Will web technologies impact on building automation systems architecture?

Gérôme Bovet, Jean Hennebert

Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Computer Science 00 (2014) 000–000

Procedia Computer Science

www.elsevier.com/locate/procedia

International Workshop on Enabling ICT for Smart Buildings (ICT-SB 2014)

Will web technologies impact on building automation systems architecture?

Gérôme Bovet[a,b], Jean Hennebert[b,c]

[a] *LTCI, Telecom ParisTech, 46 Rue Barrault, 75013 Paris, France*
[b] *iCoSys, University of Applied Sciences of Western Switzerland, Boulevard de Pérolles 80, 1700 Fribourg, Switzerland*
[c] *DIUF, University of Fribourg, Boulevard de Pérolles 90, 1700 Fribourg, Switzerland*

Abstract

Offices, factories and even private housings are more and more endowed with building management systems (BMS) targeting an increase of comfort as well as lowering energy costs. This expansion is made possible by the progress realized in pervasive computing, providing small sized and affordable sensing devices. However, current BMS are often based on proprietary technologies, making their interoperability and evolution more difficult. For example, we observe the emergence of new applications based on intelligent data analysis able to compute more complex models about the use of the building. Such applications rely on heterogeneous sets of sensors, web data, user feedback and self-learning algorithms. In this position paper, we discuss the role of Web technologies for standardizing the application layer, and thus providing a framework for developing advanced building applications. We present our vision of TASSo, a layered Web model facing actual and future challenges for building management systems.

Selection and peer-review under responsibility of Elhadi M. Shakshuki.

Keywords: Building Management System; Internet-of-Things; Web-of-Things; Architecture

1. Introduction

People are becoming more and more sensitive about their impact in terms of energy consumption. Apart the ecological trend, the interests are also economics-centred due to the inflation of energy costs. As main source of energy consumption, buildings contribute to 20% to 40% of the global energy bill in Europe and the USA[1]. In this context, better control of energy consumption in buildings is becoming necessary. The elaboration of fully automated systems is not an easy task considering the effect of the external environment (weather), the physics of the building and the actual usage patterns inside the building. There is a need for advanced information systems that would ultimately leverage on :

[a] Corresponding author. Tel.: +41-26-4296961 ; fax: +41-26-4296600.
E-mail address: gerome.bovet@hefr.ch

Selection and peer-review under responsibility of Elhadi M. Shakshuki.

- various inter-connected sensors and actuators
- a unified management of sub-systems as HVAC and lighting
- reliable short to medium term prediction of environmental factors such as external temperature, wind, etc
- precise modelling of the physical properties of the building
- a prediction of daily user activities inside the building

On the other side, we can observe that classical building management systems (BMS) are relatively simpler, often compatible with a unique technology (for example KNX or Enocean) and implementing straightforward rules (for example based on temperature thresholds). Recently, the Internet-of-Things (IoT) has come with solutions allowing to interconnect pervasive devices working on different mediums to form a global IP network. The Web-of-Things (WoT)[2], an extension of the IoT, proposes to push Web technologies down to field devices, thus promoting an uniform interaction style based on REST architectures. In this position paper, we address the scientific question of the use of IoT and WoT paradigms in the context of smart buildings. Our core idea is following the same direction, capitalizing on the strengths of the current Web for building an overall building management framework decoupled from underlying technological aspects. We then go a step further by extending the notion of thing to non-tangible entities like algorithms.

In this paper, we start by providing an overview of the evolution and trends of building management systems from several perspectives. The discussion leads up to our vision of a Web oriented management system with its related challenges that need to be addressed. We then continue with discussing a possible approach following our vision. Finally, the last section concludes our paper and provide insights on further research.

2. Evolution of Building Management Systems

The evolution of building management systems can be discussed from several angles. In this section, we focus on two dimensions characterizing infrastructural properties, *Heterogeneity* and *Pervasiveness*. We then examine a third dimension, *Pro-activeness*, that is opening new perspectives.

