Position on Interoperability Everywhere under IoT-ARM

Adriano Tavares[1], Fausto Giunchiglia[2], Hao Xu[3], and Yanchun Liang[3]

[1]DEI, University of Minho, 4800058, Portugal
[2]DISI, University of Trento, 38100, Italy
[3]CCST, Jilin University, Changchun, 130012, China
atavares@dei.uminho.pt
fausto@disi.unitn.it
{xuhao,ycliang}@jlu.edu.cn

Abstract. This position paper argues that accelerating the use of IoT-ARM (Architectural Reference Model) on new IoT-Systems' realizations requires semantic interoperability to more than architectural, device and connectivity levels but also at tool-, system stack-, language- and work-flow management-level. In doing so, an IoT-ARM ontology is proposed which extends the conceptual model of IoT-ARM method with highly cohesive Methodology Mapping, Big Data Analytic, and Architecture Implementation Roadmap facets while leveraging cross- and intra-language interoperability.

Keywords: Semantic Interoperability, Multi-Lingual Interoperability, Big Data, Open Data, IoT, Ontology, IoT-ARM.

1 Introduction

Currently, we are entering the Internet of Things (IoT)-age with IoT-systems consisting of components like sensing, heterogeneous access, information processing and, applications and services. IoT movement relies on pervasive connectivity and intelligently connects humans, devices, and systems by integrating multiple technologies under a unified management platform. Architecturally, IoT follows a serviced-oriented model and it can be split into four tiers. Thing-tier for sensing and transmission, Intelligent System-tier (i.e., Fog-tier) for early-life data analysis, aggregation and transmission, Cloud-tier for early-life or at-rest science-data analysis and storing and, Application-tier for user access and control.

Driven by the heterogeneity of IoT-related ecosystems, several problem spaces have been identified like connectivity, architecting process, big data analytics, device intelligence and data technologies that must be overcome to achieve mainstream IoT adoption. As IoT spans various industries and use cases, embedded processing will demand scalable strategies while a limited scope of standards will coexist for a long time to come as one size will not fit all [1, 2]. IoT-ARM emerged as a possible answer to the IoT multiplicity issue and it started by creating the IoT Reference Model (IoT-RM) to promote a common understanding, followed by the IoT Reference Architecture (IoT-RA) that describes essential building blocks and design choices to deal with conflicting quality attributes like functionality, performance, deployment and security [3, 4]. IoT-ARM approaches a loosely-coupled interoperability at connectivity- and semantic-level and it relies on the semantic technology to apply interoperability at architectural-level through the IoT Domain Model and IoT Information Model. It also addresses connectivity interoperability using a service-oriented communication model leveraged on the ISO OSI 7-layer model and it aims at highlighting those peculiar interoperability aspects inherent to the interoperation among different stacks, which are called interoperability features. Furthermore, it builds variation points into the software, and uses standard extension points, e.g., using standardized protocols and gateways to enable brownfield deployment [3].

From our point of view, IoT-ARM presents some drawbacks that are decelerating its use on new IoT-Systems' realizations such as:

- It only partially addresses device technology issues as its focus is only on the software stack (i.e., it does not address the whole system stack).
- Its Architecting Process is too generic and so, several projects instead of following IoT-ARM from the ground up try to show at the end of their architecture realizations how do they map to the IoT-ARM (e.g., [5]).
- It does not explicitly promote the interplay of IoT with Big Data as its main focus goes to IoT applications for track, command, control and route (TCC&R) purposes.

