Providing Interoperability to a Pervasive Healthcare System Through the HL7 CDA Standard

Albert Brugués[1,2], **Stefano Bromuri**[1], **Josep Pegueroles**[2], **Michael Schumacher**[1]

[1]University of Applied Sciences Western Switzerland (HES-SO), Sierre, Switzerland

[2]Universitat Politècnica de Catalunya - BarcelonaTech (UPC), Barcelona, Spain

Abstract

Background: Pervasive healthcare is a new paradigm of healthcare services where the patients are active participants on their own well being. The development of Pervasive Healthcare Systems (PHSs) consists on approaching monitoring solutions into the hands of the patients, and has been reported as a key to minimize the healthcare costs due to the aging of population. However, interoperability is a technological challenge not taken into account in most of the existing implementations of PHSs.
Objectives: This paper focuses on how we provide interoperability to a PHS for the management of the gestational diabetes mellitus (GDM) by using the CDA standard. In this monitoring system an Android application sends CDA documents to the server side of the system, so that the health information reported by the patient is transmitted over the Internet in an interoperable way.
Methods: The generated CDA documents report on three different aspects related with GDM that are: physiological parameters, symptoms and medications. Each one of them is encoded as a section in the body of the CDA. To build these CDA documents, different pre-installed XML templates are combined and filled by using XPath.
Results: Doctors using this system want that patients report both: the dose quantity taken and the dose prescribed for the insulin bolus. As a consequence we had to extend the SubstanceAdministration class with a new element to encode all the semantics.
Conclusions: This paper illustrates how the CDA document has been adopted to report on the health status of a GDM affected patient, and can be taken as a model to provide interoperability to other PHSs.

Keywords

Pervasive healthcare, CDA Document, Standard Vocabularies, Gestational Diabetes Mellitus (GDM), Multiagent systems

Correspondence to:

Albert Brugués
University of Applied Sciences Western Switzerland (HES-SO)
Address: Techno-Pôle 3, CH-3960, Sierre, Switzerland
E–mail: albert.brugues@hevs.ch

received: October 6, 2014
accepted: January 8, 2015
published: January 20, 2015

1 Introduction

Pervasive healthcare is a discipline involving the use of the ubiquitous computing technology in the healthcare environment [1]. As defined by Varshney [2] pervasive healthcare is the "healthcare to anyone, anytime, and anywhere by removing locational, time and other restrains while increasing both the coverage and the quality of healthcare". This new paradigm of healthcare services tries to modify the healthcare service delivery model, by moving it from a centralized approach focused on doctors to a decentralized one based on the patients [3].

The development of Pervasive Healthcare Systems (PHSs) has the potential to provide better healthcare services to an increasing number of people, and to reduce the long-term cost of the healthcare services [4]. However, when developing this kind of systems some technological challenges must be taken into account such as interoperability, scalability and security.

Interoperability is defined by the IEEE as "the ability of two or more systems or components to exchange information and to use the information that has been exchanged" [5]. However, as stated in [6] interoperability is not taken into account in most of the PHSs implementations, resulting in segmented solutions that are highly specific in nature, often known as the so called "closed" systems. These systems are not able to communicate with other components in order to support a collaborative behavior, and to achieve interoperability between different

PHSs is mandatory the use of standards such as the ones developed by the HL7 organization.

This paper focuses on the Gestational Diabetes Mellitus Management System (GDMMS) [7] is a PHS based on multiagent systems to continuously monitor women affected by Gestational Diabetes Mellitus (GDM). The multiagent platform of the system analyzes the physiological data from the patients and provides alerts when potentially dangerous situations are detected. In [8] we described how the security of the system is addressed, while in [9] we provided a study on the scalability of the system. In this work we describe how we have achieved the interoperability of this system through the compliance of the HL7 standard.

In particular, the main contribution of this paper is to show how we have used the Clinical Document Architecture (CDA) to record the monitoring status of the patient and send this healthcare-related information from the client side to the server side of the system. By doing this we extended the current functionality of the GDMMS, enabling it to collaborate with other heterogeneous systems already using the CDA standard.

