IADIS International Journal on Computer Science and Information Systems
Vol. 11, No. 1, pp. 17-31
ISSN: 1646-3692

C-WAREHOUSING: A HL7 CDA-BASED APPROACH FOR THE SECONDARY USE OF CLINICAL DATA

Fabrizio Pecoraro, Daniela Luzi and Fabrizio L. Ricci. *Institute for Research on Population and Social Policies, National Research Council, Via Palestro, 32 – 00185 – Rome, Italy*

ABSTRACT

This paper proposes a semi-automatic approach to extract information stored in a HL7 Clinical Document Architecture (CDA) and transform them to be loaded in a Data Warehouse for secondary purposes. It represents a suitable solution to facilitate the design and implementation of Extract, Transform and Load (ETL) tools that are considered the most time-consuming step of the data warehouse development process. The implementation of this framework is also proposed adopting the XSLT style sheet language that converts an original CDA XML-based document to an output XML document that can be easily loaded in the Data Warehouse. A case study is also provided to demonstrate the feasibility of the approach proposed.

KEYWORDS

Data Warehousing; Dimensional Model; HL7 CDA; EXtensible Stylesheet Language Transformation (XSLT); XML

1. INTRODUCTION

In the healthcare setting there is a growing attention on secondary uses of clinical data defined as "non-direct care use of personal health information" (Safran et al., 2007). The use of clinical data for secondary purposes provides important sources to support decision-making in different domains, such as patient safety, healthcare quality assessment, clinical and translational research including clinical trials, comparative analysis of therapy pathways and best practices application (Elkin et al., 2010). To reach this aim a comprehensive analysis is required that has to integrate clinical and administrative information provided by heterogeneous information systems often developed using different technologies, for different specialties and purposes and by different organizations (Wickramasinghe and Schaffer, 2006; Pecoraro et al., 2013). This makes it necessary to implement specific Extract, Transform and

Load (ETL) procedures devoted to convert data from source operational systems in a common data model optimized for data analysis purposes.

In healthcare different standards have been developed to facilitate system interoperability and under the perspective of data models, HL7 (Schadow et al., 2006) surely represents one of the main candidates for the integration and exchange of information (Benson, 2010), generally focused on patient's care delivery. One of the widely adopted HL7 standard is the Clinical Document Architecture (CDA) (Dolin et al., 2006) that specifies the encoding, structure and semantics of clinical documents using a XML based mark-up language. Recently, many initiatives have analyzed the importance of designing and implementing a data warehouse starting from XML documents considering the continual growth of representing data using XML documents in different domains (Sen et al., 2012; Kavitha and Vydehi, 2014; Golfarelli et al., 2001). In our vision the main aim of HL7 standards and in particular the CDA (Clinical Document Architecture) (Dolin et al., 2006) can be extended to define a common schema able to represent this information in enterprise data warehouses to be used for secondary purposes.

Aim of this paper is to define a semi-automatic approach to extract information from XML document structured using the CDA standard and transform them to be included in a data warehouse schema. To perform this task an EXtensible Stylesheet Language Transformation (XSLT) document (W3C, 2007) is defined to provide an output XML document that can be easily stored in the data warehouse logical schema. This approach is based on a conceptual framework already described in details in a previous publication using a first-order logic (Pecoraro et al., 2015). Next paragraph describes the main steps of this framework that maps the CDA components with the conceptual model concepts. The third paragraph describes how the conceptual framework has been implemented highlighting the generation of the XSLT document. After that, to demonstrate the feasibility of this approach a case study providing an example of the transformation task is proposed. Final remarks are given in the conclusion paragraph.

This study is part of the Smart Health 2.0 national project that aims to develop a regional healthcare infrastructure based on HL7 standards. It also intends to explore the use of Electronic Health Record (EHR) for secondary purposes in a clinical governance framework to assess the quality of care from the structural, organizational, financial and professional points of view (Pecoraro et al., 2014).

2. CONCEPTUAL MAPPING FROM CDA SCHEMA TO DIMENSIONAL MODEL

2.1 Data Warehouse Model

The data warehouse conceptual modelling can be formalized using the dimensional model as depicted in Figure 1.

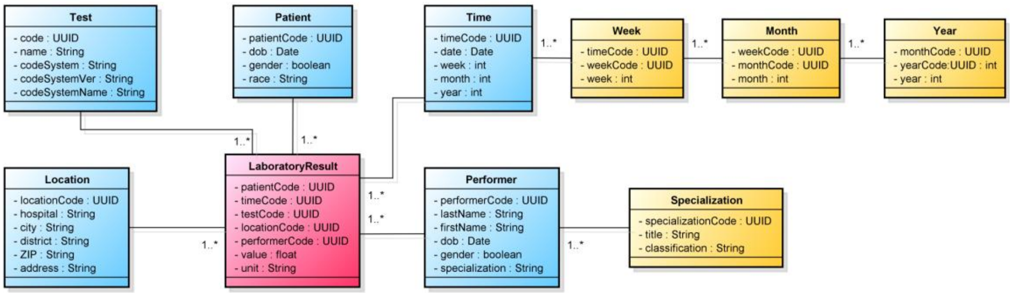

Figure 1. Example of a dimensional representing a snowflake schema composed by a fact LaboratoryResult related to five dimensions: Test, Patient, Time, Location and Performer.

