POLITECNICO DI TORINO
Repository ISTITUZIONALE

Event-Driven User-Centric Middleware for Energy Efficient Buildings and Public Spaces

Original
Event-Driven User-Centric Middleware for Energy Efficient Buildings and Public Spaces / Patti, E., Acquaviva, A., Marco, J., Ferry, P., Tomasi, R., Damien, R., Joseph, V., Macii, E.. - In: IEEE SYSTEMS JOURNAL. - ISSN 1932-8184. - 10:3(2016), pp. 1137-1146. [10.1109/JSYST.2014.2302750]

Availability:
This version is available at: 11583/2526689 since: 2016-11-24T20:38:21Z

Publisher:
IEEE

Published
DOI:10.1109/JSYST.2014.2302750

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

10 October 2026

Event-Driven User-Centric Middleware for Energy Efficient Buildings and Public Spaces

Edoardo Patti, Andrea Acquaviva, Marco Jahn, Ferry Pramudianto, Riccardo Tomasi, Damien Rabourdin, Joseph Virgone and Enrico Macii

***Abstract*—In this work, the design of an event-driven user-centric middleware for monitoring and managing energy consumption in public buildings and spaces is presented. The main purpose is to increase the energy efficiency, reducing consumption, in buildings and public spaces. To achieve this, the proposed service-oriented middleware has been designed to be event based, also exploiting the user behaviours patterns of the people who live and work into the building. Furthermore, it allows an easy integration of heterogeneous technologies in order to enable a hardware independent interoperability between them. Moreover, a Heating Ventilation and Air Conditioning (HVAC) control strategy has been developed and the whole infrastructure has been deployed in a real-world case study consisting of a historical building. Finally the results will be presented and discussed.**

***Index Terms*—Internet of things, Ubiquitous Computing, Ambient Intelligent, Pervasive Computing, event-driven middleware, web services, smart buildings, user-centred development, energy efficiency**

I. Introduction

NOWADAYS, one of the major challenges concerns the reduction of the CO_2 footprint in our cities. The European Union Directive on the *Energy Performance of Buildings* [1] reports that about 40% of energy consumptions in Europe is due to existing buildings. A big share of this is consumed by heating, cooling and lighting of public and commercial buildings. Moreover the European Union declared: *"Information and Communication Technologies have an important role to play in reducing the energy intensity and increasing the energy efficiency of the economy, in other words, in reducing emissions and contributing to sustainable growth"* [2]. Therefore, one of the main aims in today's economy and research is to increase the energy efficiency in existing Public buildings and Spaces without significant construction works. Particular emphasis is given to historical buildings, which are typically less energy efficient and impose tight deployment constraints to avoid damage by extensive retrofitting. Hence, the existing buildings should be converted as much as possible into *Smart Buildings*, exploiting also a LivingLab approach [3], in order to move forward towards the vision of the future *Smart Cities* [4].

Existing buildings are often equipped with a Building Management System (BMS) to enable a coarse grain control of Heating Ventilation and Air Conditioning (HVAC) and lighting. However, a Smart Building has to react considering the real-time user behaviour in order to provide comfort and save energy as well. To achieve this objective, from one side a finer grain monitoring and control is needed. On the other side, to properly and effectively take user behaviour into account, a user-centric approach would be required.

Pervasive technologies such as Wireless Sensor and Actuator Networks (WSAN) are nowadays available to extend existing BMS to improve monitoring and control granularity. However, to make these technologies actually widespread in public buildings, major challenges remain concerning the development of a suitable middleware complex enough to allow the interoperability of heterogeneous technologies and providing at the same time an effective reaction to environmental events and user behaviour.

In this work we present the design and implementation of an event-driven service-oriented middleware for energy efficiency in buildings and public spaces. The design has its roots in the LinkSmart Middleware, whose basics are presented in [5]. LinkSmart design allows an easy integration of heterogeneous technologies and enables a hardware independent interoperability between them. Moreover, it provides software components for implementing building energy management systems on top of these technologies, also exploiting an event-based approach.

The solution proposed in this paper aims to provide a tool for developing user-centric applications. We present a complete and realistic application of the middleware in historical buildings. In the presented case study, particular emphasis has been given to the aggregation and exploitation of occupancy information coming from heterogeneous sources, both hardware and software. Finally, a HVAC control strategy has been developed and deployed in a real-world case study, which consists of a historical building.

The rest of the paper is organized as follows. Section II reviews some background literature. Section III presents the LinkSmart middleware, which is the basis of the proposed Energy Efficient Middleware described in Section IV. Then Section V exposes the HVAC control strategy deployed in the real-world case study described in Section VI. Finally,

E. Patti, A. Acquaviva and E. Macii are with the Department of Control and Computer Engineering, Politecnico di Torino, Italy. Emails: {edoardo.patti, andrea.acquaviva, enrico.macii}@polito.it

M. Jahn and F. Pramudianto are with Department of User Centered Computing, Fraunhofer Institute for Applied Information Technology FIT, Germany. Emails: {marco.jahn, ferry.pramudianto}@fit.fraunhofer.de

R. Tomasi is with Istituto Superiore Mario Boella ISMB, Italy. Email: tomasi@ismb.it

D. Rabourdin and J. Virgone are with Centre Thermique de Lyon, INSA-Lyon, Université Claude Bernard Lyon 1, France. Emails: {damien.rabourdin, joseph.virgone}@insa-lyon.fr

the obtained results are reported in Section VII.

II. Related work

Nowadays we are trying to promote a change, which aims to convert our cities into *Smart Cities*. In Smart Cities reduction of energy consumption and CO_2 footprint is achieved by optimizing both the energy and electricity distribution network (*Smart Grid*) as well as single (smart) buildings and houses [6].

Recent development of UbiComp, AmI and IoT technologies can help to address this challenge by providing means to seamlessly interact with distributed sensors and actuators. In this context, a key challenge remains to achieve true interoperability between heterogeneous devices. Service Oriented Architectures seems to be promising along this direction [7], [6]. In this work we put this concept in operation by making an historical building a smart one.

Two relevant projects developed with the LinkSmart middleware (earlier called Hydra) [5], that we exploited also in this work (see Section III), are The Energy Aware Smart Home [10] and EnergyPULSE [11]. Both of them develop smart energy efficient applications in heterogeneous environments. The Energy Aware Smart Home includes smart metering and control of home appliances combined with novel user interaction applications. EnergyPULSE allows to monitor power consumption of devices and other values (e.g. temperature, presence) in office environments. It aims at providing a basis for new kinds of user-centric feedback systems in such environments.

These solutions take into account limited spaces, such as houses and offices. However, to move towards more complex and large buildings, a broader view is needed. Middleware technologies should implement the abstraction layers required to achieve interoperability also with existing BMS, while effectively supporting adaptation to environmental conditions and user comfort requirements.