1.1 GDMMS Pervasive Healthcare System

The aim of the GDMMS system is to help with the treatment of GDM, a type of diabetes affecting 3%–10% of all pregnant women due to an increased resistance to insulin. GDM can increase the risk of health problems developing in an unborn baby, so it is important that the glucose levels in the pregnant woman blood are under control.

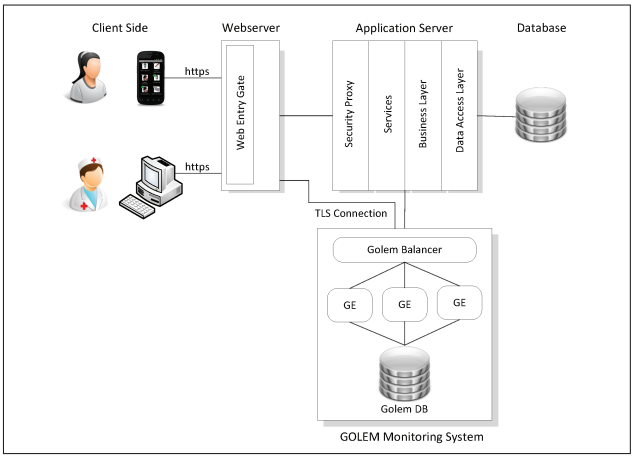

Figure 1: Architecture of the GDMMS PHS.

With GDMMS, pregnant women affected by GDM are monitored by means of using an Android smartphone, which allows them to introduce health data related with GDM such as physiological parameters, symptoms and medications. The healthcare professionals can also log on to the system to visualize and analyze data form patients. The system allows both, patients and their caretakers, being informed with historical values and receive alerts when dangerous situations are detected.

Figure 1 shows that the system is composed by five main components that are described below. The *client side* of the system can be either an Android phone, or a web browser. The former is used by patients to send health data, in terms of glucose, symptoms, blood pressure and medications taken, to the server side. The later is used by doctors to check patients' health condition. Both use Hypertext Transfer Protocol Secure (HTTPS) connections with digital certificates to exchange data.

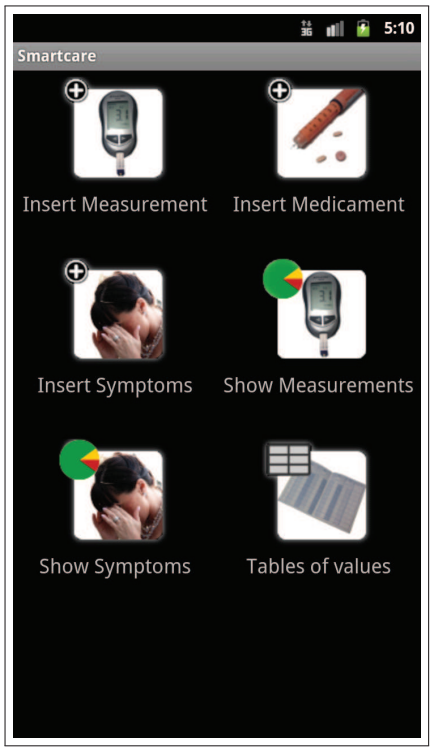

Figure 2: Main screen of the Android application.

The *Web server* layer is composed by a web entry gate which takes care about the https requests, and the certificates of the clients and a web server containing the presentation layer, which is responsible for handling the requests from the doctors and interacting with the business layer to implement the requests.

The *Application server* has four main components. The i) *security proxy* generates and sends a security token for the clients on its first request. In the following requests checks the validity of the token (i.e. it has not expired). The ii) *services* checks whether the requests have the right arguments, and forwards them to the business layer. The iii) *business layer* implements the core functionality of the system. Finally the iv) *data access layer* provide access to the data that is hosted on the database server.

