and bridges.

In some aspects, networks in IoT environments are significantly different from traditional networks. Networks to which smart objects connect have typically constrained capabilities such as unreliable channels, a narrow and erratic bandwidth, and a highly changing topology (Bormann 2014). Other distinctive feature is that these networks typically support technologies and protocols for constrained devices. Most of them are wireless protocols for low rate transmission and focused on energy saving. Examples of those technologies are ZigBee, RFID or LoRa.

This section explains how to achieve interoperability between networks or parts of a network that belong to an IoT system. At the network level, only technical interoperability is considered, given that semantic and syntactic aspects are transparent to this layer. Specific challenges in the network level include the seamless mobility of smart objects through different access networks (roaming) and secure connectivity. Also, other issues must be taken into account such as the

difficulties inherent to the operation in highly constrained environments and the use of a wide range of heterogeneous protocols (e.g. 6LowPAN, RPL, LoRa, SIGFox, etc).

### *3.2.1 Interoperability solutions for Network Layer*

The interoperability solutions introduced in this section are based on software-defined paradigms. They rely mainly on two approaches:

- Software Defined Radio: this approach can provide interoperability on the access to network, allowing seamless roaming among areas covered by SDRs, as well as a dynamic network topology.
- Software Defined Networks: this approach allows for a transparent and seamless interconnection of dissimilar IoT networks that can be on different locations.

*Software Defined Radio*

An obvious interoperability solution for the access to network is a gateway, which also allows performing protocol conversion and enabling device interoperability. Another very remarkable interoperability solution for network access is a Software Defined Radio (SDR), which is capable to solve very arduous interconnection problems that are inherent to IoT environments.

At present time, there is very limited spectrum available for IoT wireless networks. Thus, an effective use of the available spectrum is key to enable the connectivity of numerous wireless heterogeneous smart objects. As an additional problem, IoT environments suffer a high level of wireless interferences, so communication with smart objects should be seamless and highly reliable to overcome this effect.

A SDR represents an interoperability solution to overcome these problems and facilitate the access to network to multiple wireless sensors. These sensors can be working on very different radio frequencies and using very different protocols/standards, or even be non-standard. A

Software Defined Radio is a radio that has digitalized components and provides software control over radio system functionalities, such as the modulation type, the frequency and the transmit power.

As a result, SDR technology can bridge different wireless devices across different frequencies and protocols. With this approach, even non-standard devices that use a radio access network are able to interoperate with the rest of the IoT system. SDR can enhance interoperability and also set up the infrastructure for future devices so that they are not restricted by bandwidth or frequency.

*Software Defined Network solution for Network Interoperability*

Software Defined Network (SDN) solutions allow to interconnect different networks, enabling interoperability among them. Those networks can be on different locations, from different vendors, or with a different configuration or topology.

Software Defined Networking is a set of network technologies that make the network functionality abstracted, virtualized and controlled via software. As a consequence, it can be automatized, and also accessed and controlled by a network administrator. SDN is based on these main technologies: network virtualization , functional separation and automation through programmability (Kreutz et al. 2015).

In a Software Defined Network, the network functionality is decoupled in two planes: the data plane that comprises the forwarding functions, and the control plane that represents the network control. The data plane is related with data transmission and transport, and due to this separation the network routing elements (e.g. switches, routers) become mere forwarders of data packets. On the other hand, regarding the control plane, the whole logic of routing, algorithms and other services previously provided by firewalls, middleboxes, IPS, etc. are transferred to a single point

of control and decision-making called the controller.

## 3.3 Middleware Layer

The middleware is a software layer between applications and the communication network. It allows an application to abstract from the intricacies of how to send data to a service of another application. A middleware offers functionalities for this aim, such as to find and establish a connection to a service, negotiate the optimal wire and transport protocols, access applications data structures and encode the necessary data in a format appropriate for the selected protocol. (Perera et al. 2014).

### *3.3.1 Interoperability Challenges in Middleware*

IoT interoperability represents a huge challenge for middleware approaches, given that applications and a vast range of heterogeneous smart devices are expected to collaborate by exchanging information. This entails high complexity in the middleware design and development, as it must be capable of supporting interoperability covering a wide range of current devices (Bandyopadhyay and Sen 2011). Furthermore, it has also to tackle with the inclusion of potential new kinds of devices (Moumena, Mohamed, and Mohamed 2012).

The three different types of interoperability should be considered in relation with the middleware:

- Technical interoperability: to allow it, the middleware should be able to exchange information across different networks, and may use different communication technologies.
- Syntactic interoperation: to achieve it, the middleware should allow for the heterogeneous formatting and encoding structures of any exchanged information.

- Semantic interoperability: this should be permitted in the exchanges between devices and applications and services in IoT, in order to enable a common interpretation of the meaning of the exchanged information. Some middleware solutions have semantic support and use a specific ontology (e.g. Open-IoT and SOFIA2), while other do not (e.g. FIWARE).

Another challenge to be consider by IoT middleware would be to perform an abstraction of devices, data streams and interfaces to facilitate interoperability.

Finally, an IoT middleware should be continuously supported by developers to guarantee an up-to-date interoperability (Moumena et al. 2012).

