The Virtual Object as a Major Element of the Internet of Things: a Survey

Michele Nitti, Virginia Pilloni, Giuseppe Colistra, Luigi Atzori
Department of Electrical and Electronic Engineering - University of Cagliari, 09123, Cagliari, Italy
e-mail: *michele.nitti, virginia.pilloni, giuseppe.colistra, l.atzori@diee.unica.it*

***Abstract*—The Internet of Things (IoT) paradigm has been evolving towards the creation of a cyber-physical world where everything can be found, activated, probed, interconnected, and updated, so that any possible interaction, both virtual and/or physical, can take place. Crucial concept of this paradigm is that of the virtual object, which is the digital counterpart of any real (human or lifeless, static or mobile, solid or intangible) entity in the IoT. It has now become a major component of the current IoT platforms, supporting the discovery and mash up of services, fostering the creation of complex applications, improving the objects energy management efficiency, as well as addressing heterogeneity and scalability issues.**

This paper aims at providing the reader with a survey of the virtual object in the IoT world. Virtualness is addressed from several perspectives: historical evolution of its definitions; current functionalities assigned to the virtual object and how they tackle the main IoT challenges; major IoT platforms which implement these functionalities. Finally, we illustrate the lessons learned after having acquired a comprehensive view of the topic.

***Index Terms*—Internet of Things, virtual objects, IoT architectural solutions**

I. INTRODUCTION

The Internet of Things (IoT) is a promising paradigm, which integrates a large number of heterogeneous and pervasive objects with different connecting and computing capabilities, which are aimed at providing a view of the physical world through the network. The variety of devices capable to send data about the real world to the Internet is ever-increasing, so that the IoT can rely on a large range of objects, from dummy entities capable of providing just their positions (through attached tags) to objects capable of sensing the status of an environment, processing the data and sending the results to the cloud, if believed meaningful. This is making the range of possible applications increasing even more rapidly, so that nowadays our society can rely on powerful tools for introducing intelligence into living environments.

However, these applications introduce major challenges in the IoT field. The sensors needed to monitor the vital signs of a patient have limited resources, in terms of both computation and communication capabilities, so to guarantee quality requirements, the generated data have to be processed elsewhere [1]. Road-side units, traffic lights and road cameras can gather data to provide information about condition of the roads or parking lots; however, all these objects make use of different communications solutions and then a mean to make them interoperable is necessary [2]. To enhance the shopping experience of customers, goods can be enhanced with RFID tags, NFC or Quick Response (QR) codes, so that their information can be always available and pushed to users smartphones or tablets when necessary; the ability to interpret information about the environment and make choices accordingly can not be provided by these simple technologies and then must be handled by something else [3].

The above mentioned three-use cases have the final goal to provide services to the user through the Internet, but with vital information and services strictly linked to the real world provided through the key physical objects. However, the IoT will mainly be composed by objects with limited resources that enclose a wide range of different communication protocols. These stringent constraints directly impact on the design and deployment of IoT applications. To address these difficulties there is the need for technological solutions that *augment* the capabilities of the physical devices with additional functionalities and allow all of them to *talk to each other* at the *same level* to make the realisation of robust applications easier. A common solution of major IoT platforms has been the introduction of the concept of *virtual object*, which is the digital counterpart of any real (human or lifeless, static or mobile, solid or intangible) entity in the IoT and has the fundamental role to bridge the gap between the physical and the virtual world. It became a major component of the current IoT platforms enabling to perform operations otherwise hampered on real objects. Virtualisation has the ability to: make heterogeneous objects interoperable through the use of semantic descriptions; enable them to acquire, analyse and interpret information about their context in order to take relevant decision and act upon the virtual objects. Moreover, it enhances existing functionalities in the IoT supporting the discovery and mash up of services, promoting the creation of new addressing schemes, improving the objects mobility management efficiency, as well as addressing accounting and authentication issues.

This paper aims at providing the reader with a survey of the virtual object in the IoT world. Whether there are several papers analysing the technologies, standards, platforms and applications for the IoT world, there are not surveys that focus on the virtualisation of the object. Indeed, in [4], the authors explore the fundamental functional blocks of the middleware system, and based on these features proposes a classification on the existing IoT-middleware. Moreover, they present the open issues and the research scope in this area. Several works focus on research challenges and open issues to be faced for the IoT realisation in the real world, such as [5] and [6]. Even if the word virtual is commonly used in these surveys, its

DOI: https://doi.org/10.1109/COMST.2015.2498304

meaning is not well analysed, since most of the works do not ponder on the role of the virtualisation layer. However, to the best of our knowledge, our survey is the first to linger on what it means to have a virtualisation layer and which are its functionalities.

The major contributions of this paper are as follows. We provide an analysis of the definitions and assigned roles with an historical perspective, which is intended to understand the motivations that have brought scientists and technologists to introduce the virtualness aspect to address issues in different fields. We present the most common functionalities assigned to the virtual objects with the associated IoT challenges that are intended to be addressed. We review the current implementations of these functionalities in the major IoT platforms. Finally, we illustrate the lessons learned and the future challenges, and provide conclusions.

II. VIRTUAL OBJECTS DEFINITIONS AND CHARACTERISTICS

What testifies the strong interest in the virtualisation topic and the vivacity of the debate on it is, undoubtedly, the manifold definitions of *virtualness* traceable in the last decades. At a first glance, an interested reader might experience a real difficulty in understanding what a virtual object/entity/counterpart really means, which its functionalities are, and what its role in the Internet of Things world is. The goal of this Section is not to provide a complete overview of all the projects and papers on the topic; instead, we want to point out the milestones that led to the various definitions and characteristics of virtual object.

The reason today for the fuzziness around this term is a consequence of the extensive use of the word *virtual* in several contexts since the 70s, ranging from *the possibility of providing an efficient facsimile of one or more complete computer systems in a single machine* ([7] and [8]), to the *ability to provide a flexible abstraction of entities such as memory and time* ([9] and [10]). The 90s were the era of interactive virtual environments and for the first time, the words virtual and object were linked together, so that *a virtual object was defined as a digital representation of the functionalities and shape of real-world objects and by how a user can interact with it* [11]. However, in this view, there is not any real possibility to interact with the physical object as everything is only virtual.

Objects virtualisation in the IoT began with the first related technologies, i.e. RIFD tags [12] and [13]. Despite their simplicity and limitations, passive RFID-based identification systems enabled the implementation of a wide range of novel applications by bridging the gap between the physical world (i.e., tagged real-world objects) and the virtual world (i.e., application software). For example, the virtual objects proposed by Barrett et al. [14] enable to enhance labelled physical objects with digital information. In a similar way, Want et al. [15] report a variety of scenarios where virtual objects are used to augment everyday objects via embedded RFID tags. Last, but not least, the Cooltown project [16] attributes a Web page not only to people and places, but also to arbitrary things. Although these papers investigate only simple applications, they share a common definition of *virtual objects in the IoT as digital representation of the service(s) of a real world object*. As shown in Figure 1, this represents the first definition of virtual object for IoT, which can be considered still valid with additional features being added over time as discussed in the following. With respect to the definition of virtual object in the virtual environment defined in the 90s, in the IoT the virtual object must have a counterpart in the physical world, of which it exposes the services, and does not have necessarily a shape. This definition will remain essentially unchanged over time but the virtual objects will acquire new characteristics.

The digital counterparts analysed in the early IoT exposed only one specific functionality to the final users based on the particular application at hand. One of the dimensions that characterise the definition of a virtual object is the *association between real objects, i.e. their services, and the virtual objects themselves*: for example, a smartphone could expose all its services through a single virtual object or it could have distinct virtual objects based on which services are made available, e.g. one for the localisation services and one for the temperature sensing; in the same way, it is possible to use a single virtual object to collect information of the same service from several physical objects. Obviously, every choice carries its own advantages and disadvantages in terms of addressing, service discovery or resource reusability. This dimension has been addressed differently over time, as discussed in the following.

The pioneering work of Langheinrich et al. [17] back to 2000 is the first that addresses requirements for a large scale deployment of RFID tags. In their paper, even if the definition of virtual objects remains the same, they acquire a new characteristic: an *identification and addressing scheme* in order to be able to locate the virtual objects in the Internet. In this proposal, some virtual objects, such as emails and Web pages, could not be associated with a physical object. Moreover, the authors consider the possibility of having only one virtual object for a physical object (if any), then leading to a one (or less)-to-one association.

