The Interoperability of Things:

Interoperable solutions as an enabler for IoT and Web 3.0

George Hatzivasilis, Ioannis Askoxylakis, George Alexandris
Institute of Computer Science
Foundation for Research and Technology–Hellas (FORTH)
Heraklion, Crete, Greece
hatzivas@ics.forth.gr, asko@ics.forth.gr, alexandris@ics.forth.gr

Darko Anicic, Arne Bröring, Vivek Kulkarni
Siemens AG, Corporate Technology
Siemens
Munich, Germany
darko.anicic@siemens.com, arne.broering@siemens.com, vivekkulkarni@siemens.com

Konstantinos Fysarakis, George Spanoudakis
Sphynx Technology Solutions AG
Zug, Switzerland
fysarakis@sphynx.ch, spanoudakis@sphynx.ch

***Abstract*—This paper presents an overview of the interoperability concepts along with the challenges for the IoT domain and the upcoming Web 3.0. We identify four levels of interoperability and the relevant solutions for accomplishing vertical and horizontal compatibility between the various layers of a modern IoT ecosystem, referred to as: technological, syntactic, semantic, and organizational interoperability. The goal is to achieve cross-domain interaction and facilitate the proper usage and management of the provided IoT services and applications. An interoperability framework is also proposed where the involved system components can cooperate and offer the seamless operation from the device to the backend framework. This by-design end-to-end interoperation enables the interplay of several complex service composition settings and the management of the system via patterns. The overall proposal is adopted by the EU funded project SEMIoTICS as an enabler towards the IoT and Web 3.0, even when products from different vendors are utilized.**

Keywords—IoT; Web 3.0; interoperability; by-design; end-to-end; semantic; syntactic; multimode radio; multiprotocol proxy; semantic broker;

I. INTRODUCTION

Interoperability is the ability of a system to work with or use the components of another system. It is easy enough to achieve integration of different systems within the same domain or between different implementations within the stack of a specific software vendor [1]. In the current Internet-of-Things (IoT) ecosystems, the various devices and applications are installed and operate in their own platforms and clouds, but without adequate compatibility with products from different brands [2]-[7]. For example, a smart watch developed in Android cannot interact with a smart bulb without the relevant proprietary gated application provided by the same vendor. Thus, islands of IoT functionality are established that lead towards a vertically-oriented '*Intranet*-of-Things' rather than the 'Internet-of-Things'. To take advantage of the full potential of the IoT vision, we need standards to enable the horizontal and vertical communication, operation, and programming across devices and platforms, regardless their model or manufacturer. Thus, from bottom-up, four levels of interoperability emerge:

- **Technological:** includes the seamless operation and cooperation of heterogeneous devices that utilize different communication protocols on the transmission layer (e.g. WiFi, ZigBee, 802.15.4).
- **Syntactic:** establishes clearly defined and agreed formats for data, interfaces, and encodings
- **Semantic:** settles commonly agreed information models and ontologies for the used terms that are processed by the interfaces or are included in the exchanged data.
- **Organizational:** cross-domain service integration and orchestration through common semantics and programming interfaces.

Although the boundaries of each level are not strict, we consider in our methodology that technological, syntactic, and semantic interoperability enable horizontal compatibility between the involved technologies and platforms, while vertical operation is accomplished through organizational interoperability. Details regarding these four levels and the relevant state-of-the-art interoperability mechanisms are provided in the forthcoming sections.

In general, it is considered that Web 1.0 is a static 'read-only' setting, Web 2.0 or Social Web is a 'read-write' environment, and Web 3.0 or Semantic Web will

enable the 'read-write-execute' perspective [8]. At first, the users were able only to passively access web pages. Now, they can also create content and interact with sites and other users through forums, social media, etc. Next, the Web will become even more intelligent compared with the current solutions and will additionally permit the composition of more complex functionality and services by the user (e.g. [8], [9], [10]).

IoT is a main enabler of Web 3.0 [11]. The resolution of the abovementioned interoperability issues emerges as one of the most significant obstacles for materializing the concept of the Wisdom-of-Things [12]. The SEMIoTICS project [13] considers all interoperability scopes, but it will focus on the high-level administration of services and the appliance of pattern-based management strategies.

The remaining paper is organized as: Section II presents the state-of-the-art and related work. Section III details the technological interoperability. Section IV describes the syntactic interoperability. Section V refers to semantic interoperability. Section VI analyzes the organizational interoperability solutions. Section VII sketches the offered functionality of the proposed framework that will be utilized by the SEMIoTICS project. Finally, Section VIII concludes.

II. Related Work

Surveys for IoT and Industrial IoT (IIoT) are presented in [14] and [15], respectively, highlighting the state-of-the-art techniques and the main design challenges. Inter-domain interoperability is effectively supported by various solutions while intra-domain cooperation remains incomplete.

