Semantic Resource Discovery with CoAP in the Internet of Things

Ali Yachir[1], Badis Djamaa[1,2], Kheireddine Zeghouani[1], Marwen Bellal[1] and Mohammed Boudali[1]
[1]Artificial Intelligence Laboratory, Military Polytechnic School, PO BOX 17, Bordj-El-Bahri, 16111, Algiers, Algeria
[2]Centre for Electronic Warfare, Cranfield University, SN6 8LA, Shrivenham, U.K.

Keywords: Internet of Things, Smart Environments, CoAP, Device, Resource, Service, Smart Object, Semantic Web Technologies, CoRE Link Format, Request Resolution, Semantic Resource Discovery, Resource Directory.

Abstract: The Constrained Application Protocol (CoAP) is a lightweight and power-efficient Internet standard specifically designed for M2M communication in the Internet of Things (IoT). CoAP provides a set of mechanisms for IoT interactions including request/response, publish/subscribe and resource discovery. For the latter, a Resource Directory (RD) solution is proposed to register and store information about IoT resources to be queried by users. Such a solution, however, only allows syntactic discovery. In this paper, we extend CoAP with lightweight semantic-rich information by defining appropriate CoRE link format attributes describing both IoT resources and user requests. Such an extension is integrated with the RD to facilitate semantic resources discovery. Implementation and thorough evaluations of the proposed approach show important performance enhancements when compared with the default RD solution.

1 INTRODUCTION

As a natural continuity of ubiquitous computing (Weiser, 1991) and ambient intelligence (ISTAG, 2003), the Internet of Things (IoT) (Atzori et al., 2010; Suresh et al., 2014) envisages a future Internet architecture integrating both physical and cyber worlds by combining sensing and actuation with digital services. The challenge resides in ensuring seamless interoperability among a huge number of constrained heterogeneous IoT devices. Indeed, IoT devices are typically resource-constrained objects mainly characterized by limited memory, energy and processing power. Additionally, such devices communicates over Low-power and Lossy Networks (LLNs), such as IEEE 802.15.4 imposing, thus, other constraints on the reliability and amount of exchanged data. Hence, seamless integration of such devices into the Internet requires new lightweight and power-efficient protocols. Among several alternatives, the Constrained Application Protocol (CoAP) (Shelby et al., 2014) is emerging as a widespread standard that fulfil most of the above-mentioned requirements.

In CoAP-enabled IoT applications, each device is seen as an endpoint, exposing sensor readings, actuating capabilities and internal information as REST resources (Fielding and Taylor, 2002) that can be queried by clients. Moreover, CoAP-based systems usually use a CoRE Resource Directory (RD) (Shelby et al., 2017) where resource providers register their available resources for clients to query. However, the RD solution only allows syntactic and simplistic data-oriented registration and querying of resources. For this reason, automatic and intelligent discovery of required resources among huge heterogeneous ones remains inadequate with the native RD. Thus, semantic enhancement of CoAP and, obviously of RD, is a key aspect for better representing, storing, organizing, discovering and providing information generated/consumed by IoT entities. This challenge can benefit from the semantic Web technologies (Barnaghi et al., 2012; Bonino and Procaccianti, 2014) such as Resource Description Framework (RDF), Ontology Web Language (OWL) and Protocol And Rdf Query Language (SPARQL).

Extending RD with semantic-rich information may follow three major steps. First, defining a comprehensive semantic model describing all the physical and virtual entities surrounding a device such as locations, persons, appliances and resources. Second, extending CoAP to support semantic resources registration and querying of the RD. Third, integrating semantics in the RD itself to allow discovery and ranking of resources, which best match user requirements and closely meet the specified quality of service level.

Yachir, A., Djamaa, B., Zeghouani, K., Bellal, M. and Boudali, M.
Semantic Resource Discovery with CoAP in the Internet of Things.
DOI: 10.5220/0006419400750082
In *Proceedings of the 14th International Joint Conference on e-Business and Telecommunications (ICETE 2017) - Volume 6: WINSYS*, pages 75-82
ISBN: 978-989-758-261-5

Although some work has been proposed for semantic modeling of things in the literature, the three aforementioned steps are either addressed partially or are at a preliminary stage and need to be studied more deeply. Accordingly, in this paper, we propose a new semantic support for CoAP by defining appropriate CoRE Link Format (CLF) (Shelby, 2012) attributes describing both IoT devices and user requests. This work is in the continuity of our previous work presented in (Yachir et al., 2016a; Yachir et al., 2016b).

