Software Architecture for Modern Telehealth Care Systems

[1]Alar Kuusik, [2]Enar Reilent, [3]Ivor Lõõbas, [4]Marko Parve
[1, First author] *ELIKO Technology Competence Centre, Tallinn, Estonia, alar.kuusik@eliko.ee*
[2,3]*ELIKO Technology Competence Centre, Tallinn, Estonia, Firstname.Lastname@eliko.ee*
[4]*East Tallinn Central Hospital, Tallinn, Estonia, marko.parve@itk.ee*
doi:10.4156/aiss.vol3. issue2.16

Abstract

Results of several research groups indicate that modern home telehealth care systems should support patient personalization and context awareness. To deal with accompanying increase of data amount and processing complexity, a semantic reasoning approach is proposed. However, so far there are no practical, system level software architectures proposed to address all related issues within one complete solution. We describe a developed RDF blackboard based data processing solution for smart home telecare supporting off-the-self reasoning tools and existing ontologies.

Keywords: *Telecare, Patient Monitoring, Assisted Living, Smart Home, Semantics, Agent Software Architecture, Formal Reasoning*

1. Introduction

For the delivery of public healthcare, telecare (tele-homecare), an essential part of eHealth technologies, provides the cost effective way to manage burdens on public services caused by an increasing proportion of elderly and chronically ill people [1]. Additionally, to reduce demand on clinics and home visits for patients the long term human monitoring is believed to be an effective way for early discovery of health risks. Increased acceptance and adoption of preventative care regimes within 'wellbeing' programs is important, for example smoking cessation and weight reduction also require active patient monitoring at home. While the early patient telecare systems were designed just for acquisition and offline monitoring of health parameters, and were later extended with real-time monitoring of safety boundaries, then the modern patient home monitoring solutions essentially analyze the context and focus on investigation of long term trends. Certainly, to discover slight drifts in patient's data the measurement context has to be taken into account. Strong external factors like recent physical activities, stress, and irregular lifestyle may hide changes in patient's data over a long period. From one side, the existing telecare solutions are designed as classical data acquisition systems not supporting context information for detailed patient data analysis. From the other side, if context information is provided, it is usually not machine tractable (outside of the particular system) making large scale statistical data analysis virtually impossible. Present paper describes *content centric* architecture for home monitoring solutions supporting machine reasoning of semantically encoded sensor data and context information.

2. Previous work

2.1. Progress of home telecare services

Telecare and assisted living is an emerging topic of cost efficient health care. Average telecare savings from the community perspective that are mentioned in the literature or achieved in trials are about 15-30% [2]. As described by Doughty [3] et al., the first generation telecare technology solutions enable to summon help in emergency, the second generation provide automated detection of emergencies, and the third enable monitoring of deterioration of human well-being. The first generation solutions in the form of panic buttons are available to the whole well-developed world. The main disadvantage of such technology, though, is that patient may be unable to proceed with an emergency call, or one tends to use the emergency hotline improperly. Significant involvement of qualified medical staff is a disadvantage as well.

Commercial solutions of the second generation, e.g. Well@home, Zydacron, Docobo, and Philips Motiva, have been available for about 10 years allowing automated monitoring of patient's safety based on periodic measurements. One of the significant advantages of the second generation is the reduced need for professional medical assistance. However, existing practical solutions support only resting condition measurements because of complexity of processing dynamically changing context. On the other hand, trends clearly show improvements in the context awareness, e.g. use of accelerometer data in mobile telemedicine [4]. For example, recent physical activities detected prior to the heart rate measurements help to avoid faulty emergency actions while still maintaining high sensitivity of the system. For simple personalization and context awareness policy-, i.e. ECA rule-based home care systems are recommended by Turner [5] et al. Rule-based data interpretation for home care solutions has been investigated in various research projects [6, 7] but there are no standardized frameworks for simple reuse of the domain knowledge.

The third generation telecare solutions, also called lifestyle-monitoring by Barnes [8], essentially focusing on continuous monitoring and analysis of Activities of Daily Living (ADL), primarily including the duration, frequency and patterns of physical activities as described by Amaral [9] et al. giving *long term* context to medical sensor data. Wide deployment of micro-mechanical sensors on mobile phones just recently opened practical possibilities to allow ADL monitoring [10]. Continuous ADL monitoring carries essential information about degradation of well-being [9] and can be used for discovery of emergency conditions, even without medical parameter sensing [11]. Some recent telecare system prototypes with lifestyle monitoring are created by Amaral, Kaushik [12] et al., and others. However, there are several principal issues that are not addressed in such prototypes. Information describing patient daily activities may be considered very delicate and in some countries there are legal restrictions for centralized processing of such data. Amount of collected ADL information is remarkable and therefore on-site data aggregation and processing is feasible as well. Development of rules for processing broad range of ADL information is a complex and time consuming task stressing the needs for knowledge formalization and reusability among many telecare patients.

