Review of Standard Ontologies for the Web of Things

Wenbin Li
Easy Global Market
Sophia Antipolis, France
wenbin.li@eglobalmark.com

Giuseppe Tropea
University of Rome Tor Vergata
CNIT, Italy
giuseppe.tropea@cnit.it

Ahmed Abid
Easy Global Market
Sophia Antipolis, France
ahmed.abid@eglobalmark.com

Andrea Detti
University of Rome Tor Vergata
CNIT, Italy
andrea.detti@uniroma2it

Franck Le Gall
Easy Global Market
Sophia Antipolis, France
franck.le-gall@eglobalmark.com

***Abstract*—To open isolated Internet of Things (IoT) silos, the Web of Things (WoT) integrates IoT into the Web promoting an architecture with easy access APIs and applications, and ontologies further improve the data interoperability and federation capability of IoT platforms. With the increasing industry adoption of semantic technologies, a number of ontologies has been defined and standardized, covering different layers of WoT. Although the best practice in data management is to use standardized ontologies as much as possible, users without prior knowledge can hardly select the standard that meets the specific needs and context. This paper provides a general review of standardized WoT ontologies and identifies their coverage of different WoT layers. Firstly, we introduce some background on WoT and ontologies; we then present two WoT architectures, an IoT-oriented one and a Web-oriented one, as evaluation criteria for ontology coverage; in the following, we introduce and analyze the representative standardized WoT ontologies based on criteria. Lastly, we summarize our work and shed light on further directions of WoT ontology development.**

***Index Terms*—Internet of Things, Web of Things, ontology, standard, review, semantic Web, device, data, knowledge, composition**

I. INTRODUCTION

Many of existing Internet of Things (IoT) deployments are isolated with vendor specific solutions from devices and data to platforms and applications. To open such closed silos and interconnect different IoT deployments, the Web of Things (WoT) [1] provides IoT with different APIs (e.g., REST, GraphQL, SPARQL) and integrates IoT into the Web architecture, for easy management of data and services varying in scale, scope and abstraction. While WoT improves IoT connectivity and accessibility, ontologies based on semantic Web technologies provide both machine understandable and human readable specifications, that are being largely adopted in the industry for interoperability and federation purposes. Thus, through semantic Web and WoT, IoT is being evolved to be semantically interoperable and hyperconnected in open and linked environments, especially by using dereferenceable URIs instead of arbitrary identifiers and by providing interlinking between fine-grained resources (such as devices, physical entities) and their states and attributes [2].

An ontology [3] generally specifies the concepts and relations of concepts within a knowledge domain. Based on W3C standards, while many initiatives have been taken to develop ontologies for WoT [4], the development and maintenance of an WoT ontology is challenging due to heterogeneity of IoT objects (from electronic devices to physical objects), variety of data models and diversity of services. A number of them has been standardized and has achieved overall good abstraction and coverage, such as the oneM2M Base Ontology [5], the SAREF ontology [6], the ETSI ISG CIM Information Model [7], the W3C WoT Thing Description [8] and the W3C SSN and SOSA [9]. Nevertheless, the concepts and relations defined in such standardized ontologies vary from one to another, each showing a different focus and coverage on WoT architectures. Users without *prior* knowledge face difficulties when choosing suitable ontologies to meet their modeling and management needs. In order to provide a global comparison of standard WoT ontologies, this paper aims at reviewing existing standards, *de facto* standards and domain-independent WoT ontologies on their specifications and support on different WoT layers, so as to facilitate ontology usage and shed light on future ontology development.

The remainder of the paper is organized as follows: section 2 introduces two reference architectures for WoT, which are used as evaluation criteria. Section 3 presents each ontology individually, while section 4 provides the ontology comparison and future directions for ontology development. Section 5 concludes our contributions.

II. WOT REFERENCE ARCHITECTURES AND CRITERIA

To provide a thorough analysis of ontologies, we present two reference WoT architectures that are used as review criteria to evaluate the ontology coverage on the architecture. Accordingly, we name the two criteria as IoT Oriented Criterion (IOC) and Web Oriented Criterion (WOC).

