Using Archetypes and Domain Specific Languages on Development of Ubiquitous Applications to Pervasive Healthcare

Anderson L. Menezes, Carlos E. Cirilo, João L. C. de Moraes, Wanderley L. de Souza, Antonio F. do Prado

Computer Department (CD) – Federal University of São Carlos (UFSCar)
Postal Box 676 – 13565-905 – São Carlos – SP - Brazil
{anderson_menezes, carlos_cirilo, joao_moraes, desouza, prado}@dc.ufscar.br

Abstract

Pervasive healthcare focuses on the use of new technologies, tools, and services, in order to help patients to play a more active role in the treatment of their diseases. Since pervasive healthcare environments demand a huge amount of information exchange, the use of technologies like Health Level Seven (HL7) and archetypes has been proposed to provide interoperability between applications for these environments. However, the complexity of such technologies difficulties their full adoption as well as the migration from centralized healthcare environments into pervasive ones. Aiming at collaborating to bridge this gap, this paper proposes an approach to integrate archetypes into HL7 v3 messages for the development of pervasive healthcare applications. The approach suggests the use of Domain Specific Languages (DSLs), which simplify the HL7 messages modeling and allow to automate most of the messages schema codification.

1. Introduction

Currently, most countries face many problems related to healthcare, such as expensive care and low quality of health services. These problems arose due to the population increase and the insufficiency of available health professionals. This scenario has worsened mainly because of the elderly population expansion, the chronic diseases increase, and the growing demand for new treatments and technologies. For instance, in the United States the number of graduated doctors was 15,632 in 1981 and 15,712 in 2001 (increasing 0.5%), while in the same period its population rose from 225 million into 281 million (increasing 24%) [1].

The centralized healthcare models, which are still prevailing in many countries, cannot deal with these problems anymore, since they are centered on highly specialized people in big hospitals, and focused only on acute treatment cases. As consequence, distributed healthcare models have been proposed in order to allow patients to manage their own health and to decrease the caregivers’ overload. Pervasive healthcare [2] aims to enable these distributed models by employing Information and Communication Technologies (ICTs) to allow healthcare available everywhere, at anytime, and for everyone.

The main goal of Ubiquitous Computing is to allow the users to access information from anywhere, at anytime, employing any kind of device [3]. Ubiquitous Computing environments in communities, homes, and hospitals, can be extremely useful to build a pervasive healthcare model, and for this model to be patient-centered it is necessary a fast, and safe information exchange among the caregivers.

To exchange health information among the heterogeneous applications immersed in a ubiquitous environment, which supports the pervasive healthcare, it is required the use of a communication standards for enabling interoperability. Health Level Seven (HL7) [5] is an international standard for exchanging messages among heterogeneous Health Information Systems (HISs). Although the use of HL7 is widespread in healthcare environments, it has some limitations for representing clinical knowledge, and presents a high development complexity.

Archetypes [4] are structures that indicate how represent the clinical knowledge specified by experts of a specific domain, and have been employed to organize semantically interoperable information. Besides being able to be shared and reused, archetypes can be packed into HL7 messages, which in turn can be processed and interpreted by any system that employs this protocol. This way one can ensure efficiency in the exchange of clinical data and flexibility in the representation of clinical concepts.

This paper proposes an approach for the development of pervasive healthcare applications, which integrates archetypes into HL7 v3 messages by using Domain Specific Languages (DSLs) [6]. The use of a healthcare DSL allows to model the HL7 messages with archetypes and to automate most of their schema generation, simplifying the development of such applications. The remainder of this paper is structured as follows: section 2 introduces the main standards and technologies related to the proposed approach; section 3 presents the proposed approach illustrated with a case study; section 4 shows the evaluation of the application developed using the proposed approach; section 5 discuss some related work; and section 6 presents concluding remarks and further work.

2. Standards and technologies

The main standards and technologies used by the approach are related to HL7 standard, Archetypes and Domain Specific Languages.
2.1. Health Level Seven

The HL7 committee was funded in the United States in 1987 aiming at the development of communications standards to exchange health information among the various existing HISs. HL7 was gradually broadcast to many countries, and in June 2009 the Technical Committee (TC) 215 of the International Organization of Standardization (ISO) established the HL7 v2.5 as an application protocol for the electronic exchange of data in healthcare environments [7].