Several researchers have studied and resolved interoperability issues for specific system components. *IoT protocols* are detailed in [16]. They offer the main messaging functionality between the various IoT devices and simplify the programming effort for an application. The common choices are the Constrained Application Protocol (CoAP), eXtensible Messaging and Presence Protocol (XMPP), Advanced Message Queuing Protocol (AMQP), MQ Telemetry Transport (MQTT), Data Distribution Service (DDS), and Hybrid Lightweight Protocol (Hy-LP) [16], [17]. *Middleware* solutions are reviewed in [18]. They provide information discovery and orchestration of the underlying system, while promoting the Service oriented Architecture (SoA) of the modern IoT settings with enhanced scalability. Popular solutions include the Devices Profile for Web Services (DPWS), Universal Plug and Play (UPnP), and Open Services Gateway initiative (OSGi) [18], [16]. The key aspect of *Discovery* on the IoT is surveyed in [19]. Four different patterns of discovery have been identified that relate to the search for things around the client, on the network, on a directory, as well as the query for metadata. Technologies in these four respective discovery patterns are for example Bluetooth Low Energy, SSDP or mDNS, the CoRE Resource Directory, and the CoRE Link Format. *Context-aware and semantic approaches* for interoperability are mentioned in [20] and [21], respectively. XML technologies for the Semantic Web are utilized, enabling knowledge extraction, discovery, and use of resources. Thus, general or domain specific ontologies are modelled for different sectors, like e-health and transportation [21], facilitating the machine-to-machine (M2M) interaction. Integration of IoT with *cloud computing* is then deployed [22], easing the overall management of the system and performing Big Data analysis. All these application settings utilize the abovementioned technologies in order to enable device and service discovery and administration, accomplishing the inter-domain interaction [23].

The basic intra-domain interoperable approaches are presented in [23], resembling the case of smart homes. Such techniques tackle the cooperation only at the higher system layers. Semantic Information Brokers (SIB) correlate the different semantics and ontologies of the equipped products and provide a common interpretation of the various system aspects. Then, high-level common programing interfaces provide a uniform manner in developing and maintaining new applications.

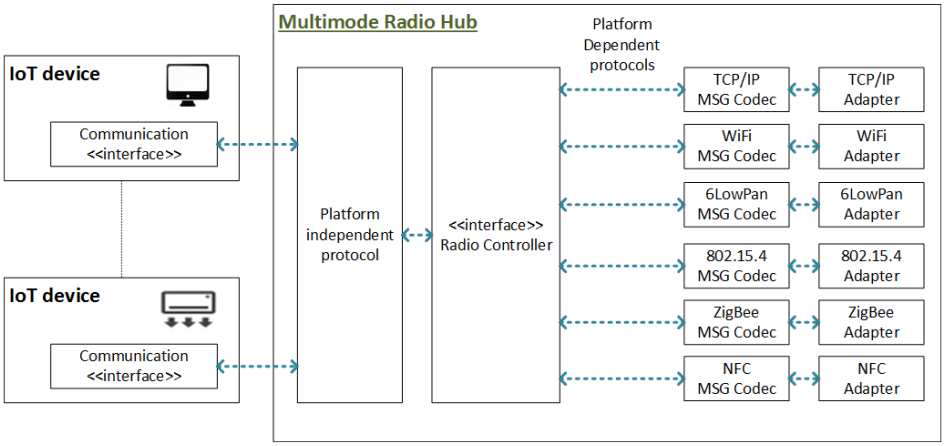

Fig. 1. Multimode Radio Hub.

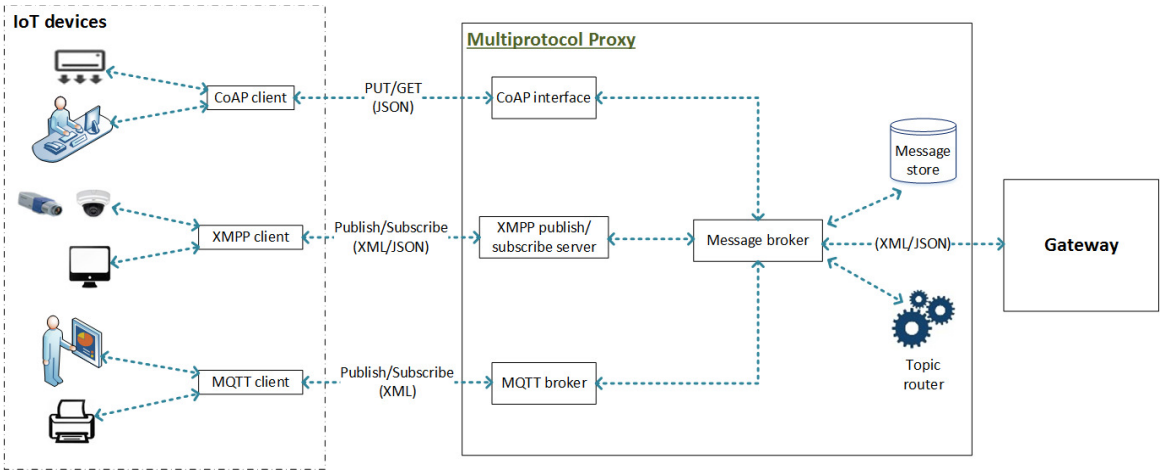

Fig. 2. Multiprotocol Proxy.