The core of this schema is the fact table that describes the measurements of the performance of a business process using qualitative and/or quantitative attributes called measures. It is surrounded by independent dimensions each one modelled using independent denormalized table or normalized hierarchy. In the first case the model is called star schema while in the second representation is called snowflake schema. Fact along with its relevant measures as well as dimensions represent the concepts of the dimensional model to be mapped with the CDA elements described in the following paragraphs.

2.2 CDA Model

CDA Release 2 Level 3 records clinical observations and services in a mark-up structured standard document based on the six backbone classes of the HL7 Reference Information Model (RIM) (Schadow et al., 2006): Act, ActRelationship, Participation, Entity, Role and RoleLink. As highlighted in Figure 2, these classes as well as their relationships are used to define two main components of the CDA document (Pecoraro et al., 2015):

1. CDA Backbone defined by the Act specializations and their relationships. For instance, the Act ClinicalDocument that represents the entry point (i.e. root) of the CDA document is composed by a set of Sections each one collecting one or more events modelled using the Act classes of the ClinicalStatement choice, such as Observation, SubstanceAdministration.
2. HL7 Hierarchy that describes subjects and objects involved in the process as well as the role played by them within the action using the n-ple <Participation, Role, Entity Player, Entity Scoper> (Luzi et al., 2009). For instance, the hierarchy <recordTarget, patientRole, Patient, Organization> represents the patient involved in the events documented in the CDA. Each HL7 Hierarchy is related with a specific Act of the CDA Backbone that describes the action performed or scheduled.

A portion of the CDA schema highlighting three HL7 hierarchies (i.e. recordTarget, performer and participant) and the CDA Backbone is shown in Figure 3 using the message information model notation.

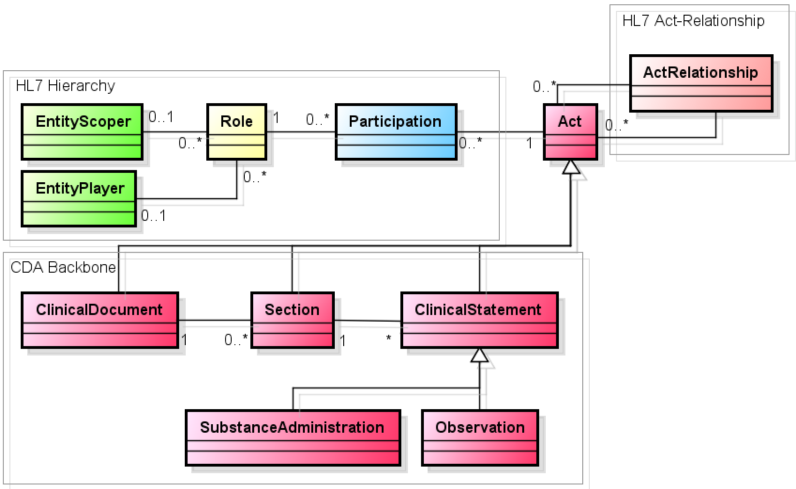

Figure 2. High level class diagram of the CDA schema modelled using the HL7 RIM core classes. The two main components of the CDA are also highlighted: 1) HL7 Hierarchy composed by the triple <Participation, Role, Entity> related to the Act class; 2) CDA Backbone defined by Act specializations and their relationships.