Although being a robust and widespread standard, the HL7 v2.x family has limitations mainly related to the clinical concepts representation and to the functional interoperability [8]. Aiming to solve these shortfalls, the HL7 v3 family was developed based on the Object Oriented (OO) paradigm [5]. This new family uses a Reference Information Model (RIM), developed based on the Object Oriented (OO) paradigm [5].

This new family uses a Reference Information Model (RIM), which provides an explicit representation of the relationships among the objects belonging to the healthcare domain. However, this advantage has a major drawback: several refinements of the RIM are needed for achieving a description of a HL7 v3 message.

An overview of HL7 v3 message development process is presented in Fig. 1. This process consists essentially of four main activities: Specify Use Case; Specify Interaction Model; Specify Information Model; and Create Message Description.

![Fig. 1. Overview of HL7 v3 message development process](image)

In Specify Use Case activity, the application requirements are identified (e.g., patient management, financial administration) and represented through use case diagram. In the Specify Interaction Model activity, the use case diagrams are detailed on sequence diagrams, which describe the information flow, the communication rules and the messages trigger events.

In the Specify Information Model activity, the RIM is refined based on the use cases and sequence diagrams specifications. According to these use cases, it is extracted a HL7 RIM subset called Message Information Model (MIM), which is then refined considering the same use cases and their corresponding sequence diagrams. This last refinement generates a model called Refined-Message Information Model (R-MIM). In R-MIM the unnecessary attributes are eliminated, cardinalities are added to the relationships and class specializations are performed. Finally, in Create Message Description activity, a HL7 message description is elaborated according to the R-MIM specifications. This description defines a suitable HL7 structure to meet the requirements of the target application domain.

Even with its reference model and a well-defined process for structuring messages, the semantic representation of clinical concepts in HL7 v3 still demands a hard work. A clinical concept must be expressed in the HL7 RIM through the combination of different generic elements (e.g., classes, auxiliary classes, attributes, relationships) with specific vocabulary codifications. This sparse fragmentation of clinical concepts among various HL7 RIM elements, and the use of different codes to give semantic meaning to the messages overload the HL7 v3 message development. Furthermore, there is a lack of HL7 v3 tools to support this entire development process.

2.2. Archetypes

The strategy adopted by the HL7 committee for representing clinical concepts has been questioned. The knowledge structure provided by the HL7 RIM may be limited to templates, which sometimes is a very simple solution for the knowledge representation at conceptual level. Due to its static nature, templates do not allow to represent socio-cultural variations (e.g., labels, measurement units, language) in the concepts, which is called Variability Problem [4].

To minimize this problem, the use of archetypes has been proposed [4]. Archetypes are pieces of knowledge that indicate how to represent concepts or information of a given domain through computable expressions. These expressions are based on a reference information model and defined in the form of structural constraints [10]. As in LEGO, in which an isolated piece has no meaning, but the combination of several pieces does, archetypes also try to express a concept through combination of different information in an integrated manner.

2.3. Domain Specific Languages

A DSL is a language designed to be useful to a specific set of tasks within a given domain [6][11]. It can be defined by a metamodel, which represents the domain knowledge of the target problem. Restricted to a specific domain, DSLs are usually small, consisting of a set of abstractions and notations, which are closed to real terms known by the experts of this domain [12]. Thus, DSLs express solutions in language and abstraction level of problem domain, reducing the translation efforts of the concepts of this domain into the solution domain [13].
The use of DSLs on applications modeling, instead of general-purpose modeling languages (e.g., Unified Modeling Language – UML), allows the creation of more specific and complete models. Thus, resources like frameworks, design patterns and components, can be included to models, creating an infrastructure that allows the execution of Model-to-Code (M2C) transformations\(^1\), to generate a greater amount of code from the modeling [14].