This paper reviews solely the interoperability issues in the modern IoT ecosystem. The involved technologies in the different system layers are detailed along with their interconnection setting. The goal is to implement end-to-end and seamless operation from the field to the backend infrastructure while interplaying with products and services from various vendors. According to our knowledge, the proposed framework is the only solution that tackles the interoperation for the four layers in a uniform manner.

III. Technological Interoperability

Technological interoperability still remains a significant barrier in IoT settings as up to 60% of the overall potential value is currently locked due to lack of compatible solutions [24]. Multimode radio equipment constitutes the main technical solution towards the integration of the various heterogeneous devices that utilize different networking and communication means.

Smart phones are a representative example. They deploy a cellular modem, which supports 7 radio interfaces and enable the connection to Global System for Mobile Communications (GSM), Code Division Multiple Access (CDMA), or Long Term Evolution (LTE) networks. Thus, a smart phone can operate with any cellular network and communicate with any other phone.

A similar approach can be followed in various IoT ecosystems, like a smart house. Home hubs, like routers and gateways, implement multimode radios and support various communication technologies (e.g. WiFi, Bluetooth, ZigBee, 802.15.4). These hubs act as bridges and provide the desired interoperable functionality. Thus, modern TVs and thermostats that use WiFi, speakers that communicate with Bluetooth, as well as switches and light bulbs that connect with ZigBee, can interact with each other, providing the user with flexible and convenient ways to interoperate with different smart home ecosystems. For example, a WiFi TV can communicate with ZigBee light bulbs through the home's multimode radio router. This setting can facilitate the installation and synchronization of new devices, and ease the connection to the network. Fig. 1 presents a typical multimode radio hub for IoT that supports 6 different communication interfaces (TCP/IP, WiFi, 6LowPan, 802.15.4, ZigBee, and NFC).

Once the devices are connected, most of the required interoperability functionality can be implemented in software. For instance, ZigBee can be developed in a networking stack if the devices support the 802.15.4 technology. Software solutions ease manufacturers to update their products, fix bugs, and add new features without requiring to redesign the underlying hardware. This capability addresses diversity and fragmentation, and can also reduce replacement and management costs.

However, security issues may raise. Deploying multiple wireless technologies in a device can potentially expose more attack points where malicious entities could inject unauthorized code and sniff network traffic. Hardware security protection mechanisms, such as cryptographic protocols or secure boot and trusted environment execution, can safeguard the system and counter such attacks (e.g. [25], [26], [27], [28]).

IV. Syntactic Interoperability

IoT vendors utilize standardized and widely used technologies and platforms in order to increase the acceptance of their products. Common solutions include the messaging protocols CoAP, XMPP, AMQP, MQTT, DDS, and Hy-LP, as well as the platforms DPWS, UPnP, and OSGi [17].

However, these solutions offer only inter-domain compatibility and they usually act as closed silos with narrow application focus, imposing specific data formats and interfaces. Mechanisms for resolving these issues and achieving horizontal interoperability include gateway proxies for the messaging protocols.

The main setting is suggested in [29]. The proposal automatically converts messages from one messaging protocol to the compatible format of another protocol. The functionality is offered among RESTful HTTP, CoAP, XMPP, MQTT, and DDS, and can be easily extended in order to support the rest of the popular protocols [17].

Fig. 2 depicts the deployment setting for the main CoAP, XMPP, and MQTT, along with the related data

formats of each protocol. CoAP operates similarly with HTTP. XMPP requires a resource server, while MQTT imposes a central broker that administrates the communication. The messages from the three distinct settings are parsed through a message broker that implements the core functionality of the multiprotocol proxy, translating the messages from one protocol to the other and managing the communication flow. The messages can be also maintained locally with a topic router, easing the discovery process of the communicating system components.

Each platform utilizes specific message protocols. Through a multiprotocol proxy they can also expand their functionality and interact with devices that support different protocols.

All these methods provide the main inter-domain interoperability features at the syntactic level. The devices can communicate seamlessly, but until this point, they cannot understand each other. Thus, additional mechanisms are required to represent and explicate the information semantics in a machine-interpretable format, as described in the following section.