A. IoT Oriented Criterion (IOC)

The first WoT architecture we present as a reference is more IoT-oriented, and we adapt it from previous work in H2020 WISE-IoT project [10], which differentiates the layers and their respective functionalities according to the level of abstraction. This simplified four layers architecture is illustrated in Fig. 1: Physical Layer comprises devices and physical entities with communications networks; Data Layer integrates raw data by means of common abstractions and manages devices and entities; Information Layer provides access to data and functions; Knowledge Layer processes information to elicit higher-level knowledge and provides knowledge management functions, such as proof and trust.

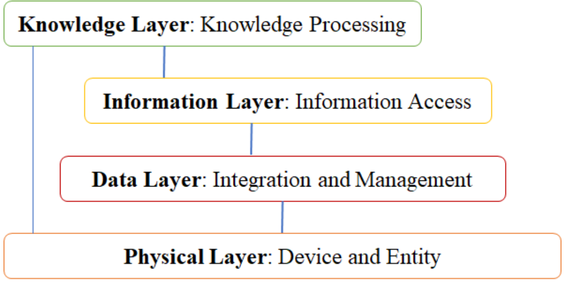

Fig. 1 IoT Oriented Reference Architecture for WoT

For each ontology, we evaluate its coverage of each layer of Fig. 1. More specifically, we review the ontology concepts and relations following four points:

- Physical Layer: definitions of device, entities, physical infrastructure and communication
- Data Layer: definitions of universal data model and device management functions
- Information Layer: definitions of interfaces and operations for data access and device management
- Knowledge Layer: definition of knowledge representation and knowledge management functions such as proof and trust

B. Web Oriented Criterion (WOC)

The second criterion is proposed in [11] as the WoT architecture most oriented to Web and applications. Its objective is to structure the galaxy of Web protocols and tools into a useful framework for connecting any device or object to the Web with data and services. Four layers (i.e., Accessibility, Findability, Sharing and Composition) are defined in this architecture as shown in Fig. 2 (adapted from [12]). Each layer comprises a number of exemplary technologies to achieve the objectives of the layer: Accessibility Layer transforms devices and physical entities to Web resources; Findability Layer enables the automatic search and discovery of Web resources; Sharing Layer enables secure data sharing services; Composition Layer integrates data and services.

Based on this criterion, we also review the ontologies in this paper following four points:

- Accessibility Layer: definitions of APIs, service protocols and description formats
- Findability Layer: definitions of Linked Data concepts, semantic Web and resource discovery components
- Sharing Layer: definitions of security mechanisms and protocols, as well as social networks for social WoT
- Composition Layer: definition of application composition, system integration and mashup platforms

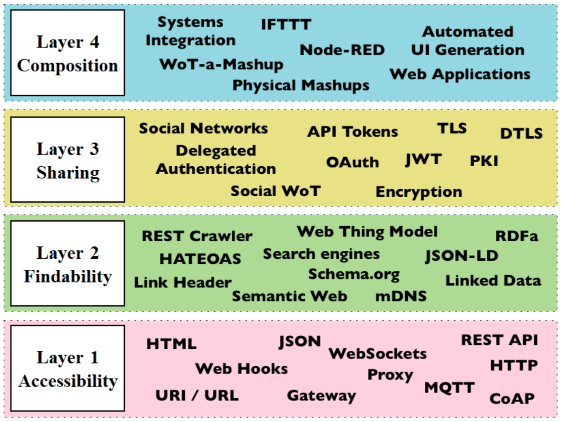

Fig. 2 Web Oriented Reference Architecture for WoT

In the following, we introduce the standard WoT ontologies individually and evaluate their coverage for IOC and WOC.

III. STANDARD WEB OF THINGS ONTOLOGIES

A. oneM2M Base Ontology

oneM2M [13] is a global open standard for Machine to Machine Communications and the IoT, providing a common M2M Service Layer with common service functions to connect and interwork devices. oneM2M specifies a Base Ontology [5] to provide a minimal number of generic concepts and capture the duality of real things and their electronic counterparts, the latter being used to make things and their functionality discoverable, registrable and remotely controllable in the network.

The core concepts of the oneM2M base ontology are depicted in Fig. 3. Due to the space limitations, all figures in the paper introduce only the most important concepts in ontologies, while full descriptions can be found in the corresponding references.