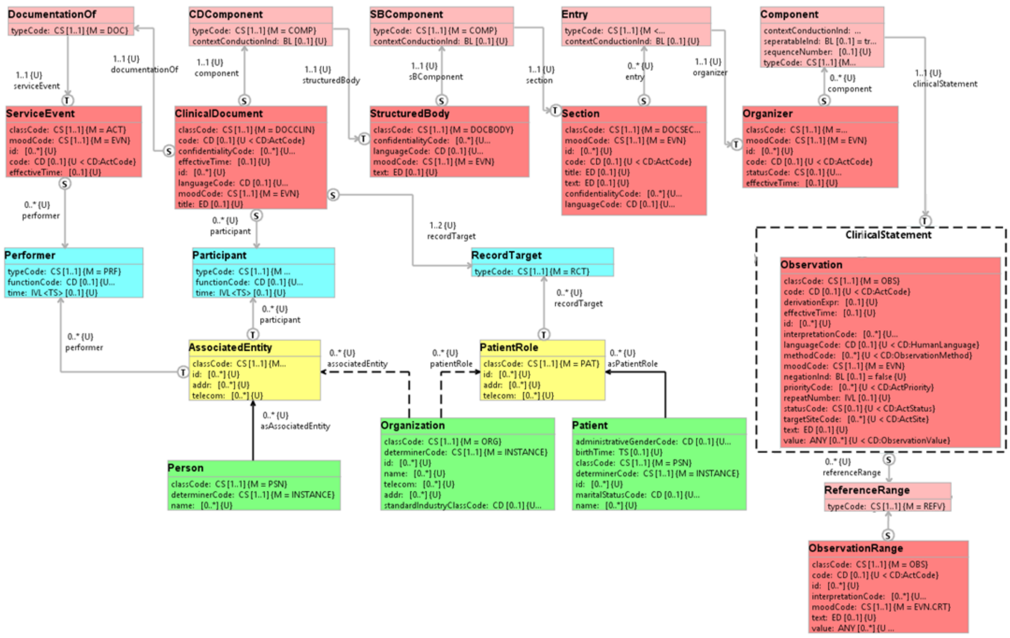

Figure 3. Portion of the of the CDA message model showing the CDA backbone and three HL7 Hierarchies: Performer, Participant and RecordTarget.

The described HL7 Hierarchy and CDA Backbone as well as the relevant complex attributes represent the components of the CDA schema to be mapped with the dimensional model concepts introduced in the previous paragraph.

2.3 Conceptual Framework

In this paragraph we describe a conceptual framework to map the CDA components (HL7 Hierarchy, CDA Backbone) with the dimensional model concepts (Fact, Dimension). To perform this mapping it is necessary that the designer have already identified the business process to be modelled as well as the level of detail to be captured (i.e. what an individual row of the fact table represents). This is an important aspect given that the granularity of the dimensional model influences the identification of both the dimensions to be modelled and the attributes and measures to be captured. This decision has to take also into account the granularity of data contained in the CDA document, that generally captures atomic data, such as value of vital signs observed during a laboratory test.

2.3.1 Identify the Fact

As already mentioned a Fact describes the relevant event to be analysed trough qualitative and quantitative measures that represent the performance of the business process and that could be analysed using statistical methods. In the CDA these information are collected in specific attributes of the stereotype Act of the RIM that represents “measurement of healthcare business processes”. For this reason in our approach the Acts that define the CDA Backbone can be considered as suitable candidates to identify the Fact of the dimensional model depending on the purpose of the analysis to be carried out and on the indicators to be developed. Examples of Act that can describe relate actions and events that constitute health care services are reported in Table 1 where examples of business processes and measures are reported.

Table 1. Example of Act classes that can be used to represent a fact table of the dimensional model

CDA class	Description	Example of processes	Measures
Act	General event that is being done, has been done, can be done, or is intended or requested to be done.	To be used when the other more specific classes aren't appropriate.	N/A
Encounter	An interaction between a patient and healthcare participant(s) to provide service(s) or assessing the health status of a patient.	Specialist and MMG visits	lengthOfStayQuantity (quantity of time when the subject is expected to be or was resident at a facility as part of an encounter)
Observation	action performed in order to determine an answer or a result value	vital signs, clinical results in general and also diagnoses, findings, symptoms,	value (data determined by the observation) interpretationCode (a qualitative interpretation of the observation)
Procedure	An event whose immediate and primary outcome (post-condition) is the alteration of the subject physical condition	conservative procedures such as reduction of a luxated join, including physiotherapy such as chiropractic treatment	N/A
SubstanceAdministration	The act of introducing or otherwise applying a substance to the subject.	Chemotherapy protocol; Drug prescription; Vaccination record	doseQuantity (amount of the therapeutic agent), rateQuantity (the speed with which the substance is dispensed)

Once the Fact has been determined, its attributes are analysed to define measures that represent a qualitative or quantitative evaluation of the business process. For instance, the Act Observation comprises two measures described by the attributes value and interpretationCode. They represent, respectively, a quantitative and qualitative measure of the event observed. In the RIM numerical information are collected in Act class attributes modelled with quantity (i.e. QTY) or physical quantity data type (i.e. PQ), whereas qualitative analysis are specified using coded data types (e.g. CV, CE, CD).

2.3.2 Identify the Dimensions

In this paper dimensions are determined based on the Zachman framework (Inmon et al., 1997) that provides a systematic information representation starting from the following questions related with the investigated event: who (persons), what (the fact), when (the time), where (the place), why (the reason) and how (the manner). To identify suitable candidates to derive dimensions we start analysing the two main structural components of the CDA document related with the fact class: 1) Acts that captures the meaning and purpose of each association with the main event as well as additional actions to determine, for instance, why the event has been performed or the criteria used to evaluate the event outcome; 2) HL7 Hierarchy that describes the functions of subjects and objects involved in a specific process, identifying for instance, who performed it (i.e. performer), for whom it was done (i.e. subject), where it was done (i.e. location). This information is captured through the attribute typeCode of the Participation class that specifies its meaning and purpose using a controlled vocabulary defined by HL7.

Table 2 summarizes examples of the different components of the CDA that can be used to identify a dimension of the schema, reporting the type and the name of the component as well as its description and the related Act class of the backbone.

