

SHARING ELECTRONIC HEALTHCARE RECORDS ACROSS COUNTRY
BORDERS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

MUSTAFA YÜKSEL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
COMPUTER ENGINEERING

SEPTEMBER 2008

Approval of the thesis

**“SHARING ELECTRONIC HEALTHCARE RECORDS ACROSS
COUNTRY BORDERS”**

submitted by **Mustafa Yüksel** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering, Middle East Technical University** by,

Prof. Dr. Canan Özgen _____
Dean, **Graduate School of Natural and Applied Sciences**

Prof. Dr. Volkan Atalay _____
Head of Department, **Computer Engineering**

Prof. Dr. Asuman Doğaç _____
Supervisor, **Department of Computer Engineering, METU**

Examining Committee Members:

Prof. Dr. İsmail Hakkı Toroslu _____
Department of Computer Engineering, METU

Prof. Dr. Asuman Doğaç _____
Department of Computer Engineering, METU

Dr. Nihat Yurt _____
Department of Information Processing, Ministry of Health

Assoc. Prof. Dr. Nihan Kesim Çiçekli _____
Department of Computer Engineering, METU

Yıldıray Kabak _____
SRDC Ltd.

Date: _____

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name : Mustafa Yüksel

Signature :

ABSTRACT

SHARING ELECTRONIC HEALTHCARE RECORDS ACROSS COUNTRY BORDERS

Yüksel, Mustafa

M.S., Department of Computer Engineering

Supervisor: Prof. Dr. Asuman Doğaç

September 2008, 141 pages

Today, the application of information and communication technologies to healthcare is on the agenda of many countries. The main aim is to make Electronic Healthcare Records (EHR) of a patient accessible anywhere at any time to all authorized users. This is even valid in the cross-border case; the European Commission has published eHealth interoperability recommendations to the EU Member States, in which the RIDE Project contributed, for the purpose of an interoperable European Health Network.

Interoperable cross-border clinical data exchange is an ambitious goal with some challenges, the most obvious one being the variety of standards. This issue gets more complicated with the locally developed standards and coding systems. Each country has its own set of standards and it is not reasonable to make all possible combinations of mappings among them during multi-party EHR exchange. Instead, what needs to be done is keeping the legacy infrastructures of the participants and agreeing on a set of common EHR standards and coding systems. Then, each country shall develop “Adapters” transforming local EHR instances to the commonly agreed formats which will most probably be based on widely accepted standards such as HL7 CDA. This approach enables the structure level interoperability. As the second step, in order to achieve semantic interoperability, coded terms from locally defined coding systems shall be translated to international counterparts.

In this thesis, our methodology is confirmed on Turkey’s National Health Information System. “Transmission Schemas” are automatically transformed to HL7 v3 CDA R2 and

CEN EN 13606 standard formats. The local coded terms are translated by developing a mapping platform based on Unified Medical Language System (UMLS).

Keywords: eHealth, Electronic Healthcare Records, National Health Information System, semantic interoperability

ÖZ

ELEKTRONİK SAĞLIK KAYITLARININ ÜLKELER ARASI PAYLAŞIMI

Yüksel, Mustafa

Yüksek Lisans, Bilgisayar Mühendisliği Bölümü

Tez Yöneticisi: Prof. Dr. Asuman Doğaç

Eylül 2008, 141 sayfa

Günümüzde sağlık alanına bilgi ve iletişim teknolojilerinin uygulanması pek çok ülkenin gündemindedir. Ana amaç bir hastanın Elektronik Sağlık Kayıtlarını (ESK) her zaman her yerden yetkili şahıslara erişilebilir kılmaktır. Bu, ülkeler-arası durum için de geçerlidir; Avrupa Komisyonu birlikte çalışabilir bir Avrupa Sağlık Ağı için üye ülkelere RIDE projesinin de katkıda bulunduğu e-Sağlık'ta birlikte çalışabilirlik tavsiyeleri yayınlamaktadır.

Birlikte çalışabilir ülkeler-arası klinik veri paylaşımı bazı zorlukları olan iddialı bir hedefdir. Bu zorluklardan en belirgin olanı standartların çeşitliliğidir. Bu durum yerel olarak geliştirilmiş standartların ve kodlama sistemlerinin geliştirilmesiyle daha karmaşık bir hal almaktadır. Her ülke kendine has standartlara sahiptir ve çok partili ESK paylaşımı durumunda bu standartların tüm olası eşleştirilmelerinin yapılması mantıklı değildir. Bunun yerine, yapılması gereken katılımcıların miras altyapılarını koruyarak ortak ESK standartları ve kod sistemleri üzerinde anlaşmaktır. Sonra, her ülke kendi yerel ESK'lerini büyük ihtimalle HL7 CDA gibi geniş kabul görmüş standartlar arasından seçilecek olan ortak biçime dönüştürecek "Adaptörler" geliştirmelidir. Bu yaklaşım yapısal seviye birlikte çalışabilirliği sağlamaktadır. İkinci adım olarak, anlamsal birlikte çalışabilirliği başarmak için yerel kodlama sistemlerinde tanımlanan kodlanmış terimlerin evrensel eşleniklerine çevrilmesi gerekmektedir.

Bu tezde yöntemimizin uygulanabilirliği Türkiye'nin Ulusal Sağlık Bilgi Sistemi üzerinde doğrulanmaktadır. "Gönderim Şemaları" otomatik olarak HL7 v3 CDA R2 ve CEN 13606'ya

dönüştürülmektedir. Yerel kodlanmış terimler de Birleştirilmiş Tıbbi Dil Sistemi (UMLS) tabanlı eşleştirici platformu ile çevrilmektedir.

Anahtar Kelimeler: e-sağlık, Elektronik Sağlık Kayıtları, Ulusal Sağlık Bilgi Sistemi, anlamsal birlikte çalışabilirlik

ACKNOWLEDGMENTS

I would like to express my sincere gratitude and appreciation to Prof. Dr. Asuman Dođaç for her encouragement and support throughout this study.

I would also like to convey thanks to jury members for their valuable comments on this thesis.

I am deeply grateful to my family for their love and support. Without them, this work could not have been completed.

I am also grateful to my dear friend Eda Kılıç for her love, continued motivating support and welcomed presence.

I am deeply indebted to my friends, Mehmet Olduz, İbrahim Taşyurt, Yıldray Kabak, Gökçe Banu Laleci Ertürkmen and all the other colleagues at the Software Research and Development Center, whose help, stimulating suggestions and encouragement helped me in all the time of research for and writing of this thesis.

I also thank the Scientific and Technological Research Council of Turkey (TÜBİTAK) for providing the financial means throughout this study.

Finally, my special thanks go to my brothers Mustafa Gökçe Baydoğan, Çağlar Kılıcıođlu and Çağlar Ata for their help, support and cheerful presence through the course of this study. Thanks for giving me a shoulder to lean on whenever I need.

To my family..

TABLE OF CONTENTS

ABSTRACT	iv
ÖZ	vi
ACKNOWLEDGMENTS	viii
DEDICATON	ix
TABLE OF CONTENTS	x
LIST OF TABLES	xii
LIST OF FIGURES	xiii
LIST OF ABBREVIATIONS	xvi
CHAPTER	
1 INTRODUCTION	1
2 BACKGROUND ON ENABLING TECHNOLOGIES AND STANDARDS	9
2.1 Health Level Seven (HL7)	9
2.1.1 HL7 Version 3	10
2.1.2 HL7 Clinical Document Architecture (CDA)	16
2.2 CEN EN 13606	19
2.3 Medical Terminologies and Terminology Servers	22
2.4 National Health Information System of Turkey	25
2.5 The Extensible Stylesheet Language (XSL) Family	38
3 MAPPING THE NATIONAL ELECTRONIC HEALTHCARE RECORDS TO INTERNATIONAL STANDARDS - CASE STUDY: TURKEY	43
3.1 The Incompatible Changes Done on the Original CDA Schema for Devel- oping “Transmission Schemas” in NHIS, Turkey	44
3.2 The Developments for Enabling CDA Conformance of NHIS Transmission Schemas	60

3.2.1	Example XSLT Rules	61
3.2.2	Implementation Details	63
4	AUTOMATIC MAPPING OF LOCAL CODED TERMS IN EHR DOCUMENTS TO INTERNATIONAL COUNTERPARTS	69
4.1	UMLS Knowledge Source Server Web Service API	70
4.1.1	Authentication Web Service	71
4.1.2	UMLSKS Web Service	72
4.1.3	UMLSKS Web Service Operations	72
4.2	The Developments for the Architecture Enabling Automatic Coded Value Translation in CDA Documents	74
4.2.1	Locating the Coded Values	75
4.2.2	Building the Local Database	77
4.2.3	Translation Methodology	81
4.2.4	Implementation Details	83
5	MAPPING HL7 CDA R2 TO CEN EN 13606	87
5.1	Developing the XML Schema Definition for CEN EN 13606 Reference Model	88
5.2	Mapping Process	90
5.2.1	Example Mappings and XSLT Rules	93
5.3	Implementation Details	99
5.4	Observations	99
6	RELATED WORK	103
7	CONCLUSION AND FUTURE WORK	106
	REFERENCES	110
A	AN EXAMPLE EXAMINATION TRANSMISSION SCHEMA INSTANCE	119
B	THE CDA CONFORMANT EXAMINATION TRANSMISSION SCHEMA IN- STANCE	131

LIST OF TABLES

TABLES

Table 2.1	Main hierarchy components of the EHR Extract Reference Model	22
Table 4.1	The Result of the findLUI Operation with Search String “asthma”	74
Table 4.2	Fetus heart beat code system	77
Table 4.3	Diagnosis type code system	78
Table 4.4	“CodedValue” Instances for “Diagnosis Type” Code System	80
Table 4.5	Universal Variants for the “Diagnosis Type” Code System	80
Table 5.1	Main hierarchy components of the EHR Extract Reference Model	92

LIST OF FIGURES

FIGURES

Figure 1.1	The European Healthcare Network Proposed by the RIDE project . . .	4
Figure 1.2	The Architecture of the Transformation Environment for NHIS, Turkey	7
Figure 2.1	RIM back-bone classes	11
Figure 2.2	RIM classes in action	13
Figure 2.3	Message Structure Generation in HL7 Version 3	13
Figure 2.4	Major components of a CDA document	17
Figure 2.5	A Sample CDA Section a Skin Care Exam Event	20
Figure 2.6	Saglik-Net as of July 2007	26
Figure 2.7	Minimum Health Data Sets in the NHDD	29
Figure 2.8	The Examination Transmission Schema from the NHDD	31
Figure 2.9	The Beginning of the Examination Transmission Schema	32
Figure 2.10	An Example First-level Section for the Examination TS	32
Figure 2.11	An Example Second-level Section for the Examination TS	33
Figure 2.12	Mapping NHDD Concepts to HL7 v3 CDA R2	34
Figure 2.13	Example Transmission Wrapper and Control Act Wrapper from the NHIS	35
Figure 2.14	Interaction Diagram of NHIS HL7 Web Services	36
Figure 2.15	Basic Elements of XSL Transformation	40
Figure 2.16	Sample XPath expression applied to an XML document	41
Figure 3.1	The First Level Elements of the CDA: CDA Header	46
Figure 3.2	The Beginning of the Examination Transmission Schema	46
Figure 3.3	A Conformant “ClinicalDocument” Introduction	47
Figure 3.4	An Example Invalid recordTarget for Newborn Registration	47
Figure 3.5	The Valid recordTarget for Newborn Registration	48

Figure 3.6	An Invalid declarationPatientRole Example	49
Figure 3.7	An Example custodian Block from the NHIS	49
Figure 3.8	An Example primaryInformationRecipient Block from the NHIS	50
Figure 3.9	An Example indirectTarget Block from the NHIS	50
Figure 3.10	An Example replacementOf Block from the NHIS	51
Figure 3.11	High Level View of the CDA Body	52
Figure 3.12	Examination Transmission Schema components	53
Figure 3.13	An Example dischargeDataset section	53
Figure 3.14	Mapping NHDD Concepts to HL7 v3 CDA R2	54
Figure 3.15	An Excerpt from the examinationDataset section	54
Figure 3.16	The Corresponding Conformant CDA Block for examinationDataset	55
Figure 3.17	An Example author Block from the testResultDataset	55
Figure 3.18	The Corresponding Conformant CDA Block for author	56
Figure 3.19	The Graphical View of the Schema of CDA Section	57
Figure 3.20	An Example dischargeDiagnosisSection from the NHIS	58
Figure 3.21	The Corresponding Conformant CDA section for dischargeDiagnosis- Section	58
Figure 3.22	An Example procedureSection from the NHIS	59
Figure 3.23	The Corresponding Conformant CDA section for procedureSection	59
Figure 3.24	XSLT Rule for Newborn Registration recordTarget	62
Figure 3.25	XSLT Rule for primaryInformationRecipient	63
Figure 3.26	The Conformant Output for primaryInformationRecipient	63
Figure 3.27	An Excerpt from the Discharge Data Set XSLT File	64
Figure 3.28	The Start-up Screen of the Transformation Environment	66
Figure 3.29	Opening New Projects and Input Files	67
Figure 3.30	NHIS2CDA Transformer in Action	68
Figure 4.1	Establish a Connection to the Authentication Web Service	72
Figure 4.2	Obtaining UMLS Data from UMLSKS Web Service	73
Figure 4.3	XPath Expression to Locate All the CodedValues	75
Figure 4.4	An Excerpt from the Output of the XPath Expression on the CDA Instance	76
Figure 4.5	CodedValue Class Diagram for Persistent Objects	79
Figure 4.6	CDA CodedValue Universalizer in Action	84

Figure 4.7	Some Example Mapped Coded Values together with the Corresponding Local Coded Values	86
Figure 5.1	The UML Class Diagram of CEN EN 13606-1 Reference Model	89
Figure 5.2	High Level View of the EN 13606 Reference Model Schema	91
Figure 5.3	Mapping CDA (generated from Transmission Schemas) to EN 13606-1	93
Figure 5.4	An Example recordTarget from CDA	94
Figure 5.5	The Corresponding Conformant EN 13606-1 demographic_extract for recordTarget	95
Figure 5.6	XSLT Rule for Transforming recordTarget to demographic_extract (Part 1)	96
Figure 5.7	XSLT Rule for Transforming recordTarget to demographic_extract (Part 2)	97
Figure 5.8	An Example observation from CDA	97
Figure 5.9	A Small part of the Corresponding Conformant EN 13606 ENTRY for observation	98
Figure 5.10	CDA to CEN 13606 Transformer in Action	100

LIST OF ABBREVIATIONS

CDA	Clinical Document Architecture	SNOMED	Systematized Nomenclature of Medicine
CEN	The European Committee for Standardization	UML	Unified Modeling Language
CUI	Concept Unique Identifier	UMLS	Unified Medical Language System
DBB	Doctor Data Bank	UMLSKS	The UMLS Knowledge Source Server
D-MIM	Domain Message Information Model	UUID	Universally Unique Identifier
EHR	Electronic Healthcare Record	W3C	The World Wide Web Consortium
EHR.com	CEN/TC251 Electronic Health Record Communication	WHO	World Health Organization
FMIS	Family Medicine Information System	WSDL	Web Service Definition Language
HL7	Health Level Seven	XML	The Extensible Markup Language
HMD	Hierarchical Message Description	XPath	The XML Path Language
HTML	HyperText Markup Language	XSD	XML Schema Definition
ICD	The International Statistical Classification of Diseases	XSL	The Extensible Stylesheet Language Family
LOINC	Logical Observation Identifiers Names and Codes	XSLT	XSL Transformations
LUI	Lexical Unique Identifier	XSL-FO	XSL Formatting Objects
MeSH	Medical Subject Headings		
MHDS	Minimum Health Data Set		
NHIS	National Health Information System		
NLM	The United States National Library of Medicine		
RIM	Reference Information Model		
R-MIM	Refined Message Information Model		

CHAPTER 1

INTRODUCTION

Today many countries and also some individual regions/provinces are developing their national or regional electronic healthcare infrastructures for collecting and sharing administrative and clinical data about the patients, healthcare professionals and organizations. While some countries like England, Canada and the Netherlands are on the way for a long time, some other countries like Mexico and Poland seem to be at the very beginning of establishing a complete national electronic healthcare infrastructure. But, in any case, today almost all countries are aware of the importance and benefits of integrating an electronic communication network into their legacy health systems for the purpose of efficient clinical and administrative data exchange.

