Interoperability of Semantically-Enabled Web Services on the WoT: Challenges and Prospects

Mahdi Bennara
mahdi.bennara@emse.fr
Mines Saint-Etienne, Univ Lyon, Univ Jean Monnet, IOGS, CNRS, UMR 5516, LHC, Institut Henri Fayol, F - 42023
Saint-Etienne, France

Maxime Lefrançois
maxime.lefrancois@emse.fr
Mines Saint-Etienne, Univ Lyon, Univ Jean Monnet, IOGS, CNRS, UMR 5516, LHC, Institut Henri Fayol, F - 42023
Saint-Etienne, France

Antoine Zimmermann
antoine.zimmermann@emse.fr
Mines Saint-Etienne, Univ Lyon, Univ Jean Monnet, IOGS, CNRS, UMR 5516, LHC, Institut Henri Fayol, F - 42023
Saint-Etienne, France

Nurten Messalti
nurten.messalti@emse.fr
Mines Saint-Etienne, Institut Henri Fayol, F - 42023
Saint-Etienne, France

ABSTRACT

The advent of the Web of Things as an application layer for the Internet of Things has led to the proliferation of Web services exposing data and functionality of the networked objects. Since resource-oriented architectures align well with the WoT architectures, RESTful services have been the go-to interface to expose the connected devices on the Web. However, the heterogeneity of descriptions of devices and services as well as underlying IoT-level protocols has led to a number of interoperability issues. Recently, the growing popularity of semantically-enabled services has led to the emergence of services described with and exchanging RDF. In this position paper, we attempt to frame the challenges encountered in enabling semantic interoperability of heterogeneous WoT services, with the help of a real world production line scenario. We also propose preliminary solutions to the main issues hampering the establishment of true semantic interoperability on the WoT.

CCS CONCEPTS

• **Information systems → World Wide Web**; **Web services**; *Web applications.*

KEYWORDS

Semantic Interoperability, Web API, Web Service, Semantic Web, Web of Things, Internet of Things.

ACM Reference Format:
Mahdi Bennara, Maxime Lefrançois, Antoine Zimmermann, and Nurten Messalti. 2020. Interoperability of Semantically-Enabled Web Services on the WoT: Challenges and Prospects. In *The 22nd International Conference on Information Integration and Web-based Applications & Services (iiWAS '20), November 30-December 2, 2020, Chiang Mai, Thailand.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3428757.3429199

1 INTRODUCTION

In the last few years, the growing popularity of RESTful Web APIs has been an enabling factor for lightweight resource-oriented data exchange. Recently the advent of new Semantic Web technologies as well as a number of standardized languages, protocols and ontologies has allowed RESTful Web APIs to efficiently expose rich semantically-enabled interfaces. In addition to that, the establishment of RDF as the universal abstract model for structured data on the Web coupled with the resource-oriented nature of REST and the thing-oriented foundation of the Web of Things (WoT) and the Internet of Things (IoT) is considered the ideal recipe for a semantically-enabled Web API infrastructure. However, despite all these advantages, RESTful Web APIs still suffer from limited semantic interoperability especially when it comes to exchanging heterogeneous formats. This gave rise to a stack of technologies that aim to address this problem, such as semantic data validation and lifting/lowering languages [22]. The reasons behind heterogeneity of formats on the Web are multiple: diversity of targeted devices, diversity of domains and their conceptual knowledge schemes, languages optimized for a certain application domain, and more [10].

The IoT is recently experiencing a rise in popularity in various application domains such as smart wearables, smart buildings, city management, eHealth and many more. Consequently, the lucrative business opportunities that came with this [21] gave birth to a multitude of competing manufacturers fabricating a wide spectrum of heterogeneous devices ranging from simple sensors and actuators to more complex equipment. This generated a number of problems [2] that continue to hamper the full exploitation of the possibilities offered by the IoT. The problems arise mainly from the differences between manufacturers using varying technologies, protocols and data formats for their device communication. Furthermore, constrained devices use optimized protocols such as BLE and binary data formats such as CBOR, while more complex devices, with advanced capabilities, use more resource-heavy protocols and formats such as Wi-Fi and JSON. This contrast in the technologies used by the building blocks of the IoT requires an additional layer of interoperability in order to allow devices to seamlessly interconnect with the rest of the components in the architecture.