The rest of this paper is organized as follows. Section 2 discusses related work applying semantic web technologies in IoT. Section 3 provides an overview of the proposed semantic model in our previous work. Section 4 defines appropriate attributes for mapping between the designed semantic model and CoAP. This is followed by the description of the proposed framework for device registration and interrogation via the resource directory (RD). Section 6 is devoted to assess the performance of the proposed mechanisms. The paper concludes in section 7.

2 RELATED WORK

Various research work have been proposed recently to improve interoperability between heterogeneous IoT devices using semantic web technologies. In (Yuan et al., 2013), a general tree-based metadata model is proposed to describe IoT entities along three clusters of information: resource, service and context information. In (Taccari et al., 2015), IoT entities and features characterizing an earthquake scenario are described using earthquake emergency management ontologies (Spalazzi et al., 2014). In (Wang et al., 2015), a framework for multisource heterogeneous information fusion in the IoT is designed. The collected sensor data are modelled using the Semantic Sensor Network (SSN) ontology (Compton et al., 2012). In (Sun and Jara, 2014), the authors propose a semantic model for IoT information organizing where object and event layers are represented using a semantic link network model. In (Jara et al., 2014), the aspects of the Semantic Web of Things (SWoT) are presented and discussed along with analysing their impact on the performance of the IoT resources. In (De et al., 2011), based on the SENSEI project and the SSN ontology, a semantic annotation framework for IoT components is proposed.

In (Ruta et al., 2013), a novel framework for SWoT based on a backward-compatible extension of CoAP is proposed. In (Kovatsch et al., 2015), a semantic description of IoT devices, based on the RESTdesc format (Verborgh et al., 2012), is designed to deal with self-configurable service composition in resource-constrained environments using CoAP. In (Yachir et al., 2016b), a service-oriented, user-centered and event-aware Framework for service discovery and selection in IoT is proposed. In (Ayari et al., 2015), a semantic approach is proposed for robots interaction with humans. In (Han and Crespi, 2017), a service provisioning architecture for smart objects with semantic annotation is proposed to enable the integration of IoT applications into the Web. In (Urbieta et al., 2016), an adaptive service composition framework for IoT-based Smart Cities is proposed. In (Roffia et al., 2016), a publish-subscribe architecture, based on a generic SPARQL endpoint, is designed for interoperability in IoT.

The works discussed above show that various semantic models are proposed to describe things, but they lack three important factors. First, some of the proposed semantic models deal only with sensors capabilities using SSN ontology without addressing actuators and the more general notion of thing ignoring its relationships with either the physical and digital worlds. These relationships are important to identify the entities to which a device is attached or it might control, the space where is located and services (context, event and/or action) it might provide. Second, the notion of quality of service (QoS) is taken into account partially without including either user preferences, the quality of the physical device or the quality of its hosted software resources. Finally, integration of the proposed models in CoAP and RD is not considered or is at a preliminary stage. Indeed, the mapping from concepts in the semantic model to CoAP protocol is not clearly defined. As a result, RD still performs simplistic resource discovery without semantic matching of user requirements and the specified quality of service parameters.

3 PROPOSED SEMANTIC MODEL OVERVIEW

This section gives a brief overview of our proposed semantic model for IoT (Yachir et al., 2016a), which describes both IoT resources and user requests along with their resolution.

3.1 Resource Description

An IoT resource includes the physical device and its hosted software services. Accordingly, its description comprises two main parts: device description and service description, as shown in Figure 1 bellow.

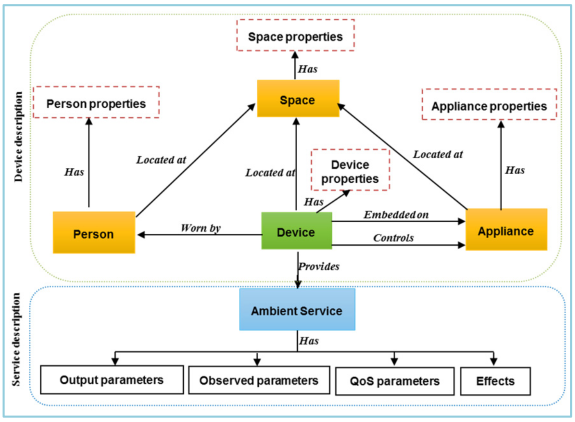

Figure 1: Proposed semantic model for device and service description.