The successfully completed RIDE project (A Roadmap for Interoperability of eHealth Systems in Support of COM 356 with Special Emphasis on Semantic Interoperability) [66] that is supported by the European Commission provided extensive analysis of the eHealth systems of all European countries and some other countries like the USA, Canada and Australia in two phases; the first one entitled “European Best Practices in Providing Semantic Interoperability in eHealth Domain” [67] representing the status in the early months of 2006 and the second one entitled “European Good Practices” [68] representing the status in the second half of 2007 supported with comparison of countries under several categories such as patient identifiers or Electronic Healthcare Records. These studies show that all countries are defining the medical content for the purpose of exchange among healthcare organizations and/or national authorities such as the Ministry of Health according to their own requirements. Based on these requirements and the purpose of usage, different names are attached to these defined contents; Emergency Data Set, Minimum Health Data Set, Patient Summary, etc. However, most of the time they are used to achieve whole or parts of an Electronic Healthcare Record.

The Electronic Healthcare Record (EHR, also called Electronic Health Record) is defined by Iakovidis [38] as “digitally stored health care information about an individual’s lifetime with the purpose of supporting continuity of care, education and research, and ensuring confidentiality at all times”. There are several EHR content standards under development which aim to provide standard interface to existing proprietary systems. These efforts include Health Level Seven Clinical Document Architecture [6], CEN EN 13606 Electronic Health Record Communication [11] and openEHR [61]. Such standards define the structure and markup of the clinical content to make EHR exchange interoperable. Eichelberg et. al. [28] presents an extensive survey and analysis of EHR standards.

After having a documented list of the necessary data elements to occur within the EHR content, which is usually called the Health Data Dictionary, the countries seek ways of representing the EHR content in wired format. There are two options at this point; to adopt an international EHR standard such as HL7 CDA, CEN EN 13606 or openEHR and apply the required national restrictions and extensions, or to develop a totally local EHR specification from scratch by defining all properties freely according to the requirements. The same kind of selection needs to be done at the messaging level as well. Examples to both cases are presented in the previously mentioned RIDE deliverables [67, 68]. While projects like Health Infoway in Canada [32], NHS in the United Kingdom [59] and Dossier Médical Personnel (DMP) in France [23] opted for the first choice and adopted HL7 CDA according to their needs, Belgium preferred to develop a local specification entitled “Kind Messaging for the Electronic Healthcare Records” (Kmehr) [44].

In parallel with these regional and national efforts, there are also initiatives and plans for achieving EHR interoperability at a higher level, such as the European level. In Europe, the European Commission grants projects for developing roadmaps for eHealth interoperability (one example is the RIDE project), organizes several conferences and workshops with the attendance of national authorities from the Member States and publishes recommendations for cross-border eHealth interoperability [63, 64] in which the successful results of the granted projects are embedded as well. Although the EU cannot mandate any laws or rules related with the health and healthcare policies of its member states since healthcare and eHealth are considered to be internal matters of the Member States, with all these efforts it expects that the Member States take these recommendations into consideration while developing their national eHealth infrastructures. The renewed deadline for establishing EU-wide EHR exchange is 2015. As another higher level effort example, the USA is trying to share EHRs among the Regional Health Information Organizations through National Health Information

Network (NHIN) [57].

Now, it is clear that local developments are being done in order to exchange EHRs in many countries and on the other hand, there is a definite need for enabling interoperability at higher levels, e.g. the European Level. It is undeniable that different countries will have different local requirements. Moreover, every country is totally free in its internal decisions, especially related with such a private issue; health. For these reasons, it is not logical to expect all countries to accept just one global model as it is. Interoperability at higher levels is only achievable if the internal infrastructures of the countries are kept as they are but some additional mechanisms are developed in order to enable interoperable cross-border clinical data exchange.

In the second version of its roadmap [69], RIDE has proposed such an architecture entitled the European Healthcare Network which was also recognized by the eHealth Interoperability Recommendation [63] of the European Commission. This architecture aims to realize integration of different medical systems in a configurable network of interconnected organizations so that the proposed European Healthcare Network is flexible by design, allowing smooth involvement of the legacy systems within the network. This is achievable by the Service Oriented Architecture (SOA) design approach. Indeed, RIDE framework does not interfere with the low level implementation details of the use case actors and services presented in Figure 1.1 in the European Healthcare Network, as long as their interacting functionalities are available to the interested parties with a standard-compliant interface.

This architecture states that, clinical data being exchanged within a country may be in any format/standard/language and need not be directly understandable by the other countries but, while sharing this data with other countries it has to be mapped to an international standard recognized by all the participating bodies. The European Commission has efforts in this respect as well; there is ongoing work in defining the structure and semantics of a European level Emergency Data Set.

In this thesis, we develop a methodology to realize such an architecture. We have identified two major requirements for this purpose:

- When common EHR format(s) or standard(s) are ready, the first thing that needs to be done is developing an Adaptor that will automatically transform the local EHR instances to the commonly agreed format. This is the only manual process that needs to be done by all the participants of the network and unfortunately its extent is different for all countries, depending on the differential distance between the local implemen-

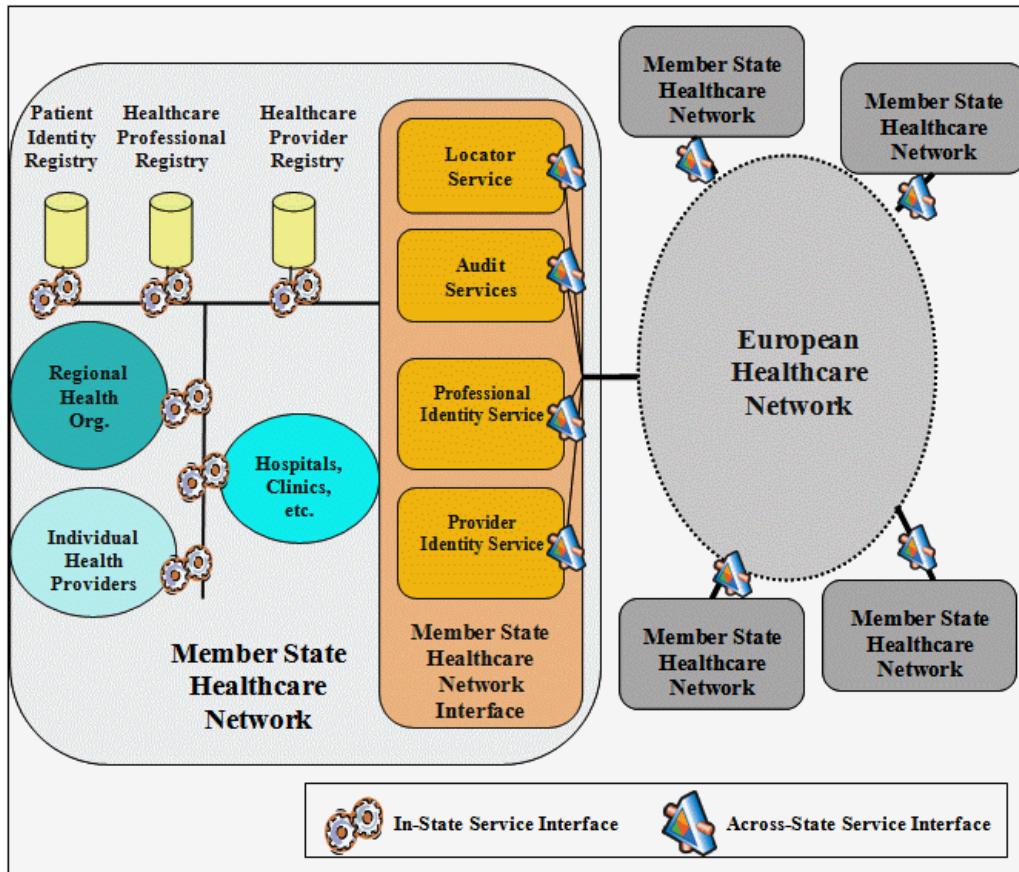


Figure 1.1: The European Healthcare Network Proposed by the RIDE project

tation and the commonly agreed format. This first step enables the structure level interoperability.

- Especially in a multilingual network, as in the case of Europe, it is definite that the mapping of the local EHR content to the commonly agreed format, which we call structure level conformance, will not suffice. It is undeniable that locally developed coded terms exist in the local EHRs and in order to achieve true semantic interoperability, those terms shall be translated to a code system or language that the other participants in the network can understand. This necessitates a Terminology Server based automatic translation environment for the coded terms. The aim of this step is enabling semantic (content) interoperability.

In our work, we prove that our methodology is applicable by developing solutions for providing cross-border data exchange support to the National Health Information System of Turkey [37]. NHIS aims to provide a nation-wide infrastructure for sharing the Electronic Health Records (EHRs). The current implementation supports the transfer of EHRs,

called the “Transmission Schema” instances from the Family Medicine Information Systems (FMIS), Hospital Information Systems (HIS) and Laboratory Information Systems to the NHIS servers at the Ministry of Health (MoH). Currently the system is being widely tested by the vendors and starting by September 30th, 2008 all the public healthcare organizations are obliged to send their EHRs to MoH through NHIS Web Services.

Our achievements can be summarized as follows:

- Turkey adopted HL7 CDA as the EHR standard but unfortunately, in order to meet all national requirements directly at the schema level, many changes have been made on the original CDA Schema that broke the conformity of local “Transmission Schemas” to CDA. While developing the schemas for the HL7 services, the consortium that is responsible from the implementation of the NHIS preferred to start from the Refined Message Information Model (R-MIM) of CDA instead of directly getting and restricting CDA Schema.

A conformant CDA document should at a minimum validate against the CDA Schema. The modifications that have been done during the localization process have broken this conformity. The generated schemas are completely HL7 v3 and CDA R-MIM conformant but still it is not possible to realize cross-border data exchange with the current status. This is an interoperability problem.

In order to overcome these problems, first analysis of all the incompatible changes that have been made on the original CDA Schema have been realized and these modifications are clearly documented. Then, an Adaptor based on XSL Transformations (XSLT) [103] has been developed to automatically generate CDA conformant EHR documents from the “Transmission Schema” instances.

CDA is the most dominant EHR standard today. It is for sure that the commonly agreed EHR format mentioned in our methodology will depend on one or more of these promising EHR standards since they have already been well accepted.

Our implementation will be deployed as a Web Service that will be acting as a proxy between the Ministry of Health and the cross-border countries capable of healthcare data exchange. Another use of the Adaptor is on the NHIS clients’ side; the clients can store the generated Transmission Schemas in native CDA format so that when a change has been done on the Web Service schemas, the previous records will still be accessible.

- By realizing the conformance of the Transmission Schemas to HL7 CDA (and CEN EN 13606 in the next item), NHIS EHR instances are now processable by everyone who is able to conform to these standards. This was truly an important step in the interoperability process. However, there are still things to be done at the semantic level of the documents. The contents of the messages are in Turkish and to achieve cross-border semantic interoperability, it is necessary to translate the elements used in the documents as well.

For this purpose, a Terminology Server based modular architecture enabling automatic mapping of the local coded terms that appear in the NHIS Transmission Schema instances has been developed. As the Terminology Server, the Unified Medical Language System (UMLS) [81] Knowledge Source Server has been used. UMLS provides secure Web Services for querying the server for finding clinical concepts, retrieving concept details, mapping clinical terms, etc. These Web Services are used for automatic translation. For instance, now it is possible to automatically replace the code “EKTANT” meaning “Ek Tani” (secondary diagnosis) from the local “Tani Tipi” (diagnosis type) code system with code “85097005” meaning “Secondary diagnosis” from the universal “SNOMED CT” [75] code system automatically.

- CDA is the most widely used EHR standard today and CDA conformity has been achieved within the scope of the first task. Although not as popular as CDA, CEN EN 13606 is the European standard for EHRs. In order to enrich the interoperability support of EHRs Turkey will exchange with other countries, another XSLT based Adaptor has been successfully developed for automatically transforming NHIS Transmission Schema instances to CEN EN 13606 conformant EHRs.

Unlike CDA, CEN 13606 does not have a common schema that is processable by the computers. For this reason, first the XML Schema Definition of 13606 reference model is realized. Then, in order to provide a more generic transformation system, the XSL Transformation rules have been written so that CDA conformant EHRs that are transformed from the Transmission Schemas by the first Adaptor are transformed into 13606 conformant EHRs. This way, development of Transmission Schema specific XSLT files is prevented, which had to be done in the CDA transformer. Instead, some Transmission Schema specific controls within the generic XSLT sufficed.

As a result, one of the first implementations of the CEN 13606 has been realized. More critically, our literature survey shows that Turkey is the first country to support cross-

border clinical data exchange with more than one international standard, that is both as HL7 CDA and CEN EN 13606 at the same time. We have provided a detailed list of our observations gained from the challenges that we faced during the implementation of 13606.

Our work shows that when a local implementation is already based on international standards as much as possible (as in the case of Turkey), the mapping efforts decrease enormously. Thus, development of a proprietary format instead of adopting a standard decreases interoperability.

The architecture of the system, which we call Transformation Environment for NHIS of Turkey, is presented in Figure 1.2. The details of the components are presented in the related chapters.

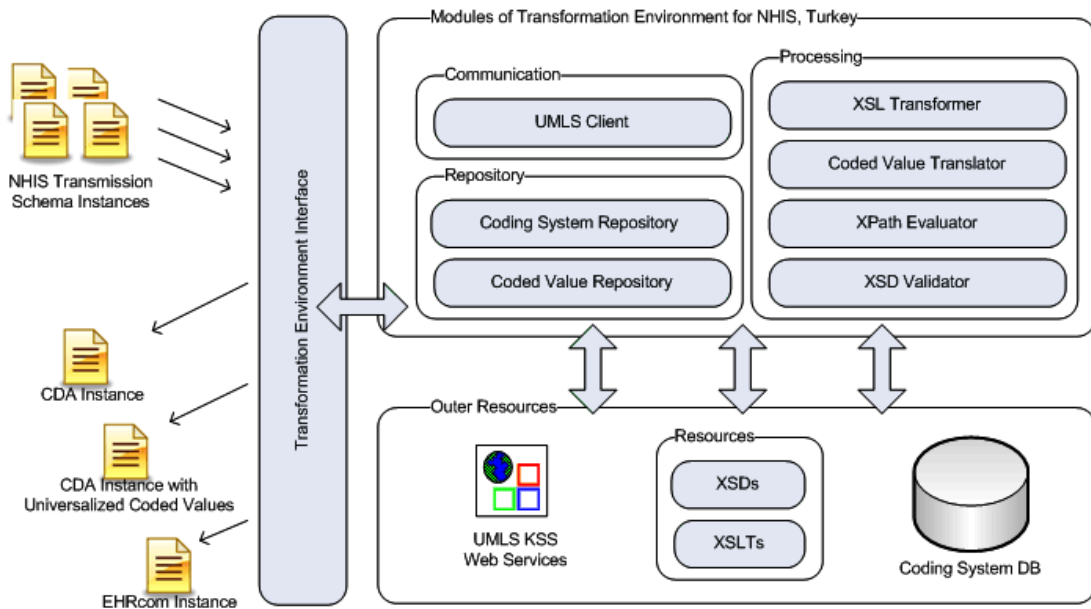


Figure 1.2: The Architecture of the Transformation Environment for NHIS, Turkey

The rest of this thesis is organized as follows: Chapter 2 summarizes the enabling technologies and standards. In Chapter 3, development of the Adaptor for automatically transforming NHIS Transmission Schema instances to HL7 CDA R2 conformant EHRs is explained. In Chapter 4, UMLS based architecture enabling automatic mapping of the local coded terms in EHRs to international counterparts is presented. Chapter 5 presents the development of the Adaptor for automatically transforming CDA instances to CEN 13606

instances. Related work in the literature in comparison with our methodology is presented in Chapter 6. Finally, Chapter 7 concludes the thesis and suggests possible future research directions.

CHAPTER 2

BACKGROUND ON ENABLING TECHNOLOGIES AND STANDARDS

2.1 Health Level Seven (HL7)

Health Level Seven (HL7) [34] is a not-for-profit ANSI [2] accredited Standards Developing Organization. Primary goal of HL7 is to provide standards for the exchange of clinical and administrative data among healthcare systems. HL7 provides standards for interoperability that improve care delivery, optimize workflow, reduce ambiguity and enhance knowledge transfer among all stakeholders, including healthcare providers, government agencies, the vendor community, fellow SDOs and patients.