---

iiWAS '20, November 30-December 2, 2020, Chiang Mai, Thailand

ACM ISBN 978-1-4503-8922-8/20/11…$15.00
https://doi.org/10.1145/3428757.3429199

The WoT, on the other hand, is also presenting several challenges related to semantic interoperability. The recent advent of semantic Web technologies as well as the lightweight RESTful Web APIs is considered the perfect opportunity to overcome this problem [4]. However much work is still needed to integrate all these technologies into a single universal framework that can be adopted in a wide variety of application domains. One of the shortcomings of REST in the WoT is the absence of an asynchronous mechanism that allows device services to notify clients about updates of their state. This can be overcome by using polling or observation modes implemented on certain protocols or by using other alternatives such as Websockets. Also, despite the efforts being made by the different normalization organizations in their quest of semantically describing the devices and their capabilities on the Web, the WoT does not yet have a universally established model. If we take the example of a temperature probe, it is considered in the SOSA/SSN [15] ontology as a *sensor* and its measure is modeled as an *observation*. In SAREF [6] it is modeled as a *device* and the observed temperature is modeled as a *measurement*. However in the WoT Thing Description [19] model it is considered a *thing* and its temperature is represented as a *property* that corresponds to a retrievable information not necessarily associated to a specific point in time.

The IoT and WoT research communities are still trying to overcome the problems we mentioned earlier, and none of the attempts made so far fully resolve the issues. The purpose of this paper is to highlight the challenges encountered when trying to integrate existing technologies into a framework enabling devices to seamlessly exchange and understand data and perform complex operations.

The remainder of the paper is organized as follows, section 2 presents a motivating example to highlight the issues and to support our proposals, section 3 presents in detail the main issues, section 4 discusses a number of proposals, section 5 presents the related work and section 6 concludes the paper.

2 USE CASE AND MOTIVATING EXAMPLE

We consider an existing production line, which is part of the IT'm Factory technological platform (https://itm-factory.fr/), that executes the following product packaging process: (1) fetch empty containers, (2) fill them with a product, (3) grab a lid to close the containers and (4) arrange them on a packaging tray.

The current disposition of the use case poses some problems hampering the full exploitation of the potential offered by the equipment. The different automatons responsible for each step are hard-wired. The addition of any new devices also requires them to be hard-wired on the automaton and their interfaces need to be hard-coded on its system. Furthermore, it is not possible to access the device directly without going through the automaton's own interface.

In order to increase the flexibility and accessibility of this disposition, we can expose interfaces of devices (sensors, actuators and automatons) as Web APIs, giving modular access to information and interfaces involved in the production line. Raw sensor data can be fully exploited by *lifting* it to RDF for piloting and monitoring applications to process and reason on it, and then feeding processed information back to the actuators by *lowering* it [22].

Let us consider the following scenarios to illustrate the limitations of the current disposition:

Scenario 1 - Adding new devices:

We wish to add other devices to the production line, for example a hygrometry sensor measuring the product's humidity before sealing and a dehumidifier that acts upon the product when readings do not conform. Right now, we have to manually *hard-wire* both of them and *hard-code* their behavior in the automaton's system. If we instead expose the devices' interfaces as Web API, we decouple them from the automaton and delegate the coordination to the control and monitoring applications.

Scenario 2 - Reusing existing devices:

We want to reuse the same equipment in different processes, for example, the robotic arm that closes the containers in the packaging process can be involved in the product quality control process as well by measuring the temperature when prompted. Right now, the robotic arm is configured to only close the containers and route them to the next automaton's conveyor belt. The robotic arm's behavior needs to be dynamically adjustable depending on information collected from the other sensors.