Table 2. Example of suitable dimensions derived from the CDA components

HL7 component	**Name**	**Example**
	Clinical Document	
HL7 Hierarchy	recordTarget	Patient involved in the event
	performer	Physician/Practitioner that carried out the event
	responsibleParty	Participant with legal responsibility
	location	Healthcare facility where the event occurred
	participant	Other involved and not mentioned participants
CDA Backbone	ServiceEvent	The main event being documented
	EncompassingEncounter	Primary encounter documented
	Section	
HL7 Hierarchy	subject	Target of the entries recorded in the document
	Clinical Statement	
HL7 Hierarchy	performer	Physician/Practitioner that carried out the event
	specimen	Part of entity typically the subject target of the observation
	participant	Other involved and not mentioned participants
	Observation	
CDA Backbone	ObservationRange	Range of values for a particular observation
	SubstanceAdministration	
HL7 Hierarchy	consumable	Substance consumed during the administration.

Moreover, there are attributes of the Fact that can be specifically used to define a degenerate dimension that is not modelled using its own table. A generic Act of the RIM contains several attributes that can be mapped in a degenerate dimension such as, code that classifies the particular kind of Act and statusCode that specifies state of the Act (e.g. active, cancelled). Another important attribute is the effectiveTime that describes time/date when the event took place. This approach of representing dimensions as keys of the fact table often occurs when the dimensional model captures atomic information with a high-level of granularity transaction.

2.3.3 Refinement of the Dimensional Model

The design of a dimensional model based on the CDA elements results in a high-level normalized data model that is typically adopted in transactional database where an high volume of transactions (insert, update, delete) is performed. Conversely, in a data warehouse environment a highly normalized schema may create inefficiencies in the retrieval as well as in the aggregation of data due to the necessity of executing a large number of joins, which greatly increases response times (Thomas and Datta, 2001). Denormalizing relations reduces the number of physical tables that need to be accessed to retrieve the desired data by reducing the number of joins needed to derive a query (Hoffer et al., 2002). For this reason the star schema is typically adopted to model data in analytical databases where a low volume of transactions (insert, update, delete) is performed with complex queries to be executed.

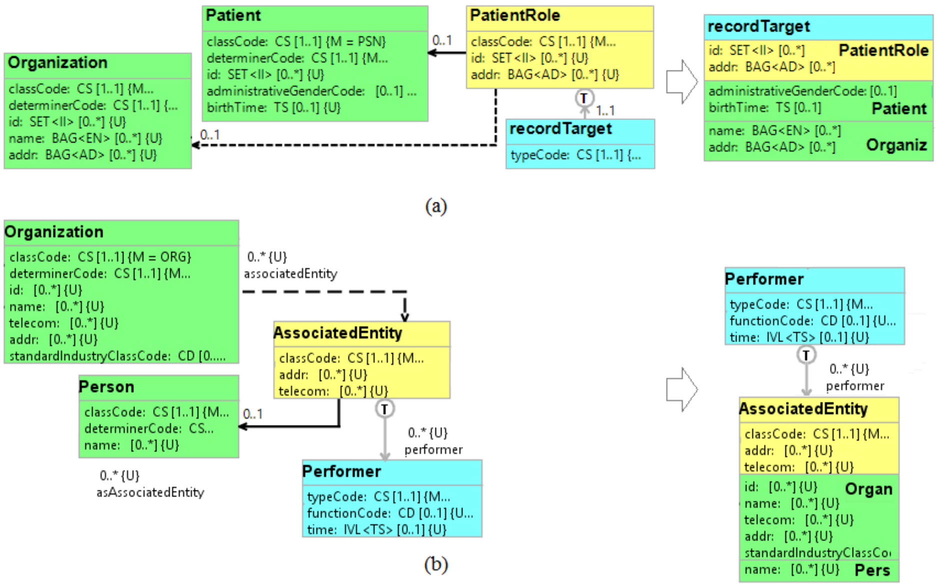

Figure 4. Examples of denormalization of two HL7 Hierarchies: a) recordTarget where attributes of Entities and Role are collapsed in the Participation class; b) performer where attributes of the Entities are collapsed in the Role assignedEntity and the Participation models a bridge to represent a many-to-many relationship with the Fact.

The denormalization is mainly applied to the HL7 hierarchies and is performed by collapsing the attributes of the classes Entities and Role in the Participation class. However, healthcare business processes can require the adoption of many-to-many relationships to represent multiple records of a specific dimension associated with the fact table. For instance, when different practitioners deliver care to an individual over different distinct time intervals or when a specialist visit is performed due to multiple diagnosis. In these cases the hierarchy cannot be fully denormalized and a bridge class should be used to model the many-to-many relationship between the fact and the hierarchy (Eggebraaten et al., 2006). An example of the application of the two denormalization methodologies is depicted in Figure 4 where the attributes of PatientRole, Patient and Organization are collapsed in the recordTarget Participation class and the Specimen is used as a bridge table to map the many-to-many relationship between the Act Observation and the part(s) of the body where the event have been performed. For this reason the attributes of PlayingEntity are collapsed in the SpecimenRole class.