### 3. Proposed Approach

The approach employs a DSL which express the healthcare domain knowledge to integrate archetypes into HL7 messages, to create communication modules for applications to pervasive healthcare.

Using the Structured Analysis and Design Technique (SADT) diagram notation [15], Fig. 2 shows an overview of the approach, which consists of two stages: Domain Engineering (DE) and Application Engineering (AE). In DE a DSL, represented by an healthcare domain metamodel, and M2C transformations, to messages schema generation, are built. In AE ubiquitous applications are developed, through the reuse of the artifacts developed in the DE stage. The metamodel is used to support the application modeling, and the M2C transformations are employed to generate the code related to the schemas of the HL7 messages with archetype.

![Fig. 2. Overview of the proposed approach](image)

#### 3.1. Domain Engineering

The DE stage is divided into four main activities as described below.

In **Specify Domain Metamodel** activity, the healthcare domain requirements are elicited, specified, analyzed and represented in a metamodel that expresses the knowledge of this domain. In Eclipse Integrated Development Environment (IDE), the Domain Engineer, guided by the HL7 RIM, specifies the metamodel using the Eclipse Modeling Framework (EMF) [16]. The HL7 RIM represents the healthcare domain information at a high abstraction level, which allows the use of different terminology in concept description of this domain [17]. Fig. 3 shows part of the developed metamodel, whose metaclasses, metaattributes and metarelationships were created from the RIM class diagram.

In **Design Domain Metamodel** activity, the metamodel specification is refined according to standards, technologies and platforms that enable the metamodel construction. The Domain Engineer, using the EMF, adds elements to the metamodel to identify the main clinical concepts of the target domain. The approach is guided by medical terminology coding, such as the Systematized Nomenclature of Medicine – Clinical Terms (SNOMED-CT) [18]. SNOMED-CT is a scheme for identifying clinical terms and concepts, being used by healthcare applications to support semantic interoperability among them. The addition of elements that address the terminology coding to metamodel allows to uniquely identify the clinical concepts represented (e.g., blood pressure, blood glucose, body temperature). For example, Fig. 4 shows part of the metamodel with the Archetype enumeration inclusion, which lists the concepts “bpArchetype” and “glucArchetype”, which correspond respectively to blood pressure and blood glucose concentration concepts defined by SNOMED-CT.

![Fig. 3. HL7 RIM metamodel](image)

![Fig. 4. Metamodel with concepts](image)

In **Implement Domain Metamodel** activity the implementation is done based on the metamodel design specification. Supported by EMF framework, the Domain Engineer automatically generates the Java code of the metamodel and of a model editor. This editor will assist during the application model specification, in the AE stage. The model editor, together with the metamodel, are available as a plugin in Eclipse IDE, allowing that the metamodel is instantiated during AE to modeling the application. Once the metamodel is based on RIM, the models created with the DSL expressed by this metamodel, also correspond to

\(^1\) Transformations consist of a set of rules that specify how an input model should be mapped into another output model or source code.
refinement that should be performed to create the HL7 messages.

In Build Model to Code Transformations activity the transformations for code generation are built with the aid of Java Emitter Templates (JET) framework [19], which allows the creation of M2C transformation.

The transformations interpret the models created using the implemented healthcare domain metamodel, generating the appropriate schemas of the messages that transport information about the entities specified in the models. Since the models also correspond to RIM’s refinement, the mapping of these models for HL7 messages into transformations is easily accomplished, since the terms and concepts expressed in models are close to those used to structure the messages.

3.2. Application Engineering

The AE stage includes the disciplines of Analysis, Design, Implementation and Testing, which are known by software development process.

The use of the metamodel to application modeling in AE, besides facilitating the application development for this domain, allows mapping the defined process for the HL7 v3 messages development (Fig. 1) into the traditional activities of Software Engineering. Thus, the RIM refinement can be performed directly on models that specify the application requirements. Moreover, the M2C transformations automate much of the Application Engineer tasks associated to HL7 messages organization and their integration to archetypes. The healthcare metamodel also allows the reuse of domain knowledge in different projects during the AE.

