ONTOLOGY BASED TELE-HEALTH SMART HOME CARE SYSTEM: ONTOSMART TO MONITOR ELDERLY

L. Nachabe[1,3], M. Girod-Genet[1], B. ElHassan[2] and J. Khawaja[2]

[1]Department of Réseaux et Services Multimédia Mobiles,
Telecom SudParis University, Evry, France
marc.girod_genet@telecom-sudparis.eu
[2]Department of Electricity and Electronics, Faculty of Engineering, Branch 1,
Lebanese University, Tripoli, Lebanon.
bachar_elhassan@ul.edu.lb
[3]American University of Culture & Education, Faculty of Science,
Beirut, Lebanon.
Linanachabe@auce.edu.lb

ABSTRACT

The population ageing is a demographical phenomenon that will intensify in the upcoming decades, leading to an increased number of older persons that live independently. These elderly prefer to stay at home rather than going to special health care association. Thus, new tele-health smart home care systems (TSHCS) are needed in order to provide health services for older persons and to remotely monitor them. These systems help to keep patients safe and to inform their relatives and the medical staff about their status. Although various types of TSHCS already exist, they are environment dependent and scenario specific. Therefore, the aim of this paper is to propose sensors and scenarios independent flexible context aware and distributed TSHCS based on standardized e-Health ontologies and multi-agent architecture.

KEYWORDS

Smart-Home, tele-Health, Multi-Agent, distributed systems, WSN, Ontology, fall, Indoor Localization

1. INTRODUCTION

The percentage of people classified as elderly is increasing in all major regions of the world. The definition of elderly is not standard and may differ from one country to another. While for most developed countries the age of 65 years is adopted as a definition of elderly, this number may decreases to 60 or even 55 in third world countries (Africa for example). The UN has not adopted a standard criterion, but generally uses 60+ years to refer to the older population [1]. According to the World Health Organization (WHO), an increase from 600 million old people in 2000 to 2 billion by 2050 is expected. By 2050, 22% of the world's population will be over 60. In Europe, the percentage of elderly will be over 30%. These numbers are highlighted by the life expectancy at birth parameter. In addition, according to the WHO, approximately, 10% of the population experience some form of disability. Already 21% of people above the age of 50 have severe vision, hearing and/or mobility problems [2].

Natarajan Meghanathan et al. (Eds) : ACSIT, SIPM, CMIT, CoNeCo - 2016
pp. 43–59, 2016. © CS & IT-CSCP 2016 DOI : 10.5121/csit.2016.60805

In general, most elderly people prefer to grow old at their own home, which is known as "aging-in place" [3]. In fact, at this age, the increasing risks of falls and strokes could threaten their live. Thus, to grow at home, vital signals should be remotely monitored to keep the elderly safe. Thanks to the evaluation of wireless senor networks, and the explosion of new embedded systems, especially wearable devices dedicated for biomedical usage, the tele-health home care systems (THCS) are realizable. According to the Market Reports Hub press release, the global tele-health market will reach $6.5 billion in 2020 at an annual growth rate of 24.2% [4]. Medical devices for self- health monitoring are expected to form 80% of wireless devices in 2016 [5].

However, these THCs are facing paramount concerns [6]. On one hand, these concerns are related to the tremendous types of sensor/actuators, communication protocols, scenarios, vendors, and service providers. In other terms, these systems are very heterogeneous, vendors dependent, non-scalable (integration of new sensors/actuators and services need technical support), and hardly upgradable (if new technology appears in the market, the overall system should be re-implemented). On the other hand, these systems should provide context-aware services capable of creating and adjusting the offered services based on the occupant requirements, preferences and situation. Moreover, social implication of tele-health system is still at the heart of the THCs barriers where the privacy and security are mandatory. Finally, these systems should offer reliability, real time monitoring, friendly user interfaces with minimum expert intervention and reasonable resource consumption (in terms of energy, processing and bandwidth). Therefore, "smart" THCS should be introduced (TSHCS). In this context, we define "smart" as the capability of the system to recognize new components (automatic nodes discovery), discover new services, and infer new information from basic collected raw data from sensors.

In this paper, we present a new TSHCS designed using ontologies to describe generated data and services. The usage of ontologies paves the way to an interoperable solution and tackles the problem of heterogeneity within the same system, and between different systems. Moreover, multi-agent architecture is introduced in our proposed TSHCS system in order to distribute the processing among different components. This leads to a flexible and more efficient power consumption solution that the existing TSHCS. All these aforementioned ideas will be clarified in the next sections.

The paper is organized as follow. In Section II, the reasons behind using ontologies and multi-Agent architectures are clarified. Overview of the retained ontologies is also presented in this section. Section III describes the general architecture of our proposed TSHCS system, called OntoSmart system. Section IV details our OntoSmart based applications called OntoSmartHome and dedicated for the patient and his/her relatives. Moreover, tests and analysis are given in this section. Finally, Section V concludes the paper.

2. BACKGROUND

2.1 Reason behind using ontologies

As many THCSs are proposed [7], different terminologies are used to describe the same properties or object. Moreover, various ranges of sensors/actuators (ambient sensors, electrical devices, biomedical sensors, etc.) from wide range of vendors supporting different wireless technologies (Bluetooth LE, LoRa, ZigBee, etc.) are used [7]. This variety caused the problem of heterogeneity management and interoperability in THCSs. In general, these systems are monolithically deployed. The integration of a new device or a new service needs the reconfiguration of the overall system, and some time the deployment of a new system [8]. For that reason, new techniques that deal with the data meaning and enhance the automatic node and service discovery are needed. These techniques are named Semantic techniques where an entity presents an aspect of the real domain described by metadata (data about data). Thus, by using semantic techniques within THCS, the raw data can be annotated by semantic data respecting a predefined semantic model to unify its description [9]. In that way, the problem of interoperability between different systems can be resolved. Ontologies are the most powerful semantic techniques. Using the formal definition, ontologies are “tools for specifying the semantics of terminology system in a well-defined and unambiguous manner” [10]. W3C recommended the use of OWL language to describe ontologies [11] . In fact, the most powerful point behind using OWL is the capability of reasoning in order to infer more significant data.