Scenario 3 - Dynamic Feedback

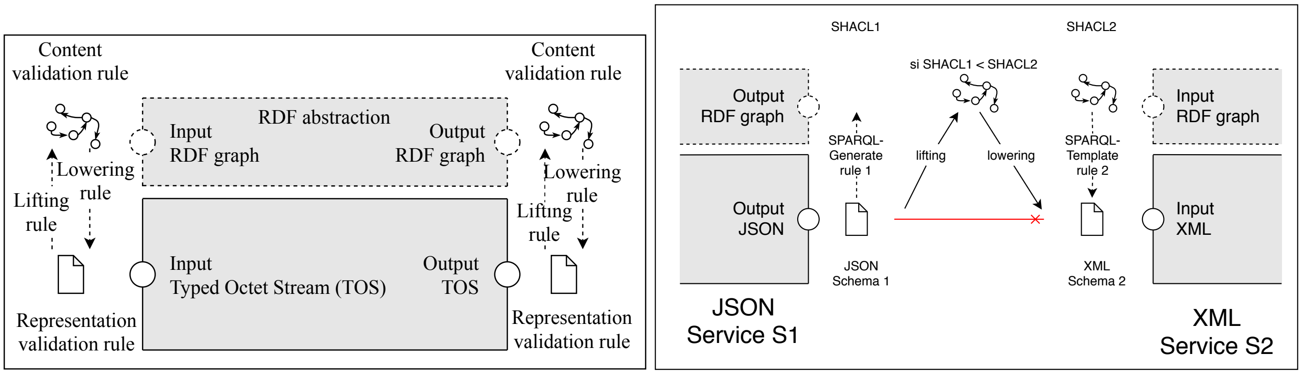
The system is required to provide intermediary indicators and statistics about various aspects of the process. For example, we want to know the percentage of product wasted in the process using a weighing sensor. Right now, the production line relies on centralized solutions such as Manufacturing Execution Systems which suffer from a number of problems in highly dynamic environments [27]. Instead we can exploit device Web APIs and aggregate information coming from different and heterogeneous sources in the process to generate new information that can be used to potentially influence the flow of the process.

3 SEMANTIC INTEROPERABILITY CHALLENGES IN THE WOT

Devices in today's WoT mostly exchange raw non-semantic data. They do not necessarily understand the full meaning of data they exchange which hampers harnessing the full benefits of the available information. An additional semantic layer needs to be implemented to allow machine-readability and consequently automation of tasks such as discovery, selection and composition. In this section we identify, investigate and highlight a number of challenges around enabling semantic interoperability in the WoT.

While RDF has proven to be an effective data model for interoperability on the application layer, its verbose serialisation formats (e.g. RDF/XML, NTriples, or Turtle) present a challenge on the presentation layer. Other than some approaches using the HDT [12] serialisation of RDF [16] or other binary representations of RDF [5], there has been little work and even fewer uptake in industry of providing WoT devices that consume and produce RDF.

Consequently, many data formats and data models exist and they compete with each other for adoption in devices in different WoT domains. Standardisation groups rather try to solve this problem by standardising data formats and service APIs [7, 13, 18]. Some works aim at tackling semantic interoperability despite the heterogeneity of data formats and service API specifications, i.e., across platforms.



(a) Web service with RDF abstractions.

(b) Heterogeneous things interacting on the WoT.

Figure 1: Semantics in the Edge: using RDF abstractions for the content of Web resource.

The use of semantic Web technologies has been investigated to facilitate semantic interoperability among these platforms [23, 30]. One challenge is to investigate how semantic interoperability can be obtained on the edge level, i.e. between devices directly, instead of between platforms [3]. The work in [22] is a starting point to investigate how constrained devices that are not natively semantic Web enabled can still interoperate with one another.

Figure 1a illustrates a typical Web service that consumes and outputs resource representations, that are octet streams typed with internet media types according to the W3C Web architecture principles. For some data formats such as JSON or XML, dedicated validation languages such as JSON Schema or XML Schema may be used. Then, the contents of the resource whose representation is given as input or output may be assumed to be an RDF graph. Adopting such an abstraction enables us to assume the service, potentially exposed by a constrained device, consumes and produces RDF. Many languages can be used to specify how an RDF graph can be generated out of octet streams (lifting), or the other way around (lowering). Finally languages such as SHACL or ShEX can be used to specify what form the content RDF graph has. New research challenges stem naturally from this vision. For example: given a JSON Schema document and a lifting rule, how do we compute the SHACL or ShEx shape that allows validation of the generated RDF?