2.1. Heterogeneity

During the last decade, building management systems have evolved from isolated networks to complex architectures composed of several sub-systems. Modern BMS improve their capabilities by interfacing with information systems storing access rights, weather forecasts or room occupancy schedules. Moreover, it is not uncommon to find various building networks inside one building. Because of physical building constraints or obsolescence, existing installations will eventually consider to evolve by interconnecting with other networks. We can here cite as example the LESO building on the EPFL Campus in Lausanne where a wireless and self-powered network of EnOcean[3] sensors had to be installed because it was not feasible to extend the existing KNX[4] wiring. Another phenomenon contributing to heterogeneity is the trend of supplementing buildings with local energy production systems like solar panels, or car charging stations having each their own management system.

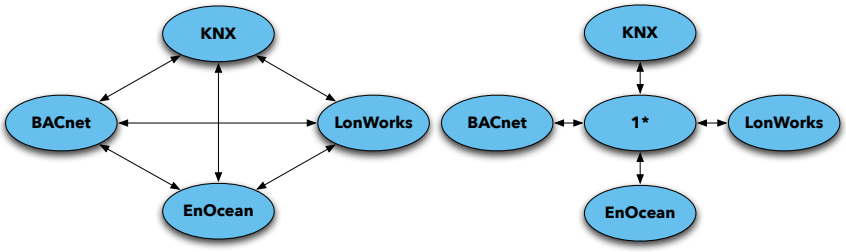

Fig. 1. N-to-N (left) and N-to-1* (right) approaches for protocol mapping in buildings.

Two different visions relying on the use of gateways are striving to interconnect together a priori incompatible networks[5], both illustrated in Figure 1. In the first approach, multi-protocol gateways ensure a *N-to-N* protocol mapping.

Besides the fact that building networks work with fully different protocol stacks that strongly limit inter-compatibility, $\frac{n*(n-1)}{2}$ mappings between BMS would be necessary. In order to cope with the aforementioned limitations, the *N-to-1** approach envisions a new application protocol common to all building networks. The number of mappings is reduced to n, which considerably simplifies the integration of new technologies. The key components of this novel *1** application must still be identified as no standard is currently defined.

2.2. Pervasiveness

Initially, building automation systems where only present in office and factory buildings, targeting an increase of comfort and energy savings. In recent years, the *Internet-of-Things* (IoT) largely contributed to a democratization of sensing devices into houses and apartments. The IoT tends to form a world-wide network interconnecting daily-life objects considered as things through the use of standard communication protocols as IPv6 and RPL among others[6]. Small sized devices are able to perceive the surrounding environment and to communicate over various physical mediums like Wi-Fi, ZigBee, Bluetooth, IEEE 802.15.4. More recently, ubiquitous computing and the Web are merging to build the *Web-of-Things*. We can nowadays find RESTful sensing devices embedding Web servers. We can give as example the Libelium[7] hardware sensing platform and its developer's kit for realizing specific applications. More recently, the Sen.Se Mother and Cookies[8] provide an end-user sensing platform including hardware and a free application for managing devices and analysing data.

2.3. Pro-activeness

The ways in which buildings are managed by the control system have also evolved since its beginnings. They shifted from static reactive algorithms to pro-active systems based on intelligent data analysis such as machine learning[9]. Those new algorithms need historical data from sensors to generate dynamic rules according to user activities. This transition has also some impact on the architecture of control systems. While traditional systems were composed of a single computation node running algorithms for the whole building and thus highly fault tolerant, machine-learning implementations tend to distribute data and algorithms over multiple dedicated nodes. The latter approach, although being more scalable for growing systems as data and algorithms can be spread over the information systems is resulting in greater complexity because of synchronization needs. In a more autonomous approach, independent and self-organising agents offering computational capabilities are distributed on gateways or even sensing devices. This vision pushes scalability at its highest level while ensuring fault tolerance. The drawback of this concept relies in the high complexity necessary for self-organization. Moreover, there is no standardized technology for distributing the machine-learning algorithms as well as for executing the resulting rules.