2 Our Approach

Our approach aims at designing a modular ontology to assist in IoT-ARM Architecting Process (i.e., IoT-ARM ontology) as an instance of UKC [6, 7]. The UKC domain, concept, and entity type cores will be extended by terminology specific to a new field of study, in order to assist semantically the IoT-ARM Architecting Process which is denominated IoT-ARM ontology. Basically, IoT-ARM ontology is a synthesis of ontologies proposed in [8, 9], extended with the Methodology Mapping facet to accommodate IoT-ARM methodology agnosticism and Big Data Analytic facet to enable interplay of IoT with Big Data. Extending IoT-ARM with the idea of Everywhere Interoperability based on a scalable semantic schema built as an instance of UKC will leverage:

- ***Multi-Lingual Interoperability*** at both cross-language and intra-language levels. Cross-language interoperability among several languages will enable their entrance into the "Open Big Data Age" while intra-language interoperability allows the use of multiple terms denoting the same concepts or more specific/general terms.
- ***Declarative design of a semantic and scalable whole design stack*** for easy customization at each IoT tier and better addressing the increasing

intelligence, security, safety, communication, timeliness, area, and power issues. The Architectural Description facet is extended with specific sub-facets representing the required knowledge for co-design strategies and propagation effects among the stack layers while the Architecture Design facet with a technological design flow sub-facet for the whole stack design. All entities including tools (e.g., design, simulation, synthesis tools) are declaratively and semantically tagged for the purpose.
- ***Semantic collaborative system design chain*** according to known workflow reference model dictated by industry horizontalization. The Architecture Implementation Roadmap facet is extended with a business collaboration workflow sub-facet embedding industry chain management knowledge.
- ***Big Data Analytic Reference Architecture*** to model Big Data Analytics space problem in terms of several levels of heterogeneity involved on its architecting and design process, at analytical types, use cases scenarios, location of analytic technology, analytic techniques, type of actionable intelligence and visualization, sources and type of data, technological platforms, spectrum of analytical workloads, etc.

To support the proposed multi-level interoperability, we are using SCROLL NLP and UKC frameworks developed by the KnowDive team at University of Trento. Following the UKC's so-called faceted approach to ontological modeling, the IoT-ARM ontology is extended by a large number of concepts (e.g., Sw, Hw and Simulation components, views, tactics, design choice, perspectives, quality attributes, system, design stack, design flow), entities (e.g., Linux, Windows, FreeRTOS, OSGi framework, ARM Cortex-M3, MPSoC, Hadoop, Oracle, OpenStack, VMWare, Cassandra, Simulink, Modelica), and highly cohesive facets (e.g., Methodology Mapping, Architectural Description, Architectural Requirements, Architecture Design, Architecture Implementation Roadmap). SCROLL NLP has been extended to support several languages by collecting linguistic resources, adapting and integrating them into a processing pipeline.

Following standardization on ontology leverages abilities of a gradually growing IoT-ARM environments (i.e., by skipping out of the "Standard War"), mapping of different vendors/providers technologies to the IoT-ARM, tooling enablement from different vendors/providers to the IoT-ARM environment and ready-made ecosystem of partners, thus enabling IoT-Systems' realizations of several and different use cases scenarios. After populating IoT-ARM ontology with several catalogs of entities divided by categories and semantically tagged with their properties and constraints, tools for managing templates of system stack and development flow through functionalities such as creation, instantiation, configuration, validation and deployment are designed and implemented. A team of experts in a given IoT application domain can create templates for specific applications and use cases scenarios (i.e., enabling some kind of application guidance) and populate the associated catalogs after validation. Furthermore, they specify mapping strategies described as semantics rules to associate tactics to design choices and then add them to the IoT-ARM tactic and design choice catalog. Later a user can instantiate existing template seeds from the catalog for his/her new IoT system realization. If a template contains abstract components/tools, then the component/tool catalog is queried to find a valid bind.

Several reasoners will be implemented to: (1) reason about the design space, (2) assist in the creation of design flows and system stacks, (3) reason about components' constraints and tool's characteristics and propagate them through the development flow and system stack structures and (4) reason about the matching between a development flow and a system stack. After reasoning about the design space and identifying all valid instances of system stack, a virtual prototyping environment can be built to explore such solution space. Such environment is a specialization or specific instance of the development flow to carry out mixed-simulation, including the dynamics of the physical process using tools such as Simulink or Modelica which are possible entities of IoT-ARM ontology populated into the catalog of tools. According to the IoT-ARM architectural description, a reasoner will be provided for each qualitative requirement to find a perspective associated with it and also to select tactics based on the functional requirements and architectural constraints. Finally, semantics rules are proposed to prioritize the way that perspectives will be applied to views as not all perspectives have equal effect on all views.

