journal homepage: www.intl.elsevierhealth.com/journals/cmpb

An HL7-CDA wrapper for facilitating semantic interoperability to rule-based Clinical Decision Support Systems

Carlos Sáez[*], *Adrián Bresó, Javier Vicente, Montserrat Robles, Juan Miguel García-Gómez*

Grupo de Informática Biomédica (IBIME), Instituto de Aplicaciones de las Tecnologías de la Información y de las Comunicaciones Avanzadas (ITACA), Universitat Politècnica de València, Camino de Vera s/n, 46022 València, Spain

ARTICLE INFO

Article history:
Received 5 December 2011
Received in revised form
1 October 2012
Accepted 2 October 2012

Keywords:
Clinical Decision Support Systems
Semantic interoperability
Reuse
HL7
CDA
Rule-based
Electronic Health Records

ABSTRACT

The success of Clinical Decision Support Systems (CDSS) greatly depends on its capability of being integrated in Health Information Systems (HIS). Several proposals have been published up to date to permit CDSS gathering patient data from HIS. Some base the CDSS data input on the HL7 reference model, however, they are tailored to specific CDSS or clinical guidelines technologies, or do not focus on standardizing the CDSS resultant knowledge. We propose a solution for facilitating semantic interoperability to rule-based CDSS focusing on standardized input and output documents conforming an HL7-CDA wrapper. We define the HL7-CDA restrictions in a HL7-CDA implementation guide. Patient data and rule inference results are mapped respectively to and from the CDSS by means of a binding method based on an XML binding file. As an independent clinical document, the results of a CDSS can present clinical and legal validity. The proposed solution is being applied in a CDSS for providing patient-specific recommendations for the care management of outpatients with diabetes mellitus.

1. Introduction

One of the key points for the success of a Clinical Decision Support Systems (CDSS) is its capability of being integrated in a Health Information Systems (HIS). An integrated CDSS is able to provide their predictions or recommendations based on patients' Electronic Health Record (EHR). Non-integrated CDSS may require an additional data entry, a workflow issue that can lead to failure the acceptance and effectiveness of a CDSS [1,2]. A successful integration of a CDSS results by adapting solutions to particular systems, e.g. by building ad hoc accesses to local relational databases or EHR. This solution, however, may remain unscalable in the sense that it would require the effort of performing a specific integration for each different HIS.

Abbreviations: API, application program interface; CDA, Clinical Document Architecture; CDSS, Clinical Decision Support Systems; EHR, Electronic Health Record; HIS, Health Information Systems; HL7, Health Level Seven; ICD, International Statistical Classification of Diseases and Related Health Problems; RIM, Reference Information Model; R-MIM, Refined Message Information Model; SNOMED-CT, systematized nomenclature of medicine-clinical terms; vMR, Virtual Medical Record; XML, Extensible Markup Language.

[*] *Corresponding author.* Tel.: +34 963 877 000x75278.
E-mail address: carsaesi@ibime.upv.es (C. Sáez).

0169-2607/$ – see front matter © 2012 Elsevier Ireland Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.cmpb.2012.10.003

Nowadays, the clinical data sharing is based on three major pillars: terminology, definition of concepts and reference model [3]. Document standards for EHR such as Health Level Seven (HL7) Clinical Document Architecture (CDA)[1] or CEN/ISO 13606[2] provide both a syntactic and a first degree of a semantic basis for the interoperability of clinical concepts between health computer systems, including HIS and CDSSs. These are complementary to standardized clinical terminologies, such as SNOMED-CT, LOINC or ICD, which complete the semantics for the accurate identification of specific clinical concepts or properties. Such semantic interoperability is particularly useful for CDSSs, not only because it permits to uniquely identify the concepts employed on their inferences [4], but also because it facilitates the deployment of CDSSs on different HIS configurations.

Another important feature for the success of a CDSS is its capability of maintaining its knowledge base up-to-date. Machine-learning based CDSSs, classified as Type Four CDSS according to the Australian National Electronic Taskforce Report 2000 [5], are able to update their knowledge dynamically incorporating new predictive models [6] or employing incremental learning methods [7]. On the other hand, CDSSs based on deductive inference engines or rule-based, mostly Type Three, base their knowledge in a knowledge-base, generally consisting in a set of rules defined by a committee of experts. These rules have to be understood by humans if they are meant to be maintained [4]. Consequently, rule-based CDSSs must simultaneously (1) 'understand' the clinical information required as input facts—i.e. map such information to rule antecedents –, (2) return the set of deducted conclusions in a form that can be semantically understood by machines and/or humans, and (3) provide a human-readable knowledge-base so that it can be easily maintained.

In this work, we propose a technical solution for facilitating the semantic interoperability for the reuse of rule-based CDSSs in different Health Information Systems by using an HL7-CDA—on its level 3—input/output wrapper template—based in an HL7-CDA implementation guide—combined with a binding configuration to map CDA data with rule facts and vice-versa. This approach has been applied to a CDSS for providing patient-specific recommendations for the care management of outpatients with diabetes mellitus.

The rest of the paper is organized as follows. Section 2 describes the required background on HL7-CDA, clinical terminologies, and rule-based CDSSs, as well as the current state-of-the-art of semantic interoperability of CDSSs. Section 3 describes the technical design of the proposed solution. Section 4 gives an overview of the case of study on a diabetes mellitus CDSS for continuous care. Finally, Section 5 summarizes our conclusions and proposes future lines of work.

2. Background

This section is divided into two parts. First, we describe the required technical knowledge following an example scenario. Next we describe current state-of-the-art in semantic interoperability of CDSSs.

2.1. *Technical background*

In order to review the required technical background of the solution, suppose a scenario where clinical documents are exchanged between two entities. On one hand, syntactic interoperability allows each part understanding the structure of the documents, i.e. where each piece of information is located in the document. On the other hand, the semantic component allows understanding the meaning of such information, i.e. what does some information mean. Thus, when semantic interoperability is permitted, the understanding by each entity of both the structure and the meaning of exchanged documents is enabled. In order to achieve such semantic interoperability in Health Information Systems, series of standards must be used [8].

HL7-CDA on its release 2 [9] is a standard for the structure and exchange of clinical documents. CDA is approved by ANSI and widely accepted as clinical documents standard, e.g. the European Health Project epSOS employs the CDA standard for the expression of the data elements used in the project.[3] Also, the Spanish Ministry of Health and Consumer Affairs decided to use CDA level 1 (document header) for the exchange of documents among national health services.[4]

A document conforming to CDA, i.e. a CDA document, is an Extensible Markup Language (XML) document following the standard defined by the CDA Refined Message Information Model (R-MIM)—in form of XML schema—, which is formed by components from the HL7 Reference Information Model (RIM). In a specific scenario, in order to allow involved individuals understanding the final structure of a CDA document, users must define 'HL7-CDA implementation guides', text documents which serve as scenario-standard by restricting the R-MIM to the specific scenario. At this point, HL7-CDA can facilitate the entities in our scenario with syntactic and partial semantic interoperability.

HL7 contains its own vocabulary which provide a first degree of semantics. However, in most cases it must be completed with clinical terminologies in order to provide the required semantics for a particular scenario. SNOMED-CT is currently one of the most extended clinical terminologies worldwide. It contains uniquely coded and, in general, unambiguous clinical concepts from most health care domains. Concepts are related at different levels of granularity and

[1] HL7 Clinical Document Architecture, http://www.hl7.org/implement/standards/product_brief.cfm?product_id=7 (last accessed September 12, 2012).