### *3.3.2 Interoperability Solutions at Middleware level*

The development of IoT middleware is an active area of scientific and industrial research, and a considerable number of interesting solutions have been developed so far (Bandyopadhyay et al. 2011). Several architectures have been proposed for interoperability in IoT, such as ARM, FIWARE, OneM2M, OpenIoT, SOFIA or UniversAAL. Some of them have been implemented, thus constituting functional IoT platforms (e.g. FIWARE or OneM2M). An IoT platform is defined as the infrastructure and middleware that allow end users to successfully interact with sensors and actuators (Mineraud et al. 2016). Therefore, a platform is a middleware solution that allows applications to seamlessly interact with the device layer, thus enabling interoperability. That means to enable the retrieval of data from sensors as well as issuing orders to actuators.

Some of the existing IoT platforms, such as FIWARE, OneM2M and OpenIoT, are open-source, whereas others, like SOFIA2 and SORACOM, are proprietary. A specific group of these proprietary platforms are cloud-centric, which means that they are hosted in the cloud. They offer a set of services that include cloud storage as a Platform-as-a-Service (PaaS) on the Cloud,

instead of a deployable self-hosted solution. Examples of cloud-centric platforms are the cloud platform solution for IoT offered by AWS, called AWS IoT, and Microsoft Azure. Next, we will describe some of the most relevant IoT platforms that represent middleware solutions for interoperability: FIWARE, OneM2M, UniversAAL, SOFIA2 and OpenIoT.

FIWARE is an open platform supported by the European Commission and which provides diverse middleware services for distributed applications and a support framework for the Internet of Things. FIWARE provides a set of public APIs for the development of applications in multiple sectors (FIWARE 2017).

The foundation of FIWARE architecture are the General Enablers (GE), which provide general-purpose functions. FIWARE provides public specifications of the GE APIs and reference open-source implementations of each GE. Additionally, FIWARE offers domain-specific enablers that provide useful functionalities for specific sectors.

The main GE is the Context Broker, which receives data from the context producers and makes it available for the context consumers. Both context producers and context consumers communicate with the Context Broker through NGSI (Next Generation Service Interface). The main purpose of the Context Broker is to make the context consumers independent from the context producers. The context consumers can obtain the data from the Context Broker on demand or subscribe to the information on which they have taken an interest. The reference implementation of the Context Broker is called Orion.

Other GEs provide additional functionalities to the platform. For instance, the CEP (Complex Event Processing) provides real-time data analysis and sends notifications when certain situations are identified, while the Big Data Analysis GE deploys the means for the analysis of both aggregated and stream data on a Cloud Computing environment.

OneM2M provides a standard for Machine-to-Machine (M2M) interoperability, which refers to the communication between devices. Under the OneM2M functional architecture, several types of nodes are defined that can connect and communicate among them at a global scale. Every node may be composed of three kinds of logical entities, namely an Application Entity (AE), which represents a M2M logical application, a so-called Common Service Entity (CSE), which contains a set of common functions of the oneM2M architecture, and a Network Service Entity (NSE), which provides access to the underlying network infrastructure. The functional architecture consists of two domains: the Field Domain and the Infrastructure Domain, which are composed of different nodes. The Field Domain is made up of Application-Dedicated Nodes (ADNs), Application Service Nodes (ASNs) and Middle Nodes (MNs), which can be embodied as physical sensors or actuators, M2M devices, and M2M gateways, respectively. The Infrastructure Domain includes an Infrastructure Node (IN), which physically corresponds to the M2M server. Regarding the nodes, each of them consists of at least either a CSE or an AE. Depending on the type of node where the CSE is incorporated, this entity can be classified as:

- IN-CSE, if it is incorporated into an Infrastructure Node.
- MN-CSE, which are the CSEs incorporated into Middle Nodes.
- ASN-CSE, if the CSE is incorporated into an Application Service Node.

Several reference points defined within the Functional Architecture are used for the communications among OneM2M entities, such as

- Mca, for communication between AEs and CSEs.
- Mcn, for communication between CSEs and NSEs.
- Mcc, for communication between same domain CSEs.

- Mcc', for communication between different domain CSEs.

OpenIoT is an open source IoT platform intended to provide semantic interoperability of IoT services. This platform is based on the use of Cloud Computing in order to allow the composition of on-demand IoT services, which can include data from multiple sensors. OpenIoT aims to provide semantic interoperability through ontologies, semantic models and semantic annotations (OPENIoT 2017).

The OpenIoT infrastructure aims to supply the means for gathering and processing data from any physical or virtual sensors. The data can be annotated in a semantic way and as per the W3C Semantic Sensor Networks (SSN) specifications and can be directed towards a Cloud Computing facility. Finally, the data can be visualized making use of suitable mashups (graphs, maps, charts, etc.).

SOFIA2 is a proprietary platform developed by Indra Company and is based on the SOFIA architecture. SOFIA (Smart Objects For Intelligent Applications) was a European research project that created a semantic interoperability platform. SOFIA2 permits the interoperability of several systems and devices in order to make real information available for IoT intelligent applications. Its goal is to achieve semantic interoperability among different applications in order to allow for the creation of multi-domain services (Sofia2 2017). The core of the platform is the Semantic Information Broker, which receives, processes and stores all the information and provides interoperability. To provide semantic interoperability, all the concepts defined in the platform are represented through the use of ontologies.