The *GOLEM monitoring system* has three main components. The i) *GOLEM balancer* is a web service that receives notifications from the business layer, consisting on the health data from the patient. The ii) *GOLEM environments* are containers which contain the monitoring agents. There is one agent per patient that is in charge of generating alerts about hypoglycemia, repeated hyperglycemia and hypertensive states, by analyzing patients'

data according to a set of abductive and deductive rules. The iii) *GOLEM database* stores the inactive monitoring agents.

Finally, the *Database* stores all patients' information: demographics, health values and alerts. We refer the interested reader to [10] for more detailed description of the system.

2 Interoperability in the GDMMS

The client side of the GDMMS PHS for the patient consists on an Android application in which the patient can enter a series of health data related with GDM. Figure 2 shows the main screen of the application with the different categories that the patient can report about. Table 1 shows the elements that can be reported in each category. The application also allows a patient to check and eventually correct the data she has entered. All data are stored encrypted in the phone and sent to the server side for further processing if the phone has network connection.

Table 1: Health data related with the GDM encoded in the body of the CDA document.

Physiological Parameters	Blood pressure Heart rate Blood sugar Weight
Symptoms	Chest pain Edema Dyspnea Blurred vision Headache Epigastric pain
Medications	Insulatard Huminsulin basale Levemir Novorapid Humalog Metformin

We have encoded all the health information from Table 1 using the CDA standard to send the data from the phone to the server side in this standard format. The client side application creates a CDA document with all the unsent health values entered by the patient, and sends it to the server side when there is network connection. In order to do this we added a package to the source code containing all the necessary classes to create the CDA (Figure 3 shows a UML diagram of the classes). The CDADocument class corresponds to the clinical document intended to send, and it contains all the necessary methods to build it. In addition, each one of the rest of the classes is linked to an XML template file already formatted according to its representation in the CDA standard. When building the document all the necessary XML templates are collected, their variable values and attributes are selected using XPath expressions, and the missing values are filled with its corresponding string representation. Figure 4 shows the XML template used for a blood glucose entry, and the XPath expressions used to fill the elements of this template.

Sending the data from the client side of the system in a standard way implies changes in the server side, in order to properly handle the connections. On the other hand, a CDA document is persistent in nature an maintained by an organization entrusted with its care [11]. These facts are reflected by the change of the relational database by a XML database using BaseX [1], which allows to store the clinical document in its original format. This also implies the use of XQuery to query the information on the CDA documents. Next we explain the structure of each part of the CDA document.

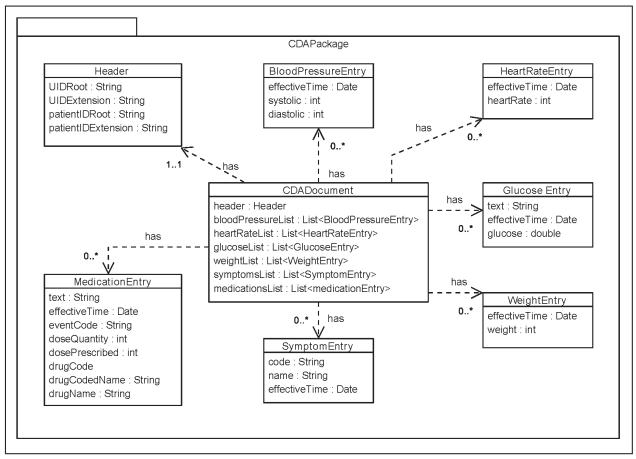

Figure 3: UML diagram of the classes used for generating the CDA document.

2.1 CDA Header

The header part of the CDA implementation contains only the mandatory elements required by the standard. The optional elements are not used in order to minimize the amount of data sent through the network interface of the mobile phone.

Every header part establishes the default context for the contents of the clinical document. The set of minimum mandatory elements to set this context includes the following:

- The identification of the document, which is defined with the `<id>` element. This XML element has two attributes: root and extension. The root attribute identifies the universe of the clinical document, and the extension provides the uniquely identification for the clinical document.