[2] CEN/TC 251 EN13606—Medical Informatics—Electronic Health Record communication.

[3] European Patients Smart Open Services, D3.5.2 Semantic Services Definition, http://www.epsos.eu/uploads/tx_epsosfileshare/D3.5.2_epSOS_Semantic-Services-Definition_01.pdf (last accessed September 12, 2012).

[4] Ministerio de Sanidad y Consumo de Espa na, Política de Estándares y Normalización de Datos, http://www.msps.es/organizacion/sns/planCalidadSNS/docs/POL_EST.pdf (last accessed September 12, 2012).

can be post-coordinated to express more complex statements. SNOMED-CT concepts can be referenced in CDA documents using the CDA data types known as ‘coded values’. Coded values in CDA must contain at least a code system and a code. The code system attribute identifies the table, terminology, ontology, etc. where the coded concept comes from. Hence, the code attribute identifies the unique code of the clinical concept in the terminology specified in the code system attribute. One CDA document may contain several concepts coming from different terminologies. In addition, a user can define custom terminology tables—associating them a code system—for specific terms. The only step left to make our scenario semantically interoperable is to include in the previous HL7 implementation guide the restrictions about where to use what terminology, restricted or not to some set of terms.

The purpose of CDA documents is to serve as a standard for the exchange of clinical documents, which should be understood both by computers and humans. The CDA narrative block is a human readable—based in HTML—representation of the structured contents of the CDA document. This block is signed by the document creator providing the document with legal validity. In some cases CDSS may be considered as a clinical test and, thus, their outputs and inputs should be legal and persisted.

Different HIS may define equivalent clinical concepts by different expressions or terms. E.g. a CDSS may take a specific granularity level of a term, different from the specific level in the HIS. Defining standard input and output documents for the CDSS, as the solution proposed in this paper, partially overcomes this issue. However, in order to approach to complete semantic interoperability and to thoroughly exploit the terminological knowledge, the aforementioned abstractions should be controlled by a external component mapping the HIS to the input document as well as the output document to the HIS, what is out of the scope of this work.

Rule-based CDSS started being used in the 1970s, as systems using custom rules engines [10,11]. Later, in the 1980s other systems were developed based on specific programming languages for the rules-inference, such as Oncocin [12], CDSS for oncology protocol management based on the Lisp programming language. Up-to-date, many other inference engines and logic-programming languages have been used in the development of rule-based CDSS, such as CLIPS, PROLOG, Jess, and SWRL [13]. Most of these technologies use their own language for representing the knowledge-base, as well as for defining the input facts and the resulting conclusions. Others provide a specific language for clinical rules, such as the Arden Syntax. However, these languages share a set of features which are related to the interoperability solution proposed in this work:

- Rule, fact and variable names should be unambiguous textual names representing a unique concept in the knowledge-base. This is to facilitate the reading of the knowledge-base since it must be easily readable by humans in order to be maintained.
- A clinical binomy variable-value can be viewed as a fact. The input data of an inference consists of a set of facts representing the set of initial facts of the fact base or working memory. Facts can be formed by names and values, and in some cases a fact can be formed by a structure—e.g. in Jess a structured fact can be composed by slots. We can therefore view a clinical variable as a fact whose name corresponds to the clinical concept related to the variable and whose value corresponds to the value of the variable—numerical, categorical, boolean, etc.
- Rules are formed by an antecedent, representing the conditions that must be accomplished to fire the rule, and a consequent, representing the conclusions derived from the rule. It is straightforward that one could chain a set of executed rules in order to link initial facts to final conclusions/facts or vice-versa—depending on whether we deal with a forward or a backward chaining problem.
- Inference conclusions can be viewed as clinical outcomes of a decision process. Final conclusions are as well represented as facts.

A rule-based CDSS based on one of these inference engines takes the role of an entity in our semantically interoperable scenario. The inference engine should, therefore, understand the information contained in received documents as well as return documents that should be understood by the entity who required the decision support. Section 3 describes our proposed solution to accomplish this task.

2.2. *State of the art*

Some approaches on interoperable CDSS following the HL7-CDA standard have been proposed up to date. In 2006, Komulainen et al. [14] described a particular solution for an evidence-based decision support service where patient data were retrieved in HL7-CDA format and decisions were based in a specific decision support model relying on guidelines written in Javascript. In 2008, Nee et al. described the SAPHIRE project [15], an intelligent healthcare monitoring architecture for cardiac patients, where the retrieval of patients’ data was solved using the IHE cross-enterprise document sharing (XDS) being then transformed into HL7-CDA for its exploitation by the CDSS. Such exploitation was based in a specific extension of the GLIF3 (Guideline Interchange Format) clinical guideline model which permits semantically retrieving data from HL7-CDA documents, as was later detailed by Laleci and Dogac [16]. More recently, in 2010 Kazezmadez et al. described in [17] a very complete approach for clinical data and knowledge interoperability for GLIF3 guideline-based CDSSs. In their approach, specific data mining models—which knowledge was coded in PMML (Predictive Model Markup Language)—can be accessed from guideline decision nodes by sending patient data and receiving decision results in HL7-CDA documents.

Previously, in 2003 an object-oriented query and expression language for CDSS named GELLO was published [18]. GELLO provides expressions to retrieve clinical data from documents based in the Virtual Medical Record (vMR) [19], an HL7 standard for providing a set of data elements for the input and output of CDSS, still under development at the writing time. This approach is therefore limited to the use of the GELLO language for rules definition. Despite current tools for the use of this language are limited, recent results [20] have provided a GELLO execution engine capable of transforming CDA documents into vMR. Relating this works to our solution, we can state that our proposal could take use of vMR in the future for the semantic interoperability of rule-based CDSS and their knowledge base, as well as in the definition of its outputs documents. A direct future line of work is, therefore, adapting the current solution to vMR elements.

As far as we are concerned, only Kazezmadez and Sartipi work, provides some kind of results from predictive models in HL7-CDA format. Relying on XPATH for accessing any kind of data contents both in input and output documents, their interoperability solution appears totally generic, that is, it does not fit any particular type of predictive model, thus a new model would require a completely new parser to build its output in CDA format. Therefore, our proposed solution could complement their method for the case of rule-based decision models, since it offers an standardized output for this type of models. In addition, and on the contrary as we do in this work, no references were provided in any of the previous cited works regarding to the narrative block, mandatory on CDA documents, and to the legal value it adds to generated CDSS outputs. We, therefore, emphasize the fact that the results of a CDSS can be considered as a unit, or individual document, with clinical and legal validity, just as other clinical test.

We should mention the OpenCDS collaborative project. Such project aims to develop open source, standard-based CDSS tools and resources [21], which include infrastructures for enabling interoperability using the HL7 vMR. Relying on Apelon DTS terminology server and Drools DSL technologies, their solution could be applied to rule-based CDSS based in the Drools business rules language. Our solution differs from theirs in the sense that we attempt to provide a general method for any type of rule-based CDSS, being lightweight using a simple but powerful mapping file, as well as providing a multi-language solution for creating human-readable and legally valid CDSS outputs.