Figure 1b illustrates a combination of two WoT services that are seemingly incompatible, but that as an abstraction generate and consume RDF, respectively. The output RDF graph, generated using a certain lifting rule, could then be lowered using the second thing's lowering rule. In this setting, the condition for the services to be composable is that the content validation rule of the first thing is more specific than the content validation rule of the second thing.

Formal definitions for the Semantic WoT: In order to achieve our goals, first there is a need to formally define concepts related to devices, data they exchange and how they behave in a semantically-enabled architecture. Formal definitions for lifting, lowering and RDF presentations have been given in [22], but no formal model has been defined for devices, descriptions, interfaces, protocols and data. Existing device description vocabularies and/or ontologies like WoT TD (and its derivations such as SWoT [1]), SOSA/SSN and SAREF only give ontological descriptions of these concepts.

Heterogeneity of protocols and data formats: The fragmentation in terms of protocol interpretation and message parsing on the IoT has led to interoperability issues between the different building blocks of the IoT [26]. This carried over to the WoT and many efforts to semantically describe the things in the WoT and their interfaces have been made (cf. section 5). Unfortunately, this contributed to increasing the heterogeneity of descriptions and complicated the process for devices to understand data they exchange.

Universal data model and rulesets: RDF, with its flexible model, has established itself as the *lingua franca* for the semantic Web. In practice, it is by far the most widely used knowledge representation model to describe resources on the Web. Semantic interoperability through RDF can be achieved by lifting and lowering raw data exchanged by devices. The rulesets defining how these two operations are carried out must be made available to communicating devices. The challenge resides in where these rulesets are stored, how they are reused and how the exchanging parties reference them.

Constrained devices: Devices sometimes are not able to accommodate the added semantic complexity of producing, consuming and treating RDF. The challenge resides in delegating the semantic layer computations to other components in the architecture. What components and how they interchange with the devices, are questions that remain unanswered.

Technological challenges: In addition to RDF for knowledge representation and REST for resource oriented communication, many non-semantic data formats, validation rule languages and technologies for lifting and lowering exist [22] and are more or less widely adopted. The main challenge lies in combining and integrating these technologies together in a single framework enabling heterogeneous devices to seamlessly exchange knowledge. In the context of the scenarios of section 2, this would mean developing a platform or a middleware that handles new and existing device interfaces as well as data they exchange.

IoT in manufacturing and business: The financial sustainability of a manufacturer depends largely on its ability to continuously improve the performance of its production units in order to control the cost price of its products. Measuring industrial performance consists of monitoring costs (materials, energy, etc.) and controlling processes (shutdowns, scrap, etc.). Today, this monitoring involves the collection and analysis of production data in real time, partly made possible by the digitization of production tools. It is in this context that the use of IoT is developing in the industrial environment. Among other things, IoTs allow industrialists to automate the recording of operating data in order to be more efficient in their process monitoring and thus implement actions more adapted to the real needs of machines and the company in general (example of application: predictive maintenance defined from the history of machine breakdowns and real time data recorded from objects). They would also give them the means to respond to the ever-increasing and ever-changing customer requirements. Facilitating the integration of these connected objects is therefore a real challenge and a real need for companies to become more competitive.

4 PROSPECTS FOR SOLUTIONS

This section presents and discusses a number of proposals aimed at resolving the problems identified in section 3. We identify a set of assessable criteria in order to evaluate our proposals' applicability: (1) simplicity: the overall complexity of operation and implementation, (2) modularity: how the components are decoupled from each other, (3) reusability: applicability in different scenarios and application domains, (4) extensibility: the ability to implement future extensions without the need for a complete overhaul. While this set of criteria might not be comprehensive, we believe that it represents a solid base to assess the applicability of the proposals presented here, and any future proposal.