2.2. Smart Home technology targeting modern telecare

Smart Home (SH) systems are the most widely used technology platforms for third generation telecare targeting home based elderly and disabled people. Such improved SH systems have (wireless) interfaces for medical sensors, typically cardiovascular and respiratory monitors, weight scales, blood sugar meters, in addition to traditional micro-climate and entertainment control interfaces. Native SH components, like movement detectors, video cameras, and home appliance control circuits, efficiently provide required ADL context data. Recently, mobile communication devices equipped with micro-mechanical sensors became useful for cost-efficient monitoring of physical activities and emergency conditions like falling down. Therefore, importance of mobile telemedicine is rapidly increasing but it is significantly harder to take into account the ADL information collected at different locations and environments. In principle, the telecare software solutions described in present paper are also applicable to mobile telemedicine. However, specific issues related to handling dynamic ADL data sources will not be deeply analyzed.

Rule- and logic-based (including Fuzzy logic-based) control is the leading method for SH implementations [13]. Therefore, rule-based patient data handling and personalization is natural for ADL aware telecare. However, as stressed by Nugent and Chen [14] et al., even without telecare functionality the SH environments are producing massive amounts of data and, until supplemented with a well-defined meaning, the potential of smart homes assisting capabilities will not be fully achieved, and propose semantic data integration approach demonstrated in their SemanticsAtHome project. The importance of semantic context data fusion has also been stressed in AALIANCE telecare roadmap [15]. The main disadvantage of conventional, non-semantic approach is complexity of (automated) reuse of existing knowledge, e.g. interpretation of rules encoded within different terminology systems. Apparently, it is very hard to copy and reuse the information and knowledge derived from a particular SH installation because a) there are no widely accepted standards for presenting sensor-actuator data, and b) systems typically use low-level data formats and its conversion into human and machine understandable formats for wide reuse is weakly motivated and a hard task.

Semantic assisted living projects claim that processing of formal semantically enriched content, its analysis, and decision support for intervention can be done more easily. As described by Redondo et al. [16], semantic representation simplifies SH service composition which is also important for adaptive telemonitoring. Additionally, semantically annotated data is more easily comprehensible for external services, e.g. medical decision support systems, nationwide Electronic Health Record (EHR) repositories (e.g. [17]), and global health statistic databases. Within the MATCH project Turner [18] proposed to use ontologies for data clustering. However, the works of both Nugent and Turner do not specify any real existing ontologies, essentially medical ones, to be used for achieving SH and telecare data interoperability in practice. Authors of [19, 20] propose different ontologies for describing a SH environment context. Comprehensive list of existing alternatives presented by W3C [21] show that is quite unlikely to agree on a single ontology to be used for smart home telecare.

3. Proposed software architecture for modern telecare systems

3.1. Functional requirements for the telecare system and its software

Based on analysis of recent related work we can say that modern telecare solutions should have the following capabilities:

- Support for ADL monitoring, in addition to the medical data acquisition.
- Support for personalization by means of customization of safety cutoff values and typical ADL behavior patterns.
- Interoperability with SH automation systems for home based patient context monitoring.
- Interoperability with mobile context sensing devices, e.g. mobile phones with positioning and acceleration sensing.
- Simple knowledge reuse and portability across different hardware based systems (sensors, communication infrastructure).
- Possible run-time renewal of medical domain expert knowledge from centralized repositories.

Semantic content-driven data processing solutions simplify satisfying described requirements for modern telecare systems. However, the existing commercial telecare systems are based on conventional client-server architecture and predefined communication protocols. Systems support restricted set of low-level messages and predefined set of hardware components. Naturally, those implementations, even though designed to be compatible with HL7 [22] patient information exchange standard, rely on predefined messages to be exchanged between the content source and the destination user.

The existing rule-based and semantic approaches of telemonitoring present theoretical State-of-the-Art and do not propose any practical implementation frameworks for their implementations. Essential feature for modern telecare systems is the extendability by means of simple reconfiguration of hardware and introduction of new knowledge in form of rules, signal processing algorithms, and services. We propose an open, agent-based software architecture for home and mobile telemonitoring which natively supports treatment of semantic content, including computationally feasible on-site formal reasoning and data aggregation.