Many researches have introduced the usage of ontologies in TSHCSs. Nguyen et al. [12] proposed to formalize smart home context-aware system using ontologies. The ontology was installed on the web server. The ontology describes the habitant, the environment parameters and the activity done by the habitant. But, to effectively resolve the problem of interoperability, advanced description about the devices (especially the sensors/actuators) is needed [13]. For example, the used communication protocols, the source of energy, the memory components, etc. Moreover, a clear mechanism for automatic node/service discovery should be defined. Chen et al. [14] proposed a three layers architecture (Semantic Layer, Intelligent Layer and Application Layer) for TSHCSs. The Semantic layer is responsible for the annotation of raw data based on an ontology that describes: the home entities, the living space, the location, the person and his/her behavior. Moreover, this layer adds relations between different classes. The Intelligent Service Layer offers the capability of interoperability and high level automation. The application Layer is left to the developer. Although Chen et al. proposed a general architecture; they however neither give details about the implementation of this architecture nor evaluate it using a real test-bed. Latfi et al. [9] also introduced ontologies to model TSHCSs in order to assist patient in accordance to his/her behavior. They conceived seven ontologies that mainly describe the patient’s history, equipment, software application, and behavior. These proposed ontologies still need to be deployed in real scenarios. Ye et al. [15] focused on modeling TSCHSs using three main classes: Concept, Context and Activity. The novelty of their resulting ontology is that it shows how the use of semantic rule can enhance the inference of new significant knowledge. In summary, all previously mentioned attempts stressed on the importance of using normalized ontologies in order to mask the problem of interoperability in TSCHS, reuse and share generated data, and create a context aware environment. However, none of these attempts give a clear solution for automatic discovery and integration of new components.

Additional researches stressed on the importance of introducing service oriented architecture for smart home. Authors in [16] proposed service oriented ontologies to describe service properties and location in smart homes relying on SUMO ontology. A service oriented rule based framework has been proposed by Leong et al. [17]. However, the service discovery is restricted to the server side (single point of failure) and relies on SOAP (XML structure which lack to the semantic annotation). Solve-D [18] is a framework for disabled users in smart home environment consisting of three main service ontologies: Generic service ontology to describe standard services for appliances, personalized service ontology and service context ontology where complex rules can be defined for advanced decision making. However, Solve-D focused on home appliances rather than fully sensor/actuator description. Meshkora et al. [19] presented ontology based COMANCHE project in order to organize and discover information needed to manage software in home environment. This project enables complex relation modeling between processes and includes rich description of devices (interfaces, location, manufacturer, processing capabilities, etc.). However, it lacks the description of the quality of service, and the data format, accuracy, precision, etc.

It is rigorous that none of these attempts combine the fully description of the sensors/actuators, data, process, and services. While reliability and service consistency are primordial in such systems, the QoS and computational resources as well as remaining battery level are needed for service modeling. Moreover, tele-health systems should integrate the monitoring data in IoT. Thus grounding methods for service accessing will facilitate the service discovery.

Therefore, the contribution of this work is OntoSmart system that relies on both MyOntoSens Ontology [20] (summarized in Figure1) and MyOntoService Ontology (depicted in Figure 3) to model TSCHSs' raw data and information. These ontologies are provided with associated management enablers, within our OntoSmart system, and will enable the handling of all the aforementioned failings of existing TSHCS architectures.

2.2 OntoSmart System Ontologies

TSCHS can be seen as a WSN (Wireless Sensor Network) encompassing different nodes (sensors, actuators, equipment, servers and gateways). Each WSN, in other term the home, can have one or more owner (the patient to be monitored) and contact persons (medical staff or relatives). Each node is used for certain process (temperature, humidity, etc.) and each actuator performs certain action (turn ON/OFF, shuttle Up/Down, etc.). That is why we used MyOntoSens ontology [20] to describe the components of our proposed system since this ontology already takes into account all the aforementioned TSHCS mandatory attributes. MyOntoSens has been designed as a modular ontology and is divided into three parts: MyOntoSensWSN, MyOntoSensNode and MyOntoSensProcess. Figure 1 summarizes the main classes of MyOntoSens ontology, as used in our proposed OntoSmart system, while Figure 2 depicts the main classes and attributes of MyOntoSensProcess part.

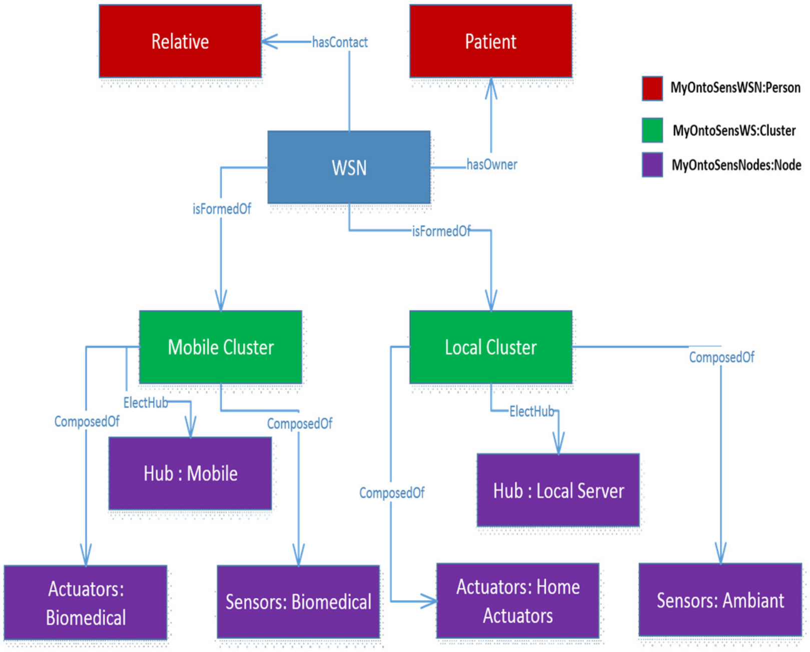


Figure 1-MyOntoSens Ontology as used in OntoSmart System

Moreover, the TSHCS system offers certain services to the patient (monitoring, notification, reminder, etc.). We are modeling these services by introducing original service ontology, called MyOntoService ontology and depicted in Figure 3.

Our new upper service ontology adopts the idea of OWL-S ontology [21]. It is composed of three main sub-ontologies (as depicted in Figure 3):

- MyOntoServiceProfile that describes the service (QoS, QoI, service constraints and service type) and enables their auto-discovery,

- MyOntoServiceProcess that describes how the service is realized based on the Input, Output Precondition, and Effect (IOPE) mechanism,

- And finally MyOntoServiceGrounding that describes how the service can be invoked (or accessed in IoT e.g. HTTP, CoAP, etc.)

