

Evaluation of a Framework to Implement Electronic Health Record Systems Based on the openEHR Standard

Diego A. Orellana¹, Alberto A. Salas¹, Pablo F. Solarz¹, Luis Medina Ruiz¹ and Viviana I. Rotger²

¹ Instituto de Bioelectrónica, Fac. de Medicina, Universidad Nacional de Tucumán, Argentina

² Departamento de Bioingeniería, Fac. de Cs Exactas y Tecnología, Universidad Nacional de Tucumán, Argentina

E-mail: pablosolarz@gmail.com

Abstract. The production of clinical information about each patient is constantly increasing, and it is noteworthy that the information is created in different formats and at diverse points of care, resulting in fragmented, incomplete, inaccurate and isolated, health information. The use of health information technology has been promoted as having a decisive impact to improve the efficiency, cost-effectiveness, quality and safety of medical care delivery. However in developing countries the utilization of health information technology is insufficient and lacking of standards among other situations. In the present work we evaluate the framework EHRGen, based on the openEHR standard, as mean to reach generation and availability of patient centred information. The framework has been evaluated through the provided tools for final users, that is, without intervention of computer experts. It makes easier to adopt the openEHR ideas and provides an open source basis with a set of services, although some limitations in its current state conspire against interoperability and usability. However, despite the described limitations respect to usability and semantic interoperability, EHRGen is, at least regionally, a considerable step toward EHR adoption and interoperability, so that it should be supported from academic and administrative institutions.

1. Introduction

The production of clinical information about each patient, coming from complementary studies, reports, derivations, admissions, diagnostics, symptoms, etc. is constantly increasing, and it is predictable the advance of such situation because science and technology contributions on healthcare. It is noteworthy that the information is created in different formats and at diverse points of care.

The use of health information technology (HIT) has been promoted as having a decisive impact to improve the efficiency, cost-effectiveness, quality and safety of medical care delivery, by making wider and faster access to information, enhancing decision support systems, making easier access to follow care with specialists, more reliable prescribing, etc.

Particularly, researchers have examined the benefits of EHRs (Electronic Health Record Systems) by considering clinical, organizational, and societal outcomes which include improvements in quality of care, reduction in medical errors, and other improvements in patient-level measures that describe the appropriateness of care [1]. However in developing countries the utilization of HIT is insufficient and lacking of standards, among other situations, having as consequences that health information is



fragmented, incomplete, inaccurate, and isolated. This leads to information silos, so that the information contained inside them cannot be used for patient care or data analysis [2]

There are some sparse efforts in Latin America (LA) to promote and to adopt standards improving conditions for an increasing adoption of e-health solutions and for interoperability as in Brazil and Uruguay from public health authorities, Chile and Argentina through private and public actors [3]. The efforts are scarcer when solutions are patient centered. Solutions based on standards such as RIM HL7 [4] and openEHR [5] are in that way, but the construction of service layers are complex and expensive, and it is necessary highly trained human resources. For such service layer we are looking for open source solutions based on standards. In this sense, the framework EHRGen [6] it is perhaps a considerable advance and promised to go in the right direction.

In the present work we evaluate the framework EHRGen, based on the openEHR standard, and having in mind its use in the public health sector, as mean to reach generation and availability of patient centered information. The evaluation refers to the services provided to develop EHR systems.

The remainder of this paper is organized as follows: In section 2, we describe the essential of the openEHR standard. In section 3 we describe the EHRGen framework. In section 4 we present the clinical domains selected to evaluate EHRGen and the evaluation results, and finally, in section 5, we present the conclusions and some proposals.

2. The openEHR standard

Two of the most important problems related to healthcare information systems are evolution and interoperability. Software engineering has addressed the first with modern agile lifecycles, multilayer architectures and abstract designs. However, in highly dynamic context such as healthcare, that approaches are not enough, probably because, the changes in medicine in three aspects: 1) breadth, due to new information is being discovered, 2) depth, finer grained details are discovered and became relevant, 3) complexity, because new relationships are added among concepts [7]. The second problem, interoperability, has been addressed by standardized communication protocols, thesaurus and more recently, reference models of information [4][5][8]. However, communication protocols lack of sufficient semantic richness to handle interrelated information that has to be transmitted as a set, thesaurus only address specific terms and the reference models became very complex when implementing domain concepts, as evidenced by their low use by applications [9].