3.2. Agent based software architecture

The proposed architecture (Figure 1) is a distributed multiagent system of asynchronous processes following classical blackboard model of Hayes-Roth [23]. The agents (typically executable processes) are running on the same hardware device (embedded controller, server, smartphone) writing data into a universal semi-realtime data-store (installed essentially on the same hardware device for high access speed) while every agent can asynchronously access *all* data inserted to the data-store. The complete system containing different hardware platforms forms a Monitoring Device Hierarchy (MDH) with several blackboard-agents sets. The first (lowest) MDH level software agents run on intelligent sensors implemented on micro-controllers. For example, a wired or wireless motion detector may be a hardware platform for the first level MDH agents. The 2nd MDH level agents and corresponding blackboard data-store operate on SH controllers or mobile devices. The higher, 3...n MDH level software agents operate on hospital servers and data clouds. The simplest practical telecare software implementation contains 2^{nd} and 3^{rd} MDH levels, SH controller and hospital server levels, both having their own blackboard data-stores.

Essential but not limited agent set for semantic telecare data processing includes:

- Sensor agents acquiring data from individual sensors and publishing data on the blackboard. Sensor agents are responsible for propagating sensor configuration information from the blackboard back to the sensor. It is important that there is no target user (agent) specified for the acquired content in advance.
- Data processing agents performing variety of signal processing, outlier detection, and aggregation tasks, as well as discarding obsolete data. Data processing also includes controlling the behavior of the system and decision making based on the given set of rules. Thus, present data processing agents are essentially formal (semantic) reasoners.
- Output agents communicating with host services (devices on higher MDH levels), for example an output agent running on SH central controller is responsible for exporting patient's data to the hospital system and downloading new configuration settings, profiles, commands, and messages.
- HCI agents used for adaption of user interfaces according to access profile on the home controller screen or the hospital web site, also for alerting and user feedback processing.

For communication between the second and higher MDH levels conventional SOAP messages may be an optimal solution. However, between (wireless) sensor devices and the gateway level device the custom communication protocols, given by sensor manufacturers, are difficult to avoid. The agents within one machine are built around a fast and transparent RDF [24] data-store implemented in shared memory for embedded home controllers, or in conventional database for servers. In the future it would be possible to use internal data-stores of reasoning tools for RDF encoded data [25]. Making use of the RDF representation for data encoding allows handling and saving different structures (like sensor readings, configurations, profiles, commands, reports, etc.) in the same universal manner. As mentioned, data written to the data-store is available to every process running on the same device. Thus, every write is like a broadcast. The RDF data-store serves three roles:

- A postbox between different agents within one controller device (including transparent external world communication with the aid of dedicated agents).
- A low-latency in-memory universal data-store for keeping data.
- A deductive database, using a rule language for rule-based generation of new facts.

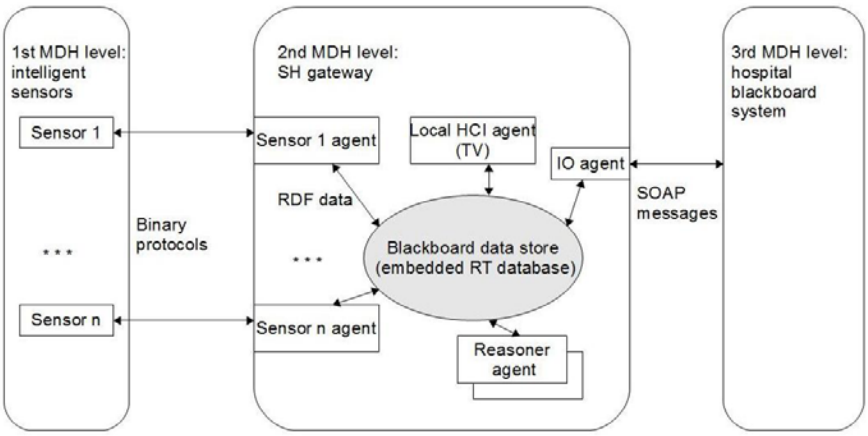

Figure. 1. 3-level example of proposed telehealth care architecture

The main advantage of the blackboard based architecture in comparison to the popular socket based systems is that there is no need to specify target user processes (of local real-time data) in advance. Agents producing the data can work independently from the receiving agents regardless of their existence. For monitoring applications it is crucial not to miss any incoming events while data processing has weaker real-time constraints. By using the proposed architecture it is always possible to add or modify content processing agents without the need to modify sensor specific and other low-level, performance optimized, agents. It is possible to run (even simultaneously) different sensors that measure and output the same parameter. As no sockets are used in the implementation of the blackboard on the MDH level 2 device, certain load on the operating system (kernel) is reduced that is essential for acquisition of streaming sensor data, e.g. ECG. Benchmarking experiments show that writing data into the implemented RDF data store is at least 100 times faster in comparison with the conventional SQLite database.