Firstly, the device description part is structured around four main entities: *space, person, appliance* and the *device* itself. Device is the key entity that takes control of the three other entities. It refers to a physical object that can be a sensor or an actuator and it has four main direct relationships: a device may be *worn by* a person or *embedded on* an appliance. It can also *control* an appliance and obviously, it *is located* at a given space. Moreover, each entity is described in its own reference ontology. Accordingly, four reference ontologies are distinguished: *Device Reference Ontology (DRO), Space Reference Ontology (SRO), Person Reference Ontology (PRO)* and *Appliance Reference Ontology (ARO).* Property attributes of such ontologies are defined in a well-known ontology denoted as a *Reference Ontology (RO).* This ontology plays the role of a global dictionary where entities/services share vocabulary.

Secondly, the service description part is structured around a core concept called *ambient service (as)*, which represents a software resource provided by a device. An ambient service can provide output parameters, observed parameters as well as effects with some quality of service level. Output and observed parameters are annotated in the reference ontology (RO) whereas the produced effect is represented as an RDF statement, in a Subject-Predicate-Object structure. Statements are triplet that consist of an Entity ("subject"), a Property ("predicate"), and the value of the Property ("object"), where this value can be another entity, an attribute or a literal. Regarding the quality of service (QoS), we distinguish the quality characterizing an ambient service (SQoS) and the quality of the device (DQoS) hosting such a service.

3.2 Request Description and Resolution

A user request is described by four main components as follows: *requested subject, requested subject property, required QoS level* and *required QoS parameters. Requested subject* is formalized as a concept, a sub concept or an individual from one of the aforementioned reference ontologies. *Requested subject property* is a specific characteristic formulized as an ontological property of the requested entity. *Required QoS level* specifies a minimum score threshold required for the quality of service. To compute a score of an ambient service, user should specify in its request the relative importance accorded for each required QoS parameter.

Candidate ambient services that satisfy functional requirements (i.e. requested subject and requested subject property) are inferred using the *Request Resolution Rule (R3)* depicted in Figure 2 using SPARQL. The inferred services are then evaluated and ranked according to the quality of service including SQoS, DQoS and user's preferences. In this paper, the R3 rule is implemented in the RD side and the proposed semantic model is mapped to the CoAP protocol and implemented under Contiki OS.

```
SELECT ?device ?as WHERE
        { ?device :hasRelationshipProperty :Subject . ?as
:isProvidedBy ?device.
        ?as :hasOutputParameters : Parameter }.........(a)
SELECT ?device ?as WHERE
        { ?device :hasRelationshipProperty :Subject . ?as
:isProvidedBy ?device.
        ?as :hasObservedParameters : Parameter }......(b)
SELECT ?device ?as WHERE
        { ?device :hasRelationshipProperty :Subject . ?as
:isProvidedBy ?device.
        ?as :hasEffects ?eff. ?eff :hasSubject :Subject.
        ?eff :hasObject :Parameter}.............................(c)
```

Figure 2: Request Resolution Rule (R3) described in SPARQL.

4 SEMANTIC-ENHANCED COAP AND CORE LINK FORMAT

CoAP follows the REST architectural style for making data and resources accessible. In fact, every resource in CoAP is identified by a URI (Uniform Resource Identifier). Clients may access resources via synchronous request/response interactions, using HTTP-derived methods GET, PUT, POST, and DELETE. Furthermore, CoAP provides a mechanism for registering, discovering and advertising resources that a given CoAP server is making available. Such discovery protocol uses the CoRE Link Format (CLF) specification as default, where a client can access the reserved */.well-known/core* URI path on the server with the POST method to register a resource, or with GET to discover available ones. GET requests can include specific attributes in the URI-query field to filter a particular resource to retrieve. Some standard attributes are defined in (Shelby et al., 2017). This work reuses existing standard attributes as much as possible along with defining others, when necessary, to fulfil the requirements of mapping our semantic model to CLF.

4.1 CLF based Resource Description

The proposed semantic device description in CLF contains both standard and new added attributes. For instance, the standard attribute *endpoint (ep)* is used for device name while a new attribute called *entity (ent)* is used to map the three main other entities of the model, namely *person, space* and *appliance*. Table 1 presents the proposed mapping of our semantic model in CLF. The mapping reuses standard attributes as much as possible. It may give new semantics to some standard attributes such as *rt* in a way that is backward compatible. The mapping also introduces a few new attributes necessary to describe the semantics of our model. All attributes, their necessity and meanings are discussed below.

