Semantic Policy-Based Data Management for Energy Efficient Smart Buildings

Vikash Kumar, Anna Fensel, Goran Lazendic, and Ulrich Lehner

The Telecommunications Research Center Vienna (FTW),
Donau-City-Straße 1, A-1220 Vienna, Austria
{kumar,fensel,lazendic,lehner}@ftw.at

Abstract. We describe how the semantics can be applied to the smart buildings, with the goal of making them more energy efficient. Having designed and implemented a semantically enabled smart building system, we discuss and evaluate the typical data management challenges connected with the implementation, extension and (re-)use of such system when employing them in the real buildings. The results demonstrate a clear benefit from semantic technologies for integration, efficient rule application and data processing and reuse purposes, as well as for alignment with external data such as tariffs, weather data, statistical data, data from other similar smart home systems. We outline the typical data management operations needed in the real life smart building system deployment, and discuss their implementation aspects.

1 Introduction

Energy efficiency is a topic of acute and growing importance, as achieving 20% savings of energy consumption by 2020 through energy efficiency is one of the key measures to keep CO2 emissions under control[1]. By 2050, EU's goal is to cut greenhouse gas emissions by 80–95% and two thirds of the energy in the EU should come from renewable sources[2]. Also widely known are regulations of EU-wide introduction of smart meters - to be implemented within the next few years - it is estimated that the "*world will have 250 million smart meters by 2015, representing a $3.9 billion market*"[3]. The related markets of energy efficiency, building automation, smart homes are accordingly rapidly growing, particularly, close to exponential in the renewable energies sector[4]. All these factors lead to the appearance of vast amounts of data, followed by the need to efficiently manage it, and enable services addressing the drastically changing energy efficiency data economy.

[1] EU climate and energy "20-20-20" package, URL: http://ec.europa.eu/clima/policies/brief/eu/package_en.htm
[2] "Energy Roadmap 2050", European Union 2012, URL: http://ec.europa.eu/energy/publications/doc/2012_energy_roadmap_2050_en.pdf.
[3] Pike Research, URL: http://seekingalpha.com/article/170629-250m-smart-meters-3-9b-market-by-2015-pike-research-says
[4] "Clean Energy Trends 2012", March 2012, URL: http://www.cleanedge.com/reports/clean-energy-trends-2012

P. Herrero et al. (Eds.): OTM 2012 Workshops, LNCS 7567, pp. 272–281, 2012.

On the technical side, as the integration of the Internet and physical reality is increasing, analytical and sensor-based systems and services tend to rely on knowledge representations with shared, well-known structures (ontologies) for improved interoperability. This data can serve as a basis for various added value services. Currently this data is often collected, processed and analyzed manually and individually by the companies, whereas automation, optimization and sharing would not only be possible but also generating an enormous value chain in such a large and important market as energy efficiency.

In this work, we present a use case for energy efficiency, supported by a semantic system combining particularly home automation techniques, smart metering and sensor data, deployed in two real buildings (a school and a factory) allowing advanced monitoring and control of energy consumption. We describe the developed system and the deployment set up, as well as the available data. Apart from contribution to establishment of the best practice techniques in implementation of such systems, we identify and study in detail the required automation processes for data, semantics and rules usage, and draw evaluation conclusions on the applicability and success of large-scale data reuse practices.

This paper is structured as follows. In Section 2, we describe the problem statement and motivation for the paper and approach. In Section 3, we discuss our energy management hardware and software design, detailing in particular the use of semantic technologies and data management. In Section 4, we present the evaluation set up and results. Section 5 discusses the related work. Finally, Section 6 concludes and summarizes our paper.

2 Problem Statement and Motivation

As the vision of Internet of Things[5] moves towards mainstream reality, several new challenges needs to be addressed for a better adaptation and interconnectivity of such systems. A common understanding of data emanating from all such systems to enable them to talk, interact and respond to events in each others' environments are essential to achieve this goal. Increased data volumes would also require smart planning of system architectures that are scalable to accommodate such data as well as perform efficient processing to derive meaningful information from them.