The service-oriented middleware presented in [12], [13] and [14], have treated the interoperability issue. *Socrades* [15], [16], [17] is a modular, adaptive and open infrastructure forming a complete Service-oriented Architecture ecosystem that will make use of the embedded capabilities. The infrastructure components are specified and it is shown how they can interact and be combined to adapt to current system specificity and requirements. While most of these solutions focus on enabling Ubiquitous Computing and Internet of Things applications, this work targets smart energy efficient buildings and aims to provide re-usable distributed components for integrating Building Automation (BA) technologies with UbiComp.

In [18], Stavropoulos et al. present *aWESoME*, a web service middleware for AmI environment. It allows the interoperability between heterogeneous devices again to provide a system that enables automation and energy savings in large buildings. Their purpose is comparable to our solution, but it differs in the approach. Indeed, our solution aims to provide event-driven and user-centric functionalities.

In addition to research projects, OPC Unified Architecture [21] should be noted as an example of an integration effort for typical BA technologies. However, following the vision of UbiComp and AmI, large buildings must be open to any kind of other commercial technologies.

With respect to previous work, the proposed service-oriented Energy Efficient Middleware aims to enable interoperability between heterogeneous devices, both wireless and wired, in order to enlarge the existing Building Management Systems. Particular emphasis has been given to users that live and work into the building, so we believe that an user-centric fashion is needed. Hence user behaviour patterns must be taken into account to develop efficient control policies that lead to energy savings. To overcome these issue, we have combined occupancy detection with schedule-based control. The proposed middleware has been designed following an event-driven approach by means of the *Rule Framework*, which is a specific part of the middleware responsible to perform actions when an event is triggered. As a result, event-driven energy management policies can be developed in an efficient and hardware independent way. The main contribution of this paper is a complete design of a novel smart building middleware and its working implementation of a complete building management system in a real historical building. Table I provides an overview of the main features of the more relevant related middleware. It can be observed that the proposed solution provides relevant features not present in other approaches founded in literature.

III. The LinkSmart Middleware

The coexistence of several Heterogeneous technologies and a lack of interoperability among them is a well-known problem in the worlds of Ubiquitous Computing, Ambient Intelligence, and the Internet of Things. While for classic Building Management Systems (BMS), efforts like OPC UA try to solve these problems by providing abstraction layers, we have to consider that other technologies find their way into the buildings as well. Future smart building systems will be UbiComp and AmI environments that have to deal with multiple, different kinds of devices, applications, and technologies. To cope with these issues of interoperability and be open to future developments, we employ a middleware approach. We start from the open source LinkSmart[1] middleware [5], which is a generic service-oriented middleware for Ubiquitous Computing, and further develop it into a middleware for smart energy efficient buildings. This middleware provides reusable and extensible components and concepts for re-occurring tasks and problems in future smart buildings.

LinkSmart provides the developers a set of components (called managers), designed in the fashion of service-oriented architecture. Each manager exposes certain functionality as a Web Service. LinkSmart applications and prototypes have been developed for different application scenarios such as energy efficient smart homes [10] and

[1] http://sourceforge.net/projects/linksmart/

TABLE I
COMPARISON BETWEEN MIDDLEWARE

Middleware	Event Driven	User Centric	Rule Engine	Device interoperability		Domain
				Wireless	Wired	
Dog home gateway [8], [9]				v	v	Home
Socrades [15], [16], [17]				v	v	Generic
aWESoME [18]				v		Public Buildings
Energy Efficient Middleware	v	v	v	v	v	Public Buildings

office environments [11]. For the smart energy efficient buildings middleware we mainly employ and extend the following LinkSmart features and concepts:

- Secure Communication. Web Service calls are routed through the LinkSmart Network Manager, which creates a SOAP (Simple Object Access Protocol) tunnel to the requested service endpoint [22]. This concept allows direct communication among all devices inside a LinkSmart network, no matter if they appear behind a firewall or NAT (Network Address Translator). The LinkSmart addressing scheme allows devices to transparently publish and use services anytime anywhere regardless of network boundaries or fixed service endpoints. Furthermore, LinkSmart provides components for enabling message encryption and trust management [23].
- Event-based Architecture. The LinkSmart Event Manager provides a publish/subscribe service for LinkSmart Web Services. This allows the development of loosely-coupled event-based systems. In smart buildings, where we deal a lot with sensor events, this mechanism is a key requirement to develop systems and applications.
- Proxy. The Proxy is a concept that describes the integration of a specific technology into a LinkSmart application. Simply put, it means abstracting a certain technology, device or subsystem to Web Services and registering it at a Network Manager. A proxy acts as a bridge between the LinkSmart network and the underlying technology. It translates whatever kind of language the low-level technology speaks into LinkSmart Web Services so the low-level technology can be used transparently by any other LinkSmart component. This concept allows us to use each low-level technology transparently inside the LinkSmart network.

LinkSmart components can be deployed in centralized or distributed fashion. Thanks to the service-oriented approach, we have a high degree of freedom when planning a deployment.

IV. MIDDLEWARE FOR ENERGY EFFICIENCY

The Energy Efficient Middleware, which is built on LinkSmart middleware, consists of three-layered architecture with an integration layer, middleware layer and application layer as shown in Figure 1.

The *Integration Layer*, which is the lowest layer, is responsible to enable the interoperability between different technologies. The *Middleware Layer* provides components

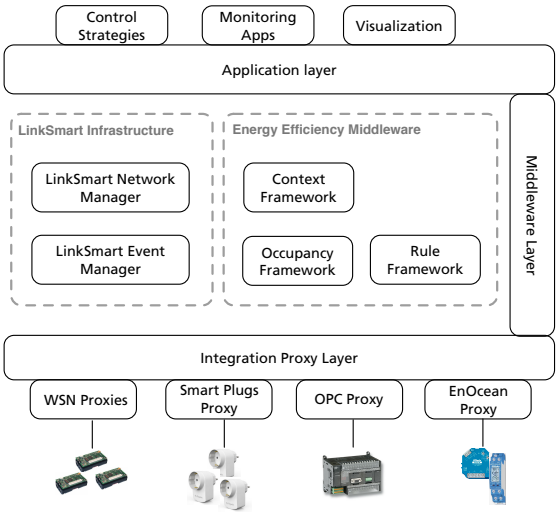


Fig. 1. Software Infrastructure

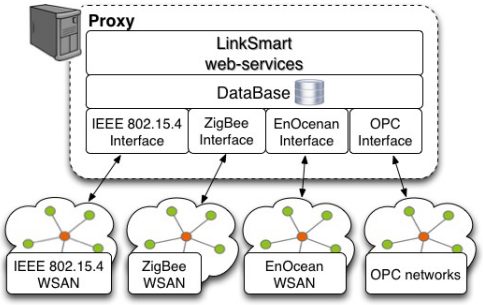


Fig. 2. Proxy scheme.

specifically designed for energy efficient smart building applications, which should support the management of reoccurring tasks. The *Application Layer* provides a set of applications that make use of the integrated system and information that is available.