“Level Seven” refers to the highest level of the International Organization for Standardization (ISO) [41] communications model for Open Systems Interconnection (OSI) [106]; the application level. The application level addresses definition of the data to be exchanged, the timing of the interchange, and the communication of certain errors to the application.

HL7 defines message structures and trigger events for this purpose. A trigger event causes the exchange of messages between these application systems, that is, when an event occurs in an HL7 compliant system, an HL7 message is sent to other HL7 compliant systems including the necessary data required by the receiving systems.

HL7 focuses on the interface requirements of the entire health care organization, while most other efforts focus on the requirements of a particular department. Moreover, HL7 develops a set of protocols that is both responsive and responsible to its members. The group addresses the unique requirements of already installed hospital and departmental systems, some of which use mature technologies.

Although HL7 Version 2.x is the most widely implemented healthcare informatics stan-

dard in the world, it has several problems that are explained in the following section. These problems led the HL7 organization to develop a more definitive version almost from scratch, namely the HL7 Version 3.

2.1.1 HL7 Version 3

Offering lots of optionality and thus flexibility, the version 2.x series of messages were widely implemented and were successful to some extent. These messages evolved over several years using a “bottom-up” approach that has addressed individual needs through an evolving ad-hoc methodology. HL7’s wide adaptation is largely attributable to its flexibility. It contains many optional data elements and data segments, making it adaptable to almost any site. However, while providing great flexibility, its optionality also makes it impossible to have reliable conformance tests of any vendor’s implementation and also forces implementers to spend more time analyzing and planning their interfaces to ensure that both parties are using the same optional features. As a result, most of the time it is not possible to interconnect two applications who claim to be HL7 v2.x compliant.

Version 3 addresses these kinds of issues by using a well-defined methodology [48] based on a reference information model. It will be the most definitive standard to date. Using rigorous analytic and message building techniques and incorporating more trigger events and message formats with very little optionality, HL7’s primary goal for Version 3 is to offer a standard that is definite and testable, and provide the ability to certify vendors’ conformance. Version 3 uses an object-oriented development methodology and a Reference Information Model (RIM) [70] to create messages. The RIM is an essential part of the HL7 Version 3 development methodology, as it provides an explicit representation of the semantic and lexical connections that exist between the information carried in the fields of HL7 messages.

HL7 v2.5 and the previous 2.x versions are ANSI approved standards but Version 3 is still in the balloting phase. The balloting is done by the members in periods of three months and all the ballot versions are available through the HL7 Web site.

Reference Information Model (RIM)

An information model is a structured specification of the information within a specific domain of interest. It expresses the classes of information required and the properties of those classes, including attributes, relationships, constraints, and states.

The Reference Information Model (RIM) [70] is the cornerstone of the HL7 Version 3 development process. An object model created as part of the Version 3 methodology, the RIM is a large pictorial representation of the clinical data and identifies the life cycle of events that a message or groups of related messages will carry. It is a shared model between all the domains and as such is the model from which all domains create their messages.

The RIM is comprised of six “back-bone” classes as shown in Figure 2.1. Every happening documented in the healthcare domain is represented by the Act class. Physical things and beings that take part in healthcare are represented by the Entity class. The Role class establishes the roles that entities play as they participate in healthcare acts. The Participation class defines the context for an Act by defining the relationship between Act and Role classes. The ActRelationship class defines the relationship between two instances of the Act class. Similarly, the RoleLink defines the relationship between two instances of the Role class.

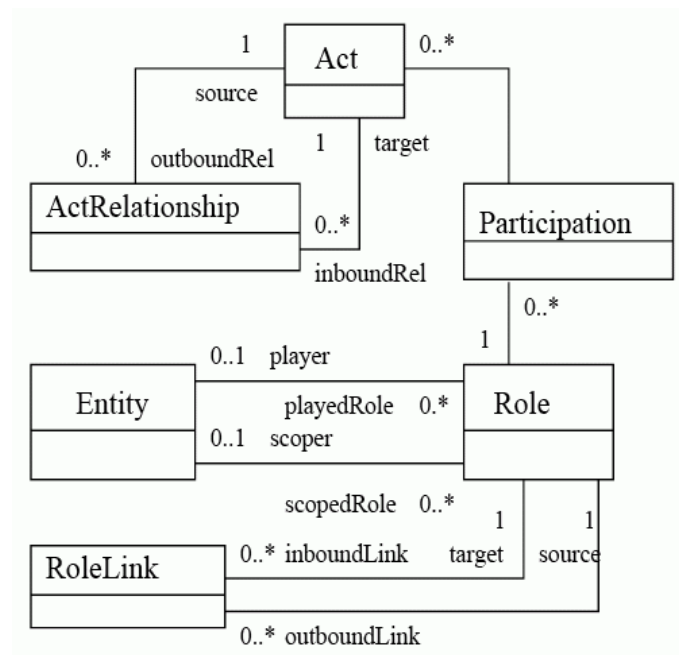


Figure 2.1: RIM back-bone classes

The Act, Entity and Role classes are further specialized to subclasses. In the HL7 representation, a new subclass is added to the RIM only when new attributes or associations are needed which are not available in the super classes.

A specialized concept which needs no further attributes or associations is represented by assigning a unique code in the controlling vocabulary to specific attributes. Therefore, these three classes include the following coded attributes, which serve to further define the concept being modeled:

- *classCode* (in Act, Entity and Role) represents the exact class or concept intended, whether or not that class is represented as a class in the RIM hierarchy.
- *moodCode* (in Act) further delineates the Act instance as an occurrence, intent, goal, etc.
- *determinerCode* (in Entity) distinguishes whether the class represents an instance or a kind of Entity.
- *code* (in Act, Entity and Role) provides for further classification within a particular classCode value, such as a particular type of observation within the Observation class. It should be noted that code should be consistent with the classCode.

The other three RIM back-bone classes - Participation, ActRelationship and RoleLink - are not represented by generalization-specialization hierarchies. Nevertheless, these classes represent a variety of concepts, such as different forms of participation or different kinds of relationships between acts. These distinctions are represented by a *typeCode* attribute that is asserted for each of these classes. For example, the “author” concept which describes the party that originates the Act can be derived by assigning the “AUT” to the Participation class of the RIM.

An example application of these RIM classes with the rough adaptation of Pregnant Observation Minimum Health Data Set from the National Health Data Dictionary of Turkey [86] is represented in Figure 2.2. In this example, the main Act class is the “Pregnant Observation” which has three participations as the subject, performer and observer. “Pregnant Observation” may be the cause for a “Procedure”. Moreover, there is a “direct authority over” relationship between the performer and observer doctors.

Message Development Framework (MDF)

HL7 Message Development Framework (MDF) [48] provides a methodology for developing HL7 messages for HL7 Version 3.0 and beyond. It is a reference manual that describes each stage of building messages, how to use the tools that support this process, and the concepts involved. MDF is used by members of HL7 Working Group.

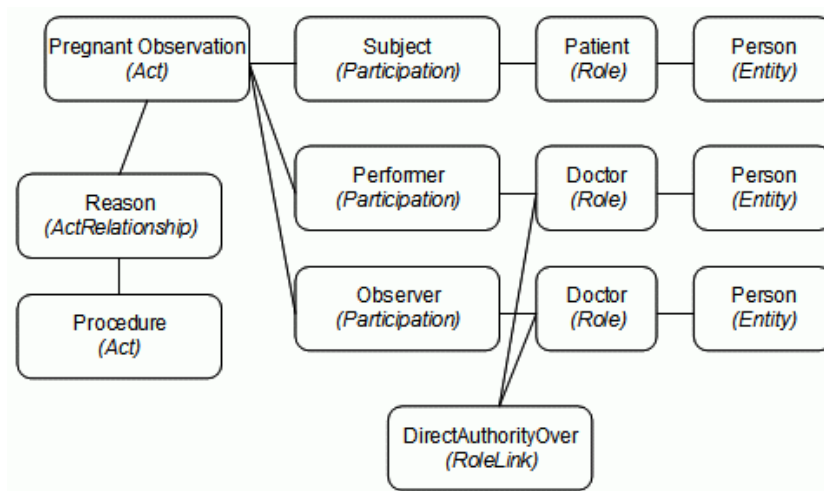


Figure 2.2: RIM classes in action

In HL7 Version 3, RIM is the source of all message contents. Figure 2.3 shows how message structures are defined based on the RIM and the MDF.

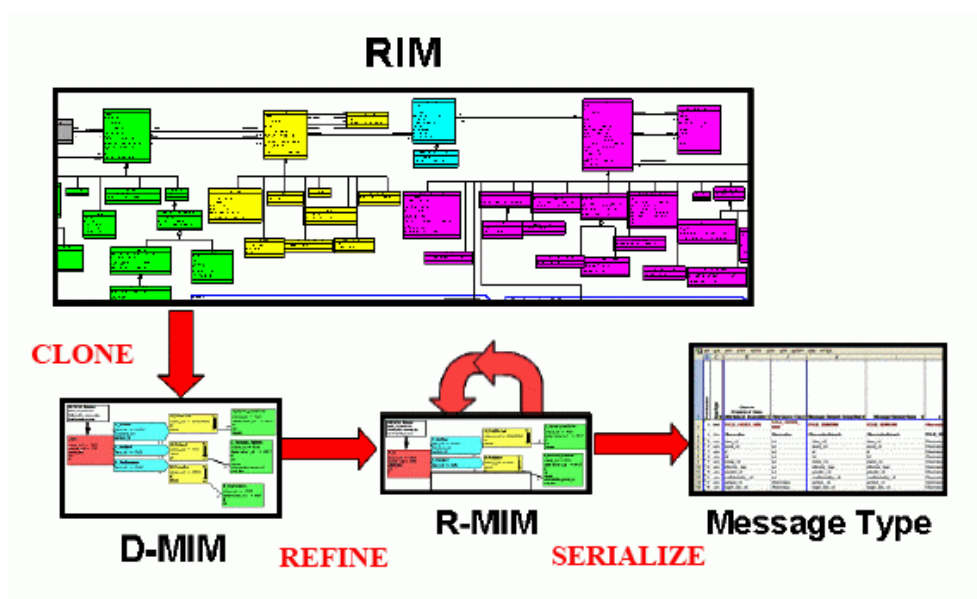


Figure 2.3: Message Structure Generation in HL7 Version 3

The first step to generate the message structures from the RIM is to derive the Domain Message Information Model (D-MIM) [22] which is the subset of the RIM that includes a fully expanded set of class clones, attributes and relationships used to create messages for

any particular domain [88]. D-MIM is derived from RIM in such a way that only the required classes, attributes and relationships for building the messages for a particular domain are included. Furthermore multiple specializations of the same RIM class may appear in a diagram with different constraints or associations. Such classes are called as clones of the RIM class.

Next step is to build the Refined Message Information Model (R-MIM) [71] which defines the information content for one or more Hierarchical Message Descriptions (HMDs) [36]. An R-MIM is derived from a D-MIM by including the necessary classes, attributes and associations used in set of messages derived from the HMDs. The result of the Version 3 process is the Hierarchical Message Definition (HMD) which is the tabular representation of the sequence of elements (i.e., classes, attributes and associations) represented in an R-MIM. The primary goal of HMD is to define the message structure without reference to any implementation technology.

HL7 also provides an XML Implementable Technology Specification [96] to express HMD in XML Schema Definitions (XSD) [99]. HL7 defines several message structures in various domains including account and billing, blood bank, clinical genomics, claims and reimbursement, laboratory etc. In addition, HL7 also specifies Clinical Document Architecture which describes the structure and semantics of clinical documents exchanged between care providers.

Refinement, Constraint and Localization

The HL7 methodology uses the Reference Information Model (RIM) and the HL7-specified Vocabulary Domains [89], and the Version 3 Data Type Specification [18] as its starting point. It then establishes the rules for refining these base standards to arrive at the information structures that specify Message Types and equivalent structures in Version 3.

The Refinement, Constraint and Localization [65] specification addresses:

- the “rules” and processes for refining the standard as described in the Message Development Framework (the process leading from RIM to HMD and XSD respectively) through constraint and extension, including which standard artifacts are subject to constraint or extension
- the definition of constraint and localization profiles
- the criteria for establishing a conformance statement

A profile is a set of information used to document system requirements or capabilities from an information exchange perspective. The documentation is expressed in terms of constraints, extensions, or other alterations to a referenced standard or base profile. The categories of profiles in HL7 include annotation, constraint, implementable, conformance, localization and conflicting profiles.

All of these profiles require the documentation of the formal constraints, extensions and annotations that are applied through the message development process. The allowed constraints and annotations that can be applied at each step of the refinement process are described in the Refinement, Constraint and Localization. They fall into six broad categories:

- *Appearance constraints* determine whether a particular element must appear in models or messages derived from the base model, and/or whether the element is precluded from appearing therein. If an element is required, then all elements that derive from it in derived models SHALL also be required.
- *Cardinality constraints* define the number of repetitions that may occur for a given element.
- *Type constraints* apply to attributes and associations. Attributes may be constrained by more specific data types, and associations may be constrained by more specific target classes. The Version 3 Data Type Specification [18] defines all HL7 data types and their properties. It also defines a specialization hierarchy of data types with the understanding that any specialized data type in the hierarchy may be substituted for its more general parent data type in the refinement process. For instance, Entity Name (EN) is parent data type of Organization Name (ON), Person Name (PN) and Trivial Name (TN).
- *Vocabulary constraints* limit the set of concepts that can be taken as valid values in an instance of a coded attribute or data type. If an attribute is derived from data type CD (Concept Descriptor) then it is a coded attribute and is subject to vocabulary constraints. As an example, while selecting the administrative gender from a general gender coding system, only “male” and “female” can be allowed as values by eliminating “undifferentiated”.
- *Other value constraints* provide for the declaration of constraints stated as text and, optionally, as testable expressions to establish “business rules”, and for the assertion of default or fixed values.

- *Annotations* provide further explanations to educate prospective users and/or implementers. These are usually used to enhance the descriptions of the elements of the base specification.

For guaranteeing interoperability, only the authorized HL7 Technical Committees can start the refinement process from the RIM and apply the abovementioned constraints while the implementers of HL7 are recommended to start from the HMD although R-MIM is allowed as well.

Transport Specifications

Until Version 3, HL7 did never deal with OSI layers lower than the application layer but starting with Version 3, HL7 became interested in the transport protocol. The HL7 Message Transport Specifications [79] provide details as to the usage of a variety of communication transports for the exchange of HL7 based content, messages and documents.

Currently HL7 v3 recommends three transport mechanisms to exchange HL7 messages:

1. Web Services Profile [92]
2. ebXML Messaging Profile [26]
3. TCP/IP based Minimum Lower Layer Profile (MLLP) [51]

In these profiles, HL7 provides detailed guidelines on how to implement a messaging infrastructure from four perspectives: Basic, Addressing, Security and Reliability.

2.1.2 HL7 Clinical Document Architecture (CDA)

HL7 Clinical Document Architecture (CDA), previously called Patient Record Architecture (PRA), is a document markup standard that specifies the structure and semantics of a clinical document (such as a discharge summary or progress note) for the purpose of exchange [6]. A clinical document includes clinical observations and services about care events. A valid CDA document is encoded in Extensible Markup Language (XML) and conforms to the CDA Schema which is derived from the CDA Hierarchical Description based on the XML Implementable Technology Specification. The CDA Hierarchical Description is derived from the CDA R-MIM through the process shown in Figure 2.3. In other words, HL7 RIM is the source of the structure and semantics of a CDA document.

So far, HL7 has released two versions of CDA. The CDA Release One (CDA R1) is the first specification derived from the HL7 RIM. It became an ANSI approved HL7 Standard in

2000. The CDA Release Two (CDA R2) became an ANSI-approved HL7 Standard in 2005 [24]. Throughout the rest of this thesis whenever CDA is mentioned, CDA R2 is pointed actually.

A CDA document has two main parts, the header and the body. In the CDA R1, only the header part is derived from the RIM. In the CDA R2, in addition to the header part, the clinical content in the document body is also derived from the RIM. Therefore the CDA R2 model enables the formal representation of clinical statements through CDA Entry classes.