A further step in the evolution of virtual object (or virtual counterpart) is provided in the work of Römer et al. [18]. The authors introduce the concept of *meta counterparts*, which represents a whole set of physical objects. For this reason, it is possible to manage the virtualisation more efficiently than having a distributed implementation, because it allows to reduce the resource consumption and to enhance the *service discovery*. This represents the first attempt to associate the virtual objects based on their functionalities: every virtual object is then able to handle not only the set of services of a single real object but also the services of a group of similar objects and then it moves to a *many-to-one the association between real and virtual objects*.

In the last 5 years, several research projects were founded to propose an architecture for the IoT, leading to a further evolution of the functionalities of virtual object.

The CONVERGENCE project [19], for example, makes use of Versatile Digital Item (VDI), a common container for all kinds of digital content, including one or more physical objects and the corresponding metadata. The proposed virtual object

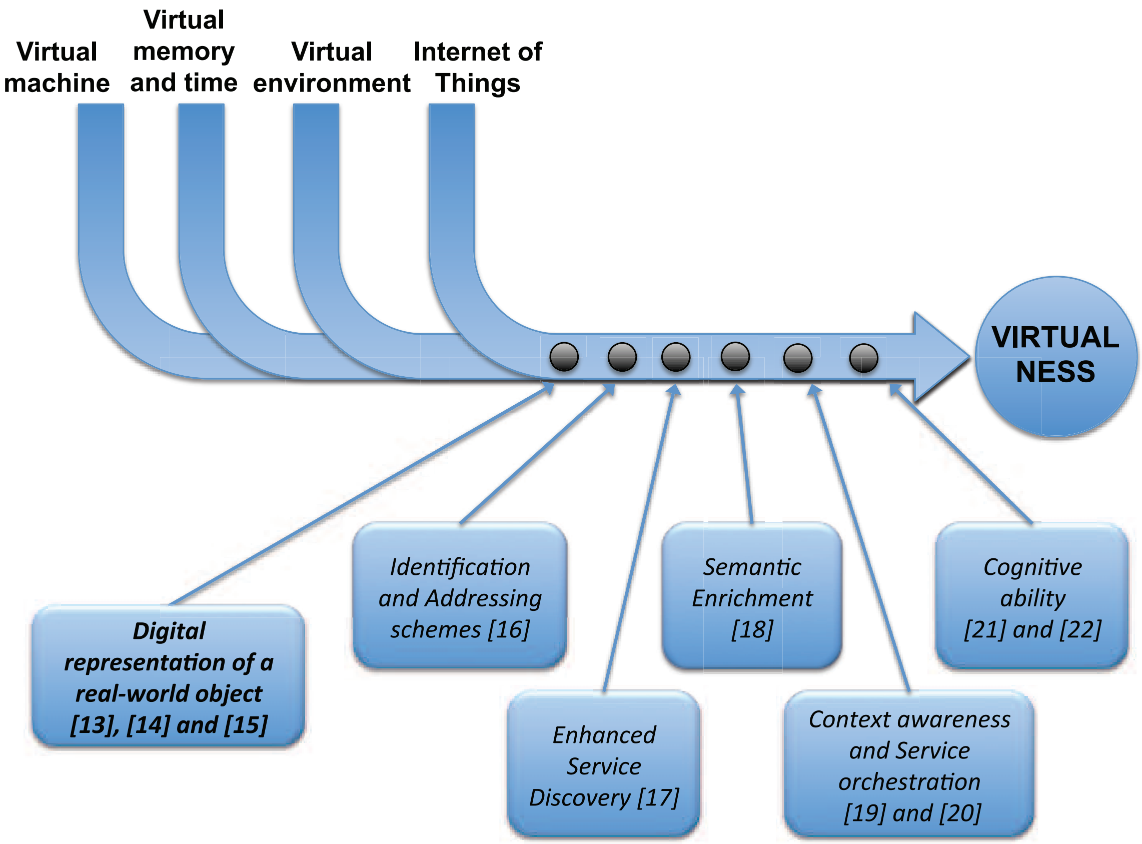

Fig. 1. Evolution of the term virtualness

is similar to the one provided in [18] with a *many-to-one association between real objects and VDI*, i.e. a single VDI can represent multiple digital contents. However, virtual objects are not anymore only digital counterparts of real world objects but now provide a *semantic enrichment* of the acquired data, which makes the discovery of services easier, since metadata is used to index the virtual objects.

Other two interesting examples of the evolution of the virtual object are SENSEI [20] and IoT-A [21]. The virtual objects, called resources in SENSEI and virtual entities in IoT-A, become *aware of the context in which the physical object operates* and then acquire the ability to enhance the data received by the real world objects with environmental information. Moreover, these projects propose for the first time methods to *orchestrate IoT services* in order to combine together several virtual objects and provide high-level services to the user or to the application. However, SENSEI and IoT-A differ for the association between real and virtual objects. SENSEI resources may be associated with one or more physical objects, thus providing the *same type of association of CONVERGENCE*. On the other hand, the IoT-A physical object is decomposed in its basic services thus providing for the first time a *one-to-many association* with the virtual entities.

Another step in the evolution of the virtual object is provided by the COMPOSE [22] and the iCore [23] projects. Virtual objects acquire a *cognitive ability*, which allows them to use the collected information to make relevant decision and acting upon it. Moreover, even if COMPOSE follows the *same type of association* of IoT-A, an interesting trait of the iCore project is that not only a virtual object can be associated to one (or more) real-world objects but also a real-world object to one or more virtual objects, then leading to a *many-to-many association*.

From these proposals it arises that the virtual objects have become the necessary element to address some key issues in the IoT, whose major ones are: quick deployment of new network elements and architectures by connecting the virtual object to external services; allowing for co-existence of heterogeneous network architectures over a common infrastructure; providing all the time reachability, even when the physical devices is temporarily unavailable; add intelligence to the object to be able to understand the context and implement self-management functions. All the operations performed by the virtual objects to provide these functionalities are extensively discussed in the following section, highlighting the key value.

In the following, we will refer to virtual object/entity/counterpart as interchangeable terms, with the meaning of digital representation of real-world object.

III. ROLE OF THE VIRTUAL OBJECT IN THE INTERNET OF THINGS

The intent of this section is to describe the functionalities that are assigned to the virtual object in the IoT arena, after quickly reviewing the major issues in the field.

A. Major issues in the IoT

The basic idea of the IoT concept is the pervasive presence of connected objects [24]. Since its birth, the concept of IoT has evolved into multiple dimensions, encompassing various technologies able to provide real-world intelligence and goal-oriented collaboration of distributed smart objects via local interconnections or global networks. As defined in [25], the current situation is still represented by a combination of many "Intranets" of Things, which are evolving into a much more integrated and heterogeneous system to converge to the melting-pot that characterises the IoT ecosystem.

Such a complex environment is faced with a long and varied list of different issues. We will survey and discuss these issues in the following.

1) Heterogeneity: One of the main enabling factors of the IoT is the integration of several technologies and communications solutions. Many efforts have been spent to develop protocols for ubiquitous [26] and pervasive [27] networking (e.g. ZigBee, Bluetooth), but each solution has their own specific characteristics and application domains. This determines a fragmentation that may hamper objects interoperability and can slow down the development of a unified reference model for the IoT.

2) Scalability: The number of objects connected to an IoT system is several orders of magnitude higher than that of devices of the conventional Internet [28]. The quick spread of connected objects is going to contribute to the deployment of a Very-Large-Scale (VLS) system of pervasively connected devices across the globe [29]. The management of such a system is very demanding, and if any fault is experienced the performance is severely threatened.

3) Identification: In order for devices to be able to communicate, the system they belong to first needs to identify them with a unique ID. The IPv4 [30] will soon be replaced by the IPv6 [31], which is going to compensate the lack of new IPv4 addresses, as well as enabling network auto-configuration, stateless operations, and low-power wireless communication nodes within the 6LoWPAN context [32]. Recently, other solutions have been developed to run in resource-constrained environments, typical of an IoT. Some examples are Message Queue Telemetry Transport (MQTT) [33] and Constrained Application Protocol (CoAP) [34]. However, their performance is optimised when used in all-IP networks, and in IoT scenarios this not always occurs. A workaround could be using a high level middleware that allows to maintain some "islands" that use arbitrary multi-hop, ad-hoc routing algorithms to deliver object's data to one or more sink nodes. An example is the Global Sensor Networks (GSN) middleware, which abstracts the details of access to sensor data [35].