To ensure automatic senor/service discovery, the ontologies are described by SWRL [19] rules and the Pellet [22] reasoner is used to infer explicit data (Section 3).

2.3 Reason behind using multi-agent architecture within our TSHCS system

The main aim of our proposed system is modularity, scalability, flexibility and distributed computing. Thus, this architecture should logically be divided in standalone modules offering certain services, capable of interacting with other modules and that can be instantiated when required. This is exactly what a software agent can do. An agent is a piece of software that can run autonomy to perform certain behavior. The composition of many agents is called Multiple Agents System (MAS) [23]. The agents interact by passing predefined messages. The MAS architecture describes how different agents are interconnecting. MAS are widely used in complex system where abstract definition of certain task is required. In our proposed TSCHS system basic functionalities are offered by agents capable of invoking other agents. This is in particular the case for our data collection and information management/sharing/retrieval functionalities. For example, we can cite the data collector agent, query agent, notification agent, writer agent, etc.

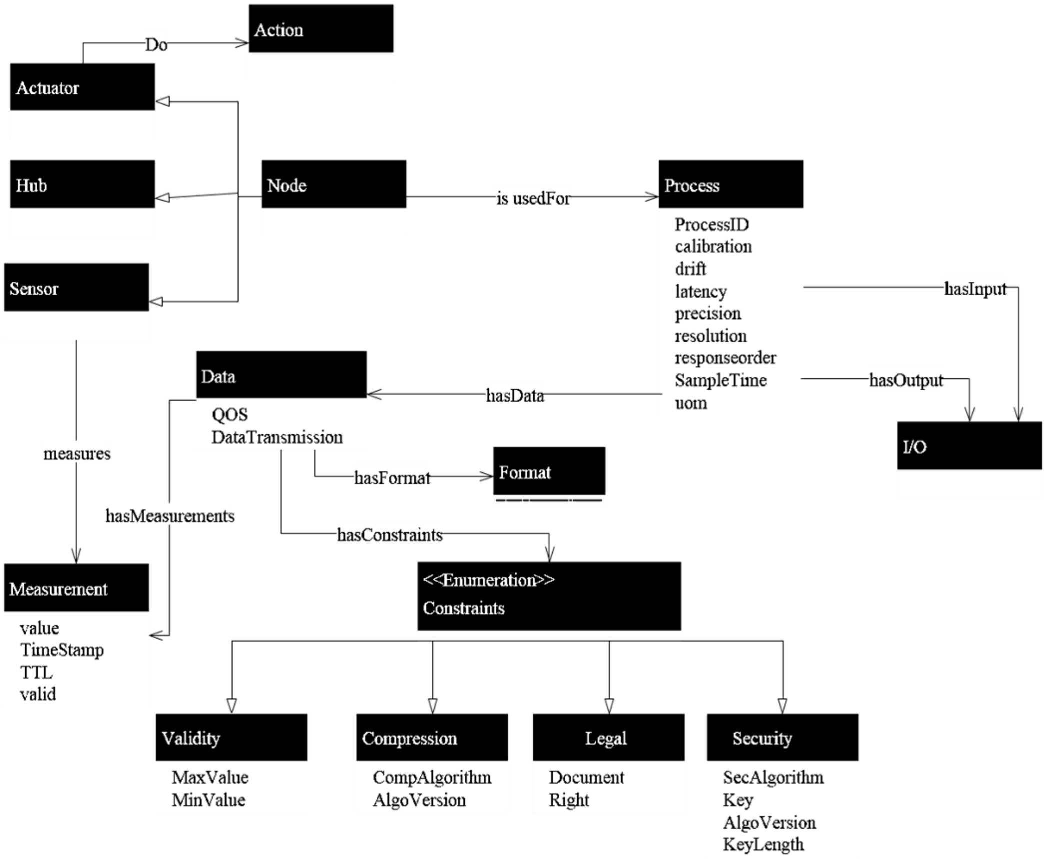


Figure 2-MyOntoSensProcess Classes Hierarchy

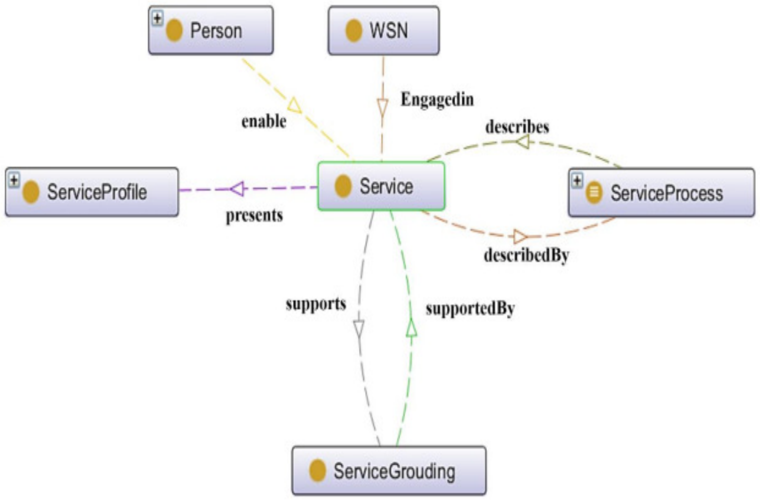


Figure 3-MyOntoService Ontology

Moreover, using expressive ontologies without defining a general mechanism for data annotation is pointless [24]. That is why we defined three main agents in our overall TSHCS system:

- Scanners: for each communication protocol (Bluetooth LE, WIFI, ZigBee) there is a data scanner that read raw data and call the adequate wrapper for semantic annotation. Moreover, for each grounding protocol (CoAP, HTTP, etc.) there is a service scanner that reads the request from external user and calls the adequate service.
- Writers: To send data from the system to external users, the adequate writer encapsulates the sent data based on the requested protocol.
- Wrappers: To add semantic description of the raw data based on MyOntoSens and MyOntoService ontologies.

Detailed description of how to use these components is depicted in section 3.