V. SEMANTIC INTEROPERABILITY

Today, the semantic technologies that enable and facilitate the interoperability in web services are commonly adapted in the IoT domain. This includes widely-used and well-studied XML schemes, like the Resource Description Framework (RDF), RDF Schema (RDFS), and Web Ontology Language (OWL) for ontologies, and the Web Services Description Language (WSDL) for services. Such technologies offer common description and representation of data and services, characterize things and their capabilities, and deal with the semantic annotation, resource discovery, access management, and knowledge extraction in a machine-readable and interoperable manner.

Towards these goals, the most notable effort in the IoT field is the Semantic Sensor Network (SSN) ontology and Sensor Observation Sampling Actuator (SOSA) ontology by the World Wide Web Consortium (W3C) community [30]. The SOSA/SSN ontologies model sensors, actuator, samplers as well as their observation, actuation, and sampling activities. The ontologies capture the sensor and actuator capabilities, usage environment, performance, and enabling contextual data discovery. This also constitutes the standardized ontologies for the semantic sensor networks. The cooperation of SSN and SOSA offers different scope and degrees of axiomatization that enable a wide range of application scenarios towards the Web of Things [11].

More specifically, the SSN ontology is a suite of general purpose ontologies. It embodies the following 10 conceptual modules: 1) Device, 2) Process, 3) Data, 4) System, 5) Deployment, 6) PlatformSite, 7) SSOPlatform, 8) OperatingRestriction, 9) ContraintBlock, and 10) MeasuringCapability. The modules consist of 41 concepts and 39 object properties.

The general approach regarding the semantic interoperability that is followed by several IoT initiatives, like the European Union (EU) funded projects Open source solution for the Internet of Things (OpenIoT) [31] and INTER-IoT [21], is the usage of the SSN/SOSA ontologies as the semantic base. The ontologies are then extended with the additional required concepts to model the targeted application scenarios. Such concepts usually include relevant standards and ontologies for specific application areas, like e-health [32], and less often extensions at the sensor level (as the relevant SSN/SOSA information is quite complete). Other similar and popular IoT ontologies include the Smart Appliance REFerence (SAREF) [33] and the MyOntoSens [34].

More recently, W3C has launched a working group called Web of Things (WoT) [35] with the goal to counter the fragmentation of the IoT and enable interoperable IoT devices and services, thereby reducing the costs of their development. A notable feature of W3C WoT approach is Thing Description (TD) [36], used to describe the metadata and interfaces of (physical) Things in a machine interpretable format. TD has been built upon W3C's extensive work on RDF [37], JSON-LD [38], and Linked Data [39]. TD defines a domain agnostic vocabulary to describe any Thing in terms of its properties, events and actions. In order to give a semantic meaning to a set of properties, events and actions for a particular Thing various semantic models can be used, e.g., SOSA/SSN, SAREF etc. One notable community effort to create a semantic schema for IoT applications is iot.schema.org [40]. It is an extension of well-known schema.org for IoT. iot.schema.org introduces a semantic model to describe a capability of a Thing. A capability is the set of affordances needed to interact with a single function of a connected Thing, e.g. an on/off switch capability. Together, W3C WoT and iot.schema.org, provide a semantic interoperability layer that enables software to interact with the physical world. The interaction is abstracted in such a way that it simplifies the development of applications across diverse domains and IoT ecosystems.

VI. ORGANIZATIONAL INTEROPERABILITY

The common interpretation of semantic information in a globally shared ontology could be quite useful. However, this is not always the case. Although several local systems may utilize popular or standardized ontologies, eventually they extend them and establish their own semantics and interfaces. The direct interaction between these systems is not feasible. The use of Semantic Information Brokers (SIBs) is proposed in [41], which correlate the required information and enable the interoperability of systems with different semantics or cross-domain interaction.

Moreover, a common and generic Application Programming Interface (API) is established by the EU funded project BIG IoT [42] between the different IoT middleware platforms. The API and the related information models are determined in cooperation with the Web of Things Interest Group at the W3C, enhancing the supported standards of this community. The API eases the development of software services and applications for different platforms according to a well-defined architecture [43].