Our work attempts to propose a simple but generic solution that can be used in most types of rule-based CDSS, since it is a non-invasive solution based just on binding standardized data to facts and vice-versa by means of a specific knowledge binding and language files. In addition, we have written an HL7-CDA implementation guide to formally describe the contents of the proposed HL7-CDA documents. This guide is completed with custom terminologies to complete the semantics required by structure and contents. Furthermore, the proposed binding method permits describing the knowledge-base using human readable terms instead of terminology codes, what facilitates the maintenance of the knowledge.

3. Proposed solution

Our approach to facilitate semantic interoperability to rule-based CDSSs consists on syntactically and semantically relate the inference-engine knowledge-base—i.e. the set of rules—to standardized HL7-CDA input and output documents via a binding-definition file. The output of the CDSS is generated following the proposed HL7-CDA template for results of rule-based inferences. Such approach is conceptually represented in Fig. 1. We will describe each component following the flow of the CDSS usage.

On the left side of the figure we can see the input and output clinical documents wrapping the CDSS. These correspond to HL7-CDA documents. An HL7-CDA implementation guide is provided as the standard template for the output documents as well as a recommended template for the input documents. In the center of Fig. 1 we can see the binding layer of the method. Its objectives are (1) reading the input CDA document and transforming it into a set of input facts compatible with the inference engine and (2) obtaining the results of the inference rules and transforming them into the output CDA document using the corresponding language terms for textual values. Finally, the right side of the figure represents the inference engine containing its knowledge-base. The rest of the section explains in detail each of these parts following the line of a CDSS use.

3.1. *CDA input document*

In order to facilitate the interoperability of the CDSS, the architecture of the CDA documents—i.e. the restrictions over the CDA R-MIM—and the corresponding terminologies—i.e. the clinical concepts permitted to be used in them—were defined in a custom HL7-CDA implementation guide. Input documents may arrive from different Health Information Systems with different types of clinical documents (e.g. non-HL7-CDA such as CEN/ISO 13606 or openEHR), or even no clinical documents may be available at all in the defect of a relational database. However, defining a standard input for the CDSS is a scalable method that avoid the coupling between the CDSS and the HIS. Thus, only an external mapping of the non-HL7-CDA HIS data to this document would be needed[5] (what overcomes the problem of concept abstraction, as introduced in Section 2). Moreover, our solution also allows accessing data from other types of XML-based foreign documents by means of XPATH expressions—XQUERY may also be used—to locate data on specific document locations, thus, allowing the reuse of the knowledge-base in other clinical settings just by modifying this binding file.

Basically, each input data is represented by a CDA entry containing a clinical statement corresponding to the clinical concept related to such data (Fig. 2). At the moment, we have identified as the most relevant clinical statements observations, procedures and substance administrations. The code assigned to the content of statements provides the semantics, assigning unique code in the specified clinical terminology. Observation should be the most prevalent statement in an evidence based CDSS input, since they represent conditions observed or measured on the patient. As defined in the CDA R-MIM, the value of observations can correspond to any data type. CDSSs can deal with most types of data, therefore in order to let our CDSS understand such types, we have added parsing and transformation procedures for physical quantities (PQ), booleans (BL), integers (INT), reals (REAL), coded values (CV) and ratios of physical quantities (RTO_PQ_PQ). The proposed method will, therefore, iterate over each entry and

[5] Several tools exist that enormously facilitate such task, such as LinkEHR [22,23], and others not so related to medical informatics could be employed such as Altova MapForce or Microsoft BizTalk.

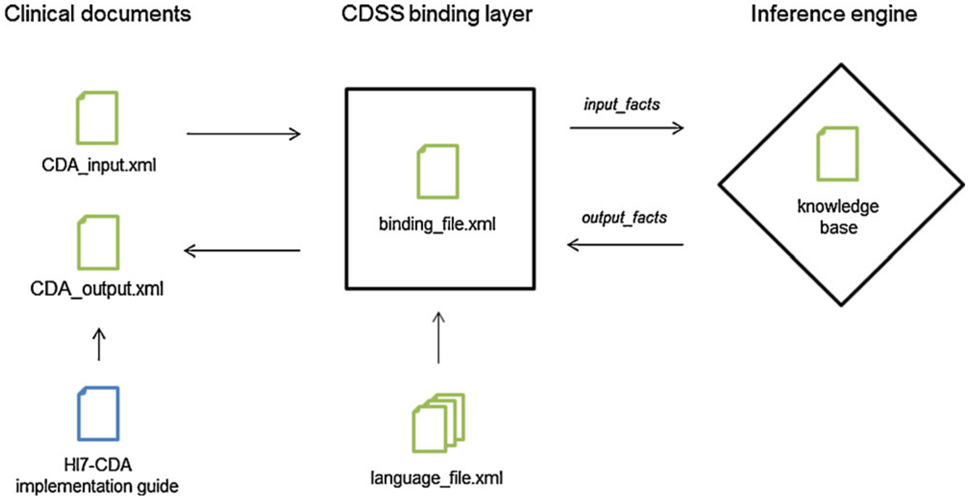
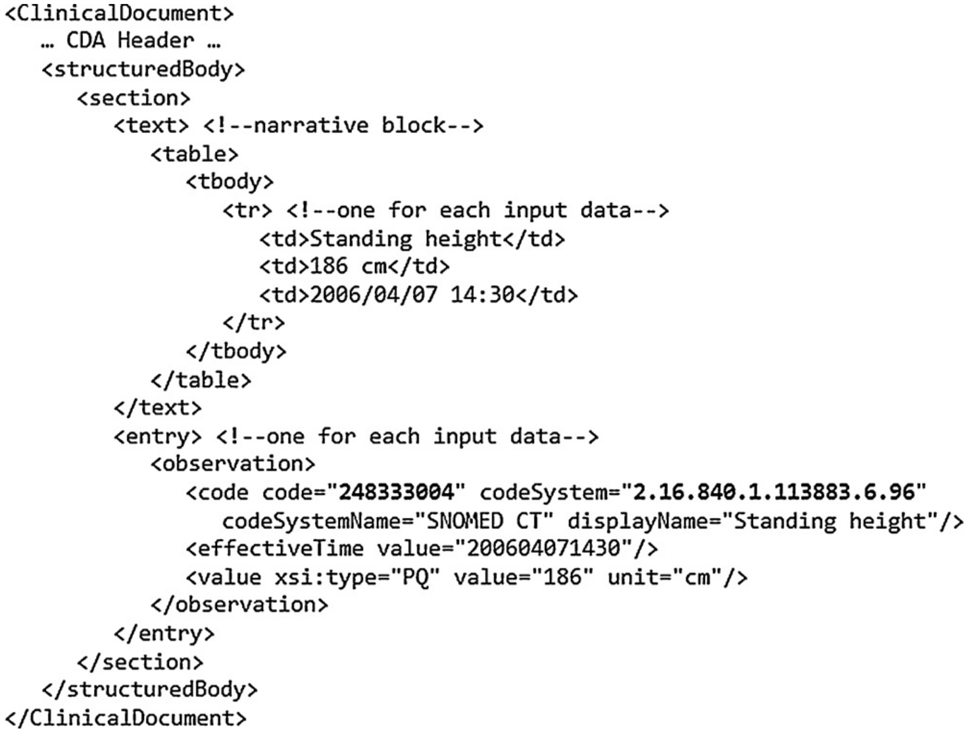


Fig. 1 – Conceptual schema of the proposed solution. Data flow between the three main components and from external inputs is represented by arrows.

transform its content into the corresponding input fact for the inference engine.

