








Enabling the Interconnection of Smart Devices Through Semantic Web Techniques

Daniel Flores-Martin¹(✉) , Javier Berrocal¹ , José García-Alonso¹ ,
Carlos Canal² , and Juan M. Murillo¹ 

¹ Universidad de Extremadura, Badajoz, Spain
{dfloresm, jberrolm, jgaralo, juanmam}@unex.es
² Universidad de Málaga, Málaga, Spain
canal@lcc.uma.es

Abstract. Nowadays, there are millions of devices connected to the Internet. This is what we know as called Internet of Things. The integration of these smart devices with the web protocols makes them more accessible and understandable by people. The purpose of these devices is to make people's lives easier. Thanks to the collaboration between devices, the possibilities that the Web of Things offers can be even more exploited. However, many manufacturers develop their own devices and protocols in order to protect their market share, limiting in many ways the collaboration between devices of different manufacturers. This paper presents a solution based on semantic web techniques with the purpose of achieving collaboration between devices regardless of the technologies and protocols developed by their manufacturers.

Keywords: Web of Things · Semantic web · Collaboration

1 Introduction and Motivations

Internet of Things (IoT) and Web of Things (WoT) main aim is making people's lives simpler. While IoT allows us to remotely monitor and control smart devices, WoT integrates Things with the Web even more intimately; hence, making those devices more accessible for applications and humans [3]. These paradigms are used in many different domains as smart home, automotive, smart cities, health-care, agriculture, wearable, etc. [2]. The great evolution of these technologies allows us to develop increasingly smart devices and truly useful services that a few years ago were unthinkable.

But the real potential of WoT comes from the collaboration among smart devices to perform complex tasks. For this to be possible, the next evolution of WoT is to ensure that smart devices can proactively collaborate among themselves [1, 5, 6].

Unfortunately, the possibility of collaboration among smart devices is still far from being realized. Indeed, manufacturers develop their own protocols so

that their devices can be seamlessly integrated with each other but not with other brands' devices. This allows manufacturers to save their market share. This also inevitably leads to the well-known *vendor lock-in* problem [4]. This phenomenon implies that if one want to obtain the maximum benefit from the WoT, (s)he must purchase devices from the same manufacturer to ensure maximum compatibility. Consequently, the interaction between devices from different manufacturers is made more difficult, or they will have to give up certain activities to be assisted by these devices.

In this paper we propose a tool that relies on web semantic techniques to achieve the collaboration among smart devices regardless of the manufacturer. We achieve this collaboration by providing smart devices with *goals* (desired states of the environment) and *skills* (abilities to influence or change the environment). These goals and skills are defined in semantic web terms and related by semantic reasoners and query languages. Therefore, the possibility of collaboration between devices is fostered while maintaining the independence of the manufacturer, without forcing any device or manufacturer to use any specific technology, in a simple way, at low cost and effectively.

The rest of the document is structured as follows. Section 2 details our proposal to interconnect different smart devices regardless of the manufacturers. Finally, in Sect. 3 some conclusions are detailed.

2 Proposal

Manufacturers usually develop devices based on their own protocols, making it difficult to integrate with devices from other manufacturers. When a manufacturer develops a device, it is develop specific applications getting associated with an API that is often distributed to developers so they can get the most out of the device and provide services. Each device has its own features that are the functionalities that are able to change the state of the context, such as varying the temperature or brightness of a room, and that are described by the manufacturers themselves. In this work we reuse these APIs to generate the different skills giving them a semantic connotation so that they can be easily interpreted and related to other skills or goals. Thus, skills can be generated regardless of the technology used by the manufacturer and offered as services to other devices in the environment that need to use them. Below the architecture that models the process of converting features into skills and offering them services between WoT devices is shown, as well as a prototype.

2.1 Proposed Architecture

The architecture has a main component, the controller, composed by several modules, the *API Consume*, *Skills and Goals Generator* and the *Skill Service*. Figure 1 shows the complete process to convert devices features into skills: first (1) APIs are consumed to discover the features of the entities, second (2) once

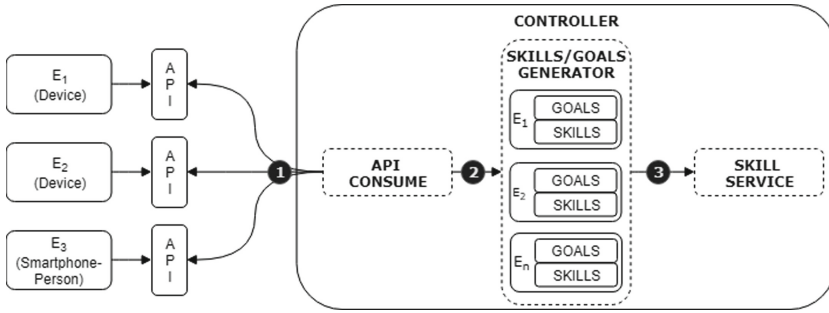


Fig. 1. Architecture for skills and goals conversion

the API has been explored, the skills are generated from the detected features, and finally (3) the skills are offered as services to be used by the rest of devices.

