International Journal of Computer Networks and Communications Security

VOL.1, NO.6, NOVEMBER 2013, 216–224
Available online at: www.ijcncs.org
ISSN 2308-9830

# The Network Architecture Designed for an Adaptable IoT-based Smart Office Solution

**Karol FURDIK[1], Gabriel LUKAC[2], Tomas SABOL[3], Peter KOSTELNIK[4]**

[1, 2] InterSoft, a.s., Florianska 19, SK-040 01, Slovakia

[2] Department of Cybernetics and Artificial Intelligence, Faculty of Electrical Engineering and Informatics,
Technical University of Kosice, Letna 9, SK-042 00, Slovakia

[3, 4] Department of Banking and Investment, Faculty of Economics, Technical University of Kosice,
B. Nemcovej 32, SK-040 01, Slovakia

*E-mail: [1]karol.furdik@intersoft.sk, [2]gabriel.lukac@tuke.sk, [3]tomas.sabol@tuke.sk, [4]peter.kostelnik@tuke.sk*

## ABSTRACT

The paper presents a prototype of the Smart Office system, which was developed as one of pilot applications of the FP7 EU project ELLIOT, http://www.elliot-project.eu. The described solution is based on the LinkSmart semantic middleware, http://sourceforge.net/projects/linksmart/, an open source technology enabling a construction of distributed enterprise Internet of Things applications. LinkSmart was enhanced on features of seamless networking and interoperability of connected devices within the another FP7 EU project EBBITS, http://www.ebbits-project.eu. Principles of the system architecture and service-based networking are discussed and demonstrated on the Smart Office solution, employing the approach of user experience monitoring and continuous evaluation for adjusting the application to desired knowledge, social, and business parameters.

**Keywords:** *Internet of Things, Semantic Middleware, Peer-to-peer Networking, Smart Office, User Experience Evaluation.*

## 1 INTRODUCTION

The concept of Smart Office, as an application type belonging to the Internet of Things (IoT) domain, typically covers features supporting intelligent behavior of work environment, namely office rooms [1], [2]. A network of dedicated sensors, actuators, and various specialized devices is employed to adjust the settings of temperature, light intensity, humidity, vibrations, noise level, and other in-room environment parameters in accordance with preferences of users – office employees [3]. In line with the IoT principles [4], the networked sensors and devices are acting as autonomous and adaptable entities that exchange produced and consumed data by means of Internet-like protocols, provide services embedded directly in the devices, update their inner status and generate events as responses to changes in the outside context [5].

From the perspective of adaptable office room environment, the Smart Office concept is related to the domain of well established home automation solutions. The difference, however, is in aligning such solutions to the underlying business processes that drive daily operations in the office on a basis of projects, scheduled work tasks that are regularly performed by the office employees with the given roles. Besides the sensors and devices monitoring the conditions of work environment in terms of temperature, light intensity, etc., the Smart Office solution needs to communicate with various project management systems, databases and information systems that control a regular office work [6]. Comfort of individual users, which is often considered as the primary target in home automation applications, should be balanced in the Smart Office environment with other business-specific objectives such as an overall effectiveness and quality of work, energy savings, meeting the scheduled deadlines and project milestones, etc. Consequently, the challenge is to design a proper structure of IoT-based Smart Office application that would be able to support the following features:

- Flexibility and scalability, achieved by an adaptable architecture of networked sensors, devices, and information resources, empowered by a suitable communication network infrastructure, open interfaces, and service-oriented IoT middleware;
- User experience monitoring and evaluation functionality that enables adjustment of the applied solution to particular office type and specific business needs.

These two key issues have been specifically addressed in the European FP7 R&D projects EBBITS (http://www.ebbits-project.eu) and ELLIOT (http://www.elliot-project.eu), in which authors of this paper are/were involved. The former project EBBITS aims at investigation and development of a general architecture for IoT applications, based on an open source semantic middleware LinkSmart 2.0 [7]. The EBBITS project is namely focusing on an overall usability of IoT in mainstream enterprise systems, seamless networking and interoperability of devices, services, and information resources in end-to-end business applications [8], [9]. The later project ELLIOT uses the same LinkSmart middleware; however, it is more focused on an adjustment of IoT applications to the needs and requirements of users. The provided LinkSmart-based ELLIOT platform, which communicates with a basic IoT application in an open Living Lab environment [10], enables monitoring and evaluation of user experience in an on-the-fly mode. The Smart Office solution and the respective network architecture, presented in section 4, was designed and developed within the ELLIOT project as one of its pilot applications.