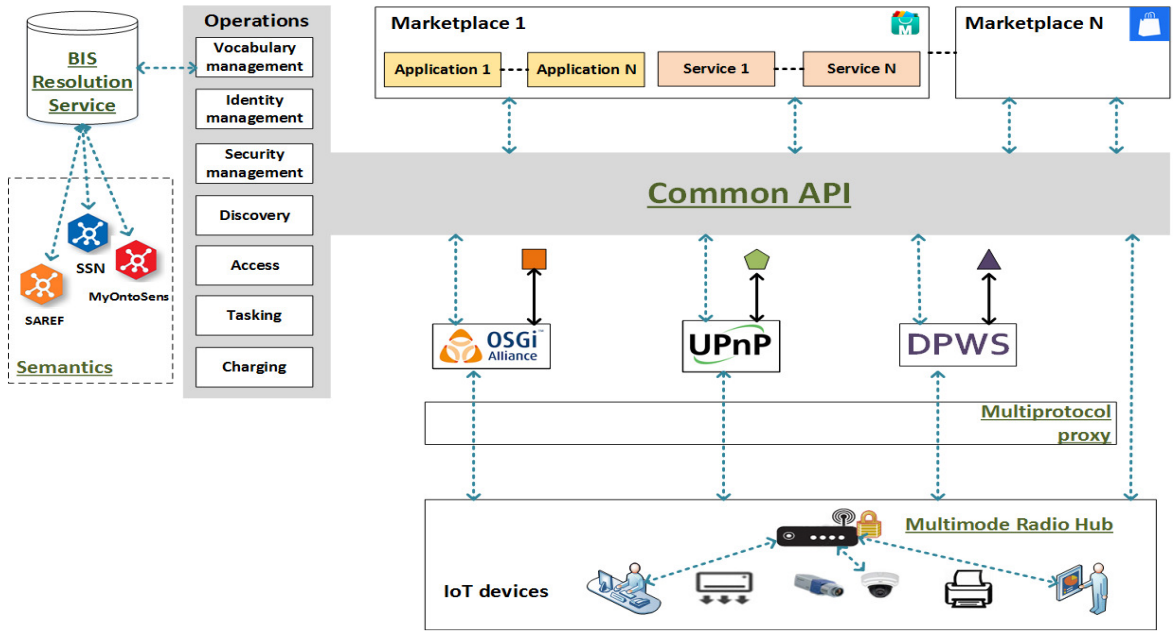

Fig. 3. The SEMIoTICS interoperability framework (adjusted and extended from [42]).

Thus, the cooperation of an SIB with the abovementioned common API permits complex service composition and added value applications. Such APIs provide well-defined functionalities that can also implement interoperability on device-, fog-, and cloud-level. The main functionalities include: i) identity management and registration to resources, ii) resource discovery based on user-defined criteria, iii) access to data and meta-data (e.g. publish/subscribe of data streams), iv) command forwarding to things, v) vocabulary management of semantic information, vi) security management (key management, authentication, authorization, etc.), and vii) charging and billing management for using the provided assets. Being able to monetize IoT resources is also key to establishing business models for a flourishing ecosystem [44], which is crucial for organizational interoperability.

The manufacturer's resources are advertised on the marketplace. Clients can discover the offered applications and gain access to them. In the near future, it is expected that there will be multiple marketplaces for IoT products [42]. The marketplaces could be set for each application domain (e-health, smart home, etc.) or there could be multiple marketplaces for a single domain but set by different vendors.

As the developers comply with the defined interfaces, the marketplaces enhance the organizational interoperability. In cooperation with SIBs, the cross-domain IoT vision is further fostered. Thus, a modern IoT application can utilize services from different manufacturers and implement horizontal interoperable solutions that also utilize the three vertical interoperability layers, accomplishing seamless operation from the device end to the backend infrastructure.

VII. SEMIoTICS Interoperability Solution

The SEMIoTICS proposal utilizes the aforementioned state-of-the-art mechanisms for the four interoperability levels. It implements by-design cross-domain operation and interaction, and enables the interplay with all layers. The overall framework is depicted in Fig. 3 (similarly with [42]).

Once these mechanisms have been placed, we need a systematic way to model the ecosystem's features and administrate the provided functionality. Pattern-driven techniques are utilized for these purposes, such as [42] and [45]. The pattern-driven framework is built upon existing IoT platforms and guarantees the secure and dependable actuation. A semi-automatic behavior is supported that evaluates the integration of the various system components and orchestrates the interoperable operations. Thus, five core access settings are enabled under this framework, as detailed below.

With the **cross platform** pattern (as described in [42]), applications or services access resources from multiple platforms though the common interfaces. Thus, an application, like an air quality monitor, can discover platforms (of the same or different vendor) that process related data and support the same interfaces and data formats. The requested information is then collected by the various compatible platforms enabling the required functionality.

The **Platform-scale independence** pattern [42] integrates the resources from platforms at different scale. Cloud/server level platforms can host high volumes of data from a vast amount of devices. Fog level platforms interact with nearby devices in the field and maintain information in a constraint spatial scope. The device level platforms have direct communication with the things, managing small amounts of data. Through SEMIOTIC, an application can uniformly aggregate information for the

different scale platforms (e.g. collect air quality values for a specific area via cloud or raw data via a platform at the fog/device level).

Platform independence pattern [42] refers to distinct platforms that implement the same functionality, like an IoT parking service in different cities. The platforms may utilize different equipment and techniques in order to discover a free parking spot (e.g. via radar-based sensors or smart cameras on the street lamps). Hench, a single driver application can interoperate with both platforms in a uniform manner without requiring any changes.