UniversAAL is a semantic and distributed open-source platform designed for the development of integrated Ambient Assisted Living, e-Health and AHA applications. In UniversAAL, every component, including the services and the real world, is semantically

annotated and represented in terms of ontologies. Every interaction is also modelled in a semantic way, making use of the RDF/Turtle format (Hanke et al. 2011).

## 4. Global Interoperability

Many architectures for interoperability have been proposed and implemented at the moment. Some IoT platforms such as FIWARE, OpenIoT, OneM2M or SOFIA2 can be considered as global interoperability solutions in the sense that the IoT system that they represent provides inner interoperability across its different levels.

Regarding the concept of global interoperability, it must be noted that IoT platforms provide intra-platform interoperability but they do not support interoperability regarding external IoT systems and platforms. Each platform uses its own architecture and it represents a vertical silo of isolated information, as it is not directly accessible by other IoT systems (Jacoby et al. 2016). For example, from a semantic perspective, platforms use different ontologies and semantic structures, so that the meaning of the information from one platform cannot be interpreted by another platform. Although semantic translations among platforms are possible, these tasks are usually complex *ad hoc* solutions (i.e. they cannot be generalized). Therefore, for the achievement of a more global, inter-platform interoperability, horizontal solutions for integrating those vertical silos must be provided (Jacoby et al. 2016).

An obvious solution for this problem would be the general acceptance of a common reference standard for IoT. This would facilitate interoperability at all levels, including the inter-platform case. Though, as it was mentioned before, the current multi-standard situation in IoT makes this approach very unlikely. No common de facto global standard has been foreseen for the middle-

term future (Tan, L., & Wang 2010). To overcome this situation, many partial and specific solutions have been developed. These solutions for interoperation among different IoT systems or platforms apply only to the device or data level and in an incomplete, non-transparent and non-seamless way (World Economic Forum 2015) . Although interoperability among platforms is a major concern in many application domains (e.g. e-Health), very few IoT architectures have addressed interoperability and integration issues among platforms. An example of these rare initiatives are the IoT platforms i-Core and Butler that were designed to be interoperable among them, but lacked interoperability with other platforms. Also, some other projects have recently addressed solutions for interoperability among platforms.

Probably one of the best examples of them is the European Horizon 2020 initiative INTER-IoT (InterIoT 2016) that proposes a novel solution for enabling interoperability among different IoT platforms across all their levels or layers, including semantic interoperability. It also facilitates the discovery, orchestration and composition of applications and services provided by different platforms. This open-source solution aims to guarantee a seamless integration of heterogeneous IoT technologies, and a horizontal integration of vertical systems.

### *4.1 The Inter-IoT solution for Global Interoperability among IoT Platforms*

Integration between heterogeneous elements is usually done at the device or network level, and it is typically limited to data collection (InterIoT 2016). In contrast, INTER-IoT offers a more complete and global solution, based on a set of methods and interoperability solutions across all different layers.

Next, the main solutions and benefits of INTER-IoT are summarized:

The main INTER-IoT solutions and benefits are:

- At the **Device level:** the INTER-IoT solution will enable the seamless inclusion of new IoT devices and their interoperation with existing heterogeneous ones. This will allow a fast growth of smart object ecosystems. As a solution at this level, INTER-IoT will provide a Device-to-Device gateway that allows any type of data transfer, thus making the device layer more flexible by decoupling the gateway into two independent parts: a physical part that only manages network access and communication protocols, and a virtual part that handles all other gateway operations and services. If connection is lost, the virtual part will remain functional and will answer the API and Middleware requests. The gateway will be modular to allow the addition of optional service blocks, in order to adapt to specific cases. It will support many network technologies such as ZigBee, LoRa, WiFi or PLC, and transport protocols (e.g. CoAP, Multipath TCP).
- At the **Network level:** INTER-IoT will provide seamless support for smart objects mobility (roaming) and information routing. INTER-IoT solution is based on SDN and NFV. It will create virtual networks that can be controlled through an API. Additionally, INTER-IoT will enable offloading, roaming and secure seamless mobility, important aspects in IoT that are related to interoperability at the network level.
- At the **Middleware level:** this solution provides a seamless resource discovery and management system for smart objects and their basic services. This will allow the global exploitation of smart objects in large-scale IoT systems. Different modules at this level will provide services to manage the virtual representation of smart objects, thus creating an abstraction layer to access all their features and information.

- At the **Application and Services level:** the main benefits are the use, discovery and combination of heterogeneous services from different IoT platforms by means of the INTER-IoT service discovery, service catalog and service composition.

In addition to the technical interoperability achieved with these solutions, INTER-IoT also aims to guarantee syntactic and semantic interoperability.

Regarding syntactic interoperability INTER-IoT performs a data format conversion among platforms, to put the information into the required syntax for the receiver platform.

Regarding semantic interoperability, INTER-IoT allows a common interpretation of data and information from different platforms and heterogeneous data sources (Ganzha et al. 2016), with a novel approach that provides universal semantic translation among platforms.