4. Semantic content presentation

We propose using full RDF representation of data within the system, starting right from the sensors. The RDF format (and associated OWL-based ontology systems) for defining and describing relations and concepts is emerging for different computerized data processing applications using “the semantic web” standards and technologies. RDF data (knowledge) encoding simplifies integration with (different) existing formal knowledge processing tools. As noticed by Tian [26] et al., the multimodal reasoning has advantages for efficient processing medical domain information that natively contains both rules and behavioral data. There exist some XML-based semantic data representation solutions developed for sensors, like SensorML, Hydra middleware and other dedicated middlewares for managing medical data of assisted living (e.g. [27]). However, those solutions are not fully RDF compatible which makes formal reasoning more complex, which also applies to encoding and handling of structures from various different domains. From the other side, XML format has certain communication overhead which is especially problematic for wireless sensors. The additional advantage of using standard RDF is the possibility of using existing SPARQL tools as well as new developed querying methods [28].

In our approach we stress to use real existing ontologies available on the web. That way we can guarantee that correct interpretation of information is always possible for content user (while correct use still remains the responsibility of the user). The proper use of keywords in medical domain is especially important. For example for “heart rate” there are around 20 different terms in use, some of them are equal while some have specific flavor. Inadequate labeling and later interpretation of sensor signals may lead to critical situations for patients. From the other side, if used ontologies are published and accessible, automated conversion of information is a relatively simple task. The widely accepted good medical taxonomy is SNOMED CT (Systematized Nomenclature of Medicine - Clinical Terms) [29]. Wordnet linguistic vocabulary can satisfy the broadest range of information formalization needs. However, the large number of existing and competing semantic sensor ontologies [30] demonstrate that simultaneous support for multiple terminology systems is a must.

Tables 1 and 2 show representation examples of semantic medical sensor data encoded into RDF triplets. Every sensor agent (adapter) in the system is identified by a URI. Using this URI all relevant information concerning any particular agent is easy to find, for example agent's configuration data, description, and output data. Table 1 presents configuration data about the pulse oximeter sensor agent, a particular wireless instrument for taking photoplethysmographic (PPG, tissue transparency) measurements and outputting heart rate and blood oxygen saturation level (SpO_2).

Table 1. RDF coding of PPG sensor description:

Subject	Property	Value
http://www.eliko.ee/demo/ssg/schema#nonin_onyx2	http://www.eliko.ee/demo/ssg/schema#Type	http://www.csiro.au/Ontologies/2009/SensorOntology.owl#Sensor
http://www.eliko.ee/demo/ssg/schema#nonin_onyx2	http://www.csiro.au/Ontologies/2009/SensorOntology.owl#ModelNumber	9560BT
http://www.eliko.ee/demo/ssg/schema#nonin_onyx2	http://www.csiro.au/Ontologies/2009/SensorOntology.owl#measures	http://bioinfo.icapture.ubc.ca/subversion/SIRS/clinicalphenotype.owl#HeartRate
http://www.eliko.ee/demo/ssg/schema#nonin_onyx2	http://www.csiro.au/Ontologies/2009/SensorOntology.owl#measures	http://bioinfo.icapture.ubc.ca/subversion/SIRS/clinicalphenotype.owl#SaturationO2
http://www.eliko.ee/demo/ssg/schema#nonin_onyx2	http://xmlns.com/wordnet/1.6/Configuration	#3021
#3021	http://xmlns.com/wordnet/1.6/Sleep	1

The first triple says that the agent identified by the URI "http://www.eliko.ee/demo/ssg/schema#nonin_onyx2" is registered as a physical PPG sensor (with the dedicated software agent) in our system. The second fact records the model number of the sensor used and the two following facts indicate that the sensor measures heart rate and SpO_2. The last two rows demonstrate the usage of RDF hierarchy – one triple represents the configuration data of the agent and the other one (linked through the system internal ID #3021) is a PPG device agent parameter with a value (sleep time 1 sec). URIs for different concepts are taken directly from ontologies, if available, or are otherwise composed artificially. For example "urn:snomed-ct:271650006" represents SNOMED CT term "271650006 Diastolic blood pressure (observable entity)".