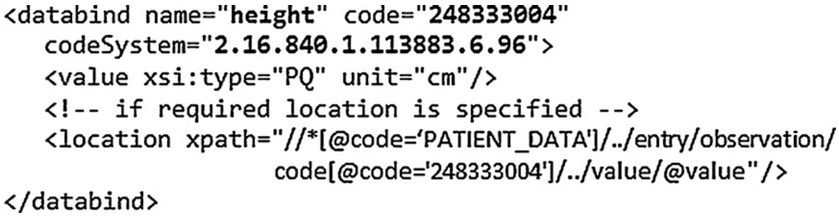
3.2. *Binding from CDA input document to input facts*

In the center of Fig. 1 we can see the binding layer of the method. The first objective of this component is reading the input CDA document and transforming it into a set of input facts compatible with the inference engine. A fact representing a clinical concept is compatible with the inference engine if the rules related to such clinical concept are able to match this fact. We previously stated that a maintainable knowledge base must define rules in a manner that they are easily understood by humans. A good approach for this is assigning unambiguous textual names to fact names. In addition, rules manage fact values supposing a given physical unit or set of possible values, on the contrary conclusions may be incorrect. This layer, therefore, matches each entry of the document to a specific fact by means of matching the union of the code and code system of the clinical concept to the textual name employed in the knowledge base. In the same process physical units or value validity are checked in order to accept a valid input document. Improvements on this procedure such as automatic unit conversion are proposed as



```
<ClinicalDocument>
   … CDA Header …
   <structuredBody>
      <section>
         <text> <!--narrative block-->
            <table>
               <tbody>
                  <tr> <!--one for each input data-->
                     <td>Standing height</td>
                     <td>186 cm</td>
                     <td>2006/04/07 14:30</td>
                  </tr>
               </tbody>
            </table>
         </text>
         <entry> <!--one for each input data-->
            <observation>
               <code code="248333004" codeSystem="2.16.840.1.113883.6.96"
                  codeSystemName="SNOMED CT" displayName="Standing height"/>
               <effectiveTime value="200604071430"/>
               <value xsi:type="PQ" value="186" unit="cm"/>
            </observation>
         </entry>
      </section>
   </structuredBody>
</ClinicalDocument>
```

Fig. 2 – Structure and contents of CDA input document. The clinical concept 'Standing height' is here the only input data.



```
<databind name="height" code="248333004"
   codeSystem="2.16.840.1.113883.6.96">
   <value xsi:type="PQ" unit="cm"/>
   <!-- if required location is specified -->
   <location xpath="//*[@code=‘PATIENT_DATA']/../entry/observation/
                    code[@code='248333004']/../value/@value"/>
</databind>
```

Fig. 3 – An example of a data bind in the binding file. The data with the specified code and code system in the input document is matched to the knowledge-base fact named ‘height' and its value is restricted to a physical quantity using centimeters as unit.

future work. Fig. 3 shows an example of the data binding file that provides the configuration to this procedure.

It is straightforward that the data structure of the input facts will depend on the facts format of the inference engine on which the CDSS is based. Some inference engines may require the input to be in a custom text format, others may accept data structures or objects in a specific programming language. It is out of the scope of the present work to cover all types of formats, but to provide a generic method to facilitate the integration of such systems. However, as described in next section, a case of use based on the Jess inference engine was developed and a generic data model based on Java objects was provided. Input facts will be created according to the required format.

3.3. *Inference*

After input facts are created these are passed to the inference engine including them in its working memory. Pattern matching will then attempt to match fact names with the antecedents of human-readable rules. After a match, rule consequents update the fact base with the corresponding conclusions. In addition, rule contextual information such as bibliographic references on which the rule is based, evidence or risk levels, or associated pathologies can be added to rule consequents. Furthermore, an interesting outcome of an inference-based CDSS is to provide the user with an optional full information about the resultant inference process. We previously stated that executed rules could be chained in order to link initial facts to final conclusions. On the contrary to artificial intelligence planning, where the expected result is a set of optimal chained actions, the general result of a logic inference process is a modification in the fact base—regardless of the final objective—, so by default the chain of executed rules is not provided. If tracking is required, each rule may add a specific fact relating itself to its input and output facts, what permits tracing a graph of executed rules from initial to resultant facts. A simple example of a rule for calculating the body mass index is shown in Fig. 4.

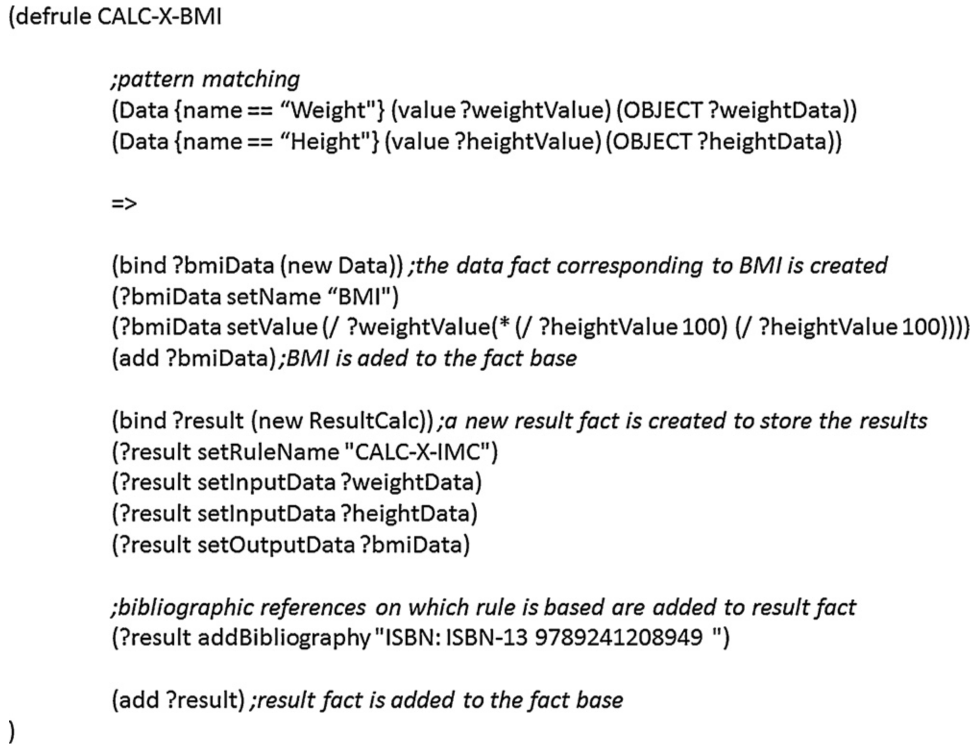
3.4. *Binding from rule results to CDA output document*

Once inference finishes, results must be converted into an HL7-CDA document. Output facts are also represented in an inference-engine specific format or data structures and must be translated into CDA. Inversely to the input facts creation, the binding file relates the human-readable fact names to their code and code systems. The binding process translates output facts to the CDA format specified in the HL7-CDA implementation guide. HL7-CDA documents must contain a human readable narrative block, therefore all the content obtained from the inference and rule results should be provided not only well structured and codified—machine understandable—but also in a manner the user can directly read it. For such a purpose, language files are used to obtain the required texts depending on the language specified in the CDSS context. Anyway, since the CDA output content is semantically structured and codified, third systems could easily take the contents and translate them to other languages—e.g. during a mapping process to other document format. In the following, the CDA output is described.