- The type of document, specified with the `<code>` element. We use the LOINC code "51855-5" with name "Patient Note", as it describes well the type of clinical document we want to generate because

[1] http://www.basex.org

[2] http://s.details.loinc.org/LOINC/51855-5.html

it states "*A patient authored note is generated by a patient, or a patient agent, acting in a non-clinical role to provide clinically relevant information*" [2].

- The creation time of the document, defined with the `<effectiveTime>` element.
- The confidentiality of the document, defined with the `<confidentialityCode>` element.
- The patient (or patients) whose document belongs to. All the patients' information is inside the `<recordTarget>` element. It can include the name, gender, address and other information, but the only mandatory field is its identification specified with an `<id>` element.
- The author of the document, can be someone or some device with the role of author. The author is defined with the `<author>` element, and its child `<assignedAuthor>`. Again, the author can be specified by providing its name, address, phone, email, but the only mandatory field is its identification specified with the `<id>` element. In our case the patient and the author of the document are the same person.
- The organization that is in charge of maintaining the document, defined by the `<custodian>` element. Although the name, address, telephone and other information about the organization can be specified, the only mandatory element is the `<id>` element.

Table 2: LOINC codes used to identify each section of the body.

Section	Code	Display name
Physiological Parameters	8716-3	Vital signs
Symptoms	10164-2	History of present illness
Medications	10160-0	History of medication

2.2 CDA Body

The body part of the CDA implementation is a XML structured body divided in three different `<component>` elements each one with one `<section>` element. Each section encodes one of the following groups: physiological parameters, symptoms, and medications. In addition, each section has `<entry>` elements encoding the medical information from the second column of the Table 1. Furthermore, the sections are identified by a LOINC code using the `<code>` element using its corresponding code and display name attributes (see Table 2). Other standard vocabularies used in the body are SNOMED CT to encode physiological parameters, ICD-10 for symptoms and ATC for medication.

2.2.1 Physiological Parameters

The section corresponding to the physiological parameters can have four different kind of entries wrapped by `<entry>` elements, each one coding a different physiological parameter. These parameters can be the blood pressure, the heart rate, the blood sugar, or the weight. All of these physiological parameters are encoded as observations using the `<observation>` element. An observation is an act which can be though as a "non-altering" procedure that results in a value [11]. In the case of this section a value is a physical quantity of a physiological parameter, although it can be virtually anything.

Table 3: SNOMED CT codes used to identify the physiological parameters.

Physio. param.	Code	Display name
Blood pressure	251076008	Cuff blood pressure
Systolic blood p.	271649006	Systolic BP
Diastolic blood p.	271650006	Diastolic BP
Heart rate	364075005	Heart rate
Blood sugar	302789003	Capillary blood glucose measurement (procedure)
Weight	363808001	Body weight measure

Each physiological parameter is identified by its corresponding SNOMED CT code using the `<code>` element (see Table 3), specifies its measure units in the unit attribute of the `<value>` element, and has associated metadata such as the time of the measurement specified in the value attribute of the `<effectiveTime>` element. Figure 5 shows the encoding of the blood pressure. The class code attribute of the `<observation>` element defines the kind of the act that is, while the mood code attribute describes its placement in time. In this case the value of the mood

```
<container xmlns="urn:hl7-org:v3" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <entry>
        <observation classCode="OBS" moodCode="EVN">
            <code code="302789003" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED CT" displayName="
                Capillary blood glucose measurement (procedure)"/>
            <text></text> <!-- XPath: //entry/observation/text -->
            <effectiveTime value="?"/> <!-- XPath: //entry/observation/effectiveTime/@value -->
            <value xsi:type="PQ" value="?" unit="mmol/L"/> <!-- XPath: //entry/observation/value/@value -->
        </observation>
    </entry>
</container>
```

Figure 4: XML template for the blood glucose entry and XPath expressions to select components.

code of all physiological parameters is "EVN" as it defines an act that has already occurred.