Another important step to be performed to refine the dimensional model is to resolve complex data types. In fact, several attributes of the CDA are coded using a complex data type that consists in a set of fields used to describe the value along with its properties. For instance, the attribute code of the class Observation is coded using the Concept Descriptor (CD) data type that contains eight attributes to model the code of the particular kind of Observation carried out as well as the information about the coding system used to represent it. A possible solution to represent a complex data type is to store each property in a single column of the relevant table excluding properties that are not needed for the business process analysis. For instance a CD can be mapped using only two attributes: code and codeSystem to store the code of the event occurred and the system used to represent it. Moreover, different attributes of the RIM assume multiple values, such as the interpretationCode that specifies a set of rough qualitative interpretation of an Observation based on a HL7 nomenclature (e.g. “is decreased”, “is below alert threshold”, “is moderately susceptible”). These attributes can be modelled either creating a separate table to store each instance or capturing only a single value, such as the first reported in the document.

3. CONCEPTUAL FRAMEWORK IMPLEMENTATION

The workflow to transform and load data stored in a CDA document in the data warehouse is shown in Figure 5 highlighting two main sub-processes.

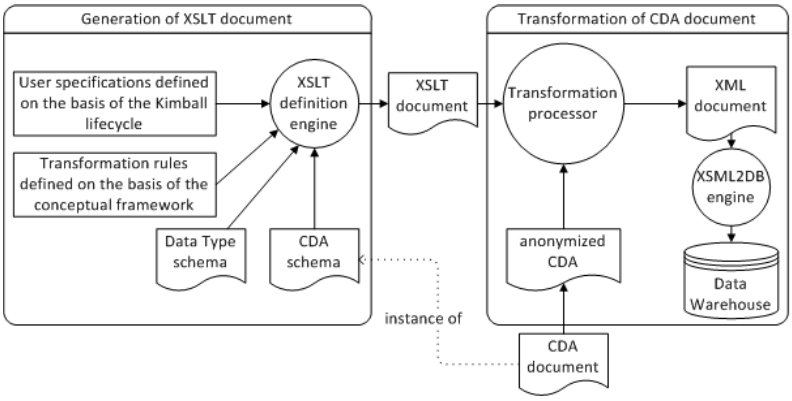

Figure 5. Transformation process to load a CDA document in a Data Warehouse

In the first part of the conceptual framework XSLT document is created taking into account the node chosen to represent the fact of the dimensional model. Moreover, the relevant CDA schema is considered to identify RIM stereotype of each element as well as the cardinality of each relationship, while the data type schema specifies the cardinality and the type of data of each attribute of a specific node. This task is performed by the XSLT definition engine that is further described in the next paragraph.

In the second part of the workflow the XSLT document is used to process a CDA document represented using the XML format in order to produce an output XML document that can be further managed to be mapped into a relational, object-relational or XML-native database. In this perspective, different XML data warehouse architectures have been proposed in the literature to represent complex data as XML documents, such as XCube (Hummer et al., 2003), X-Warehousing (Boussaid et al., 2006), XML-OLAP (Part et al., 2005). to be physically integrated into an Operational Data Store (ODS) and further analyzed using statistical and business intelligence methodologies. These representations converge toward a unified model that differ in the number of XML documents used to store facts and dimensions (Mahboubi et al., 2009). In this paper transformed XML documents are organized on the basis of X-Warehousing architecture where each XML embed the facts stored in the original CDA document as well as their related dimensions. This transformation is performed by a XSLT processor, such as the Open Source SAXON XSLT engine developed by Saxonica Limited.

Note that, to comply with the privacy regulations the original CDA document must be anonymized. However, this activity has not been discussed in the paper given that it has to be applied to the CDA before applying the proposed conceptual framework.

3.1 Generation of the XSLT document

Figure 6 reports the four main components (i.e. templates) of the XSLT document.