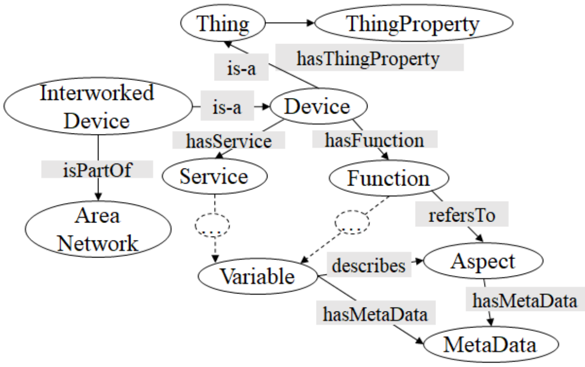

Fig. 3 oneM2M base ontology core concepts

In this base ontology, a *Thing* is whatever a real entity can be identified in the oneM2M system. The *Device*, sub-class of *Thing*, is an entity interacting electronically with its environment, via a network. Similarly, a *Service* is an electronic representation of a *Function* within a network. While a *Function* describes the human understandable meaning of a *Service*, the *Service* describes how such functionality is represented in a communication network, and how it can be discovered and accessed by electronic means. A *ThingProperty* denotes a property of a Thing, which can be observed or influenced by *Devices*, or constitutes static data about a *Thing*. A *ThingProperty* also describes a certain *Aspect* with *MetaData*. For example, the "indoor temperature" describes the Aspect "Temperature" measured by a temperature sensor in "Celsius" unit. The *Aspect* is the real-world attribute that a *Function* relates to, or the quality of a Variable. The *Variable* class instantiates entities that store data changing over time, which is used to describe some real-world *Aspects* with *MetaData*. Being a sub class of *Device*, the *InterworkedDevice* is a *Device* in an *AreaNetwork* that does not support oneM2M interfaces and can only be accessed by communicating through an interworking proxy entity.

The oneM2M base ontology is continuously evolving along with the oneM2M specification release. The current version includes the core concepts of the IOC's Physical Layer, Data Layer and Information Layer, while seeking a simplified specification with a small number of concepts and relations in it. Regarding the WOC, all ontologies in this paper are based on semantic web technologies and thus, to different extents, all of them cover the Findability Layer. Besides, the oneM2M ontology focuses more on machine-to-machine than Web applications, and by defining services and functions it also covers the WOC's Accessibility Layer.

B. SAREF

The SAREF (Smart Appliances REFerence) ontology [6] has been developed and standardized by the European Commission in close cooperation with ETSI (European Telecommunications Standards Institute) to provide a modular and domain-independent semantic layer for smart appliances. During the specification, a number of reference ontologies or taxonomies were analyzed [14], and SAREF is a convergence result of the relevant ontologies for the IoT domain. The core concepts of SAREF are depicted in Fig. 4.

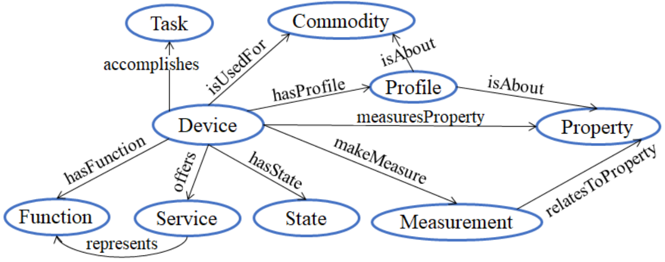

Fig. 4 SAREF core concepts

The starting point of the SAREF ontology is the concept of *Device* representing tangible objects designed to accomplish one or more *Tasks* in diverse types of locations and associated with *States*. The SAREF ontology offers a list of basic *Functions* that can be combined towards more complex functions in a single device. A S*ervice* can represent one or more functions offered by a device that wants its functions to be discoverable, registerable, and remotely controllable by other devices in the network. A service also specifies the device that is offering the service, the functions to be represented, and the input and output parameters necessary to operate the service. Furthermore, a device in the SAREF ontology can be characterized by a *Profile* that can be used to collect information about a certain *Property* or *Commodity* (e.g. energy or water) for optimizing its usage in the home/building in which the device is located. Together with the *Measurement*, *Property* and *UnitOfMeasure*, the ontology allows to relate different measurements from a given device for different properties measured in different units.

Although SAREF is domain-independent ontology, it provides building blocks that allow separation and recombination of different parts of the ontology depending on specific needs. Thus, SAREF is designed as a set of simple core ontology patterns, that can then be instantiated for multiple engineering-related verticals [15]. Based on this, a number of extensions has been or are being developed and standardized to facilitate the domain-specific modeling and management, for examples, SAREF for Energy, SAREF for Environment and SAREF for smart cities. Each domain-specific extension inherits and reuses parts of SAREF core concepts that are relevant for the domain and adds new concepts. The SAREF extensions ease its adoption and extension by industrial stakeholder, while ensuring easy maintenance of its quality, coherence, and modularity.