4) Plug and Play: As soon as a device joins a network, it is registered to it, and its available services and resources are advertised. In IoT systems, the challenge is to make this process automatic and dynamic, despite its high level of heterogeneity and the difficulty to acquire, analyse and interpret information about the context. Objects should be "plug and play", i.e. some parameters, which enable the object to immediately interact with other objects, are configured without the need for human intervention.

5) Search and Discovery: One of the main requirements to build an application in a distributed system such as the IoT, is the dynamic discovery of the services provided by distributed objects [36]. When a device joins the IoT, everything is unknown about it, even the exact location and the format of exchanged data. Discovery mechanisms enable devices to automatically register themselves and advertise their services on the network. An interesting overview, comparison and analysis of some discovery services in the IoT is performed in [37].

6) Constrained Resources: In the IoT, available resources such as energy, storage, processing, and node capability to perform a service, are often limited [38]. This is the case, for example, of limited energy amounts of battery powered wireless sensor nodes, and scarce processing capabilities of RFID tags. Resources management is therefore needed to improve their usage. Cooperation among nodes and use of optimisation algorithms for resource and service allocation [39], even using cloud [40] or fog computing [41], are trending topics in the IoT.

7) Quality management: It refers to the whole system of operations that ensures that quality requirements in the IoT are achieved. This concept encompasses:i) Quality of Service (QoS), i.e. the set of service requirements in end-to-end communications (e.g. delay, jitter, available bandwidth and packet loss) [42] and [43]; ii) Quality of Experience (QoE), i.e. the overall acceptability of an application or service, as perceived subjectively by the end user depending on its expectations and the context [44]; iii) Quality of Information (QoI), i.e. the degree of accuracy of the information provided by a service, with reference to its counterpart in the real world [45]. Quality management is particularly problematic in the IoT due to heterogeneity and mobility issues. As surveyed in [46], end-to-end quality management in highly heterogeneous networking environment requires high operational costs. This is particularly heightened in the case of IoT, where applications with different quality requirements (i.e., non real-time, quasi real-time, and real-time) are accomplished at the same time.

8) Mobility: Typical IoT scenarios are characterised by mobile objects. Mobility management is fundamental to provide seamless connectivity independently from where the objects are placed or moved to. Moreover, it also needs to consider data inconsistency that may result from the change of context of a mobile object. As surveyed in [47], objects can move:i) intra-domain, i.e. moving between different cells of the same system and ii) inter-domain, i.e. moving between different backbones, protocols, technologies, or service providers. While the intra-domain is supported by several protocols [48], the inter-domain mobility, especially among heterogeneous systems, is an open issue.

9) Security and Privacy: Security and privacy are critical issues for the IoT. As proposed in [49], security and privacy requirements can be classified in: resilience to attacks, data authentication, access control and client privacy. The fulfillment of these requirements affects the privacy and security of all the involved stakeholders [50]. In the IoT, although each node might be perfectly safe by itself, the interaction with other nodes may threaten its security. The design of security

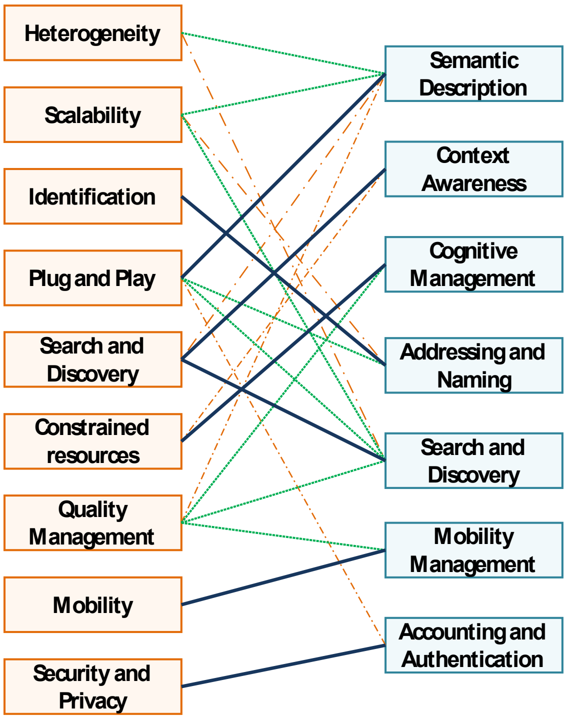

Fig. 2. Relations between IoT issues and virtualization layer functionalities. Blue solid lines connect functionalities with the main issue they are intended to address; green dashed lines connect functionalities with other issues that are improved by them; orange dash-dot lines connect functionalities with issues that may be improved by them, depending on the way they are implemented.

mechanisms, specific for the IoT, is still in its infancy, but the real challenge is to approach this issue with a holistic view.

B. Functionalities Associated to Virtual Objects

As it arises from the analysis of the previous Subsection, the hardest challenge of the IoT is to be able to address the deployment of applications involving heterogeneous objects, often moving in big environments, in a way that satisfies the quality requirements of the application itself, while not overloading the network resources. The virtualization of objects seems to be the perfect answer to this problem, as it is intended to support:

- quick deployment of new network elements and architectures [51];
- co-existence of heterogeneous network architectures over a common infrastructure [52];
- all the time reachability [53];
- self-management of network objects, achieved through context awareness [54].

In the following, we analyse the key functionalities that are provided by virtualization layers to satisfy the IoT issues described in Section III-A (Figure 2 visually shows this relation between issues and functionalities). We distinguish between *virtualization-specific* functionalities, which are introduced by the virtualization layer, and *virtualization-enhanced* functionalities, which are functionalities that already exist in IoT architectures, but their performance is improved by the use of a virtualization layer.

1) Virtualization-Specific Functionalities:

a) Semantic Description: One of the most critical requirements of the IoT is that of making heterogeneous objects plug-and-playable and interoperable. As soon as an object joins a network, it needs to be immediately provided with mechanisms that enable interaction with the external world. Network discovery mechanisms are used to dynamically discover and configure new virtual objects at runtime. The most effective and efficient solution to represent IoT objects is by using semantic technologies.

The result of the semantic description of an object, or a group of objects, is the virtual object model. The virtual object model includes: objects' characteristics; objects' location; resources, services and quality parameters provided by objects. Whenever a new object is detected, it is associated with a new virtual object instantiation [55].

The first attempts of standardisation in semantic description came from the Wireless Sensor Network (WSN) field. The W3C's Semantic Sensor Networks (SSN) ontology [56], and the OGC Sensor Web Entablement suite [57] of XML-based standards are just some examples of semantic description languages. These are still used in many virtualization architectures (e.g., SENSEI [20] and IoT-A [21]).

Not only is semantic description able to cope with heterogeneity and to provide interoperability among objects, but it is also very powerful in supporting search and discovery operations [4]. Indeed, search and discovery mechanisms are usually used to find the virtual objects that are most suitable to perform a given application's task (or part of that task) [58]. At this purpose, search engines are used, which are usually based on semantic search technology.

b) Context Awareness: This function is the capability to acquire, analyse and interpret information about the environment. Awareness is needed to simplify the discovery of information about the object itself when this information: cannot be easily reached, is not made explicit in the required format, or needs aggregation with other information sources before it can be used [59]. Discovered information is represented by resources and services that are made available by other nodes, as well as data gathered by them. Although context awareness can be performed in traditional IoT by smart objects, in virtualization layers this functionality is assigned to virtual objects. In this way, awareness is enabled even in objects with limited capabilities: smarter tasks that they would not be able to perform are provided by the virtualization layer, which enables awareness mechanisms. Therefore, it can be considered a virtualization-specific functionality.

c) Cognitive Management: This virtualization-specific functionality is strictly related to context awareness: the knowledge acquired about the environment where the virtual object is operating is used to make relevant decisions and act upon them [60]. Whenever a virtual object perceives a change in the context, it processes information through the use of optimisation algorithms and predictive models [61]. According to the obtained results, the virtual object autonomously reconfigures the object it refers to in order to adapt its behaviour to changes.