An alternative to address those problems, particularly evolution and interoperability, is to separate information and knowledge into two levels of model. The former is most familiar to developers - the level of software object models and database schemas, denoted here by the term reference model (RM) - and is used to build information systems. Concepts such as role, address, party, observation, evaluation, etc. are in the RM. It must be small in size, in order to be comprehensible, and contain only non-volatile concepts in order to be maintainable. The second level is the knowledge level (KM), requiring its own formalism(s) and structure, and it is where the numerous, volatile concepts of most domains are expressed [10]. Concepts such as Patient, Medical doctor, Blood Pressure, Lipid studies, obstetric summary, etc. are in the KM.

The standard openEHR implements that two level approach. KM is built on domain concepts such as: patient, hospital, pressure, body weight, etc. all expressed using constraints on instance structures of an underlying RM. Such constraints are expressed in a model called archetype model.

Persistence and instance data are in the RM, and changes in KM (from now Archetype Model – AM-) does not affect the RM. Changes in AM mean create, update or delete concepts made as archetypes. Figure 1 shows RM and AM as layers of the software architecture, which are specified by the standard. It is added a service model layer (SM), which represents the software layer of processing, presentation, configuration tools, etc.

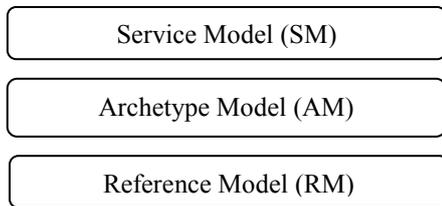


Figure 1. RM and AM are specified by openEHR standard. SM represents the services added by the application.

The figure 2 (took from [11]) shows the relationship among models and user roles. On the left, an expert on information technology creates an information layer by instantiating the RM in a particular implementation platform. On the right, a domain expert creates archetypes and templates. Archetypes are made with references to RM classes. Templates are particular configurations on archetypes, they permits to reuse archetypes for application in different use cases, specialties, etc., by grouping, predefining values, etc. Medical terminologies are added to use controlled vocabularies. Archetype Definition Language (ADL) is the specific language to create archetypes.

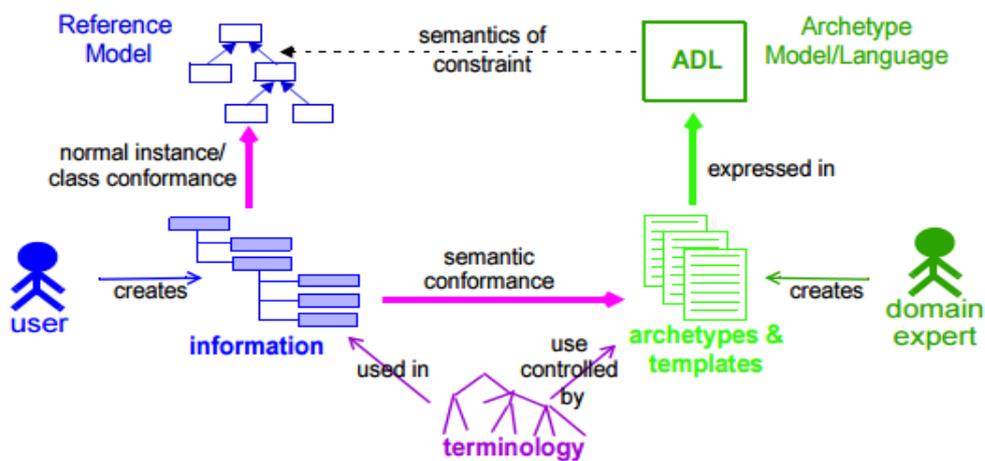


Figure 2. Architecture elements in relation to user roles.