A CDA header defines the context of the document by providing information on authentication, the encounter, the patient, and the involved providers whereas the CDA body includes the clinical report. The body part can be either an unstructured blob or a structured hierarchy which involves one or more section components. Within a section, narrative blocks and CDA entries are defined. Machine-processable clinical statements are represented by these CDA entries whereas the narrative blocks are human readable forms of these clinical statements. Figure 2.4 [6] depicts the major components of a CDA document.

```
<ClinicalDocument>
  ... CDA Header ...
  <structuredBody>
    <section>
      <text>...</text>
      <observation>...</observation>
      <substanceAdministration>
        <supply>...</supply>
      </substanceAdministration>
      <observation>
        <externalObservation>...
        </externalObservation>
      </observation>
    </section>
    <section>
      <section>...</section>
    </section>
  </structuredBody>
</ClinicalDocument>
```

Figure 2.4: Major components of a CDA document

The “ClinicalDocument” is the root element of the document. The header is defined between the <ClinicalDocument> and the <structuredBody> tags. Sections reside in the

“structuredBody” element. Each section can contain a single narrative block located in the “text” element. A narrative block is the human readable portion of the section when rendered with an appropriate stylesheet. Section also contains CDA entries which are used to represent structured content. CDA entries are machine-processable portions of the sections.

CDA entries are derived from the shared HL7 Clinical Statement Model [15] which provides a consistent representation of clinical statements across various Version 3 specifications. The model describes the clinical statements using the following entry classes which are derived from the RIM classes:

- *Act* is derived from the RIM Act class. It is a generic purpose class which is used when the other more specific classes of the model are not appropriate for defining the clinical information.
- *Observation* is derived from the RIM Observation class. It is used for representing clinical observations.
- *ObservationMedia* is derived from the RIM Observation class. It is used for representing multimedia which is logically part of the document.
- *SubstanceAdministration* is derived from the RIM SubstanceAdministration class. It is used for representing medication related events.
- *Supply* is derived from the RIM Supply class. It is used for representing the provision of a material between entities.
- *Procedure* is derived from the RIM Procedure class. It is used for representing procedures.
- *RegionOfInterest* is derived from the RIM Observation class. It is used for referencing specific regions in images.
- *Encounter* is derived from the RIM PatientEncounter class. It is used for representing the interaction between a patient and care provider.
- *Organizer* is derived from the RIM Act class. It is used for grouping clinical statements having a common context.

Figure 2.5 gives a sample CDA section describing a physical examination of the skin in which a rash is identified. The “text” element of the section narratively describes the examination. Then the “entry” element contains clinical statements describing the physical

examination of the skin in more detail in terms of coded values. A region of interest is described using RegionOfInterest entry class which is further related to a hand image taken from an image library.

As it can be seen in Figure 2.5, the section element is coded with a value from LOINC coding system [47] and the observation element is coded with a value from the SNOMED CT coding system [75] which actually means “rash”.

The generic CDA specification can be constrained through the document-level, section-level and entry-level templates. The unconstrained CDA specification is called “CDA Level One”. When section-level templates are applied to an unconstrained CDA document, it is called “CDA Level Two”. “CDA Level Three” is the CDA specification with entry-level (and optionally section-level) templates applied.

In our architecture, first of all, CDA R2 conformant documents are being generated from the “Transmission Schema” instances of National Health Information System of Turkey. The local coded elements appearing in those CDA documents are being automatically translated to universal variants as well and finally, CEN EN 13606-1 conformant EHRs are being generated from CDA instances.

2.2 CEN EN 13606

CEN/TC 251 [77] is the technical committee on Health Informatics of the European Committee for Standardization (CEN) [10]. Its mission is to achieve compatibility and interoperability between independent health systems and to enable modularity by means of standardization.

The CEN Prestandard ENV 13606:2000 Electronic Healthcare Record Communication [13] is a message-based standard for the exchange of electronic healthcare records. The standard defines an EHR information model and does not attempt to specify a complete EHR system; it focuses instead on the interfaces. ENV 13606 was intended to be the first fully-implementable EHR standard but only subsets of it were implemented in some European countries [28]. These implementation experiences showed its uselessness especially related with the optionality problem.

In 2001, CEN/TC 251 decided to revise ENV 13606 into a full European Standard, taking into account the existing implementation experience and to adopt the openEHR [61] archetype methodology with the name EN 13606, also known as Electronic Healthcare Record Communication (EHRcom). CEN EN 13606 consists of five parts:

```

<section>
  <code code="8709-8" codeSystem="2.16.840.1.113883.6.1"
    codeSystemName="LOINC"/>
  <title>Skin Exam</title>
  <text>Erythematous rash, palmar surface, left index
    finger.<renderMultiMedia referencedObject="MM2"/>
  </text>
  <entry>
    <observation classCode="OBS" moodCode="EVN">
      <code code="271807003" codeSystem="2.16.840.1.113883.6.96"
        codeSystemName="SNOMED CT" displayName="Rash"/>
      <statusCode code="completed"/>
      <targetSiteCode code="48856004" codeSystem="2.16.840.1.113883.6.96"
        codeSystemName="SNOMED CT"
        displayName="Skin of palmer surface of index finger">
        <qualifier>
          <name code="78615007" codeSystem="2.16.840.1.113883.6.96"
            displayName="with laterality"/>
          <value code="7771000" codeSystem="2.16.840.1.113883.6.96"
            displayName="left"/>
        </qualifier>
      </targetSiteCode>
      <entryRelationship typeCode="SPRT">
        <regionOfInterest classCode="ROI OVL" moodCode="EVN" ID="MM2">
          <id root="2.16.840.1.113883.19.3.1"/>
          <code code="ELLIPSE"/>
          <value value="3"/>
          <!-- .. -->
          <value value="4"/>
          <entryRelationship typeCode="SUBJ">
            <observationMedia classCode="OBS" moodCode="EVN">
              <id root="2.16.840.1.113883.19.2.1"/>
              <value mediaType="image/jpeg">
                <reference value="lefthand.jpeg"/>
              </value>
            </observationMedia>
          </entryRelationship>
        </regionOfInterest>
      </entryRelationship>
    </observation>
  </entry>
</section>

```

Figure 2.5: A Sample CDA Section a Skin Care Exam Event

1. *Reference Model* specifies the information architecture of the EHR data exchanged between systems and services.
2. *Archetypes Interchange Specification* specifies the Archetype Model and the language which are used to constrain the data.
3. *Reference Archetypes and Term Lists* presents some reference archetypes and specifies the code lists which are used with the standard.
4. *Security* specifies the privileges and regulations necessary to access data.
5. *Interface Specification* defines a set of interfaces by which the artifacts defined in 13606 Parts 1, 2 and 4 may be requested and provided.

Currently, Parts 1 to 4 are published standards and Part 5 is at the final stages of the balloting phase. The reference model involves four packages: EXTRACT, DEMOGRAPHICS, SUPPORT and PRIMITIVES. The last two packages are related with the data types. The DEMOGRAPHICS package provides a minimal data set to define the various persons, software agents, devices and organizations that are referenced within the EHR extract. The EXTRACT package defines the root class of the reference model; EHR EXTRACT. The main hierarchy components (classes) of the EHR Extract Reference Model are given in Table 2.1 [11] together with some examples.

An EHR-Extract is the root class of part or all of the EHRs of a single patient. It contains zero or more Folders while Folders may have nested Folders. A Composition resides in a Folder via links or directly in the EHR-Extract. Sections are contained in Compositions while Sections may have nested Sections. An Entry is either contained within a Composition or located under a Section. An Entry contains Element instances which are optionally contained within a Cluster hierarchy.

In CEN EN 13606-1, a clinical statement can be represented by an Entry hierarchy. An Entry cannot contain further Entry classes but it is possible to reference other Entry classes via links. Complex data structures such as a table, a tree or a time series are represented by Clusters. The leaf node of the EHR-Extract hierarchy is the Element class which holds a single data value.

Part 3 of the 13606 standard also provides mappings for some of the HL7 Clinical Statement classes such as Act, Procedure and Supply. In our architecture, CDA R2 conformant EHR instances are automatically transformed to CEN EN 13606-1 conformant EHR instances, which has never been done before. Our system is one of the first implementations

Table 2.1: Main hierarchy components of the EHR Extract Reference Model

EHR HIERARCHY COMPONENT	DESCRIPTION	EXAMPLES
EHR_EXTRACT	The top-level container of part or all of the EHR of a single subject of care, for communication between an EHR Provider system and an EHR Recipient.	(Not applicable)
FOLDER	The high level organisation within an EHR, dividing it into compartments relating to care provided for a single condition, by a clinical team or institution, or over a fixed time period such as an episode of care.	Diabetes care, Schizophrenia, Cholecystectomy, Paediatrics, St Mungo's Hospital, GP Folder, Episodes 2000-2001, Italy.
COMPOSITION	The set of information committed to one EHR by one agent, as a result of a single clinical encounter or record documentation session.	Progress note, Laboratory test result form, Radiology report, Referral letter, Clinic visit, Clinic letter, Discharge summary, Functional health assessment, Diabetes review.
SECTION	EHR data within a COMPOSITION that belongs under one clinical heading, usually reflecting the flow of information gathering during a clinical encounter, or structured for the benefit of future human readership.	Reason for encounter, Past history, Family History, Allergy information, Subjective symptoms, Objective findings, Analysis, Plan, Treatment, Diet, Posture, Abdominal examination, Retinal examination.
ENTRY	The information recorded in an EHR as a result of one clinical action, one observation, one clinical interpretation, or an intention. This is also known as a clinical statement.	A symptom, an observation, one test result, a prescribed drug, an allergy reaction, a diagnosis, a differential diagnosis, a differential white cell count, blood pressure measurement.
CLUSTER	The means of organising nested multi-part data structures such as time series, and to represent the columns of a table.	Audiogram results, electro-encephalogram interpretation, weighted differential diagnoses.
ELEMENT	The leaf node of the EHR hierarchy, containing a single data value.	Systolic blood pressure, heart rate, drug name, symptom, body weight.

of the CEN EN 13606 as well.

For more information on EHR standards such as HL7 CDA, CEN EN 13606 or openEHR, a survey and analysis of EHR standards by Eichelberg et. al. [28] can be studied.

2.3 Medical Terminologies and Terminology Servers

Terminologies are a collection of terms or concepts with subsumption relationships between them, which are used to build a classification hierarchy of concepts. Terminologies provide a framework within which communities can communicate and express ideas in a consistent manner and facilitate unambiguous information sharing [4]. Medicine has a long tradition in structuring its domain knowledge through terminologies and coding schemes for diseases, medical procedures and anatomical terms such as SNOMED [75], LOINC [47], READ Codes [62], MeSH [50] and ICD-10 [39]. Many EHR standards such as HL7 CDA and CEN EHRcom encourage the usage of clinical terms based on universal or local terminology systems. Some well-known universal terminology systems that are also used in our work are:

- SNOMED (Systematized Nomenclature of Medicine) [75], is a systematically organized computer processable collection of medical terminology covering most areas of clinical information such as diseases, findings, procedures, microorganisms, and pharmaceuticals. It allows a consistent way to index, store, retrieve, and aggregate clinical data across specialties and sites of care. SNOMED Clinical Terms (CT) is a compositional concept system based on Description Logic [21], which means that concepts can be specialised by combinations with other concepts.
- Logical Observation Identifiers Names and Codes (LOINC) [47] appeared as a universal standard for identifying laboratory observations and clinical results. Since its inception, it has expanded to include not just medical and laboratory code names, but also nursing diagnosis, nursing interventions, outcomes classification, and patient care data set. It also includes document codes specifying the particular kind of document (e.g. History and Physical, Discharge Summary, Progress Note) that are very important for annotating the meaning of HL7 CDA sections.
- The International Statistical Classification of Diseases and Related Health Problems 10th Revision (ICD-10) [39] is a coding of diseases and signs, symptoms, abnormal findings, complaints, social circumstances and external causes of injury or diseases, as classified by the World Health Organization (WHO) [91]. It is being commonly used in Turkey for a long time.
- READ Codes [62] are a coded thesaurus of clinical terms which enable clinicians to make effective use of computer systems. The codes facilitate the access of information within patient records to enable reporting, auditing, research, automation of repetitive tasks, electronic communication and decision support.
- Medical Subject Headings (MeSH) [50] is a huge controlled vocabulary for the purpose of indexing journal articles and books in the life sciences. MeSH vocabulary is created and updated by the United States National Library of Medicine (NLM) [85].

Since the available medical terminology systems are often developed for purposes with different requirements, the way the knowledge is organized differs between these heterogeneous terminologies; they exhibit considerable variability both in terms of coverage and granularity [73]. This created the need for homogeneous multi-purpose Clinical Terminology Servers that try to allow the consistent and comparable entry of clinical data, e.g. patient observations, findings and events.

A Clinical Terminology Server usually provides the following services to client applications:

- management of external references to clinical concepts
- management of internal representations of clinical concepts
- mapping natural language to clinical concepts
- mapping clinical concepts to medical classification schemes
- management of extrinsic information about clinical concepts

The Unified Medical Language System (UMLS) [81] is one of such clinical terminology servers. It is a controlled compendium of many medical vocabularies, also providing a mapping structure between them. The UMLS is composed of three main knowledge components: Metathesaurus, Semantic Network and SPECIALIST Lexicon:

- The Metathesaurus forms the base of the UMLS and it comprises over one million biomedical concepts and five million concept names, all of which are from over a hundred controlled vocabularies and classification systems such as ICD-10, SNOMED and LOINC. The purpose of the Metathesaurus is to provide a basis of context and inter-context relationships between these various coding systems and vocabularies to provide a common basis of information exchange among the variety of clinical systems.
- The UMLS Semantic Network is designed to categorize concepts in the UMLS Metathesaurus and provide relationships among the concepts. It has 135 semantic types and 54 semantic relationships.
- The SPECIALIST Lexicon contains syntactic (how words are put together to create meaning), morphological (form and structure) and orthographic (spelling) information for biomedical terms.

GALEN [31] is another clinical terminology server which defines medical concepts in a descriptive logics model, called GRAIL, supporting subsumption.

In our architecture UMLS Metathesaurus is used for automatically universalizing the local coded values that exist within the National Health Information System of Turkey messages.

2.4 National Health Information System of Turkey

In Turkey, the Health Transformation Programme [78] was launched in 2003 and eHealth is one of the most critical actions presented in the action plan of the programme. This programme boosted the developments in eHealth at the national level. As a result, eHealth Project of Turkey has begun.

Turkey's National Health Information System (NHIS) Project [29] was initiated on January 30, 2003 with the participation of representatives from governmental institutions, non-governmental organizations, universities and the private sector under the coordination of Ministry of Health in order to establish cooperation among the sectors and national health information system's infrastructure.

NHIS to be established with the contribution of all the actors working in the health sector in the country is based on sharing a functional database which is accessible by authorized people and institutions with defined access rights, that covers all the citizens and in which each individual can reach his/her information that consists of all the data concerning the health from the birth and throughout his/her life on a spine of communication network with high bandwidth throughout the entire country and using the technologies reaching telemedicine applications in professional practice.

Based on the pre-defined goals of NHIS, several achievements have been done which are summarized in the following sections. More information is available in our publications [37] and [105].

Saglik-Net: The National Health Network

Saglik-Net is the conversion of the existing LAN-WAN into a true health network platform providing linkages, services and data repositories (e.g. minimum data sets of Electronic Healthcare Records) to all authorized parties in the health sector. This National Health Platform should be recognized, respected and trusted as the secure national platform for everything that is Health Information, either systems or services or both.

Saglik-Net is operational now and it is still being developed. The NHIS is built upon Saglik-Net and as of July 2007, the progress is as presented in Figure 2.6.