SAREF ontology is defined essentially as a device-centric ontology including sensors and actuators, which focuses on functions and measurements given by devices. Some definitions within SAREF are closely related to oneM2M concepts, and an official mapping between SAREF and oneM2M is also defined [6]. Similar to oneM2M, the SAREF ontology seeks to cover with a minimum number of concepts IOC's Physical, Data and Information layers, and WOC's Accessibility and Findability layers.

C. WoT Thing Description

The WoT TD (Thing Description) [8] is defined within W3C's WoT working group as a thing-centric formal model and a common representation for WoT. A *Thing* is an abstraction of a physical or virtual entity which interacts and participates in the WoT, and a TD describes the metadata and interfaces of Things. The TD formal model can either be used within JSON as a schema or, more powerfully, within a JSON-LD description to represent the knowledge about things in a machine-understandable way. The main classes of WoT are depicted in Fig. 5 with three main modules:

- The core TD ontology within blue rectangles, reflecting WoT's paradigm of Properties, Actions and Events.
- The data schema ontology within green rectangles, reflecting the terms for data types and data validation.
- The security schema, reflecting the security mechanism and associated configuration requirements.

In the core ontology with blue rectangles, besides the *Thing* concept, the *Property* is used for sensing and controlling parameters by exposing internal state of the thing; the *Action* offers functions of the *Thing* by modelling the invocation of its physical processes, and functions manipulates the internal state of the thing when this state is not exposed as a *Property*. The

Event describes the event sources that asynchronously push messages, thus being used for the push model of communication. *Form* is defined to present the metadata that specify the manner of accessing to the service of the *Thing*, and *Links* affiliate things with external things based on the Web link specifications.

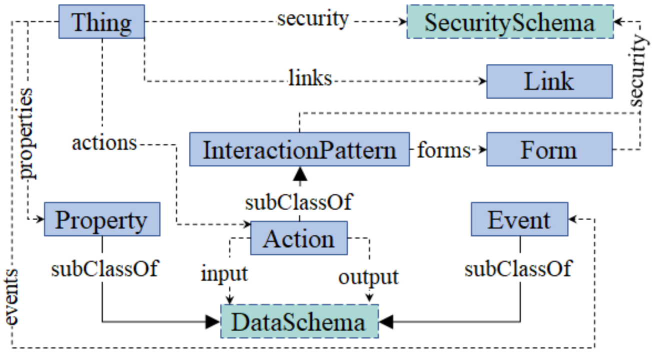

Fig. 5 WoT Thing Description core concepts

The *DataSchema* ontology presents the supported data types (e.g., ArraySchema, ObjectSchema) and enables the linked data declaration of data types. The *SecuritySchema* ontology is designed by reusing parts of the Web-oriented OpenAPI [16] and by adding full coverage of IoT-centered protocols such as CoAP and MQTT. It includes well-established security mechanisms such as authentication, authorization and encryption with different possible levels of security configurations.

WoT TD provides powerful enough metadata to capture different communication scenarios, identified by binding together URI schemes, content types and security mechanisms (authentication, authorization, confidentiality, etc.). It provides a set of interactions, based on a small vocabulary, that makes it possible to integrate diverse devices and to allow diverse applications to interoperate. Since the its objective is to make the Web an interoperable platform for IoT data, the focus point is mainly on the Data Layer of IOC with high-level schemas for describing dynamic behaviors of things; also, by modeling both the syntax and the semantics of behaviors, it supports the Findability Layer of WOC, while its security schema covers the Sharing Layer.

D. SSN and SOSA

The SSN (Semantic Sensor Network) and SOSA (Sensor, Observation, Sample, and Actuator) are a set of ontologies that describe sensors, actuators, samplers as well as their observations, actuation, and sampling activities [9], which have been published as both a W3C recommendation and as an OGC implementation standard. The ontology set adopts a modular architecture with SOSA as a self-contained core with SSN as an extension, to add expressivity and breadth supporting a range of sensing- and actuating-based cases such as scientific monitoring, industrial infrastructures and IoT.