## 2 RELATED RESEARCH

Architectures for IoT applications have been already investigated in numerous R&D projects and research works, applied in commercial products, and reflected in several standardization activities. Detailed survey of approaches and available architecture frameworks can be found, for example, in [11] or [12]. Namely, the project SENSEI (http://www.ict-sensei.org) introduces a modular architecture of communication, resource-based business logic, and application maintenance layers, where physical devices and their interactions are internally represented by respective semantic models [13], [14]. Similar concept of semantic middleware for IoT has been proposed in the project HYDRA (http://www.hydramiddleware.eu) and consequently adopted in the LinkSmart solution [15], [16]. Furthermore, related issues on the context awareness of networked devices, secure RFID communication and processing of generated events, entity identification and discovery services, alignment with business processes, etc., have been addressed in several IoT-related R&D projects such as, for example, BRIDGE (http://www.bridge-project.eu), CUBIQ (http://www.cubiq.jp), and other projects cooperating within the IERC (http://www.internet-of-things-research.eu), the leading European initiative for research in the IoT domain.

A general architecture for IoT applications is specifically investigated within the project IoT-A (http://www.iot-a.eu), which also belongs to the IERC participants. IoT-A provides a reference architecture, which can be considered as a generalization of previous research approaches.

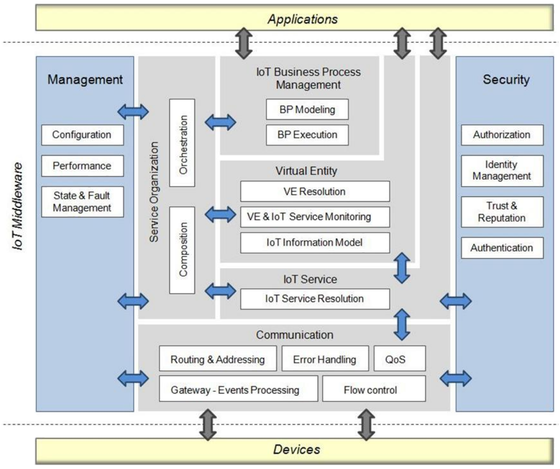


*Fig. 1. IoT-A reference architecture, functional view*

The functional view of the IoT-A reference architecture [17], depicted in Figure 1, follows the modular structure of functional blocks organized into layers, as it was proposed e.g. in SENSEI [14]. The middleware layer, encapsulated by the *Management* and *Security* modules, includes the *Communication* module that acts as a gateway to the connected physical devices. Data and interfaces provided by the devices are processed and transformed to respective services by means of the *IoT Service* and *Service Orchestration* modules. Connection to external applications and fusion of services into more complex structures of business processes is enabled by the *IoT Business Process Management* module. Internal *Virtual Entity* module serves as a virtualization mechanism that converts data resources and IoT services into a single level of abstraction; it enables an uniform manipulation with received/generated data and services according to a business logic expressed by applied semantic models or business rules. The virtualization is driven by the underlying *IoT Information Model*, proposed as an object representation of the Virtual Entity and related concepts for connected services, their descriptions, attributes, actual data values, metadata, and associations between the services.

Networking of physical devices, sensors and various information resources of connected legacy systems is enabled in the IoT-A reference architecture by the Communication module, which acts upon the IoT Information Model and includes a specific semantic representation of the service - resource - device interaction. From the technology perspective, the IP addressing and its compressed version defined by 6LoWPAN [18] is proposed for achieving addressability of each service or resource regardless of the hosting device. The unambiguous addressing of resources and devices can be handled by solutions based on HTTP mapping mechanism such as, for example, CoAP (Constrained Application Protocol, [19]).

The presented approaches towards the modular IoT architecture have been already employed in several prototype applications of home automation, room networking [13], or business processes support in enterprise systems [20]. However, some of features required by Smart Office IoT applications, namely the issues of scalability and performance in a real business environment, as well as the adaptability based on the user experience evaluation, have not been fully addressed and could be seen as opportunities for further investigations.

## 3 IOT-ENABLING MIDDLEWARE

The LinkSmart middleware [7], briefly introduced in previous sections, provides a semantic and networking infrastructure for building enterprise IoT applications. Conceptually it follows the same principles of modularity and structure of functional layers as those proposed by the IoT-A reference architecture. The current implementation of LinkSmart 2.0, produced and released as an outcome of the project EBBITS, provides a unified semantic representation of devices as data resources providing IoT services. The inner structure of functional modules and components, so-called managers, together with an outlined data flow inside the LinkSmart middleware, is depicted in Figure 2.

The core *Thing - Virtual Entity* module acts as the main semantic virtualization mechanism for representing and handling all the data and services provided by the connected sensors, devices, and information resources of various types. To ensure the compliance with the IoT-A reference architecture, the *IoT System Ontology* of LinkSmart has been aligned to the *IoT Information Model* of IoT-A [17], [21]. Namely, the *iot-a:Entity* concept was taken as the top-most class and its subclass *iot-a:PhysicalEntity* was taken as the parent concept of the *ebbits:DigitalThing* – former top class of the ebbits-LinkSmart system ontology. Furthermore, the *ebbits:DigitalThing* was extended on the obligatory *iot-a:Attribute* property, which is defined by the *iot-a:VirtualEntity* concept. On the service representation level, the *iot-a:Service* was taken as the parent concept for the *ebbits:Service*.