As shown in Figure 2.6, the National Health Data Dictionary (Ulusal Sağlık Veri Sözlüğü, USVS) [86], the Health Coding Reference Server (Sağlık Kodlama Referans Sunucusu, SKRS) [74], legacy systems (e.g. Personnel, Financing, Health Statistics) and some network management components are already connected to the Saglik-Net. Telemedicine applications

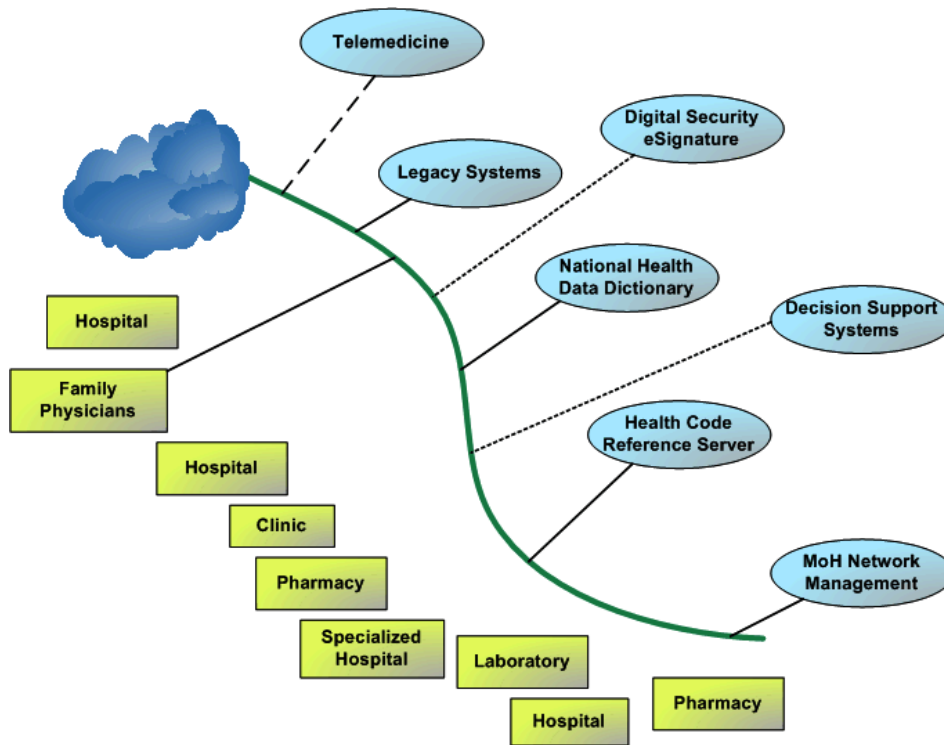


Figure 2.6: Saglik-Net as of July 2007

and the more comprehensive digital security mechanisms on top of the current Web Services Security module are about to be connected. Among the users of the network, currently only the Family Medicine Information System (FMIS) is fully connected, whereas the rest of users (primary, secondary, tertiary healthcare providers, payer institutions, etc.) will start connecting by September 30th, 2008. The software companies in Turkey have to comply with the standards developed by the Ministry of Health. In this way, interoperability among NHIS servers and various Hospital/Laboratory/Clinic/etc. information systems are provided. The Electronic Healthcare Records are based on HL7 Clinical Document Architecture (CDA) and use the National Health Data Dictionary, and the relevant coding systems.

It is expected that latest by year 2009, all the nodes in Figure 2.6 are fully connected to Saglik-Net. In the following sections, the major components of the Saglik-Net, some of which are abovementioned, are described.

The Health Coding Reference Server (HCRS)

In order to provide common coding/classification systems that are available to all healthcare players, MoH Department of Information Processing developed the Health Coding Reference Server (HCRS) [74] which encapsulates all the international and national coding systems used

in Turkey within a publicly accessible server.

Some of the coding systems available from HCRS are ICD-10 [39], Drugs, ATC (Anatomic, Therapeutic, and Chemical Classification System), Associations, Clinics, Specializations, Careers, Health Application Instructions, Supplies, Vaccines, Baby Monitoring Calendar, Pregnant Monitoring Calendar, Child Monitoring Schedule and Parameters.

The current version of HCRS is 2.0 and it is shared online through Web Services (through SOAP [76] requests and responses) that are available at the address: <http://212.175.169.157-/SKRSServis2/service.asmx>. A tabular form that allows querying through Web browsers is also available at the address: http://sbu.saglik.gov.tr/SKRS2_Listesi/.

The concept of Health Coding Reference Server (HCRS) is similar to the “vocabulary domain / value set” mechanism of HL7 v3. All the software companies doing business for Turkish health market are obliged to use the HCRS in their software and to design their products for fast adaptation to the latest updates in the HCRS, latest in 7 days from the date of update.

The National Health Data Dictionary (NHDD)

The National Health Data Dictionary (NHDD) [86] is developed to enable the parties to share the same meaning of data, and use them for the same purpose. The data whose definition and format determined within the NHDD establishes a reference for the information systems used at health institutions. Thus, the content interoperability among different applications is provided through the NHDD. The stable versions of NHDD will be annually revised to include improvements as comments are received from the field.

NHDD is composed of data sets and data elements conforming to ISO/IEC 11179-4 Standard [42]. Currently, there are 46 Minimum Health Data Sets and 261 data elements. Some example data elements are:

- Address
- Name
- Main Diagnosis
- Vaccination
- Treatment Method
- Diastolic Blood Pressure

- Healthcare Institution
- Marital Status

The data groups used for data collection are called Minimum Health Data Set (MHDS) and are formed from the NHDD as shown in Figure 2.7. In other words, MHDSs define the data sets which emerge at the time of presenting a certain service, for example, Infant Monitoring Data Set or Pregnant Monitoring Data Set. Currently Health MHDSs are completed and work is in progress for developing the other two sets, namely, Administrative and Financial.

MHDSs yet cover the most critical health data but in the future, these data sets will be extended to the full Electronic Healthcare Record schemas. Some example MHDS are:

- Citizen/Foreigner Registration MHDS
- Medical Examination MHDS
- Prescription MHDS
- Pregnant Monitoring MHDS
- Cancer MHDS
- Inpatient MHDS

MHDS has a changing and updateable structure. In other words, as more data is collected from the field, MHDS will be updated in certain periods to meet the new needs.

Currently, the MHDSs are being used for the National Decision Support System that enables the analysis of health data with methods such as data mining to determine the health policies of the MoH, Turkey.

The data elements within the Minimum Health Data Sets are mostly coded with coding systems and all these coding systems are available at the Health Coding Reference Server (HCRS). If a data element is defined in the National Health Data Dictionary as coded or classified, then the related coding/classification system is given both within the definition of the data element and in the “HCRS System Code” field. There are two possibilities for a coded element: either the value is gathered from a coding system such as ICD-10, healthcare institutions, specialties, etc. or the value is of parametric kind such as gender, or marital status.

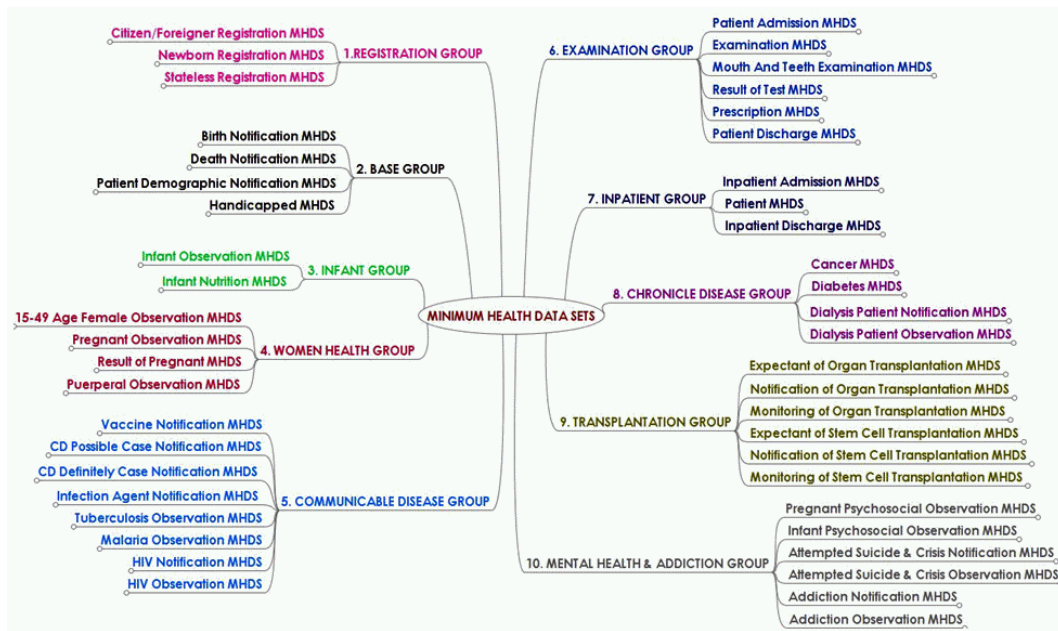


Figure 2.7: Minimum Health Data Sets in the NHDD

The Healthcare Professional Registry

Ministry of Health is authorized to provide the work licenses to the physicians in Turkey. The diploma/specialty information of the medical professionals is recorded together with their Turkish citizenship numbers in the Doctor Data Bank (DDB) [19]. As of October 2007, there are 162,446 registered doctors in the data bank. This includes all the physicians who obtained a license since 1923.

The Doctor Data Bank, that is, the Healthcare Professional Registry serves two purposes: The first one is that most of the payment providers control the health service and the prescriptions according to the physicians' specialty. For example, when a rule indicates that only the physicians with a certain specialty can prescribe certain medicines, it becomes possible to check whether the doctor who has signed the prescription has the required specialty. The related institutions signed an official protocol and started their implementations based on DDB in the last quarter of 2006. The second use of the DDB will be for authorizing access to the EHRs of the patients according to patient consent.

Data Collection and Sharing in the National Health Information System

The National Health Data Dictionary, the Minimum Health Data Sets and the Health Coding Reference Server provide the information space used in the messages to be exchanged between

the peripheral systems and National Health Information System. It should be noted that these data sets do not have a wire format and the most important decision to be taken at such nation-wide projects is whether to use a standard format. Most of the time, standards do not cover all of the identified information requirements in a project; therefore, the implementers choose to develop their own proprietary format instead of a standard format. However, this decreases interoperability. In the National Health Information System, HL7 v3 is selected because of the following reasons:

- HL7 is the most widely used electronic healthcare standard. Although, National Health Information System is to be used locally in Turkey, when it comes to communicate with other countries, the systems should be ready.
- After the completion of the NHIS, all of the medical information systems used in the nation-wide healthcare institutes should be adapted to communicate with the NHIS. Being based on a widely-used standard will facilitate the interoperability to a large extent.
- HL7 v3 provides mechanisms to extend the messages according to the requirements of a project.
- Specifically, the version 3 of the HL7 standard is selected rather than HL7 v2.x because of optionality problems of v2.x. Additionally, HL7 v3 is a standard whose conformance can be tested. In other words, it becomes possible to test the software clients running on the peripheral medical institutes that provide data to the NHIS servers. This is not possible in the 2.x versions of HL7.

In the following sections, the details of the implementation are presented.

Development of the Transmission Schemas

In the current version of the NHIS, the Transmission Schema instances are regarded as HL7 v3 messages and localized according to the Turkey's HL7 Profile [30].

Generally, the "Transmission Schemas" contain a main Minimum Health Data Set (from which the transmission schema is named after) and a set of auxiliary MHDSs that helps the interpretation of the main MHDS. An example transmission schema for "Examination" defined in the National Health Data Dictionary is shown in Figure 2.8 where the "Examination" data set is sent together with the "Newborn Registration" or "Citizen/Foreigner Registration" data sets. Furthermore, "Patient Admission" and "Patient Discharge" data sets are also

required. If there are any “Test Result” data sets or “Prescription” data sets, they are also sent along with the “Examination” data set.

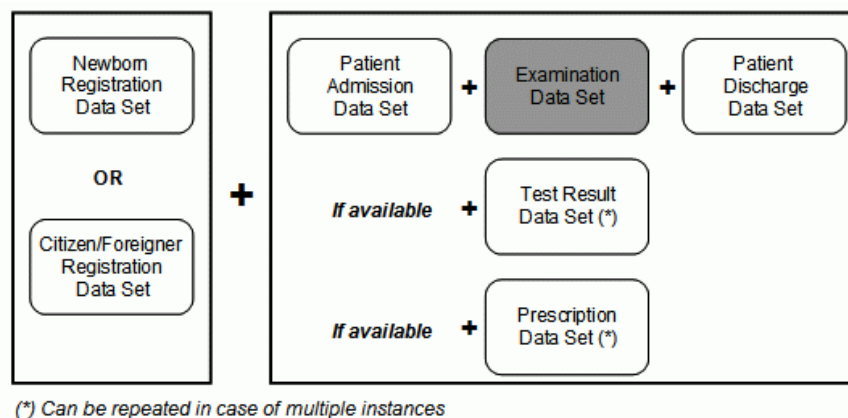


Figure 2.8: The Examination Transmission Schema from the NHDD

During the localization process, the rules which are set in the “HL7 Refinement, Constraint and Localization” [65] are applied. In other words, the CDA R-MIM [8] is edited and then converted to HMD [36] and XSD [99], respectively. However, the original HL7 CDA schemas are modified which breaks the CDA conformance of NHIS Transmission Schemas, since a conformant CDA document should at a minimum validate against the CDA Schema [7]. However, since the messages are derived from CDA R-MIM, the current versions of NHIS EHRs are based on HL7 v3 CDA R2.

It should be noted that there is no specific HL7 v3 Domain for all of the Transmission Schemas. For example, there is no HL7 Domain that is suitable for “Pregnant Psychosocial Observation” or “Communicable Disease Probable Case Notification”. Therefore, the CDA, which provides a generic mechanism to identify the contents of electronic healthcare documents, is selected as the wire format.

The transmission schemas are generated as follows: First, three new code systems are created in which there is a unique code for each of the artifacts:

1. NHIS Document Type Code System (DocumentType-CS) gives the codes for the “Transmission Schemas”.
2. NHIS Data Set Code System (DataSet-CS) contains the codes for data sets in MHDS (Minimum Health Data Sets).

3. NHIS Data Section Code System (DataSection-CS) specifies the codes for data elements in NHDD (National Health Data Dictionary).

```
<examination classCode="DOCCLIN" moodCode="EVN">
  <id root="2.16.840.1.113883.3.129.2.1.3" extension="11333439-08ab-42c5-ec2d-17064c153456" />
  <code code="MUAYENE" codeSystem="2.16.840.1.113883.3.129.2.2.1" codeSystemName="Döküman Tipi"
    codeSystemVersion="1.0" displayName="Muayene MSVS (Vatandaş/Yabancı)" />
```

Figure 2.9: The Beginning of the Examination Transmission Schema

Each “Transmission Schema” is wrapped with a root element named after the main data set in the transmission. For example, as shown in Figure 2.9, the root tag of the “Examination Transmission Schema” is `<examination>`. In this example, the document type (“Döküman Tipi” in Turkish) is “Examination” (“MUAYENE” in Turkish) and this code is obtained from the DocumentType-CS Code System whose object identifier (OID) is “2.16.840.1.113883.3.129.2.2.1”.

```
<examinationDataset classCode="DOCSECT" moodCode="EVN">
  <id root="2.16.840.1.113883.3.129.2.1.4" extension="4e7e0004-8e9e-44d2-9a9e-099d071e646a" />
  <code code="MUAYENE" codeSystem="2.16.840.1.113883.3.129.2.2.2" codeSystemName="Veriseti"
    codeSystemVersion="1.0" displayName="Muayene Veriseti" />
```

Figure 2.10: An Example First-level Section for the Examination TS

The “Data Sets” in the “Transmission Schemas” correspond to the first-level “Sections” in the CDA. The name of the opening tag of the “Data Set” is obtained by concatenating the name of the dataset with the “Dataset” keyword. The “code” of a “Data Set” is retrieved from the DataSet-CS Code System. For instance, in the example given in Figure 2.10, the opening tag is `<examinationDataset>` and the code is specified as “Examination” (“MUAYENE”). This code is obtained from the DataSet-CS (“Veriseti”) Code System whose object identifier (OID) is “2.16.840.1.113883.3.129.2.2.2”.

The data sections that wrap the NHDD data elements are represented by nesting new “section” elements in the data set’s “section” elements. The opening tag of a data section is obtained by concatenating the data element’s name with the “section” keyword. For example,

the diagnosis (“TANI” in Turkish) data element is introduced to the examination data set with the <diagnosisSection> XML element as shown in Figure 2.11.

```

<examinationDataset classCode="DOCSECT" moodCode="EVN">
  <id root="2.16.840.1.113883.3.129.2.1.4" extension="4e7e0004-8e9e-44d2-9a9e-099d071e646a" />
  <code code="MUAYENE" codeSystem="2.16.840.1.113883.3.129.2.2.2" codeSystemName="Veriseti"
  codeSystemVersion="1.0" displayName="Muayene Veriseti" />
  ...
  <diagnosisSection classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5" extension="6c4b1e87-c4f4-42b0-aaeb-05e23a77ed5a" />
    <code code="TANI" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kismi"
    codeSystemVersion="1.0" displayName="Tani Versinin Oldugu Bolu" />
    <text>Hastada vaskuler bagirsak bozuklugu teshis edilmistir.</text>
    <component typeCode="COMP" contextConductionInd="true">
      <diagnosis moodCode="EVN" classCode="OBS">
        ...
        <value code="K55" codeSystem="2.16.840.1.113883.6.3" codeSystemName="ICD-10"
codeSystemVersion="1.0" displayName="BARSÄGIN VASKÜLER BOZUKLUKLARI" />
      </diagnosis>
    </component>
  </diagnosisSection>
  ...