3.5. *CDA output document*

Rules can be viewed as a causal relation of antecedents to consequents. The HL7-CDA provides the ‘entryRelationship' class which relates two acts in many types of semantics—see Table 1 in [9]. However, we were not able to find any typology of entryRelationship capable of relating two sets of acts. This would be the case of a rule that deduces a set of output facts from a set of input facts. On the contrary, for the case that a rule consequent consists of a single clinical act we found the ‘RSON' entryRelationship type to be suitable for two reasons: (1) source can represent instances of most types of acts, including observations and procedures—the latter valid for clinical recommendations—and (2) the HL7-CDA XML schema allows the target to be an organizer class, what allows including a set of input facts. In addition, source act, i.e. the rule conclusion, can use the ‘reference' class to indicate the bibliographic references on which the rule is based. However, in order to be generic, we propose other solution to consider the aforementioned case of multiple rule consequents. The entryRelationship method could, anyway, be used for specific cases or combined with the generic solution, as we will see in Section 4.

The generic method for representing rules in the output CDA document is based on representing each rule by an entry class containing an organizer which consists of the different information provided by the rule. The organizer's code corresponds to the rule name from a code system representing the problem's knowledge base. A code system is provided to code each organizer component that is codified according with its content. Then, the organizer is composed by the following components: (1) an organizer containing the set of input



```
(defrule CALC-X-BMI

        ;pattern matching
        (Data {name == "Weight"} (value ?weightValue) (OBJECT ?weightData))
        (Data {name == "Height"} (value ?heightValue) (OBJECT ?heightData))

        =>

        (bind ?bmiData (new Data)) ;the data fact corresponding to BMI is created
        (?bmiData setName "BMI")
        (?bmiData setValue (/ ?weightValue(* (/ ?heightValue 100) (/ ?heightValue 100))))
        (add ?bmiData);BMI is aded to the fact base

        (bind ?result (new ResultCalc)) ;a new result fact is created to store the results
        (?result setRuleName "CALC-X-IMC")
        (?result setInputData ?weightData)
        (?result setInputData ?heightData)
        (?result setOutputData ?bmiData)

        ;bibliographic references on which rule is based are added to result fact
        (?result addBibliography "ISBN: ISBN-13 9789241208949 ")

        (add ?result) ;result fact is added to the fact base
)
```

Fig. 4 – Example rule for calculating the body mass index (BMI). Rule is written in JESS.

facts, (2) an organizer containing the set of output facts, these two mandatory, and the following optional components, (3) an observation indicating the evidence level of the rule, (4) an observation indicating the risk level of the rule, and (5) an organizer containing the pathologies associated with the rule. Additionally, the organizer can contain reference elements to indicate bibliographic references on which the rule is based. An example of an entry representing a rule for calculating the body mass index is shown in Fig. 5.

Additionally, to accomplish with the HL7-CDA requirements, the CDA output document provides both the header and the narrative block displaying the contents of the structured results. For the narrative block we decided to include each rule information in a table, where first column indicates the name of each information item and second their content. However, many other solutions could be provided for this representation. Once this narrative block is signed, provides legal validity to the output of the CDSS.

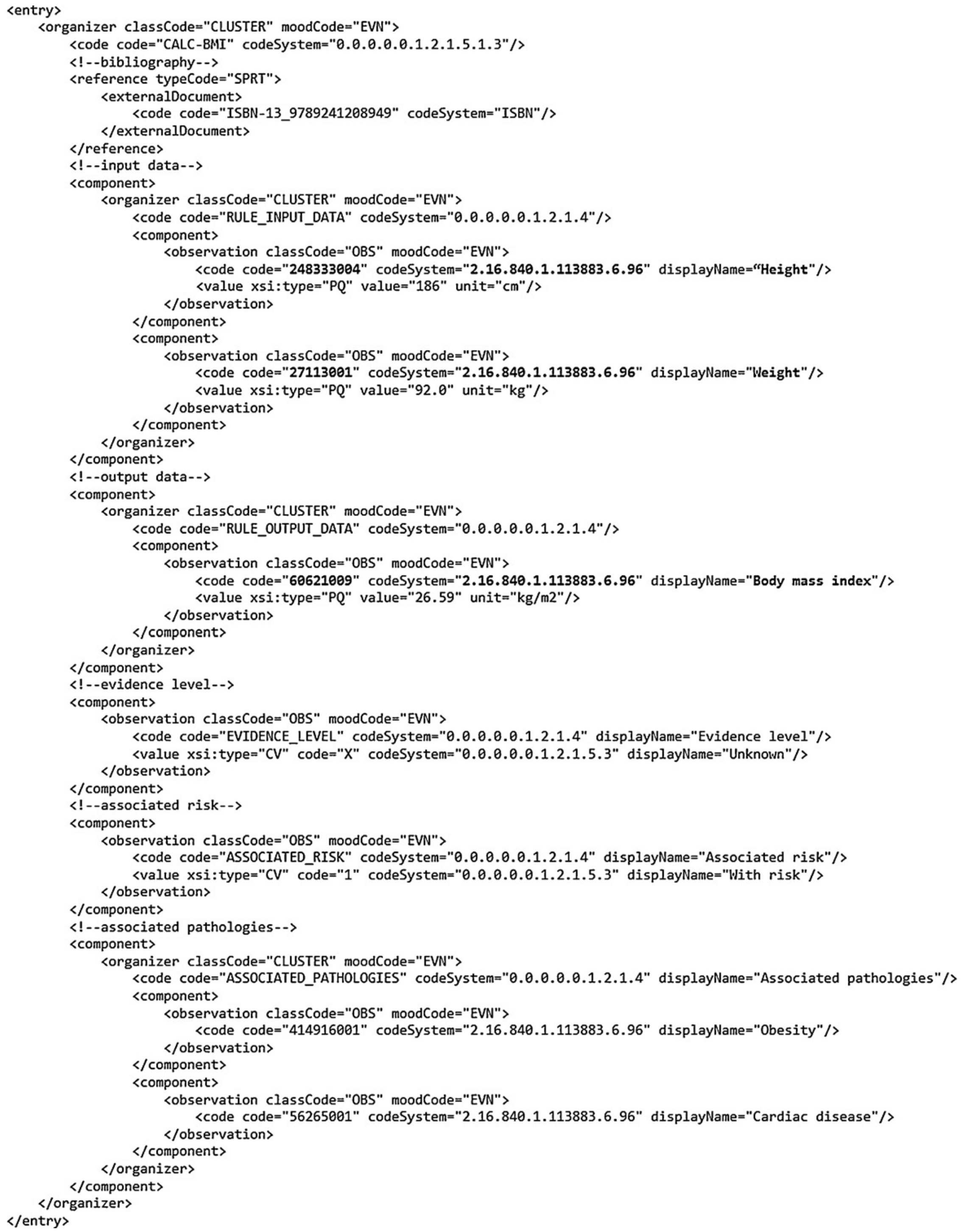
At this point, the entity who requested the decision support have access to the inference results in a standardized, codified, and contextually annotated format, that is, semantically interoperable. Again, as described in the input document subsection, non-HL7-CDA systems could interpret the resultant CDA document by means of an external mapping, whether to other clinical document or to a relational database. Computers can process rule results in order to incorporate data to patients' EHR or to display results in graphical user interfaces. Additionally, the fact that each rule makes reference to its input and output facts allows drawing a graph representing the inference process carried out to achieve the resultant conclusions, thus improving the interpretability of the rule-based CDSS.