The rest of this section describes in more detail all the components for each layer.

A. Integration Layer

The proposed infrastructure leverages upon an ICT infrastructure made of heterogeneous monitoring and actuation devices, such as Wireless Sensor and Actuator Networks (WSAN). In order to improve backwards compatibility, the infrastructure supports also wired technologies that exploit different protocols, such as BACnet, LonWorks, etc.

The Proxies represents the lowest layer of the proposed Energy Efficient Middleware (Figure 1), which is

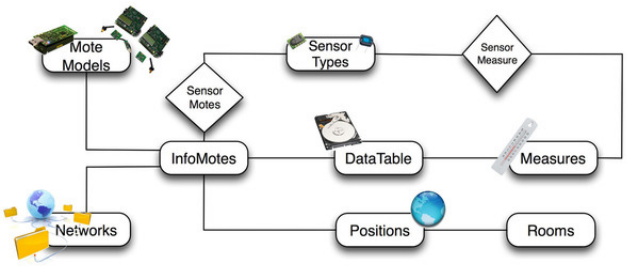


Fig. 3. Local Database scheme.

in charge of enabling the interoperability between heterogeneous technologies exploiting a Web Services approach. Its main purpose is to allow the remote devices' control. Furthermore, it allows the remote reconfiguration of sensor node parameters, such as the sampling rates of monitored physical quantities. Starting from the concept of LinkSmart's Proxy, we extended its functionalities in order to provide flexibility and reliability to the whole infrastructure with respect to possible backbone network congestions or failures. Indeed all the environmental data, coming from the sensor nodes, are collected into a local database and pushed into the infrastructure via an event-based approach thanks to the Event-Manager. Figure 2 shows in detail how we develop them. The Proxy can be seen as a software consisting of three sub-layers: i) Device Interface; ii) Local Database and iii) Web Services Interface.

The Proxy runs in a PC and communicates directly with the heterogeneous networks receiving information from various devices, regardless of the adopted communication protocols, hardware or the network topology. Hence, each network needs a dedicated software interface, which is the key to ensure the communication. It interprets the environmental data (e.g. temperature, humidity, etc.) and stores them in the integrated database (DB), which represents the second layer of the Proxy's stack.

As shown in Figure 3, the local database consists of ten tables to store: i) environmental information; ii) end-node hardware and configuration settings; iii) information about the motes' position in the considered rooms. In detail, *DataTable* collects the environmental measurements coming from the devices. The *Measures* table lists the types of physical property measures carried by the end-nodes. *MoteModels* contains the list of deployed commercial devices used for monitoring and controlling. *Networks* saves the information to identify a device network, such as communication channel and Network name. *InfoMotes* table contains the list of the configuration settings for all the motes in the networks (eg. sampling time). *SensorTypes* stores the list of all the possible sensors wired on the adopted end-nodes, and *SensorMotes* reports exactly what kind of sensors are installed for each mote. The table *SensorMeasure* shows the association between the sensor and the physical property that it detects. *Rooms* saves the architectural rooms' features managed by a Proxy. Finally, *Positions* stores the coordinates, associated to a room, where each device is deployed.

The LinkSmart Web service layer interfaces the device networks to the rest of the infrastructure, making the remote management and control easier. At that layer, the real-time data, collected by devices, are immediately sent to the cloud thanks to the Event-Manager. In order to avoid any additional delay in the real-time communication, the publication and storing of the incoming information are executed at the same time using two different threads, so the management of the database does not affect the publication of real time data.

Particular emphasis was given to the possibility to reconfigure each node, changing, for instance, some parameters about power management. In this scenario, the end-user sends the new configuration via the middleware to the Proxy and stores it in the DB. Then, the new settings will be automatically sent to the receiver device, when it wakes up from the sleeping period, through the specific network software Interface.

Particularly, we have developed different interfaces to manage respectively the following WSANs:

- Plugwise and ST Microelectronics Smart Plug commercial end-node to monitor power energy consumption and to switch on/off the appliances connected to the mains. Both of them exploit the ZigBee protocol;
- Our end-node prototype built on Texas Instruments CC2530 system on chip to monitor air temperature and illuminance leveraging the ZigBee protocol;
- Crossbow Telos rev. B open source end-node to monitor air temperature, relative humidity and illuminance, which exploits IEEE 802.15.4 communication protocol;
- EnOcean protocol stack commercial end-nodes to monitor air temperature, relative humidity, illuminance, occupancy and to actuate heating and lighting plant, respectively.

In addiction, an interface has been developed to allow the interoperability with the OPC Unified Architecture, which incorporates all the functionalities provided by different standards, such as BACnet or LonWorks. Hence, the backwards compatibility with wired technologies is enabled and integrated into our middleware.

Thanks to its modularity based on the Proxies deployment, the proposed Energy Efficient Middleware is suitable for integration and extension of the already existing BMS with new commercial-off-the-shelf sensors and actuator networks.

B. Middleware Layer

1) Event-Based Communication: Building Automation systems typically need to react upon events happening in the building. Hence, sensors publish events leading to a certain reaction, such as switching on the light upon an incoming motion event. The LinkSmart Event Manager provides us with the basic functionality of a topic-based publish/subscribe mechanism for LinkSmart Web Services.

As described in the previous section each of our low-level technology proxies publishes sensor events to an Event

Manager. These events are typically sensor measurements such as temperature, motion, brightness, power consumption, etc. Each event is published under a certain topic including the id of the sensor it belongs to and contains the measurement and a timestamp. Event topics are based on a hierarchical format providing basic semantic information about the type of event. For example, an event topic for publishing a simple temperature measurement would look like this:

$$MEASUREMENT/SENSOR/1234/Temperature$$

where $MEASUREMENT/SENSOR$ is an identifier for the type of event, 1234 is the sensor id and $Temperature$ is the type of measurement.