The cognitive management functionality is extremely important for many aspects [55].

Automatically detection and scheduling of events by the objects, which also react by performing a self-configuration of their functioning parameters.

Resource usage is optimised (e.g. energy, processing capabilities, communication bandwidth, storage). Optimisation algorithms are implemented to efficiently allocate resources to tasks, provided that required quality conditions are still satisfied.

Fault management is performed whenever the virtual object cannot communicate with the object it is associated with. In this case, the virtual object: estimates the object state using cognitive/prediction methods; proposes alternative links through which data can be forwarded.

Learning mechanisms are implemented for predicting future situations and preventing future faults.

2) Virtualization-Enhanced Functionalities:

a) Addressing and Naming: As soon as an object is installed in a network, it is fundamental that an address and/or a name is assigned to it. This is a virtualization-enhanced functionality, as in traditional IoT it is in charge of addressing real objects. In a virtualization layer, it gives addresses and names to newly created virtual objects, so that their resources, services and data are accessible by any other entity on the Internet [5], in a way that scales well when the network size grows.

The location-dependent addressing scheme used in the current Internet architecture [5] is not suitable with the modern vision of IoT, which is based on sharing data. Therefore, information-centric architectures are supposed to be the future of IoT. These architectures are particularly desirable in virtual object based architectures, thanks to their inclination towards semantic search mechanisms. For this reason, naming is a key element for virtual objects' identification and interaction [55].

b) Search and Discovery: The nature of IoT is that of a dynamic system where objects continuously join, move and leave the platform. As such, a virtualization layer should be able to manage virtual objects that degrade, vanish or even re-appear. In this context, automated discovery and search mechanisms are needed to achieve a scalable and accurate interoperability among virtual objects [62]. This functionality is virtualization-enhanced: it is the virtual objects' counterpart of search and discovery performed on real objects. Since virtual objects enable those mechanisms that expose the characteristics and capabilities of the physical objects they virtualise, the search and discovery processes identify which virtual objects provide the resources and services that best fit the quality level required by the application assigned to the IoT system. As introduced in Section III-B1a, the semantic description model eases the discovery of virtual objects. Thanks to semantic engines, the use of a common language enables the research among devices with extremely different characteristics, e.g. a sensor and a smartphone can both respond to the request of sensing temperature. Furthermore, the extended use of the cloud in virtualization architectures makes search and discovery much quicker, even achieving real-time search [63] (see Section IV-B).

In particular, Peer-to-Peer (P2P) discovery systems, which are based on information-centric technologies, are preferable to other discovery mechanisms for their capacity of using semantic information to discover resources and services [5]. In fact, P2P approaches make use of a distributed infrastructure, over which resources and service descriptions are spread and managed [64]. Using semantic-based algorithms, relevant information is retrieved in a distributed way. Not only are scalability and availability improved, but also achieved search results are more accurate with respect to other mechanisms.

c) Mobility Management: Mobility management is not a negligible issue for virtual objects: when an object moves from one place to another, its virtual counterpart needs to take notice of its migration. This task may result very difficult, particularly when the object becomes unavailable for long periods of time due to lack of connectivity. In these cases, the cognitive capacities of the virtual object are extremely important: it is required to be resilient to context changes, by trying to maintain its operational functionalities with an acceptable quality level [55]. This further implies trying to maintain a good quality communication link with its related object whenever possible, also taking advantage of opportunistic and eventually intermittent connections. Finally, since the virtual object may be storing out of date information of the object, a parameter indicating how much time is passed from the last time the virtual object received data from its physical counterpart is required. If needed, the virtual object should be able to use predictive models on this data, in order to estimate the state of its associated object.

d) Accounting and Authentication: Accounting and authentication IoT functionality is enhanced by virtual objects, which describe the users/entities whose access is authorised, as well as their owner and/or creator. Their access needs to be secured through the use of encryption keys [55].

As for data security, virtual objects can use semantics in order to build a universal trust management system [65]. In such a system, autonomous trust management procedure and modules process and interpret trust descriptions, and give explicit meaning to trust data using semantic annotation.

IV. VIRTUAL OBJECTS IN PROPOSED IOT ARCHITECTURE SOLUTIONS

Nowadays, almost every IoT architecture makes use of a layer of virtualisation, where things of the real world and related information, become accessible every time and everywhere by the upper layers. The functionalities highlighted in Section III-B are common to most of the current platforms and it is then not practical to classify them based on the features they implemented; for this reason, we present a simple taxonomy based on the type of association between real and virtual objects. Indeed, this association represents the very first implementation choice and it has a direct impact on the performance of the platform. For example, the creation of a virtual object for every service of a single physical object (one-to-many association) can facilitate the service orchestration but it requires to replicate the semantic information of the object in every virtual object. On the other hand, by representing

TABLE I
IMPLEMENTATION OF THE VIRTUALISATION LAYER FUNCTIONALITIES IN IOT ARCHITECTURES.

	ETSI M2M and oneM2M	FI-WARE	SENSEI	IoT-A	COMPOSE	iCore
Semantic Description	Under consideration in a future release	X	X	X	X	X
Context Awareness	NO	X	X	X	X	X
Cognitive Management	NO	NO	NO	NO	Application-related	X
Addressing and Naming	Use of a globally unique identifier (SCL-ID)	Use of an entity ID and type	Use of a globally unique identifier (not specified)	Use of Uniform Resource Identifier (URI)	X	Use of Uniform Resource Locator (URL)
Search and Discovery	X	X	X	X	X	X
Mobility Management	Not defined	Not defined	Handled by the Communication Service Layer	Physical Entities periodically update their locators	Web Objects periodically update their location	Use of Persistent URL (PURL)
Accounting and Authentication	X	X	X	X	X	X
Association real-virtual objects	1-to-1	1-to-1	Many-to-one	One-to-many	One-to-many	Many-to-many

a whole set of physical objects with a single virtual object (many-to-one association) it is possible to manage them more efficiently, because it allows applications to reuse the same virtual object and then reduce the resource consumption.

In the following, we present the major architectural proposals in the field of the IoT, pointing out the role and the importance of the virtual object inside them. In Table I, we visually show the matchup between the functionalities presented in the previous Section and the different architectures.

Finally, we highlight the common building blocks needed to implement a virtualisation layer.

A. Architectures Existing in Literature

1) One real object for one virtual object:

a) ETSI M2M and oneM2M: In January 2009, the European Telecommunications Standards Institute (ETSI) [66] started an effort to create a robust M2M common Service Layer. In 2012, intensive efforts on synchronisation of M2M standard activities led to the oneM2M Global Initiative [67] with the goal to develop globally agreed technical specifications for a common M2M Service Layer.

The virtualisation layer is represented by the M2M Service Capability Layer for the network, the M2M Device and the M2M Gateway.

When an application on a M2M Device that is not always connected wants to send some data to another application, it writes data to a resource (the equivalence of a virtual object, associated only to a single real world object) in the M2M Service Capabilities in the Network Domain. The resource is a globally uniquely addressable entity in the RESTful architecture and its representation shall be transferred and manipulated using one of the four basic methods of a RESTful architecture (create, retrieve, update and delete).

At the moment (ETSI M2M Specifications rel. 1) the content of the resources is "opaque" at Service Capabilities Layer i.e. the applications, writing and reading a resource, must agree somehow on the format of the resource content, but it is under consideration the addition of a semantic description of the resources in future releases of the standard. The other virtualisation-specific functionalities are not currently contemplated.

b) FI-WARE: FI-WARE [68] is based upon Generic Enablers (GEs), which offer reusable and commonly shared functions, serving a multiplicity of usage areas across various sectors. The Core Platform provided by the FI-WARE project is based on GEs linked to several FI-WARE Technical Chapters; one of these chapters is related to the IoT.

Sensors as well as other real-world things (e.g. rooms, persons) are modelled as virtual things having an ID, a type and several attributes, so that every object is associated only to one single virtual thing.