Translation among different platform ontologies is a very complex task. It is difficult and laborious to set up the alignments and rules between two specific platform ontologies. Moreover, readying the setup for all the possible combinations among any pair of existing platforms is an unfeasible task. The Inter Platform Semantic Mediator component (IPSM) will perform ontology-to-ontology translation. Though, instead of considering an overflowing number of combinations, for the sake of simplicity semantic translations will always be performed between a specific ontology of an IoT platform and the IPSM central modular ontology. This will drastically reduce the number of combinations of alignments and matches that must be set. Thus, this approach will allow universal semantic interoperability among platforms, thanks to this simplification.

The INTER-IoT solution can be employed in any application domain, or across different domains in which there is a need for interconnection and/or interoperability.

## 5. Use Cases

In this section, we explain two representative use cases of IoT that strongly rely on interoperability: Smart Cities and e-Health.

### *5.1 Smart City*

The use case of a smart city is a clear example of interoperability at many different levels because it includes wide sensor networks, gateways and middleware platforms that will handle and analyze the data collected from a common domain or several of them. IoT platforms enable the addition of value to the data gathered in the city and which applications can use to offer useful services to the citizens. The whole set of applications and services can comprise a single domain or several of them (Zanella et al. 2014). The potential benefits of interoperability on a large scale and across a whole city are manifold and important: relevant improvements in innovation, economic growth and well-being can be expected.

Another important objective in the area of smart cities that is highly related to interoperability is the adoption of a common standard and information model. By sticking to those shared standards and the aforementioned informational models, cities may accomplish the envisioned transformation with the least impact, thereby merging the intervening forces to contrive an ecosystem within which systems can link up and collaborate. This makes possible the fashioning of solutions, both interoperable and portable, that may be reproduced and tailored to the perceived needs and requirements of every concerned city.

IoT platforms provide a set of tools catering for different functionalities. They guarantee an innovation ecosystem for the creation of new applications and Internet services.

Next, we briefly present and explain a real smart city example.

*Valencia Smart City*

The city of Valencia (Spain) is the first case in Europe of a practically total integration of public services in a Smart City (above 95%). Regarding the technology and tools used, Valencia Smart City presents the adoption of FIWARE open data APIs and platforms to release open data. Web and mobile applications can make use of this data to offer a variety of services, such as route calculation and real-time estimated timing of the different public transport options (subway, bus and tramlines), bike rental information (availability, nearby stations), information about parking lots, etc. Furthermore, the Valencia Smart City Platform (VLCi Platform) provides an integral view of the city and its management and enables the improvement of the decision-making processes.

Across the city, a sensor network collects a wide variety of data from the environment, such as traffic information, public transport information, air pollution, noise, etc. Those sensors connect to several gateways, which perform data preprocessing tasks and send the data to the FIWARE Context Broker. The Context Broker mediates between data producers and consumers in order to allow access to the information regardless of its source. The information is processed in real time by a CEP (Complex Event Processing) component of the IoT platform, which identifies patterns and triggers events based on the application of predefined rules. The CEP enables instant response to changing conditions.

The gathered data is classified into different collections and stored in public repositories on CKAN, which is the most widely used software to build Open Data portals. Moreover, dynamic data is accessible in real time through the Real Time Open Data API.

Finally, a variety of applications and services make use of the available data. General tendencies can be identified or predicted by making use of Big Data analytics on aggregated datasets.

Applications and services can obtain data from the system upon request by making use of the defined APIs. In addition, the applications can also subscribe to events generated by the CEP or the Context Broker in order to receive information of interest whenever it is available.

The system described above is possible thanks to interoperability at different levels. Sensors and gateways interact at the device level and technical interoperability is needed in order to have network connectivity. The gateway also provides interoperability at the network level (routing) and syntactic interoperability through common data representations (JSON) and communication protocols (NGSI). At an upper level, the Context Broker receives and manages all the requests addressed to the IoT platform and provides interoperability at the middleware level. Finally, semantic interoperability is needed in order to ensure that the meaning of the data sent the IoT platform is understood by any application, system or service that consumes this information. Seeking to attain semantic interoperability in FIWARE, the authors of (Kovacs et al. 2016) propose the incorporation of knowledge-based semantic processing agents (KSPAs) to the platform. These agents can be incorporated to the FIWARE Data Model as hidden processing agents or can offer their semantic services to the applications, systems and services. Figure 1 shows the high-level architecture employed to enable the different interoperability levels in the proposed use case.