Fig. 6 shows the key concepts within SOSA, emphasizing the common patterns used by the three activities (i.e., actuation, sampling and observation) with classes stacked in different colors where they play a similar role [17]. Generally, the ontologies recognize *Sensors* that make *Observations* about some *ObservableProperty* of a *FeatureOfInterest*, they include *Samplers* that make a *Sampling* of some *FeatureOfInterest* to produce a *Sample*, and they also model *Actuators* making some *Actuations* on some *ActuatableProperty* of a *FeatureOfInterest*. The ontology also takes into account the spatial aspect of an observation, sampling or actuation, which can be associated with the *FeatureOfInterest*, the *Sensor*, *Sampler* and/or the *Platform*.

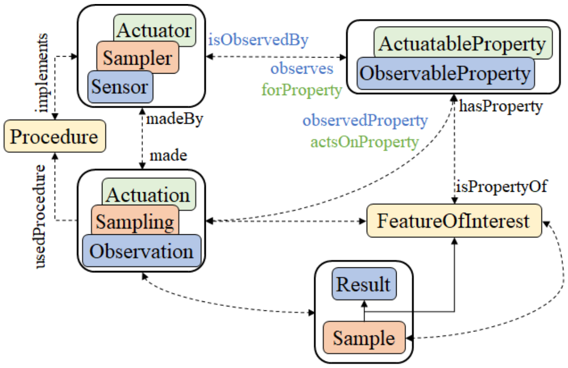

Fig. 6 SOSA core concepts

The modularization strategy adopted enables to segment the ontologies into smaller parts, providing users with just the knowledge they require and reducing the scope as much as possible to what is strictly necessary in a given use case. SSN imports the SOSA ontology, adds additional axioms and concepts, that are organized into several other components in addition to the core SOSA. Specifically, these SSN components consider the *FeatureOfInterest* as the target object in the physical world to interact with, and *Procedures* as a reusable workflow or method. *Results* are used to attach properties to device activities. *System* is a physical piece of technology, while *System Capabilities* and *Deployment* represent the lifecycle of a deployed system.

Regarding the IOC coverage, SSN and SOSA mainly cover the Physical Layer (including sensors, actuators and samplers of the IoT world) and the Data Layer to model the corresponding data. Comparing them to oneM2M and SAREF, the ontologies provide a rich specification on the target layers, while Information Layer and Knowledge Layer are not included. Besides WOC's Findability Layer, the *System* specifications of SSN and SOSA include system compositions that cover WOC's Composition Layer.

E. CIM Information Model

The ISG CIM (Industry Specification Group for cross-cutting Context Information Management) was formed in ETSI in 2017, to improve interoperability and reduce deployment problems for Smart City (and other) services which use IoT and/or metadata [18]. The goal of the ISG CIM is to issue technical specifications to enable multiple organizations to develop interoperable software implementations of a cross-cutting context information management layer, which enables applications to discover, access, update and manage context information from many different sources, as well as to publish it through interoperable data publication platforms. The Information Model of ISG CIM [7] is a work item dedicated to designing a data-centric ontology that supports the data management by the CIM APIs.

The current information model of ISG CIM is based on RDF standards to capture high-level relations between entities (i.e. IoT devices, group of devices or non-IoT information) and properties of entities, as shown in Fig. 7.

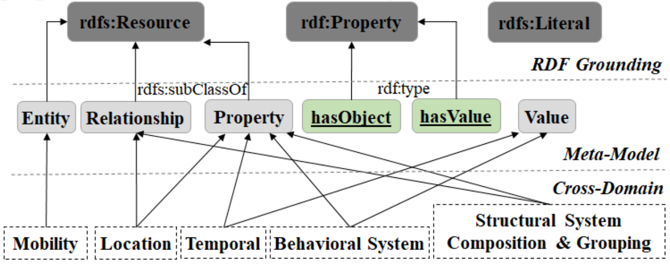

Fig. 7 CIM Information Model

In Fig. 7, the Meta-Model level takes RDF as groundings and extends to general concepts including *Entity*, *Relationship* of Entities, *Property* and *Value*. The *hasObject* and *hasValue* are defined to enable RDF reification, based on the blank node pattern, to leverage the property graph model [19]. While RDF triples could be seen as just an alternative way to capture context information, they start from a very different premise, placing the emphasis on semantic rather than structural information. The property graph could be characterized as standing midway between traditional object-oriented or entity-attribute-value models and purely semantic models, such as RDF. RDF reification is the default way to model property graphs, allowing to attach properties to relationships, which RDF does not directly support.

