Model-Driven IoT

Muzaffar Artikov

Software Engineering Department
Urgench branch of Tashkent University of Information Technologies

April 18, 2018

Abstract. Smart technologies employ sensors, actors, communication protocols and home servers to increase the efficiency of city services and processes. Information and Communication Technology (ICT) helps optimizing the processes, while the Internet of Things (IoT) provides the platform for managing a multitude of small sensor and actor devices for smart technologies. Smart technologies promises not only a more efficient management of resources, but also an increase in the quality of services provided, while still remaining cost-effective.

From a technical point of view, "Smart X" systems can be seen as a large, sensor-based, distributed information system with data collection, data processing and data support components. Nowadays, smart systems and software architectures are reaching new levels of complexity, necessitating appropriate engineering methodologies. Model-Driven Software Engineering provides the required foundations for formally define generic architectures and development of software support on the top of IoT. To cope with ever growing complexity of such software architectures, model-driven approaches can extensively be applied to modeling of the IoT technologies as well as developing and maintaining software systems and platforms for "Smart X" technologies. As the models of these architectures and platforms become huge and complex over time, model-driven development is necessary to cope with development and evolution challenges arising from the IoT applications. This research proposal aims at (1) studying the state of the art in model-driven IoT, (2) extended research in model-driven IoT focusing on the "Smart X", (3) model-driven IoT for developing software architectures and platforms for IoT-based systems. As the proof of concepts, it further focuses on applying model-driven engineering, new upcoming trends in model-driven IoT, and its adoption in developing smart technology architectures.

1 Introduction

The internet of the present day is enriched with a huge amount of various devices and micro services. These devices and services operate for different purposes, yet work together to achieve common goals. The various types of sensors, actuators and devices are being developed to provide a range of services. These sensors and devices have created a new technological trend today. This trend has been named the "Internet of Things" [16], "Network of Things" [6] or "Web of Things" [7].

In the framework of IoT, more and more devices are equipped with network connectivity to autonomously provide "smarter" services, forming the Internet of Things (IoT). Applications are wide-ranging, and have variously been termed "Smart X", including Smart Homes, Smart Factories (Industry 4.0), Smart Government, Smart City, Smart Grid, Smart Traffic Control, and many more [10].

Smart technologies employ sensors and actors to increase the efficiency of services and processes, including environmental sustainability, energy efficiency, mobility, health care, safety, and security. ICT helps to optimize the processes, while IoT provides the platform for managing a multitude of small sensor, actor devices, communication protocols and home servers.

IoT [16] is the basic concept for developing the network of things equipped with built-in technologies for interaction with each other or with the environment. The concept was formulated in 1999 as an understanding of the prospects for the widespread use of radio frequency identification means for the interaction of physical objects with each other and with the external environment.

Definition 1.1 ***Internet of Things***
The Internet of Things is about installing sensors (RFID, IR, GPS, laser scanners, etc.) for everything, and connecting them to the internet through specific protocols for information exchange and communications, in order to achieve intelligent recognition, location, tracking, monitoring and management [1].

The concept of smart technologies arises from the need to manage, automate, optimize and explore all aspects of daily life that could be improved and optimized by information technologies. The software paradigm IoT, being a core concept behind smart smart technologies, is largely perceived as a collection of interconnected "things" within smart technologies.

The IoT-based smart applications are realized by interconnected systems of heterogeneous hardware, software, and embedded systems: these cyber-physical systems introduce new levels of complexity, requiring appropriate engineering methodologies to support formally rigorous software and systems development [10]. **Model-Driven Engineering (MDE)** provides fitting foundations and is considered as an enabling technology for advancing smart technology applications.

A collection of software and hardware components for smart systems can be viewed as a huge reference architecture based on model-driven software platform. Thus, a number of model-driven reference architectures [8] and models

[17] are introduced for smart system development. These architectures and models contemplate smart system architectures as a blueprint which provides an appropriate level of abstraction for the development process of smart systems. Model-driven reference architectures are used to represent and define different development aspects of smart system architectures consisting of different views, viewpoints, software services and components like home servers, communication protocols, sensors, activators, etc., in a single, huge architecture. For instance, several model-driven approaches are investigated in [3] and [15] utilizing different modeling language profiles (e.g. standard UML profiles [12]) in development of smart system architectures.

