ISSN 1392 – 1215 ELECTRONICS AND ELECTRICAL ENGINEERING 2010. No. 9(105)
ELEKTRONIKA IR ELEKTROTECHNIKA

T 115 MEDICINE TECHNOLOGY
MEDICINOS TECHNOLOGIJA

Data Acquisition Software Architecture for Patient Monitoring Devices

A. Kuusik, E. Reilent, I. Lõõbas, A. Luberg
ELIKO Competence Centre, Estonia,e-mail: alar.kuusik@eliko.ee

Introduction

Telecare (tele-home-care) provides a recognized solution to control an increase of medical expenses caused by an increasing proportion of elderly and chronically ill people [1]. As well described already by Doughty et al. [2], the modern telecare solutions should be able to monitor of slow deterioration of well-being and discover health risks early. It is believed that discovery of slight deviations in health condition requires, additionally to medical parameter measurements, *lifestyle-monitoring* as outlined by Barnes [3]. Long term patient, especially such requiring lifestyle monitoring, causes significant increase of amount of data gathered and its handling involution that is usually not addressed on system architecture level in complex way. From one side, the existing telecare solutions are designed as classical data acquisition systems not providing flexibility to add context information. From the other side – which is more important – if the context information is provided in human readable way or binary encoded, it is not machine processable outside of a single institution. Present paper describes content centric software architecture for telecare systems that simplifies machine processing of patient and context information through the semantic representation of data and semantic reasoning.

Previous work

For the patient lifestyle monitoring the use of Smart Home (SH) control platforms is a common approach. Monitoring of the duration, frequency, and patterns of daily activities, e.g. sleeping and training times, give an important context for acquired medical measurements data and can be used for discovery of emergency conditions [4, 5]. As presented by Chen et al. [6], the SH environments are producing massive amounts of data from sensors and, until the data is imbued with well-defined meaning, the potential use of the system for describing lifestyle context for patients is rather limited. Obviously, it is difficult to unify and organize the human lifestyle information using low, communication message level terminology system. The use of widely accepted ontologies (controlled, relational vocabularies) allows higher level interpretation and reasoning of information by a user. Therefore it is already proposed to use ontologies for describing (smart) environment context [7]. Within MATCH project, Turner proposes to use ontologies for data clustering [8]. However, while the works of both Chen and Turner show that semantic enrichment of context information simplifies its processing, they do not specify any practical ontologies to use and propose how the semantic data shall be handled within a real telemonitoring system starting from the information source (semantic sensor) up to the personal health record database.

Essential components for patient modern telemonitoring include patient profile and automated sensor handling. From a theoretical side, the policy (i.e. rule) based home care systems are promising for personalization and simple customization by Turner and others [8, 9]. Again, described solutions represent prototype implementations not compatible with actual health records and practical data acquisition systems by means of sensor integration and existing formal reasoning tools. Situation awareness and environmental condition detection is sometimes performed ubiquitously by several sensor motes [10] but semantic data exchange between those motes is not applicable for practical implementations.

Proposed software architecture for modern telecare monitoring systems

Having based on analysis of recent related work we can say that modern telecare solutions should support:

- Lifestyle monitoring, in addition to medical sensing;
- Personalization of patient policies;
- Simple sensor integration, automated service invocation and runtime reconfiguration;

Semantic content driven data processing and rule based reasoning solution should satisfy described requirements for modern telecare systems. However, the commercial telecare systems described above have conventional client server based data acquisition system architecture with fixed communication protocols and preknown set of supported sensing devices. Naturally, those implementations, even designed to be compatible with with HL7 v3 XML standard [11] are targeting single application use. The rule based and semantic approaches of telemonitoring described above present theoretical State-of-the-Art and do not propose any practical implementation frameworks for their realizations. Essential feature for telecare systems is the extendability by means of simple reconfiguration of hardware and introduction of new knowledge in form of rules, data processing executables, etc.

We propose an open agent based software architecture, which supports semantic data processing and (soft) real-time time reasoning with patient policies.

Multiagent system

The proposed architecture is a distributed multi-agent system of independent asynchronous processes that follows the classical blackboard communication model of Hayes-Roth [12] - the agents (executable processes) within the same hardware device (embedded controller, server) communicate by writing data to the central datastore (within the same hardware device) and every agent can access all data inserted to the datastore.

The main advantage of the blackboard (pull mode) communication based architecture is that, opposite to popular socket based communication, there is no need to specify target user processes (of local real-time data) in advance. There is always a possibility to add or modify content processing agents without the need to modify sensor and other hardware related agents.