4. Case of study on a diabetes mellitus CDSS

The solution proposed in this paper have been successfully applied in the development of a CDSS for care management of outpatients with diabetes mellitus. This CDSS is integrated into a telemedicine health platform of a Spanish multinational that is being introduced into the healthcare business.

The objective of the CDSS is to provide physicians and patients with recommendations about patients' lifestyle, diet and physical activities, as well as alerts about the risk of suffering diseases related to diabetes. Recommendations and alerts are inferred using patient data by means of a rule-based decision support module integrated in the system using the solution proposed in this paper and the Jess inference engine.

Patients' last status can be transformed in any moment into an HL7-CDA document, which contains last clinical data as well as data captured by specific telemedicine measurement devices. Such CDA document representing a patient's state conforms with the defined implementation guide and thus can be used as input for the decision support module. Data were coded following SNOMED-CT terminology and custom terminology tables and their code systems were included for specific domains. The binding file relates codifications of a sum of 104 terms to the human-readable names used in the knowledge base. The knowledge base employed by the system to infer recommendations was written in the Jess rules language based in the ADA's recommendation guide for care management [24]. Rules were divided into rules for the assessment of physicians and rules for the assessment of patients,

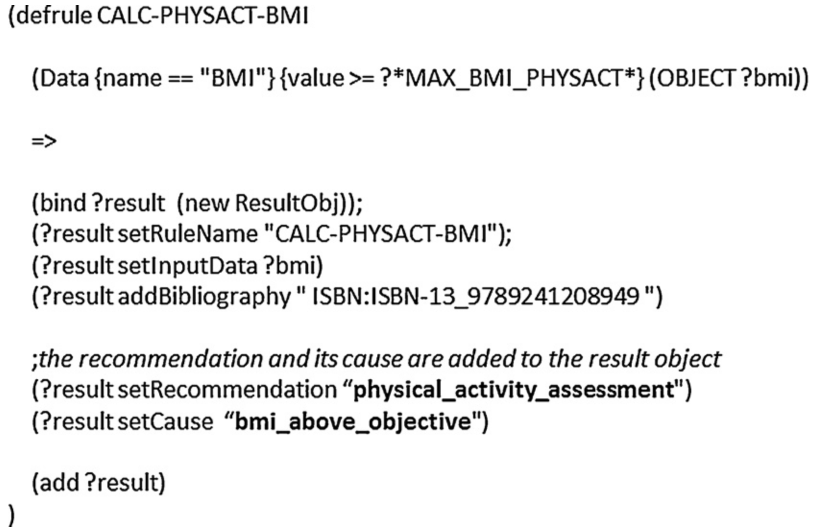


```
<entry>
    <organizer classCode="CLUSTER" moodCode="EVN">
        <code code="CALC-BMI" codeSystem="0.0.0.0.0.1.2.1.5.1.3"/>
        <!--bibliography-->
        <reference typeCode="SPRT">
            <externalDocument>
                <code code="ISBN-13_9789241208949" codeSystem="ISBN"/>
            </externalDocument>
        </reference>
        <!--input data-->
        <component>
            <organizer classCode="CLUSTER" moodCode="EVN">
                <code code="RULE_INPUT_DATA" codeSystem="0.0.0.0.0.1.2.1.4"/>
                <component>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="248333004" codeSystem="2.16.840.1.113883.6.96" displayName=“Height"/>
                        <value xsi:type="PQ" value="186" unit="cm"/>
                    </observation>
                </component>
                <component>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="27113001" codeSystem="2.16.840.1.113883.6.96" displayName="Weight"/>
                        <value xsi:type="PQ" value="92.0" unit="kg"/>
                    </observation>
                </component>
            </organizer>
        </component>
        <!--output data-->
        <component>
            <organizer classCode="CLUSTER" moodCode="EVN">
                <code code="RULE_OUTPUT_DATA" codeSystem="0.0.0.0.0.1.2.1.4"/>
                <component>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="60621009" codeSystem="2.16.840.1.113883.6.96" displayName="Body mass index"/>
                        <value xsi:type="PQ" value="26.59" unit="kg/m2"/>
                    </observation>
                </component>
            </organizer>
        </component>
        <!--evidence level-->
        <component>
            <observation classCode="OBS" moodCode="EVN">
                <code code="EVIDENCE_LEVEL" codeSystem="0.0.0.0.0.1.2.1.4" displayName="Evidence level"/>
                <value xsi:type="CV" code="X" codeSystem="0.0.0.0.0.1.2.1.5.3" displayName="Unknown"/>
            </observation>
        </component>
        <!--associated risk-->
        <component>
            <observation classCode="OBS" moodCode="EVN">
                <code code="ASSOCIATED_RISK" codeSystem="0.0.0.0.0.1.2.1.4" displayName="Associated risk"/>
                <value xsi:type="CV" code="1" codeSystem="0.0.0.0.0.1.2.1.5.3" displayName="With risk"/>
            </observation>
        </component>
        <!--associated pathologies-->
        <component>
            <organizer classCode="CLUSTER" moodCode="EVN">
                <code code="ASSOCIATED_PATHOLOGIES" codeSystem="0.0.0.0.0.1.2.1.4" displayName="Associated pathologies"/>
                <component>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="414916001" codeSystem="2.16.840.1.113883.6.96" displayName="Obesity"/>
                    </observation>
                </component>
                <component>
                    <observation classCode="OBS" moodCode="EVN">
                        <code code="56265001" codeSystem="2.16.840.1.113883.6.96" displayName="Cardiac disease"/>
                    </observation>
                </component>
            </organizer>
        </component>
    </organizer>
</entry>
```

Fig. 5 – Example of the CDA entry representing a rule result from calculating the body mass index. The main organizer element contains the rule information and results: bibliography, input data, output data, evidence level, associated risk and associated pathologies. The data corresponding to the calculus of the body mass index in our example is in bold.

where 106 rules were defined in the former group and a subset of 46 were used in the latter.

A customization of the current solution was included here to differentiate between intermediate—calculus—rules and recommendation rules. Recommendation rules can provide information regarding to patient follow-up, accomplishment of objectives or treatments. This way, rules that provide recommendations include extra information in their consequents that allow including the entryRelationship class in the CDA output document relating the recommended procedure—using the procedure class—with the set of data reason of the recommendation, as explained in last section.