A typical case in point is the proliferation of commercial smart home and building systems [3][4][5][6][9] where the common goal is to use latest technologies for improving the quality of living and, in most cases, also to enable energy savings. Even with the success of these systems in standalone buildings, their shortcomings in the broader vision of smart cities cannot be ignored where there is an increasing need for reduction in efforts for their setup and standardization of underlying technologies right from data representation to intra as well as inter building communications. Besides this, expansions in such installations are themselves constantly vulnerable to changes like appliance manufacturers, types of energy used in different buildings, etc. notwithstanding the changes required in the user interfaces [10].

In this paper, we argue that providing a semantic layering on top of a data prosuming ecosystem consisting of sensors and receivers provides manifold added value and

[5] Internet of Things: `http://en.wikipedia.org/wiki/Internet_of_Things`

benefits such as: (a) enabling easy exchange and integration of data among interconnected smart building systems, (b) providing "intelligence" to the system by way of reasoning almost with little or no overhead costs, (c) minimizing adaptation costs for installations in new environments, and (d) ease of scalability in terms of addition of new and existing setups in the ecosystem.

We validate our claims in the setup of a building automation system containing sensors to gather data and actuators and central controllers to control various devices. Superimposing a semantic layer on top of it enabled the introduction of policy-based intelligence to the system and heterogeneous useful apps and services for the users. The whole setup was installed at a school in Kirchdorf, Austria, and then later replicated on a factory floor in Chernogolovka, Russia. The results demonstrated a very high efficiency in the amount of time required to replicate the data layer and adapt the policies between the setups despite the core components of the system being vastly different.

3 System Description

While the general architecture of our implemented smart energy efficient building deployment is shown in Figure 1, in the following subsections we describe the hardware, semantic software and user interface parts of our system.

3.1 Hardware – Physical Layer Settings

The installation was carried out on two separate locations and different types of buildings. The installation in Kirchdorf in Austria consists of sensors, meters for measuring consumption of electrical energy in three classrooms, plugs for measuring electrical consumption of individual appliances, and a power management service (PMS), for monitoring and controlling the state of all computers installed in these three rooms. The installation at factory in Chernogolovka in Russia includes only sensor devices for measuring room temperatures and temperatures of incoming and outgoing airflow of heating system in factory. In the complete version of our system, the following set up has been applied:

Sensors. Sensor devices for measuring temperature, humidity and light were battery powered autonomous systems pushing their data over wireless LAN internet connection to a cloud service exposing collected data via REST interface.

Meters. Power consumption meters were installed in the housing of already existing distribution cabinets at school. Collected data was exposed over a serial interface on metering device and transferred to persistent storage through a wireless bridge to be finally exposed via REST interface for external use.

Smart Plugs. Smart plugs from Plugwise[6] were used to measure the consumption of individual devices at school in Kirchdorf, consisting of one beverage machine, one coffee machine and a wall socket where pupils in school connect their laptops. The data of smart plugs is exposed over Web Service interface for external use.

---

[6] Plugwise: `http://www.plugwise.com`

Power Management Service (PMS). The PMS is a service for monitoring the state of computers installed in the three classrooms as well as for shutting down and restarting them remotely.

Hardware Setup Overview. The whole trial setup consists of sensors, meters, plugs and PMS where all delivered data is accessed either by REST or Web Service interface processed by connector software where it is prepared for semantic storage in an OWLIM semantic repository.