- ***ep := endpoint (mandatory).*** The name of the registering device, unique within that domain. Its maximum length is 63 bytes.
- ***et := endpoint type (mandatory).*** The URI of the device domain conceptualization named in our model as *Device Reference Ontology (DRO)*;
- ***rt := resource type (mandatory)***. The functional parameters of the ambient service provided by the device in *et*. As mentioned in section 3, a functional parameter can be an output/observed parameter or an effect. Hence, *rt* is formatted to support both. First, an output/observed parameter is described in *rt* as follow:

  rt= "Service|Output|Output_RO"

  where *Service* indicates the name of the ambient service; *Output* indicates the name of the output parameter; and *Output_RO* indicates the reference ontology where the output parameter is described. Second, an effect is described by six fields in *rt* as follow:

  rt="Service|Effect|Subject|Subject_RO|Predicate|Object"

  where *Service* and *Effect* indicate the name of the ambient service and the effect respectively; *Subject* and *Subject_RO* indicate respectively the name and the reference ontology of the subject of the effect. Finally, *Predicate* and *Object* indicate the predicate name and the object of the effect.
- ***obs := observable (optional)***. Indicates whether the described output parameter in an *rt* field is observable. So, *obs* is combined with *rt* to describe an observed parameter. A resource is observable if its *obs* equals 1.
- ***ent := entity (mandatory).*** Indicates the name of the entity *(space, appliance and/or person)* which has a relationship with the device referenced in *ep* and *et*. Accordingly, *ent* is structured on three fields as follow:

  ent="Space|Appliance:Embedded|Person"

  where *Space* is the name of the location where the device is situated; *Appliance* is the name of the appliance controlled by the device. *Embedded* indicates whether the device is embedded on the appliance or not. Finally, *Person* is the name of the person wearing the device. It should be noted that a given device could be only on one of the three flowing states at the same time: worn by a person; embedded on an appliance or free.
- ***entro := entity reference ontology (mandatory).*** Indicates the URIs of the domain conceptualization of the entities specified in the *ent* field. In other words, it contains the three above mentioned reference ontologies namely: *SRO*, *PRO* and *ARO*. Accordingly, *entro* is structured as follow:

  entro= "SRO|ARO|PRO"
- ***sqos := service QoS (optional).*** Contains the quality of service (QoS) parameters of the ambient service referenced in *rt*. It is structured on *n* fields such as *n* is the number of QoS parameters. Each field is divided on three sub fields indicating the name of the quality parameter, its current value and a flag (min/max) indicating whether the parameter is to maximize *(min/max=1)* or to minimize *(min/max=0)*. Thus, *sqos* is structured as follow:

  sqos="$sqos_1$:val_1: max/min|…|$sqos_n$:val_n:max/min"
- ***dqos := device QoS (optional).*** Contains the parameters of quality of service (QoS) of the device referenced in *ep* and *et*. *dqos* has exactly the same structure as *sqos* where the key word *dqos* is used instead of *sqos*.

Table 1: Resource Description Mapping to CLF.

Concept	Concept attribute	CoAP attribute	Attribute type
Device	Device name	Endpoint (**ep**)	Standard attributes
	Device Reference Ontology	Endpoint Type (**et**)	
Ambient Service	Ambient Service parameters	Resource Type (**rt**)	
	Observable	Observable (**obs**)	
Space, Appliance, Person	Entity Names	Entity (**ent**)	Added attributes
	Entity Reference Ontologies	Entity reference ontology (**entro**)	
QoS	Service QoS	**sqos**	
	Device QoS	**dqos**	

Having introduced and detailed the necessary mapping attributes, let us see the corresponding semantic description of a device named "*Imote2Sensor*" located in the "*Kitchen*" and characterized by *energy level* and *reliability* as *dqos* parameters. This device provides a service "*getTemperature*" having "*temperature*" as an output parameter that can be observed. In addition, the provided service is characterized by a *response time* and *energy cost* as *sqos* parameters. The CoAP message corresponding to such description is:

```
</pathRes1>;ep="Imote2Sensor";et="http://emp.org/Ontologie
s/Device.owl";ent="Kitchen";
entro="http://emp.org/Ontologies/                Space.owl;
dqos="Energy_level:70:1|Reliability:0.6:1";            rt="
getTemperature|temperature|http://emp.org/Ontologies/
```

4.2 CLF based Request Description

Having presented the proposed mapping of our semantic resource description into the CoRE Link Format carried in CoAP messages, this section follows the same approach to map user requests. Hence, as mentioned in section 3.1, a user request is described by four main components, namely: *requested subject, requested subject property, required QoS level, required QoS parameters*.