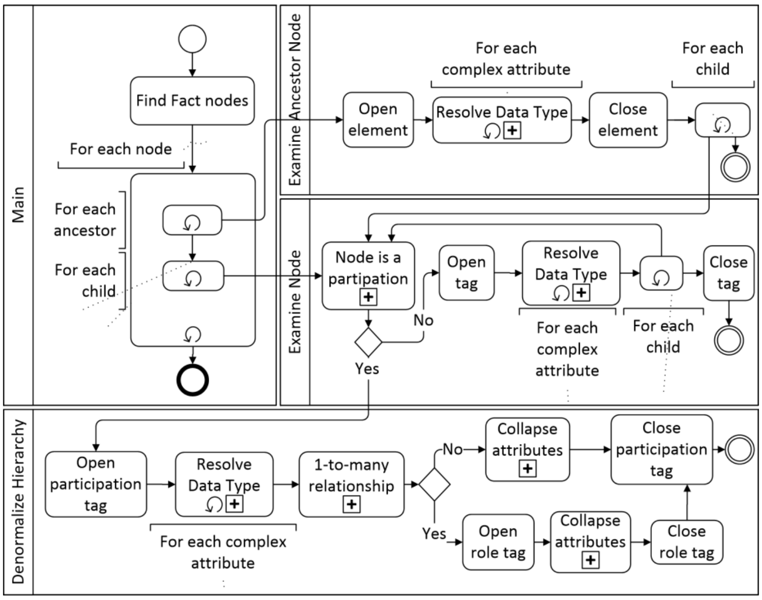

Figure 6. Business process to generate the XSLT document

Each template that compose the XSLT document is identified by a specific pool using the BPMN notation. It highlights the different activities to be executed to transform a CDA structured document in a XML document.

In particular:

1. Main. As highlighted in Figure 7 it finds all the nodes that match with the class chosen by the data warehouse designer to represent the fact table of the dimensional model (e.g. Observation). Starting from each node it navigates the XML document in both directions: each ancestor is explored by the Examine Ancestor Node template, while each child is analyzed by the Examine Node template.

```
<xsl:template match="/">
    <model>
        <xsl:for-each select="//observation">
            <xsl:element name="{name(.)}">
                <!-- examines the children of the fact node -->
                <xsl:for-each select="*">
                    <xsl:call-template name="examineNode">
                        <xsl:with-param name="node" select="." />
                    </xsl:call-template>
                </xsl:for-each>
                <!-- examines all the ancestor of the fact node -->
                <xsl:variable name="ancList" select="ancestor-or-self::*" />
                <xsl:for-each select="$ancList">
                    <xsl:if test="parent::*">
                        <xsl:call-template name="examineAncestorNode">
                            <xsl:with-param name="node" select="../." />
                            <xsl:with-param name="toRemove" select="name(.)"/>
                        </xsl:call-template>
                    </xsl:if>
                </xsl:for-each>
            </xsl:element>
        </xsl:for-each>
    </model>
</xsl:template>
```

Figure 7. Portion of the XSLT document highlighting the Main template

2. Examine Ancestor Node. It includes the node passed as input in the transformed document considering its resolved attributes. Moreover, each child is analyzed by the Examine Node template.
3. Examine Node. It checks if the stereotype of the node received as input is a Participation. In this case the node is passed to the Denormalize Hierarchy, otherwise it is included in the output document along with its resolved attributes. Moreover, each child is recursively analyzed by this template to be included in the output document. Once all children have been analyzed the tag of the relevant node is closed.
4. Denormalize Hierarchy. As shown in Figure 8 starting from a participation node the 4-ple <Participation, Role, Entity Player, Entity Scoper > is analyzed and a denormalized node is reported taking into account the multiplicity of the relationship between the participation and the act class. If the multiplicity is 1-to-1 the complex attributes of role and entity nodes are resolved by the Resolve Data Type function and collapsed in the output schema as children of the participation node using the function Collapse attributes. Otherwise if the relationship is 1-to-many the hierarchy cannot be fully denormalized and a bridge class is needed. To accomplish this task the attributes of entity nodes are resolved and included in the schema as children of the role node.

```
<xsl:template name="denormalizeHierarchy">
    <xsl:param name="node" />
    <xsl:param name="multiplicity" />
    <xsl:element name="{name($node)}">
        <xsl:for-each select="@*">
            <xsl:attribute name="{name(.)}">
                <xsl:value-of select="."/>
            </xsl:attribute>
        </xsl:for-each>
        <xsl:for-each select="$node/*">
            <xsl:if test="my:isStereotype(., 'Role') = 'true'">
                <!-- if is a 1-to-many relationship adds the role tag-->
                <xsl:if test="compare($multiplicity, 'true') = 0">
                    <xsl:element name="{name(.)}">
                        <!-- this template resolves role and
                        entities attributes and collapses them -->
                        <xsl:call-template name="collapsingAttributes">
                            <xsl:with-param name="role" select="." />
                        </xsl:call-template>
                    </xsl:element>
                </xsl:if>
                <xsl:if test="compare($multiplicity, 'true') != 0">
                    <xsl:call-template name="collapsingAttributes">
                        <xsl:with-param name="role" select="." />
                    </xsl:call-template>
                </xsl:if>
            </xsl:if>
            <xsl:if test="my:isStereotype(., 'Role') != 'true'">
                <xsl:call-template name="resolveDataType">
                    <xsl:with-param name="node" select="." />
                </xsl:call-template>
            </xsl:if>
        </xsl:for-each>
    </xsl:element>
</xsl:template>
```

Figure 8. Portion of the XSLT document highlighting the Denormalize Hierarchy template

Moreover, the following functions have been implemented and used in the above-described templates, identified by a rectangle with the plus sign against the bottom line:

a) Resolve Data Type: it analyses a complex attribute and store each property in a single column of the relevant table on the basis of the data type schema. However, attributes that assume multiple values (e.g. value of the Observation class) are modeled creating a bridge table to associate each attribute instance to the relevant node.
b) Node is a participation: it checks if a relevant node belongs with a participation stereotype of the HL7 RIM on the basis of the CDA schema.
c) 1-to-many relationship: it examines whether the multiplicity of the relationship between the relevant node and its father is 1-to-many on the basis of the CDA schema.
d) Collapse attributes: starting from the participation node, this function collects the attributes of both role and entity nodes and collapse them in a single node after resolving data types.