From the functional point of view, the most important parts of the RM, are the demographic RM and the EHR RM. As it is shown in left side of Figure 3 (took from [12]) classes of demographic model are: Roles, Agent, Person, Organization, Address, etc. They are the base for demographic archetypes as Patient, Healthcare Provider, Healthcare consumer, Payer, etc.

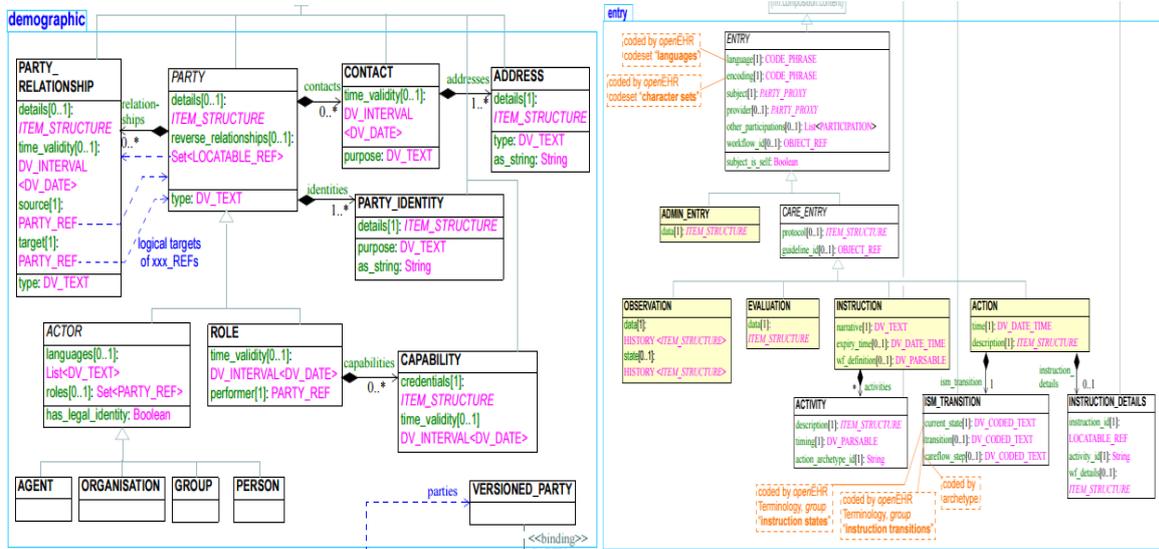


Figure 3. Left: Class diagram of the demographic part of RM. Right: Class diagram of the EHR part of RM.

Figure 3 on right side (took from [13]) shows, in UML (Unified Modeling Language), part of the EHR RM, from the entry package, with the classes Observation, Evaluation, Instruction and Action which are subtypes of an Entry of a clinical record, and corresponds to stages of a hypothetical clinical process. They are the base for the Entry archetypes as Blood Pressure, Lipid studies, obstetric summary, etc.

The concepts in RM: Roles, Agent, Person, Organization, Address, Entry, Observation, Evaluation, Instruction, Action, are made up with abstract attributes: DV_TEXT, DV_DATE, DV_CODED_TEXT, ITEM_STRUCTURE, etc. which are independent of languages and platforms. It is expected they are stable although domain concepts evolve.

Domain concepts, represented as archetypes, are more unstable and follow the dynamic of medical concept as described in section 1. In fig 4 we show, in ADL language, part of the definition of the archetype Temperature, which is an Observation. Some of the attributes of the composing temperature are, as can be seen, ELEMENT and C_DV_QUANTITY. The definition is usually managed from a tool in the service layer with a more amenable interface.

```

ELEMENT[at0004] matches {-- Temperature
  value matches {
    C_DV_QUANTITY <
      property = <[openehr::127]>
      list = <
        ["1"] = <
          units = <"°C">
          >
        ["2"] = <
          units = <"°F">
          precision = <|1|>
    >
  }

```

Figure 4. Part of the definition of an archetype

Interoperability in openEHR must be at RM and AM layers to achieve interoperability both at semantic level (corresponding to AM) and at syntactic level (corresponding to RM). Applications which will exchange information should use a set of agreed archetypes made on a RM basis. To this

end there is a repository of archetypes called Clinical Knowledge Manager (CKM) at [14], which is an international, online clinical knowledge resource with a set of reusable archetypes. Templates however do not need to be agreed because they are related to each particular application and they are not the clinical concepts to be exchanged.