3. ONTOSMART GENERAL SYSTEM ARCHITECTURE

Our proposed OntoSmart system comprises a WSN formed of two main clusters: the Wireless Body Area Network (WBAN) encompassing the sensors attached to the body of the patient for vital signals monitoring, and the local cluster formed of a local server and sensors/actuators related to ambient parameters and home appliances. It is worthy to note that the sensors used in our proposed scenario are only for testing purposes. These sensors could be extended to any other nodes required in TSHCSs without affecting the overall system architecture. Figure 4 depicts our proposed OntoSmart system architecture.

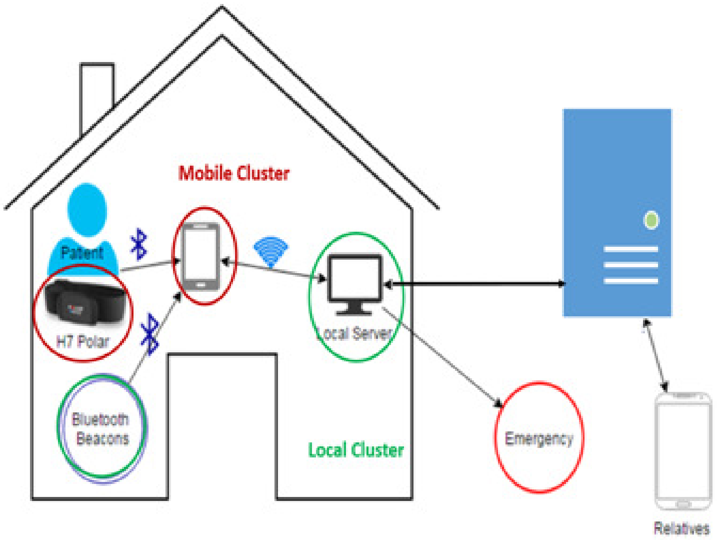


Figure 4- OntoSmart General Architecture

Falls, heart attack and stroke are among the leading causes of hospitalization for the elderly and patients. The chances of surviving a fall, heart attack or stroke are much greater if the senior gets help within an hour [25] For that reasons we mainly focused on monitoring the heart rate of the patient, deducing his/her posture (sitting, moving, falling, etc.), and his/her location (in which room he/she is). Moreover ambient processes are captured such as humidity, light intensity and temperature.

The details of each cluster are given in the next sections.

3.1 WBAN Cluster

The main aim of the WBAN Cluster is to collect patient's vital signals in order to send it to the local server. Here comes the role of the Data Collector Agent (DCA), also called the Cluster Hub. In our scenario, the WBAN Cluster is formed of the heart rate sensor, acceleration sensor and the smart phone of the user which is playing the role of the Cluster Hub. To capture the heart rate, the H7Polar sensor [19] is used. This senor sends the heart rate via its Bluetooth LE interface.

Therefore, a Bluetooth LE Scanner (BLES) agent is installed on the patient's smart phone in order to:

- identify any Bluetooth LE device,
- and recognize the offered service due to the use of UUID defined in standard GATT Profiles [26]

This scanner gives the first step toward an automatic sensor discovery.

To determine the posture of the patient, the smart phone built-in three acceleration sensor is used. We assume that the smart phone is placed in a vertical position on the hip of the patient. Therefore, the patient is not moving if the acceleration is equal to the gravitational force. We thus have in this case:

- If $|a_y| > 9$ m/s^2 then the patient is standing.

- If $|a_x| > 9$ m/s^2 then the patient is lying.
- If $6 < |a_x| < 8$ m/s^2 and $2 < |a_y| < 5$ and $2 < |a_z| < 5$ then the patient is sitting.

After identifying the node's characteristics, these latter should be sent to the local server for semantic annotation and further treatments. JSON [27] message format was adopted due to its lightness and simplicity. Therefore, a JSON Writer (JS) agent is installed on the smart phone in order to encapsulate data and send it to the local server.

If other types of sensors are used (like ZigBee), the only think that should be done is to install the convenient scanner (like ZigBee Scanner). Likewise, if data from the smartphone is encapsulated using a message format other than the JSON messages, the suitable writer agent should be compelled. In fact, the use of these data agents and writers enable the automatic discovery of any added nodes and enhance the flexibility and scalability of the system where any nodes can be added without the need to rebuild the overall system. What is really required is to ensure that the communication protocol used by the device has the suitable data scanner agent.

3.2 WSN Cluster

The WSN Cluster is formed of the sensors dedicated for indoor localization, the ambient parameters sensors, and the local server (DCA or cluster Hub). In our case, we use the RadBeacon Bluetooth LE sensors to identify the location of the patient. Moreover, the ambient parameters (light intensity, humidity and temperature) are retrieved from the built-in sensors in the patient's smartphone in order to be sent to the remote server using JSON messages. We installed, on the local server, the following agents:

- BLES agent: to identify Bluetooth LE sensors and retrieve the required information in order to invoke the adequate wrapper.
- JSON Scanner (JS) agent: to retrieve data sent from the smartphone and invoke the adequate wrapper.
- Node Wrapper (NW) agent: to add semantic annotation about the sensors/hubs based on MyOntoSensNode ontology.
- Process Wrapper (PW) agent: To add semantic annotation about the process used by the sensors based on MyOntoSensProcess.

In that way, the local server will contain all the semantic description of the home devices. Because more than one patient can use the same WSN sensors (e.g. indoor localization sensors can be used to locate more than one patient in the same home), the object property *"measuredFor"* has been added between the sensors measurements (Measurement class of MyOntoSensProcess ontology) and the patient (Patient Class of MyOntoSensWSN ontology). To this point, all nodes are automatically discovered. Regarding the service discover, we added SWRL [21] rules to infer basic services offered by the system. All processes are considered as service (e.g. heart rate, temperature, etc.). This inference is insured by using Rule 1. Composed services are deduced from additional rules. For example, using the heart rate sensor implies that the maximum heart rate service can be used for notification purposes (Rule 2).

(Rule 1) *Process(?P) -> Service(?P) .*

(Rule 2) Process (HR) -> Service (MaxHR), hasEffect (MaxHR, Notification).

Moreover, the object property *"enable"* between patient individual and service individual is used for allowing the patient to enable/disable any service offered by the system. Only enabled services are retrieved when invoking the query dedicated for service discovery.