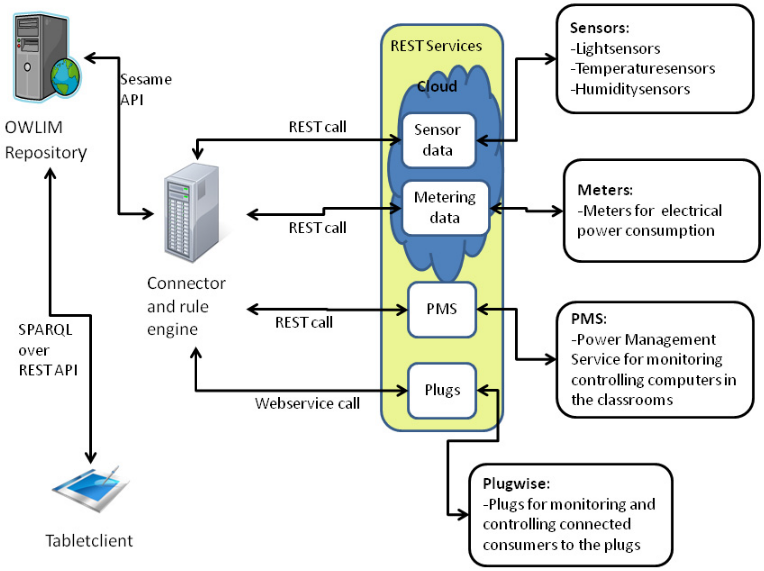

Fig. 1. SESAME-S School Setup at Kirchdorf, Austria

3.2 Data and Software Layer – Semantics

An OWLIM-Lite [7] based semantic repository is central to the data layer of our smart building architecture. OWLIM is a Sesame SAIL (Storage and Inference Layer) where Sesame[7] itself is a framework for analyzing and querying RDF data stored in a triple store [1]. This serves as the triple store for all the data received from various sensors and also provides RDFS-optimized reasoning capabilities necessary for our use cases. Rules in the native PIE format are supported but have limited expressive capabilities.

The connector software itself reads data from school via REST and Web Services on the one side and prepares and stores that data in OWLIM repository by using native Java Sesame 2.0 API on the other side. Besides this functionality and due to limitations of OWLIM for implementing custom rules, the connector software is doing custom reasoning over data in OWLIM store via SPARQL Construct queries

[7] SESAME: `http://www.openrdf.org/`

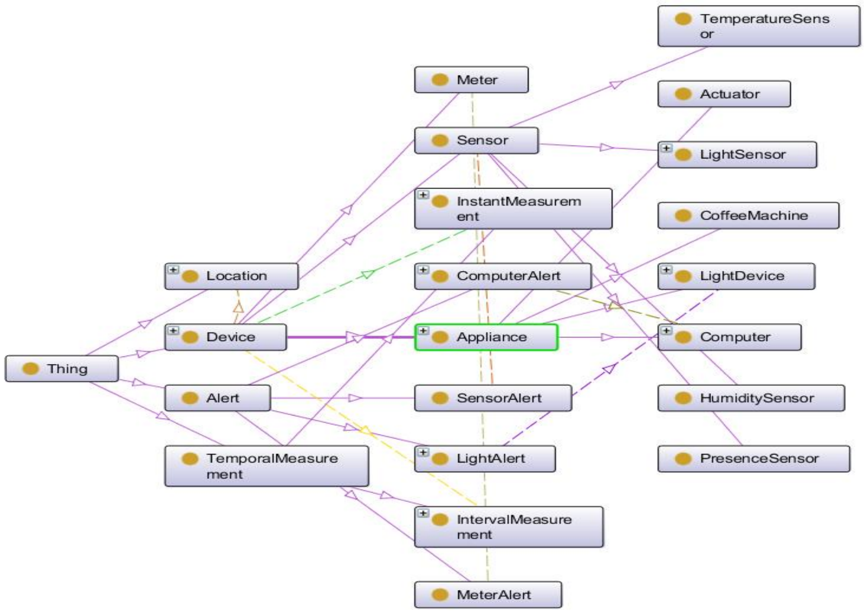

Fig. 2. Core of SmartBuilding Ontology

implementing rules for our use cases. Client side devices like tablets and phones query the repository utilizing REST interface offered by OWLIM store.

SmartBuilding Ontology. The ontology defining our data structure is described below, and its class hierarchy is shown in Figure 2. Disjoint classes represent various aspects of a smart building environment: Devices, Locations, Time bound measurements (Temporal Measurement) and Alerts.

Several datatype properties are used to describe device features like control type (manual or automatic), function (heating, cooking, etc.), power source (battery, electricity, self generating, etc.), state (ON/OFF/Idle), specifications, manufacturer, name, etc. Instances of the subclass "InstantMeasurement" shows instantaneous readings of devices in the "Device" class by datatype property "hasMeasurement" at the instant represented by another datatype property "atInstant". "IntervalMeasurement", another subclass of TemporalMeasurement is similar to "InstantMeasurement" with the only difference being the time for measurements being represented by two data properties (instead of one), "intervalStarts" representing the start of measurement interval and "intervalEnds" representing end of the measurement interval.