In the proposed mapping of Table 2, *Requested subject* maps to the newly introduced link format attribute *entity (ent)*, whereas *requested subject property* is mapped to the *resource type (rt)* attribute. Moreover, *required QoS parameters* are mapped to the defined *sqos* and *dqos* attributes in CoRE link format. Finally, *required QoS level* is mapped to a *semantic threshold (sr)* attribute similar to that defined in (Ruta et al., 2013).

Table 2: User Request Description Mapping to CLF.

Request attributes	CoAP attributes	Attribute type
Required QoS Level	Semantic threshold (**sr**)	Standard attributes
Requested subjet param.	Resource Type (**rt**)	
	Observable (**obs**)	
Requested subject	Entity (**ent**)	Added attributes
Required QoS Param.	Service QoS (**sqos**)	
	Device QoS (**dqos**)	

Using these attributes, a user requesting the "*temperature*" of the "*kitchen*" with high energy level, very high reliability, a medium response time and a low energy consumption as QoS preferences can retrieve good matches above a threshold of 0.6 by issuing the following CoAP query:

```
coap://adressRD? ent="kitchen"; rt="temperature"; dqos=
"Energy_level: high | Reliability: very high"; sqos=
"Response_Time:medium | Energy_Cost: low" ; sr="0.6"
```

5 PROPOSED ARCHITECTURE

Having presented our model for extending CoAP's resource discovery with semantics, we have designed and implemented a framework integrating the proposed semantic model along with its representation in CoAP. The architecture of such a framework, shown in Figure 3, is structured around five main components namely: *Resource Directory* (*RD*), *Device*, *Border Router*, *User Interface* and *Reference Ontologies Server (ROS)*. *RD*, *Device* and *Border Router* components are implemented under Contiki 3.0 OS using Cooja simulator whereas *User Interface* and *ROS* are implemented respectively in standards web browser and web server.

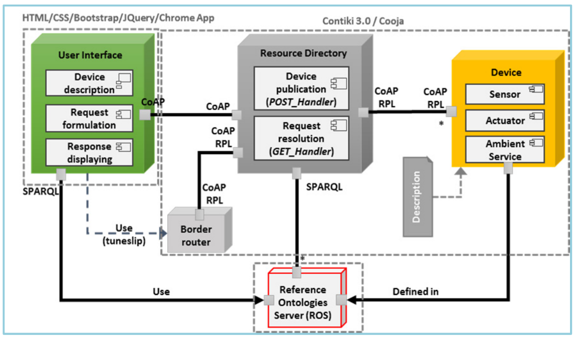

Figure 3: Framework for Resource Description, Discovery and Retrieving.

Firstly, an *RD* is a web entity used as a repository that stores information about web resources (devices and services) and implements two main REST interfaces *POST_Handler* and *GET_Handler*. The first interface is dedicated for resources registration whereas the second one is dedicated for lookup of those resources using the R3 rule with SPARQL (see Section 3.2).

Secondly, a *device* in the proposed framework can be a sensor and/or an actuator with resources encapsulated as ambient services. A device can send to the RD either its description via POST or a request through a GET message. Thirdly, the *Border Router* is a node that relies a PC or a Smartphone to the network through a Serial Line Internet Protocol (SLIP) (Romkey, 1988) or any available network interface. Inside the constrained network, packets are routed using RPL (Winter, 2012) or any routing protocol deployed within the network.

Fourthly, *Reference Ontologies Server (ROS)* is a web server that stores and provides access to all the reference ontologies, namely: *ARO, DRO, PRO, SRO and RO*. Information from such ontologies can be retrieved or updated using the SPARQL language. An example scenario using concrete ontologies is given in (Yachir et al., 2016a).

Finally, *user interface* is developed based on the Copper plugin (Copper CoAP, 2016), which is a CoAP agent for Firefox. In this work, we have developed two new plugins/apps for Google Chrome and Android OS allowing seamless interactions between users and objects using smartphones. Such applications are developed using HTML5/CSS3, Bootstrap, JQuery and Chrome APIs. The developed application provides dedicated interfaces for both device registration and user request specification. The developed application and a simple use case scenario implemented in Contiki OS using Cooja are reported in the following link: *https://www.dropbox.com/s/cnbyzpzmarrw5f6/ChromeAndroidCopper.mp4*

6 PERFORMANCE EVALUATION

6.1 Experimental Model and Performance Metrics

The proposed semantic model over CoAP is abbreviated *SemRD* for *Semantic Resource Directory*. To evaluate the performance of such a model, we implemented it in Contiki OS. To put results into context, *SemRD* was compared with the standard RD solution. A scenario comprising a simulated network composed of thirty (30) emulated Sky motes and a resource directory was used in our evaluations. The topology is depicted in Figure 4. Each mote can play the role of either a client requesting available resources from the RD/*SemRD* or a provider registering its resources at the RD/*SemRD* or both roles. In all cases, a node is considered aware of the RD before accessing it.

