Architecture for Self-organizing, Co-operative and Robust Building Automation Systems

Franck Bernier(1), Joern Ploennigs(2), Dirk Pesch(3), Susanne Lesecq(4), Twan Basten(5), Menouer Boubekeur(6), Dee Denteneer(7), Fred Oltmanns(8), François Bonnard(1), Matthias Lehmann(2), Mai Tuan Linh(2),Alan Mc Gibney(3), Susan Rea(3), François Pacull(4), Claire Guyon-Gardeux(4), Laurent-Frédéric Ducreux(4), Safietou Thior(4), Martijn Hendriks(5), Jacques Verriet(5), Szymon Fedor(6)

(1) Schneider Electric Industries, (2) Technische Universität Dresden, (3) Cork Institute of Technology, (4) Commissariat à l'Energie Atomique et aux Energies Alternatives, (5) TNO-ESI, (6) United Technology Research Center Ireland, (7) Philips Research, (8) Ennovatis.

***Abstract*— This paper provides an overview of the architecture for self-organizing, co-operative and robust Building Automation Systems (BAS) proposed by the EC funded FP7 SCUBA[1] project. We describe the current situation in monitoring and control systems and outline the typical stakeholders involved in the case of building automation systems. We derive seven typical use cases which will be demonstrated and evaluated on pilot sites. From these use cases the project designed an architecture relying on six main modules that realize the design, commissioning and operation of self-organizing, co-operative, robust BAS.**

Keywords*— monitoring and control, self-organisation, robustness, adaptive, service oriented architecture, co-operative services, interoperability, semantics, building management systems.

I. Introduction

Large scale embedded Monitoring and Control (M&C) systems in energy management, transportation, security and safety often co-exist alongside each other with little co-operation within and among heterogeneous systems which hampers the increasing demand to operate the whole system optimally. A good example is building management where a wide range of vendor specific, heterogeneous M&C systems are in use for heating, ventilation, air-conditioning (HVAC), lighting, access control, fire, safety, and security systems. The devices of these individual systems can often cooperate in pre-planned ways such as to support a decentralized control application. However, at the system layer these systems usually lack co-operation as interfaces and semantics for interoperability are missing. The market is quite fragmented with many vendors providing monitoring and control systems for buildings using a wide range of wired and increasingly wireless communication and networking technologies [1]. While a range of standards exist for communication technologies [2], the systems are split into individual domains for HVAC control, lighting control, fire and security applications, etc. with little to no standardization. These systems are usually installed by different system integrators and most of the applications at the device and system layers are vendor proprietary. This heterogeneity leaves little capacity for interoperability and co-operation among the different devices and systems [3]. This creates many problems along the life cycle in particular the engineering, commissioning and operation of these systems.

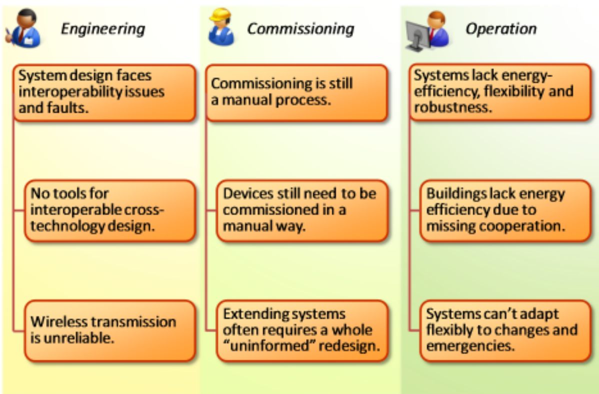

Fig. 1. Current Situation.

Fig. 1 summaries the key issues that have been identified with existing M&C systems that result from these heterogeneous architectures. Systems engineering is problematic as special care needs to be taken with regard to the interoperability problem that is prevalent at device and system levels. Most of the interoperability problems appear during the system installation and commissioning phase and typically require a duplication of effort to work around these problems. The maintenance of the system is cumbersome as replacements and flexible extensions such as energy monitoring systems are complex to include into an existing system. The operation of the building suffers in the end due to regularly failing components, lack of co-operation, and the need to individually control each system. Therefore, the systems barely adapt to the changing goals of its stakeholders (facilities managers, building owners, or tenants); and cannot evolve with time-variant dynamics or are scarcely able to deal with unexpected or unpredictable behavior. Non-co-operation among individual M&C systems not only hampers evolution but more significantly the effective control during a rapidly changing environment.