The different Monitoring Device Hierarchy (MDH) software agents are running on intelligent sensors, Smart Home (SH) controller(s), hospital servers, PC clusters, etc. The simplest practical telecare software implementation contains telemonitoring gateway and hospital MDH levels both having their own blackboard datastore. Main agents by functionalities we propose:

- Sensor agents acquiring data from individual sensors, publish it on the blackboard and send the configuration info published on the blackboard back to sensor. Important is that there are no target user (agent) specified for the acquired content in advance. In the case the sensor agent receives raw data from hardware device it has to convert data into the semantic representation described below,
- Data processing agents performing variety of signal processing data inconsistency discovery tasks, essentially presenting data processing agents are (semantic) formal reasoners,
- Output agents communicating with host services (devices on higher MDH levels), for example an output agent running on SH central controller is responsible to export patient data to hospital system and download new configuration settings,
- HCI agents for displaying profile based selection of information on home screen or hospital web site.

Data presentation

All data in the system has to be presented semantically by using well-defined terms that are understood by different parties in the same way for supporting later processing, exchange and unified interpretation.

In our approach we stress to use the ontologies available on the web. That way we can guarantee that correct interpretation of information is always possible for content user (correct use still remains user responsibility). Especially important is the proper use of keywords in medical domain - for „heart rate“ for example, around 20 different terms are in use, some of them are equal, some have specific flavor. Inadequate labeling and later interpretation of sensor signals may lead to critical situations for patients. From the other side, if the used ontologies are published and accessible, automated conversion of information is a simple task. Good, worldwidely accepted ontology (vocabulary) for medical domain is proposed by SNOMED CT [13]. Right now we are using several different ontologies, further narrowing of selection will be done.

Different agents inside a system module (e.g., a set-top box or a aggregation server) are determined to share a common data model and data storage/exchange environment (blackboard). The schema of data representation within such system has to be universal and flexible. Therefore we propose to use RDF triples to present all the data, starting right from the sensors. The RDF-based data representation (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. Using RDF data (knowledge) coding makes it possible to integrate *existing* formal reasoners and other knowledge processing tools. There are some XML-based semantic data presentation solutions developed for sensors like SensorML, Hydra middleware. However, those solutions are not fully RDF compatible which makes formal reasoning more complex. From the other side, XML format has a certain communication overhead, especially for wireless sensors. The additional advantage of using standard RDF allows using existing SPARQL tools.

A fact (a data item) on the blackboard in our system has the following fields:

- Subject: *id* of whatever has the *property*;
- Property: name of the property of the subject. In RDF terminology this would be the *predicate*;
- Value: value of the property. In RDF terminology this would be the *object.*

For example, PPG sensor (Bluetooth Nonin Onyx II device to gather and formalize pulse and blood saturation readings) could be described by following triples:

- *PPG_URI, registeredAs_URI, sensor_URI;*
- *PPG_URI, measures_URI, pulse_URI;*
- *PPG_URI, measuers_URI, saturationO2_URI;*
- *PPG_URI, modelNumber_URI, 9560BT.*

Realization of agents

The blackboard is implemented as a transparent custom shared memory datastore. There is no separate process for handling the datastore, agents in the system can use a set of API calls to insert and query data (triples) from shared memory. The datastore serves as a mid-term memory for storing data and as a postbox between different agents. It can be also seen as a deductive database for reasoner, using a rule language for rule-based generation of new facts.

We demonstrate the common model of an agent on the sensor adapter (agent) for the selected PPG sensor. The agent runs always in background, monitoring whether the sensor is online or not. When the sensor device is turned on the agent establishes connection immediately and receives all the output data generated by the device, which is inserted into the database in the form of triples. The agent also queries regularly the database for getting its configuration parameters or other input data.

The example output of a single measurement looks as following („#3920“ is a unique key for binding the particular set of different triples into one entity):

- *PPG_URI, sample_URI, #3920;*
- *#3920, pulse_URI, 73;*
- *#3920, saturationO2_URI, 98;*
- *#3920, timestamp_URI, 1264423298.*

With our architecture we can apply several formal reasoners as different data processing agents in parallel for the same physical datastore. For data processing and decision making, e.g. emergency condition detection, simple hardcoded agents can be installed, as well as full formal reasoners like Jena, Gandalf and others for handling thousands of patient and context related facts. For example, we can describe processing of semantically represented rules and data of SpO2 meter with a blackboard agent executing Prolog programs.