Data properties "atInstant" and "hasAlertState" describe the time as well as the state of the alerts in "Alert" class respectively. "hasAlertMessage" describes the message accompanying a particular alert instance. Object properties like "alertFromSensor", "alertFromComputer", etc. connect the SensorAlert (& ComputerAlert respectively) as domain to the Sensor (or Computer) generating that alert as range. The "locatedAt" object property connects the "Device" class as domain with "Location" class as range. "locationHas" is the inverse of the "locatedAt" property. The "hasInstantMeasurement" object property has "Device" class as domain and "InstantMeasurement" class as range. The "hasIntervalMeasurement" class similarly connects "Device" and "IntervalMeasurement" classes respectively.

The Policy Based Approach towards Intelligent Decision Making. A policy based approach towards providing personalized and flexible services to users lies at the core of our design that enables easy adaptation of the framework to different environments. With the data arriving from heterogeneous sources structured and updated in the central repository almost in real time, we utilize the feature to create complex rules and policies to administer this setup. Several services and end user interfaces were created in which rule based modeling of user side constraints were utilized in the monitoring and administration of the system. In the first phase of implementation, most of our tasks involved monitoring of the system and only one use case takes care of administrative jobs.

The OWLIM uses TRREE (Triple Reasoning and Rule Entailment Engine) which is a forward-chaining R-Entailment [2] for providing in-memory reasoning and query evaluation functionalities. Existing rule-sets can be extended with custom rules expressed in the native Pie format. However, several limitations in this format (like not supporting mathematical operators) greatly restrict the potential advantages one could harness with custom rules.

We therefore resorted to using SPARQL[8] Construct queries as a workaround for this handicap. The end user services in our setup do a constant querying of the status of changes in repository at regular intervals. These intervals are flexible and in all our installations, fixed at 15 minutes. Each time an application sends a query to the repository, a set of SPARQL Construct queries are executed by the connector software. Each of these Construct queries contains the logic and constraints relating to the user/system policies. If the constraints are matched with existing environmental conditions represented in the repository, new triples pertaining to respective alerts are generated and added to the triple store. In our setup, all these triples are added to the class "Alert", connecting them to the device and time at which they were generated along with a message. The Application queries the "Alert" class each time to check if a new alert was generated in the system since the last check.

3.3 End User Interfaces and Client Side Data Acquisition

The end user interfaces were developed on top of the Android platform and consisted of a tablet and a smartphone interface allowing comparisons in the usage scenarios. The features include real time feedback on power consumption visualized as line chart, real time data on appliance level (ON/OFF/Idle), historical comparison of past consumption, remote shutdown mechanism for computers and a notification system that reminds the user of unused computers (Figure 3). We used an ACER ICONIA Tab A501 with a 10.1" WXGA display in our evaluations.

The mobile phone application version has a user interface design consistent to the tablet application, but is adapted to the special constraints of the device's smaller

[8] SPARQL: `http://www.w3.org/TR/rdf-sparql-query/`

Fig. 3. Dashboard of the Tablet Software

form factor (like splitting the features into different screens). Because the phone version is not supposed to run all the time, we enhanced the notification system with the standard Android notifications. This behavior is same as receiving a text message reminding users of any idle computers.

4 Evaluation of Data Management

The evaluations of the installation and its services were carried out on our setup at a school in Kirchdorf, Austria. The parts of evaluations showing the flexibility of the system to adapt and scale to a new setup are explained in this paper.

4.1 Evaluation Setup

As described in Section 3, the school setup consists of 3 rooms attached with light, humidity and heat sensors as well as smart meters receiving consumption data from various devices. The semantic data management and services described in Section 3 were tested for their ability to adapt themselves to a new smart building setup. The total volume of data collected from this installation over several months is amounting to ca. 3 million triples. We also observed the effort needed to formally induct the data from a new setup classified according to our existing, generic in its nature, ontology.