Using this kind of event topic format, software components interested in certain events can subscribe for those. Wildcards can be used for event subscription to subscribe to groups of events. For example, an application that would be interested in all sensor events (like a central persistence application) could subscribe for the topic

$$MEASUREMENT/SENSOR/.*$$

2) Context and Ontology Frameworks: The Context and Ontology Frameworks are two complementary components, which together manage semantic knowledge about the application domain and the implemented system. This includes meta-data about sensors and actuators but also their relation to domain model objects such as appliances, buildings and rooms. Semantic knowledge is stored in a RDF (Resource Description Framework) database management system (OWLIM) and can be queried and manipulated through a SPARQL API. SPARQL is a query language for retrieving and manipulating data that is stored in RDF format. In addition, for application developers the Context Framework provides a convenient entry point by exposing a simple JSON (JavaScript Object Notation) API. Hence, developers can query any kind of information from a rich domain model. This could be the location or capabilities of a sensor but also a list of all sensors in a room, or an actuator with a certain control capability.

3) Occupancy Framework: Occupancy of rooms and spaces, both private and public, has a major impact on defining and assessing efficiency of HVAC and lighting systems in public spaces and building and thus represent a key factor for correct accounting of consumption and its optimization [24]. While a wide number of heterogeneous technologies for measuring or estimating occupancy are available, aggregation and exploitation of occupancy information still remains a challenge. For this reason, the middleware layer includes an Occupancy Framework which is in charge of processing raw occupancy data monitored in the environment, performing processing and fusion tasks [25] and integrate the results with existing context information, so it can be finally used for energy assessment, optimization and forecast.

Occupancy can be observed in real-time using a very large number of heterogeneous sensors [26] exploiting very diverse mechanisms. It can be observed that the choice of the optimal solution really depends on the specific domain and that more than one technology might be in place in a given environment. For such reason it is fundamental to adopt a single, consistent occupancy model. Within the proposed middleware, the occupancy model is used, on the one hand, to facilitate fusion among different occupancy sources and, on the other hand, to ease handling of occupancy information by applications.

Figure 4 provides an overview of the Occupancy model adopted in the proposed Energy Efficient Middleware.

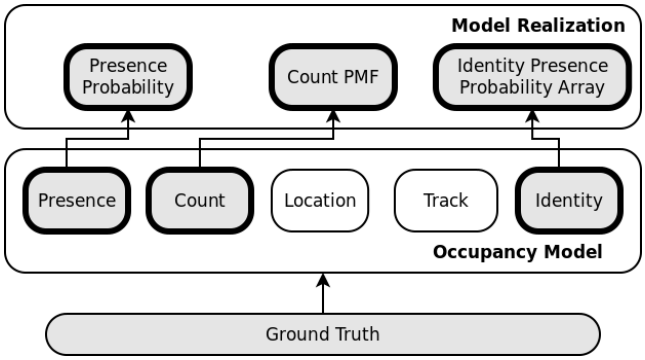


Fig. 4. The occupancy model

The model proposed in [26] including presence, count, location, track and identity has been initially considered and, based on the initial requirements, simplified to only include parameters which are relevant for energy optimization purposes. While only presence information has been considered sufficient, also the count and identity model has been kept to cope with specific use-cases such as the optimization of crowded rooms or the application of personalized energy set-points based on the identity of present people. To ease processing and fusion of occupancy data, a multi-modal model has been chosen [27]. Presence can be described as $P_{presence}$ i.e. the probability of having at least one person in the selected area. Count C_{PMF} is modelled as a Probability Mass Function (PMF) i.e. a statistical distribution including, for each unit i the probability $C_{PMF}(i)$ that at least i persons are present in the selected area. Identity I is modelled as plain array of all identities, which have been observed in the selected area, each with its presence probability.

Figure 5 provides an overview of the process followed by the occupancy framework to synthesize occupancy information.

Occupancy-related events are collected from the field by means of proxies, which generate LinkSmart Events carrying raw occupancy readings. In current status support for the following physical sensors is available: PIR (Passive Infra-Red) sensors, US (Ultra-sound) sensors, Hall Effect contact sensors. Moreover, occupancy information can be inferred from "Check-in" information (voluntarily provided by users declaring their presence in rooms) and internet-based Time-tables configured to store known occupancy schedules of rooms i.e. any web-based calendar using the iCalendar standard [28].

Raw occupancy-related events are processed by a set

of Level 1 Occupancy Proxies hosting technology-specific moving-window algorithms suitable to process one or more type of sources and derive a homogeneous occupancy event in terms of $(P_{presence}, C_{PMF}, I)$. Such pre-processed occupancy information is further fused on a per-room basis by Room-oriented Occupancy proxies.

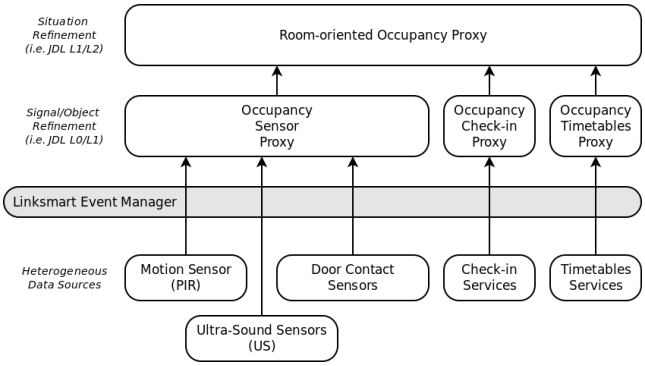


Fig. 5. Occupancy Framework overview

All available components of the occupancy information model of each room are made available within the middleware and can thus be exploited by control strategies and monitoring applications. In the deployment described in following sections, only presence probability has been used for control decisions through a threshold-based mechanism i.e. the control strategy assumed room occupied if presence probability was higher than 0.75. Nevertheless the Occupancy framework allows for more complex control strategies e.g. commanding the windows based on the count of people in the room or changing the set points for temperature based on the presence of a known user, which has specific thermal preferences.

4) Rule Framework: Typical building management functions can be expressed in rules: The system listens to certain events, processes them based on given knowledge and algorithms and performs a resulting action. Hence, a specific control strategy can be developed composing different rules between them. Usually, in closed systems these rules are rather limited, e.g. HVAC control is often based on simple schedules and a temperature set point. In contrary, our Rule Framework allows a fully flexible implementation of any kind of rule-based system. The framework provides standard interfaces as a basis for specific rule implementations. These rule implementations can be plugged together in a rule engine that executes the rules on incoming events. Rule logic and contextual information needed to execute a rule are kept separately, following the principle of separation of concerns. This allows to reuse rule implementations in different contexts, e.g. to apply the same HVAC control strategy in different rooms, but with different settings, depending of the peculiarities of the room itself or even its occupants.

C. Application Layer

In the proposed distributed infrastructure (Figure 1), the *Application Layer* represents the highest layer. It is dedicated to development of distributed event-based user-centric applications in order to manage buildings and post process data coming from the lower layers. At that level the interoperability between different devices is enabled as well as between third-party software.

In order to manage buildings, deployed devices and to promote user power awareness we developed: i) the HVAC Control System ii) an Android App and iii) a Web Portal.