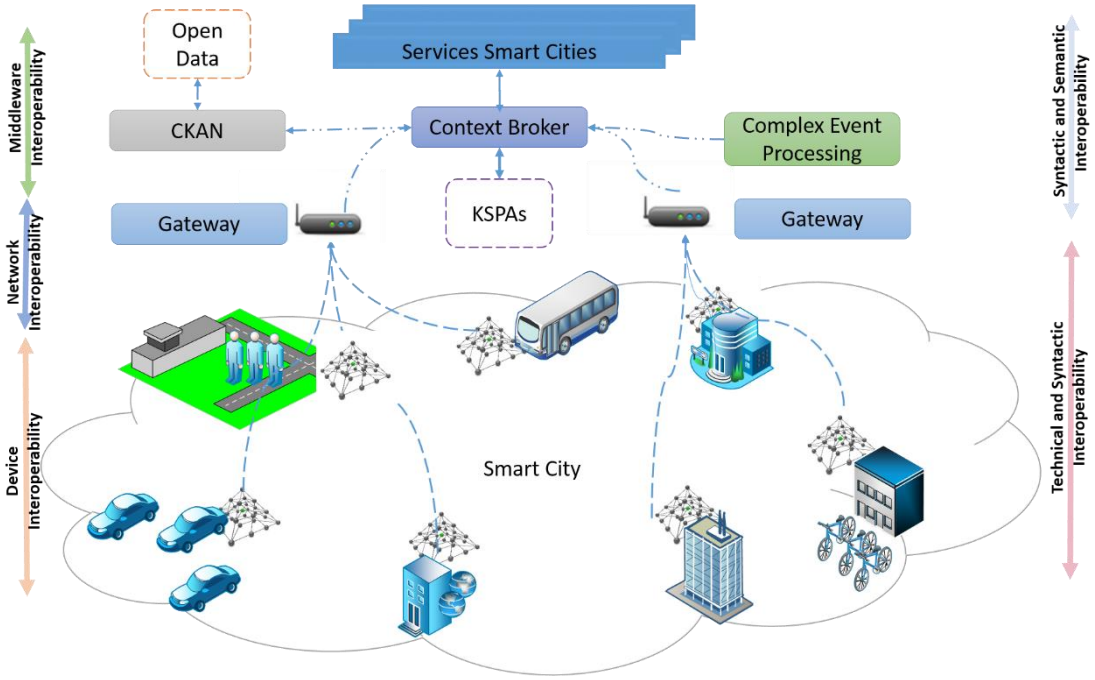


**Figure 1: Smart City case study schema**

### *5.2 e-Health Platform*

IoT interoperability can offer critical benefits in the area of health. Some of these benefits are the improvement in the comfort level of patients, the provision of remote patient monitoring which makes it possible to provide healthcare even in remote locations and an associated cost reduction due to a decrease in the number of hospital visits. Moreover, the use of wearable sensors and mobile devices allows for real-time monitoring and Big Data analytics can help to personalize healthcare and treatments (Farahani et al. 2017).

The development of an e-Health platform relies on interoperability at different levels. The sensors need to connect to a network in order to share their information. A gateway, which can be a physical device or an application running on a mobile device, provides access to the Internet in order to make the data available. A middleware platform provides integration and permits access to information in a transparent way, thus shielding the particulars of the devices from the applications making use of the data. In order to make sure that the data shared across the system

is interpreted correctly and proper actions are taken, semantic interoperability is required (Yin, 2015).

Next, we will present an example of the possible use of IoT interoperability solutions in the e-Health domain. In this example, a fictional continuous care system based on OneM2M will be explained. Continuous care allows elderly people and patients with chronic disease to reduce their visits to the doctor. Figure 2 illustrates how this e-Health platform would be implemented.

Sensors and medical devices monitor physiological variables, such as heart rate and oxygen saturation. These devices communicate with the gateway using wireless protocols, such as Zigbee or Bluetooth. In the gateway, the Common Service Entity of the Middle Node (MN-CSE) sends the data to the Common Service Entity of the Infrastructure Node (IN-CSE), which in this case is hosted in the Cloud by the OneM2M provider.

The Application Entities of the Infrastructure Node (IN-AE) interact with the IN-CSE and retrieve the information in order to provide health and environment monitoring services. This interaction is based on a publish/subscribe scheme. Hence, the platform sends the updated parameters to the doctor, who interprets the results. The system also performs some data processing in order to identify abnormal situations and send a warning to the emergency services when a potentially dangerous situation arises.

Regarding the different types of interoperability needed for this platform, the use of common standard wireless protocols provides device level interoperability, while the gateway provides network level interoperability. The middleware level interoperability is obtained by making use of an IoT platform. Technical interoperability is accomplished thanks to the use of communication protocols while the use of common data representations provides syntactic interoperability. Finally, the definition of data semantics though the use of ontologies allows for

semantic interoperability. Ontologies, which are represented as OWL files, describe the system as well as the meaning and purpose of the data. Currently, OneM2M provides one ontology, termed the oneM2M Base Ontology, which is the minimal ontology required for interoperability. This ontology can be extended using domain-specific ontologies. Hence, semantic interoperability would be achieved by the incorporation into the platform of ontologies describing concepts from the health domain.