```

Figure 2.11: An Example Second-level Section for the Examination TS

As shown in Figure 2.11, the “code” of this “diagnosisSection” is “TANI” which is obtained from the DataSection-CS Code System whose OID is “2.16.840.1.113883.3.129.2.2.3”. The value of this data element is set in the “diagnosis” element under the “component” element. The “diagnosis” element is derived from the RIM Observation (OBS) Class, and its value is retrieved from ICD-10 code system. In other words, the values for the data elements are given with CDA Entry classes such as Observation or Procedure and they are associated to the related “section” element through the “component” element.

Figure 2.12 summarizes the relationships between the artifacts of NHDD, the “Transmission Schemas” and the HL7 CDA R2. Once this mapping is defined, the constraints implied through these mappings are reflected to the schemas by modifying the CDA Level One schema.

This mapping also briefly summarizes the changes that are applied to the original HL7 CDA R2 schema. In our work, we have developed an Adaptor that automatically generates HL7 CDA R2 conformant documents from the “Transmission Schema” instances.

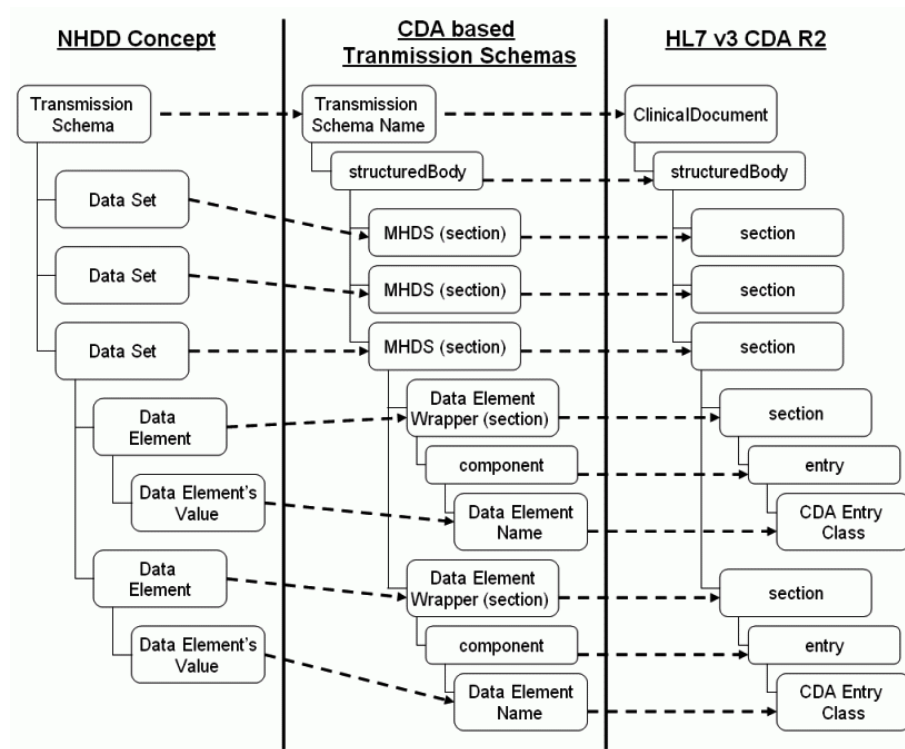


Figure 2.12: Mapping NHDD Concepts to HL7 v3 CDA R2

Development of the Communication Infrastructure

A “Transmission Schema” instance constitutes a message’s payload. In other words, “Transmission Schema” instances should be encapsulated in messages. For this purpose, “HL7 Transmission and Control Act Wrapper” is used. The transmission wrapper provides information on the id, creation time, sender, receiver of the message. The sender and receiver block contains the logical id of the sender and the receiver. When the transport software transmits the message, they are converted to the real physical addresses. In other words, this information is used by the application level software to convey the sender/receiver information to the transport software. Example Transmission Wrapper and Control Act Wrapper can be found in Figure 2.13.

As it is described in Section 2.1.1, HL7 Version 3 provides three transport specifications - ebXML, Web Services and MLLP - for the exchange of HL7 based content, messages and documents. Among them, Web Services Profile is the most promising, as it is based on widely-used Web Services Technology. Therefore, in the National Health Information System implementation, Web Services Profile is used for the communication infrastructure. The Basic Profile and the Security Profile of Web Service Profile have been implemented. For security, WS-Security Username Token Profile [93] over Secure Sockets Layer (SSL) is

```

<MCCI_IN000001TR01 xmlns="urn:hl7-org:v3">
  <id root="2.16.840.1.113883.3.129.2.1.2" extension="5b50d046-b567-4fba-a365-60c220c9fae1" />
  <creationTime value="20080402102643" />
  <responseModeCode code="Q" />
  <interactionId root="2.16.840.1.113883.3.129.2.1.1" extension="MCCI_IN000001TR01" />
  <processingCode code="P" />
  <processingModeCode code="T" />
  <acceptAckCode code="AL" />
  <receiver typeCode="RCV">
    <device classCode="DEV" determinerCode="INSTANCE">
      <id root="2.16.840.1.113883.3.129.1.1.5" extension="USBS" />
    </device>
  </receiver>
  <sender typeCode="SND">
    <device classCode="DEV" determinerCode="INSTANCE">
      <id root="2.16.840.1.113883.3.129.1.1.5" extension="SRDC-Deneme" />
    </device>
  </sender>
  <controlActEvent classCode="CACT" moodCode="EVN">
    <subject typeCode="SUBJ">
      <!-- CDA based Transmission Schema instance -->
      <examination classCode="DOCCLIN" moodCode="EVN">
        ...
      </examination>
    </subject>
  </controlActEvent>
</MCCI_IN000001TR01>

```

Figure 2.13: Example Transmission Wrapper and Control Act Wrapper from the NHIS

used.

Details of NHIS Web Services

In the current version of the NHIS implementation there are 25 HL7 Web Services. These are: “15-49 Age Female Observation”, “Mouth and Teeth Examination”, “Vaccine Notification”, “Infant Nutrition”, “Infant Observation”, “Infant Psychosocial Observation”, “Communicable Disease Definite Case Notification”, “Communicable Disease Probable Case Notification”, “Diabetes”, “Dialysis Notification”, “Dialysis Observation”, “Birth Notification”, “Pregnant Observation”, “Pregnancy Termination”, “Pregnant Psychosocial Observation”, “Patient Demographics Notification”, “Cancer”, “Puerperal Observation”, “Examination”, “Death Notification”, “Test Result”, “Citizen/Foreigner Registration”, “Stateless Person Registration”, “Newborn Registration” and “Inpatient”.

Almost for each HL7 Web Service there are four operations; namely “Insertion”, “Update”, “Deletion” and “Query” operations. All of these individual operations are synchronous but overall, the NHIS behaves asynchronously. The clients, which are the vendor applications deployed at healthcare organizations’ premises, send their “Transmission Schema” instances

through the insertion operation. This operation performs only syntax validation against the related schema and responds with an acknowledgement about the result of the validation. If the invocation of the first operation is successful, then the message is stored at the NHIS servers for detailed processing. This detailed processing involves the semantic validation of the content of the message, which is briefly described in the following “Validation of the Transmission Schema Instances” section. Then, at any time, the client invokes the query operation to query the semantic validation result by using the previously sent document’s Universally Unique Identifier (UUID) [87]. If the detailed content processing of the document is successful, a positive acknowledgement is received from the query operation; otherwise, the errors encountered in the semantic validation phase are reported to the user. The clients are also able to update and delete the previously inserted documents with the help of update and delete operations. The UML Interaction Diagram of NHIS HL7 Web Services in Figure 2.14 shows these interactions and the application roles used in Turkey.

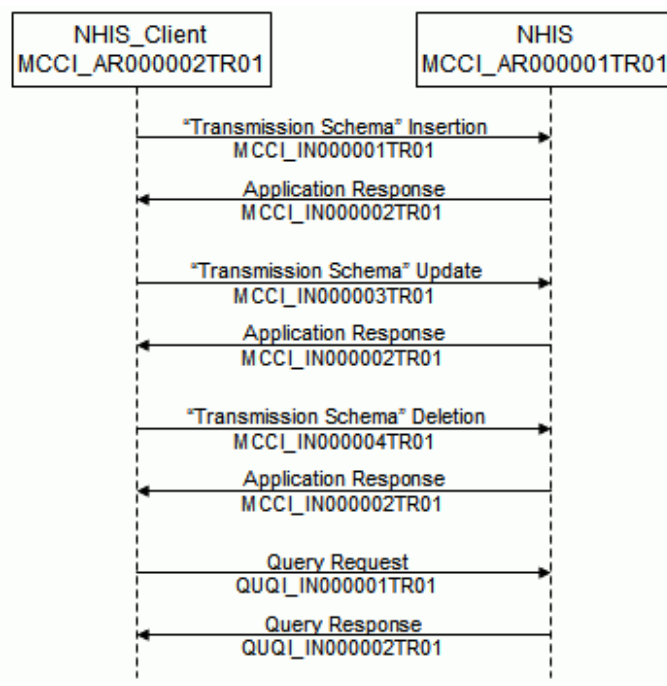


Figure 2.14: Interaction Diagram of NHIS HL7 Web Services

Apart from the HL7 Web Services, there are 16 more Web Services that neither the services nor their content conform to HL7 or any other eHealth standard. These are native synchronous Web Services usually developed for some risky communicable diseases. “Malaria

Notification” and “Tuberculosis Notification” are two such examples. They have Web-based forms on the MoH central servers as well and it is expected that healthcare professionals will send these comparably rare observations through these forms. These primitive Web Services are not within the scope of our work and throughout the rest of this document whenever NHIS Web Services are mentioned, NHIS HL7 Web Services are meant.

Validation of the Transmission Schema Instances

A two phase validation technique is applied for the validation of the incoming messages to the NHIS. In the first phase, which is called syntax validation phase, an incoming document instance is validated against the related XML Schema Definition (XSD) of the “Transmission Schema” by the greeting operations; namely the insertion and update operations. If successful, the message is conveyed to the second phase which is called the semantic validation phase.

The semantic validation phase checks the values in the data elements and the relationships between them. The semantic constraints are categorized into five classes:

1. *MERNIS Central Demographics Management System*: In Turkey, every citizen has a unique identifier and these identifiers are maintained in a system called MERNIS (Central Demographics Management System) [49]. The patient identifiers in the messages should be validated against this system.
2. *Doctor Data Bank (DDB)*: The identifiers of the healthcare professionals that appear in the messages should be validated against this DDB.
3. *Value formats*: Values of some data elements should obey some specific formats. As an example, HL7 date can be of the form YYYYMMDD.
4. *Coded elements*: The coded elements should have values from the Health Coding Reference Server (HCRS).
5. *Business rules*: There are some rules among the message elements such as the examination end date should be later than the examination begin date. There are more complex clinical business rules as well. These rules are defined and documented [40] in collaboration with the healthcare professionals and the administrative staff of the healthcare organizations.

Digital Security

The Technical and Scientific Research Council (TUBITAK) [80] is the authorized body for providing solutions in Digital Security in Turkey. For this purpose, TUBITAK is developing the National Electronic Identity Verification System and the Smart Identity Card. These solutions will be available for both individuals and institutions.

In Turkey, digital signatures were legislated in 2004, mobile signatures were legislated in 2006 and the integration of Family Medicine Information Systems (FMIS) [1] with digital signatures has already been realized.

Planned Developments

NHIS, Turkey is operational now and it is at the end of its public testing phase. By September 30th, 2008 all the public healthcare organizations are obliged to send collect and send their Minimum Health Data Sets. The deadline for the private and university healthcare organizations is January 2009. It is expected that it will be possible to collect data from 90% of the field (primary, secondary, tertiary healthcare providers, family physicians, etc.) by year 2009.

The data flow in NHIS is not always one-way, that is from the healthcare institutions to NHIS servers. The authorized parties such as General Practitioners can also query and retrieve the healthcare records from the NHIS servers. Hence, the sharing of medical records among healthcare providers will be possible in the future when the necessary legislations are passed. Currently the work is going on determining legal ground about the access rights of all types of users. The importance of this thesis work will increase enormously when sharing of the Electronic Healthcare Records is broadened, especially in the cross-border case.

Finally, an e-appointment system is being developed on Saglik-Net which will allow the General Practitioners using the Family Medicine Information System to arrange appointments for their patients in the hospitals.

In our architecture the National Health Information System of Turkey is the subject of our cross-border eHealth interoperability methodology.

2.5 The Extensible Stylesheet Language (XSL) Family

The Extensible Stylesheet Language (XSL) Family [101] is a family of recommendations for defining Extensible Markup Language (XML) [95] document transformation and presentation. It is developed by the W3C [90] XSL Working Group. In comparison, it is like CSS

[16] in HTML but it is indeed much more capable in transformation. It consists of three parts:

- XSL Transformations (XSLT) [103] is a language for transforming XML documents.
- The XML Path Language (XPath) [98] is an expression language used by XSLT to access or refer to parts of an XML document.
- XSL Formatting Objects (XSL-FO) [102] is an XML vocabulary for specifying formatting semantics.

XSL is a language for expressing stylesheets. Given a class of arbitrarily structured XML documents, an XSL stylesheet is used to express intentions about how that structured content should be presented; that is, how the source content should be styled, laid out, and paginated onto some presentation medium, such as a window in a Web browser or a hand-held device, or a set of physical pages in a catalog, report, pamphlet, or book.

Originally intended to perform complex styling operations, like the generation of tables of contents and indexes, XSL is now used as a general purpose XML processing language.

XSL Transformations

Extensible Stylesheet Language Transformations (XSLT) [103] is an XML-based language used for the transformation of XML documents into other XML or “human-readable” documents such as HTML or plain text. A transformation expressed in XSLT describes rules for transforming a source tree into a result tree. The transformation is achieved by associating patterns with templates. A pattern is matched against elements in the source tree. A template is instantiated to create part of the result tree. The result tree is separate from the source tree. The structure of the result tree can be completely different from the structure of the source tree. In constructing the result tree, elements from the source tree can be filtered and reordered, and arbitrary structure can be added.

A transformation expressed in XSLT is called a stylesheet. A stylesheet contains a set of template rules. A template rule has two parts: a pattern which is matched against nodes in the source tree and a template which can be instantiated to form part of the result tree.

The basic elements of XSL Transformations are presented in Figure 2.15.

The most recent version of XSLT is XSLT 2.0 [104] which represents a significant increase in the capability of the language. In our architecture XSLT 2.0 is used for two purposes;

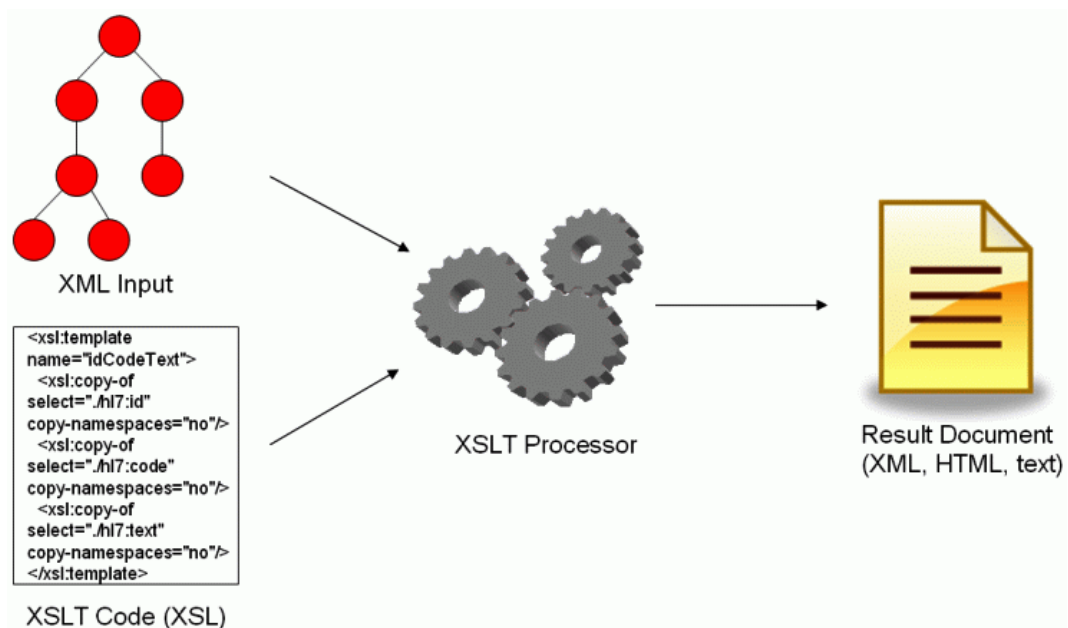


Figure 2.15: Basic Elements of XSL Transformation

generating HL7 CDA R2 conformant EHR documents from the “Transmission Schema” instances of National Health Information System of Turkey and generating CEN EHRcom conformant EHR documents from the HL7 CDA R2 instances.