The same sensor adapter writes its output data (of one measurement) as shown by the four rows in the Table 2. The first triple defines the data to be a sample by the particular agent and links all triples together as one object. The remaining three triples describe the result of the measurement – indicating that the measured heart rate value was 73 beats per minute and the blood oxygen saturation was 98%. Also the timestamp of the moment when the measurement was taken is recorded.

Table 2. RDF coding of PPG sensor output data:

Subject	Property	Value
http://www.eliko.ee/demo/ssg/schema#nonin_onyx2	http://www.eliko.ee/demo/ssg/schema#Sample	#3920
#3920	http://bioinfo.icapture.ubc.ca/subversion/SIRS/clinicalphenotype.owl#HeartRate	73
#3920	http://bioinfo.icapture.ubc.ca/subversion/SIRS/clinicalphenotype.owl#SaturationO2	98
#3920	http://knowledgeweb.semanticweb.org/iriba/ontologies/ResultOntology#timestamp	1264423298

5. Semantic reasoner agent integration

Proposed software architecture with RDF data-stores natively supports semantic rule-based reasoning – dedicated agent(s) can read any data from the blackboard data-store and write output back there for any other agent to use. Some applications for reasoning agents can be outlined:

- Derivation of personal safety threshold parameters from potentially large set of expertise facts, which are difficult to handle correctly by humans.
- Continuous real-time validation of safety requirements of patient supporting dynamically changing rules.
- Aggregation of sensor data with semantically annotated output.
- Interpretation of semantic context information.

- Profile based selection of HCI information optimized for patient itself or medical professional, etc.

Through our architecture we can apply several formal reasoners as different data processing agents in parallel for the same physical data-store. Apparently, the feature of concurrent processing is crucial to maintain real-time service for incoming sensor data. For real-time sensitive decision making, e.g. emergency condition detection, simple hard-coded rule-processing agents can be installed, as well as formal reasoning tools like Jena, CHR, Gandalf, Otter (Prover9), and others, for processing thousands of patient and context related facts.

5.1. Example of semantic reasoning with Prolog

For better illustration of using formal reasoning in the system, we can examine an example of processing of semantically represented rules and medical data with a blackboard agent executing Prolog programs.

To demonstrate some issues of the patient's data processing in home telecare system let us consider the following case of monitoring SpO_2 level on people with lung diseases. Too low oxygen saturation level in blood may have lethal consequences for such patients. However, from the healthcare practice it is known that normal saturation level depends on diagnosis of the particular patient. For example, normal SpO_2 indication for healthy people is 96 – 99%, less than 95% indicates respiratory insufficiency, and less than 90% indicates hypoxia with the need of emergency treatment. On the other hand, for people with chronic obstructive pulmonary disease (COPD) the normal SpO_2 value can be as low as 88 – 92%. It is also possible that individual harmless level of SpO_2 for a given patient is assigned by a doctor and standard limits should be discarded.

Due to that essential personalization feature, our knowledge base inside a home telecare controller may contain contradicting data. For example, based on the described patient's illnesses the rule system assumes that the minimal SpO_2 lower threshold value is “96” while at the same time other records suggest the threshold to be “90” (e.g. said by the doctor). Therefore, a defeasible logic inspired [31] reasoning is used. Our rule system keeps additional records related to the genuine data indicating origin of data and respective priority system.

For reasoning on given knowledge-base we use SWI-Prolog which is executed as a child process spawned by the corresponding adapter agent. The adapter agent feeds Prolog with blackboard data and queries. The following rule examples assume the data items are presented as individual triples separately forming facts with three arguments, e.g.: “fact(subject, property, value)”.