The encoding of the blood pressure differs from the other physiological parameters as it consists of two different parameters, the systolic and the diastolic blood pressure. This relationship is encoded with two different `<entryRelationship>` elements inside the `<observation>` element. The encoding of the heart rate also differs from the other physiological parameters as it is measured in beats per minute (bpm). This fact is expressed with a `<denominator>` element which is a child of the `<value>` element.

Table 4: ICD-10 codes used to identify the symptoms.

Symptom	Code	Display name
Chest pain	R07.4	Chest pain, unspecified
Edema	O12.0	Gestational oedema
Dyspnea	R06.0	Dyspnoea
Blurred vision	H53.8	Other visual disturbances
Headache	R51	Headache
Epigastric pain	R10.1	Pain localized to upper abdomen

2.2.2 Symptoms

The section corresponding to the symptoms can have six different kind of entries, each one coding a different symptom. The symptoms, as the physiological parameters, are encoded as observations with the `<observation>` element. The identification of the symptom is done using the ICD-10 vocabulary (see Table 4 for the codes). In each symptom entry the child `<code>` element of the `<observation>` provides the identification of the symptom. Every symptom has associated metadata corresponding to the time in which the symptom occurred. Figure 6 shows an example encoding the headache symptom.

2.2.3 Medications

The section corresponding to the medications can have six different kind of entries, each one coding a different medication. All the medications of this section are kinds of insulin as the PHS is focused on the management of the GDM. In this section the medications are encoded using the `<substanceAdministration>` element. This element is intended to represent the administration of a particular substance, e.g. a medication, immunization or other substance to a patient [11].

Table 5: ATC codes used to identify the medication.

Medication	Code	Display name
Insulatard	A10AC01	insulin (human)
Huminsulin basale	A10AD01	insulin (human)
Levemir	A10AE05	insulin detemir
Novorapid	A10AB05	insulin aspart
Humalog	A10AB04	insulin lispro
Metformin	A10BA02	metformin

Each medication is identified using its ATC code (see Table 5). In addition the `<name>` element provides the name of the medication as it appears in the mobile application. Figure 7 shows an example of the encoding of one drug. The metadata associated with the medication entry are: i) an optional comment related with the entry that the patient can write into the mobile application wrapped with the `<text>` element, ii) the time when the medication was taken encoded with two `<effectiveTime>` elements, and iii) the dose amount taken by the patient and the dose amount prescribed by the doctor, both expressed as insulin units (IU).

In the mobile application the time at which a medication was taken is specified with two elements: a time stamp, and a text specifying when was the dose taken e.g. before breakfast, after breakfast, etc. Besides, the `<substanceAdministration>` element of the CDA specifies the dose frequency with `<effectiveTime>` elements using the General Timing Specification (GTS) data type. The GTS data type allows to express complex timings as

```
<observation classCode="OBS" moodCode="EVN">
    <code code="251076008" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED CT" displayName="Cuff blood
        pressure"/>
    <effectiveTime value="201301221746"/>
    <entryRelationship typeCode="COMP">
        <observation classCode="OBS" moodCode="EVN">
            <code code="271649006" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED CT" displayName="
                Systolic BP"/>
            <value unit="mm[Hg]" value="120" xsi:type="PQ"/>
        </observation>
    </entryRelationship>
    <entryRelationship typeCode="COMP">
        <observation classCode="OBS" moodCode="EVN">
            <code code="271650006" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMED CT" displayName="
                Diastolic BP"/>
            <value unit="mm[Hg]" value="80" xsi:type="PQ"/>
        </observation>
    </entryRelationship>
</observation>
```

Figure 5: Blood pressure measurement encoded in the CDA.

a set of time intervals, using different kind of operations such as intersections, unions and differences. Thus, we specify the time of administration of a medication as the intersection of a Time Stamp (TS) data type and a Event-Related Periodic Interval of Time (EIVL) data type. The EIVL data type is used to represent events that are tied to meals and sleeping. The `<event>` element is used to specify the specific event with its code attribute. The codes which can be used are fixed by the HL7 standard. Table 6 shows the ones we have used in our application.