The **cross application domain** pattern [42] setting extends the previous ones, with applications or services accessing now information not only from several platforms, but also from platforms that process data from different application domains or verticals. Therefore, the application can gather data for the air quality and traffic, in order to propose the least polluted routes to bicyclists.

Higher-level service facades pattern [42] expand the abovementioned platform functionality to high-level services. Henceforth, services can also interact through the common API, acting as facades for IoT platforms and implementing value-added operations. For instance, the air quality monitoring application can interact with a platform that maintains related information with aggregated data. The application can also interact with a service that aggregates data from a second platform, which on the other hand does not possess the computational capabilities to aggregate these pieces of knowledge or maintain long-term assets.

Once the aforementioned settings are deployed, services can be composed and reused while data can be integrated by various platforms. The goal is to achieve dynamic information discovery and orchestration of the underlying system components. An application can interoperate across different domains and platforms. Thus, the user can integrate services dynamically and adapt the available services according to his/hers needs, even during travelling in different cities or countries. All these features contribute towards the fruitful interplay of IoT and Web 3.0, supporting the vision of user-composed intelligent services with complex functionality [9].

TABLE I. SUPPORTED INTEROPERABILITY FEATURES

Feature	SEMIoTICS	Big IoT	OpenIoT	INTER-IoT
Technological interoperability	Yes	No	No	No
Syntactic interoperability	Yes	No	No	No
Semantic interoperability	Yes	Yes	Yes	Yes
Organizational interoperability	Yes	Yes	Yes	No
Pattern-based modelling	Yes	Yes	No	No
Pattern-based semi-automatic management	Yes	No	No	No

Table I summarizes the main interoperability features for four EU funded IoT projects (SEMIoTICS, BIG IoT [42], OpenIoT [31], INTER-IoT [21]). The main efforts for cross-domain operation are focused on the semantics and the high-level programming interfaces. SEMIoTICS advances the current solutions by also resolving the compatibility issues at the lower layers. Moreover, the pattern-driven modelling and management guarantees the correct operation of the system and simplifies the integration process of new components and service settings.

Nevertheless, there remain several unsolved open issues. Automatic charging becomes a fundamental element once the seamless operation is achieved. Security is always a main concern (see [46]). Enforcing authorization and access control of the available features is still a challenging task.

VIII. CONCLUSIONS

The SEMIoTICS project concerns all four levels of interoperability, but the research efforts focus on the systematic modelling and administration of the cross-domain interaction. The main goal is to establish interoperability patterns that will facilitate the modelling and real-time management of the underlying IoT ecosystem. SEMIoTICS will formally analyze the five core interoperability patterns that are suggested by the related BIG IoT project [42]. These patterns cover the main compatibility issues for composing services from inter- to cross-domain topologies.

ACKNOWLEDGMENT

This work has received funding from the European Union Horizon's 2020 research and innovation programme under grant agreement No. 780315 (SEMIoTICS), as well as the Marie Skodowska-Curie Actions (MSCA) Research and Innovation Staff Exchange (RISE), H2020-MSCA-RISE-2017, under grant agreements No. 777855 (CE-IoT) and No. 778229 (Ideal Cities).

REFERENCES

[1] Ganzha, M., Paprzycki, M., Pawlowsji, W., Szmeja, P. and Wasielewska, K., 2016. Semantic technologies for the IoT – an Inter-IoT perspective. 1st International Conference on the Internet-of-Things Design and Implementation (IoTDI), IEEE, Berlin, Germany, pp. 271-276.

[2] Fysarakis, K., Hatzivasilis, G., Papaefstathiou, I. and Manifavas, C., 2016. RtVMF – A secure Real-time Vehicle Management Framework with critical incident response. IEEE Pervasive Computing Magazine (PVC) – Special Issue on Smart Vehicle Spaces, IEEE, vol. 15, issue 1, pp. 22-30.

[3] Hatzivasilis, G., Papaefstathiou, I. and Manifavas, C., 2017. Real-time management of railway CPS. 5th EUROMICRO/IEEE Workshop on Embedded and Cyber-Physical Systems (ECYPS 2017), IEEE, Bar, Montenegro, 11-15 June.

[4] Hatzivasilis, G., Papaefstathiou, I., Plexousakis, D., Manifavas, C. and Papadakis, N., 2017. AmbISPDM: Managing Embedded Systems in Ambient Environment and Disaster Mitigation Planning, Applied Intelligence, Springer, pp. 1-21.

[5] Hatzivasilis, G., Papaefstathiou, I. and Manifavas, C., 2017. SCOTRES: Secure Routing for IoT and CPS, IEEE Internet of Things Journal (IoT-J), IEEE, vol. 4, issue 6, pp. 2129-2141.