3. EHRGen

EHRGen[6] is an open code framework that allows the creation of EHRs oriented by clinical knowledge management which implements the openEHR standard. As such, EHRGen produces an information model, which implements the RM. The KM is implemented with archetypes which can be downloaded from the international CKM, or developed with framework such as founded in [15]. Both archetypes, took from CKM and created *ad hoc*, are made by clinical domain experts and define one part of the Knowledge Base (KB) of the EHRs. The archetypes created may be reused in different contexts. In addition, KB has terminologies which constrain and specifies the clinical language to reach understandability and to contribute to semantic interoperability. Templates are component of the KB too and can be implemented by EHRGen. Templates allow, on one hand to aggregate archetypes according to a particular need, and on the other hand to filter part of archetypes.

The architecture of EHRGen [16] is based on the implementation of the information repository under the RM of openEHR standard and the implementation of the repository for the KB. The components of the EHRGen architecture are shown in the fig 5.

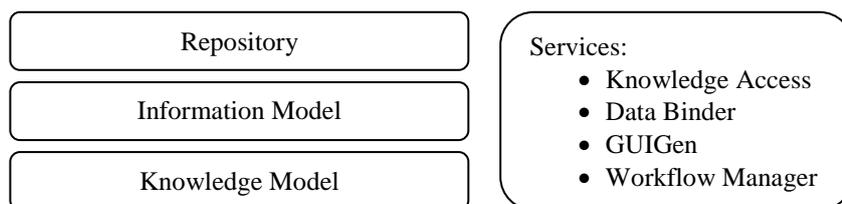


Figure 5. EHRGen architecture.

Knowledge Access (KA) provides access to the KB, Information Model (IM) implements the class model of the openEHR standard which is related to the repository and the documents by means of object-relational mappings, while Data Binder creates IM structures and validates data coming from the user interface. Workflow Manager assists the user in the process to create, update and view patient information. GuiGen allows to automate the interface component creation. There are other components of the EHRGen architecture not considered here because they do not participate in our evaluation.

The software technologies with which EHRGen was made up are: JDK6, Grails 1.37. For the repository the options are MySQL or PostgreSQL.

The methodology to create an electronic health record system, proposed by EHRGen developers [16], consists of the following processes:

1. Base modeling: to create (or to select) the archetypes which will represent the clinical concepts.
2. Templates modeling: to create (or to select) the templates by selecting and composing the archetypes that made up the clinical records according to each particular clinical domain.
3. Creation of workflows: to define user access interfaces and stages according to the different healthcare profiles.
4. Creation of domains: a domain allows us to manage (and to reuse) a group of templates according to a clinical domain.
5. User interface generation: EHRGen support automatic generation in HTML format, according to templates and the GuiGen components configuration.

6. The production or use of the software system: that is, creation, updating and visualization of clinical records.

EHRGen does not provide tools for the first process. For creating and editing archetypes we can use the Archetype Editor de Ocean Informatics [15]. The second, third, fourth y sixth processes are carried out with the EHRGen framework and are described and evaluated in the next section. The fifth process is fulfilled practically without user intervention because the framework creates the HTML interfaces with preconfigured components and the templates definitions.

4. Evaluation of EHRGen

In this section we show an evaluation of the framework EHRGen described previously. The evaluation refers to the process of creation of EHR Systems. To that end we have considered two clinical domains as perspectives: 1) A form to track diabetic patients elaborated by the public healthcare authority of Argentina, 2) An emergency medical history from a public hospital from the province of Tucumán, Argentina.

The first domain contains data such as: Physical Examination (Weight, Height, Head Circumference, Abdominal girth, feet exam, cardiac frequency, respiratory frequency), Laboratory test results (blood glucose, cholesterol, HDL, LDL, creatinine, etc.), Education to patient (states of the education such as: taught, assimilated, accomplished, etc.), Current Medication, Smoking habits, Explorations (ECG, dental exam, eyes exam, etc.), Prescribed medication, etc.