It is anticipated that if the sensor data that is gathered by one system is made available to others in the building, this can lead to improved overall system operation and robustness. In

[1] SCUBA – Self-organizing, Cooperative robUst Building Automation

addition to this, we find that at a system level often contradictory goals such as operating energy-efficiently, with high service, and safety quality at a low price, lead to inefficient operation and occupant discomfort. Addressing such complex system goals in a robust manner requires moving from individual, centralized systems with single points of failures towards collaborative distributed M&C systems that cooperate both at device and system level to serve the overall system wide operational strategies in an efficient and robust manner. The SCUBA project aims to address the challenges outlined above by proposing a self-organizing and self-adaptive architecture to form the basis for co-operative and robust M&C systems at device, system and building operation levels.

II. System Scope and Use Cases

SCUBA starting point has been to take a full building life cycle perspective leading to a target workflow as illustrated in Fig. 2. The flow starts at the engineering stage when it is required to have the proper tool chain to define the systems and devices to be implemented in the real building. Then a key hurdle is the commissioning phase when we install and configure the equipment and devices on site. Once installed and configured the SCUBA architecture demonstrates its capability to be versatile and adjust to context changes within the environment for example in emergency situations.

Along this workflow several stakeholders benefit from the SCUBA architecture. Fig. 3 summarizes the targeted users and their interaction with the SCUBA system. These users not only encapsulate human users but also other "client" systems. The aim of the SCUBA system is to provide an added value intermediate between traditional building systems/devices and the defined SCUBA users. The following describes the users and associated use cases that have been identified and are used to drive the requirements of the SCUBA architecture:

System integrators require support to design and extend existing systems by selecting interoperable devices and creating reliable wireless system designs. Commissioning is still a very manual process that is not well integrated with the design. A technology independent self-organization approach that automates commissioning and provides easy means to extend existing system is required.

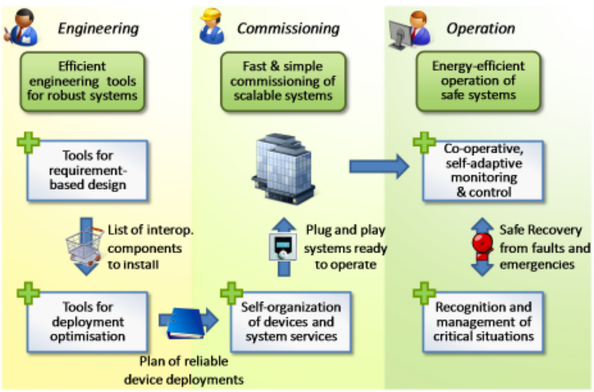

Fig. 2. SCUBA Workflow.

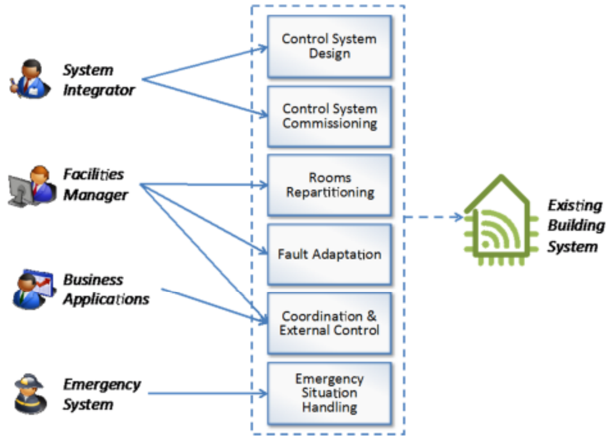

Fig. 3. SCUBA Stakeholders and Use Cases.

Facility Managers are looking for a system that easily integrates different technologies, is robust to failures and adaptive to flexible usage. Nowadays, failures usually require manual and timely intervention of staff. The new system should be able to minimize the impact of device failures on operational objectives such as comfort or energy efficiency. In a similar way, the system should efficiently adapt to a change in building requirements for example an open space being repartitioned or changes in some room usage.