[6] Vilalta, R., Mayoral, A., Pubill, D., Casellas, R., Martinez, R., Serra, J., Verikoukis, C. and Munoz, R., 2016. End-to-end SDN orchestration of IoT services using an SDN/NFV-enabled edge node. Optical Fiber Communications Conference and Exhibition (OFC), Anaheim, CA, USA, pp. 1-3.

[7] Serra, J., Pubill, D., Antonopoulos, A. and Verikoukis, C., 2014. Smart HVAC control in IoT: energy consumption minimization with user comfort constraints. The Scientific World Journal, Hindawi, vol. 2014, article ID 161874, pp. 1-11.

[8] Kadyan, S. and Singroha, R., 2014. Web 3.0 in library services: an utilitarian effect. Journal of Informaiton Management, SPLP, vol. 1, no. 2, pp. 159-166.

[9] Murugesan, S., Rossi, G., Wilbanks, L. and Djavanshir, R., 2011. The future of Web apps. IEEE IT Professional, IEEE, vol. 13, issue 5, pp. 12-14.

[10] Scovotti, C. and Jones, S. K., 2011. From Web 2.0 to Web 3.0: implcations for advertizing courses. Journal of Advertising Education, vol. 15, issue 1, pp. 6-15.

[11] Zeng, D., Guo, S. and Cheng, Z., 2011. The Web of Things: a survey. Journal of Communicaitons, Academy Publisher, vol. 6, no. 6, pp. 424-438.

[12] Zhong, N., Ma, J., Huang, R., Liu, J., Yao, Y., Zhang, Y. and Chen J., 2013. Research challenges and perspectives on Wisdom Web of Thigns (W2T). The Journal of Supercomputing, Springer, vol. 64, issue 3, pp. 862-882.

[13] SEMIoTICS, 2018-2020: https://www.semiotics-project.eu

[14] Li, S., Xu, L. D. and Zhao, S., 2015. The internet of things: a survey. Information Systems Frontiers, Springer, vol. 17, issue 2, pp. 243-259.

[15] Xu, L. D., He, W. and Li. S., 2014. Intenet of Things in industries: a survey. IEEE Transactions on Industrial Informatics, IEEE, vol. 10, issue 4, pp. 2233-2243.

[16] Al-Fuqaha, A. et al., 2015. Internet of Things: a survey on enabling technologies, protocols, and applications, IEEE Communication Surveys & Tutorials, IEEE, vol. 17, issue 4, pp. 2347-2376.

[17] Hatzivasilis, G., et al., 2018. The Industrial Internet of Things as an enabler for a Circular Economy Hy-LP: A novel IIoT Protocol, evaluated on a Wind Park's SDN/NFV-enabled 5G Industrial Network. Computer Communications – Special Issue on Energy-aware Design for Sustainable 5G Networks, Elsevier, vol. 119, pp. 127-137.

[18] Razzaque, M. A., Milojevic-Jevric, M., Palade, A. and Clarke, S., 2016. Middleware for Internet of Things: a survey. IEEE Internet of Things Journal (IoT-J), IEEE, vol. 3, issue 1, pp. 70-95.

[19] Bröring, A., Datta, S.K. and Bonnet, C., 2016. A Categorization of Discovery Technologies for the Internet of Things. 6th International Conference on the Internet of Things (IoT 2016), ACM, 7-9 November, Stuttgart, Germany, pp. 131-139.

[20] Perera, C., Zaslavsky, A., Christem, P. and Georgakopoulos, D., 2013. Context aware computing for the Internet of Things: a survey. IEEE Communcations Surveys & Tutorials, IEEE, vol. 16, issue 1, pp. 414-454.

[21] Ganzha, M. et al., 2017. Semantic interoperability in the Internet of Things, as overview from the INTER-IoT perspective. Journal of Network and Computer Applications, Elsevier, vol. 81, issue 1, pp. 111-124.

[22] Botta, A., Donata, W., Persico, V. and Pescape, A., 2016. Integration of Cloud computing and Internet of Things: a survey. Future Generation Computer Systems, Elsevier, vol. 56, issue 1, pp. 684-700.

[23] Korzum, D. G., Balandin,S. I. and Gurtov, A. V., 2013. Deploment of smart spaces in the Internet of Things: overview of design challenges. Internet of Things, Smart Spaces, and Next Generation Networking, Springer, LNCS, vol. 8121, pp. 48-59.

[24] Manyika, J., et al., 2015. Unlocking the potential of the Internet of Things. McKinsey Global Institute Report, McKinsey&Company, June 2015, pp. 1-4.

[25] Lin, J., Yu, W., Zhang, N., Yang, X., Zhang, H. and Zhao, W., 2017. A survey of Internet of Things: architecture, enabling technologies, security and privacy, and applications, IEEE Internet of Things Journal, IEEE, vol. 4, no. 5, pp. 1125-1142.