To investigate the feasibility of the proposed approach, a case study was developed by building a ubiquitous application in the Electronic Health Record (EHR) domain. This application allows hypertensive patients to record their blood pressure data on their mobile device. In short intervals, these records are automatically transmitted to a server, so that caregivers can analyze them from a desktop or smartphone and quickly act if needed.

This application consists of three parts: the first one, which runs on the patient’s mobile device, records and persists the pressure data reported by the patient, as well as transmits them to a remote server through HL7 messages with the archetypes that defines the blood pressure concept [20]; the second one, which runs on the server, receives these messages, handles the archetype-based data contained therein and persists them in a database; the last one, also executed on the server, provides a Web interface that allows the caregivers to view the pressure data of their patients. The AE activities are detailed below, presenting examples based on application described as case study.

In the Analysis activity the application is specified from its requirements, which are elicited, specified, analyzed and verified. For the specification, the Application Engineer uses UML techniques such as class, sequence and use cases diagrams. An example is the use case diagram shown in Fig. 5, which specifies the requirements to pressure data recording the submission of these data formatted in HL7 v3 messages with archetypes to server and the access to these data by cardiologist.

In Design activity the application specification is refined with the technologies and hardware and software platforms which enable to implement the application, such as Java Micro Edition (Java ME). Based on the use case diagrams, the Application Engineer also performs in this activity the modeling of terms and concepts of the target problem domain as an instance of the metamodel built in DE, selecting the classes and attributes relevant to application domain.

Fig. 6 shows, on left, the tree view message model built using the metamodel definitions through the plug-in available in Eclipse IDE. On the right there is the object diagram illustrating the instantiation of concepts from the healthcare metamodel. This model describes the information associated to the application domain for blood pressure records exchanging. As depicted in the use case diagram in Fig. 5, the patient records his blood pressure in his mobile device. This record is translated into an Observation in the message model. The association between the patient and his corresponding blood pressure record is indicated by a Participation element. This latter comprises a Person and an Organization, which describe respectively the patient’s attributes and the health institution where the patient receives medical care. Moreover, there is also in the model a second Participation associated to a Qualified Entity that represents the caregiver responsible for the patient’s care.
By adopting the coding terminology used on metamodel (SNOMED-CT), the archetypes that define the clinical concepts represented in the messages models are combined with their corresponding code. For example, in the model of Fig. 6, in the code attribute of the Observation element it has been assigned the SNOMED-CT code that identifies the blood pressure clinical concept. Moreover, in its value field it has been assigned the archetype-based data that correspond to the blood pressure concept, as proposed in [9].

In Implementation and Testing activity, the application is implemented with its communication module. In Eclipse IDE, the Application Engineer assisted by the JET framework, performs the M2C transformations to generate the structure of the messages that will be exchanged by the applications to transport the health data. After the partial schema generation, the Application Engineer performs the required supplements. Moreover, the other components that comprising the application such as user interfaces, data persistence, and the logic to handle the messages’ content, are also developed. At the end of implementation, tests are performed, providing the feedback that indicate whether is necessary or not to return to the previous activities of AE.

4. Application Evaluation

Among the motivations for the integration of archetypes into HL7 messages is to meet the real needs of healthcare applications’ end users by providing them with agility and efficiency in health data exchanging. Aiming at assessing the feasibility of the proposed approach to produce applications that meet these characteristics, a prototype of the developed application has been presented to twenty-six hypertensive patients and five cardiologists. As evaluation approach, the TAM model [21] was employed. TAM allows to evaluate the perceived usefulness and easiness of use from users’ viewpoint, considering the effects of external variables, such as application characteristics, development process, training and usage intent, being the most used model for this [22].

The developed application has been deployed on the cell phones of the patients, who received instructions regarding the application functionalities, so that they could hold the pressure data record in their devices. Using digital devices, the patients measured and recorded their systolic and diastolic pressure for five consecutive days. The blood pressure archetype-based data encapsulated in HL7 messages were automatically sent from patients’ cell phone to the server, so that cardiologists could analyze them through the Web interface.