3 Opportunities Addressed

By extending IoT-ARM ontology to be more focused on main IoT current issues faced by new IoT-related system realizations, such as, technical, architecture, hardware, privacy and security, standard and business challenges, the conceptual model of IoT-ARM method is improved by tackling its poor methodological completeness and application guidance as pointed in [10] while promoting:

- Multi-sourced and multi-lingual big data analytics supporting real-time open data.
- Interoperability at stakeholder-level while mitigating IoT market fragmentation and industry disjointed tooling ecosystem.
- 3C (Computing, Control and Communication)- and 3S (Scalability, Security and Safety)-convergences in the new 'Cloud + Edge' computing paradigm through a holistic collaborative design chain approach encompassing end-device, connectivity layer, gateway, and services running in the cloud with design metrics or quality attributes factored in at every level.
- Some level of automation and application guidance to the IoT-ARM architecting process by approaching semantic whole system stack as well as semantic design flow with interoperability at tool-level.

Acknowledgments. The authors would like to acknowledge the assistance of Drs. Gabor Bella, Feroz Farazi and Enzo Maltese at the University of Trento, and PhD students Sandro Pinto and Paulo Garcia at the University of Minho, for their useful discussions and efforts on writing related proposals. The first author is grateful to the support of FCT - Fundação para a Ciência e Tecnologia in the scope of the project: PEst-UID/CEC/00319/2013. The last two authors are grateful to the support of the NSFC (61272207, 61300147, 61472158).

References

1. Karimi, K., Atkinson, G.: What the Internet of Things (IoT) Needs to Become a Reality. In: Freescale White Paper (2013)
2. Xing, Z.H., Zhong, Y.F.: Internet of Things and its future. In: Huawei Communicate, Issue 54 (2010)
3. Bassi, A., Bauer, M., Fiedler, M., Kramp, T., Kranenburg, R., Lange, S., Meissner, S.: Enabling Things to Talk: Designing IoT solutions with the IoT Architectural Reference Model. ISBN 978-3-642-40403-0, Springer (2013)
4. Kim, Y.-W.: IoT Reference Architecture versus IoT Platform From Standardization Viewpoint, http://docbox.etsi.org/Workshop/2014/201407_IOTWORKSHOP/ETRI_KIM.pdf (2014)
5. Kiljander, J., D'Elia, A., Morandi, F., Hyttinen, P., Takalo-Mattila, J., Ylisaukko-Oja, A., Soininen, J.-P., Cinotti, T.S.: Semantic Interoperability Architecture for Pervasive Computing and Internet of Things. In: IEEE Open Access Journal, Volume 2 (2014)
6. Tawfik, A., Giunchiglia, F., Maltese, V.: A Collaborative Platform for Multilingual Ontology Development. Technical Report, Trento University (2014)
7. Ganbold, A., Farazi, F., Giunchiglia, F.: Managing Language Diversity across Cultures: the English-Mongolian Case Study. Technical Report, Trento University (2013)
8. Akerman, A., Tyree, J.: Using Ontology to Support Development of Software Architectures. In: IBM Systems Journal (45:4), pp. 813-825 (2006)
9. Ameller, D., Franch, X.: Ontology-Based Architectural Knowledge Representation: Structural Elements Module. In: Advanced Information Systems Engineering Workshops, Lecture Notes in Business Information Processing, Springer Berlin Heidelberg, pp. 296-301 (2011)
10. Nikitina, S.: Translating Qualitative Requirements into Design Choices: Evaluating the Method Proposed in the Architectural Reference Model for the Internet of Things. Master thesis, http://iotforum.org/wp-content/uploads/2014/10/Nikitina_S_Thesis.pdf (2014)