Integrating the Node-RED server in an IoT platform for ECA rules management

Giuseppe Desolda[a], Francesco Greco

[a] *University of Bari, Via Orabona 4, 70125 Bari, Italy*

Abstract

EFESTO-5W is a Web platform that implements a visual paradigm for allowing non-technical users to define the behavior of smart objects. This paper presents the integration of the Node-RED server in EFESTO-5W to make more robust, efficient, and reliable the management of ECA rules.

Keywords

Internet of Things, End-User Development, Tailoring Environments

1. Introduction

The Internet of Things (IoT) has emerged in the last 10 years as a new paradigm in modern wireless telecommunications. The basic idea is the pervasive presence of a variety of devices, the so-called smart objects, which can be worn, installed at home, guided, etc., and that can interact with people and each other. However, research in the field of IoT has focused mainly on technical aspects, for example, how to program networks of sensors and actuators and how to ensure their interoperability.

Despite the evident advantages of IoT, there are still some issues to be resolved to increase its impact and bring out its full potential. Although smart objects cover many user needs, people often feel the need to use them in a personalized way or to synchronize their behavior to solve situational needs. However, this activity today is performed by programming the smart object's behavior, thus requiring programming skills that non-technical people, which are the majority of IoT users, do not have. To enable all IoT users to use smart objects more flexibly and effectively, there is a need for technological and methodological solutions that support also non-technical users in defining the behavior of smart objects. To meet these needs today there are already some websites or mobile app (e.g. IFTTT [6]) that, offering visual mechanisms, support the definition of synchronizations between smart objects (e.g., if a volumetric sensor detects a human in the room then the lamp is turned on). However, these tools only support simple customizations and they are not very useful in real scenarios and everyday situations.

In order to directly involve non-technical users in a meaningful configuration of their smart objects, it is necessary to devise new approaches, based on high-level abstractions and appropriate interaction paradigms. In the last years, EFESTO-5W has been proposed as a web platform that supports non-technical users in synchronizing the smart object's behavior by visually defining Event-Condition-Action (ECA) rules (see for example **Figure 1**) [2; 3]. Despite EFESTO-5W has been developed as a professional prototype, some technical aspects go beyond the platform itself and deserve a strong effort during the platform life. In particular, since EFESTO-5W requires the registration of smart objects and web services, one of the most important challenges is the heterogeneity of the technologies and related network protocols (e.g., HTTP, MQTT, web socket, QoS, CoAP), security access mechanisms (OAuth, OAuth 2.0), and data format/syntax. In the previous version of EFESTO-5W, this challenge has been solved by implementing different modules that manage each aspect. However, during the time the smart objects manufacturers update and improve their protocols, security access mechanisms, add or remove

EMPATHY: Empowering People in Dealing with Internet of Things Ecosystems. Workshop co-located with AVI 2020, Island of Ischia, Italy
EMAIL: giuseppe.desolda@uniba.it (G. Desolda); f.greco66@studenti.uniba.it (F. Greco)
ORCID: 0000-0001-9894-2116 (G. Desolda)

some functionalities, change the data format or syntax, thus requiring a continuous refactoring of the smart objects catalog and of the platform modules and components.

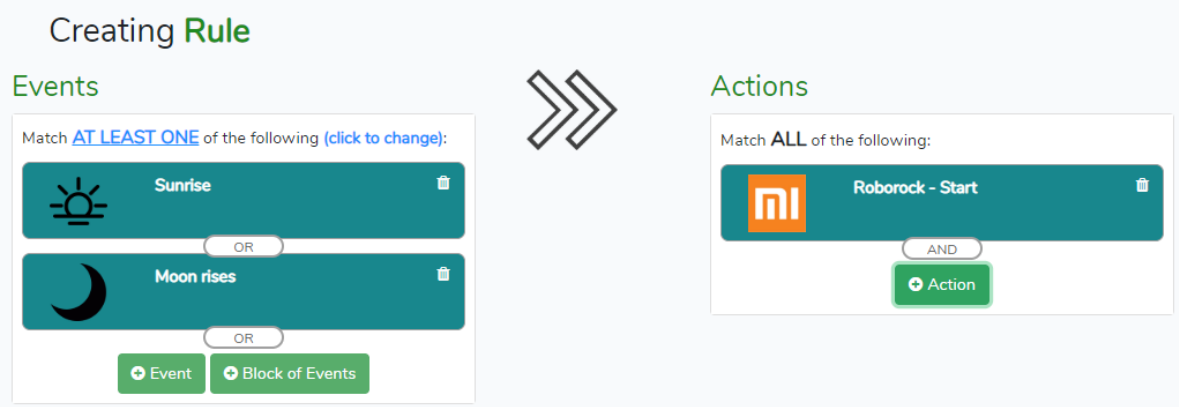

Figure 1: Example of ECA rule created by using EFESTO-5W: the Roborock vacuum cleaner starts the house cleaning when the sun rises (in the early morning) or when the moon rises (in the night).