This architecture has already taken into account the ETSI M2M specification and has extended it to incorporate Open Mobile Alliance Next Generation Services Interface (OMA NGSI) activities. The OMA NGSI Context Management standard provide interfaces to manage and exchange Context Information about Context Entities. A Context Entity is defined as any entity which has a state. Attribute values of defined entities becomes the Context Information that applications have to be aware of in order to support context-awareness. Context Information is any volatile or persistent information, which describes the state of a Context Entity. Context Entities could be users, devices, places, or in general the things as defined in FI-WARE.

2) Many real objects for one virtual object:

a) SENSEI: The SENSEI project [20] enables the integration of heterogenous and distributed Sensor and Actuator Network (SAN) islands into a homogenous framework for real world information and interactions. It provides an abstraction level of resources corresponding to the real world consisting of Real World Entities or Entities of Interest (EoI); the Resource Layer is the core of the architecture since it encompasses all necessary functions that facilitate the interaction of services with the EoIs.

The concept of Resource is central since it provides a unifying abstraction for simple devices such as sensors, actuators, processors or software components. Resources may be associated with one or more EoIs for which they can either provide information or provide control over their surrounding environment, thus providing a many-to-one association between real and virtual objects. The Support Services enable discovery, composition and dynamic creation of Resources and support long term interactions. The Community Management is in charge of providing identity management for all entities in the system, privacy and security features.

3) One real object for many virtual objects:

a) IoT-A: The IoT-A project [21] extends the models introduced in SENSEI and proposes an architectural reference model for the IoT.

The core of the project is the entity that constitutes the "things" in the IoT. The entities are composed by two distinct counterparts: a physical counterpart (physical entity), which is a discrete, identifiable part of the physical environment and can be almost any thing such as a human, an animal, a car, a closed or open environment, and so on. A virtual counterpart (virtual entity), which represents the physical entity in the digital world.

Entities are associated with resources, i.e. software components that provide information about physical entities or enable the controlling of devices. A virtual entity can be associated with one or more resources that enable interaction with the physical entity. Ideally there is only one physical entity for each virtual entity. However, it is possible to associate the same physical entity to several virtual entities: the physical object is decomposed in its basic services, each of them handled by a virtual entity, thus providing a one-to-many association with the virtual entities. This association is important in look-up and discovery processes.

Resources are heterogeneous since they can represent on-device resources, stored data or network resources. In order to access them, the model makes use of services, which provide a well-defined and standardised interface, offering to the user all the functionalities for interacting with physical entities.

b) COMPOSE: The design of COMPOSE [22] architecture makes use of IoT-A as reference, thus providing its same type of association.

COMPOSE is basically a platform with which external stakeholders can interact. Physical objects are known as Web Objects, which provide the feed for the platform, offering physical resources to all internal processing components up to developers and end-users. To overcome the heterogeneity of Web Objects, Service Objects are created, which exhibit standard API towards the rest of the components within the COMPOSE platform.

Furthermore, Service Objects are enhanced by semantic metadata and stored in a registry. The enriched description can be used by the discovery mechanism to supply external users with information on the characteristics and functional aspects of the Service Object. Developers interact with the platform via a portal that enables to locate existing entities (in the form of Service Objects represented in the platform and COMPOSE applications) which are of interest to them. New COMPOSE applications are created potentially based on existing ones, which are enhanced with cognitive capability.

4) Many real objects for many virtual objects:

a) iCore: The iCore framework [23] is organised in three levels: in the Virtual Object (VO) level, virtual representations of any real-world object are dynamically created and destroyed, so that any physical object becomes virtually "always on"; at this level cognitive technologies are used in order to maintain a constant link between real-world object and VO so to ensure self-management and self-configuration. In the upper layer, the Composite Virtual Object (CVO) level, a cognitive mashup of semantically interoperable VOs is created, in order to offer an automated ability to aggregate VOs to meet application requirements; the CVO enables the reuse of existing VOs outside of their initial domain. The last layer is the Service layer, which has the role to translate the application requirements into services to be fulfilled by the CVO level.

VOs and CVOs are implemented as web resources using RESTful Web Services and stored in the VO and CVO registry to be accessible for the authorised parties. They are described through semantics and contextual information, which are used for the specification and high-level description, registration, discovery and access/invocation of existing VOs and CVOs. Moreover, this description enables VOs' reusability, so that a single VO can be associated to many real objects and vice versa.

B. Building Blocks for a Virtualisation Enabled Architecture

In Figure 3, we highlight the common building blocks needed to implement a virtualisation layer. The core component is the process representing the virtual object, which is the one that interfaces with the physical object and the external services and applications. Following the description in Section III-B, in this figure we show virtualisation specific and enhanced blocks. In the first category, semantic description can be considered the only required functionality; however, with the exception of ETSI M2M, a context awareness module is implemented in all modern architectures. The ability to make relevant decisions based on the knowledge acquired, offered by a cognitive management module, is only implemented in the iCORE and COMPOSE architectures, and in the latter case is only application-related. Regarding the virtualisation enhanced blocks, we can find the same modules as in classic architectures with the exception that they act directly on the virtual objects instead of the physical objects. In the IoT scenario, mobility management will surely be a core building block, however not every architecture defines it explicitly.

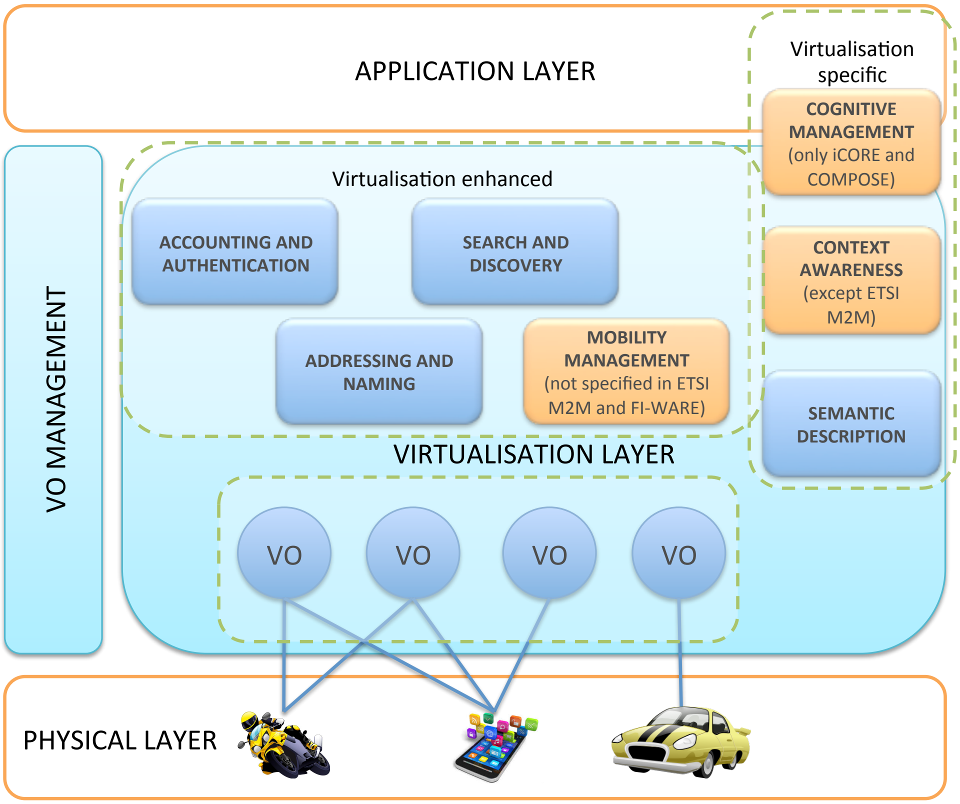

Fig. 3. High level components for a generic virtualisation enabled architecture

In order for virtual objects to manage the functionalities defined in Section III-B, a management block is needed to:

1) *Create* them: whenever a new object is detected, it is associated with a new virtual object instance [55], based on the information it provides about its resources and functionalities. This information is then translated into a semantic language that is understandable by the other objects that belong to the system.

2) *Maintain* and *coordinate* them during their whole lifecycle: mechanisms are provided to enable interoperation among virtual objects instances [69], in order to prevent and resolve conflicts and instabilities deriving from concurrent invocations of the same virtual object.

3) *Delete* virtual objects when they are not used anymore.