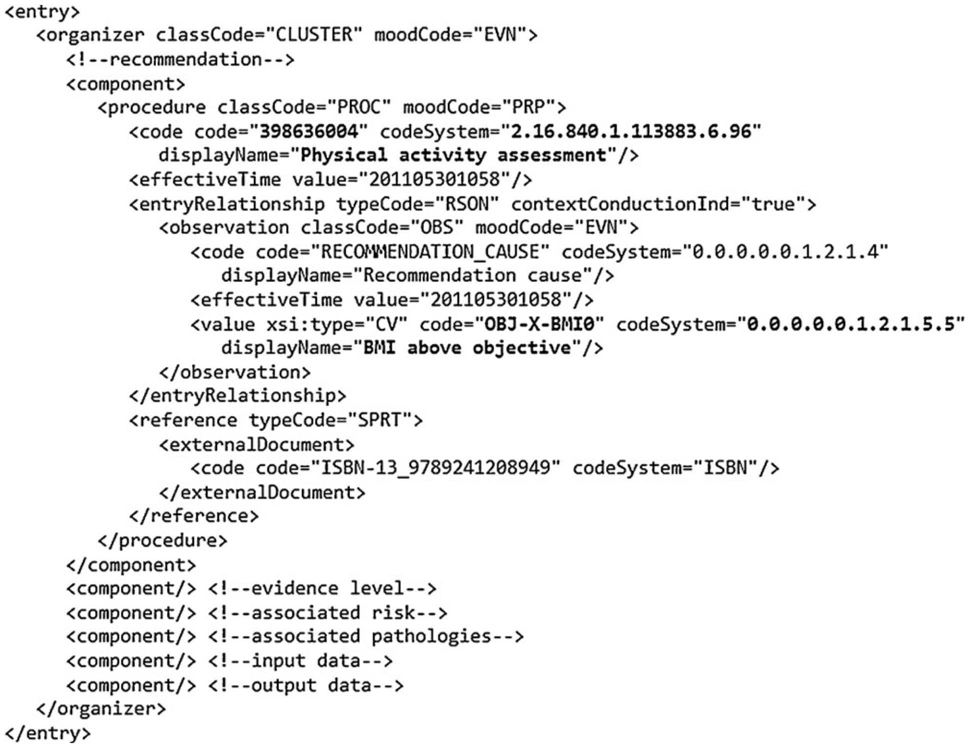


```
(defrule CALC-PHYSACT-BMI

   (Data {name == "BMI"} {value >= ?*MAX_BMI_PHYSACT*} (OBJECT ?bmi))

   =>

   (bind ?result  (new ResultObj));
   (?result setRuleName "CALC-PHYSACT-BMI");
   (?result setInputData ?bmi)
   (?result addBibliography " ISBN:ISBN-13_9789241208949 ")

   ;the recommendation and its cause are added to the result object
   (?result setRecommendation “physical_activity_assessment")
   (?result setCause  “bmi_above_objective")

   (add ?result)
)
```

Fig. 6 – Rule for providing a recommendation for a physical activity assessment when the body mass index is above a specified threshold. Rule is written in Jess.

An example of this can be observed in Figs. 6 and 7. Fig. 6 shows a rule that includes in its result a recommendation for a physical activity assessment when the body mass index is above a specified threshold, while including the corresponding cause—this rule would be fired after the body mass index is calculated in the intermediate rule in Fig. 4. Fig. 7 shows the CDA structure corresponding to the output of such recommendation rule, which provides the same components as in the previous example in Fig. 5, as well as the aforementioned entryRelationship element, which semantically relates the procedure associated with the recommendation with its cause. Thus, a complete CDA output could include both entries in Figs. 5 and 7, which could be linked between each other based in their output and input data components respectively.

In order to evaluate the effectiveness of the proposed approach under this settings, we carried out a set of continuous tests based in a simulation of ten different patient profiles defined and approved by the physicians in the project. Each



```
<entry>
   <organizer classCode="CLUSTER" moodCode="EVN">
      <!--recommendation-->
      <component>
         <procedure classCode="PROC" moodCode="PRP">
            <code code="398636004" codeSystem="2.16.840.1.113883.6.96"
               displayName="Physical activity assessment"/>
            <effectiveTime value="201105301058"/>
            <entryRelationship typeCode="RSON" contextConductionInd="true">
               <observation classCode="OBS" moodCode="EVN">
                  <code code="RECOMMENDATION_CAUSE" codeSystem="0.0.0.0.0.1.2.1.4"
                     displayName="Recommendation cause"/>
                  <effectiveTime value="201105301058"/>
                  <value xsi:type="CV" code="OBJ-X-BMI0" codeSystem="0.0.0.0.0.1.2.1.5.5"
                     displayName="BMI above objective"/>
               </observation>
            </entryRelationship>
            <reference typeCode="SPRT">
               <externalDocument>
                  <code code="ISBN-13_9789241208949" codeSystem="ISBN"/>
               </externalDocument>
            </reference>
         </procedure>
      </component>
      <component/> <!--evidence level-->
      <component/> <!--associated risk-->
      <component/> <!--associated pathologies-->
      <component/> <!--input data-->
      <component/> <!--output data-->
   </organizer>
</entry>
```

Fig. 7 – CDA structure corresponding to the output of rule in this figure. Note in bold letters the codified terms outputted from rule in Fig. 6 (in bold in such figure) as well as the rule input data, which serves as a link with previous rule output (represented in bold in Fig. 5).

patient showed a specific configuration of demographic and clinical data, associated to specific diseases or risks. Tests were carried out in parallel to the development of the system and rules, and consisted in (1) performing unitary tests of rules, (2) checking the consistency of the binding file with respect to the standardized CDA documents, (3) checking the consistency of mappings between the system database and the standardized CDA documents, knowledge base, and language files, and (4) validating the accurate recommendations and assessments of each simulated case given their conditions. The consistency test of CDA documents with the system database was carried out by the external institution in charge of the design and development of the telemedicine system. The initial tests helped in selecting the appropriate terminological concepts to be used within the CDA documents and the binding file. Tests also allowed to find codification errors that made some recommendation rules not to execute and thus making the final results to be incomplete or inaccurate. Separating the knowledge-base human readable terms from the semantically annotated concepts in standardized CDA documents facilitated the correction of most errors based on the intermediate binding file. As a result, the proposed HL7-CDA wrapper may permit the reuse of the developed diabetes mellitus rule-based CDSS in different Health Information Systems, only requiring external mappings to the standardized HL7-CDA documents.

5. Conclusions and future work

Using standards for data interchange facilitate the development and integration of computer technologies. The solution proposed in this paper, based in HL7-CDA and clinical terminologies, facilitates the scalability and reuse of rule-based CDSSs avoiding ad hoc solutions for specific clinical data repositories, and leads CDSSs to semantic interoperability. Specifically, according to the Semantic Interoperability Roadmap for Europe in [3] our approach would lead a rule-based CDSS between Level 2 and Level 3 of interoperability, facilitating the transition to Level 3. The binding of codified clinical terms into a human-readable facts makes easier the comprehensibility and maintenance of the CDSS knowledge-base since rules can be read and updated using representative human-readable texts. The results of the CDSS, while described in HL7-CDA documents, have legal validity and can be understood both by humans and machines. Finally, by providing in each rule result its input and output facts, inference results can be chained to construct a graph detailing the complete inference process, thus improving the interpretability of the decision support.

The proposed solution has been adopted in a telemedicine health system in a CDSS module for providing patient specific recommendations and alerts about diabetes mellitus.