It is known that normal SpO2 value for healthy people is 96-99%, whereby the value < 95% indicates respiratory insufficiency and the value < 90% indicates hypoxia with the need of emergency treatment. However, for people with COPD (chronic obstructive pulmonary disease) a normal SpO2 diagnosed value is between 88-92%. Suppose we have a precalculated fact in our memory datastore being specific to the patients's critical SpO2 value: *patient_URI, critical_SpO2_URI, 90.* The respective agent could check whether the SpO2 value is critical or not by querying Prolog the following rule, which answers „false“ if no SpO2 sample is found below critical value and „true“ if all preconditions of the rule are satisfied (and variable X is instantiated to one particular SpO2 value which is under the limit):

spo2_problem(X) :-
fact('PPG_URI', 'sample_URI', Y),
fact(Y, 'saturationO2_URI', X),
fact(patient_URI,'critical_SpO2_URI', Z),
X<Z.

For uploading data to next MDH levels, we have a separate agent responsible for exchanging facts with a data aggregation server for sending relevant data items from the local memory database to the server and downloading/receiving new commands and configurations, which will be stored back into the memory database available for all other agents.

Transmitted data amounts could become relatively large, for example, one pulse-oxymeter sample as it was presented previously (if encoded into full human-readable string) takes 350 bytes. Compression or re-encoding of URIs containing triples is possible, though one must consider the trade-off for compatibility with potential data consumers in other MDH layers.

For reducing the load of communication channels, and especially central aggregation servers (can incorporate monitoring data from several hundreds of patients), a considerable amount of raw sensor data processing and analyzing is done locally on home telemonitoring gateway. Therefore, only the results/reports of the data processing and irregular individual values shall be uploaded by default, while other levels have possibility to query additional raw (gathered) data.

Implementation and testing

For testing the proposed architecture, we implemented the software for enhanced DVB receiver. This device is equipped with Bluetooth and Zigbee interfaces to support wireless medical and presence sensors. The device has 300MHz 32bit MIPS type CPU and non-RT Linux OS. Wireless Onyx II 9560 SpO2 sensor was used for testing, see (Fig. 1.). The processed information was sent to the hospital server using SOAP (Simple Object Access Protocol) messages.

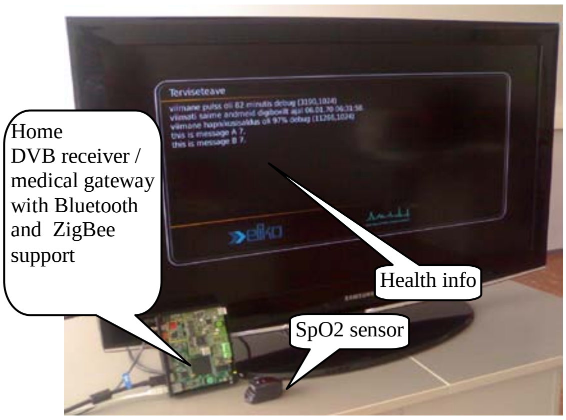

Fig. 1. Instrumentation of home monitoring system

Blackboard on home controller is a custom memory database, reasoning agents use SWI-Prolog (Version 5.6.58) engine. Conventional Postgres database and Jena reasoner run on hosptial server. The implementation appears to fulfill requirements of simple customizability and can handle semantic content. The most critical issue is, as expected, the performance of the reasoner on smart home controller, because of weak hardware platform. In real tests we measured an average (of 750 tests) of Prolog reasoning time of 310 ms of 4500 facts and 420ms of up to 7500 facts. The measured worst case reasoning time was 2sec, which is clearly sufficient to discover patient emergency conditions quickly enough and the home factset should not exceed 1-2 thousand facts by our expectations.

Future work and conclusions

The further work will focus on optimization of the software implementation and integration of feasible set of medical and smart environment domain ontologies. The target is real life use of the developed telecare architecture and its software implementation. Experiments show that the proposed software architecture satisfies real life needs of modern patient home monitoring solutions, semantic reasoning and computerized personalization.