The place where virtual object instances are located depends on the architecture requirements. Not every existing virtual object-based architecture defines precisely where the virtualization layer is located. However, most of them place it in the *cloud*, as it offers integration among monitored data, stored data, analytics tools, visualisation platforms and client delivery [70]. Not only are the processing capabilities distributed among multiple intelligent objects, but also storage is distributed, making real-time searches easier to be performed [63]. Communication between virtual objects is web-based, and it usually makes use of RESTful technologies. Nevertheless, sending all the data collected by IoT elements to the cloud may have two main drawbacks: 1) the increase in latency and required bandwidth that may affect the device resources; 2) the need to manage a huge number of objects at the same time causes an increase in complexity, which leads to an inefficient use of the cloud resources (e.g. available energy), as well as higher execution times.

This approach may not be acceptable for real-time applications, especially for those where emergency situations need to be handled [71]. For this reason, implementing some of the virtual objects functionalities close to the places where the *real objects* are located, needs to be considered. For example, in smart home energy management systems the whole data collection, data processing, and decision making process can be performed locally, without the need to overload the network with communication messages coming from and to the smart home system. Embedding the virtual object management functionality on physical objects solves these problems. However, it can only be implemented on objects with high processing capabilities. Moreover, communication interfaces are not common to all objects, but they depend on the communication protocols used by them.

A trade off between these two solutions is represented by the introduction of a *middleware*, which should act as a gateway among physical objects and the cloud, implementing all the communication interfaces that are needed. Furthermore, it should be able to distinguish when data should be immediately delivered to the cloud, or it is more convenient to process data locally and send only low amounts of aggregated data to the cloud [72]. As it is easy to infer, this is the most complex solution, which requires high self-management capabilities. Optimisation mechanisms need to be taken into account to enhance the use of the resources available, while contemporarily satisfying quality requirements [73].

V. LESSONS LEARNED AND CURRENT CHALLENGES

Keeping in mind the initial definition of virtual object as a *digital representation of a real word object*, we have reviewed the milestones that led to its evolution and have analysed its basic functionalities. Even if we are probably just at the midway of its advancement, as clearly arising from this review, we can enhance its definition as follow:

> *a virtual object is a digital representation, semantically enriched, of a real world object (human or lifeless, static or mobile, solid or intangible), which is able to acquire, analyse and interpret information about its context, to augment the potentialities of the associated services for the benefits of the quality of life of humans as final consumer of the real world data.*

Additionally, we can summarise the lessons learned and the current challenges as follows:

1) several independent works and projects make use of the concept of virtual object, however there is *no common association between virtual and real object*. It is still unclear if the virtual object should match all the services of a single real object or only a subset of them.

Let us consider the case of a car, as shown in Figure 4: when cars are exhibited at an automobile manufacturing company, there is no need to associate a virtual object to every single car, but it could be associated to all the cars for monitoring reasons [19][20], then providing a many-to-one association between real and virtual objects and reducing the resource consumption of the system. When a customer buys a car, she can use it [21]:i) as a smart meter; ii) to register the car to the insurance company; iii) to obtain remote maintenance from the automobile manufacturer database. For every service offered by the same physical object, i.e. the car, a virtual object is created (one-to-many association). Otherwise, she can also decide to register a single virtual object, which interfaces with several services offered by the car, e.g. which handles the car insurance and smart meter applications [23], and re-use another virtual object for other purposes, such as the remote maintenance, to handle both her car and her motorcycle, then leading to a many-to-many association between real and virtual object.

2) It is important to find a *tradeoff between the number of replicas of the same information and their reusability*. On the one hand, the creation of a virtual object for each service helps the service discovery when a particular service is needed by an application. On the other hand, creating a single virtual object for each real world object allows the same information to be available for different services and then lessens the memory consumption. Indeed, a many-to-many association between real and virtual objects can provide a higher degree of freedom in the design of the virtualisation layer; the specific choice is related to many factors and will depend on the role that the virtual object will play in the coming years: as an endpoint to find information useful for applications, creating a virtual object for every service could be the best solution; with the advent of the Information Centric Networking (ICN) approach, the service discovery operations will be content-centric instead of location-centric, so the virtual objects will act only as gateways to access the virtual world and then a single virtual object for each real entity could be enough.

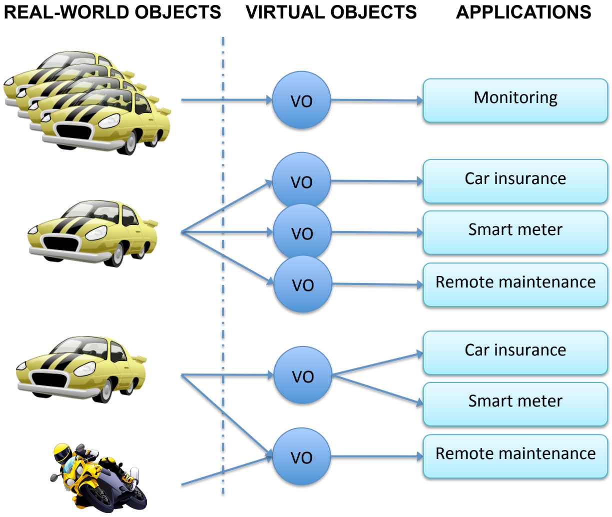

Fig. 4. Equivalence between real-word and virtual objects

3) The role of the virtualisation layer differs from architecture to architecture; however *some functionalities can be considered basic* and are present in most of them. As we have seen in Section IV-B, this applies to the virtualisation enhanced functionalities as well as to the use of: *semantic descriptions* to represent an object in the virtual world; *context awareness* to absorb all incoming data from external objects and performs calculations and transformations on them. In the following years, we foresee that virtual objects will be able to manage in a *cognitive way* the accumulated data in order to make decisions and to act upon the physical devices.

4) *Interoperability* is one of the main concerns regarding IoT middleware solutions. Each architecture that has been investigated in this paper proposes a different virtual object model, described with different semantic languages. The consequence is that virtual objects belonging to different architectures are not able to communicate, and thus they cannot cooperate. Interoperability among the different IoT architectures heavily depends on the available APIs and their usability. Compliance of different architectures elements to a common reference model needs to be provided, in order to enable their coexistence. To overcome semantic ambiguity, we foresee two possible solutions: on the one hand, the development of comprehensive shared information models can facilitate semantic interaction among different virtual object implementations; on the other hand, semantic interaction can be achieved by providing appropriate semantic mediators (translators) at each architecture's end to facilitate the conversion to the virtual object format which each architecture understands.

5) The expected upsurge of the number of objects involved in the IoT [24], is going to exacerbate the *scalability* issue in the next few years. Virtual object life cycle will need to be thoroughly managed, so that virtual objects are deleted as soon

as they are not needed anymore. One of the future challenges related to virtualisation will certainly be virtual object *garbage collection*.

6) The creation of an effective Internet of (every)Things will pass through the definition of an augmented virtual object; it will encompass the capability to autonomously and adaptively interact with the surrounding environment, in order to dynamically deploy applications for the benefit of the humans, so as to improve their quality of life. Which principles and rules should govern the virtual objects behaviour is still to be understood. One of the proposed approaches is that of giving them a behaviour similar to that of the humans in the real world, which, with more or less complex social rules, undertake effective interactions in a scalable way. This so called *Social IoT* is expected to bring to a social network of virtual objects that exchange data and services following the friendships among them in a scalable and trustworthy way [74]. Other solutions propose to follow the principles of the mankind *neural system* to drive the interactions, as it is the most effective solution proposed by the nature to support intensive communications in complex networks [75]. The evolution and possible use cases that will be engendered by applications of these solutions are still unknown, but it is straightforward that they can be easily enabled by virtual objects, thanks to capability of cope with complex networks.

VI. CONCLUSIONS

In this paper, a comprehensive study of the role of virtual object proposed in the literature for the IoT is presented. We first surveyed the milestones that led to the various definitions and characteristics of virtual object. We then investigated the reasons why the use of virtual objects is becoming so strategic in the development of new IoT applications. At this purpose, we analysed the basic IoT issues, and the functionalities associated to virtual objects that have been conceived to solve these issues. The existing architectures that make use of virtual objects have been surveyed and compared.

What we learnt from this study is that the virtual object helps addressing IoT challenges but, even if the different implementations share similar functionalities, at the moment there is no standard format or any recommendation to regulate their usage. We finally discuss some still open issues and emerging trends that will pave the way for future researches.