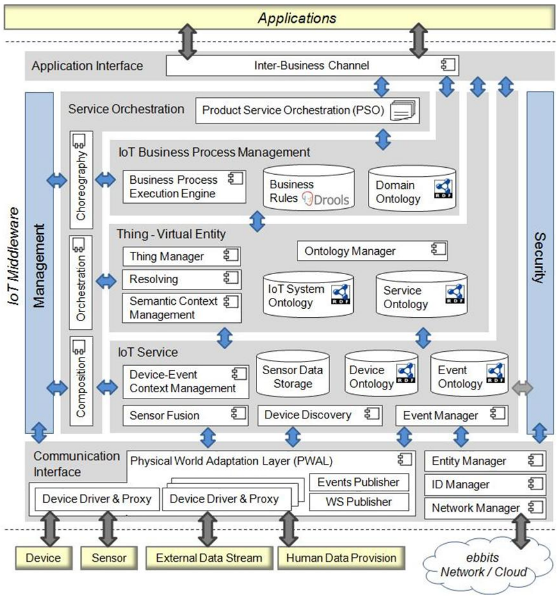


*Fig. 2. Modular architecture of the LinkSmart IoT middleware in the EBBITS project*

Semantic infrastructure of the RDF ontologies, provided by the *Thing - Virtual Entity* module, enables uniform manipulation with the services extracted from the connected information resources. This way, various heterogeneous devices, sensors, data streams, external databases, or even data produced as outcomes of monitored human activities can be handled in an interoperable manner. Composition of these virtual services into required service chains is enabled by the *Service Orchestration* module, which dynamically builds an output sequence of services in accordance with the business rules specified within the *IoT Business Process Management* module. To formally define which events are relevant and which services need to be invoked during an orchestrated service chain, the so-called *Product Service Orchestration* (PSO) list is generated in the XML format and is then provided for external applications by means of the *Inter-Business Channel* component.

The connection of various types of devices to the LinkSmart middleware is enabled by the *Physical World Adaptation Layer* (PWAL), containing a set of drivers for communication with a different device family together with the technology-specific logic needed to manipulate the adapted physical devices. PWAL in its current version supports the communication based on the Wireless Sensor Network (WSN) technology (i.e., ZigBee and 6LoWPAN protocols), WLAN, Bluetooth wireless data transmission, as well as the EPC Global standard of the Low Level Reader Protocol (LLRP) for RFID readers. Events generated by the connected devices are captured in PWAL, transformed to respective OSGi-compliant web service calls, and then forwarded to the inner middleware modules by means of *Events Publisher* and *WS Publisher* components. Supporting components of the PWAL, such as *ID Manager* and *Entity Manager*, provide an unambiguous identification of both the physical devices and their virtual representations, as well as an initial data fusion of the raw sensor-generated information, which is later, in the *IoT Service* module, enriched by a broader semantic context of extracted virtual services.

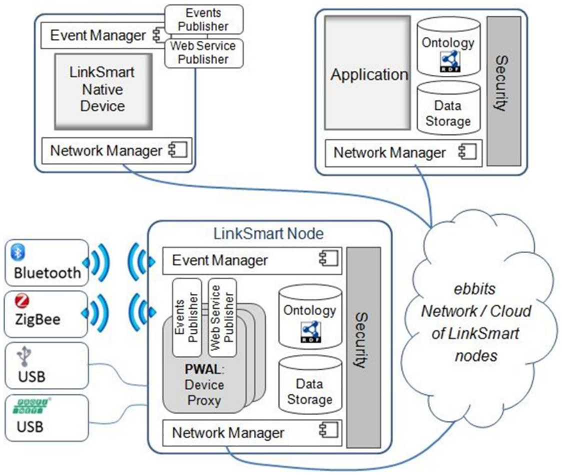


*Fig. 3. The network of LinkSmart nodes*

Modular architecture of LinkSmart enables to construct the network of devices quite flexibly, according to the needs and requirements of Smart Office or any other particular type of IoT application. The LinkSmart network consists of nodes communicating with each other in a peer-to-peer (P2P) mode, as it is presented in Figure 3. The network connection for each of peer nodes, together with routing of events and messages, is handled by the *Network Manager*, which employs the OSGi framework (http://www.osgi.org) to implement the JXTA and SOAP protocols. For inclusion of a physical device into the network it is required to run a minimal set of LinkSmart modules, namely PWAL with a proper driver, the *Network Manager* enabling the P2P network connection, and *Event Manager* that transforms data generated by the device to the LinkSmart-compatible events and RESTful web services. Communication and data transfer inside the LinkSmart P2P network is then performed upon the subscriber-publisher model [22], which enables an autonomous behavior for each of the networked nodes.

## 4 ARCHITECTURE OF THE IOT SMART OFFICE SOLUTION

Flexible and adaptable architecture of IoT network based on the LinkSmart middleware is of specific advantage especially for the Smart Office concept, which has been investigated and applied within the FP7 project ELLIOT. As it was outlined above in the introductory section, the Smart Office solution should support continuous monitoring and evaluation of user interactions with the installed infrastructure of sensors and devices controlling the office room settings [3], [6]. To support these features, ELLIOT adopts the approach proposed in [23], which enables to evaluate the monitored user experience by means of its knowledge, social, and business characteristics. Corresponding K-S-B dimensions are graphically represented by the diagram depicted in Figure 4.