Relying on third party semantic converters: In some scenarios, such as the ones from section 2, devices cannot exchange RDF natively, which prompts the need to use third party components: intermediary services that lift or lower data to and from RDF using the appropriate rulesets and tailor the results to match the appropriate schema or shape. This can be taken a step further by exposing an entire parallel service-oriented architecture to allow devices in an existing environment to interoperate on the semantic level. We can rely on specialized transformation nodes, such as the ones defined in [11], in order to create a pipelined modular transformation flow capable of more than just lifting and lowering. While this solution is modular, reusable and extensible, it increases the complexity of a WoT architecture and introduces multiple points of failure.

Layered Ontologies and Semantic Negotiation: Ontological heterogeneity can be solved by relying on low level highly reusable and extensible ontologies that are less expressive but simple to work with in a wide variety of scenarios. This way we solve the interoperability problem at the low level first. Expressivity can be obtained by using higher level more complex (or domain-specific) ontologies that build on the low-level ones, but do not necessarily guarantee interoperability. Any node can negotiate, depending on its capabilities, the ontological level to be used when communicating with it. Low-level ontologies are simple to understand by humans, easy to parse by machines, computationally efficient for constrained devices and able to be hosted on a low memory local device, but offer limited reasoning possibilities and hinder the discoverability of the elements they describe.

Service-oriented communication: The data silo approach for the WoT suffers from a number of issues in the open environment of the Web. Instead we encourage exposing sensor data, actuator interfaces and other devices uniformly [26] as services effectively abstracting and decoupling them from the physical devices. REST not only enforces the uniformity of the service interfaces but is also resource-oriented which is in accordance with the thing-oriented foundation of the WoT [14]. This enables simpler compositions of various real devices into a composite virtual one that can perform a composite value-added function such as aggregating sensing data and performing complex commands, regardless of the underlying physical implementation [24]. However, a publish/subscribe or an event-based model for data access is sometimes more optimal than a resource-oriented one, especially in notification-heavy scenarios.

Plug-and-Communicate Approach: Building an environment where new devices are communication-ready with a minimum amount of configuration would be a huge step forward in enabling extensibility and flexibility in any WoT architecture. By providing reusable *protocol bindings* and *message parsing modules*, we can reduce the complexity overhead for integration of new devices to an existing architecture. This can be achieved by utilizing services as a reusable foundation for how devices will connect and exchange data, emphasizing on a high level of loose coupling. The result would be a modular architecture where modules are easily reused, replaced, interchanged or extended with new functionality, without the need to hardwire or hardcode them [28], answering the needs illustrated by scenarios 1 and 2 of section 2. In addition to that, smaller services and components are more easily describable (understandable), reusable, extensible and composable [29].

5 RELATED WORK

For a long time, semantic interoperability has been a problem in large-scale distributed systems [17] and on the Web in particular [25]. The heterogeneity of models and formats to represent knowledge originating from multiple sources was inevitable. Some applications eventually need to consume heterogeneous *meaningfully overlapping* [8] data in order to offer added-value to users.

Early proposals lacked the widely adopted technologies and standards we have today to support their abstract models, and did not take into account scalability. The problem of semantic interoperability is defined in [9] as *the faculty of interpreting knowledge imported from other languages at the semantic level*, and the authors show that the best approach to ensuring interoperability of knowledge representations is through transformation. The problem of transformations between knowledge representation languages is discussed and addressed in [8] with a highlight on the importance of meaning preservation. In [11], the authors propose *Transmorpher*, a framework allowing composition of reusable abstract elemental transformations while preserving meaning.

In the last few years, the standardization organizations have taken a particular interest in pushing recommendations for semantically enabled device description formalisms. The WoT Thing Description [19] is a W3C proposed recommendation, it is a formal model to represent descriptions of physical or virtual things in the WoT context, which are considered entry points for M2M interactions with the thing. A TD contains metadata about the thing, possible interaction affordances (property affordances, action affordances, event affordances), data schemas for exchange with the thing and Web links to express formal/informal relations with other things or resources. WoT TD is supplemented by WoT Protocol Binding Templates [20], which aims at allowing different objects communicating using heterogeneous protocols to interact seamlessly using the abstract model exposed by its TD. The SOSA/SSN Ontology [15] is a W3C recommendation aimed at describing sensors and actuators and the aspects revolving around their properties, actions and exchanged data. SSN includes a core ontology called SOSA (Sensor, Observation, Sample and Actuator) to describe its elementary classes and properties. SSN and SOSA offer different degrees of granularity for descriptions which makes them able to be applied and adapt to a wide range of applications including the WoT. The SAREF ontology [6] and its extensions is an effort made by the TNO and standardized by the ETSI in order to specify recurring core concepts shared by appliances in various domains. The main concept of SAREF is the *device*, and depending on the domain (home, building, energy, etc.) there are additional key concepts that help describe related aspects.