The second domain has the following data: chief complaint, personal and family history, physical exam, patient values (cardiac frequency, respiratory frequency, axillary temperature, blood pressure, laboratory test results, oxygen saturation, Glasgow scale), fulfilled procedures, consultation request, prescribed medication ,etc.

Both domains share some data but have some different data too. Thus, we pretend to create two perspectives on a unique patient centered database.

The elicitation tasks were relative to the objective of carry the paper forms to computer user interface, thus there were considered only issues about term interpretations, content of fields, etc., but not respect to changes in format or content. Both clinical forms were considered for respective template designs. Previously we have analyzed the archetypes that should be to composite the templates.

By following the processes indicated in the previous section, we begin specifying archetypes. For space reasons we synthesize the process with few cases which represents all found situations.

Many of the data components of both forms, diabetes and emergency, there exist in archetypes in the CKM, for example: Physical exam (called openEHR-EHR-OBSERVATION.exam.v1), Laboratory test results (in various archetypes called: openEHR-EHR-OBSERVATION.lab_test-blood_glucose.v1, openEHR-EHR-OBSERVATION.lab_test-blood_gases.v1, openEHR-EHR-OBSERVATION.lab_test-esr.v1, etc.), weight (openEHR-EHR-OBSERVATION.body_weight.v1), temperature (openEHR-EHR-OBSERVATION.body_temperature.v1), Blood pressure (openEHR-EHR-OBSERVATION.blood_pressure.v1), family history (openEHR-EHR-EVALUATION.family_history.v1), Glasgow scale (openEHR-EHR-OBSERVATION.glasgow_coma_scale.v1), procedures (openEHR-EHR-ACTION.procedure.v0), education (openEHR-EHR-ACTION.health_education.v1), etc.

Data not included in CKM archetypes, must be included in new archetypes according to clinical concepts specified by domain experts. There are only few archetypes in such a sense. Two of them:

- Patient history (or patient antecedents): according to domain experts contains only a free text field.
- Patient destination: Describes the immediately death, derivation, inpatient or outpatient condition.

The created archetypes are added to the local KB.

The next step is to design the templates. A template is generated with the interface shown in figure 6. It has an identification, a name and the archetypes that the template must contain.



Figure 6. Template creation interface.

As templates are directly related with the user interfaces, their design must into account, besides archetypes, the custom grouping of data in medical history. Some selected groups are: Patient history, Physical exam, Laboratory. The first one corresponds directly with the previously *ad hoc* designed archetype patient_history. Physical exam has an archetype in the CKM, described as “Use to record details about findings on physical examination of the subject of care. This may include a narrative description of the findings, a framework in which to nest detailed CLUSTER examination archetypes, and a clinical interpretation of the findings”. Thus, the cluster may contain other archetypes as: blood pressure, weight, etc., then we may to generate a template with such archetype. But it is not possible to use archetypes with cluster as data type, at least until version OpenEHRGen v0.8 beta1

An alternative way is to generate a template with the archetypes: blood pressure, weight, etc., but that is not possible either, because the template creator tool of the framework allows incorporating only one archetype. That is, there not be possible with the framework tools to create templates with more than one archetype.

A third alternative design is to create one template per archetype. In that manner we could to use the CKM archetypes and to develop an interoperable system both at syntactic level because RM, and at semantic level because AM. But in this last case we found two problems related with the usability of the system. To create one template for each archetype means that final user of system will have one screen per archetype, i.e. one screen for blood pressure, one for height, one for weight, one for glucose, etc. transforming the current use of a unique preprinted form in tens of screens. The second problem is related to the fact that definitions of archetypes are the result of consensus from various point of views, thus the use of all parameters in each particular application would be an unnecessary overcharge. In the table 1 it is showed the elements of the blood pressure archetype. They are 14 parameters, some of this are only used in rare applications. In majority of application, especially in emergency, it is only necessary diastolic and systolic values, and other values are unavailable or unnecessary.