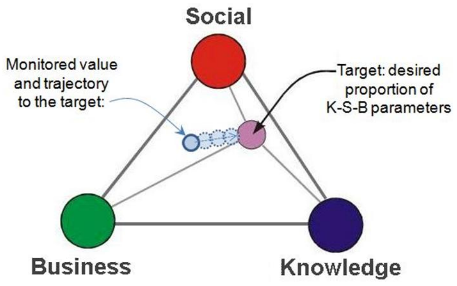

Social
Monitored value and trajectory to the target:
Target: desired proportion of K-S-B parameters
Business
Knowledge

*Fig. 4. The K-S-B experience model [23]*

The software platform calculating values of K-S-B characteristics from the monitored key performance indicators (KPIs) has been developed as an outcome of the ELLIOT project [24] and was provided for further testing by six pilot applications – IoT use cases in areas of e-Health, green services and energy efficiency, NFC technologies, and logistics. Due to the adaptable web service interfacing and modular IoT-oriented architecture, the LinkSmart middleware was used as the main implementation technology for the ELLIOT platform.

In line with the adopted methodology of co-creation and user driven development [25], the Smart Office prototype, as one of the ELLIOT pilot applications, was designed and installed in an open Living Lab environment [10]. The focus of the Smart Office use case was given to the following features:

- Energy savings achieved by an optimized use of the office room equipment;
- Alignment of the office room environment to business processes such as planned work schedules, milestones in running projects, etc.;
- Comfort of individual users – office employees, by enabling personalized settings of light intensity, temperature, and other parameters of the office room environment.

The main objective of the pilot was to apply the user experience monitoring mechanism of ELLIOT for finding a proper balance between the work environment settings, individual comfort of the office employees, overall work efficiency, and reduction of energy consumed by the office during regular working hours.

The Smart Office prototype was installed in premises of RWE IT Slovakia, one of leading IT solutions providers in the Kosice region in Slovakia, the application partner for the pilot use case in the ELLIOT project. The test office room, which belongs to the administration department, is occupied by eight office employees that use standard HW/SW equipment such as personal PC workstations, shared scanner/printer, and various end-user applications accessing intranet databases and information resources. Light intensity in the office room is controlled by manual light switches; temperature is regulated by a central heating system (in winter) or by a central regulation panel of the air-conditioning (in the summer season).

Based on the specified objectives of the pilot use case and the equipment available in the test office room, a set of sensors has been proposed for monitoring the energy consumption (Plugwise Circle devices), indoor/outdoor temperature, humidity, and light intensity (Plugwise Sense and Switch components, Arduino Uno board with TMP 102 and BH1750FVI sensors), and for sensing the occupancy of a workplace (Passive Infra Red motion sensors SE-10, RFID Card Reader of MFRC500 and related components, together with RFID Tags/Cards). The air condition controlling device, built upon the RTD-NET Daikin interface, was used for monitoring and remote control of the air conditioning installed in the office room.

Distribution of specified sensors and devices in the testing office room, as it was deployed for testing the Smart Office prototype in the ELLIOT use case, is presented in Figure 5. Data generated by sensors are regularly gathered and processed in the *Control Unit* located in the office room. The Control Unit is built on an embedded computer running the PWAL component of the LinkSmart middleware (see above, in section 3), which hosts services interfacing the installed devices, both sensors and actuators. Data generated by sensors are represented as LinkSmart-compatible events, which are then forwarded via Internet socket-based communication channels into the inner LinkSmart modules of the *Smart Office application server*. The communication may operate in an opposite way – the application server can generate commands that are forwarded via the Control Unit to actuators; e.g., to switch on/off the air conditioning or lights.

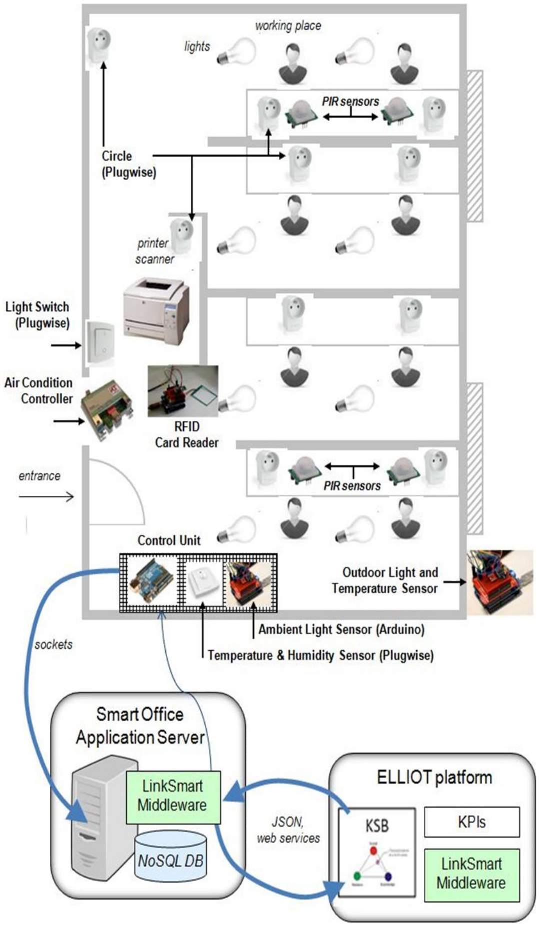


*Fig. 5. The Smart Office network – distribution of sensors and connection to the ELLIOT platform*