Emergency Systems are representative of systems that need to override the normal building control operation such as a typical fire security system. Emergency systems require secure interfaces to utilize the SCUBA system to "force" a state change in some building entity as dictated by the emergency situation that is occurring.

Business Applications represents any application that extends the SCUBA system capabilities. The number of applications interacting with buildings is constantly growing from energy management, demand side control, facility management, etc. Such applications mainly need to have access to existing control system data and potentially apply some changes to the control system settings and operating modes.

Building systems and/or devices represent existing standard or proprietary devices or systems that have the capability to communicate through wired or wireless communication channels. Examples of such systems/devices can be lighting control devices, occupancy sensors, HVAC systems, or a room reservation system. These devices/systems are not part of the SCUBA system but rather are used, controlled and coordinated by it.

III. Requirements

Different requirements on the solution were derived from the use cases sketched in the last section. The most significant requirements that have been identified are described below.

Distribution and redundancy of functionality: Distribution of functionality and segmentation of building systems is important to make the approach scalable from small to large building scenarios. Redundancy on the other hand improves reliability of the building control system. Both aspects should be totally transparent to other components of the SCUBA system in order to maximize the potential of co-operation between system components so that communication is not artificially limited.

Multi-protocol communication: Several protocols have already been established in building automation. A new protocol or even simple modification of existing ones would render most existing installations and available devices incompatible to the SCUBA approach. For the highest compatibility and applicability, the architecture should run on top of existing communication protocols without modification of the devices software stack.

Cross-technology communication: Devices of different communication protocols (i.e. ZigBee sensor and a LON actuator) that are located in different physical network segments should be bound together as if they were located on the same physical segment and technology. This would simplify many integration and communication issues in large systems.

Flexible configuration management: Building usage and operation becomes more flexible. BAS should be easily reconfigurable and support various configurations at the same time and be able to switch between configurations seamlessly and autonomously.

Extended coordination and business functionality: It should be possible to add some logic to coordinate operations of different devices. For example force shutters to operate in a different mode depending on indoor presence. The SCUBA capabilities should be also easily extensible with components that implement business functionality.

Detection and commissioning of new devices: The system should be able to detect new devices, their types and manufacturers using discovery mechanisms or via the reporting functionalities available in the communications protocol used by the devices. The system should further support the commissioning of devices via common protocol functionality as far as this is supported.

IV. Related Work

Middleware approaches follow a similar goal of integrating heterogeneous technologies as SCUBA does. One example is the HYDRA middleware which allows developers to incorporate heterogeneous physical devices into their applications by offering easy-to-use web service interfaces [4]. Devices that can be integrated are described via an ontology called *LinkSmart*. An OPC-UA based middleware approach for BAS was demonstrated in [5] for BACnet, KNX, LON and ZigBee. The OPC UA information model allows transparent object-oriented description of device data [6].

eHome is another service oriented middleware for home automation which is based on an OSGi gateway [7]. It operates as a central middleware that processes all communications. This removes decentralized communication abilities from the devices and creates a single point of failure. On the other hand, eHome also approaches the question how such systems can be designed and therefore has a more holistic approach [8].

Multi-agent systems provide more flexibility in this aspect and making it easier to support the addition of supplementary control logic. *ThinkHome* is a combination of a knowledge base and a multi-agent system that supports co-operation of heterogeneous building automation systems to achieve a good balance between user comfort and energy efficiency [9]. The control agents generate usage profiles and find the right trade-off between energy efficiency and comfort.

These platforms provide device and service discovery functionality, but, do not support the self-organization of the device layer itself as it requires more context knowledge and intelligence such as the location of devices [10]. This knowledge can be retrieved from the design [8]. Therefore, the design of systems is an important aspect to realize a holistic self-organizing, co-operative system.

V. General Architecture

Based on the identified use cases and requirements, a general high level end to end architecture is proposed which provides the self-organization and co-operative capabilities between the systems already deployed within a building. In order to provide the capabilities various software/modules need to be developed and can be organized in a tool and a middleware layer. The modules in these layers are implemented based on a service-oriented architecture using SOAP web services, to provide easy accessibility and extensibility to other services as required. The tool and middleware layers are added on top of the device layer of existing or new systems. The device layer can consist of heterogeneous technologies integrated by the middleware layer.