At the Cross-Domain level, extensions are made to further describe the general concepts at the meta-model level, which include other concepts such as mobility, location, temporal, system behavior, and system structure and composition.

- Mobility defines the stationary, movable or mobile characteristics of an entity;
- Location differentiates and provide concepts to model the coordination based, set based or graph-based location;
- Temporal specification includes property and values for temporal property definitions;
- Behavioral system includes properties and values to describe system state, measurement and reliability;
- System composition and grouping provides a way to model system of systems in which small systems are composed together to form a complex system following specific patterns.

This ontology mainly focuses on IOC's Data Layer to provide generic models for data interoperability and federation purposes. The information model has rather rich expressivity comparing to other standard ontologies in this paper, while it does not cover other IOC layers. The information model also covers the WOC's Findability Layer and defines several composition patterns, spanning the Composition Layer as well.

IV. REVIEW SUMMARY AND FUTURE DIRECTIONS

We hereby in Table I summarize our review, which re-groups information for each ontology from previous sections, along with three dimensions: the coverage degree, the covered WoT layers, and other ontology extensions in addition to the canonical WoT layers. Regarding the coverage degree, we introduce "general" and "rich" coverage: general coverage means the ontology provides a minimal subset of the concepts and relations relevant to each layer, while rich coverage means it covers all common concepts. It's important to point out that the coverage here takes into account the ontology implication in each layer, and the concepts defined in each ontology vary from one to another.

TABLE I. ONTOLOGY COVERAGE FOR WOT

Ontology	Coverage of WoT Layers	
oneM2M	*Coverage*	- General coverage
	IOC	• Physical Layer, Data Layer, Information Layer
	WOC	• Accessibility Layer, Findability Layer
SAREF	*Coverage*	- General coverage - Domain specific extensions
	IOC	• Physical Layer, Data Layer, Information Layer
	WOC	• Accessibility Layer, Findability Layer
WoT TD	*Coverage*	- General coverage - Data schema and security schema
	IOC	• Data Layer
	WOC	• Findability Layer, Sharing Layer
SSN& SOSA	*Coverage*	- Rich coverage - Spatial aspects
	IOC	• Physical Layer, Data Layer
	WOC	• Findability Layer, Composition Layer
CIM Information Model	*Coverage*	- Rich coverage - Mobility, temporal, location, system behavior and system composition
	IOC	• Data Layer
	WOC	• Findability Layer, Composition Layer

We can see that IOC's Physical and Data layers are well covered by most of the ontologies, while Information and Knowledge layers are not well supported. About WOC's coverage, Findability Layer is apparently supported by all ontologies, and Sharing Layer is poorly covered due to the lack of security-management-related specifications.

Based on the review, we now point out several potential directions for improvements of further WoT ontology development and standardization.

Tradeoff between Ontology Richness and Usability. By definition, domain-independent ontologies capture the most frequent/common concepts. Often times they tend to be terse, limiting the model's expressiveness due to the desire of constraining the raw number of core concepts. Hence, we advocate for a stronger modularity when doing modelling work within a large abstract domain, concentrically building richer additional components around the kernel ideas. Only the kernel needs to be imported by users in the simplest scenarios, while extra modules are used when the use case needs to capture the additional subtleties. W3C SSN and SOSA is an example to provide richer concepts, without overloading the base specification, Concentric modularity helps avoiding a fixed balance between ontology richness and usability, which could hinder large-scale adoption of the model.

Knowledge Representation. One of the key advantages of semantic technology is to enable knowledge discovery from raw data, as well as knowledge management (e.g., proof, trust, semantic rules). Nevertheless, a multi-faceted representation of knowledge is poorly supported in the existing standards we have examined, and, consequently, knowledge management is narrowed-down in real-world cases. More efforts are expected, in order to investigate how more complex knowledge representations of the WoT universe can be captured (and then exploited) in our standardized models.

Transversal Concepts Support. Besides the cross-domain specifications, a number of transversal (cross-layer) concepts have clearly emerged in the WoT universe, such as temporal concepts, geography concepts, security and privacy, etc. Although each of them involves further specifications, due to the unavoidable use of such concepts in various contexts, standard ontologies will make a difference to include representative transversal concepts or link related ontologies as references in the ontology as in the CIM information model for instance. WoT Thing Descriptions with specifications of security model is also an example.