Various efforts have been made in order to use Web services as an infrastructure for data exchange between devices on the WoT. ConnectOpen [26] is a platform architecture enabling automatic integration of heterogeneous IoT devices and exposing the consolidated data they exchange via REST APIs. The ConnectOpen platform relies on automatically generated communication agents that are adapted to the specifications of the device being integrated to the platform. These agents are deployed on gateways that allow devices to have connectivity to the central platform.

6 CONCLUSION

In order for the WoT to be a distributed network of information where knowledge can be consumed regardless of its format, the problem of semantic interoperability has to be addressed first. In this paper, we presented, with the help of a motivating example, the various difficulties encountered by the scientific community in achieving this goal. We identified, investigated and highlighted a number of open questions that need to be answered in order to overcome these difficulties. We then followed by a discussion on the possible avenues for solutions to the main issues hampering the establishment of true semantic interoperability for a service-based WoT. We proposed preliminary approaches to address these issues and showed the interest of having a Web API driven architecture to expose devices and their capabilities. We showed that relying on RDF and the stack of accompanying technologies and standards together with RESTful Web services can be the key in designing semantically interoperable WoT architectures.

ACKNOWLEDGMENTS

We would like to thank Audrey Cerqueus for her help in building the scenario and identifying issues with the existing solutions.

REFERENCES

[1] Francesco Antoniazzi and Fabio Viola. 2019. Building the Semantic Web of Things Through a Dynamic Ontology. *IEEE Internet of Things Journal* 6, 6 (2019), 10560–10579.

[2] Eleonora Borgia. 2014. The Internet of Things vision: Key features, applications and open issues. *Computer Communications* 54 (2014), 1–31.

[3] Federica Cena, Armin Haller, and Maxime Lefrançois. 2020. Semantics in the Edge: Sensors and actuators in the Web of Linked Data and Things. *Open Journal Of Semantic Web* (2020), 1–10.

[4] Pierre-Antoine Champin. 2013. RDF-REST: a unifying framework for web APIs and linked data. In *Services and Applications over Linked APIs and Data (SALAD), workshop at ESWC, May 2013, Montpellier (FR), France*. 10–19.

[5] Victor Charpenay, Sebastian Käbisch, and Harald Kosch. 2018. Towards a binary object notation for RDF. In *European Semantic Web Conference*. Springer, 97–111.

[6] Laura Daniele, Frank den Hartog, and Jasper Roes. 2015. Created in close interaction with the industry: the smart appliances reference (SAREF) ontology. In *International Workshop Formal Ontologies Meet Industries*. Springer, 100–112.

[7] ETSI. September 2014. Machine-to-Machine communications (M2M); mIa, dIa and mId interfaces. Technical Specification 102 921 V1.3.1, ETSI.

[8] Jérôme Euzenat. 2001. An Infrastructure for Formally Ensuring Interoperability in a Heterogeneous Semantic Web. In *Proceedings of SWWS'01, The first Semantic Web Working Symposium, 2001, California, USA*. 345–360.

[9] Jérôme Euzenat. 2001. Towards a Principled Approach to Semantic Interoperability. In *Proceedings of the IJCAI-01 Workshop on Ontologies and Information Sharing, 4 and 5 August, 2001, Seattle, Washington, USA*. 19–25.

[10] Jérôme Euzenat and Heiner Stuckenschmidt. 2003. The 'Family of Languages' Approach to Semantic Interoperability. *Borys Omelayenko, Michel Klein. Knowledge transformation for the semantic web* (2003), 49–63.