Table 1: data components of the CKM blood pressure

Blood Pressure	
Data	Systolic Diastolic Mean Arterial Pressure Pulse Pressure Comment
State	Position Confounding factors Exertion Sleep status Tilt
Protocol	Cuff size Location
Events	Any event 24 hour average

An alternative solution, which was our choice, is to create an archetype with the data elements required for the template. That archetype has the following data: weight, height, blood pressure, abdominal girth, feet exam, cardiac frequency, respiratory frequency. Then we have created the template shown in fig 7, on that archetype basis.

Figure 7. Template for physical exam in diabetes domain.

The methodology, consisting in developing *ad hoc* archetypes, because the limitation of one archetype per template of the templates creation tool, was generalized for both diabetes and emergency domains.

In fig 8, on left, we show the main screen of the diabetes domain, and on right, the emergency domain. In both, the list below “REGISTRO CLÍNICO” (Clinical registry) corresponds to the templates (and archetypes) designed.

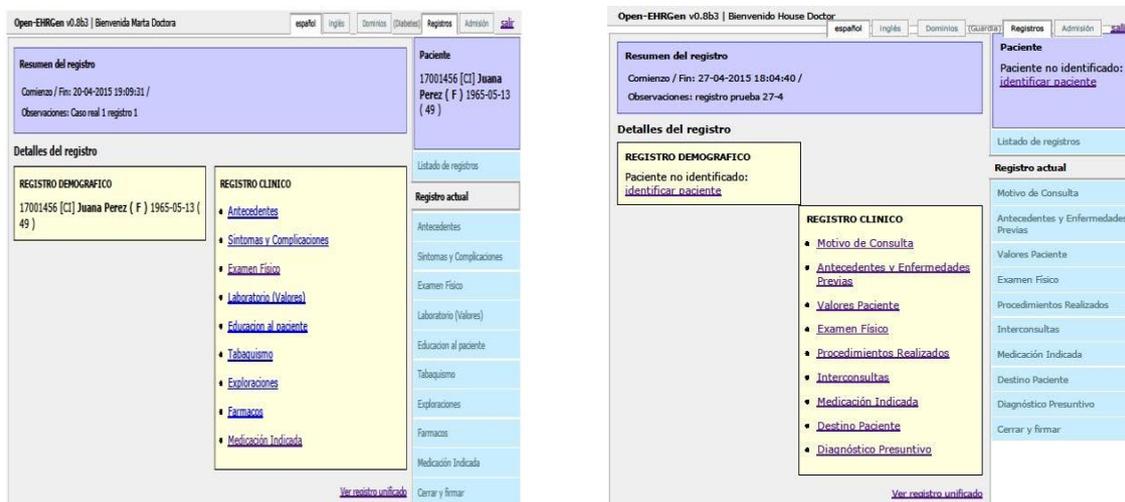


Figure 8. Main screens of domains diabetes (left) and emergency (right).

The diabetes and emergency domains, share the RM, because all archetypes are designed based on its elements. However, although there is much information shared by those domains, and particularly by the designed archetypes and templates, the use of *ad hoc* archetypes entails interoperability problems at knowledge level, because they are not shareable with systems operating with CKM archetypes. Moreover, the archetypes *physical_exam_diabetes* and *physical_exam_emergency* are slightly different, for example, the *feet_exam* element there exist only in the first and the *Glasgow scale* element there exist only in the last, thus, the data components of shared data are interoperable, but semantic of data are not. For example, there is not defined semantic to interpret which both archetypes contains temperature, because it is not a concept in a knowledge common model; however the values, comments and measurement units are shared in RM.

A different situation occurs with the archetypes Patient and Prescribed Medication, they are the same in both domains, thus although the templates would be different, concepts in both domains can be shared.

5. Conclusions

We have evaluated a framework to generate EHRs. The EHRs generation process includes: software creation artifacts, data repositories and user interfaces. The framework also allows to create the necessary workflows for the development of record editing tasks and it puts in the stage for speaking a tool for electronic medical records based on a standard. The framework was evaluated through the provided tools for final users, that is, without intervention of computer programmers or experimented users about documents with configurable metadata. To evaluate the framework we have used two clinical domains as perspectives: 1) a form to track diabetic patients, 2) an emergency medical history. They have both some common and some different data elements, and they have some concepts involved in both domains which are implemented in CKM, while some are not.