The result of this process is a XML document that can be subsequently pruned and grafted considering the specifications of the user with a particular attention on nodes considered unnecessary for the purpose of the business process analysis.

4. TRANSFORMATION OF A CDA DOCUMENT: A CASE STUDY

In this paper, the proposed approach is tested on a case study that analyses current and historically relevant vital signs. This information is collected in different specifications of the CDA schema produced by different organizations during different events, depending also on the national implementations. For instance, in Italy this information is stored and exchanged using the Report that collects results based on observations generated by laboratories and the Discharge letter that gathers information relative to the patient's hospitalization. At international level HL7 has released an implementation guide, the Continuity of Care Document (CCD) (HL7, 2007), to share patient clinical data specifying the structure and semantics of a patient summary clinical document. In this paper the attention will be focused on the vital signs section of the CCD that models individual's clinical findings, such as blood pressure, heart rate, respiratory rate, height, weight, body mass index, head circumference, crown-to-rump length, and pulse oximetry.

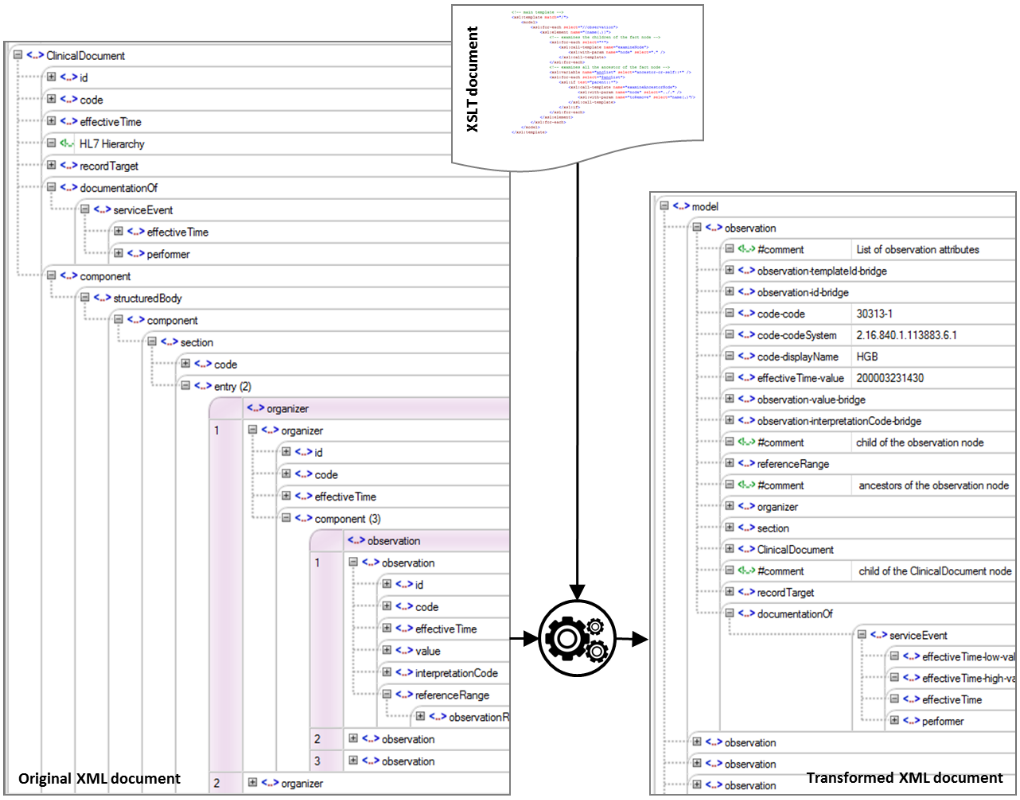

Figure 9. Transformation of the original XML document structured based on the HL7 CDA standard schema in a dimensional model oriented XML document based on the XSLT document

For the purpose of our case study we choose the class Observation as a fact of the dimensional model given that it describes an "action performed in order to determine an answer or a result value". This is the starting point to transform the CDA document in a XML document to be loaded in the data warehouse as reported in the example depicted in Figure 9, where the main template that implements the function to visit the XML tree is based on the proposed methodology. Navigating the tree in a child-parent direction each Observation node will include its ancestors with relevant attributes, such as organizer, section and ClinicalDocument. Moreover, both children of the ClinicalDocument node (i.e. recordTarget and documentationOf) are included in the model as children of the Observation node, along with their children. Subsequently, the tree is parsed in a parent-child direction and the only child of the Observation node (i.e. referenceRange) is included in the model. During these activities each attribute is analyzed and resolved through the template Resolve Data Type taking into account the HL7 data type they are belonging to and also considering if they are multi- or single-valued attribute. This task will be better analyzed in the following when the denormalization of HL7 hierarchies is addressed.