3. Towards a Web-of-Things Architecture

Following our discussion on *heterogeneity*, *pervasiveness* and *pro-activeness*, we believe that Web technologies can largely contribute to build a Web-oriented BMS architecture. In this section we define and motivate such a vision at the convergence of these three dimensions.

We envision a Web architecture for building automation systems that is:

- **Homogeneous:** by providing a common application layer responsible for ensuring compatibility between various building networks and IoT devices;
- **Pervasive:** by being lightweight enough to run on tiny and constrained daily-life objects and things in general;
- **Auto-configurable:** by minimizing the installation effort, following the plug-and-play concept augmenting the user experience and thus making it accessible for a large variety of people;
- **Autonomous:** by distributing data and algorithms in a self-organised manner avoiding single points of failure and optimizing resource allocation.

We call this vision the *Trendy Automation System Software (TASSo)*, complementing and building upon the Web-of-Things paradigm. In our point of view, a *thing* can be of any kind, whether physical like a tangible sensor or totally virtual such as an automation algorithm.

3.1. Objective and Motivation

Our objective is to develop a framework that enables to build overall automation systems combining various networks using different technologies.

We thus propose to extend and transform BMS into *things-oriented building networks* (TOBN) by introducing autonomous and inter-compatible things. The TOBN will therefore inherit and extend key properties of the Web, such as a strong interaction style allowing communications between various hard-/software platforms, a scalable architecture supporting billions of resources, as well as a total independence to network topology and medium. Such a TOBN is suitable for (i) providing a high level automation system, (ii) extending the perception of things to virtual system components such as algorithms and rules and (iii) providing an easy to use framework for the development of complex WoT applications.

3.2. Challenges

Many challenges need to be addressed before our approach can be transposed to building automation systems. First, the basic concepts of the Web need to be pushed to devices in order to make them TOBN-compatible. This is reflected in making native automation devices like sensors and actuators accessible through RESTful APIs following the WoT paradigm. Here, smart gateways are a promising answer since they allow a transparent access to device properties and states. First attempts have showed the feasibility of exposing uniform REST services allowing access to building networks such as KNX and EnOcean[10][11]. Since IoT devices offer IP connectivity as key foundation, they can be endowed with a built-in Web server exposing a REST API. Besides standardizing the interaction style, a daunting task remains at the interoperability of machine-to-machine communications, especially regarding the semantics of data (e.g. format, encoding, etc.). Currently, communicating with devices requires a strong knowledge on how to provide or read data. This is a serious obstacle towards an homogenisation between different BAS. Techniques for discovering available services and automating their consumption with machine-readable languages are essential conditions for composing mashups.

The democratization of affordable building automation equipment induces new issues. Basic users having limited technological competences, the installation process should be as easy as switching devices and accessing a GUI for configuring their basic properties (name and location). An auto-configuration system has to handle the rest of the work (IP, naming system, functionality discovery, data storage and load balancing) without any human in the loop. While several technologies offering auto-configuration exist, none of them is tailored for the context of building automation systems, being either to heavy or flat structured like ZeroConf[12]. Because of intrinsic properties of BMS like the dynamics of certain devices that can appear, disappear and even move inside a building, some Web concepts that rely on static resources have to be adapted to such scenarios. Currently, the Web relies on the underlying existing DNS architecture for name to IP address translation. Although this architecture fulfils some needs, it is not suited for auto-configuring distributed sensor networks. Globally and from an external point of view, the Web gives the impression of being distributed, but in reality it is based on a totally pre-configured architecture that allows only little dynamism. This behaviour is actually incompatible with the vision of distributed sensing and knowledge.

4. TASSo - A Layered Model

In order to better understand the TASSo framework, we make use of the layered model shown in Figure 2. The framework is divided such that each layer can run isolated from each other. This model, originating from the classical decomposition[13] is extended and reorganised according to the aforementioned challenges. As previously explained, the TASSo platform builds on top of the WoT, meaning that each module is considered as a thing. Following this assumption, every functionality is accessible through a REST API.