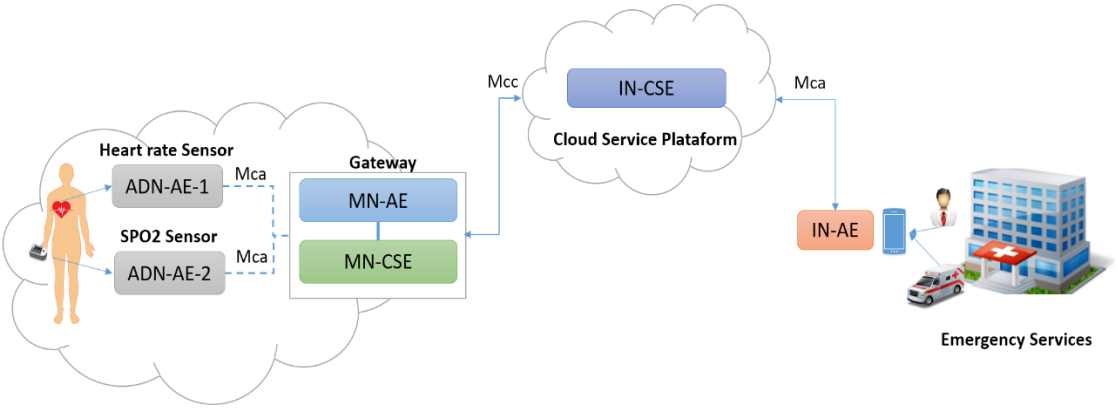


**Figure 2: e-Health case study schema**

## 6. Conclusions and Outlook

The concept of interoperability in IoT is defined and thoroughly explained in this chapter. Additionally, its crucial role and importance in IoT has been discussed, alongside with the general lack of interoperability at this moment, the problems that preclude its actual achievement, and the currently existent IoT standards. Interoperability has been analysed across several layers of IoT systems by studying the challenges posed by its achievement, possible technical solutions and use cases. Finally, the concept of global interoperability has been presented and an overview of the current situation of interoperability has been discussed in main outline.

IoT is going to be the next revolution of the information era and the next step in the path of modern society towards full digitalization. Interoperability is the key to unlock an immense untapped potential of IoT. To make it possible, a global reference standard for IoT is expected and very necessary. Its existence, in conjunction with a widespread acceptance and implementation, will solve the interoperability problem and allow the world to benefit from the full potential of IoT. More than a dozen possible global standards for IoT exist already. Among others, AllJoyn is pointed out as one of the best positioned candidates for becoming the future reference standard for IoT, due to a strong support from some very relevant technological firms. IoTivity is also a promising option as it is the candidate with the strongest support from the open source community. However, at present, all these possibilities seem to be uncertain and unclear, and their chances of success are considerably slim. No global reference standard is expected for the middle-term future.

The lack of a global reference standard and the vast heterogeneity in IoT environments are the main factors preventing interoperability in IoT. In this multi-standard scenario, new initiatives for enabling global interoperability seem destined to play a major role. It is important to mention the existence of the INTER-IoT initiative, which will allow for interoperability among different IoT platforms and across all their levels or layers. By means of that, INTER-IoT may play an essential role in the achievement of global interoperability and the disappearance of vertical silos. Thus, new initiatives and solutions capable of solving the interoperability problem may be key to unleash the enormous latent potential of the Internet of Things, which is still waiting to be awaken.

**References**

Alaya, M. Ben, Y. Banouar, T. Monteil, C. Chassot, and K. Drira. 2014. “OM2M: Extensible

ETSI-Compliant M2M Service Platform with Self-Configuration Capability." *Procedia Computer Science* 32:1079–86. Retrieved (http://dx.doi.org/10.1016/j.procs.2014.05.536).

Allseen Alliance. 2017. "AllJoyn Framework." Retrieved May 1, 2017 (https://allseenalliance.org/framework).

Aloi, G. et al. 2017a. "Enabling IoT Interoperability through Opportunistic Smartphone-Based Mobile Gateways." *Journal of Network and Computer Applications* 81(October 2016):74–84.

Aloi, G. et al. 2017b. "Enabling IoT Interoperability through Opportunistic Smartphone-Based Mobile Gateways." *Journal of Network and Computer Applications* 81(October 2016):74–84. Retrieved (http://dx.doi.org/10.1016/j.jnca.2016.10.013).

Atzori, Luigi, Antonio Iera, and Giacomo Morabito. 2010. "The Internet of Things: A Survey." *Computer Networks* 54(15):2787–2805. Retrieved (http://linkinghub.elsevier.com/retrieve/pii/S1389128610001568).

Bandyopadhyay, Debasis and Jaydip Sen. 2011. "Internet of Things: Applications and Challenges in Technology and Standardization." *Wireless Personal Communications* 58(1):49–69.

Bandyopadhyay, Soma, Munmun Sengupta, Souvik Maiti, and Subhajit Dutta. 2011. "A Survey of Middleware for Internet of Things." Pp. 288–96 in *Recent Trends in Wireless and Mobile Networks: Third International Conferences, WiMo 2011 and CoNeCo 2011, Ankara, Turkey, June 26-28, 2011. Proceedings,* edited by A. Özcan, J. Zizka, and D. Nagamalai. Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved (https://doi.org/10.1007/978-3-

642-21937-5_27).

Bormann, C. M.Ersue A.Keranen. 2014. “Terminology for Constrained-Node Networks RFC 7228.” Retrieved August 12, 2016 (https://tools.ietf.org/html/rfc7228#section-2.3.2).

Department of Defense of United States of America. 2008. *Dictionary of Military and Associated Terms*.

Diallo, Saikou Y., Heber Herencia-zapana, Jose J. Padilla, and Andreas Tolk. 2011. “Understanding Interoperability.” 84–91.

ETSI. 2013. *Interoperability Best Practices*. Retrieved (https://portal.etsi.org/CTI/Downloads/ETSIApproach/IOT_Best_Practices.pdf).