3.3 Remote Semantic Storage and Management Server

The main aim of the remote semantic server is to save the fully semantic description of all the TSHCSs. It plays the role of semantic registry. While intelligence treatment for each home is located on its local server, the intelligent cooperation between different TSHCSs, and advanced data analysis for statistical purposes can be conducted on this remote server. In our scenario, the remote server encompasses the following agents:

- SPARQL Agent: to query semantic information saved on the remote/local server in order to retrieve new semantic information.
- CoAP Agent: to publish the data in order to be used by external users for remote monitoring (e.g. patient's relatives and medical staff). The CoAP [28] is a Rest-full protocol used for constraint networks. It uses the UDP protocol as transport layer and the reliability mechanism is driven by the application layer. We used the CoAP Confirmable messages to ensure the required reliability in our proposed TSCHS system. In that way, the Remote server will play the Role of CoAP monitoring server allowing remote users to access the data by using CoAP GET requests.
- Notification Agent: to notify external users about abnormal patient's attitude and dangerous measured values. In our case, we notify the user if the patient is falling or if his/her heart rate exceeds the normal value. It is worthy to note that the security mechanism is not within the scope of this study. That's why we based our test-bed authentication mechanisms on Android GCM due to its implementation simplicity, and by assuming that all our system users have a Gmail account. Let's precise here that for a commercial application, stronger authentication mechanisms will obviously have to be designed and implemented.

Fig. 5 summarizes all the aforementioned agents and building blocks location and interactions.

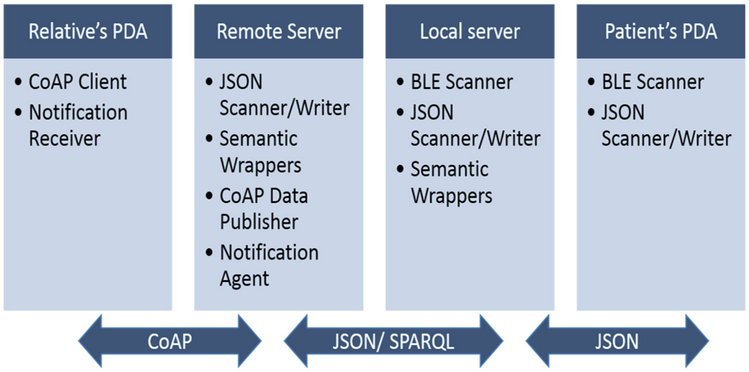


Figure 5-Implemented Agents and interactions as deployed in OntoSmart System

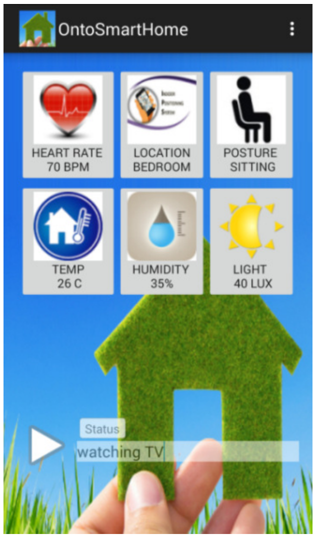
The remote server will invoke the WSN Wrapper in order to add the new WSN and Patient into its semantic registry. The PatientID (identifier generated by the remote server) is now returned to the smartphone and saved in its shared Preference (local database on the Smartphone) for being used when needed (for patient identification). The patient can now launch its OntoSmartHome application and log into the OntoSmart system using his/her Gmail account.

4. ONTOSMARTHOME APPLICATIONS

Two applications have been developed for OntoSmart system: the first application (detailed in Section A) is dedicated for the patient in order to initialize the system and display the monitoring values; the second application (detailed in Section 4.2) is dedicated for relatives for remote patient's monitoring. The same monitoring interface (depicted in Figure 6) is retained for the patient and relative applications.

4.1 Patient's OntoSmartHome Application

We have provided our OntoSmart TSHCS system with a patient-dedicated application and corresponding applicative agents (see Figure 5). This application, depicted in Figure 6 and Figure 7, is integrated within the Smartphone of the patient. As already mentioned in the previous sub-section (i.e. 3.3), the patient launches its OntoSmartHome application and logs into the OntoSmart system using his/her Gmail account. Once logged in, the patient can now add relatives (identified by their email). These relatives can thus monitor the patient by sending CoAP Get requests to the remote server and receives notification through a dedicated application (see section 3.5). The initialization phase, where the sensors are discovered, begins now. The Bluetooth LE sensors are recognized and displayed on the screen. For each discovered sensors, the user should precise the purpose of its usage. In particular, for Indoor localization, a setup phase is needed, where the user should walk during 20 seconds in the room in order to collect the minimum and maximum RSSI for each room [29]. When the setup phase is completed, the main page of the application is displayed showing the patient's heart rate and posture, as well as the ambient temperature, humidity and luminosity (see Figure 6). The patient can also add a status (similar to the idea used in social media) to inform the relatives about a special feeling or event, etc. The displayed value will be modified once a new value is measured. Note that a compromise between system accuracy and battery consumption is mandatory in such systems. That's why, after practical testing, we have chosen to acquire 50 samples/sec from the sensors. In addition, these modified values will be sent to the local server and the remote server via the JW in order to be updated on the CoAP server and registered in the remote ontology repository.