We consider several aspects to be improved as future lines of work. Semantics can be improved by studying the handling of postcoordinated data, mapping data based in a simple level of concept abstraction —these currently left to external mapping procedures—, as well as analyzing future possibilities of the entryRelationship HL7-CDA class for representing rule results. The definition of archetypes to standardize data interchange of CDSS would help in the task of mapping data from Health Information Systems to CDSS. In addition, the new vMR specification will be studied as input and output documents. In the functional aspect, the automatic unit conversion in both directions—input and output—can be a key aspect if this is not handled by the external system where the CDSS is to be integrated. Additionally, we will study how to reuse the binding file in different CDSS by means of including mappings from different clinical domains. An extension to machine learning based CDSS is other of the future lines of work. A solution for this type of CDSSs would be quite similar as the current proposal, however, special care should be taken for probability and uncertainty management in the resultant document using HL7-CDA.

Conflict of interest

No conflict of interest.

Acknowledgements

We thank Fagor Electrodomésticos S.Coop for their support and funding in the development of this work, specially to Juan Ramón Inurria and Jorge de Antonio Prieto. We also thank the colaboration from Universidad de Mondragon in the design of the general architecture of the telemedicine system, specially, Félix Larrinaga. This work has been partially supported by the Health Institute Carlos III through the RETICS Combiomed, RD07/0067/2001.

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at http://dx.doi.org/10.1016/j.cmpb.2012.10.003.

REFERENCES

[1] E.S. Berner, Clinical decision support systems: state of the art, in: Agency for Healthcare Research and Quality, 2009, AHRQ Publication No. 09-0069-EF.

[2] K. Kawamoto, C.A. Houlihan, E. Andrew Balas, D.F. Lobach, Improving clinical practice using clinical decision support systems: a systematic review of trials to identify features critical to success, British Medical Journal (Clinical Research Ed.) 330 (April (7494)) (2005) 765+.

[3] European Comission, Semantic Interoperability for Better Health and Safer Healthcare, Deployment and Research Roadmap for Europe, 2009.

[4] Standards Australia Limited, Guide to the Principles and Desirable Features of Clinical Decision Support Systems, SAA HB. Standards Australia, 2007.

[5] D. Brand, Electronic Decision Support for Australia's Health Sector, Technical Report, National Electronic Support Taskforce, 2002.

[6] C. Sáez, J.M. García-Gómez, J. Vicente, S. Tortajada, J. Luts, D. Dupplaw, S. Van Huffel, M. Robles, A generic and extensible automatic classification framework applied to brain tumour diagnosis in healthagents, The Knowledge Engineering Review 26 (3) (2011) 283–301.

[7] A. Cornuéjols, On-line learning: where are we so far? in: M. Michael, S. Lorenza (Eds.), Ubiquitous Knowledge Discovery, Lecture Notes in Computer Science, Vol. 6202, Springer, Berlin Heidelberg, 2010, pp. 129–147.

[8] K. Engel, B. Blobel, P. Pharow, Standards for enabling health informatics interoperability, Studies in Health Technology and Informatics 124 (2006) 145–150.

[9] R.H. Dolin, L. Alschuler, S. Boyer, C. Beebe, F.M. Behlen, P.V. Biron, A.S. Shvo, HL7 clinical document architecture, release 2, Journal of the American Medical Informatics Association: JAMIA 13 (January (1)) (2006) 30–39.

[10] R.A. Miller, H.E. Pople, J.D. Myers, Internist-1, an experimental computer-based diagnostic consultant for general internal medicine, New England Journal of Medicine 307 (8) (1982) 468–476.

[11] E.H. Shortliffe, Computer-Based Medical Consultations: Mycin, Elsevier, North Holland, New York, 1976.

[12] E.H. Shortliffe, A.C. Scott, M.B. Bischoff, A.B. Campbell, W.va. Melle, C.D. Jacobs, Oncocin: an expert system for oncology protocol management, in: Seventh International Joint Conference on Artificial Intelligence, Vancouver, 1981.

[13] OpenClinical: Artificial Intelligence Systems in Clinical Practice, 2011, Retrieved from: http://www.openclinical.org/aisinpractice.html

[14] J. Komulainen, I. Kunnamo, P. Nyberg, M. Kaila, T. Mantyranta, M. Korhonen, Developing an Evidence Based Medicine Decision Support System Integrated with EPRS Utilizing Standard Data Elements, Riva del Garda, Italy, 2006.

[15] O. Nee, A. Hein, T. Gorath, N. Hulsmann, G.B. Laleci, M. Yuksel, M. Olduz, I. Tasyurt, U. Orhan, A. Dogac, A. Fruntelata, S. Ghiorghe, R. Ludwig, SAPHIRE: intelligent healthcare monitoring based on semantic interoperability platform: pilot applications, Communications, IET 2 (February (2)) (2008) 192–201.

[16] G.B. Laleci, A. Dogac, A semantically enriched clinical guideline model enabling deployment in heterogeneous healthcare environments, IEEE Transactions on Information Technology in Biomedicine 13 (2) (2009) 263–273.

[17] R.S. Kazemzadeh, K. Sartipi, P. Jayaratna, A framework for data and mined knowledge interoperability in clinical decision support systems, International Journal of Healthcare Information Systems and Informatics 5 (1) (2010) 37–60.

[18] M. Sordo, O. Ogunyemi, A.A. Boxwala, R.A. Greenes, GELLO: an object-oriented query and expression language for clinical decision support, in: AMIA Annual Symposium Proceedings, 2003, p. 1012.

[19] K. Kawamoto, G. Del Fiol, H.R. Strasberg, N. Hulse, C. Curtis, J.J. Cimino, B.H. Rocha, S. Maviglia, E. Fry, H.J. Scherpbier, V. Huser, P.K. Redington, D.K. Vawdrey, J.C. Dufour, M. Price, J.H. Weber, T. White, K.S. Hughes, J.C. McClay, C. Wood, K. Eckert, S. Bolte, D. Shields, P.R. Tattam, P. Scott, Z. Liu, A.K. McIntyre, Multi-national, multi-institutional analysis of clinical decision support data needs to inform development of the HL7 virtual medical record standard, in: AMIA Annual Symposium Proceedings 2010, 2010, pp. 377–381.

[20] J. Mei, H. Liu, G. Xie, S. Liu, B. Zhou, An OCL-compliant GELLO engine, Studies in Health Technology and Informatics 169 (2011) 130–134.

[21] K. Kawamoto, OpenCDS: An Open-Source, Standards-Based, Service-Oriented Framework for Scalable CDS, in: 12th International HL7 Interoperability Conference, Orlando, FL USA, May 2011.

[22] J.A. Maldonado, D. Moner, D. Boscá, J.T. Fernández-Breis, C. Angulo, M. Robles, Linkehr-ed: a multi-reference model archetype editor based on formal semantics, International Journal of Medical Informatics 78 (8) (2009) 559–570.

[23] C. Angulo, P. Crespo, J.A. Maldonado, D. Moner, D. Pérez, I. Abad, J. Mandingorra, M. Robles, Non-invasive lightweight integration engine for building EHR from autonomous distributed systems, International Journal of Medical Informatics 76 (Suppl. 3) (2007) S417–S424.

[24] American Diabetes Association, Standards of medical care in diabetes-2010, Diabetes Care 33 (January (Suppl. 1)) (2010) S11–S61.