The Smart Office application server, built on customized LinkSmart modules and deployed as a LinkSmart node in P2P network, provides a means for business logic of virtualization, orchestration, and context-aware fusion of data/events obtained from sensors. It also provides a local storage of received data, semantic reasoning and execution of business rules. Data stored in the system database and ontology on the application server are fused to calculate the values of pre-defined KPIs of user experience, then they are wrapped by JSON format and transmitted as RESTful web services to the remote ELLIOT platform, which evaluates the corresponding K-S-B characteristics and sends the resulting distance from the target (cf. Figure 4) as a feedback for adjusting the Smart Office system accordingly. Since the ELLIOT platform is also deployed as a LinkSmart node, the communication is enabled in both directions by dedicated web service interfaces.

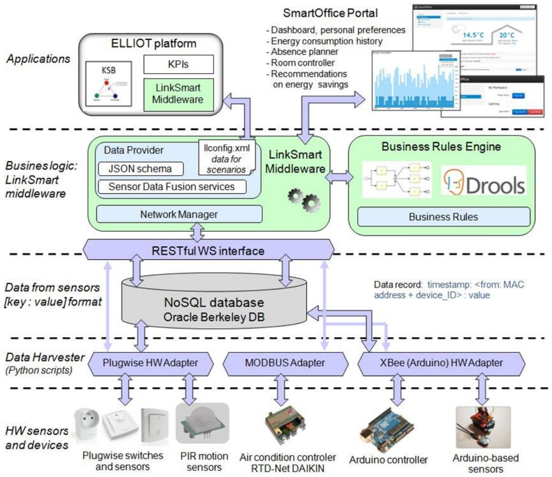
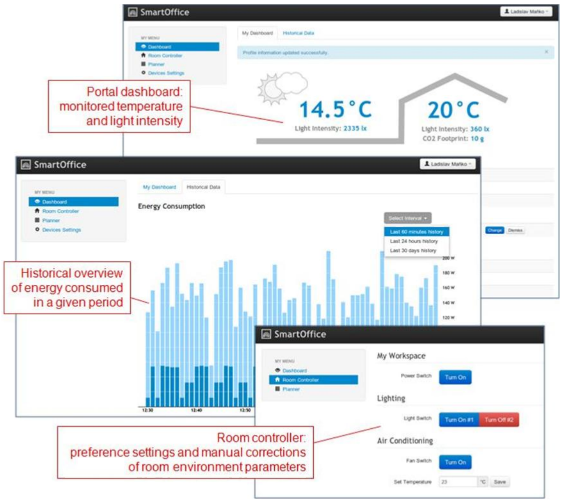

Applications
ELLIOT platform
KSB
KPIs
LinkSmart Middleware
SmartOffice Portal
- Dashboard, personal preferences
- Energy consumption history
- Absence planner
- Room controller
- Recommendations on energy savings
14.5°C
20°C
Busines logic: LinkSmart middleware
Data Provider
JSON schema
Sensor Data Fusion services
llconfig.xml data for scenarios
LinkSmart Middleware
Network Manager
Business Rules Engine
Drools
Business Rules
RESTful WS interface
Data from sensors [key : value] format
NoSQL database Oracle Berkeley DB
Data record: timestamp: <from: MAC address + device_ID> : value
Data Harvester (Python scripts)
Plugwise HW Adapter
MODBUS Adapter
XBee (Arduino) HW Adapter
HW sensors and devices
Plugwise switches and sensors
PIR motion sensors
Air condition controler RTD-Net DAIKIN
Arduino controller
Arduino-based sensors

*Fig. 6. Architecture of the Smart Office modules*

Inner structure of modules employed in the Smart Office system is presented in Figure 6. The solution consists of several system layers loosely coupled by means of web service interfaces. The core of the Smart Office application is built on the LinkSmart IoT middleware, which was customized as follows:

- Hardware devices were wrapped by PWAL that implements the corresponding drivers and proxy interfaces enabling a low level communication between hardware components and the internal business logic. In addition, an intermediate level of *Data Harvester*, implemented by a set of Python scripts, was included as a customized Event Manager.
- Data from sensors are pushed into a key-value store, where they are persisted and used for further processing – for example, to generate an overview of consumed energy for a given time period, etc. The persistent data storage was implemented as *NoSQL database* and is complementary to the existing RDF ontologies, where all the connected devices, generated types of events, and maintained information structures are represented as well.
- The IoT Business Process Management module was extended on business process modeling capabilities and the corresponding *Business Rules Engine*, built on the open source JBoss Drools platform (http://www.jboss.org/drools/).
- The *Data Provider* component was developed as a customized PSO, which provides high-level data fusion and aggregation facilities such as, for example, a calculation of CO2 footprint for a given workplace. Data Provider also calculates respective KPI values and provides the web service interfacing towards the remote ELLIOT platform.