HEART RATE 70 BPM

LOCATION BEDROOM

POSTURE SITTING

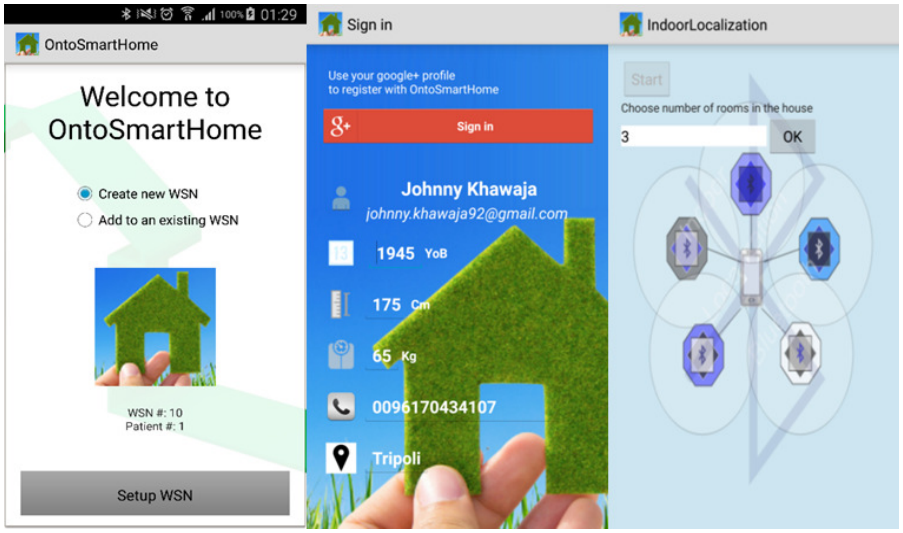
TEMP 26 C

HUMIDITY 35%

LIGHT 40 LUX

Status: watching TV

Figure 6- Monitoring Interface of OntoSmartHome Application



OntoSmartHome

Welcome to OntoSmartHome

● Create new WSN
○ Add to an existing WSN

WSN #: 10
Patient #: 1

Setup WSN

Sign in

Use your google+ profile to register with OntoSmartHome

Sign in

Johnny Khawaja
johnny.khawaja92@gmail.com

1945 YoB

175 Cm

65 Kg

0096170434107

Tripoli

IndoorLocalization

Start

Choose number of rooms in the house

3 OK

Figure 7-Patient's OntoSmartHome application

4.2 Relatives OntoSmartHome Application

The first time the application is installed, the relative should successfully log in using his/her Gmail account. Afterward, the relative is asked to enter the patient’s email. The patient and relative emails are sent via the JSON Writer (JW) to the remote server where a SPARQL query is invoked to verify the relative permissions. Figure 8 depicts the overall authentication process for the relative’s mobile application.

If the relative is able to monitor the patient, CoAP GET requests are sent every minute to the CoAP remote server in order to retrieve the values.

Our OntoSmart system provides the discovery of any used BLE sensor due to the use of BLE Scanner which discovers the node, the node wrapper which adds the node’s semantic description to the semantic registry, and the SPARQL agent that executes the node and service discovery queries. Moreover, publishing the monitoring patient’s data through a CoAP server permits the relatives to instantly monitor these data by issuing CoAP GET requests. Further analysis and advanced diagnostics can be conducted on the remote server by domain experts in order to help the patient. Moreover, due to the use of semantic registry on the remote server, new domain expert ontologies can be imported to infer severe situations related, for example, to heredity problems or environmental diseases.

4.3 Notifications

Two notification services are implemented and offered within our proposed OntoSmart system: Abnormal Heart Rate and fall detection. The abnormal heart rate will be calculated based on the age of the patient [30].

Regarding the fall detection, the algorithm based on the concept developed by Sposaro *et al.* [31] was adopted. A fall begins with a sudden and significant decrease in the acceleration amplitude “a”. The free fall period causes the acceleration’s amplitude to drop significantly below 9.8m/s2 threshold during a short time. This represents the period of time when the actual fall is taking place. After this “free-fall-Period”, the acceleration experiences a sudden spike as soon as the body hits the floor. The amplitude “a” crosses an upper threshold suggesting a fall. This threshold is determined to be 3g (around 27m/s2). Consequently, if the acceleration “a” crosses a lower and an upper threshold during certain observation time window, a fall is suspected. This window is often determined to be less than one second. To prevent false alarms, a timer for one minute is enabled allowing the patient to cancel the notification. Otherwise a notification will be sent to relatives mobile phones. In our implementation, the GCM notification server is used as detailed in Figure 9.

4.3 Testing & Results

In fact our testing process focused on two main principles: the requirements (as highlighted in Section I) of THSCs and the mobile application itself. For that purposes, the applications were tested on Samsung GALAXY S4 (Android 4.4.2 KITKAT-API 19 and Android 5.0.1-API 21 Lollipop after upgrade) and SONY XPERIA SL (Android 4.1.2 Jelly Bean- API 16). We first considered one patient per WSN. We installed the application on two different patients’ mobile. For each patient, one relative was added. We first insure that the ontology is well classified. Due to the use of incremental reasoner, the services were discovered sequentially. For example, once the H7 Polar is paired, the heart rate service was added on the server. Moreover, the user was able to enable/disable the services by his/her own without the intervention of experts. At certain stage, the Patient 1 decided to disable the indoor localization services. Immediately, on the relative smart phone, the location field was replaced by “Unknown”. This proves the ability of our proposed system to offer services based on the occupants needs and requirements.

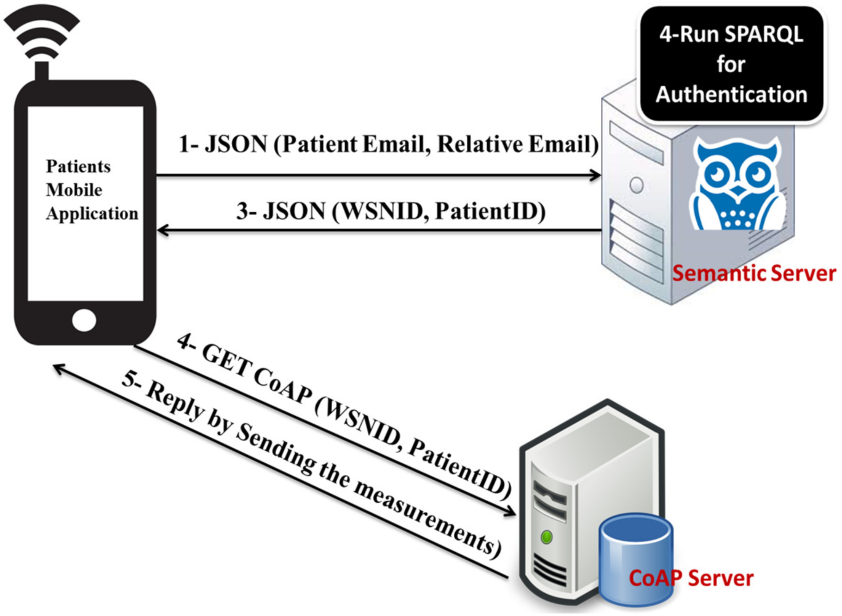

4-Run SPARQL for Authentication
Patients Mobile Application
1- JSON (Patient Email, Relative Email)
3- JSON (WSNID, PatientID)
Semantic Server
4- GET CoAP (WSNID, PatientID)
5- Reply by Sending the measurements)
CoAP Server

Figure 8-Authentication Process for Relative's mobile app.