Domain Specific Extension. WoT is an enormous horizontal domain that covers many vertical domains such as manufacturing, energy, environment and so on. Both cross-domain and domain specific ontologies are needed when working within a specific vertical domain, and usually this is realized by manually federating ontologies and create mapping between overlapping concepts. The adoption of ontologies will be largely simplified if standard ontologies also provide domain specific extensions such as in SAREF, or the mapping from domain-specific to cross-domain concepts is specified during ontology development.

V. CONCLUSION

Semantic Web of Things becomes more and more favorable to interoperability of closed IoT resources and to creation of friendly interfaces, in order to further the shared usage of data and services. The process of standardizing ontologies for the WoT, to serve as a reference for both academia and industry applications, should target well-established criteria and keep a consistent and simple enough structure. In this paper, we reviewed some standard models, by analyzing their concepts and coverage of WoT-conformant architectures, so as to (a) help lowering the barrier for traditional IoT stakeholders when embracing semantic technologies, and to (b) draw attention of the IoT community towards ontology development and standardization. As a future work, we are investigating richer evaluation criteria to illuminate the ontologies in the WoT domain. We are also enlarging our survey, reaching out to other popular WoT and IoT ontologies.

ACKNOWLEDGMENT

The research leading to these results has been supported by the Fed4IoT project, which receives funding from the European Union's Horizon 2020 Research and Innovation Programme under grant agreement No. 814918. The content of this article does not reflect the official opinion of the European Union. Responsibility for the information and views expressed in this article lies entirely with the authors.

REFERENCES

[1] D. Guinard and V. Trifa, Building the Web of Things: With Examples in Node.Js and Raspberry Pi, 1st ed. Greenwich, CT, USA: Manning Publications Co., 2016.

[2] W. Li and G. Privat, “Cross-Fertilizing Data through Web of Things APIs with JSON-LD,” in Services and Applications over Linked APIs and Data, ESWC2016 workshop, 2016

[3] J. Cardoso and A. P. Sheth, Eds., Semantic Web Services, Processes and Applications. Springer US, 2006.

[4] Linked Open Vocabularies for Internet of Things, http://lov4iot.appspot.com/

[5] TS-0012 oneM2M Base Ontology, https://git.onem2m.org/MAS/BaseOntology

[6] Smart Appliances Reference Ontology and oneM2M Mapping, https://www.etsi.org/deliver/etsi_ts/103200_103299/103264/02.01.01_60/ts_103264v020101p.pdf

[7] CIM Information Model, https://portal.etsi.org/webapp/WorkProgram/Report_WorkItem.asp?WKI_ID=51351

[8] Web of Things (WoT) Thing Description, https://www.w3.org/TR/wot-thing-description

[9] Semantic Sensor Network Ontology, https://www.w3.org/TR/vocab-ssn/

[10] WISE-IoT D1.2, http://wise-iot.eu/en/deliverables/

[11] D. Guinard, “A web of things application architecture: Integrating the real-world into the web,” PhD Thesis, ETH Zurich, 2011.

[12] What is the Web of Things – Web of Things, https://webofthings.org/2017/04/08/what-is-the-web-of-things/

[13] oneM2M, http://www.onem2m.org/

[14] Ontologies - Smart Appliances Project, https://sites.google.com/site/smartappliancesproject/ontologies

[15] M. Lefrançois, “Planned ETSI SAREF Extensions based on the W3C&OGC SOSA/SSN-compatible SEAS Ontology Paaerns,” in *Workshop on Semantic Interoperability and Standardization in the IoT, SIS-IoT*, 2017.

[16] OpenAPI Specification, https://github.com/OAI/OpenAPI-Specification/blob/master/versions/3.0.1.md#securitySchemeObject

[17] Armin Haller et al., “The Modular SSN Ontology: A Joint W3C and OGC Standard Specifying the Semantics of Sensors, Observations, Sampling, and Actuation,” Semantic Web Journal, 2018.

[18] ISG CIM Work Programme, https://portal.etsi.org/tb.aspx?tbid=854&SubTB=

[19] W. Li, G. Privat, J. M. Cantera, M. Bauer, and F. L. Gall, “Graph-based Semantic Evolution for Context Information Management Platforms,” in *2018 Global Internet of Things Summit (GIoTS)*, 2018, pp. 1–6.