Farahani, Bahar et al. 2017. “Towards Fog-Driven IoT eHealth: Promises and Challenges of IoT in Medicine and Healthcare.” *Future Generation Computer Systems*. Retrieved (http://dx.doi.org/10.1016/j.future.2017.04.036).

FIWARE. 2017. “FIWARE.” Retrieved May 1, 2017 (https://www.fiware.org/tag/iot/).

Ganzha, Maria and Marcin Paprzycki. 2016. “Journal of Network and Computer Applications Semantic Interoperability in the Internet of Things : An Overview from.”

Ganzha, Maria, Marcin Paprzycki, Wieslaw Pawlowski, Pawel Szmeja, and Katarzyna Wasielewska. 2016. “Semantic Technologies for the IoT-An Inter-IoT Perspective.” Pp. 271–76 in *Internet-of-Things Design and Implementation (IoTDI), 2016 IEEE First International Conference on*.

Gubbi, Jayavardhana, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. 2013a.

"Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions." *Future Generation Computer Systems* 29(7):1645–60. Retrieved (http://dx.doi.org/10.1016/j.future.2013.01.010).

Gubbi, Jayavardhana, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. 2013b. "Internet of Things (IoT): A Vision, Architectural Elements, and Future Directions." *Future Generation Computer Systems* 29(7):1645–60. Retrieved (http://dx.doi.org/10.1016/j.future.2013.01.010).

H. Tschofenig, J. Arkko, D. Thaler, D.McPherson. 2015. *Architectural Considerations in Smart Object Networking - RFC 7452*. Retrieved (https://tools.ietf.org/pdf/rfc7452.pdf).

Hanke, Sten et al. 2011. "universAAL -- An Open and Consolidated AAL Platform." Pp. 127–40 in *Ambient Assisted Living: 4. AAL-Kongress 2011, Berlin, Germany, January 25--26, 2011*, edited by R. Wichert and B. Eberhardt. Berlin, Heidelberg: Springer Berlin Heidelberg. Retrieved (https://doi.org/10.1007/978-3-642-18167-2_10).

Hans van der Veer, Anthony Wiles. 2008. *Achieving Technical Interoperability - the ETSI Approach*.

InterIoT. 2016. "InterIoT." Retrieved June 1, 2017 (http://www.inter-iot-project.eu/).

International Organization for Standarization. 2003. *Information Technology Vocabulary, Fundamental Terms:ISO/IEC 2382-01*. Geneva.

Jacoby, Michael, Aleksandar Antonić, Karl Kreiner, Roman Łapacz, and Jasmin Pielorz. 2016. "Semantic Interoperability as Key to IoT Platform Federation." Pp. 3–19 in *International*

*Workshop on Interoperability and Open-Source Solutions*.

Kovacs, E. et al. 2016. "Standards-Based Worldwide Semantic Interoperability for IoT." *IEEE Communications Magazine* 54(12):40–46.

Krco, Srdjan, Boris Pokric, and Francois Carrez. 2014. "Designing IoT Architecture (S): A European Perspective." Pp. 79–84 in *Internet of Things (WF-IoT), 2014 IEEE World Forum on*.

Kreutz, Diego et al. 2015. "Software-Defined Networking: A Comprehensive Survey." *Proceedings of the IEEE* 103(1):14–76.

Linux Foundation. 2017. "IoTivity." Retrieved May 1, 2017 (https://www.iotivity.org/).

McKeown, Nick et al. 2008. "OpenFlow: Enabling Innovation in Campus Networks." *ACM SIGCOMM Computer Communication Review* 38(2):69–74.

Mckinsey Global Institute. 2015. *The Internet of Things : Mapping the Value Beyond the Hype*.

Mineraud, Julien, Oleksiy Mazhelis, Xiang Su, and Sasu Tarkoma. 2016. "A Gap Analysis of Internet-of-Things Platforms." *Computer Communications* 89:5–16.

Molina, B. 2014. "Empowering Smart Cities through Interoperable Sensor Network Enablers." Pp. 7–12 in.

Moumena, A., C. Mohamed, and N. Mohamed. 2012. "Challenges in Middleware Solutions for the Internet of Things." in *Proc. of The 2012 International Conference on Collaboration Technologies and Systems (CTS 2012)*.

Nunes, Bruno Astuto A., Marc Mendonca, Xuan-Nam Nguyen, Katia Obraczka, and Thierry Turletti. 2014. “A Survey of Software-Defined Networking: Past, Present, and Future of Programmable Networks.” *IEEE Communications Surveys & Tutorials* 16(3):1617–34.

Omnes, Nathalie, Marc Bouillon, Gael Fromentoux, and Olivier Le Grand. 2015. “A Programmable and Virtualized Network & IT Infrastructure for the Internet of Things: How Can NFV & SDN Help for Facing the Upcoming Challenges.” Pp. 64–69 in *Intelligence in Next Generation Networks (ICIN), 2015 18th International Conference on*.

OPENIoT. 2017. “OPENIoT.” Retrieved May 1, 2017 (http://www.openiot.eu/).