Portal dashboard: monitored temperature and light intensity

Historical overview of energy consumed in a given period

Room controller: preference settings and manual corrections of room environment parameters

Profile information updated successfully.

Light Intensity: 2335 lx

Light Intensity: 360 lx
CO2 Footprint: 10 g

Set Temperature 23 °C

*Fig. 7. Views of the Smart Office portal*

In addition to that, the *Smart Office portal* was developed as a specialized web-based user interface (see Figure 7) that enables users to interact with the installed sensors and devices, adjust system settings, specify personally preferred values of the office environment parameters such as temperature, light intensity, or humidity, introduce a schedule of work, business trips, or holidays for employees, access the CO2 footprint overview, view a history of energy consumed on a workplace or in the office room as whole, etc. To support the optimization of energy consumption, the portal automatically generates personalized recommendations on possible energy savings, based on a comparison of user preferences and actually monitored values of temperature or light intensity. Responses on the generated recommendations, together with other information on consumed energy, CO2 footprint, workplace occupancy, etc., are the most important data for computing the specified KPIs and consequently for the user experience evaluation. From the perspective of employed technology, the portal was implemented on the Ruby framework (http://rubyonrails.org), while the interfacing and data transfer from/to the LinkSmart middleware is accomplished by RESTful web services.

The data flow schema of the Smart Office system is presented in Figure 8. The main source of input data in the Smart Office system architecture is the *Data Source* layer, which covers both low-level data generated by sensors and usage log data generated by users interacting with the web interface of the Smart Office portal.

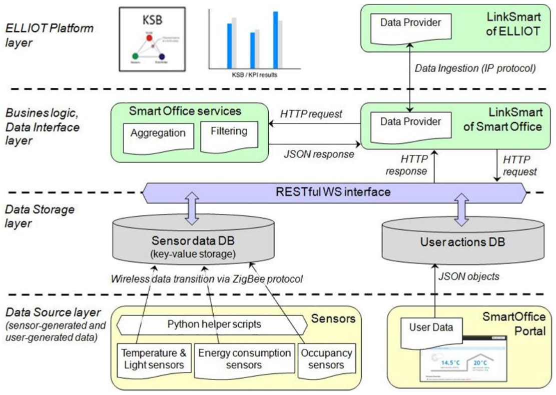


*Fig. 8. Data flow in the Smart Office system*

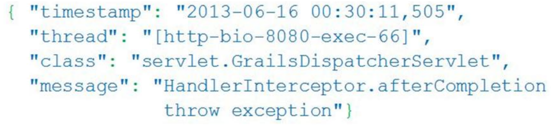
Data from sensors are transmitted to the Data Harvester mechanism (see above, in Figure 6), implemented as an extension of the PWAL component of LinkSmart by means of dedicated Python helper scripts. The data are forwarded via ZigBee protocol and are pushed to the *Data Storage* layer. Here the received sensor-generated data are persistently stored in the *Sensor data* database, a part of the Data Storage layer that is composed of key-value data records. In such a data record, the key is a concatenation of a timestamp (unix timestamp format), device MAC address and device ID. The last element of the data entry is the value itself. Entries in the data log are separated by ":". For example, the data record:

```
1342787246.8:001fd05f5453:0:23.3
```

contains fields as follows:

- `1342787246.8` – timestamp (i.e., seconds from January 1st, 1970);
- `001fd05f5453` – MAC address of a peer node (e.g., an XBee module) with a device connected to the IoT LinkSmart network (in this case, the MAC address corresponds to the temperature sensor);
- `0` – ID of particular device connected to the peer node (this information is needed since there can be more devices connected to a single node);
- `23.3` – actual value on the sensor (i.e., in this case, the value on the temperature sensor is 22.3 Centigrade).