A. Tool layer

The *tool layer* provides different engineering tools for users to support the design, operation and maintenance of the SCUBA system. The tools address the issues of system integrators and facility managers to design and maintain a system efficiently [11]. The architecture integrates three key tools to support the design and redesign of building automation systems:

- An *automated design tool* [12] that supports the planner to create a system specification and a system design from requirements. The tool allows users to derive a design from a technology independent requirement specification. It automatically creates various designs for these templates including selecting and binding the required components. These detailed designs specify which devices shall be installed in each room or building segment.
- A *device deployment optimization tool for wireless sensor/actuator network* installations that optimizes the device's positions according to signal propagation and functional requirements [13]. This avoids common problems with indoor signal propagation and reduces the costs of over dimensioned network topologies. The resulting system specification contains all relevant information for deploying the automation system including the requirements, the devices to install and their installation positions in each building area.

- A *coordination scheme editor* that supports the user in developing, generating, and validating co-operation rules for the self-x middleware. Such rules are also used to realize the self-organization. The tool creates these rules out of the system design. The coordination scheme editor provides a simple user friendly way to edit these rules and implement supplementary rules that provide additional coordination and co-operation functionality on the self-x middleware.

B. *Middleware layer*

The middleware layer provides three modules to enable self-organizing and co-operation among building systems. It was a central requirement of SCUBA to simply integrate existing device technologies without any modifications to their software. This way SCUBA ensures the applicability to new and old building automation systems. The middleware layer groups three key modules:

- A *system model service* provides central, extensible model of the installed system to different services. The model represents the system on the one hand as designed and on the other hand as installed and supports different technologies. The model is ontology-based, modular and easily extensible [14]. The design tools store and exchange design information, the self-x middleware extracts and updates commissioning information, and the strategy manager uses it to retrieve information about the system control design.
- A *self-x middleware* manages the access to the device layer for the SCUBA services. The middleware provides an abstraction layer on top of the different technologies [15]. It retrieves and offers data from the devices in the system and act as a gateway between technologies to minimize technology-specific constraints and integrating fault-tolerance mechanisms. It also executes the binding of devices and the coordination rules to realize the self-organization [16]. Thus, self-organization and self-adaptation are included within the global system in a natural and unified comprehensive way.
- A *strategy and co-operation manager* provides added value to BAS that is enabled by the SCUBA architecture. Services such as scenario management, fault adaptation, and co-operation for energy-efficient buildings can be deployed by this module. Therefore, the manager monitors the system status, detects events and scenarios and adapts the system operation strategy and configuration accordingly.

Fig. 4 shows the different modules of the SCUBA system and their dependencies. Each module is independent from any technology or protocol. The interfaces between modules are implemented using common Web Service technologies. The interactions between the self-X middleware and the modules of the device layer rely on a high level coordination protocol.

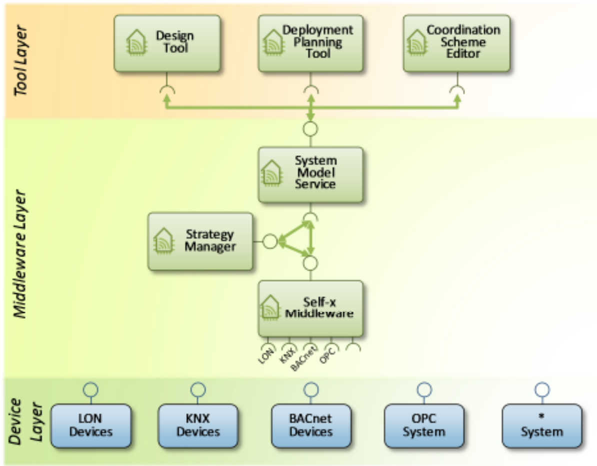

Fig. 4. SCUBA General Architecture.

VI. EXPERIMENTS

To validate that SCUBA delivered components fully address projects requirements and in order to demonstrate how valuable these features are from a system integrator or facility manager perspective, two experimental sites will be equipped with a SCUBA based prototype of a building automation system. This section will describe both experimental sites, the prototype building management system that will be implemented along with the associated validation and demonstration objectives of these experiments.