Acknowledgements

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

References

1. The e–Health Innovation Professionals Group. Online: http://www.health–informatics.org/tehip/tehipstudy.PDF.
2. **Doughty K., Cameron K., Garner P.** Three generations of telecare of the elderly // Journal of Telemedicine and Telecare, 1996. – Vol. 2. – No. 2. – P. 71–80.
3. **Barnes N. M., Edwards N. H., Rose D. A. D., Garner P.** Lifestyle monitoring – technology for supported independence // IEEE Computing and Control Engineering Journal, 1998. – Vol. 9. – No. 4. – P. 169–174.
4. **Amaral T., Hine N., Arnott J. L.** Integrating the Single Assessment Process into a lifestyle–monitoring system // 3rd International Conference On Smart homes and health Telematic (ICOST 2005), 2005. – P. 42–49
5. **Sixsmith A.** An evaluation of an intelligent home monitoring system // Journal of Telemedicine and Telecare, 2000. – Vol. 6. – No. 2. – P. 63–72.
6. **Chen L., Nugent C. D., Mulvenna M. D., Finlay D. D., Hong X.** Semantic Smart Homes: Towards Knowledge Rich Assisted Living Environment, in Intelligence on Intelligent Patient Management (Ed. by McClean S., Millard P, El–Darzi, Nugent C). – Springer, 2009.
7. **Kim E., Choi J.** An Ontology–Based Context Model in a Smart Home, Notes in Computer Science. – Publisher Berlin/Heidelberg: Springer, 2006.
8. **Turner K. J., Docherty L. S., Wang F., Campbell G. A.** Managing Home Care Networks (in Robert Bestak, Laurent George, Vladimir S. Zaborovsky and Cosmin Dini (eds.)) // Proc. 8th Int. Conf. on Networks, IEEE Computer Society, 2009. – P. 354–359.
9. **Augusto J. C., Liu J., McCullagh P., Wang H., Yang J.–B.** Management of uncertainty and spatio–temporal aspects for monitoring and diagnosis in a Smart Home // International Journal of Computational Intelligence Systems. – Atlantis Press, 2008. – No. 1(4). – P. 361–378.
10. **Leonaite A., Vainoras A.** Heart Rate Variability during two Relaxation Techniques in Post-MI Men // Electronics and Electrical Engineering. – Kaunas: Technologija, 2010. – No. 5(101). – P. 107–110.
11. **Dimitrov D. Tz., Guergov S., Ralev N. D.** Multifunctional Adaptive System for Physiotherapy with Measurement Devices // Electronics and Electrical Engineering. – Kaunas: Technologija, 2010. – No. 5(101). – P. 111–114.
12. **Hayes–Roth B.** A blackboard architecture for control // Artificial Intelligence, 1985. – No. 3(26). – P. 251–321.
13. International Health Terminology Standards Development Organisation. Online: http://www.ihtsdo.org (2010).

Received 2010 02 15

A. Kuusik, E. Reilent, I. Lõõbas, A. Luberg. Data Acquisition Software Architecture for Patient Monitoring Devices // Electronics and Electrical Engineering. – Kaunas: Technologija, 2010. – No. 9(105). – P. 97–100.

Patient remote monitoring has continuously rising importance for aging countries. Computer based assisted living systems are too difficult to use by elderly people, therefore it has been proposed to extend home multimedia devices with patient monitoring functionalities. However, such implementations are usually platform and sensor specific making their extension and reuse complicated and time consuming. We propose an agent based software architecture for embedded patient monitoring devices that is built around a common database. The solution is open for adding any kind of medical sensors or signal processing software just by writing small software adapter. System can be reconfigured during operation. As one of the unique feature the system has built in formal reasoner to detect inconsistencies among sensor data and process patient unique safety rules in real time. The solution is prototyped. Ill. 1, bibl. 13 (in English; abstracts in English and Lithuanian).

A. Kuusik, E. Reilent, I. Lõõbas, A. Luberg. Duomenų surinkimo programinės įrangos iš pacientų stebėsenai skirtų įrenginių architektūra // Elektronika ir elektrotechnika. – Kaunas: Technologija, 2010. – Nr. 9(105). – P. 97–100.

Pacientų nuotolinių stebėjimo sistemų svarba ypač didėja „senstančiose“ valstybėse. Vyresnio amžiaus žmonėms sunkiau naudotis kompiuterių sistemomis, todėl buvo pasiūlyta pacientų stebėsenos funkcijas įdiegti į multimedijos sistemas. Tačiau jutiklių taikymas tokiose sistemose sukelia nemažai problemų. Esamiems pacientų stebėsenos įrenginiams pasiūlyta programinės įrangos architektūra, pritaikoma įvairių tipų jutikliams ar signalų apdorojimo programinei įrangai. Vienas iš esminių siūlomos sistemos pranašumų – paciento būsenos parametrų stebėsena esamuoju laiku lyginant jų tikroviškumą. Il. 1, bibl. 13 (anglų kalba; santraukos anglų ir lietuvių k.).