The framework allows solving EHRs generation based on requirements such as: knowledge about the fundamental concepts of the openEHR standard, CKM and local knowledge databases access, technical skill respect to the framework use.

The standard openEHR proclaim the reuse of concepts already implemented. Thus, concepts such as temperature, weight, etc., should be incorporated to the designed grouping of domain concepts, called

templates, and should not be built from scratch. But the current state of the framework imposes some limitations over that idea. 1) Archetypes with CLUSTER as datatype are not allowed, 2) templates cannot incorporate more than one archetype and 3) archetypes must be incorporated at all without being possible to filter unnecessary data. Those limitations have made necessary the creation of ad hoc archetypes, even when many components are already in CKM, breaking down semantic interoperability even between our test domains: diabetes and emergency. Moreover, many designed archetypes are no sense as clinical concepts because they are a group of very different concepts sharing particular use cases, as is the case of laboratory values. An option would have been a lot of user interfaces. However, the reference model remains unmodified, allowing data interchange, although with a considerable effort. In addition, we consider that the current state of automatic user interface creation conspire against usability.

It is clear that despite the described limitations respect to semantic interoperability, EHRGen is, at least regionally, a considerable step toward EHR adoption and interoperability. It makes easier to adopt the openEHR ideas and provide an open source basis with a set of services.

We consider that EHRGen should be turned into a framework which receives contributions from other groups and the interest from public health institutions, having in mind that it is a considerable progress aimed at addressing the problems of dispersion, incompleteness, lack of standardization that characterize health information in Latin America.

References

- [1] Nir Menachemi and Taleah H Collum. *Benefits and drawbacks of electronic health record systems*. Risk Manag Healthc Policy. 2011; 4: 47–55. Doi: <http://dx.doi.org/10.2147/RMHP.S12985>
- [2] Daniel Luna, Alfredo Almerares, John Charles Mayan, Fernán González Bernaldo de Quirós, Carlos Otero. *Health Informatics in Developing Countries: Going beyond Pilot Practices to Sustainable Implementations: A Review of the Current Challenges*. Healthc Inform Res. 2014 Jan;20(1):3-10.
- [3] Beatriz de Faria Leão. Electronic Health Record Initiatives in South America. 11th International HL7 Interoperability Conference - IHIC2010 May 14 - 15, 2010. RJ, Brasil
- [4] Reference Information Model – Health Level 7. RIM HL7. <http://www.hl7.org/implement/standards/rim.cfm>
- [5] openEHR. <http://openehr.org/>
- [6] <https://code.google.com/p/open-ehr-gen-framework/>
- [7] Rector, A. L. Clinical Terminology: Why Is It So Hard? Yearbook of Medical Informatics 2001.
- [8] Unified Medical Language System UMLS. <http://www.nlm.nih.gov/research/umls/>
- [9] Barry Smith and Werner Ceusters. *HL7 RIM: An Incoherent Standard*. Studies in Health Technology and Informatics, vol. 124 (2006), 133–138
- [10] Beale T. *Archetypes: Constraint-based Domain Models for Future-proof Information Systems*. Eleventh OOPSLA Workshop on Behavioral Semantics: Serving the Customer (Seattle, Washington, USA, November 4, 2002).
- [11] openEHR architecture overview. Release 1.0.1. <http://openehr.org/releases/trunk/architecture>.
- [12] openEHR architecture demographic reference model. Release 1.0.1. http://openehr.org/releases/trunk/architecture/rm/demographic_im.pdf
- [13] openEHR architecture electronic health record reference model. Release 1.0.1. http://openehr.org/releases/trunk/architecture/rm/ehr_im.pdf
- [14] Clinical Knowledge Manager CKM. <http://openehr.org/ckm/#>
- [15] Ocean Archetype Editor. <http://ocean-archetype-editor.software.informer.com/>
- [16] Pazos P. EHRGen: Generador de Sistemas Normalizados de Historia Clínica Electrónica Basados en openEHR. 41 JAIIO - CAIS 2012 - ISSN: 1853-1881. Pp. 212-228.