ACKNOWLEDGMENT

This work has been partially supported by the projects SocialMobility, funded by the workprogramme “P.O.R. FESR 2007-2013 Regione Sardegna - Asse VI Competitività 6.2.1 a”, CUP F25C10001420008, and Artemis JU Demanes, Design, Monitoring and Operation of Adaptive Networked Embedded Systems, Grant agreement No. 295372.

REFERENCES

[1] I. Khan, F. Belqasmi, R. Glitho, and N. Crespi, “A multi-layer architecture for wireless sensor network virtualization,” in *Wireless and Mobile Networking Conference (WMNC), 2013 6th Joint IFIP*. IEEE, 2013, pp. 1–4.

[2] L. Sánchez, V. Gutiérrez, J. A. Galache, P. Sotres, J. R. Santana, J. Casanueva, and L. Muñoz, “Smartsantander: Experimentation and service provision in the smart city,” in *Wireless Personal Multimedia Communications (WPMC), 2013 16th International Symposium on*. IEEE, 2013, pp. 1–6.

[3] S. Karpischek, F. Michahelles, F. Resatsch, and E. Fleisch, “Mobile sales assistant-an nfc-based product information system for retailers,” in *Near Field Communication, 2009. NFC’09. First International Workshop on*. IEEE, 2009, pp. 20–23.

[4] S. Bandyopadhyay, M. Sengupta, S. Maiti, and S. Dutta, “Role of middleware for internet of things: A study,” *International Journal of Computer Science and Engineering Survey*, vol. 2, no. 3, pp. 94–105, 2011.

[5] E. Borgia, “The internet of things vision: Key features, applications and open issues,” *Computer Communications*, vol. 54, pp. 1–31, 2014.

[6] D. Miorandi, S. Sicari, F. De Pellegrini, and I. Chlamtac, “Internet of things: Vision, applications and research challenges,” *Ad Hoc Networks*, vol. 10, no. 7, pp. 1497–1516, 2012.

[7] R. P. Goldberg, “Survey of virtual machine research,” *Computer*, vol. 7, no. 6, pp. 34–45, 1974.

[8] ——, “Architecture of virtual machines,” in *Proceedings of the workshop on virtual computer systems*. ACM, 1973, pp. 74–112.

[9] D. R. Jefferson, “Virtual time,” *ACM Transactions on Programming Languages and Systems (TOPLAS)*, vol. 7, no. 3, pp. 404–425, 1985.

[10] K. Li and P. Hudak, “Memory coherence in shared virtual memory systems,” *ACM Transactions on Computer Systems (TOCS)*, vol. 7, no. 4, pp. 321–359, 1989.

[11] S. R. Ellis, “What are virtual environments?” *Computer Graphics and Applications, IEEE*, vol. 14, no. 1, pp. 17–22, 1994.

[12] H. Vogt, “Efficient object identification with passive rfid tags,” in *Pervasive computing*. Springer, 2002, pp. 98–113.

[13] I. Farris, A. Iera, L. Militano, and S. C. Spinella, “Performance evaluation of rfid tag-based ?virtual? communication channels,” in *Communications (ICC), 2014 IEEE International Conference on*. IEEE, 2014, pp. 2897–2902.

[14] R. Barrett and P. P. Maglio, “Informative things: How to attach information to the real world,” in *Proceedings of the 11th annual ACM symposium on User interface software and technology*. ACM, 1998, pp. 81–88.

[15] R. Want, K. P. Fishkin, A. Gujar, and B. L. Harrison, “Bridging physical and virtual worlds with electronic tags,” in *CHI*, vol. 99, 1999, pp. 370–377.

[16] T. Kindberg, J. Barton, J. Morgan, G. Becker, D. Caswell, P. Debaty, G. Gopal, M. Frid, V. Krishnan, H. Morris *et al.*, “People, places, things: Web presence for the real world,” *Mobile Networks and Applications*, vol. 7, no. 5, pp. 365–376, 2002.

[17] M. Langheinrich, F. Mattern, K. Römer, and H. Vogt, “First steps towards an event-based infrastructure for smart things,” in *Ubiquitous Computing Workshop (PACT 2000)*, 2000, pp. 1–13.

[18] K. Römer, T. Schoch, F. Mattern, and T. Dübendorfer, “Smart identification frameworks for ubiquitous computing applications,” *Wireless Networks*, vol. 10, no. 6, pp. 689–700, 2004.

[19] CONVERGENCE, “Convergence,” 2010. [Online]. Available: http://www.ict-convergence.eu/

[20] SENSEI, “Integrating the physical with the digital world of the network of the future,” 2008. [Online]. Available: http://www.ict-sensei.org/

[21] IoT-A, “Internet of things - architecture,” 2010. [Online]. Available: http://www.iot-a.eu/

[22] COMPOSE, “Collaborative open market to place objects at your service,” 2012. [Online]. Available: http://www.compose-project.eu/

[23] iCore, “Empowering iot through cognitive technologies,” 2011–2014, funded by the European Commission under the 7th Framework Programme. [Online]. Available: http://www.iot-icore.eu/

[24] L. Atzori, A. Iera, and G. Morabito, “The internet of things: A survey,” *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.

[25] M. Zorzi, A. Gluhak, S. Lange, and A. Bassi, “From today’s intranet of things to a future internet of things: a wireless-and mobility-related view,” *Wireless Communications, IEEE*, vol. 17, no. 6, pp. 44–51, 2010.

[26] J. Zheng and M. J. Lee, “Will ieee 802.15. 4 make ubiquitous networking a reality?: a discussion on a potential low power, low bit rate standard,” *Communications Magazine, IEEE*, vol. 42, no. 6, pp. 140–146, 2004.

[27] D. Saha and A. Mukherjee, “Pervasive computing: a paradigm for the 21st century,” *Computer*, vol. 36, no. 3, pp. 25–31, 2003.

[28] F. Mattern and C. Floerkemeier, “From the internet of computers to the internet of things,” in *From active data management to event-based systems and more*, K. I. PabloGuerrero, Ed. Springer, 2010, pp. 242–259.

[29] D. J. Cook and S. K. Das, "Pervasive computing at scale: Transforming the state of the art," *Pervasive and Mobile Computing*, vol. 8, no. 1, pp. 22–35, 2012.

[30] J. Postel and J. Reynolds, "Internet official protocol standards," Internet Requests for Comments, RFC, 1998.

[31] S. Weber and L. Cheng, "A survey of anycast in ipv6 networks," *Communications Magazine, IEEE*, vol. 42, no. 1, pp. 127–132, 2004.

[32] L. Mainetti, L. Patrono, and A. Vilei, "Evolution of wireless sensor networks towards the internet of things: A survey," in *Software, Telecommunications and Computer Networks (SoftCOM), 2011 19th International Conference on*. IEEE, 2011, pp. 1–6.

[33] U. Hunkeler, H. L. Truong, and A. Stanford-Clark, "Mqtt-sa publish/subscribe protocol for wireless sensor networks," in *Communication Systems Software and Middleware and Workshops, 2008. COMSWARE 2008. 3rd International Conference on*. IEEE, 2008, pp. 791–798.

[34] Z. Shelby, K. Hartke, C. Bormann, and B. Frank, "Constrained application protocol (coap), draft-ietf-core-coap-13," *Orlando: The Internet Engineering Task Force–IETF, Dec*, 2012.

[35] K. Aberer, M. Hauswirth, and A. Salehi, "Infrastructure for data processing in large-scale interconnected sensor networks," in *Mobile Data Management, 2007 International Conference on*. IEEE, 2007, pp. 198–205.

[36] D. Guinard, V. Trifa, S. Karnouskos, P. Spiess, and D. Savio, "Interacting with the soa-based internet of things: Discovery, query, selection, and on-demand provisioning of web services," *Services Computing, IEEE Transactions on*, vol. 3, no. 3, pp. 223–235, 2010.

[37] S. Evdokimov, B. Fabian, S. Kunz, and N. Schoenemann, "Comparison of discovery service architectures for the internet of things," in *Sensor Networks, Ubiquitous, and Trustworthy Computing (SUTC), 2010 IEEE International Conference on*. IEEE, 2010, pp. 237–244.