In this paper a new version of EFESTO-5W is proposed: it overcomes the aspects underlined above delegating all the related issues to a third-party application that controls the smart objects and governs the ECA rules behaviors. Different technological solutions have been evaluated, for example Node-RED, crosser, n8n.io, ioBroker, AWS IoT, Zenaton, SiteWhere.io considering different requirements: open-source project, reliable and fast application, installable on a web server, availability of a wide catalog of web services and smart objects, management of workflows that can be shaped as EFESTO-5W ECA rules, availability of APIs to export/import services and ECA rules to EFESTO-5W. Considering all these constraints, Node-RED has been selected as the best candidate. In the next section, the most important details of this integration are reported.

2. Empowering EFESTO-5W with the Node-RED server

Node-RED is an open-source programming tool for wiring together hardware devices, APIs and online services by using a visual paradigm [8]. Its wide community daily adds, updates and improves a catalog of thousands of smart objects and web services. The Node-RED platform is written in NodeJS and it is mainly composed of 1) a web-based user interface that supports the visual creation of the flows and 2) a web server that manages the user-defined flows. However, it has been demonstrated that the graph metaphor proposed by Node-RED UI is not suitable for non-technical users [7]. In addition, the visual creation of flows often requires technical aspects and script coding that can be managed only by programmers.

In order to combine the powerful visual metaphor of EFESTO-5W with the robust, efficient, and reliable features of the Node-RED server, we recently integrated the two systems. In this way, both the smart objects' life cycle and ECA rule execution management are delegated to the Node-RED server still maintaining the easy composition paradigm proposed in EFESTO-5W.

The integration of the two systems has required the development of new modules in EFESTO-5W. First, since smart objects and web services are codified in Node-RED as JSON strings, the EFESTO-5W smart objects catalog is based on these JSON descriptors. As shown in the system architecture reported in **Figure 2**, in the *Service Layer* there is a *Service Descriptor* repository that now stores all the web services and smart objects descriptors codified by using the Node-RED syntax. Second, the *Rule Generator* in the *Interaction Layer* has been modified so that every time an ECA rule is visually defined by a user it is translated into a JSON string representing the ECA rule according to the Node-RED workflow syntax. In addition, since some web services and smart objects included in the rule can use authentication credentials, such data are not immediately included in the JSON file since they would be exposed to attacks like man-in-the-middle. When this JSON file is sent to the Logic Layer installed on a Web server, in particular to the *Rule Manager* component, it is completed with all the missing authentication data safely stored in the server database. When completed, this file is sent to the Node-

RED server that immediately executes it. From that moment on, Node-RED is in charge of the rule management, with all the related aspects like the event listening, communication protocols, authentications mechanisms, etc.