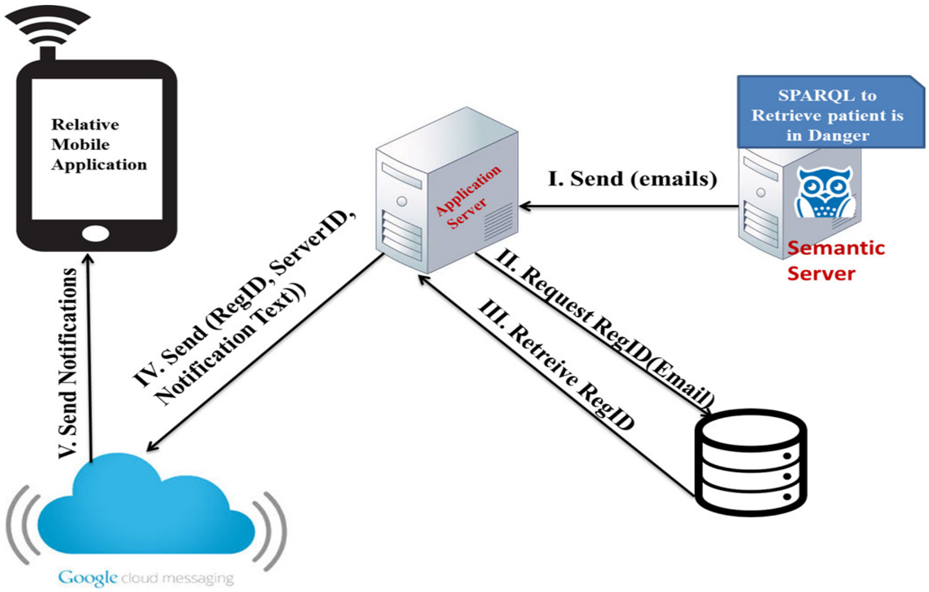


Figure 9-Notification process using GCM Server.

Then, we considered the scenario where two patients are sharing the same WSN, i.e. sharing the same Rad Beacons for the indoor localization. The distance between rooms was about 20 m and three Rad beacons were carried out and setup in three different zones at the patient's home. For testing purposes only, the remote server and the local server were on the same machine, but working on different port numbers. The ontology was well classified without any ambiguity. This insures the reusability of the proposed system by more than one occupant.

The privacy concerns were addressed by the use of email accounts for application installation and the authentication mechanism when a relative requests to monitor a patient. It is evident that this authentication mechanisms, as well as security concerns, could be enhanced by implementing advanced protocols. However, this enhancement does not affect the overall architecture, but only require the adoption of a new security/authentication agent invoked when needed.

After insuring the well functionality of the system, we focused on measuring the complexity of the system on the server. For both cases, the tests were conducted for about 8 hours. Table 1 depicts the time needed to load and classify the ontology, as well as the CPU and memory usage.

Table 1- System Evaluation on the Server

		Case 1	**Case 2**
Ontology	Load	24 sec	35 sec
	Classify	90 sec	75 sec
	Model Size	1403	1403
	Inferred Model size	1019	1000
CPU	Max. CPU	76%	60%
	Avg. CPU	35%	27%
	Memory	425 692 K	415 728 K

We can note that the ontology is classified within 2 minutes. The size of the inferred model reflects the importance of SWRL rules used in the proposed system. As few data is required to be sent for the mobile application to the semantic server, lowest bandwidth will be consumed. The potential of the semantic rules is inferring all needed data to discover the properties of a service. For example, all HR services will be inferred by just sending that the heart process is offered by H7 Polar sensor (see Rule 1 and Rule 2). We are actually working on studying the performance of the server on large scale (more than 10 patients registered in the application) to estimate the probability of crashing.

We passed than to evaluate the performance of the patient's mobile application. The time needed for node/service discovery was maximum 1 second. The patient's application was tested for about 8 hours; the battery consumption was maximum 12 % and minimum 5%, while the average CPU usage was 8%. Table 2 depicts the processing, network and battery usage on the patient smart phone in initiation mode (done only when the application is loaded for the first time) and the normal mode where the measurements are taken periodically.

This evaluation ensures that our proposed application is not dramatically affecting the resources on the smart phone. In fact, these consumptions can be decreased by the use of ambient sensors or biomedical accelerometer sensors instead of the built-in sensors. In that way, the data will be received directly on the server for semantic annotation and treatments. Aiming to determine the falling detection, we tried to test to detect a falling after standing, from bed or from a chair. These fallings were successfully detected. We can note that the posture and the falling status are not 100% accurate because we were using the Smartphone embedded sensors for the measurements (for economy purposes). These measurements can be enhanced by using dedicated accelerometer sensors capable of detecting more precisely the posture of the patient.

It is now the time to evaluate the relatives application. We first tested the authentication process. Only authenticated relatives were able to monitor the patients and receive notifications. We tested the network and battery consumption (depicted in Table 3) of the relatives for about 8 hours.

Table 2- Resources Consumption for the Patient Smart Phone

	Initialization Phase		**Main Activity**	
	Max	Average	Max	Average
CPU %	5%	2%	11%	8%
Network Usage	16 Kb	12 Kb	40 Kb	36 Kb
Battery Usage	7%	5%	12%	10%

On the relatives' side, the measured values and patient's status and alarms were displayed in real time. The CPU usage on the relative's application attempts as average 0.56 % and as maximum 1 %. These resources consumption are considered insignificant due the use of CoAP messages (1280 bytes). The values were displayed in real time and the notification when abnormal values are detected was received within 1 second.

Table 3- Resources Consumption on the Relative Mobile Application

Average CPU %	0.56%
Average Network Usage	35 Kb
Average Battery Usage	0.40%

5. CONCLUSION

In summary, although our OntoSmart system does not actually provide the best solution for fall detection (only because Smartphone embedded sensors were used instead of dedicated sensors), it offers the flexibility to enhance this detection, whenever needed, by just adding the necessary sensors. Our OntoSmart system is the first step toward a “Plug and Play” system where patients can easily add sensors and actuators. Service developers will only focus on ads-on services and applications without the need to deal with sensor integration and configuration. In addition to the passive intervention devices, active intervention devices for reminder systems or medical assistance can be added to OntoSmart System. More agents are under construction for example, remind patients to take his/her medication or alert him/her if a stove burner is left on. Surely, actuators can be added to the system. All these concerns do not need to re-invent the wheel, but just upgrade the system by adding adequate agents. More tests are being conducted to measure the efficiency of the remote semantic storage and management server when dealing with large amount of TSHCSs. Additional scanners and writers are also under development, especially ZigBee and HL7 scanners/writers. Finally, more security and privacy concerns are under investigations.