Due to the space constraints and better readability long URIs (as in Tables 1, 2) actually used in the rules are abbreviated in the following examples and replaced by intuitive short strings, e.g. “ppg_sensor_uri” stands for the identifier of the sensor agent managing the PPG sensor device (http://www.eliko.ee/demo/ssg/schema#nonin_onyx2). We can monitor all available SpO_2 data (as seen in Table 2) samples produced by the given agent and evaluate the emergency level with the given Prolog rule:

```
spo2_emergency_condition :-
        get_spo2_threshold(patient_uri, T),
        fact(ppg_sensor_uri, sample_uri, X),
        fact(X, spo2_uri, N),
        N<T.
```

The rule succeeds (Prolog answers 'true') if there exists at least one measured SpO_2 sample that is below the threshold value calculated by the rule “get_spo2_threshold” and fails in all other cases. The following rule “get_spo2_threshold” is meant to resolve all contradictions in the facts defining the threshold and return allowed minimal SpO_2 value for the given patient:

```
/* first alternative – if threshold given directly by a doctor, use it, ignore others */
get_spo2_threshold(P, T) :-
        fact(P, profile_uri, Y),
```

```
        fact(Y, thresholds_uri, Z),
        fact(Z, lower_SpO2_limit_uri, L),
        fact(L, set_by_uri, W),
        belongs_to_class(W, medical_personnel_uri),
        fact(L, threshold_value_uri, T),!.

/* second alternative – maybe there exists a threshold computed previously by the rule system itself */
get_spo2_threshold(P, T) :-
        fact(P, profile_uri, Y),
        fact(Y, thresholds_uri, Z),
        fact(Z, lower_SpO2_limit_uri, L),
        fact(L, set_by_uri, W),
        belongs_to_class(W, rule_system_uri),
        fact(L, threshold_value_uri, T),!.

/* third alternative – if no threshold records found (by doctors or by rules), give some general default */
get_spo2_threshold(P, T) :- T = 97.
```

The real threshold values are kept as regular facts under the patient's profile and each of them has sub-record indicating whom it was written by. Therefore, one parameter (e.g. threshold value) may have any number of values assigned if set by different sources. As values like SpO_2 threshold do not change very often but are used quite frequently then they are usually calculated in advance and stored on the blackboard, however they could be derived also during run time every time they are needed by any other rule. Under usual circumstances the rule “get_spo2_threshold” returns a value calculated by the system itself. However, if there exists a SpO_2 threshold value given directly by a doctor then it overrules all other possible values. For the extreme case, where no threshold records are found from the set of facts the rule returns a hard-coded value of “97”.

Another (simplified) rule (written in three alternatives in Prolog syntax) demonstrates how the system might derive the current minimal SpO_2 value for the given patient. If the patient has COPD diagnosed the value will be “88”, in case of any other pulmonary disease “92”, otherwise “96”:

```
calculate_lower_spo2_limit(P,V) :-
        fact(P, profile_uri, X),
        fact(X, diseases_uri, Y),
        fact(Y, disease_uri, copd_uri),
        V=88.

calculate_lower_spo2_limit(P,V) :-
        fact(P, profile_uri, X),
        fact(X, diseases_uri, Y),
        fact(Y, disease_uri, D),
        is_subclass_of(D, lung_disease_uri),
        V=92.

calculate_lower_spo2_limit(P,V) :- V=96.
```

The acquired value is stored under the patient's profile facts and its source is set to “rule system” indicating that the data is not given by medical staff but was automatically generated by rules instead. Responsible medical professionals can add new data at any time and it becomes superior from the rules' point of view – the software rule system has to take such “super user” facts into consideration and cannot erase or modify them.

As the central blackboard incorporates contextual and environmental sensor data besides pure medical data it is natural to make use of the contextual data in rules to enrich medical knowledge of the patient's condition. For example, if a heart rate sample is found to be below 50 beats per minute we can predict health (or device malfunctioning) problems. Alternatively, if there is a sample above 90, derivation cannot be made immediately. With sub-queries we try to justify the high pulse first and only if it is not possible then positive result (problem found) is returned. Say, if the patient has had significant physical movement activities 10 minutes prior to the pulse measurement, we have no reason to raise alarm. Contrariwise, if the patient has not moved around or has even been in bed then a heart rate over 90 indicates health problems. Contextual data needed for calculations in this case comes typically from the set of PIR sensors in the patient's home environment and/or accelerometer sensors attached to the patient (mobile phone, pedometer, wrist watch). The rule of the main query in Prolog syntax to raise alarm is as follows:

```
problem_suspected :-
        fact(_, 'sample_URI', Y),
        fact(Y, 'pulse_URI', X),
        fact(Y, 'timestamp', T),
        (
                X<50;
                X>90,
                (
                        not(was_active_between(T, T-(60*10)));
                        was_in_bed_between(T, T-(60*10))
                )
        ).
```