For the sake of this evaluation, we developed an application running the two RD solutions above UDP in a constrained 6LoWPAN network in Contiki. At the routing layer, the RPL protocol is deployed to ensure routing between RDs and other network nodes. At the link layer, simulations are conducted both with the ContikiMAC (Dunkels, 2011) Radio Duty Cycling (RDC) protocol and without using RDC (NullRDC). Simulation configuration parameters are given in Table 3.

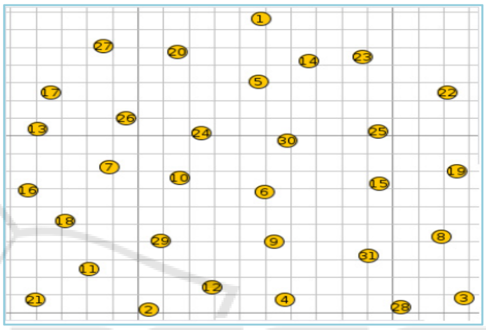

Figure 4: Simulated network topology.

To be able to draw conclusions on the time/cost performance of evaluated approaches, *Registration time* for devices, *Response time* for requests, Average *Response size* and the *Radio duty cycle*, as a proxy of energy consumption, were measured when varying the number of clients and providers. Registration (response) time is measured as the time spent from sending a description (request) until receiving a confirmation (response), averaged over all successful registrations (requests). The packet size is measured at the routing layer and averaged over all sent packets per node. The network duty cycle, as an indicator of energy consumption, is measured using Contiki's power profiler (Dunkels et al., 2011).

Table 3: Simulation configuration parameters.

Parameter	**Value**
Number of nodes	31
Type of nodes	Wisemote
Network	300m × 300m
Simulation time	600 secondes
Routing protocol	RPL
Transmission range	60 m
RDC / MAC / adaptation layer	ContikiMAC Null RDC / CSMA / 6LoWPAN
OS	Contiki 3.0/Cooja

6.2 Results and Discussion

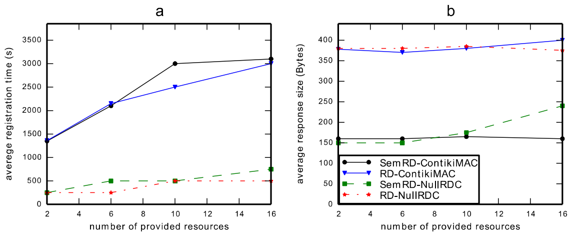

Figure 5: Performance when varying provided resources (number of clients = 5).

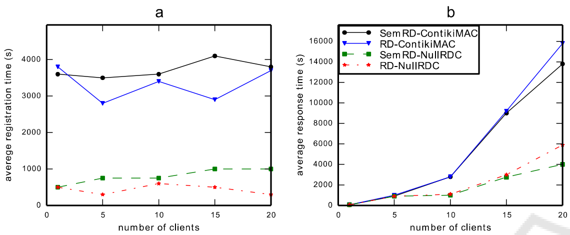

Figure 6: Performance when varying number of clients (number of resources = 16).

As we can see in Figure 5.a, the average registration time of a resource increases with the number of provided resources in the network. This increase is more visible in networks deploying RDC. When comparing the two approaches, *SemRD* takes more time in resource registration. This is because *SemRD* packet size is bigger than that of RD, due to the embedded semantic attributes. However, when it comes to the response size (Figure 5.b), *SemRD* showed a clear amelioration, with responses of about half the size of that of RD, thanks to the embedded semantics that allowed fine-grained filtering of responses. This in turn translated into a decrease in response time (results not showed for space reasons).

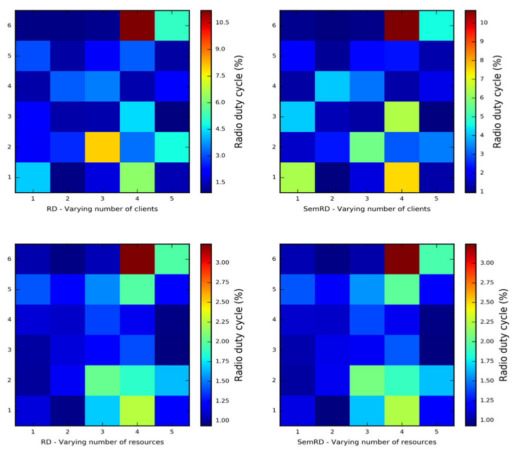

Figure 7: Energy consumption (duty cycle).