The *HVAC Control System* implements the rules described in Section V to optimize thermal energy consumptions.

The *Android App* is a mobile software suitable for Android Tablets. It has been developed to increase the building awareness and enhance any possible maintenance works providing building information exploiting both Augmented and Virtual Reality.

The purpose of the *Web Portal* is to make the end-user aware about energy consumption and to encourage energy saving by displaying statistics of the collected environmental data coming from the heterogeneous devices through graphs and tables. Furthermore, it allows the user to compare buildings, offices and public spaces.

V. Control System

The presented middleware allows developing monitoring and event-driven control systems in a service-oriented way, abstracting from heterogeneous technologies. On the integration layer, technology proxies publish sensor events through the LinkSmart Event Manager. Monitoring and control application subscribe for events, also via the LinkSmart Event Manager. Both, subscribing and publishing is done by Web Service calls to the Event Manager.

The HVAC control system only listens to certain sensor events, relevant for controlling the HVAC system. In the following we describe this HVAC control system. The control strategy is based on intermittent use of the HVAC systems. The economic advantage of intermittent use of HVAC systems in intermittently-occupied buildings (e.g. universities, office buildings) is no longer in doubt [29]. Our strategy combines schedule-based control with occupancy detection. Schedules are defined for each test room, typically modelling standard working hours, Monday to Friday from 9:00 AM to 6:00 PM with one hour lunch break. Table II shows the defined temperature set points for the different periods specified by scheduled and detected occupancy in a room.

TABLE II
Set Point Selection

Scheduled Occupancy	Occupancy Detected	Set Point (° C)	
		Winter Period	Summer Period
True	True	23	26
True	False	20	29
False	True	23	26
False	False	6	35

Depending on the difference between the temperature set point and the actual indoor temperature, the fan speed of

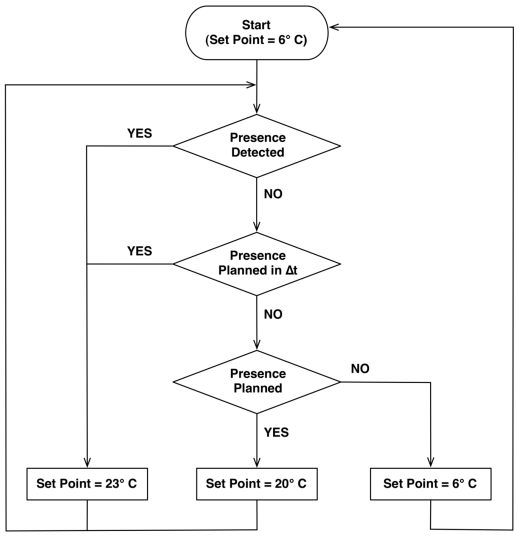


Fig. 6. HVAC Control Strategy

the fan coil is selected. In our case studies, fan speed can range from 0 to 3.

To avoid turning on the fan coil in case of a very short period of presence (e.g. a person just entering and leaving an office very quickly) we have included a system of timers, implementing a *Timer-based Occupancy Confirmation*. After recognizing a positive occupancy event, the system waits for some time before starting up the HVAC. Vice versa, the system waits some time after a negative occupancy event has been recognized before switching off the HVAC.

Another important feature of the HVAC control system is a period of pre-heating after a long period of non-occupancy, e.g. in the morning. To guarantee a comfortable temperature in the morning, the system starts to heat up the room a certain time before the scheduled occupancy starts. The duration differs for each room, depending on its size, walls, exposition, etc. Figure 6 shows a flow chart of the HVAC control strategy for the winter period. Note that pre-heating is only applied in winter. During summer it is assumed that buildings cool down to a comfortable temperature during the night.

The implementation of the HVAC control strategies is fully integrated with the Middleware, based on the Rule Framework as described in section IV-B4. The HVAC control for each room is independent so that all configuration parameters (e.g. pre-heating period, temperature set points) can be set depending on the characteristics of the room. The rule engine, managing a set of rules, expects certain input events to act on. For the HVAC control these are occupancy, actual indoor temperature and time. Based on these input parameters and the configuration parameters the rule engine calculates the set point temperature and the according fan speed:

First, the *occupancy confirmation rule* computes i) if a room is occupied and ii) when it is planned to be occupied. This computation is based on occupancy information, which exploits both motion sensor events and post-processed data from the Occupancy Framework. However the probabilistic approach has been presented as possible feature but it is not used in the presented algorithm. In our case the definition of occupancy is based on motion sensors and timers: The system waits for two consecutive motion events before switching to occupied mode and waits a certain time before switching back to not-occupied mode. Second, depending on the occupancy mode, the *set point rule* calculates the set point temperature based on occupancy, schedules, and pre-heating periods. Then, based on the delta between the actual indoor temperature and the set point, the *fan speed rule* calculates the fan speed (0..3) to heat up or cool down the room. After processing all rules the fan speed will be published as an event to activate the respective fan coil controller. Note, that the resolution to identify the right fan coil controller is provided by the Context Framework. For application development no knowledge about concrete sensor ids or low-level technologies is needed.

VI. Case Study

The *historical building*, which was investigated in this research, dates back to the beginning of the 16th century. This site is approximately $20,000m^2$. The presence of frescos and the tight walls require a careful placement of sensors and design of the network topology.

In order to verify the system effectiveness, an empirical test has been performed. We chose two rooms as *Test Rooms*, where the innovative proposed control strategy has been deployed. Simultaneously, we monitored two *Reference Rooms*, with the same structural characteristics of test rooms. The different rooms chosen cover a good diversity in insulation, thermal inertia, windows areas and internal gains or occupancy. In the reference rooms, fan coils are running from 5:00 a.m. to 8:00 p.m. during workdays and from 6:00 a.m. to 12:00 a.m. on Saturdays. On Sunday and during holidays they are off. Moreover, users are free to change the fan speed from 0 to 3 as they prefer. The rooms were selected to implement and extend the existing Building Management System with new Sensors and Actuators Network infrastructure. The rooms were chosen for case studies based on the following criteria: energy saving potential according to their architecture, services and occupancy characteristics.

The first couple of rooms consists of two *Royal Rooms* used as private offices. Every room has an area of about $62.40m^2$, a frescoed domed roof with the maximum height of $7.25m$ and the minimum of $5.70m$, and a capacity of 3 working desks. The second couple of rooms consists of two private offices, each of them has an area of about $16.10m^2$, a pitched roof with the maximum height of $3.90m$ and the minimum of $2.68m$, and a capacity of 2 working desks.