5. CONCLUSION

The paper presents a systematic approach to extract clinical information from CDA documents and to transform them in a XML document to be loaded in a data warehouse for secondary purposes. It is based on a conceptual framework that maps the primitives of the CDA schema with the concepts of the dimensional model. The transformation procedure proposed is based on the widely diffused XSLT style sheet language. It analyses the original XML document structured on the basis of the CDA schema to derive the fact as well as the relevant measures and dimensions of the data mart schema without specific user requirements, thus representing the original information on the basis of the snowflake schema. The result of this transformation is a XML document

This approach will be further tested on a wider set of clinical documents based on different CDA specifications, such as discharge report forms, prescription of pharmacological products and specialist visits, patient summary. This semi-automatic procedure will be applied on the Smart Health 2.0 national project that aims to develop a regional healthcare infrastructure based on HL7 standards. It will be used to develop a dashboard to assess the quality of healthcare service provided in the framework of continuity of care. Starting from a set of selected quality indicators this approach will enable to extract data from CDA documents stored in the Electronic Health Record (EHR) and semi-automatically transform and store them in a data warehouse for secondary purposes in a clinical governance framework.

REFERENCES

Boussaid O. et al., 2006. X-warehousing: an XML-based approach for warehousing complex data. *In Advances in Databases and Information Systems*, pp 39-54

Dolin RH. et al, 2006. HL7 Clinical Document Architecture, Release 2. *In Journal of the American Medical Informatics Association*, Vol. 13, pp 30-39.

Elkin PL. et al, 2010. Secondary use of clinical data. *In Studies in Health Technology and Informatics*, Vol. 155, pp 14-29.

Golfarelli M. et al, 2001. Data warehouse design from XML sources. *Proceedings of ACM International Workshop on Data Warehousing and OLAP.* Atlanta, GA, USA, pp 40-47.

HL7 Implementation Guide. CDA Release 2 – Continuity of Care Document (CCD). Ann Arbor, Mich: Health Level Seven, Inc. 2007.

Hoffer JA. et al., 2002. *Modern database management*. Prentice Hall, Upper Saddle Review, New Jersey.

Hümmer W. et al., 2003. XCube: XML for data warehouses. *Proceedings of the 6th ACM international workshop on Data warehousing and OLAP*, pp 33-40.

Kavitha P. and Vydehi S., 2014. Query Processing of XML Data Warehouse Using XML Pattern Matching Techniques. *In International Journal of Engineering Research and Technology*. Vol. 3.

Kimball R. and Ross M., 2002. *The Data Warehouse Toolkit Second Edition: The Complete Guide to Dimensional Modelling.* Wiley Computing Publishing, New York.

Luzi D. et al, 2009. A medical device Domain Analysis Model based on HL7 Reference Information Model. *Proceedings of International Conference on Medical Informatics in a United and Healthy Europe (MIE)*. Sarajevo, Bosnia and Herzegovina, pp 162-166.

Mahboubi H. et al., 2009. X-wacoda: an xml-based approach for warehousing and analyzing complex data. *In Data Warehousing Design and Advanced Engineering Applications: Methods for Complex Construction*, pp 38-54.

Park et al., 2005. XML-OLAP: a multidimensional analysis framework for XML warehouses. *In Data Warehousing and Knowledge Discovery*. pp 32-42.

Pecoraro F. et al, 2013. Secondary uses of EHR systems: A feasibility study. *Proceedings of IEEE International Conference on E-Health and Bioengineering (EHB)*. Iasi, Romania, pp 1-6.

Pecoraro F. et al, 2015. Designing a Data Warehouse Dimensional Model Based on the HL7 Clinical Document Architecture. *Proceedings of International Conference on Health Informatics (HEALTHINF).* Lisbon, Portugal, pp 284-292.

Schadow G. et al, 2007. The HL7 Reference Information Model under scrutiny. *In Studies in Health Technology and Informatics*, Vol. 124, pp 151-156.

Sen S. et al, 2012. Integrating XML Data Into Multiple Rolap Data Warehouse Schemas. *In International Journal of Software Engineering & Applications*, Vol. 3, pp 197-206.

Thomas H. and Datta A. A conceptual model and algebra for on-line analytical processing in decision support databases. *Information Systems Research*, Vol. 12, No. 1, pp 83–102.

Wickramasinghe N. and Schaffer JL., 2006. Creating knowledge-driven healthcare processes with the intelligence continuum. *In International Journal of Electronic Healthcare*, Vol. 2, pp 164-174.

World Wide Web Consortium (W3C). XSL Transformations (XSLT), Version 2.0, W3C Recommendation 23 January 2007. Available at: http://www.w3.org/TR/xslt20/