Figure 6.a and 6.b present the average registration and response times of the evaluated solutions when varying the number of clients. These figures show a similar behavior to that of Figure 5. Indeed, while the registration time is fairly independent from the number of clients (Figure 6.a), *SemRD* registered higher registration times than RD because of the additional semantic attributes. This, however, has the advantage of lowering response times, achieved by *SemRD,* because of minimizing their sizes as can be seen in Figure 6.b. It should be noted, from this figure, that response times of both SemRD and RD increase with the number clients.

Finally, Figure 7 presents energy consumption topography of *SemRD* and RD when varying both providers and clients. As can be seen in this figure, nodes' duty cycles are equitably balanced (similar color-heat per node) for both RD and *SemRD*. The higher heats (yellow to red) observed in some nodes can be due to their location (i.e. neighboring the RD node). Overall, Figure 7 clearly show the load balancing aspect of both approaches along with the lower duty cycle (lower energy consumption).

7 CONCLUSION

This paper proposed a concrete lightweight mapping and implementation of our previous semantic model (Yachir et al., 2016a) in CoAP. This mapping is achieved by defining appropriate CoRE Link Format attributes describing both IoT devices/resources and user requests. An RD-centered framework was also designed to facilitate IoT resources publication and retrieving using appropriate user interfaces communicating through semantic-enhanced CoAP. Simulation results have shown the performance of such mechanism when compared with the default RD solution.

Future work consists on more simulations and testbed experiments to validate the model at a larger scale. Enhancing the CoAP mapping along with considering other description formats is also planned. Currently, the authors are actively working to integrate, in the proposed semantic model, the publish/subscribe mechanism.

REFERENCES

Atzori, L., Iera, A., Morabito, G., 2010. The Internet of Things: A survey. In *Computer Networks 54 (2010) 2787–2805.*

Ayari, N., Chibani, A., Amirat, Y., Matson, E. T., 2015.

A novel approach based on commonsense knowledge representation and reasoning in open world for intelligent ambient assisted living services. In *Proc. IEEE/RSJ Int. Conf. Intell. Robot. Syst. (IROS'15), pp. 6007-6013.*

Barnaghi, P., Wang, W., Henson, C., Taylor, K., 2012. Semantics for the Internet of Things: Early Progress and Back to the Future. In *International journal on Semantic web and information systems. Vol. 8, Issue 1*.

Bonino, D., Procaccianti, G., 2014. Exploiting semantic technologies in smart environments and grids: Emerging roles and case studies. In *Science of Computer Programming 95, 112–134, Elsevier*.

Compton, M., Barnaghi, P., Bermudez, L., Garcı´a-Castro, R., Corcho, O., Cox, S., Taylor, K., 2012. The SSN ontology of the W3C semantic sensor network incubator group. In *Web Semant Sci Serv Agents World Wide Web 17:25–32*.

Copper CoAP, 2016. "Copper (Cu) CoAP user-agent - JavaScript CoAP Implementation."[Online]. Available: *http://people.inf.ethz.ch/mkovatsc/copper.php.* [Accessed: 21-Apr-2016].

De, S., Barnaghi, P., Bauer, M., Meissner, S., 2011. Service modelling for the Internet of Things. In *Federated Conference on Computer Science and Information Systems (FedCSIS), Szczecin, Poland.*

Dunkels, A., 2011. The ContikiMAC Radio Duty Cycling Protocol. *Swedish Institute of Computer Science (SICS), Technical Report T2011:13, ISSN 1100-3154.*

Dunkels, A., Eriksson, J., Finne, N., Tsiftes, N., 2011. Powertrace: Network-level power profiling for low-power wireless networks. In *Swedish Institute of Computer Science,* Technical Report T2011:05.

Fielding, R. T., Taylor, R. N., 2002. Principled design of the modern web architecture. In *ACM Trans. Internet Technol. 2 (2) 115-150.*

Han, S. N., Crespi, N., 2017. Semantic service provisioning for smart objects: Integrating IoT applications into the web. In *Future Generation Computer Systems, http://dx.doi.org/10.1016/j.future.2016.12.037*

ISTAG, 2003. Ambient intelligence: from vision to reality. In *Eur Comm Rep 1–31*.