In order to enable a finer grain monitoring and management system for the selected environments, it has been installed the following devices:

- to detect the users presence
- to monitor fan-coil internal air temperature;
- to control the fan-coil set points and fan speeds;

- to monitor the indoor air temperature and to control the fan-coil. While in the Reference Rooms the users can use it to change the fan speed autonomously as they prefer. Instead in the Test Rooms it acts only as temperature sensor and the control commands have been disabled;

In order to extend the existing systems in the historical building, we adopted wireless devices mainly based on EnOcean technology, which are configured as mesh networks.

VII. Results

In this Section we discuss the results about the HVAC control system (see Section V) deployment in the real-world case study, introduced in Section VI.

The deployment of the whole Energy Efficient Middleware and the new heterogeneous devices begun in December 2012. The first step consisted of a monitoring only phase to collect environmental data. Then, we gradually puts in the selected rooms more technologies in terms of sensors and actuators. In the historical building the first HVAC control experiments comes back to January 2013. Then, the deployment of control strategies, as described in Section V, has been started on February 19th. Hence results about both heating and cooling are discussed.

During the test, particular emphasis has been given to user comfort. In fact the absolute value of the difference between indoor air temperature in reference and test rooms is lower than $1\,^{\circ}\mathrm{C}$, so discomfort is not perceived by users.

A. Energy Metrics

Before discussing results, we introduce here the metric adopted to evaluate relative energy savings between reference and test rooms.

The first parameter is $Q_{v,i}$, which is the air flow of the fan coil for each speed i. Using the percentage of the maximum air flow, we quantify it with the following formula:

$$Q_{v,i} = Q_v \cdot p_i \tag{1}$$

with Q_v the maximum air flow of the fan coil in kg/s and p_i the air flow in % in relation to fan speed (cf. Table III). The values for both Q_v and p_i are provided by the fan coil datasheet.

TABLE III
Air Flow per Fan Speed

Fan Speed (i)	Air Flow (p_i)
3	100%
2	80%
1	60%
0	0%

Furthermore knowing the following parameters: i) fan coil temperature, ii) fan speed, iii) fan coil air flow, iv) indoor temperature, and v) air heat capacity (i.e. a constant in our case), we can estimate the instantaneous power at each time step of the fan coil with the following formula:

$$P(t) = Q_{v,i} \cdot C_p \cdot (T_f(t) - T_i(t)) \tag{2}$$

with $P(t)$ power of fan coil in Watts, $Q_{v,i}$ the air flow at each time step in kg/s, C_p the air heat capacity in $J/kg/K$, T_f the fan coil temperature, and T_i the indoor temperature at each time step. Hence, the energy consumption are calculated as:

$$Econs = \int_{day} P(t)dt \tag{3}$$

Finally, the relative energy savings are given by:

$$Esaved = \frac{Econs_{ref} - Econs_{test}}{Econs_{ref}} \tag{4}$$

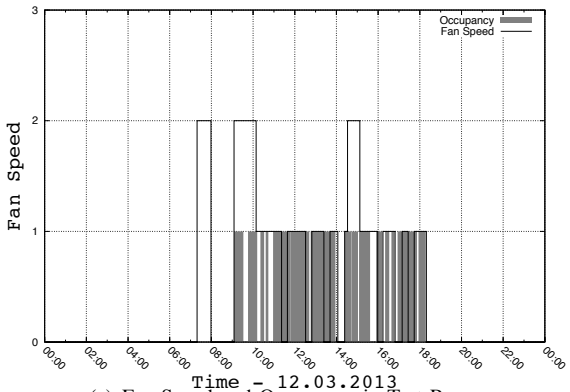
B. Runtime Observations

Figure 7 reports the results about the *Royal Rooms* (see Section VI) on 12th March 2013. It shows observations and differences between reference and test room. On this day, fan speed has been on 1 in the reference room all day long. From interviews with the occupants we know that they tend to keep the fan coil running almost all time and rarely change the speed.

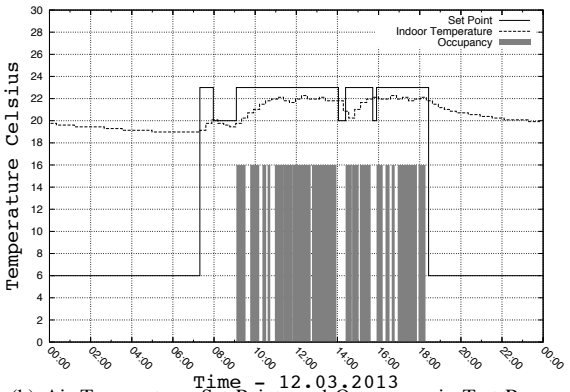
Figure 7(a) shows the fan speed and occupancy pattern. From the 7:00 AM to 8:00 AM, we can observe the preheating phase. It starts even if there is no presence detected in the room and switches on the fan-coil on fan speed to 2. Then, from 9:00 AM to 6:00 PM, the HVAC control system starts working taking into account the users' presence in the room. The occupancy data in Figure 7(a) shows post-processed data from the motion sensor deployed in that room. It shows the occupancy after applying the *Timer-based Occupancy Confirmation*. Moreover, we can observe that the control system turns off the fan-coil when the Indoor Temperature reaches the set point defined in Table II.

Figure 7(b) relates the Indoor Temperature with the Set Point, as defined in the HVAC control strategy (Table II), and the office Occupancy. We can observe how the defined Set Points variation successfully affects the Indoor Temperature trend as required. Moreover, we can note how the *Timer-based Occupancy Confirmation* influences the Set Points variation. Indeed during the day, the office is unoccupied several times for short periods but the Set Point is constant at $23\,^{\circ}\mathrm{C}$. It changes from $23\,^{\circ}\mathrm{C}$ to $20\,^{\circ}\mathrm{C}$ from 2:00 PM to 2:30 PM because of lunchtime and then from almost 3:30 Pm to 3:55 PM.

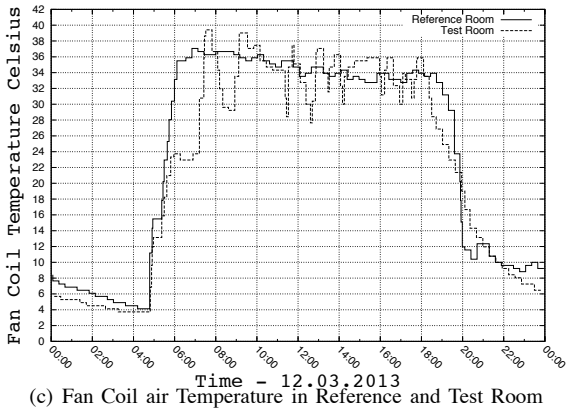
Finally Figure 7(c) shows the different patterns of fan-coil activity, expressed in fan coil temperature, in both rooms, reference and test. Instead, Figure 7(d) reports the energy consumption $P(t)$ calculated following the formula 2 for both reference and test rooms. While the fan coil in the reference room stays at the same speed during the whole day, we can see the changes based on our control strategy in the test room. The comparison indicates the potential energy savings: i) during sunrise, between 5:00 AM and 7:00 PM; ii) in the evening, among 6:00 PM and 8:00 PM; iii) during non-occupied periods throughout the day, thanks to the combination of occupancy schedule and occupancy detection.