[38] A. Sehgal, V. Perelman, S. Kuryla, and J. Schonwalder, "Management of resource constrained devices in the internet of things," *Communications Magazine, IEEE*, vol. 50, no. 12, pp. 144–149, 2012.

[39] V. Pilloni, P. Navaratnam, S. Vural, L. Atzori, and R. Tafazolli, "TAN: a distributed algorithm for dynamic task assignment in wsns," *Sensors Journal, IEEE*, vol. 14, no. 4, pp. 1266–1279, 2014.

[40] M. Kovatsch, S. Mayer, and B. Ostermaier, "Moving application logic from the firmware to the cloud: Towards the thin server architecture for the internet of things," in *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2012 Sixth International Conference on*. IEEE, 2012, pp. 751–756.

[41] F. Bonomi, R. Milito, J. Zhu, and S. Addepalli, "Fog computing and its role in the internet of things," in *Proceedings of the first edition of the MCC workshop on Mobile cloud computing*. ACM, 2012, pp. 13–16.

[42] E. Crawley, H. Sandick, R. Nair, and B. Rajagopalan, "A framework for qos-based routing in the internet," Internet Requests for Comments, RFC, 1998.

[43] I. Farris, L. Militano, A. Iera, A. Molinaro, and S. C. Spinella, "Tag-based cooperative data gathering and energy recharging in wide area rfid sensor networks," *Ad Hoc Networks*, vol. 36, pp. 214–228, 2015.

[44] J. Kishigami, "The role of qoe on iptv services style," in *Multimedia, 2007. ISM 2007. Ninth IEEE International Symposium on*. IEEE, 2007, pp. 11–13.

[45] S. Haller, S. Karnouskos, and C. Schroth, *The internet of things in an enterprise context*. Springer, 2009.

[46] H. El-Sayed, A. Mellouk, L. George, and S. Zeadally, "Quality of service models for heterogeneous networks: overview and challenges," *annals of telecommunications-annales des télécommunications*, vol. 63, no. 11-12, pp. 639–668, 2008.

[47] I. F. Akyildiz, J. Xie, and S. Mohanty, "A survey of mobility management in next-generation all-ip-based wireless systems," *Wireless Communications, IEEE*, vol. 11, no. 4, pp. 16–28, 2004.

[48] D. Saha, A. Mukherjee, I. S. Misra, and M. Chakraborty, "Mobility support in ip: a survey of related protocols," *Network, IEEE*, vol. 18, no. 6, pp. 34–40, 2004.

[49] B. Fabian and O. Gunther, "Distributed ons and its impact on privacy," in *Communications, 2007. ICC'07. IEEE International Conference on*. IEEE, 2007, pp. 1223–1228.

[50] R. H. Weber, "Internet of things–new security and privacy challenges," *Computer Law & Security Review*, vol. 26, no. 1, pp. 23–30, 2010.

[51] N. M. K. Chowdhury and R. Boutaba, "Network virtualization: state of the art and research challenges," *Communications Magazine, IEEE*, vol. 47, no. 7, pp. 20–26, 2009.

[52] N. Niebert, S. Baucke, I. El-Khayat, M. Johnsson, B. Ohlman, H. Abramowicz, K. Wuenstel, H. Woesner, J. Quittek, and L. M. Correia, "The way 4ward to the creation of a future internet," in *Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on*. IEEE, 2008, pp. 1–5.

[53] N. Chowdhury and R. Boutaba, "A survey of network virtualization," *Computer Networks*, vol. 54, no. 5, pp. 862–876, 2010.

[54] C. Perera, A. Zaslavsky, P. Christen, and D. Georgakopoulos, "Context aware computing for the internet of things: A survey," *Communications Surveys & Tutorials, IEEE*, vol. 16, no. 1, pp. 414–454, 2014.

[55] iCore, "D3.1: Virtual object requirements and dependencies," 2012, funded by the European Commission under the 7th Framework Programme.

[56] M. Compton, P. Barnaghi, L. Bermudez, R. GarcíA-Castro, O. Corcho, S. Cox, J. Graybeal, M. Hauswirth, C. Henson, A. Herzog *et al.*, "The ssn ontology of the w3c semantic sensor network incubator group," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 17, pp. 25–32, 2012.

[57] M. Botts, G. Percivall, C. Reed, and J. Davidson, "Ogc® sensor web enablement: Overview and high level architecture," in *GeoSensor networks*, A. Stefanidis and S. Nittel, Eds. Springer, 2008, pp. 175–190.

[58] iCore, "D3.2: Real object awareness and association," 2013, funded by the European Commission under the 7th Framework Programme.

[59] D. Preuveneers and Y. Berbers, "Internet of things: A context-awareness perspective," *The Internet of Things: From RFID to the Next-Generation Pervasive Networked Systems*, pp. 287–307, 2008.

[60] D. Kelaidonis, A. Somov, V. Foteinos, G. Poulios, V. Stavroulaki, P. Vlacheas, P. Demestichas, A. Baranov, A. R. Biswas, and R. Giaffreda, "Virtualization and cognitive management of real world objects in the internet of things," in *Green Computing and Communications (GreenCom), 2012 IEEE International Conference on*. IEEE, 2012, pp. 187–194.

[61] V. Foteinos, D. Kelaidonis, G. Poulios, P. Vlacheas, V. Stavroulaki, and P. Demestichas, "Cognitive management for the internet of things: A framework for enabling autonomous applications," *Vehicular Technology Magazine, IEEE*, vol. 8, no. 4, pp. 90–99, 2013.

[62] D. Bandyopadhyay and J. Sen, "Internet of things: Applications and challenges in technology and standardization," *Wireless Personal Communications*, vol. 58, no. 1, pp. 49–69, 2011.

[63] J. Bai, "Feasibility analysis of big log data real time search based on hbase and elasticsearch," in *Natural Computation (ICNC), 2013 Ninth International Conference on*. IEEE, 2013, pp. 1166–1170.

[64] G. Di Modica, O. Tomarchio, and L. Vita, "Resource and service discovery in soas: A p 2 p oriented semantic approach," *International Journal of Applied Mathematics and Computer Science*, vol. 21, no. 2, pp. 285–294, 2011.

[65] V. Terziyan, O. Kaykova, and D. Zhovtobryukh, "Ubiroad: Semantic middleware for context-aware smart road environments," in *Internet and Web Applications and Services (ICIW), 2010 Fifth International Conference on*. IEEE, 2010, pp. 295–302.

[66] ETSI, "European telecommunications standards institute," 2015, certified ISO 9001:2008. [Online]. Available: http://www.etsi.org/ralacucc

[67] oneM2M, "Standard for m2m and the internet of things," 2012. [Online]. Available: http://www.onem2m.org/

[68] FI-WARE, "Core platform of the future internet," 2011–2014, co-founded by the European Commission under the 7th Framework Programme. [Online]. Available: http://www.fi-ware.org/

[69] M. Nitti, L. Atzori, and I. P. Cvijikj, "Network navigability in the social internet of things," in *Internet of Things (WF-IoT), 2014 IEEE World Forum on*. IEEE, 2014, pp. 405–410.

[70] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[71] P. Vlacheas, R. Giaffreda, V. Stavroulaki, D. Kelaidonis, V. Foteinos, G. Poulios, P. Demestichas, A. Somov, A. R. Biswas, and K. Moessner, "Enabling smart cities through a cognitive management framework for the internet of things," *Communications Magazine, IEEE*, vol. 51, no. 6, 2013.

[72] A. Manzalini, R. Minerva, F. Callegati, W. Cerroni, and A. Campi, "Clouds of virtual machines in edge networks," *Communications Magazine, IEEE*, vol. 51, no. 7, pp. 63–70, 2013.

[73] G. Colistra, V. Pilloni, and L. Atzori, "The problem of task allocation in the internet of things and the consensus-based approach," *Computer Networks*, vol. 73, pp. 98–111, 2014.

[74] L. Atzori, A. Iera, G. Morabito, and M. Nitti, "The social internet of things (siot)–when social networks meet the internet of things: Concept, architecture and network characterization," *Computer Networks*, vol. 56, pp. 214–228, 2012.

[75] H. Ning and Z. Wang, “Future internet of things architecture: like mankind neural system or social organization framework?” *Communications Letters, IEEE*, vol. 15, no. 4, pp. 461–463, 2011.