6. Solution testing

For testing feasibility of proposed telecare software architecture and semantic data encoding we implemented the prototype solution containing a home controller (2nd MDH level device), set of wireless Bluetooth and Zigbee sensors and PC based server emulating hospital server (3rd MDH level). Communication between the home controller and the hospital server is based on SOAP messages always initiated by the home controller for network safety reasons. According to the proposals of Continua Health Alliance [32] the home controller is realized as an advanced DVB-T receiver equipped with Ethernet and wireless sensor interfaces. The home controller has standard Linux running on 300MHz 32bit MIPS type CPU. Its blackboard is an original RDF memory database following the ideas described. Reasoning agents use SWI-Prolog (Version 5.6.58) engine. The hospital server runs conventional Postgres database. The implementation fulfills the performance requirements and interoperability framed by the hospital personnel. The most critical issue, as expected, is embedded reasoner performance due to the relatively weak hardware platform and the real time demands of the sensor communication. In benchmark tests we measured an average Prolog reasoning time of 310 ms for 4500 facts, and 420ms for up to 7500 facts. 750 tests were executed. The measured worst case reasoning time measured was 2 seconds. Achieved real life reasoning response is clearly sufficient to discover patient emergency occasions quickly enough for realistic medical knowledge base of some thousand facts.

7. Conclusions

For efficient utilization of ADL information, required for modern telecare, the data driven approach has advantages over constrained predefined solutions. Semantics driven approach is well suitable for interoperability in medical domain having strong terminology standardization background. However, existing semantic telecare proposals did not offer *practical* solutions for flexible representation of data through the accepted medical code systems that are suitable for machine reasoning. In this paper we described a flexible agent based software architecture solution that is compatible with RDF data representation, unrestricted amount of domain ontologies and supporting existing reasoning tools. Feasibility and efficiency of proposed telehealth care system architecture and data encoding solution was successfully evaluated in tests on a real telemonitoring system. Further enhancement of the system will focus on development of specific data processing agents for data validation and aggregation.

8. Acknowledgments

This research has been supported by the European Regional Development Fund, Competence Centre program of Enterprise of Estonia.

9. References

[1] http://www.health-informatics.org/tehip/tehipstudy.PDF (2005).
[2] G. Paré, M. Jaana, C. Sicotte, “Systematic review of home telemonitoring for chronic diseases: The evidence base.”, Journal Am Med Inform Association 2007;14, pp. 269–277; pp. 2007
[3] K. Doughty, K. Cameron, P. Garner, “Three generations of telecare of the elderly”, Journal of Telemedicine and Telecare, vol. 2, no. 2, pp. 71-80, 1996.
[4] L.C. Jatobá, U. Großmann, J. Ottenbacher, S. Härtel, B. von Haaren, W. Stork, K.D. Müller-Glaser, and K. Bös, “Obtaining Energy Expenditure and Physical Activity from Acceleration Signals for Context-aware Evaluation of Cardiovascular Parameters”, IFMBE Proc. 18, Springer-Verlag Heidelberg, pp. 475–479, 2007
[5] Feng Wang, Kenneth J. Turner, “Towards Personalised Home Care Systems“, In Proc. of the PETRA International Conference on Pervasive Technologies related to Assistive Environments, pp. L2.1-L2.7, 2008.
[6] J. C. Augusto, J. Liu, P. McCullagh, H. Wang and J.-B. Yang, “Management of uncertainty and spatio-temporal aspects for monitoring and diagnosis in a Smart Home.”, International Journal of Computational Intelligence Systems, 1(4), pp. 361-378, Atlantis Press, 2008
[7] K. Du et al., “HYCARE: A hybrid context-aware reminding framework for elders with mild dementia, in Proc of the 6th Int Conference on Smart Homes and Health Telematics”, (ICOST), Lecture Notes in Computer Science, Springer, Vol 5120/2008, Ames, IA, USA, pp. 9-17, 2008
[8] N. M. Barnes, N. H. Edwards, D. A. D. Rose, P. Garner, “Lifestyle monitoring - technology for supported independence.”, IEEE Computing and Control Engineering Journal, vol. 9, nr. 4, 1998, pp. 169-174, 1998
[9] T. Amaral, N. Hin and J. L. Arnott, “Integrating the Single Assessment Process into a lifestyle-monitoring system”, in 3rd International Conference On Smart homes and health Telematic (ICOST), pp. 42–49, 2005
[10] G. Bieber, J. Voskamp, and B. Urban, “Activity recognition for everyday life on mobile phones”, in HCI (6), vol. 5615/2009 of Lecture Notes in Computer Science, pp. 289–296, Springer, 2009
[11] A. Sixsmith, “An evaluation of an intelligent home monitoring system”, Journal of Telemedicine and Telecare 6 (2), pp. 63-72, 2000
[12] A. R. Kaushik , B. G. Celler, “Characterization of PIR detector for monitoring occupancy patterns and functional health status of elderly people living alone at home”, Technology and Health Care, v.15 n.4, pp. 273-288, 2007
[13] D. Cook and S. K. Das, “How Smart are our Environments? An Updated Look at the State of the Art”, Journal of Pervasive and Mobile Computing, 2007
[14] L. Chen, C.D. Nugent, MD. Mulvenna, D.D. Finlay DD, X. Hong, Semantic Smart Homes: Towards Knowledge Rich Assisted Living Environment, in book Intelligence on Intelligent Patient Management; Ed by S. McClean et al., Springer, ISBN 978-3-642-00178-9, pp. 279-296, 2009
[15] G. Van Den Broek, F. Cavallo and C. Wehrmann, “AALIANCE Ambient Assisted Living Roadmap”, The Netherlands, 2010
[16] Rebeca P. Díaz Redondo, Ana Fernández Vilas, Manuel Ramos Cabrer, José Juan Pazos Arias, Jorge García Duque, Alberto Gil Solla, “Enhancing Residential Gateways: A Semantic OSGi Platform”, IEEE Intelligent Systems, vol. 23, no. 1, pp. 32-40, 2008