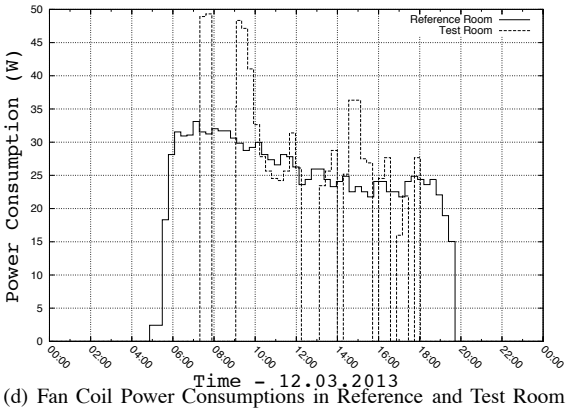
(a) Fan Speed and Occupancy in Test Room



(b) Air Temperature, Set Points and Occupancy in Test Room



(c) Fan Coil air Temperature in Reference and Test Room



(d) Fan Coil Power Consumptions in Reference and Test Room

Fig. 7. Royal Rooms runtime observations

In conclusion, for the analysed working day the energy consumption in the reference rooms, $Econs_{ref}$, was about 1.23 kWh, however the $Econs_{test}$ was approximately 0.84 kWh. Hence, ensuring the same comfort to the users, the relative saving was about 30%.

C. Energy Savings

Table IV presents the results about Heating strategy performed in the private offices during one selected week, 4th - 8th March. Both rooms were unoccupied from Monday to Wednesday. It worth emphasizing that in the reference room the users can change the fan speed autonomously as they prefer. Even if the results show that in the reference room users turn on the fan-coil responsibly, there are consumption, ranging between 0.01 kWh and 0.33 kWh, that are saved in the test room. On Thursday the test room was occupied all day long, while the reference was still unoccupied, wasting again 0.19 kWh. Finally on Friday both rooms were occupied and thanks to the HVAC control strategies it has been saved almost 0.548 kWh delivering the same comfort. Indeed the difference between indoor air temperature in reference and test rooms is lower than 1 °C.

Table V reports the results about the Cooling strategy deployed in the same private offices. Specifically, it has been selected three days in July, the hottest, the coolest and a normal day. During the hottest day, in the reference we recorded 7.53 kWh and in the test 1.60 kWh, thus obtaining a savings of about 79%. However, during the coolest day, which was cloudy, the energy waste was reduced about 82%. Finally, during a normal day, the average energy savings is around 50%. Even during these tests, it has been ensured the same comfort to the users in both rooms.

TABLE IV
HEATING SAVINGS ON HISTORICAL BUILDING PRIVATE OFFICES

Day	Cons. on ref. room (kWh)	Cons. on test room (kWh)	Savings	
			(kWh)	(%)
Monday	0.14	0.00	0.14	100%
Tuesday	0.01	0.00	0.01	100%
Wednesday	0.33	0.00	0.33	100%
Thursday	0.19	1.20	-1.01	-518%
Friday	0.98	0.432	0.548	57%

TABLE V
COOLING SAVINGS ON HISTORICAL BUILDING PRIVATE OFFICES

Day	Cons. on ref. room (kWh)	Cons. on test room (kWh)	Savings	
			(kWh)	(%)
Hottest	7.53	1.60	5.93	79%
Normal	2.38	1.14	1.24	52%
Coolest	0.94	0.17	0.77	82%

VIII. UNIVERSALITY OF THE ENERGY EFFICIENT MIDDLEWARE

The proposed Energy Efficient Middleware has been designed to enable interoperability between heterogeneous commercial devices both wireless and wired, regardless of their network topologies. Thanks to this feature, it is suitable for any kind of public building and spaces. Indeed, it can be deployed in modern buildings exploiting the existing building automation system, which often is wired,

but it can also be deployed in historical buildings, where wireless technology are suitable because they do not need construction works. Moreover, thanks to the Rule Manager, it provides a tool for the implementation and customization of fine grain control strategies for both HVAC and Lighting systems.

In this paper we have discussed a real case study, which involved an historical building, but Jahn et al. have applied the middleware in an energy aware smart home scenario [10]. The scenario deals with demand-response issues simulating a washing machine controlled based on flexible energy prices and an additional mobile app to visualize energy consumption of appliances to home-owners.

Furthermore, an earlier version of the energy efficient middleware has been applied for an energy-monitoring case study in an office environment [11], in a modern building. Integrated technologies included a wireless sensor network based on Arduino and ZigBee communication and smart plugs. This installation has also been used for the development of a pervasive game to motivate energy conserving behaviour in office spaces [30].

In addiction, a number of components of the proposed system are also being proposed in different application scenarios, leveraging the common Linksmart-based infrastructure. For example, a slightly modified version of the Context Manager and the Rule Framework are being evaluated for optimizing the energy efficiency in industrial environments [31]. In conclusion, we can affirm that, while the devices and the optimization goals in place differ significantly from the use case described in this paper, the proposed approach is generic enough to allow the use of similar formalisms and technologies for integrating devices, describing rules and implementing control strategies.

IX. Concluding Remarks

In this work the Energy Efficient Middleware has been described, which aims at improving energy efficiency of public buildings and spaces exploiting both event-driven and user-centric approaches. Moreover it allows interoperability among heterogeneous devices, both wireless and wired. Hence, exploiting the proposed middleware, the combination of different technologies, both existing and emerging, can be easily achieved.

The various middleware layers and the applications developed on top of it have been described. Moreover a HVAC control strategy has been implemented and the results about its real-world deployment have been discussed, which indicate that energy savings has been achieved.

Acknowledgment

This work was supported by SEEMPubS, DIMMER and Tribute, which are European FP7 research projects.

References

[1] European Parliament, "Directive 2010/31/EU of the European Parliament and of the Council of 19 May 2010 on the energy performance of buildings," 2010.

[2] Communication from the commission to the European Parliament, the Council, the European Economic and Social committee and the committee of the Regions, "Addressing the challenge of energy efficiency through information and communication technologies," 2008.

[3] M. Jahn, E. Patti, and A. Acquaviva, "Smart energy efcient buildings. a living lab approach." in *International Conference on Smart Grids and Green IT Systems*, Aauchen, Germany, May 2013.