They were here when you were a child be there with them when they are going. THSCs will surely help elderly feeling gladder and confortable by staying at home and connected to their family.

REFERENCES

[1] “WHO | Definition of an older or elderly person.”

[2] “WHO | Ageing and life-course.”

[3] J. Cocco, “Smart Home Technology for the Elderly and the Need for Regulation,” pp. 85–108, 2011.

[4] V. Gruessner, “Global Telehealth Market Expected to Grow Through 2020 - mHealthIntelligence.” mHealth Intelligence, 2015.

[5] M. Memon, S. Wagner, C. Pedersen, F. Beevi, and F. Hansen, “Ambient Assisted Living Healthcare Frameworks, Platforms, Standards, and Quality Attributes,” Sensors, vol. 14, no. 3, pp. 4312–4341, Mar. 2014.

[6] W. K. Edwards and R. E. Grinter, “At Home with Ubiquitous Computing: Seven Challenges,” Proc. 3rd Int. Conf. Ubiquitous Comput., pp. 256–272, 2001.

[7] Q. Ni, A. García Hernando, and I. de la Cruz, The Elderly’s Independent Living in Smart Homes: A Characterization of Activities and Sensing Infrastructure Survey to Facilitate Services Development, vol. 15, no. 5. 2015.

[8] R. Preet, “A Case for Ending Monolithic Apps for Connected Devices,” HotOS, pp. 1–7, 2015.

[9] F. Latfi, B. Lefebvre, and C. Descheneaux, “Ontology-based management of the telehealth smart home, dedicated to elderly in loss of cognitive autonomy,” CEUR Workshop Proc., vol. 258, 2007.

[10] D. Allemang and J. Hendler, Semantic web for the working ontologist: effective modeling in RDFS and OWL. Elsevier, 2011.

[11] S. U. D. L. M. A. K. S. L. D. S. D. E. Deborah L. McGuinness (Knowledge Systems Laboratory and A. F. van. H. vu. n. Frank van Harmelen (Vrije Universiteit, “OWL Web Ontology Language Overview.” .

[12] T. Van Nguyen, W. Lim, H. Nguyen, D. Choi, and C. Lee, “Context Ontology Implementation for Smart Home,” 2010.

[13] W. Paper, “Internet of Things : Wireless Sensor Networks Executive summary.”

[14] L. Chen, C. Nugent, M. Mulvenna, D. Finlay, and X. Hong, “Semantic smart homes: Towards knowledge rich assisted living environments,” Stud. Comput. Intell., vol. 189, pp. 279–296, 2009.

[15] J. Ye, G. Stevenson, and S. Dobson, “A top-level ontology for smart environments,” Pervasive Mob. Comput., vol. 7, no. 3, pp. 359–378, 2011.

[16] J. H. Kim, H. Kwon, D. H. Kim, H. Y. Kwak, and S. J. Lee, “Building a service-oriented ontology for wireless sensor networks,” Proc. - 7th IEEE/ACIS Int. Conf. Comput. Inf. Sci. IEEE/ACIS ICIS 2008, conjunction with 2nd IEEE/ACIS Int. Work. e-Activity, IEEE/ACIS IWEA 2008, no. March 2016, pp. 649–654, 2008.

[17] C. Y. Leong, A. R. Ramli, and T. Perumal, “A rule-based framework for heterogeneous subsystems management in smart home environment,” IEEE Trans. Consum. Electron., vol. 55, no. 3, pp. 1208–1213, 2009.

[18] M. Sohn, S. Jeong, and H. J. Lee, “Self-evolved ontology-based service personalization framework for disabled users in smart home environment,” Proc. - 7th Int. Conf. Innov. Mob. Internet Serv. Ubiquitous Comput. IMIS 2013, pp. 238–244, 2013.

[19] E. Meshkova, J. Riihijarvi, P. Mahonen, and C. Kavadias, “Modeling the home environment using ontology with applications in software configuration management,” 2008 Int. Conf. Telecommun., pp. 1–6, 2008.

[20] L. Nachabe, M. Girod-Genet, and B. ElHassan, “Unified Data Model for Wireless Sensor Network MyOntoSens Ontology,” IEEE Sens. J., 2015.

[21] T. George and B. George, “OWL-S : Semantic Markup for Web Services.”

[22] B. Parsia and E. Sirin, “Pellet: An owl dl reasoner,” in Third International Semantic Web Conference-Poster, 2004, vol. 18.

[23] J. Ferber, Multi-agent systems: an introduction to distributed artificial intelligence, vol. 1. Addison-Wesley Reading, 1999.

[24] G. Zonta, P. Jr, J. Daltio, and C. B. Medeiros, “A Mechanism for Propagation of Semantic Annotations of Multimedia Content,” vol. 5, no. 4, pp. 332–342, 2010.

[25] H. Lentzner, D. Ph, D. Hoyert, D. Ph, K. N. Robinson, and D. Ph, “Trends in Causes of Death,” no. 1, pp. 1–9, 2001.

[26] B. Technologies, “Bluetooth ® low energy technology.”

[27] M. Lanthaler and C. Gütl, “On using JSON-LD to create evolvable RESTful services,” in Proceedings of the Third International Workshop on RESTful Design, 2012, pp. 25–32.

[28] Z. Shelby, K. Hartke, and C. Bormann, “RFC 7252: The Constrained Application Protocol (CoAP).” p. 112, 2014.

[29] X. Zhu and Y. Feng, “RSSI-based Algorithm for Indoor Localization,” vol. 2013, no. May, pp. 37–42, 2013.

[30] L. Nachabe, M. Girod-Genet, B. ElHassan, and F. Aro, “Applying Ontology to WBAN for mobile application in the context of sport exercises,” in Proceedings of the 9th International Conference on Body Area Networks, 2014, pp. 204–209.

[31] F. Sposaro and G. Tyson, “iFall: An Android Application for Fall Monitoring and Response.”