4.2 Results

While the initial creation of the connector software linking various data sources to central repository took a typical amount of required man hours, once it is implemented, linking additional sources in general was very easy. Below we describe the adaptability of this framework on the basis of various criteria as tested in real installations:

Adding New Buildings to the System. Dramatically reducing the efforts required in adding new buildings to this framework according to us is the most outstanding achievement of this work. Most of the existing energy monitoring systems use Excel sheets for maintaining their data in a structured format and integrating such data back into new systems or extending it for changes in existing systems is very cumbersome [10]. On the other hand, to induct a new building installation to the framework of our SESAME system, the raw data coming from various sources in this building along with respective timestamp and source information need to be provided to our data layer preferably via a REST/Web service interface.

Since our *SmartBuilding* ontology is very generic in nature, any new location names, devices and corresponding alert types can be readily added to extend the ontology as well as in the Connector software. Such a modification to include data from Russian factory setup in our main repository took not more than half a day by a single person. We also argue that for the adaptation of ontology, a semantic expert is not necessarily needed as including a new setup requires only an extension of the ontology and not the redefining or modification of underlying logic.

Adding New Devices. Addition of a new device to the system can be done in the following steps: (a) installation of the device (b) inserting an instance of it in the appropriate class (Appliance/Meter/Sensor) of the ontology and connect it to other entities as needed (c) provide a REST/Web Service interface of the data coming from this device. For step (c), if there is no way to monitor individual consumption (e.g. toaster, fans, etc.) then it can also be connected via a Plugwise smart plug for which we have installed interfaces to the system. Most of the smart meters and sensors in the market already provide a standard REST/Web service access to their data and thus our system doesn't require any specific non standard access to data making it very easy to add a new device to the system.

In our setup, we tested this with the addition of sensor data in a factory setup in Chernogolovka in Russia where three combined sensor sets are involved for measuring inside room temperature, incoming airflow temperature, outgoing airflow temperature and one Boolean value indicating whether the heater is turned on or off. As the setup here uses the same hardware sensor devices as the setup in school at Kirchdorf/Austria no adaptation of connector software was needed, except for REST access interface.

Aligning Old and New Data. The process of aligning new data to the system besides hardware installation requires adaption of connector software in relation to new REST/Web Service interface offered by the new device added to the system. This means introducing new interface classes in Java to represent the new data structure.

Another aspect in aligning new data refers to end user interface and presentation of new data to the user. The principle in our system uses SPARQL Construct queries for custom rules and resulting triples representing alerts generated by our rules. The system requires only an addition of new SPARQL query for retrieving new alerts based on new use case and change of labeling to conform to this use case and to be meaningful for the user. As the presentation layer is totally decoupled from underlying data layer, custom individual user interfaces for every use case are possible and require no deep understanding of the whole system.

Adapting the Policies. Since the policies for our school setup was created mostly for monitoring the system based on daily routine of school, we could adapt most of these generic looking policies for the Russian factory setup as well. An example was sending out alerts if light was left on after school hours (for the school setup) or after factory working hours (for the factory setup).

Although the effort needed for adapting the policies manually was around half a day for each policy, significant knowledge of semantic web technologies was required to achieve this. We therefore decided to work on an automatic policy adaptation tool that would take care of such policy adaptation tasks automatically reducing the dependence on core semantic expertise [8].

Since all our application services use structured data from the repository and include intelligent reasoning over them, we argue that just by simplifying the data acquisition from a new setup to our framework makes this framework highly scalable along with the value addition of using semantic intelligence.

5 Related Work

Home automation has been defined in [6] as "The introduction of technology within the home to enhance the quality of life of its occupants, through the provision of different services such as telehealth, multimedia entertainment and energy conservation." There has been extensive work in making home automation systems by using different technologies and utilizing innovative techniques [3][4][5]. The aim of authors in [3] is to achieve a more efficient remote control and monitoring of networked enabled devices in a house. They investigate the use of ZigBee towards creating a flexible automation system. The system has however not been tested with real installations and hence the scalability cannot be ascertained. Another example of such an implementation using knowledge representation techniques for smart university application is shown in [6]. This approach is based on semantic web service middleware enriched with capabilities like dynamic composition adapted specifically for university buildings and similar educational facilities. Such a system is therefore far from generic to be implemented across the board in all kinds of buildings. After an initial study into such systems, we believe that our approach provides a unique mix of the qualities (mentioned in earlier sections) that can be practically used in mainstream smart building systems.