MDE is the modern day approach of software system development which supports well-suited abstraction concepts for development activities. It intends to improve the productivity of the design and development, maintenance activities, and communication among various actors and stakeholders of a system. As the main concept in MDE, models are well-suited for designing, developing and producing large-scale software projects. MDE brings several main benefits such as a productivity boost, models become a single point of truth [5]. Models are the main artifacts in MDE. They are well-suited for designing, developing and producing large software systems. Software models are the documentation and implementation of software systems [9].

This proposal focuses on the field of model-driven IoT for smart systems. Its objectives are manifold: (1) studying the state of the art in model-driven IoT, (2) extended research in model-driven IoT focusing on smart technologies, (3) applying MDE concepts to develop software architectures and platforms for IoT-based systems.

This research proposal is structured as follows: Section 2 motivates the research field defining its contributions and novelty. Section 3 investigates existing model-driven IoT approaches. The main research objectives of this proposal are sketched in Section 4. Section 6 discusses the main expected benefits of the research. This paper ends up in Section 7 with some conclusions.

2 Motivation

Realization of smart systems requires to understand the complex processes, so that proper decisions can be made based on available data. It is important to manage large and distributed networks of devices. These devices need to communicate, they need to be self-stable against errors and possible failures. In addition, they produce a lot of data that needs to be analyzed and stored. This requires to consider all aspects like above in modeling.

All challenges of smart distributed systems can be more or less eased by generic reference architectures [5], [6], [8]. From a technical point of view, all IoT-based smart systems can be treated as a large, sensor- and actor-based, distributed information system. These systems can then be generalized as a large, sensor-based software reference architecture [8] which include main activities like data collection by sensors, data processing by servers, control by activators and data support for actors.

MDE provides a collection of modeling concepts and the detailed separation of different views and concerns [8]. For instance, components have well-defined interfaces and ports. MDE benefits from its capabilities to model different, but integrated views and behavior of IoT systems which are eventually made executable for different smart platforms. The basic idea of the smart system vision is the pervasive presence of a variety of things or objects such as sensors, actuators, mobile phones, etc. which are able to interact with each other and cooperate with their pairs to reach common goals [1] [16].

3 State of Art

There are several MDE approaches for developing IoT applications, e.g. the Sirius-based ThingML language [4]. That existing model-driven approaches provide very expressive modeling of the IoT-based smart architectures, possibly with code generation. The motivation for model-driven development is to describe a system on a higher level of abstraction. This is usually done in UML and other languages by diagrams modeling specific aspects or views of model-driven architectures for smart systems.

MDE techniques are proposed to reduce the severity of IoT applications' development. In such an approach, applications are specified using high-level abstractions using models. These models are then used to produce deployable source code. For instance, PervML [13] enables developers to specify their software architectures at abstraction levels through a set of models (in UML).

Ciccozzi and Spalazzese introduced MDE4IoT [2], a MDE Framework supporting the modeling of Things and self-adaptation of Emergent Configurations of connected systems in the IoT-based smart systems. As the IoT systems consist of several connected software services and hardware components, there might be possible failures in performance of the overall system because of some non-responding devices. According to the article considers, in such cases the system should adapt to work and sustain without a need for these inactive devices, and re-install and maintain its activities. In order to avoid such failures, MDE4IoT is meant to exploit the combination of a set of domain-specific modeling languages to achieve separation of concerns.