The `<substanceAdministration>` element only allows the codification of one dose. However, in the mobile application the patient can type both, the insulin dose prescribed by the doctor and the insulin dose really injected. This is because diabetes is a self-managed disease, so the patient has certain degree of autonomy in deciding which is the right dose of insulin she needs, as the blood sugar levels depends on the type and amount of meals taken. To encode the insulin dose taken by the patient we use the `<doseQuantity>` element defined by the standard, and we added the `<dosePrescribed>` element to encode the dose prescribed by the doctor. This is the only XML element we have defined in the whole CDA in order to be able to encode all the health-related data specified into the mobile application. The addition of locally defined XML elements is something allowed by the standard as in the Section 1.4 it states "*Locally-defined markup may be used when local semantics have no corresponding representation in the CDA specification.*"

Table 6: Event related timing codes used to identify times of the day.

Time of the day	Code	Meaning (from lat.)
Before breakfast	ACM	ante cibus matutinus
After breakfast	PCM	post cibus matutinus
Before lunch	ACD	ante cibus diurnus
After lunch	PCD	post cibus diurnus
Before dinner	ACV	ante cibus vespertinus
After dinner	PCV	post cibus vespertinus
Later	ICV	inter cibus vespertinus

The addition of the `<dosePrescribed>` element has some implications with respect the interoporability with other systems. In particular, the validation of a CDA containing extensions must be done in stages [11]. The first stage should validate the extension content, by using W3C Schema or ISO Schematron that must be provided to the rest of applications. Once the extensions are validated, these must be removed before other validations occur. This is something that can be done using a XSLT stylesheet.

3 Discussion

Interoperability has been taken into account in previous research on PHSs [12, 13, 14]. We also can find in the literature research projects which have already used the CDA standard to provide interoperability. All these systems already using the CDA can interoperate with ours by combining them in different ways. Although these integrations are not straightforward, the use of the CDA standard can minimize the efforts to achieve them.

In [15] the authors report a Home Telecare System (HTS) consisting of a patient database and a report system. The database stores parameters extracted from raw signals of vital signs, whereas the report system takes the data from the database to perform analysis on it. The report system is also in charge of generating the clinical document of the patient by first converting the information stored on the database to XML format. Whereas in this system the CDA report only contains vital signs, in GDMMS the CDA report contains information about symptoms and medications too. Furthermore, in GDMMS the use of the mobile phone allows the mobility of the patient while being monitored. Thus, the GDMMS could complement that system by extending its capabilities. To achieve that inclusion relationship the information about symptoms and medications could be sent to the database, and the report system should include these sections in the generated CDA reports.

A smart home healthcare system is presented in [16]. This system is for monitoring Alzheimer patients in their homes, while GDMMS is for monitoring GDM patients everywhere. In this smart home the data about different activities are collected through motion sensors, preprocessed using different algorithms (sensory based, video based, location tracking), and stored in XML format. Each activity includes information about type of activity, sensor information, name of the person, activity name, identification of the sensor location, and occurrence time of the activity. A CDA document is generated reporting the activities of the Alzheimer patients, which can be transmitted to all registered healthcare systems with the smart home. This system and GDMMS monitors different kind of diseases. Thus, they could have a comorbidity relationship by combining the functionalities of both systems.

In [17] the authors propose a novel framework focused on medication treatment manager, to provide safety with respect the medication by coping with Adverse Drug Events (ADEs). Their architecture is composed of two subsystems the patient site and the medical site. Patient site has a body area network with sensors measuring the blood pressure and the heart rate, and a Mobile Base Unit

```
<observation classCode="OBS" moodCode="EVN">
    <code code="R51" codeSystem="2.16.840.1.113883.6.3" codeSystemName="ICD10" displayName="Headache"/>
    <effectiveTime value="201305212115"/>
</observation>
```

Figure 6: Headache symptom encoded in the CDA.