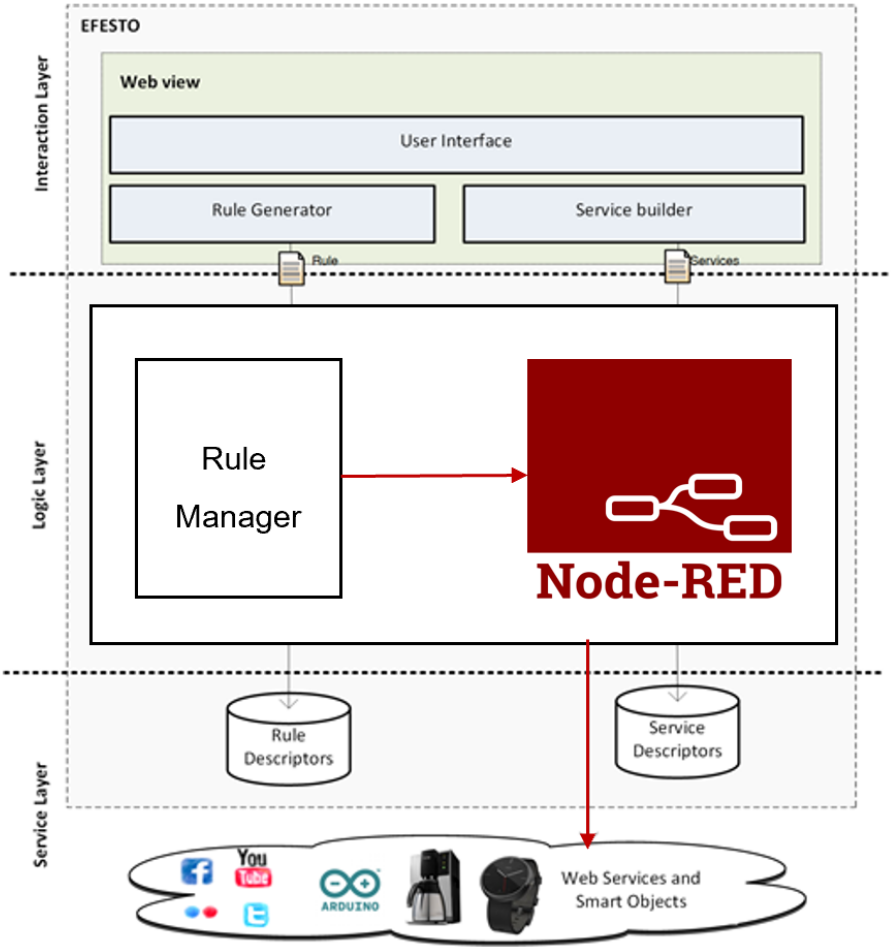

Figure 2: The new architecture of EFESTO-5W integrated with the Node-RED server.

3. Conclusion and Future Work

This paper has presented the new version of the EFESTO-5W platform, which aims to become more reliable, effective, stable, and efficient, and to reduce the maintenance effort thanks to the integration of the Node-RED server. This new version of EFESTO-5W will be used in real contexts, in particular in the EMPATHY PRIN project, which aims to investigate if and how domain experts, even without skills in IT, are able to define smart environments by using tools like EFESTO-5W. Different domains will be considered in this project, for example cultural heritage involving professional guides that will define smart museums, or Ambient Assisted Living involving caregivers that will design smart homes, for example orchestrating the behavior of conversational agents and other smart objects that help patients with Parkinson's disease [5].

As future work, new interaction paradigms for ECA rule definitions will be investigated, for example exploiting tangible programming and/or cross-device interaction mechanisms [4]. In addition, since this tool will be used by end users in real contexts, to improve its internal qualities we are adopting the GQM-based approach presented in [1].

This workshop could be an ideal venue to present the new advances of EFESTO-5W and to discuss with other researchers pro and cons of the proposed approach.

4. Acknowledgements

This work is partially supported by the Italian Ministry of University and Research (MIUR) under grant PRIN 2017 “EMPATHY: EMpowering People in deAling with internet of THings ecosYstems”.

5. References

[1] Ardimento, P., Baldassarre, M.T., Caivano, D. and Visaggio, G. (2004). Multiview Framework for Goal Oriented Measurement Plan Design. In *Proc. of the International Conference on Product Focused Software Process Improvement (PROFES '04)*. Springer Berlin Heidelberg, 159-173.

[2] Ardito, C., Desolda, G., Lanzilotti, R., Malizia, A. and Matera, M. (2019). Analysing Trade-offs in Frameworks for the Design of Smart Environments. *Behaviour & Information Technology*, 39(1), 47-71.

[3] Balducci, F., Buono, P., Desolda, G., Impedovo, D. and Piccinno, A. (2020). Improving smart interactive experiences in cultural heritage through pattern recognition techniques. *Pattern Recognition Letters*, 131, 142-149.

[4] Desolda, G., Ardito, C., Jetter, H.-C. and Lanzilotti, R. (2019). Exploring spatially-aware cross-device interaction techniques for mobile collaborative sensemaking. *International Journal of Human-Computer Studies*, Volume 122(February 2019), 1-20.

[5] Dimauro, G., Nicola, V.D., Bevilacqua, V., Caivano, D. and Girardi, F. (2017). Assessment of Speech Intelligibility in Parkinson’s Disease Using a Speech-To-Text System. *IEEE Access*, 5, 22199-22208.

[6] IFTTT Inc. *IFTTT*. Retrieved from https://ifttt.com/ Last Access June 10, 2019.

[7] JS Foundation. *Node-RED*. Retrieved from http://nodered.org/ Last Access February 28th, 2020.

[8] Namoun, A., Nestler, T. and De Angeli, A. (2010). Conceptual and Usability Issues in the Composable Web of Software Services. In: F. Daniel and F. Facca (Eds.) *Current Trends in Web Engineering - ICWE 2010* Lecture Notes in Computer Science, Vol. 6385, Springer Berlin Heidelberg, 396-407.