The research presented in [3] uses the MDE principles to build a holistic development methodology involving a common, semantically expressive abstraction model, to specify a smart space with its specific services. It proposes the Resource-Oriented and Ontology-Driven Development (ROOD) methodology, which improves traditional MDE-based tools through semantic technologies for rapid prototyping of smart spaces according to the IoT paradigm. In the framework of ROOD, the Smart Space Modeling Language (SsML) was developed based on UML, that defines a Domain Specific Model (DSL). It can be used for describing high-level behaviors, interactions and context information of the entire smart space. It further defines the processing aspects related to the sensing and actuating capabilities of the smart objects, as well as the context model they manage; moreover, encapsulate these concepts into RESTful resources.

Patel et. al. [11] presents a multi-stage model-driven approach for IoT application development, based on identification of the skills and responsibilities of the various stakeholders involved in the process. The approach uses configurable modeling languages that are customized for a particular stakeholder task and application area, where abstractions available to a specific stakeholder are generated from information provided by other stakeholders at previous stages. The approach is complemented by methods for generating code and mapping tasks that lead to the deployment of node-level code on composite devices.

ThingML [5] is another domain-specific modeling framework built on Sirius. In ThingML, the state machine diagrams are used in several embedded domains to model the behavior of specific objects e.g. the discrete behavior of components. In the MDE paradigm of ThingML, the states of hardware components are managed by defining finite state machines.

4 Research Objectives

There are several domain-specific MDE approaches and tools for developing IoT-based architectures and applications. However, research in model-driven IoT is in its early stages and extended research in the field is required.

As explained in Section 3, there are already several domain-specific MDE notations and tools that can be used for designing, modeling and developing IoT-based smart system architectures and applications. They can be distinguished by different design aspects and perspectives such as views, viewpoints, components, communication protocols, etc. However, MDE is not fully utilized for developing model-driven IoT approaches.

The core objectives of this proposal are threefold:

- **Studying The State of Art.** This research initially indends to study the state of the art in model-driven IoT approaches in order to identify the MDE tools and approaches [5], [3] dedicated especially to develop the IoT-based smart systems. The most existing MDE approaches and tools for IoT-based software systems employ standard UML profiles developed on the top of EMF [14]. As long as these tools are open-source and their underlying UML meta-models of these approaches can be further developed and extended.

- **Research.** The most important, challenging and long-running part of this research focuses on extended research in model-driven IoT focusing on the smart technologies. This research aims to utilize the full potential of MDE in developing advanced model-driven IoT systems, applications and architectures.

- **Validation.** This research further focuses on applying a proposed approach to real-world applications as the proof of concept.

5 Proposed Approach

The IoT-based smart system architectures are very large software platforms and architectures. Thus, for the sake of simplicity and to achieve effective results, the large-scale smart system architectures can be developed using MDE trends. After their initial deployment, they further have to be maintained, consisted, evolved, yet remain flexible and sustainable. These efficient results can also be achieved using MDE trends.

In order to achieve efficient research results in model-driven IoT, research should cover several steps:

- **Meta-Model.** UML proposes a general modeling concepts for developing model-driven approaches. As long as UML can not directly applied to the IoT domain, the standard concepts of UML have to be extended and adapted according to the requirements of that domain. In the initial phase, this research proposes to develop a meta-model for developing model-driven IoT approach extending the standard UML profiles. This will be achieved by developing an adapter which can automatically extend standard UML profiles according to certain adaptation rules.

- **Tool.** This research further focuses on developing general model-driven application dedicated especially to IoT systems. It aims to support flexible and sustainable services such as model designing editor, model repository, etc. This application will be developed using the meta-model developed in the initial step.

6 Expected Benefits

In the course of this research, several benefits are expected to be achieved. This research starts by investigating the existing MDE approaches for IoT-based smart systems. The state of the art will be studied and documented for further research. The core research aims to bring several advantages to the field of model-driven IoT.

- **Adaptation:** Nowadays, MDE promises high-level model-driven technologies and concepts like UML. But, as long as IoT is a novel domain, the current potential of MDE is not fully utilized in developing IoT systems. As the result of this research, IoT systems benefit from the well-developed MDE concepts and technologies. To this end, standard UML profies will be extended and adapted to meet the requirements of IoT-based system development. In the resulting domain-specific meta-model, the standard UML profiles will be enriched with the domain-specific aspects of IoT-based smart systems.