(MBU) which coordinates the sensor network and notifies the monitoring to the medical site. The medical site is in charge to store the sensed parameters at the patient site and to send to the MBU information related with the prescription such as treatment goals in terms of monitored signs, important ADEs that may occur, and ADE detection patterns. From the medical site to the patient site the drug prescription information is encoded using an own schema, while in the reverse channel like in GDMMS the reports of the monitoring are provided by using the CDA. The functionality related with ADEs of that system could be included in the GDMMS, and complementing it with an inclusion relationship.

All the reviewed systems address the interoperability by generating CDA documents, which report the monitoring state of the patient. However, none of them provides information about how they have followed the standard nor the structure of the generated CDA reports.

On the other hand, the industry sector is more concerned with interoperability. In particular HL7 in conjunction with Continua have published the Personal Healthcare Monitoring Record (PHMR), an implementation guide for the CDA which constrains the elements of the CDA header and body. The PHMR is intended to represent measurements captured by monitoring devices such as glucometers, blood pressure cuffs, etc. to transmit the information about the measurements in a standard way. In fact this implementation guide has a required section on the sensors used for the measurements. Our proposal differs from the PHMR in the sense that is more oriented towards the patient. This is reflected in the fact that the PHMR has no symptoms section as these are subjective, and therefore can be only reported by the patient. So that we decided to use the CDA as it is more general and already comprises what the PHMR models.

4 Conclusions and Future Work

Most of the PHSs we can find in the literature do not consider the interoperability as a concern [6]. In this paper we have described how we have implemented the HL7 CDA standard in a particular PHS in order to achieve interoperability. We used the CDA to produce a clinical document which states in three different sections the physiological parameters, the symptoms and the medication taken by the patient. We believe our work can be taken as a model to provide interoperability to further implementations of PHSs.

Furthermore, we conclude that there is a need to extend the CDA specifications in order to consider patients' involvement in the monitoring process. For example, in GDMMS we have two types of insulin doses, the one taken by the patient and the one prescribed by the doctor, and in the CDA standard there is only one attribute to encode specific dosages. Thus, we added the `<dosePrescribed>` element to model the insulin dose prescribed by the doctor and we used the `<doseQuantity>` element of the standard to specify the insulin dose taken by the patient. We also assume that in a real scenario, a code for the organization owning the document should be requested to the HL7 organization in order to identify it with a unique ISO Object Identifier (OID).

As a future work we plan to move the multiagent platform from the server side to the client side of the system. Thus, the alerts generated by the monitoring agents can be encoded as a new section inside the CDA document already generated by the mobile phone.

Acknowledgments

This work has been funded by Hasler Foundation and HES-SO RCSO ISNet.

References

[1] Arnrich B, Mayora O, Bardram J, Tröster G. Pervasive Healthcare Paving the Way for a Pervasive, User-centered and Preventive Healthcare Model. Methods of Information in Medicine. 2010;49(1):67–73.

[2] Varshney U. Pervasive Healthcare and Wireless Health Monitoring. Mobile Networks and Applications. 2007;12(2-3):113–127.

[3] Bardram JE. Pervasive Healthcare as a Scientific Discipline. Methods of Information in Medicine. 2008;47(3):178–185.

[4] Varshney U. Pervasive Healthcare. IEEE Computer. 2003;36(12):138–140.

```
<substanceAdministration classCode="SBADM" moodCode="EVN">
    <text>optional comment related with the entry</text>
    <effectiveTime xsi:type="TS" value="201305211250"/>
    <effectiveTime xsi:type="EIVL" operator="A">
        <event code="ACD"/>
    </effectiveTime>
    <doseQuantity value="2.5" unit="IU"/>
    <dosePrescribed value="2" unit="IU"/>
    <consumable>
        <manufacturedProduct>
            <manufacturedLabeledDrug>
                <code code="A10AE05" codeSystem="2.16.840.1.113883.6.73" codeSystemName="ATC" displayName="insulin
                    detemir"/>
                <name>Levemir</name>
            </manufacturedLabeledDrug>
      </manufacturedProduct>
    </consumable>
</substanceAdministration>
```

Figure 7: Levemir medication encoded in the CDA.