A. *Integration Platform*

The platform consists of two office rooms dedicated to room control experimentations. The two rooms are equipped with several devices to control user comfort as shown in Fig. 5. This platform is intended for testing room control solutions only from a system integration and user acceptance point of view. All comfort equipment is fully controllable via commercially available controllers and actuators. In addition, each room is fully instrumented with a set of traditional comfort sensors. Sensors, controllers and actuators are all interconnected and communicating over a LON fieldbus.

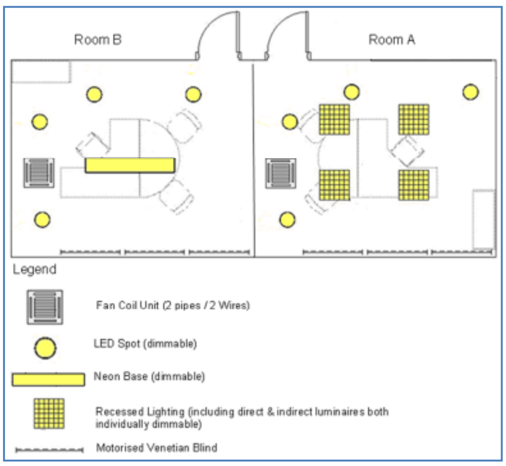

Fig. 5. Integration Platform Floor plan.

We will execute the following tests on this platform:

Design and commissioning: Establishing the detailed system design and bill of components of a building control system usually represents a significant workload especially for large buildings. In addition, requirements are not always clear and well defined during the early phase of a project such as the tendering phase. The design should address these topics and provide a design as close to requirements as possible for an optimized quotation while minimizing the man-hour investment. Within this context, the system integrators need productivity tools to support the definition of the most appropriate and complete technical solution to meet site specific requirements. The test evaluates the productivity increase in using the systematic design approach to design and commission the platform with the self-x middleware.

Co-operative inter-system control: Building systems, such as lighting, heating, ventilation, cooling, blinds, etc., work often quite isolated. They are designed separately, installed with various technologies, and integrated only vertically in backend-systems. A horizontal co-operation between systems is often not realized. The test will show how a collaborative system can save energy, by enabling new ways of co-operation among existing devices in the platform.

Flexible usage: One growing need in building management is the so called "space management" that leads to frequent repartitioning of the spaces into rooms (i.e. addition/removal of separation walls) or reallocation of existing rooms to a different usage (i.e. change usage of a room from cubicle to individual, office to coffee corner ...). This means that the room control has to be frequently reconfigured to be kept consistent with the resulting partitioning and usage of the room. The test case will show the SCUBA ability to allow fast and easy space reconfiguration. A movable partition wall has been installed in the platform. If it is opened or closed the system configuration should automatically adapt to the new situation by changing device bindings such that the coordinated control controls one combined or two separated rooms, respectively.

Device failure (reconfiguration): Failures of some control components (i.e. loss of a temperature sensor, loss of an actuator...) are not rare especially in large buildings where hundreds of such devices may be installed. Failures may have significant impact on the energy efficiency and can remain undetected for a long period of time. For example, a failure of an occupancy sensor could lead to the heating control always being active, if it is occupancy dependent, the long delay times in the system make it nearly unnoticeable by users if it is un-signaled. It is thus important for a facility manager to have means to detect and possibly mask such failures. The increasing number of devices installed in buildings permits switching to alternative devices that provide similar sensor or actuator capabilities, in case of device failures. The capability of the system to detect such failures and to reconfigure the room control system to mask the missing device and limit the impact of the failure both on comfort and energy efficiency, will be shown in the test with the example of alternative lighting control with different lights and sun blinds.

Device failure (virtual sensors): As the number of devices in buildings increases so too does the amount of available data from devices. This allows new approaches of virtual sensing to be used by replacing failed real sensors by soft sensors that use predictive or correlative models. The test will validate SCUBA enhanced abilities to accommodate a sensor failure by automatically activating a virtual sensor that provides estimated sensor values from alternative sources and manage cascading faults.