[26] Hatzivasilis, G., Floros, G., Papaefstathiou, I. and Manifavas, C., 2016. Lightweight Authenticated Encryption for Embedded On-Chip Systems. Information Security Journal: A Global Perspective, Taylor & Francis, vol. 25, issue 4-6, pp. 151-161.

[27] Chen, C., Raj, H., Saroiu, S. and Wolman, A., 2014. cTPM: a cloud TPM for cross-device trusted applications, 11th USENIX Symposium on Networked Systems Design and Implementation (NSDI), 2-4 April, Seattle, WA, USA, pp. 187-201.

[28] Paverd, A. J. and Martin, A. P., 2012. Hardware security for device authentication in the smart grid, International Workshop on Smart Grid Security (SmartGridSec), Springer, LNCS, col. 7823, pp. 72-84.

[29] Al-Fuqaha, A., Khreishah, A., Guizani, M., Rayes, A. and Mohammadi, M., 2015. Toward better horizontal integration among IoT services. IEEE Communications Magazine, IEEE, vol. 53, issue 9, pp. 72-79.

[30] Haller, A., et al., 2018. The SOSA/SSN ontology: a joint WeC and OGC standard specifying the semantics of sensors, observations, actuation, and sampling. Semantic Web, IOS Press, vol. 1-0X, pp. 1-19.

[31] Soldatos, J. et al., 2015. OpenIoT: Open source Internet-of-Things in the Cloud. Interoperability and Open-Source Solutions for the Internet of Things, Springer, LNCS, vol. 9001, pp. 13-25.

[32] Cameron, J. D., Ramaprasad, A. and Syn T., 2015. An ontology of mHealth. 21st Americas Conference on Information Systems (AMCIS), Puerto Rico, pp. 1-11.

[33] Daniele, L., den Hartog, F. and Roes, J., 2015. Created in close interaction with the industry: the smart appliances reference (SAREF) ontology. International Workshop Formal Ontologies Meet Industries (FOMI), Springer, LNBIP, vol. 225, pp. 100-112.

[34] Bajaj, G., et al., 2017. A study of existing ontologies in the IoT-domain, HAL Archives, hal-01556256, pp. 1-24.

[35] Web of Things (WoT): https://www.w3.org/WoT/

[36] Thing Description (TD): https://w3c.github.io/wot-thing-description/

[37] RDF: https://www.w3.org/TR/rdf11-concepts/

[38] JSON-LD: https://www.w3.org/TR/2014/REC-json-ld-20140116/

[39] Linked Data: https://www.w3.org/standards/semanticweb/data

[40] iot.schema.org: https://github.com/iot-schema-collab & http://iotschema.org/

[41] Kiljander, J. et al., 2014. Semantic interoperability architecture for pervasive computing and Internet of Things. IEEE Access, IEEE, vol. 2, pp. 856-873.

[42] Bröring, A., Schmid, S., Schindhelm, C.-K., Kheli, A., Kabisch, S., Kramer, D., Phuoc, D., Mitic, J., Anicic, D. and Teniente, E., 2017. Enabling IoT ecosystems through platform interoperability. IEEE Software, IEEE, vol. 34, issue 1, pp. 54-61.

[43] Schmid, S., Bröring, A., Kramer, D., Kaebisch, S., Zappa, A., Lorenz, M., Wang, Y. and Gioppo, L., 2017. An Architecture for Interoperable IoT Ecosystems. 2nd International Workshop on Interoperability & Open Source Solutions for the Internet of Things (InterOSS-IoT 2016) at the 6th International Conference on the Internet of Things (IoT 2016), 7. November 2016, Stuttgart, Germany, Springer, LNCS., vol. 10218, pp. 39-55.

[44] Schladofsky, W., Mitic, J., Megner, A.P., Simonato, C., Gioppo, L., Leonardos, D. and Bröring, A., 2017. Business Models for Interoperable IoT Ecosystems. 2nd International Workshop on Interoperability & Open Source Solutions for the Internet of Things (InterOSS-IoT 2016) at the 6th International Conference on the Internet of Things (IoT 2016), 7. November 2016, Stuttgart, Germany. Springer, LNCS. Volume 10218, pp. 91-106.

[45] Petroulakis, N., Spanoudakis, G. and Askoxylakis, I., 2016. Patterns for the design of secure and dependable software defined networks, Computer Networks, Elsevier, vol 109, issue 1, pp. 39-49.

[46] Hernandez-Serrano, J., Munoz, J.L., Bröring, A., Esparza, O., Mikkelsen, L., Schwarzott, W., Leon, O. and Zibuschka, J., 2017. On the Road to Secure and Privacy-preserving IoT Ecosystems. 2nd International Workshop on Interoperability & Open Source Solutions for the Internet of Things (InterOSS-IoT 2016) at the 6th International Conference on the Internet of Things (IoT 2016), 7. November 2016, Stuttgart, Germany. Springer, LNCS. Volume 10218, pp. 107-122.