Perera, Charith, Arkady Zaslavsky, Peter Christen, and Dimitrios Georgakopoulos. 2014. “Context Aware Computing for the Internet of Things: A Survey.” *IEEE Communications Surveys and Tutorials* 16(1):414–54.

Sofia2. 2017. “Sofia2.” Retrieved June 1, 2017 (http://sofia2.com/).

Soursos, S. et al. 2016. “Towards the Cross-Domain Interoperability of IoT Platforms.” Pp. 398–402 in *2016 European Conference on Networks and Communications (EuCNC)*.

Tan, L., & Wang, N. 2010. “Future Internet: The Internet of Things.” Pp. 376–80 in *Advanced Computer Theory and Engineering (ICACTE), 3rd International Conference*.

Telecommunication Standarization Sector of ITU. 2014. *Common Requirements and Capabilities of a Gateway for Internet of Things Applications*. Retrieved (https://www.itu.int/rec/T-REC-Y.2067/en).

The Institute of Electrical and Electronics Engineers. 1990. “IEEE Standard Computer

Dictionary. A Compilation of IEEE Standard Computer Glossaries - IEEE Std 610."

Whitmore, Andrew, Anurag Agarwal, and Li Da Xu. 2015. "The Internet of Things—A Survey of Topics and Trends." *Information Systems Frontiers* 17(2):261–74. Retrieved (http://dx.doi.org/10.1007/s10796-014-9489-2).

World Economic Forum. 2015. *Industrial Internet of Things : Unleashing the Potential of Connected Products and Services*. Retrieved (http://www3.weforum.org/docs/WEFUSA_IndustrialInternet_Report2015.pdf).

Yacchirema, Diana, Carlos Palau, and Manuel Esteve. 2016. "Smart IoT Gateway For Heterogeneous Devices Interoperability." *IEEE LATIN AMERICA TRANSACTIONS, VOL. 14, NO. 8, AUG. 2016* 14(8):3900–3906.

Zanella, a, N. Bui, a Castellani, L. Vangelista, and M. Zorzi. 2014. "Internet of Things for Smart Cities." *IEEE Internet of Things Journal* 1(1):22–32. Retrieved (http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6740844).

AUTHORS

**Regel Gonzalez-Usach:** received his M.Sc. in Telecommunication at the Universitat Politècnica de Valencia, focused on networking, electronics and systems. She pursued her thesis at the University of Stuttgart under the topic of Multipath TCP and the development of new algorithms for this protocol. She is currently performing a Ph.D on Networking at the Universitat Politècnica de Valencia in Spain. Her research activities and interests cover Future Internet, Internet of Things, Multipath TCP, AAL and software applications.

*e-mail:* regonus@upv.es

**Diana Yacchirema** received the M.Sc. degree in Communications Technologies, Systems and Networks from the Universitat Politecnica de Valencia in 2011 and the M.Sc. degree in Management of Communications and Information Technology from the Escuela Politécnica Nacional, Quito-Ecuador, in 2009. Currently, she is a Ph.D Student in the Escuela Técnica Superior de Ingenieros de Telecomunicación at the Universitat Politécnica de Valencia, Spain. Her research activities and interests cover a wide range of subjects related to Internet of Things, sensor networks and network security.

*e-mail:* diayac1@doctor.upv.es

**Matilde Julián:** received her M.Sc. in Telecommunication and Biomedical Engineering and her Ph.D at the Universitat Politècnica de Valencia. Her research activities and interests cover a wide range of subjects related to biomedical signal processing, clinical applications, Future Internet, Internet of Things and AAL.

*e-mail:* majuse@upv.es

**Manuel Esteve:** Professor in Telematics Engineering. He leads the Distributed Real-Time Systems Lab (DRTSL) at UPV Communications Department. He has been involved in research and development projects during last 20 years for the application of multimedia and real-time technologies to industry, medicine, education, and Command and Control Information Systems (C2IS). Currently he is communications and information systems advisor of the Signal Brigade of the Spanish Army. As researcher and director of the DRTSL has large expertise in the following research lines: Command and Control Information Systems development (specifically C4ISR systems), Real Time Systems for emergency management, Virtual reality/ Real world integration for training, Video streaming and codification, Sensors networks deployment and simulation, and Tactical communications. DRTSL has developed SIMACOP, a Friendly Force Tracking C4ISR, used by the Spanish Army as SIMACET FFT. He is author of more than 150 papers in high quality magazines and conferences.

*e-mail:* mesteve@dcom.upv.es

**Carlos E. Palau** received his M.Sc. and Ph.D (Dr.Ing.) degrees, both in telecommunication engineering, from the Universitat Politécnica de Valencia in 1993 and 1997, respectively. He is Full Professor in the Escuela Técnica Superior de Ingenieros de Telecomunicación at the Universitat Politecnica de Valencia. He has more than 18 years of experience in the ICT research area in the area of Networking. He has collaborated extensively in the R&D of multimedia streaming, security, networking and wireless communications for government agencies, defence and European Commission. Currently he is leading the ITC-30 project Inter-IoT, focused in the achievement of global interoperability in IoT. He is author and co-author of more than 120 research papers and member of the TPC of several IEEE, ACM and IFIP conferences. He is Senior Member of IEEE.

*e-mail:* cpalau@dcom.upv.es