At this point, the controller knows all information about the entities, so it builds its ontology with this data. The purpose of this ontology is to relate the discovered entities through their skills and goals. The resolution of the goals of a given entity is to find a skill that is capable of solving it adequately. In order to obtain additional information about the entities and search for skills and goals more accurately, we use reasoners to detect what types of entities we can find within the context. After that, the entities mapping is done through SPARQL queries. Regarding the goals, these are currently defined manually to show the viability of the architecture. This will be done automatically in the future.

2.2 Use Case

To evaluate the feasibility of the proposed architecture, a prototype focused on a smart office environment is being developing. This use case has been chosen because it is being carried out in our laboratory.

The smart office has a variety of smart devices from different manufacturers, such as a smart air conditioning and a smart illumination. Both smart devices are connected to the office router. In order to integrate the devices with the defined architecture we have installed a Raspberry Pi. This device is able to manage the entities connected to the same network, both devices and people. The Raspberry Pi will be in charge of managing them, so it takes the role of the controller module in the previous defined architecture. In order to validate the viability of the proposal, the devices present in the network must be specified in this controller. In addition, when employees arrive to the office they must also be incorporated to the knowledge base of the controller so that it has evidence of them. The goals of the entities are specified manually through the controller’s web interface, while the skills are built from the information in their APIs and incorporating extra information from external ontologies. In Table 1 we show an example of the SPARQL query to obtain all goals related to temperature. In this case, the air conditioner is discovered as well as all its goals and skills.

Table 1. SPARQL query for “temp” solution

All present skills that cover the “temp” goal

```

1 SELECT * WHERE {
2   ?Goal a :Goal .
3   ?Goal :requiresSkill ?skill .
4   ?Goal rdfs:comment ?comment .
5   FILTER (CONTAINS( lcase( str(?Goal) ) ,
6     lcase("temp" ) ) ) .
7   ?Entity a :Entity .
8   ?Entity :hasSkill ?skill . }

```

Result

Goal	g_setTemp
Skill	sk_setTemp
comment	Change temp
Entity	airconditioner

To validate the proposed architecture the definition of the skills and goals has been carried out manually. As future work this will be done automatically.

3 Conclusions

In this paper we addressed the interconnection problems to favor the collaboration of smart devices of the WoT. The proposed architecture uses the semantic web for the interconnection of devices and is validated through a prototype. The prototype is a proof of concept that allows us to understand the interconnection limitations and possibilities in a smart scenario. This work is a another step towards achieving a higher level of interconnection in WoT.

Acknowledgments. This work was supported by the Spanish Ministry of Science and Innovation through projects TIN2015-69957-R, TIN2014-53986-REDT, and TIN2015-67083-R (MINECO/FEDER, UE), by the Department of Economy and Infrastructure of the Government of Extremadura (GR15098, IB18030), by the European Regional Development Fund (ERDF) and by 4IE project (0045-4IE-4-P) funded by the Interreg V-A España-Portugal (POCTEP) 2014-2020 program.

References

1. Al-Fuqaha, A., Guizani, M., Mohammadi, M., Aledhari, M., Ayyash, M.: Internet of Things: a survey on enabling technologies, protocols, and applications. *IEEE Commun. Surv. Tutor.* **17**(4), 2347–2376 (2015)
2. Gubbi, J., Buyya, R., Marusic, S., Palaniswami, M.: Internet of Things (IoT): a vision, architectural elements, and future directions. *Future Gener. Comp. Sy.* **29**(7), 1645–1660 (2013)
3. Guinard, D., Trifa, V.: Building the Web of Things: With Examples in Node.js and Raspberry Pi. Manning Publications Co., Greenwich (2016)
4. Roman, R., Zhou, J., Lopez, J.: On the features and challenges of security and privacy in distributed Internet of Things. *Comput. Netw.* **57**(10), 2266–2279 (2013)
5. Taivalsaari, A., Mikkonen, T.: A roadmap to the programmable world: software challenges in the IoT era. *IEEE Softw.* **34**(1), 72–80 (2017)
6. Yafei, D., Guanyu, L., Hui, Z.: Semantic space-based semantic collaboration method in semantic Web of Things. *Comput. Appl. Softw.* **2**, 002 (2016)