[11] Jérôme Euzenat and Laurent Tardif. 2001. XML Transformation Flow Processing. In *Proceedings of the Extreme Markup Languages 2001 Conference, 12-17 August, 2001, Montréal, Quebec, Canada*. 61–72.

[12] Javier D Fernández, Miguel A Martínez-Prieto, Claudio Gutiérrez, Axel Polleres, and Mario Arias. 2013. Binary RDF representation for publication and exchange (HDT). *Journal of Web Semantics* 19 (2013), 22–41.

[13] Open Connectivity Foundation. 2017. oneIoTa. URL https://oneiota.org/.

[14] Dominique Guinard, Vlad Trifa, and Erik Wilde. 2010. A resource oriented architecture for the web of things. In *2010 Internet of Things (IOT)*. IEEE, 1–8.

[15] Armin Haller, Krzysztof Janowicz, Simon Cox, Danh Le Phuoc, Kerry Taylor, and Maxime Lefrançois. 2017. *Semantic Sensor Network Ontology*. W3C Recommendation. W3C. https://www.w3.org/TR/2017/REC-vocab-ssn-20171019/.

[16] Henning Hasemann, Alexander Kröller, and Max Pagel. 2012. RDF Provisioning for the Internet of Things. In *2012 3rd IEEE International Conference on the Internet of Things*. IEEE, 143–150.

[17] Sandra Heiler. 1995. Semantic interoperability. *ACM Computing Surveys (CSUR)* 27, 2 (1995), 271–273.

[18] Cullen Jennings, Z Shelby, J Arkko, A Keranen, and Carsten Bormann. August 2018. Sensor Measurement Lists (SenML).

[19] Sebastian Käbisch, Takuki Kamiya, Michael McCool, Victor Charpenay, and Matthias Kovatsch. 2020. *Web of Things (WoT) Thing Description*. W3C Recommendation. W3C. https://www.w3.org/TR/wot-thing-description/.

[20] Michael Koster. 2018. *Web of Things (WoT) Protocol Binding Templates*. Technical Report. W3C. https://www.w3.org/TR/2018/NOTE-wot-binding-templates-20180405/.

[21] Vlad Krotov. 2017. The Internet of Things and new business opportunities. *Business Horizons* 60, 6 (2017), 831–841.

[22] Maxime Lefrançois. 2018. RDF Presentation and Correct Content Conveyance for Legacy Services and the Web of Things. In *Proceedings of the 8th International Conference on the Internet of Things*. ACM, 43.

[23] Aref Meddeb. 2016. Internet of things standards: who stands out from the crowd? *IEEE Communications Magazine* 54, 7 (2016), 40–47.

[24] Thomas Menzel, N Karowski, D Happ, V Handziski, and A Wolisz. 2014. Social sensor cloud: An architecture meeting cloud-centric iot platform requirements. *9th KuVS NGSDP expert talk on next generation service delivery platforms* (2014).

[25] Aris M. Ouksel and Amit Sheth. 1999. Semantic interoperability in global information systems. *ACM Sigmod Record* 28, 1 (1999), 5–12.

[26] Nuria Pazos, Michael Müller, Marco Aeberli, and Nabil Ouerhani. 2015. ConnectOpen-automatic integration of IoT devices. In *2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)*. IEEE, 640–644.

[27] B Saenz de Ugarte, A Artiba, and R Pellerin. 2009. Manufacturing execution system–a literature review. *Production planning and control* 20, 6 (2009), 525–539.

[28] Paweł Stelmach. 2013. Service Composition Scenarios in the Internet of Things Paradigm. In *Doctoral Conference on Computing, Electrical and Industrial Systems*. Springer, 53–60.

[29] Ruben Taelman and Ruben Verborgh. 2017. Declaratively Describing Responses of Hypermedia-driven Web APIs. In *Proceedings of the Knowledge Capture Conference*. ACM, 34.

[30] Aparna Saisree Thuluva, Darko Anicic, and Sebastian Rudolph. 2018. IoT semantic interoperability with device description shapes. In *European Semantic Web Conference*. Springer, 409–422.