The XML Path Language

The XML Path Language (XPath) [98] is a language used by XSLT, and also available for use in non-XSLT contexts, for addressing the parts of an XML document. In addition to this primary addressing purpose, it also provides basic facilities for manipulation of strings, numbers and booleans. XPath uses a compact, non-XML syntax to facilitate use of XPath within URIs and XML attribute values. XPath gets its name from its use of a path notation as in URLs for navigating through the hierarchical structure of an XML document and selecting nodes by a variety of criteria.

XPath models an XML document as a tree of nodes. There are different types of nodes; element nodes, attribute nodes and text nodes. XPath defines a way to compute a string-value for each type of node. XPath fully supports XML Namespaces.

The primary syntactic construct in XPath is the expression which is evaluated to yield an object. The most important kind of expression is a location path. A location path selects a set of nodes relative to the context node. The result of evaluating an expression that is a location path is the node-set containing the nodes selected by the location path. Location

paths can recursively contain expressions that are used to filter sets of nodes. A sample XPath expression applied to an XML instance is presented in Figure 2.16.

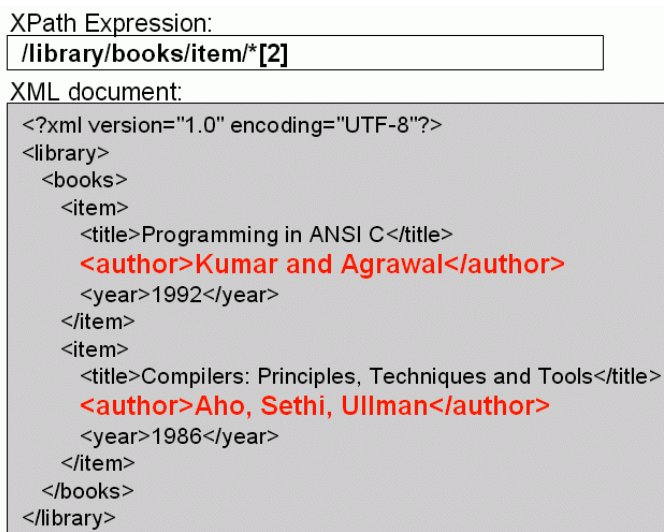


Figure 2.16: Sample XPath expression applied to an XML document

In our architecture XPath is used for locating and gathering all the nodes that contain coded values in a CDA instance before universalizing those values by the help of an extendible system based on UMLS.

XSL Formatting Objects

XSL Formatting Objects (XSL-FO) [102] is an XML vocabulary for specifying formatting semantics. Formatting is the process of turning the result of an XSL transformation into a tangible form for the reader or listener. The vocabulary of formatting objects supported by XSL represents the set of typographic abstractions available to the designer. Semantically, each formatting object represents a specification for a part of the pagination, layout, and styling information that will be applied to the content of that formatting object as a result of formatting the whole result tree.

The basic processing steps of XSL-FO is as follows: The XML document to be formatted is fed into an XSLT processor together with XSLT codes, just as been described in Figure 2.15, but this time XSLT codes use the XSL-FO vocabulary. This transformation process converts the XML into XSL-FO. Then, this XSL-FO document is passed to the XSL formatter which converts the XSL-FO document into some format which is usually more

“human-readable”. The most common output formats of XSL-FO are Portable Document Format (PDF), PostScript (PS) and Rich Text Format (RTF).

Currently, XSL-FO is not used in our architecture but as a future work, PDF reports of conformant EHR instances can be generated automatically with the help of XSL-FO.

CHAPTER 3

MAPPING THE NATIONAL ELECTRONIC HEALTHCARE RECORDS TO INTERNATIONAL STANDARDS - CASE STUDY: TURKEY

As described in detail in Section 2.4, The National Health Information System of Turkey preferred HL7 Version 3 as the messaging standard and CDA as the base for EHR standard for the following reasons:

- HL7 is the most widely used electronic healthcare standard. Although, NHIS is to be used locally in Turkey, when it comes to communicate with other countries, the systems should be ready.
- After the completion of the NHIS, all of the medical information systems used in the nation-wide healthcare institutes should be adapted to communicate with the NHIS. Being based on a widely-used standard will facilitate the interoperability to a large extent.
- HL7 v3 and CDA provide mechanisms to extend the messages according to the requirements of a project.
- Specifically, the version 3 of the HL7 standard is selected rather than HL7 v2.x because of optionality problems of v2.x. Additionally, HL7 v3 is a standard whose conformance can be tested. In other words, it becomes possible to test the software clients running on the peripheral medical institutes that provide data to the NHIS servers. This is not possible in the v2.x versions of HL7.

- It was not possible to find specific domains and thus ready specific message structures for the Minimum Health Data Sets such as “Puerperal Observation” or “Infant Psychosocial Observation” that are defined within the National Health Data Dictionary. However, CDA offers a generic and extensible model for representing any kind of data.
- CDA is the mostly widely preferred and implemented EHR standard.

Turkey adopted CDA as the EHR standard but unfortunately, in order to meet all national requirements directly at the schema level, many changes have been done on the original CDA Schema that broke the conformity of local “Transmission Schemas” to CDA. The generated schemas are completely HL7 v3 and CDA R-MIM conformant but still it is not possible to realize cross-border data exchange with the current status.

In this chapter, we first analyze and list all of the incompatible changes that have been done on the original CDA Schema and then develop an adaptor based on XSL Transformations (XSLT) to automatically generate CDA conformant EHR documents from the “Transmission Schema” instances.

3.1 The Incompatible Changes Done on the Original CDA Schema for Developing “Transmission Schemas” in NHIS, Turkey

As described in the Enabling Technologies and Standards section, currently NHIS, Turkey has 25 HL7 Version 3 Web Services for collecting the defined Transmission Schemas (thus the Minimum Health Data Sets) from the hospital/laboratory/family medicine/clinic information systems. The names of the services are identical with the related Transmission Schema: “15-49 Age Female Observation”, “Mouth and Teeth Examination”, “Vaccine Notification”, “Infant Nutrition”, “Infant Observation”, “Infant Psychosocial Observation”, “Communicable Disease Definite Case Notification”, “Communicable Disease Probable Case Notification”, “Diabetes”, “Dialysis Notification”, “Dialysis Observation”, “Birth Notification”, “Pregnant Observation”, “Pregnancy Termination”, “Pregnant Psychosocial Observation”, “Patient Demographics Notification”, “Cancer”, “Puerperal Observation”, “Examination”, “Death Notification”, “Test Result”, “Citizen/Foreigner Registration”, “Stateless Person Registration”, “Newborn Registration” and “Inpatient”.

While developing the schemas for these services, the consortium that is responsible from the implementation of the NHIS preferred to start from the Refined Message Information

Model (R-MIM) of CDA instead of directly getting and restricting CDA Schema [9]. The rationale of the consortium for this selection is applying all the local requirements of National Health Data Dictionary and Business Rules directly at the XSD level as much as possible since they believe that this way the integration of the vendors, most of which are not familiar with healthcare standards, will be much easier. Our experience and relationship with the vendors show that this is not true indeed.

During the localization process, the rules which are set in the “HL7 Refinement, Constraint and Localization” [65] are applied. In other words, the CDA R-MIM is edited and then converted to HMD and XSD, respectively with the help of HL7 Version 3 tools: R-MIM Designer, RoseTree and V3 Generator [35]. The schemas are available in HL7 Integration Guide for Turkey [43]. Although the constraint rules defined in HL7 Refinement, Constraint and Localization are valid rules that do not break the conformance of developed messages to HL7 v3 standard, CDA has an autonomous position in this respect. A conformant CDA document should at a minimum validate against the CDA Schema [7]. The modifications that have been done during the localization process have broken this conformance.

In this section, the list of incompatible changes done on the original CDA Schema will be presented. While doing so, since most of the cases are common for all 25 messages and it is not possible present each of the messages individually, the examples will usually be given through a valid “Examination” message (i.e. Transmission Schema). The complete message can be found in Appendix A and also in our web page [55].

The differences are presented in breadth-first traversal of the CDA:

1. As it is clearly visible in Figure 3.1, the root element of a CDA document must be “ClinicalDocument”. However, in the Transmission Schemas this value is replaced with the name of the Transmission Schema, e.g. “examination”, “cancer”, “diabetesNotification”, etc. as it is seen in Figure 3.2:
2. The “ClinicalDocument” should definitely have a “typeId” element with fixed values as presented in Figure 3.3. In the Transmission Schemas this mandatory element is removed.
3. There are a lot of changes related with the “recordTarget” element of CDA which is used for presenting the details of the subject of the document, that is the patient. There are three different types of patient registrations in the NHIS; Citizen/Foreigner Registration, Newborn Registration and Stateless Registration. They are all used for

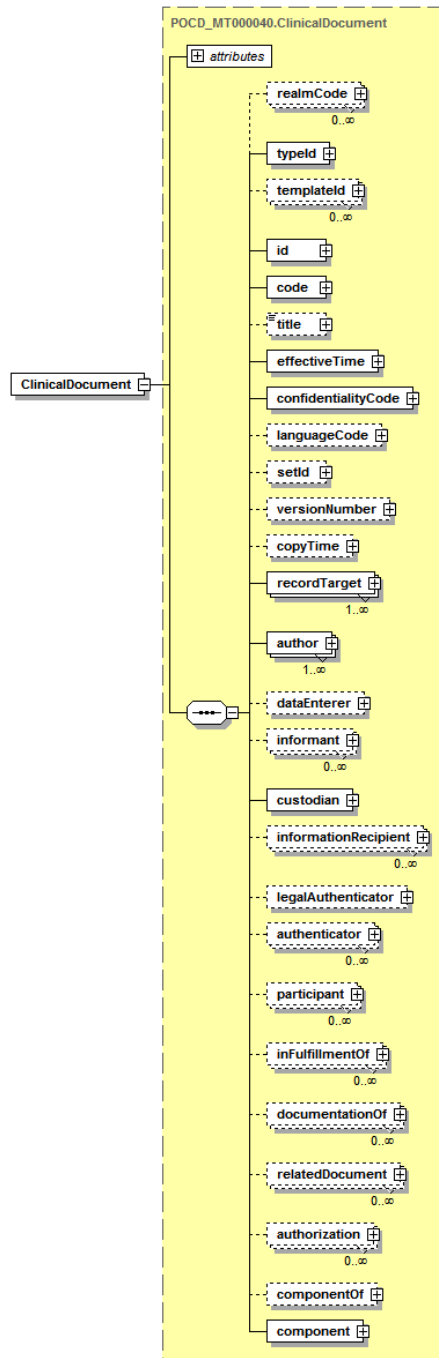


Figure 3.1: The First Level Elements of the CDA: CDA Header

```

<examination xmlns="urn:h17-org:v3" classCode="DOCCLIN" moodCode="EVN">
  <id root="2.16.840.1.113883.3.129.2.1.3" extension="11333439-08ab-42c5-ec2d-17064c153456" />
  <code code="MUAYENE" codeSystem="2.16.840.1.113883.3.129.2.2.1" codeSystemName="Döküman Tipi"
  codeSystemVersion="1.0" displayName="Muayene MSVS (Vatandaş/Yabancı)" />

```

Figure 3.2: The Beginning of the Examination Transmission Schema

```

<ClinicalDocument xmlns="urn:hl7-org:v3" classCode="DOCCLIN" moodCode="EVN">
  <typeId root="2.16.840.1.113883.1.3" extension="POCD_HD000040"/>
  <id root="2.16.840.1.113883.3.129.2.1.3" extension="11333439-08ab-42c5-ec2d-17064c153456" />

```

Figure 3.3: A Conformant “ClinicalDocument” Introduction

presenting the demographics of the patient. But, while they are mapped to Transmission Schemas, only Citizen/Foreigner Registration MHDS content has been fit into the original recordTarget schema, the other two MHDSs has broken down the schema by renaming elements and adding some new elements. An example incompatible record-Target for Newborn Registration is presented in Figure 3.4.

```

<recordTarget typeCode="RCT" contextControlCode="OP">
  <babyPatientRole classCode="PAT">
    <patientBaby classCode="PSN" determinerCode="INSTANCE">
      <administrativeGenderCode code="1" codeSystem="2.16.840.1.113883.3.129.1.2.21"
codeSystemName="Cinsiyet" codeSystemVersion="1.0" displayName="Erkek"/>
      <birthTime value="20080209"/>
      <multipleBirthOrderNumber value="1"/>
      <guardian classCode="GUARD">
        <code code="MTH" codeSystem="2.16.840.1.113883.5.111" codeSystemName="RoleCode"
codeSystemVersion="1.0" displayName="Mother"/>
        <guardianMother determinerCode="INSTANCE" classCode="PSN">
          <id root="2.16.840.1.113883.3.129.1.1.1" extension="12345678905"/>
        </guardianMother>
      </guardian>
    </patientBaby>
  </babyPatientRole>
</recordTarget>

```

Figure 3.4: An Example Invalid recordTarget for Newborn Registration

First of all, the child of the recordTarget should be “patientRole”, not “babyPatientRole”. Then this “patientRole” should have an “id” element but in our case “babyPatientRole” does not have it. “patientBaby” should be replaced with “patient” and this element cannot have an element entitled “multipleBirthOrderNumber”, it should be removed. As another problematic change, the name of the child of “guardian” element should be “guardianPerson”, not “guardianMother”. Finally for now, “id” element of “guardianMother” should belong to “guardian” role in fact.