Emergency handling: For some emergency situations, the normal operation of a room appliances and state must be overridden. This is typically the circumstance in case of a fire where lights, shutters, openings need to be set to a particular state to maximize visibility and to limit the spread of the fire. These exceptional scenarios are quite easy to specify as they usually consist of a list of states to be applied as is for each appliance. On the other hand, they are quite difficult to implement with traditional building systems where the switching logic needs to be explicitly programmed. The test demonstrates SCUBA ability to (i) define and apply emergency scenarios, (ii) cooperate with the fire emergency system (which typically is not a part of the SCUBA system due to legislation), and (iii) provide access and control to an external application.

B. Pilot Building

The SCUBA achievements will be demonstrated in a large scale scenario at a real building. The pilot building (cf. Fig. 6.) is located at Savoie Technolac, Bourget du Lac, France. It contains of 10 offices, a meeting room, a technical room and two rest rooms. The rooms are equipped with a simple occupancy monitoring system and a HVAC system.

The pilot building is used to execute the following tests:

WSN Deployment Design: Large scale wireless indoor deployments often face problems with unreliable signal propagation. The test will validate the quality improvements achieved by the wireless deployment planning tool that optimizes the number and position of wireless devices to ensure reliable communications between sensors, actuators and backend systems.

Legacy system integration: Many existing buildings contain existing legacy systems that are only accessible as black-box. The test will demonstrate SCUBA ability to cope with such a closed system, by enabling energy-efficient demand-driven on/off control that integrates with the other available systems.

Co-operative multi-technology control: The test will implement a co-operative energy efficient control system across heterogeneous protocols in a multi-vendor, multi-standard scenario. The test case addresses individualized lighting, cooling and heating control that utilizes various off-the-shelf embedded devices such as smart plugs and wired and wireless devices.

Fig. 6. Pilot Building

VII. CONCLUSION

SCUBA aims to add self-organizing capabilities to building automation technologies to improve co-operation, robustness, and flexibility. SCUBA does not develop a new protocol that will be hard to establish in a saturated market. Instead it extends existing technologies with simple modules to improve the design, commissioning and operation of the building automation systems. By this, it delivers the required adaptability to react to changes in building design and operation that become increasingly dynamic. On the other hand, automation systems have to work deterministically to provide reliable comfort to occupants and estimate performance to operators. Therefore, SCUBA follows a systematic approach that covers the whole building system life cycle from design, to commissioning, operation, and re-commissioning. This allows improvements along each phase for different stakeholders. Fig. 7 summarizes the vision of SCUBA to change the current shortcomings as outlined in Fig. 1. It addresses the interoperability problem with a systematic design approach that helps users in selecting interoperable devices. The issue of indoor signal propagation for wireless devices is addressed by a device deployment optimization. System commissioning is supported by a self-organization middleware that utilizes design knowledge from an ontology-based model database. Extensions of existing systems are supported by reengineering approaches that are currently developed to feed system knowledge back into the design. The middleware can be connected to different protocols and functions as an automatic gateway. The self-organization of the device layer improves the reliability of the system as it preserves its decentralized communication. All modules provide self-documenting web services that allow the easy implementation of additional services. One such service is the strategy manager that harmonizes different building operation strategies and handles emergencies in the system.

We expect, based on our preliminary results, that the architecture introduced in this paper is capable of achieving this goal. To ensure this, various test cases were specified that tightly align to the use cases. The partners successfully implemented the architecture and first test covering the design were successfully completed.

ACKNOWLEDGEMENTS

The authors wish to acknowledge the support of the EU Commission under the FP7 SCUBA project (Grant Agreement No 288079) in part funding the work reported in this paper.

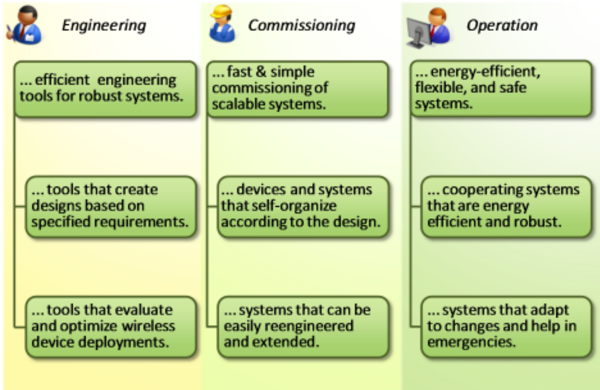

Fig. 7. SCUBA Vision.