4.1. Adaptation Level

As data analysis technologies such as machine learning are gaining momentum in building optimization, we introduce a new layer dedicated to this task. As its name says, the Adaptation Level will adapt the management of the

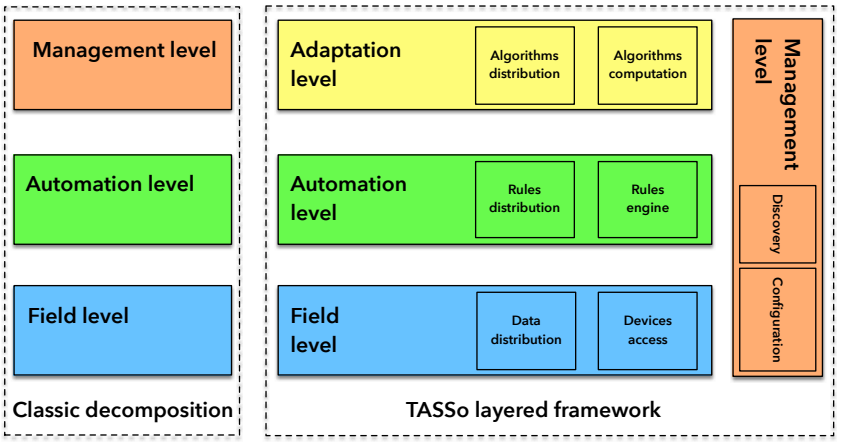

Fig. 2. Level based architecture of building automation systems. The left part shows the classic state-of-the-art decomposition. The right part shows our proposition, the TASSo layered framework.

building according to stochastic events such as user activities or weather. To achieve this goal, those algorithms require historical data generated by field devices. The outcoming rules will be distributed and executed by the Automation Level.

Many algorithms currently developed are rather CPU and memory intensive, browsing large quantity of historical data to build parametric models. For this reason we aim at extending the WoT paradigm to the notion of distributed virtual things accessible over a RESTful API. We envision to distribute algorithms among the network on field devices or gateways having some processing time to offer. Reducing the communication costs is a key constraint in this context. Algorithms should be distributed as close as possible to the location of historical data, thus avoiding network overload and traffic congestion.

4.2. Automation Level

The Automation Level will execute automation rules either generated by the Adaptation Level or statically defined by users. We can distinguish two different types of rules that will probably be handled differently: event based and data oriented. In the first case, they will be executed in response to events generated by field devices. In the latter, they will be continuously executed and could take benefit from historical data, as for example control loops.

In the same way as for the Adaptation Level, the execution of those rules can be distributed on field devices or gateways having computational capacities. The rule distribution module will try to find the most appropriate location for the rules according to the relative devices and data placement in order to minimize communication costs.

4.3. Field Level

This level offers access to the sensing and actuation capabilities of devices. Each capability is matched to a REST service that is directly attached to a physical action. URIs pointing to the resource are built according to the location of the device. We here propose to use the name of the device and its location as DNS name of the resource instead to put it as location-path like it is the case in the traditional WoT approach. For example, a temperature sensor located in the bathroom on the first floor of a house would result in the following DNS name: temperature.bathroom.1stfloor.home. This approach, although requiring a naming system, allows to uniquely identify non-IP devices only accessible through a smart gateway without knowing to which it is attached, and thus enables roaming of devices.

This layer also handles the distribution of historical data over multiple databases. We here advocate, when possible, to store data directly on the device producing it. As this is not possible on very constrained hardware, the producing devices will push it on other nodes offering storage services, selected based on their proximity. In order to limit the

amount of stored data, we introduce a registration mechanism, where clients such as the Adaptation Level requiring historical data will announce their needs by specifying which resource has to be stored for how long.