[5] Institute of Electrical and Electronics Engineers. IEEE Standard Computer Dictionary. A Compilation of IEEE Standard Computer Glossaries; 1991.

[6] Baumgarten M, Mulvenna MD. Cognitive Sensor Networks: Towards Self-adapting Ambient Intelligence for Pervasive Healthcare. In: Proc. 5th International Conference on Pervasive Computing Technologies for Healthcare (PervasiveHealth); 2011. p. 366–369.

[7] Schumann R, Bromuri S, Krampf J, Schumacher MI. Agent Based Monitoring of Gestational Diabetes Mellitus (Demonstration). In: Proc. 11th International Conference on Autonomous Agents and Multiagent Systems - Volume 3 (AAMAS); 2012. p. 1487–1488.

[8] Bromuri S, Krampf J, Schumann R, Schumacher M. Enforcing Security in Pervasive Healthcare Monitoring Gestational Diabetes Mellitus. In: Proc. 4th International Conference on eHealth, Telemedicine, and Social Medicine (eTELEMED); 2012. p. 221–226.

[9] Krampf J, Bromuri S, Schumacher M, Ruiz J. An Agent Based Pervasive Healthcare System: A First Scalability Study. In: Electronic Healthcare. vol. 91 of Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering. Springer Berlin Heidelberg; 2012. p. 128–137.

[10] Bromuri S, Schumacher MI, Stathis K, Ruiz J. Monitoring Gestational Diabetes Mellitus with Cognitive Agents and Agent Environments. In: Proc. 2011 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT); 2011. p. 409–414.

[11] Boone KW. The CDA ™Book. Springer; 2011.

[12] Baird S, Dawson-Haggerty S, Myung D, Gaynor M, Welsh M, Moulton S. Communicating Data from Wireless Sensor Networks Using the HL7v3 Standard. In: Proc. 2006 International Workshop on Wearable and Implantable Body Sensor Networks (BSN); 2006. p. 4 pp.–186.

[13] Nikolidakis SA, Georgakakis E, Giotsas V, Vergados DD, Douligeris C. A Secure Ubiquitous Healthcare System Based on IMS and the HL7 Standards. In: Proc. 3rd International Conference on PErvasive Technologies Related to Assistive Environments (PETRA). ACM; 2010. p. 42:1–42:7.

[14] Warren S, Lebak J, Yao J, Creekmore J, Milenkovic A, Jovanov E. Interoperability and Security in Wireless Body Area Network Infrastructures. In: Proc. 27th IEEE-EMBS International Conference of the Engineering in Medicine and Biology Society (EMBC); 2005. p. 3837–3840.

[15] Garsden H, Basilakis J, Celler BG, Huynh K, Lovell NH. A Home Health Monitoring System Including Intelligent Reporting and Alerts. In: Proc. 26th IEEE-EMBS International Conference of the Engineering in Medicine and Biology Society (EMBC). vol. 2; 2004. p. 3151–3154.

[16] Khan WA, Hussain M, Khattak AM, Afzal M, Amin B, Lee S. Integration of HL7 Compliant Smart Home Healthcare System and HMIS. In: Impact Analysis of Solutions for Chronic Disease Prevention and Management. vol. 7251 of Lecture Notes in Computer Science. Springer Berlin Heidelberg; 2012. p. 230–233.

[17] Koutkias V, Chouvarda I, Triantafyllidis A, Malousi A, Giaglis GD, Maglaveras N. A Personalized Framework for Medication Treatment Management in Chronic Care. IEEE Transactions on Information Technology in Biomedicine. 2010;14(2):464–472.