Jara, A. J., Olivieri, A.C., Bocchi, Y., Jung, M. Kastner, W., Skarmeta, A.F., 2014. Semantic Web of Things: an analysis of the application semantics for the IoT moving towards the IoT convergence. In *Int. J. Web and Grid Services*, Vol. 10, Nos. 2/3, pp.244–272.

Kovatsch, M., Hassan, Y.N., Mayer, S., 2015. Practical Semantics for the Internet of Things: Physical States, Device Mashups, and Open Questions. In *5th IEEE International Conference on the Internet of Things (IoT), Seoul, South Korean, P 54-61*.

Roffia, L., Morandi, F., Kiljander, J., D'Elia, A., Vergari, F., Viola, F., Bononi, L., Cinotti, T. S., 2016. A Semantic Publish-Subscribe Architecture for the Internet of Things. In *IEEE Internet of Things Journal, DOI 10.1109/JIOT.2016.2587380.*

Romkey, J. L., 1988. Nonstandard for transmission of IP datagrams over serial lines: SLIP.

Ruta, M., Scioscia, F., Pinto, A., Di Sciascio, E., Gramegna, F., Ieva, S., Loseto, G., 2013. Resource annotation, dissemination and discovery in the Semantic Web of Things: a CoAP-based framework. In *IEEE Internet of Things and Cyber, Physical and Social Computing. DOI 10.1109/GreenCom-iThings-CPSCom.*

Shelby, Z., 2012. CoRE Link Format. *IETF RFC 6690.*

Shelby, Z., Hartke, K., Bormann, C., 2014. The Constrained Application Protocol (CoAP). *IETF RFC 7252*.

Shelby, Z., Koster, M., Bormann, C., Van Derstok, P., 2017. CoRE Resource Directory. *Internet Draft IETF.*

Spalazzi, L., Taccari, G., Bernardini, A., 2014. An iot ontology for earthquake emergency evaluation and response. In *International Symposium on Collaborative Technologies and Systems (CTS), IEEE*.

Sun, Y., Jara, A. J., 2014. An extensible and active semantic model of information organizing for the Internet of Things. In *Pers Ubiquit Comput, 18:1821–1833. DOI 10.1007/s00779-014-0786-z. Springer-Verlag London.*

Suresh, P., Daniel, J. V., Parthasarathy, V., Aswathy, R. H., 2014. A state of the art review on the Internet of Things (IoT) history, technology and fields of deployment. In *Proc. IEEE Int. Conf. Sci., Eng. Manage. Res. (ICSEMR'14), Chennai, India, pp. 1-8*.

Taccari, G., Bernardini, G., Spalazzi, L., D'Orazio, M., 2015. Earthquake Emergencies Management by Means of Semantic-Based Internet of Things. *In IoT360, Part II, LNICST 151, pp. 318–327. DOI: 10.1007/978-3-319-19743-2 43.*

Urbieta, A., Gonz´alez-Beltr´an, A., Ben Mokhtar, S., Anwar Hossain, M., Capra, L., 2016. Adaptive and context-aware service composition for IoT-based smart cities. In *Future Generation Computer Systems, http://dx.doi.org/10.1016/j.future.2016.12.038*

Verborgh, R., Haerinck, V., Steiner, T., Van Deursen, D., Van Hoecke, S., De Roo, J., Van de Walle, R., Vallès, J. G., 2012. In Functional Composition of Sensor Web APIs. *In Proc. SSN, Boston, USA*.

Wang, F., Hu, L., Zhou, J., Hu, J., Zhao, K., 2015. A semantics-based approach to multi-source heterogeneous information fusion in the internet of things. In *Soft Comput. DOI 10.1007/s00500-015-1899-7, Springer-Verlag.*

Weiser, M., 1991. The computer for the 21st century. In *Sci Am 265:94–104*.

Winter, T., 2012. RPL: IPv6 routing protocol for low-power and lossy networks.

Yachir, A., Djamaa, B., Mecheti, A., Amirat, Y., Aissani, M., 2016a. A comprehensive semantic model for smart object description and request resolution in the internet of things. In *The 7th International Conference on Ambient Systems, Networks and Technologies (ANT 2016). Procedia Computer Science 83, 147 – 154*.

Yachir, A., Amirat, Y., Chibani, A., Badache, N., 2016b. Event-Aware Framework for Dynamic Services Discovery and Selection in the Context of Ambient Intelligence and Internet of Things. In *IEEE Trans. Automation Science and Engineering 13 (1): 85-102.*

Yuan, Y., Wu, X., Lu, Y., 2013. A Flexible Framework for Representations of Entities in the Internet of Things. In *ISCTCS, CCIS 320, pp. 295–302.*