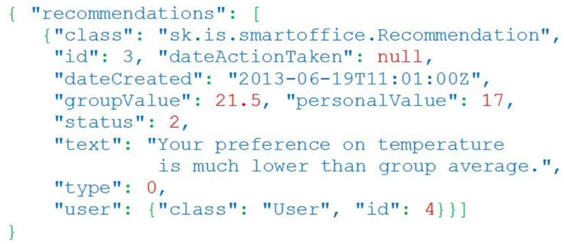
The second part of the Data Storage layer is the *User actions* database where all information about performed manual control actions, login sessions, energy saving recommendations and user profile data is stored. The user-generated data, regularly collected from the Smart Office portal logs and user entries, are persistently stored in a multi-field format of key-value pairs. For example, log record on exception occurrence in a given web application thread, caused by a particular Java class, is represented in Figure 9.



```
{ "timestamp": "2013-06-16 00:30:11,505",
  "thread": "[http-bio-8080-exec-66]",
  "class": "servlet.GrailsDispatcherServlet",
  "message": "HandlerInterceptor.afterCompletion
              throw exception"}
```

*Fig. 9. Log record sample*

The data in the Data Storage layer, both sensor- and user-generated, are accessible by means of implemented web services via built-in RESTful interface. Since the data in the Data Storage layer are stored in raw form without any pre-processing, the *Data Interface* layer, which includes a complete business logic of the LinkSmart middleware, serves as the main aggregation and filtering tool for obtaining all the data required to compute particular KPIs. The corresponding KPI extraction services, which are available in the Data Provider component, retrieve properly filtered data and generate the JSON output as it is presented in Figure 10.



```
{ "recommendations": [
   {"class": "sk.is.smartoffice.Recommendation",
    "id": 3, "dateActionTaken": null,
    "dateCreated": "2013-06-19T11:01:00Z",
    "groupValue": 21.5, "personalValue": 17,
    "status": 2,
    "text": "Your preference on temperature
             is much lower than group average.",
    "type": 0,
    "user": {"class": "User", "id": 4}}]
}
```

*Fig. 10. Example of a KPI extraction service*

Once the data for KPIs are properly extracted, fused and aggregated by native LinkSmart services, a set of KPI computation services is provided for remote calls. The JSON document generated by extraction services is taken as an input for the computation of actual KPI values. Both the KPI extraction and computation services are available as RESTful Smart Office interfaces and serve as the main access point for the ELLIOT platform to perform the data ingestion process. The separation of these two types of services provides the overall architecture with higher level of modularity, since it provides the ELLIOT platform with functionality to integrate services capable of computing custom KPIs (e.g. by means of clustering user profiles, etc.). Based on a comparison of the known KPI values with inner semantic structures of K-S-B parameters [26], the ELLIOT platform internally computes the resulting position of user experience and the distance from the specified target (see above in Figure 4). These results, sent back to the Smart Office system, can be used for adjusting the application settings towards a desired quality of the user experience.

## 5 CONCLUSION

The presented prototype of the Smart Office solution, integrating features of modular IoT middleware and continuous evaluation of user experience, was developed as a pilot use case of the ELLIOT project. To demonstrate the capabilities of user experience monitoring and evaluation in a co-creative Living Lab environment, the prototype was tested on three scenarios, namely:

1. *Exploration of a suitable occupancy sensing device*, where the effectiveness and user acceptance of motion sensors, RFID cards, and energy consumption monitoring were compared;
2. *Process-based adaptation of the office work environment*, which compared the scheduled and individual ad-hoc planning of work tasks and work absences (holidays, meetings, business trips, etc.);
3. *Simulated competition on energy savings*, where a serious gaming approach, enabled by the Smart Office portal, was adopted to facilitate energy savings.

The scenario-based tests were focused on the suitability of the provided technology and on the overall acceptance of the solution by the involved users. The IoT network architecture of the system prototype was evaluated as suitable and advantageous for the Smart Office system, namely in terms of scalability and adaptability of installed devices. The overall acceptance, calculated by an average usage frequency of provided functions, reached 80% [27]. The obtained results are promising; however, further improvements could be achieved by testing of the Smart Office prototype on a broader scale of users and/or in office rooms of various types.

## 6 ACKNOWLEDGEMENT

The work presented in the paper was supported by the ELLIOT project (http://www.elliot-project.eu) and the EBBITS project (http://www.ebbits-project.eu), co-founded by the European Commission within the contracts No. 287560 and No. 257852, respectively.

## 7 REFERENCES

[1] C. Le Gal, J. Martin, and G. Durand, “SmartOffice: An Intelligent and Interactive Environment”, Managing Interactions in Smart Environments, Springer London, 2000, pp. 104-113.

[2] C. Ramos, G. Marreiros, R. Santos, and C.F. Freitas, “Smart Offices and Intelligent Decision Rooms”, Handbook of Ambient Intelligence and Smart Environments, Springer US, 2010, pp. 851-880.

[3] O.P. Vidal, “Smart Office: Wireless Sensor Network for Energy Monitoring and User Profiling”, Electrical Engineering Automatic Control, Master's Degree Project, Kungliga Tekniska Högskolan, Stockholm, 2010.

[4] K. Ashton, “That ‘Internet of Things’ Thing”, RFID Journal, June 22, 2009.

[5] E. Aarts, R. Harwig, and M. Schuurmans, “Ambient Intelligence, The Invisible Future”, McGraw Hill, 2001, pp. 235-250.

[6] O. Akribopoulos et al, “Making P-Space Smart: Integrating IoT Technologies in a Multi-office Environment”, Mobile Wireless Middleware, Operating Systems, and Applications, LNCS, Springer, Vol. 65, 2013, pp. 31-44.