- **Domain-Specific MDE Tool:** After developing a domain-specific meta-model for IoT, a domain-specific MDE tool will be developed based on that meta-model. The resulting domain-specific tool helps to design different perspective of IoT systems resulting in model-driven IoT systems.

While doing research, the results of research activities will be published and discussed among scientists and researchers in the field.

7 Conclusion

This proposal focuses on model-driven IoT. Initially, this proposal has briefly studied the state of the art in MDE approaches for IoT systems, applications and architectures. These discussions have shown that the research in the field of model-driven IoT is still in its early stages. So, extended research in the field of model-driven IoT is needed.

As noted, MDE offers great opportunities for development and support for IoT-based smart systems. MDE will accelerate the process of developing smart systems and applications for them, qualitatively complete and further expand at the next evolution stages.

References

[1] L. Atzori, A. Iera, and G. Morabito. The internet of things: A survey. *Computer networks*, 54(15):2787–2805, 2010.

[2] F. Ciccozzi and R. Spalazzese. Mde4iot: supporting the internet of things with model-driven engineering. In *International Symposium on Intelligent and Distributed Computing*, pages 67–76. Springer, 2016.

[3] I. Corredor, A. M. Bernardos, J. Iglesias, and J. R. Casar. Model-driven methodology for rapid deployment of smart spaces based on resource-oriented architectures. *Sensors*, 12(7):9286–9335, 2012.

[4] F. Fleurey, B. Morin, and A. Solberg. A model-driven approach to develop adaptive firmwares. In *Proceedings of the 6th International Symposium on Software Engineering for Adaptive and Self-Managing Systems*, pages 168–177. ACM, 2011.

[5] F. Fleurey, B. Morin, A. Solberg, and O. Barais. Mde to manage communications with and between resource-constrained systems. In *International Conference on Model Driven Engineering Languages and Systems*, pages 349–363. Springer, 2011.

[6] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami. Internet of things (iot): A vision, architectural elements, and future directions. *Future generation computer systems*, 29(7):1645–1660, 2013.

[7] D. Guinard, V. Trifa, and E. Wilde. A resource oriented architecture for the web of things. In *Internet of Things (IOT), 2010*, pages 1–8. IEEE, 2010.

[8] R. Kateule and A. Winter. Architectural design of sensor based environmental information systems for maintainability. In *Nachhaltige Betriebliche Umweltinformationssysteme*, pages 87–96. Springer, 2018.

[9] A. G. Kleppe, J. B. Warmer, and W. Bast. *MDA explained: the model driven architecture: practice and promise*. Addison-Wesley Professional, 2003.

[10] D. Kuryazov, C. Schöneberg, and A. Winter. TOWARDS COLLABORATIVE SMART CITY MODELING, 2018.

[11] P. Patel and D. Cassou. Enabling high-level application development for the internet of things. *Journal of Systems and Software*, 103:62–84, 2015.

[12] J. Rumbaugh, I. Jacobson, and G. Booch. *Unified modeling language reference manual*. Pearson Higher Education, 2004.

[13] E. Serral, P. Valderas, and V. Pelechano. Towards the model driven development of context-aware pervasive systems. *Pervasive and Mobile Computing*, 6(2):254–280, 2010.

[14] D. Steinberg, F. Budinsky, E. Merks, and M. Paternostro. *EMF: eclipse modeling framework*. Pearson Education, 2008.

[15] K. Thramboulidis and F. Christoulakis. Uml4iot—a uml-based approach to exploit iot in cyber-physical manufacturing systems. *Computers in Industry*, 82:259–272, 2016.

[16] R. H. Weber and R. Weber. *Internet of things*, volume 12. Springer, 2010.

[17] C. Yin, Z. Xiong, H. Chen, J. Wang, D. Cooper, and B. David. A literature survey on smart cities. *Science China Information Sciences*, 58(10):1–18, 2015.