After the application usage period, TAM-based evaluation forms were filled out by the patients and cardiologists. In these forms the users specified their agreement level with a particular statement about the evaluated application based on the Likert scale [23]. The results of this evaluation are summarized in Table 1. For each statement the patients’ evaluation is at the top of each table cell, whereas the doctors’ evaluation is at the bottom.

The favorable results regarding the perceived usefulness of the application (statements 1 and 2) and the easiness of use (statements 3 and 4) reflect, in large part, the employment of the HL7 standard for communication and archetypes for representation of the involved clinical concepts. The combined use of these backend technologies has enabled to streamline information exchanging between patients and doctors, as well as to meet the information needs in an efficient and interoperable way, indirectly impacting on users’ satisfaction.

5. Related Work

Due mainly to the complexity existing in HL7 v3 messages development process and the lack of support tools that cover it in its entirety, various proposals to facilitate the use of the standard can be found in literature.

In [24] it is discussed the implementation of a modeling tool for HL7 v3 messages developed from Eclipse IDE and EMF framework. In order to handle the artifacts, the developed tool allows the use of both a graphical editor and a text editor based on a tree schema, through which the model are serialized in the XML Metadata Interchange (XMI) standard for later use.

Works [25] and [26] present, respectively, an architecture and a discussion about the implementation of a software factory focused in healthcare domain. The goal on developing the software factory was to automate the creation of communication interfaces, called “collaboration ports”, strictly following the HL7 v3 messages development process. In [27] a solution is presented to assist developers in handling messages from HL7 v2.x family, which are still widely used worldwide. The authors’ proposal is based on the use of a programmatic DSL to, among other features, support the HL7 v2.x messages creation, transmission, interpretation and validation.

<table>
<thead>
<tr>
<th>Statements</th>
<th>Level of Agreement</th>
<th>Fully Agree</th>
<th>Partially Agree</th>
<th>Neither Agree nor Disagree</th>
<th>Partially Disagree</th>
<th>Fully Disagree</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. It is useful send/receive data through this technology.</td>
<td>22 (84.6%)</td>
<td>3 (11.5%)</td>
<td>1 (3.9%)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. The use of this technology can streamline the doctor-patient relationship on decision-making.</td>
<td>24 (92.3%)</td>
<td>3 (60%)</td>
<td>2 (40%)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. The use of this technology interfered on daily activities.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. It was difficult to record/access data.</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

399
The approach proposed in this paper is based on several characteristics of the work described above. However, it has its own contributions through evolution and adaptation of conceptions of the related work. Among the contributions of this work, there is the use of archetype to represent clinical concepts, which are integrated with HL7 messages. Moreover, the approach proposes the use of encoding terminology, such as SNOMED-CT, which facilitates the identification of clinical terms carried in the messages exchanged between the applications. A DSL is also used based on a healthcare domain metamodel, which enables modeling and automatic schema generation to the communication modules of the applications in that domain.

6. Conclusions and Future Work

This paper presented an approach for ubiquitous application development to pervasive healthcare, which integrates archetypes to HL7 messages via DSLs. The integration of the archetypes to HL7 messages supports agility and efficiency in communication between heterogeneous HISs, once the communication becomes interoperable and clinical concepts can be easily identified. DSLs use allows the creation of an infrastructure for message schema generation, and their reuse in different application of target domain, which reduces the applications development efforts.

A ubiquitous application in the EHR domain has been developed using the proposed approach to investigate its feasibility. The application has been presented to a group of qualified users and evaluated by them in relation to its usefulness and perceived easiness. The evaluation aimed at verifying how striking is the use of archetypes and HL7 messages to the users’ satisfaction, obtaining positive results. The favorable results demonstrate the potential of the proposal to develop health applications that address the real needs of users in a pervasive healthcare environment.

This work is still continued with research and case studies through the development of other applications in order to improve the approach, such as: definition of a systematic testing activity in order to maximize the defects correction on the messages schemas developed; improvements on the healthcare metamodel to provide a greater variety of archetypes so that applications to control other Chronic Non-communicable Diseases (CNDs), like Diabetes, can be developed; and optimizations in the M2C transformations that allow further automation of messages schemas generation.

7. References