[7] SourceForge.net, “Link Smart Middleware”, http://sourceforge.net/projects/linksmart/, accessed: October 4th, 2013.

[8] V. Vajda, K. Furdik, J. Glova, and T. Sabol, “The EBBITS Project: An Interoperability platform for a Real-world populated Internet of Things domain”, Proceedings of the Znalosti (Knowledge) 2011 conference, VSB-Technical

University of Ostrava, Czech Republic, 2011, pp. 317-320.

[9] P. Brizzi, D. Conzon, F. Pramudianto, M. Paralic, M. Jacobsen, C. Pastrone, R. Tomasi, and M. A. Spirito, “The ebbits platform: leveraging on the Internet of Things to support meat traceability”, Proceedings of the EFITA-WCCA-CIGR Conference ‘Sustainable Agriculture through ICT Innovation’, Torino, Italy, June 23-27, 2013.

[10] H. Schaffers, S. Budweg, R. Ruland, and K. Kristensen, “Collaborative Environments to Support Professional Communities: A Living Lab Approach”, Leveraging Knowledge for Innovation in Collaborative Networks, IFIP Advances in Information and Communication Technology, Springer Berlin Heidelberg, Volume 307, 2009, pp. 635-642.

[11] N. Bui et al, “D1.1 SOTA report on existing integration frameworks/architectures for WSN, RFID and other emerging IoT related Technologies v0.9”, Deliverable of the FP7 project IoT-A, Contract No. 257521, 2011.

[12] L. Atzori, A. Iera, and G. Morabito, “The Internet of Things: A survey”, Computer Networks: The International Journal of Computer and Telecommunications Networking, Elsevier, Volume 54, Issue 15, October 2010, pp. 2787–2805.

[13] A.P. Castellani, N. Bui, P. Casari, M. Rossi, Z. Shelby, and M Zorzi, “Architecture and Protocols for the Internet of Things: A Case Study”, Pervasive Computing, IEEE, Vol. 10, Issue 2, 2011, pp. 19-27.

[14] P. Barnaghi et al, “D3.6 Final SENSEI Architecture Framework”, Deliverable of the FP7 project SENSEI, Contract No. 215923, 2011.

[15] P. Kostelnik, M. Sarnovsky, and K. Furdik, “The Semantic Middleware for Networked Embedded Systems Applied in the Internet of Things and Services Domain”, Scalable Computing: Practice and Experience (SCPE), Scientific International Journal for Parallel and Distributed Computing, Volume 12, No. 3, September 2011, pp. 307-315.

[16] M. Eisenhauer, P. Rosengren, and P. Antolin, “Hydra: A development platform for integrating wireless devices and sensors into ambient intelligence systems”, The Internet of Things, Springer US, 2010, pp. 367-373.

[17] C. Magerkurth et al, “D1.4 Converged architectural reference model for the IoT v2.0”, Deliverable of the FP7 project IoT-A, Contract No. 257521, 2012.

[18] Z. Shelby and C. Bormann, “6LoWPAN: The wireless embedded Internet”, Wiley Series on Communications Networking & Distributed Systems, Wiley, 2009.

[19] Z. Shelby et al, “Constrained Application Protocol (CoAP) - draft-ietf-core-coap-18”, https://datatracker.ietf.org/doc/draft-ietf-core-coap/, accessed: October 8th, 2013.

[20] S. Meyer, A. Ruppen, and C. Magerkurth, “Internet of Things-Aware Process Modeling: Integrating IoT Devices as Business Process Resources”, LNCS, Vol. 7908, Springer, 2013, pp. 84-98.

[21] M. Knechtel et al, “D4.M36 Semantic Mechanisms and Models”, Deliverable of the FP7 project EBBITS, Contract No. 257852, 2013.

[22] K. Furdik, G. Lukac, “Events Processing and Device Interoperability in a Smart Office IoT Application”, Proceedings of the 23rd Central European Conference on Information and Intelligent Systems (CECIIS 2012), University of Zagreb, Croatia, 2012, pp. 387-394.

[23] A. Bifulco and R. Santoro, “A Conceptual Framework for Professional Virtual Communities”, Collaborative Networks and Their Breeding Environments, IFIP, Springer US, Vol. 186, 2005, pp. 417-424.

[24] F. Cattaneo et al, “D2.4 Final Version of the ELLIOT Platform”, Deliverable of the FP7 project ELLIOT, Contract No. 258666, 2012.

[25] C. Durugbo, J. Riedel, and K. Pawar, “Towards a Unified Model of Co-creation”, Proc. of 17th International Conference on Concurrent Enterprising, Aachen, Germany, 2011.

[26] M. Pallot and K. Pawar, “A holistic model of user experience for living lab experiential design”, Proc. of 18th International Conference on Engineering, Technology and Innovation (ICE), Munich, Germany, 2012.

[27] K. Furdik et al, “D7.1.2.2 Evaluation of Elliot solutions in Final Release”, Deliverable of the FP7 project ELLIOT, Contract No. 258666, 2013.