[4] A. Caragliu, C. Del Bo, and P. Nijkamp, "Smart cities in europe," VU University Amsterdam, Faculty of Economics, Business Administration and Econometrics, Serie Research Memoranda 0048, 2009.

[5] M. Eisenhauer, P. Rosengren, and P. Antolin, "A development platform for integrating wireless devices and sensors into ambient intelligence systems," in *Sensor, Mesh and Ad Hoc Communications and Networks Workshops, 2009. SECON Workshops '09. 6th Annual IEEE Communications Society Conference on*, June 2009, pp. 1–3.

[6] C. Warmer, K. Kok, S. Karnouskos, A. Weidlich, D. Nestle, P. Selzman, J. Ringelstein, A. Dimeas, and S. Drenkard, "Web services for integration of smart houses in the smart grid," in *Grid-Interop - The road to an interoperable grid*, 2009.

[7] S. Karnouskos, "The cooperative internet of things enabled smart grid," in *Proceedings of the 14th IEEE International Symposium on Consumer Electronics*, ser. ISCE2010, Braunschweig, Germany, GE, November 2009.

[8] D. Bonino and F. Corno, "The dog gateway: Enabling ontology-based intelligent domotic environments," in *IEEE Transactions on Consumer Electronics*, 2008.

[9] F. Corno and D. Bonino, "Interoperation modelling for intelligent domotic environments," in *Lecture notes in computer science*, ser. LNCS 5859, 2009.

[10] M. Jahn, M. Jentsch, C. Prause, F. Pramudianto, A. Al-Akkad, and R. Reiners, "The energy aware smart home," in *5th International Conference on Future Information Technology*, May 2010.

[11] M. Jahn, T. Schwartz, J. Simon, and M. Jentsch, "Energypulse: tracking sustainable behavior in office environments," in *Proceedings of the 2nd International Conference on Energy-Efficient Computing and Networking*. New York, NY, USA: ACM, 2011.

[12] A. Sleman and R. Moeller, "Micro soa model for managing and integrating wireless sensor network into ip-based networks," in *Second International Conference on Computational Intelligence, Communication Systems and Networks*, ser. CICSyN, 2010, pp. 137–142.

[13] N. B. Priyantha, A. Kansal, M. Goraczko, and F. Zhao, "Tiny web services: design and implementation of interoperable and evolvable sensor networks," in *Proceedings of the 6th ACM conference on Embedded network sensor systems*, ser. SenSys '08. New York, NY, USA: ACM, 2008, pp. 253–266.

[14] M. Zamora-Izquierdo, J. Santa, and A. Gmez-Skarmeta, "An integral and networked home automation solution for indoor ambient intelligence," *Pervasive Computing, IEEE*, vol. 9, no. 4, pp. 66–77, 2010.

[15] D. Guinard, V. Trifa, S. Karnouskos, P. Spiess, and D. Savio, "Interacting with the soa-based internet of things: Discovery, query, selection, and on-demand provisioning of web services," *Services Computing, IEEE Transactions on*, vol. 3, no. 3, 2010.

[16] G. Candido, A. Colombo, J. Barata, and F. Jammes, "Service-oriented infrastructure to support the deployment of evolvable production systems," in *IEEE Transaction on Industrial Informatics*, vol. 7, no. 4, November 2009.

[17] G. Candido, F. Jammes, J. Barata, and A. Colombo, "Generic management services for dpws-enabled devices," in *35th Annual Conference of IEEE Industrial Electronics*., ser. IECON '09. IEEE, November 2009, pp. 3931 – 3936.

[18] T. G. Stavropoulos, K. Gottis, D. Vrakas, and I. Vlahavas, "awesome: A web service middleware for ambient intelligence," *Expert Systems with Applications*, vol. 40, no. 11, September 2013.

[19] [Online]. Available: http://getgreenbox.com/

[20] [Online]. Available: http://www.energy-aware.com/products-powertab.html

[21] [Online]. Available: https://www.opcfoundation.org/UA/

[22] F. Milagro, P. Antolin, P. Kool, P. Rosengren, and M. Ahlsén, "SOAP tunnel through a P2P network of physical devices," in *Internet of Things Workshop, Sophia Antopolis, Sep*, 2008, pp. 1–3.

[23] M. Hoffmann, A. Badii, S. Engberg, R. Nair, D. Thiemert, M. Matthess, and J. Schuette, "Towards semantic resolution of security in ambient environments," in *Developing Ambient Intelligence*. Springer Paris, 2008, pp. 13–22.

[24] V. Erickson, M. Carreira-Perpinan, and A. Cerpa, "Observe: Occupancy-based system for efficient reduction of hvac energy," in *10th International Conference on Information Processing in Sensor Networks*, ser. IPSN, 2011, pp. 258–269.

[25] J. Llinas, C. Bowman, G. Rogova, A. Steinberg, E. Waltz, and F. White, "Revisiting the jdl data fusion model ii," in *In P. Svensson and J. Schubert (Eds.), Proceedings of the Seventh International Conference on Information Fusion*, ser. FUSION 2004, 2004, pp. 1218–1230.

[26] T. Teixeira, G. Dublon, and A. Savvides, "A survey of human-sensing: Methods for detecting presence, count, location, track, and identity," *ACM Computing Surveys*, vol. 5, pp. 1–41, 2010.

[27] M. Hofmann, M. Kaiser, H. Aliakbarpour, and G. Rigoll, "Fusion of multi-modal sensors in a voxel occupancy grid for tracking and behaviour analysis," in *12th International Workshop on Image Analysis for Multimedia Interactive Services, Delft, The Netherlands*, vol. 1, 2011.

[28] B. Desruisseaux, "Internet calendaring and scheduling core object specification (icalendar)," RFC 5545 (Proposed Standard), Internet Engineering Task Force, sep 2009, updated by RFCs 5546, 6868. [Online]. Available: http://www.ietf.org/rfc/rfc5545.txt

[29] G. Fraisse, J. Virgone, and J. Brau, "An analysis of the performance of different intermittent heating controllers and an evaluation of comfort and energy consumption," *HVAC&R Research*, vol. 3, no. 4, pp. 369–386, 1997.

[30] J. Simon, M. Jahn, and A. Al-Akkad, "Saving energy at work: The design of a pervasive game for office spaces," in *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia*, ser. MUM '12. New York, NY, USA: ACM, 2012.

[31] P. Brizzi, H. Khaleel, P. Cultrona, F. Pramudianto, D. Conzon, M. Knechtel, R. Tomasi, and M. Spirito, "Bringing the internet of things along the manufacturing line: A case study in controlling industrial robot and monitoring energy consumption remotely," in *Proceedings of the 18th IEEE International Conference on Emerging Technologies and Factory Automation*, ser. ETFA 2013, 2013.