4.4. Management Level

Finally, the Management Level offers services for discovering and configuring the other aforementioned layers. In comparison to the state-of-the-art model, we decided to place this layer vertically. We argue this choice by the fact that each layer needs discovery and configuration capabilities. As previously explained, being able to compose mashups between things is mandatory, and thus requires a system that allows things to automatically discover each other along with the services they expose. The Semantic Web is a promising field where technologies allowing machines to share and reuse data with no human in the loop are proposed. In this context, ontologies are used for sharing vocabularies and associations of definitions, while RDF[14] is the common language to express the meta-data. Devices will register their resources among with their RDF description within the Management level, and thus make them discoverable for the rest of the network.

5. Conclusions

In this paper we have provided an overview on the evolution of building management systems from several perspectives: *heterogeneity*, *pervasiveness* and *pro-activeness*. Then, we described our vision of a Web-oriented building management architecture, at the convergence of those three dimensions. In this vision, the management system is built upon the paradigm of the *Web-of-Things* in which everyday objects become Web-enabled, offering a uniform interface to communicate with things. We make a step further by extending this concept to algorithms and rules considered as virtual things providing the same uniform interaction style. Following this vision allows to build a unified management relying on smart gateways and native Web-enabled devices, thus providing a solution to heterogeneity. Moreover, we have identified the challenges that remain to be addressed, and have presented a possible layered model. In our approach, the building management system is decomposed in independent functional layers that are self-organizing in order to ease installation and configuration.

Moreover, some open points need to be further investigated. First, standard formats for exchanging and executing data-driven algorithms and rules have to be defined. Then, a way for managing virtual things (algorithms and rules) in a stateful way is necessary, while current REST architectures rely on a stateless approach. Once those shortcomings resolved, the way to fully Web-enhanced building management systems will be paved and will allow new promising applications.

References

1. Perez-Lombard, L., Ortiz, J., Pout, C.. A review on buildings energy consumption information. *Energy and Buildings* 2008;**40**:394–398.
2. Guinard, D.. *A Web of Things Application Architecture - Integrating the Real World into the Web*. Ph.D. thesis; ETHZ; 2011.
3. Enocean. `http://www.enocean.com/en/home/`; 2014.
4. Knx association. `http://www.knx.org/`; 2014.
5. Jung, M., Reinisch, C., Kastner, W.. Integrating building automation systems and ipv6 in the internet of things. In: *Proc. of the 6th International Conference on Innovative Mobile and Internet Services in Ubiquitous Computing*. 2012, .
6. Guo, B., Zhang, D., Wang, Z.. Living with internet of things: The emergence of embedded intelligence. In: *Proc. of the 4th IEEE International Conference on Cyber, Physical and Social Computing*. 2011, p. 297 – 304.
7. Libelium development platform. `http://www.libelium.com/`; 2014.
8. Sen.se mother and cookies. `https://sen.se/`; 2014.
9. Ridi, A., Zakaridis, N., Bovet, G., Morel, N., Hennebert, J.. Towards reliable stochastic data-driven models applied to the energy saving in buildings. In: *Proceedings of the International Conference on Cleantech for Smart Cities and Buildings (Cisbat '13)*. 2013, .
10. Bovet, G., Hennebert, J.. Offering web-of-things connectivity to building networks. In: *Proc. of the 2013 ACM international joint conference on Pervasive and ubiquitous computing*. 2013, .
11. Bovet, G., Hennebert, J.. Web-of-things gateway for knx networks. In: *Proc. of the IEEE European Conference on Smart Objects, Systems and Technologies (Smart SysTech 2013)*. 2013, .
12. Guttman, E.. Zeroconf host profile applicability statement. `http://files.zeroconf.org/draft-ietf-zeroconf-host-prof-01.txt`; 2001.
13. Iso 16484-2:2004 building automation and control systems (bacs). Tech. Rep.; International Organization for Standardization; 2004.
14. W3C, . Resource description framework. `http://www.w3.org/RDF/`; 2014.