REFERENCES

[1] Itard, L.; Meijer, F.; Vrins, E.; Hoiting, H.: Building Renovation and Modernisation in Europe: State of the art review. Jan. 2008.

[2] Kastner, W.; Neugschwandtner, G.; Soucek, S.; Newman, H. M.: Communication Systems for Building Automation and Control, Proc. of the IEEE, vol. 93, no. 6, pp. 1178–1203, 2005.

[3] Dietrich, D.; Bruckner, D.; Zucker, G.; Palensky, P.: Communication and Computation in Buildings: A Short Introduction and Overview, IEEE Trans. Ind. Electron., vol. 57, no. 11, pp. 3577–3584, Nov. 2010.

[4] Eisenhauer, M.; Jahn, M.; Pramudianto, F.; Sabol, T.; Kostelnik, P.; Hreno, J.: Towards a Generic Middleware for Developing Ambient Intelligence Applications. In: 2nd Workshop of the EEB Data Models Community, CIB Conference W078-W12, France, 2011

[5] Granzer, W.; Kastner, W.: Information Modeling in Heterogeneous Building Automation Systems. In: 9th IEEE WFCS, Germany, 2012.

[6] IEC 62541-5: OPC UA - Part 5: Information model, 2011.

[7] Retkowitz, D.; Kulle, K.: Dependency Management in Smart Homes. DAIS - 9th IFIP Int. Conf. on Distributed Applications and Interoperable Systems, vol. 5523, LNCS, pages 143-156. 2009.

[8] Norbisrath, U.; Armac, I.; Retkowitz, D.; Salumaa, P.: Modeling eHome Systems. MPAC - 4th Intl. Workshop on Middleware for Pervasive and Ad-Hoc Computing, page 4, 2006.

[9] Reinisch, C.; Kofler, M.J.; Iglesias Vazquez, F.; Kastner, W.: ThinkHome Energy Efficiency in Future Smart Homes, EURASIP Journal on Embedded Systems, vol. 2011, article ID 104617, 2011.

[10] Kjaer, K.E.: A survey of context-aware middleware, SE - 25th Conf. on IASTED Int. Multi-Conf.: Software Engineering, pp. 148-155, 2007.

[11] Mc Gibney, A.; Rea, S.; Lehmann, S., Thior, S.; Lesecq, S.; Hendriks, M.; Gardeux, C.; Mai, L.T.; Pacull, F.; Ploennigs, J.; Basten, T.; Pesch, D.: A Systematic Engineering Toolchain Approach for self-organising Building Automation Systems, IEEE IECON, 2013.

[12] Dibowski, H.; Ploennigs, J.; Kabitzsch, K.: Automated design of building automation systems. IEEE Trans. Ind. Electron., vol. 57, no. 11, pp. 3606 – 3613, 2010.

[13] Mc Gibney, A.; Klepal, M.; Pesch, D.: Agent-Based Optimization for Large Scale WLAN Design, IEEE Trans. Evol. Comput., vol. 15, no. 4, pp. 470–486, Aug. 2011.

[14] Ploennigs, J.; Hensel, B.; Dibowski, H.; Kabitzsch, K.: BASont – A modular, adaptive Building Automation System Ontology, IEEE IECON, pp. 4827–4833, Oct. 2012.

[15] Ducreux, L.-F.; Guyon-Gardeux, C.; Lesecq, S.; Pacull, F.; Thior, S. R.: Resource-based middleware in the context of heterogeneous building automation systems, IEEE IECON, pp. 4847–4852, Oct. 2012.

[16] Pacull, F.; Ducreux, L.-F.; Thior, S.; Moner, H.; Pusceddu, D.; Yaakoubi, O.; Guyon-Gardeux, C.; Fedor, S.; Lesecq, S.; Boubekeur, M. Pesch, D.: Self-Organisation for Building Automation Systems: Middleware LINC as an Integration Tool. IEEE IECON, 2013.