[17] Bruce Moulton, Zenon Chaczko, Mark Karatovic, "Updating Electronic Health Records with Information from Sensor Systems: Considerations Relating To Standards and Architecture Arising From the Development of a Prototype System", JCIT, Vol. 4, No. 4, pp. 21 - 26, 2009
[18] K. J. Turner, Liam S. Docherty, Feng Wang and Gavin A. Campbell, "Managing Home Care Networks", In Proc. of 8th Int. Conf. on Networks, pp. 354-359, 2009
[19] E. Kim and J. Choi, "An Ontology-Based Context. Model in a Smart Home", In Proc. of UWSI Workshop on Ubiquitous Web Systems and Intelligence, pp. 11-20, 2006
[20] V. Ricquebourg, et al., "Context Inferring in the Smart Home: An SWRL Approach", in Proc of the 21st Int Conf on Advanced Information Networking and Applications Workshops, Vol. 02, ISBN:0-7695-2847-3, pp. 290-295, 2007
[21] http://www.w3.org/2005/Incubator/ssn/wiki/Review_of_Sensor_and_Observations_Ontologies (2011)
[22] http://www.ihe.net (2011)
[23] B. Hayes-Roth, "A blackboard architecture for control", Artificial Intelligence, 26(3), pp 251–321, 1985
[24] www.w3.org/RDF (2011)
[25] B. Parsia, "An Introduction to Prolog and RDF", XML.com, http://www.xml.com/pub/a/2001/04/25/prologrdf/, 2001
[26] J. Tian et al., "Multi-Modal Reasoning Medical Diagnosis System Integrated With Probabilistic Reasoning", Int Journal of Automation and Computing 2, pp. 134-143, 2005
[27] Ma Shou-ming, Wang Ru-chuan, Ye Ning , "Using Context Prediction for Elderly Health Monitoring in Pervasive Computing Environments ", JDCTA: International Journal of Digital Content Technology and its Applications, Vol. 5, No. 1, pp. 16 - 25, 2011
[28] Sung Wan Kim , "Improved Processing of Path Query on RDF Data Using Suffix Array", JCIT: Journal of Convergence Information Technology, Vol. 4, No. 3, pp. 45 - 52, 2009
[29] http://www.ihtsdo.org (2011)
[30] M. Compton, C. Henson, L. Lefort, H. Neuhaus and A. Sheth, "A Survey of the Semantic Specification of Sensors", In Proc of the 2nd International Workshop on Semantic Sensor Networks, 8th Int Semantic Web Conference (ISWC), Washington, pp. 17-32, 2009
[31] D. Nute, "Defeasible Logic", In Proceedings of the Applications of prolog 14th international conference on Web knowledge management and decision support, pp. 151 - 169, 2003
[32] R. Carroll, R. Cnossen, M. Schnell, D. Simons, "An Interoperable Personal Healthcare Ecosystem", IEEE Pervasive Computing, Vol. 6, No. 4, pp. 90-94, 2007