6 Conclusions

We have presented a semantically enabled real life smart buildings deployment, and have described and analyzed the data processing techniques. We have shown the feasibility of such deployment in use, and the benefits the semantic technologies are bringing to such development. While many data operations are shown to be made very efficiently, it would be of interest to see the aspects of even larger scale deployments of that character, e.g. as a part of a smart city. The main contributions of this work are as follows:

- Real life use case with real data and users, its design, implementation, deployment, as an innovative development for the semantic web community;
- Illustrated clearly the cases of added value of the semantic technologies in the real-life settings;

- Analyzed the typical data management implementations for this smart building use case, and showed their implementation feasibility with the current state of art.

The collected and semantically represented data has a large potential for being combined, extended and reused for numerous scenarios and parties, such as grid operators seeking the balance on their smart grids, utilities, looking after optimal energy trading prices - based on real, and not synthetic user profiles, and municipalities, looking for more information about the citizens. Mechanisms to adequately access and commercialize such data within services (in particular, complying with the rigid security and privacy requirements typical for smart home use cases) are certainly among the next most relevant research questions.

Acknowledgments. This work is supported by the FFG COIN funding line, within the SESAME-S project (http://sesame-s.ftw.at). FTW is supported by the Austrian government and the City of Vienna within the competence center program COMET. The authors thank the SESAME-S project team for valuable contributions.

References

1. Rusher, J.: Semantic Web Advanced Development for Europe (SWAD-Europe). Workshop on Semantic Web Storage and Retrieval - Position Papers
2. ter Horst, H.J.: Combining RDF and Part of OWL with Rules: Semantics, Decidability, Complexity. In: Gil, Y., Motta, E., Benjamins, V.R., Musen, M.A. (eds.) ISWC 2005. LNCS, vol. 3729, pp. 668–684. Springer, Heidelberg (2005)
3. Gill, K., Yang, S.-H., Yao, F., Lu, X.: A zigbee-based home automation system. IEEE Transactions on Consumer Electronics 55(2), 422–430 (2009)
4. Byun, J., Park, S.: Development of a self-adapting intelligent system for building energy saving and context-aware smart services. IEEE Transactions on Consumer Electronics (2011)
5. Bromley, K., Perry, M., Webb, G.: Trends in Smart Home Systems, Connectivity and Services (2003), http://www.nextwave.org.uk
6. Stavropoulos, T.G., Tsioliaridou, A., Koutitas, G., Vrakas, D., Vlahavas, I.: System Architecture for a Smart University Building. In: Diamantaras, K., Duch, W., Iliadis, L.S. (eds.) ICANN 2010, Part III. LNCS, vol. 6354, pp. 477–482. Springer, Heidelberg (2010)
7. Kiryakov, A., Ognyanov, D., Manov, D.: OWLIM – a Pragmatic Semantic Repository for OWL. In: Proc. of International Workshop on Scalable Semantic Web Knowledge Base Systems (SSWS 2005), WISE 2005, New York City, USA, November 20 (2005)
8. Kumar, V.: A Semantic Policy Sharing and Adaptation Infrastructure for Pervasive Communities. In: Proc. of ACM World Wide Web (WWW-PhD Symposium) 2012, Lyon, France, pp. 155–160 (2012)
9. Tomic, S., Fensel, A., Schwanzer, M., Kojic-Veljovic, M., Stefanovic, M.: Semantics for Energy Efficiency in Smart Home Environments. In: Sugumaran, V., Gulla, J.A. (eds.) Applied Semantic Technologies: Using Semantics in Intelligent Information Processing. Taylor and Francis (2011)
10. Capehart, B. L., Middelkoop, T.: Handbook of Web Based Energy Information and Control Systems, 565 pages. Fairmont Press (July 26, 2011)