The conformant recordTarget for this Newborn Registration should be as presented in Figure 3.5.

```
<recordTarget>
  <patientRole>
    <id root="2.16.840.1.113883.3.129.1.1.1" extension="12345678905.20080209.1"/>
    <patient>
      <administrativeGenderCode code="1" codeSystem="2.16.840.1.113883.3.129.1.2.21"
codeSystemName="Cinsiyet" codeSystemVersion="1.0" displayName="Erkek"/>
      <birthTime value="20080209"/>
      <guardian classCode="GUARD">
        <id root="2.16.840.1.113883.3.129.1.1.1" extension="16892264390"/>
        <code code="MTH" codeSystem="2.16.840.1.113883.5.111" codeSystemName="RoleCode"
codeSystemVersion="1.0" displayName="Mother"/>
        <guardianPerson/>
      </guardian>
    </patient>
  </patientRole>
</recordTarget>
```

Figure 3.5: The Valid recordTarget for Newborn Registration

4. The CDA conformance rules clearly state that, within a CDA document there can only be one recordTarget unless the document is written for a group of people or public health monitoring purposes. In other words, if the document belongs to one person, there should be exactly one recordTarget. However, not in all messages of the NHIS but in the “Patient Demographics”, although it is created for just one patient there are two recordTarget blocks, one for the usual patientRole and the second for reporting patient declared attributes such as address and telecom details. The name of the second element is “declarationPatientRole” and this usage is against the CDA. In our architecture, the elements of this “declarationPatientRole” are moved under the first recordTarget and second recordTarget is deleted. An example invalid declarationPatientRole is displayed in Figure 3.6.
5. Another CDA header attribute is the “custodian” element which represents the healthcare organization that is in charge of maintaining the document. In NHIS, this part is used for providing the id of the healthcare organization that created the Transmission Schema, which is consistent with the original purpose. However, two incompatible


```

<recordTarget typeCode="RCT" contextControlCode="OP">
  <declarationPatientRole classCode="PAT">
    <addr use="TMP">
      <censusTract>15753</censusTract>
      <additionalLocator>3. cadde, 49. sokak, No:7/15</additionalLocator>
    </addr>
    <telecom value="tel:3122877690" use="MC"/>
    <patientDeclarationPatient determinerCode="INSTANCE" classCode="PSN">
      ...
    </patientDeclarationPatient>
  </declarationPatientRole>
</recordTarget>

```

Figure 3.6: An Invalid declarationPatientRole Example

changes have been done. First, “representedHealthcareOrganization” seen in Figure 3.7 should be replaced with “representedCustodianOrganization”. Secondly, its “classCode” attribute cannot be “ENT” representing the “Entity” class, its default value is “ORG” that stands for the “Organization” class.

```

<custodian typeCode="CST">
  <assignedCustodian classCode="ASSIGNED">
    <representedHealthcareOrganization classCode="ENT" determinerCode="INSTANCE">
      <id root="2.16.840.1.113883.3.129.1.1.6" extension="4553"/>
    </representedHealthcareOrganization>
  </assignedCustodian>
</custodian>

```

Figure 3.7: An Example custodian Block from the NHIS

6. The CDA header attribute “informationRecipient” represents a recipient who should receive a copy of the document. In our case, this recipient is always the Ministry of Health. While modeling this situation, again some incompatible renaming have been on the original CDA Schema. An example from the Examination Transmission Schema is presented in Figure 3.8. Here, “primaryInformationRecipient” should be replaced with “informationRecipient”, “recipient” with “intendedRecipient” and “representedMinistryOfHealth” with “receivedOrganization”.
7. In “Patient Demographics”, “Communicable Disease Definite Case Notification” and “Communicable Disease Probable Case Notification” messages a new target entitled

```

<primaryInformationRecipient typeCode="PRCP">
  <recipient classCode="ASSIGNED">
    <representedMinistryOfHealth classCode="ORG" determinerCode="INSTANCE">
      <id root="2.16.840.1.113883.3.129.1.1.6" extension="5881"/>
    </representedMinistryOfHealth>
  </recipient>
</primaryInformationRecipient>

```

Figure 3.8: An Example primaryInformationRecipient Block from the NHIS

“indirectTarget” has been defined for providing the address of next of kin, if available. However, there is no CDA header attribute named “indirectTarget”. This new element can only be represented as the “participant” attribute of CDA header. An example “indirectTarget” element from the NHIS is available in Figure 3.9. Here, “indirectTarget” should be renamed as “participant” and “nextOfKin” as “associatedEntity”. Moreover, the position of “indirectTarget” (or “participant”) element is also wrong. It cannot appear just after the recordTarget as it can be seen in Figure 3.1.

```

<indirectTarget typeCode="IND" contextControlCode="OP">
  <nextOfKin classCode="NOK">
    <addr use="H">
      <censusTract>15753</censusTract>
      <additionalLocator>3. cadde, 49. sokak, No:7/15</additionalLocator>
    </addr>
  </nextOfKin>
</indirectTarget>

```

Figure 3.9: An Example indirectTarget Block from the NHIS

8. With the help of “parentDocument” attribute of its header, CDA supports revision/addenda/ transformation of documents. As it is presented in the “Details of NHIS Web Services” section of the “Enabling Technologies and Standards” chapter, NHIS also supports update and deletion of previously submitted documents. In these operations the abovementioned functionality of CDA is being used but there is one slight inconsistency. An example block is presented in Figure 3.10. Here, the name of the element should be “relatedDocument” instead of “replacementOf” that is made up by the consortium.

```

<replacementOf typeCode="RPLC">
  <parentDocument classCode="DOCCLIN" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.3" extension="11201438-08ab-42c5-9c2d-17064c133456"/>
    <code code="MUAYENE-YENIDOGAN" codeSystem="2.16.840.1.113883.3.129.2.2.1"
      codeSystemName="Döküman Tipi" codeSystemVersion="1.0" displayName="Muayene MSVS (Yenidoğan)"/>
    <versionNumber value="1"/>
  </parentDocument>
</replacementOf>

```

Figure 3.10: An Example replacementOf Block from the NHIS

9. Up to this item, the incompatible changes that are done on the CDA header have been presented. From this item on, the ones that are done on the body will be presented.

As it is explained in the Enabling Technologies chapter and also as visible in Figure 3.11, the body of the CDA starts under the “component” element of root element; “ClinicalDocument”. In Turkey, the Transmission Schemas use “structuredBody”. All the Minimum Health Data Sets except the Patient Registration (i.e. Citizen/Foreigner Registration, Newborn Registration and Stateless Registration) contained within a Transmission Schema are represented as first level “section”s and these “sections” are bound to the “structuredBody” through “component” ActRelationship. However, inappropriately with the CDA schema, these components are numbered sequentially in the NHIS. Figure 3.12 shows the components of the Examination Transmission Schema.

The component names are renamed with concatenation of sequential numbers, like “component1”, “component2” ... “componentN”. This is against the CDA schema; they should all be “component”.

10. The CDA body is composed of nested “section”s and these sections are bound to the parent element through “component” ActRelationships. In the NHIS, the first level sections represent the MHDSs of a Transmission Schema. However, the name of these “section”s are changed to the name of the corresponding MHDS concatenated with “Dataset” keyword, e.g. “dischargeDataset”, as it is seen in Figure 3.12. These tags should be replaced with “section” tag as in Figure 3.13. This renaming does not cause any information loss in fact, because as it is seen in Figure 3.13, the semantics of the sections are described with coded terms through their “code” elements. The code system “Veriseti” (means “data set”) is defined by the NHIS for coding the data set sections. In this specific example, code “CIKIS” (means “discharge”) is chosen from this code system to describe the section. So, by renaming the section, we have only

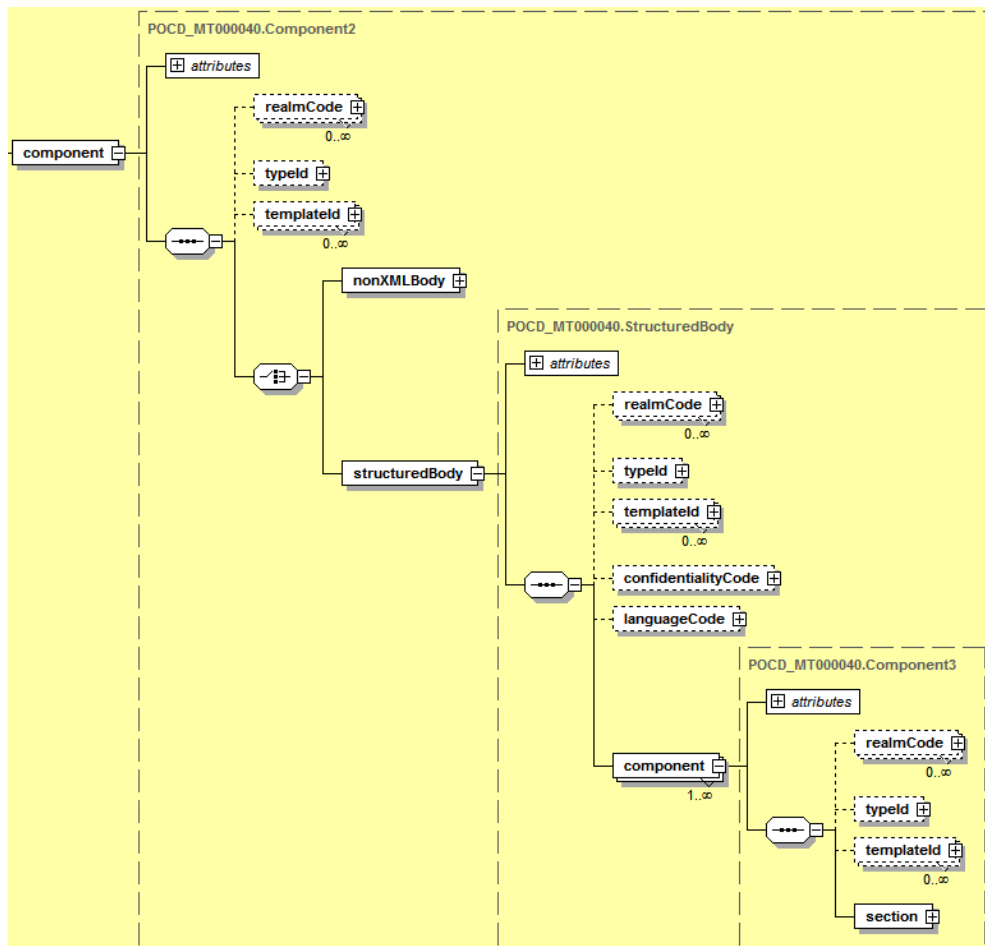


Figure 3.11: High Level View of the CDA Body

eliminated duplicate information but did not lose anything.

11. Similar changes that have been explained in the previous two items have been done on lower level elements, i.e. the Data Elements of the NHDD, as well. The mapping among the NHDD concepts, CDA based Transmission Schemas developed in the NHIS and the original CDA R2 is shown in Figure 3.14. In the Transmission Schemas, the data set sections are composed of sections for representing the data elements of the NHDD. These sections are bound to the parent section with components. However, again these components are named like “component1”, “component2” ... “componentN” and the name of the sections are replaced with the name of the data element concatenated with the “Section” keyword as seen in Figure 3.15.

The corresponding conformant CDA block should look like as in Figure 3.16.

12. In the NHDD, some MHDSs contain healthcare professional information that may be

```

<examination classCode="DOCCLIN" moodCode="EVN">
  [CDA Header]
  <component typeCode="COMP" contextConductionInd="true">
    <structuredBody classCode="DOCBODY" moodCode="EVN">
      <component1 typeCode="COMP" contextConductionInd="true">
        <testResultDataset classCode="DOCSECT" moodCode="EVN">
          ...
        </testResultDataset>
      </component1>
      <component2 typeCode="COMP" contextConductionInd="true">
        <dischargeDataset classCode="DOCSECT" moodCode="EVN">
          ...
        </dischargeDataset>
      </component2>
      <component3 typeCode="COMP" contextConductionInd="true">
        <examinationDataset classCode="DOCSECT" moodCode="EVN">
          ...
        </examinationDataset>
      </component3>
      <component4 typeCode="COMP" contextConductionInd="true">
        <receptionDataset classCode="DOCSECT" moodCode="EVN">
          ...
        </receptionDataset>
      </component4>
      <component5 typeCode="COMP" contextConductionInd="true">
        <prescriptionDataset classCode="DOCSECT" moodCode="EVN">
          ...
        </prescriptionDataset>
      </component5>
    </structuredBody>
  </component>
</examination>

```

Figure 3.12: Examination Transmission Schema components

```

<section classCode="DOCSECT" moodCode="EVN">
  <id root="2.16.840.1.113883.3.129.2.1.4" extension="28661575-af59-45be-bdf5-3590385801df" />
  <code code="CIKIS" codeSystem="2.16.840.1.113883.3.129.2.2" codeSystemName="Veriseti"
    codeSystemVersion="1.0" displayName="Çıkış Veriseti" />
  ...

```

Figure 3.13: An Example dischargeDataset section

different from the author of the CDA document. These professionals are modeled with the “author” participation of the CDA sections. However, during the localization process some inconsistencies have been generated on the Transmission Schemas.

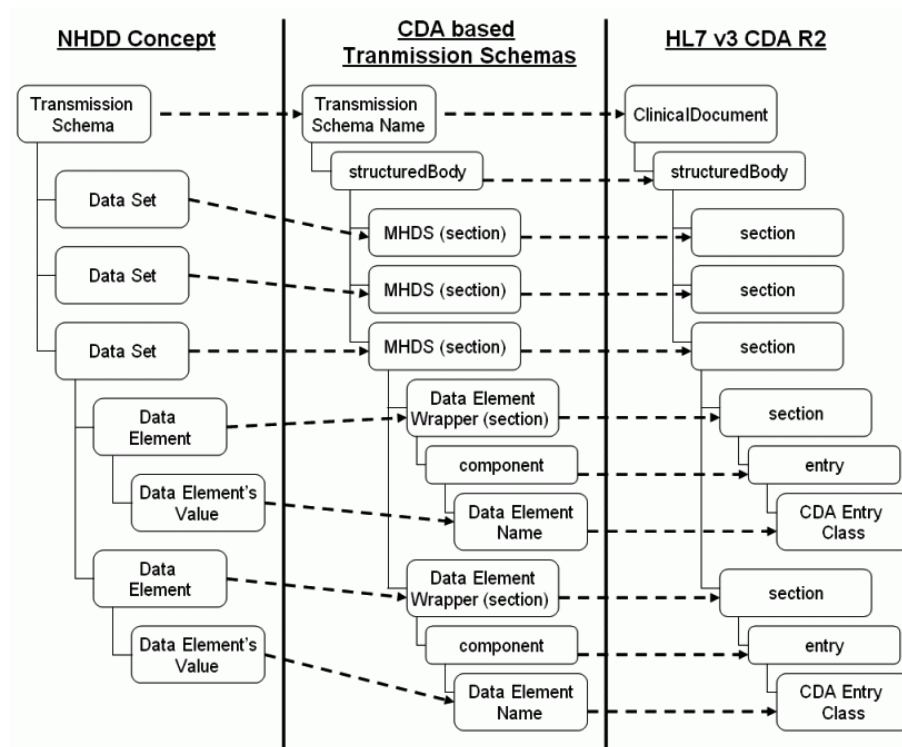


Figure 3.14: Mapping NHDD Concepts to HL7 v3 CDA R2

```

<examinationDataset classCode="DOCSECT" moodCode="EVN">
  ...
  <component1 typeCode="COMP" contextConductionInd="true">
    <examProtocolNoSection classCode="DOCSECT" moodCode="EVN">
      <id root="2.16.840.1.113883.3.129.2.1.5" extension="de380a98-d31f-4a33-a878-7366da629737" />
      <code code="PROTOKOLNO" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
        codeSystemVersion="1.0" displayName="Protokol No Bilgisinin Olduğu Bölüm" />
      ...
    </examProtocolNoSection>
  </component1>
  <component2 typeCode="COMP" contextConductionInd="true">
    <reportSection classCode="DOCSECT" moodCode="EVN">
      <id root="2.16.840.1.113883.3.129.2.1.5" extension="cba90686-fa28-4a0e-9ee6-342d7bcb3302" />
      <code code="RAPOR" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
        codeSystemVersion="1.0" displayName="Rapor Verisinin Olduğu Bölüm" />
      ...
    </reportSection>
  </component2>
  ...
</examinationDataset>

```

Figure 3.15: An Excerpt from the examinationDataset section

```

<section classCode="DOCSECT" moodCode="EVN">
  ...
  <component typeCode="COMP" contextConductionInd="true">
    <section classCode="DOCSECT" moodCode="EVN">
      <id root="2.16.840.1.113883.3.129.2.1.5" extension="de380a98-d31f-4a33-a878-7366da629737" />
      <code code="PROTOKOLNO" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
        codeSystemVersion="1.0" displayName="Protokol No Bilgisinin Olduğu Bölüm" />
      ...
    </section>
  </component>
  <component typeCode="COMP" contextConductionInd="true">
    <section classCode="DOCSECT" moodCode="EVN">
      <id root="2.16.840.1.113883.3.129.2.1.5" extension="cba90686-fa28-4a0e-9ee6-342d7bcb3302" />
      <code code="RAPOR" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
        codeSystemVersion="1.0" displayName="Rapor Verisinin Olduğu Bölüm" />
      ...
    </section>
  </component>
  ...

```

Figure 3.16: The Corresponding Conformant CDA Block for examinationDataset

Figure 3.17 shows an example “author” element from the Test Result Data Set.

```

<author typeCode="AUT" contextControlCode="OP">
  <testDoctor classCode="ASSIGNED">
    <id root="2.16.840.1.113883.3.129.1.1.1" extension="12345678905"/>
  </testDoctor>
</author>

```

Figure 3.17: An Example author Block from the testResultDataset

First of all, the “author” must have a “time” element that represents the time of the documentation. Second difference is in the name of the Role class; it should be “assignedAuthor”. It cannot be renamed to “testDoctor” as in the case of our example. The corresponding conformant CDA block is given in Figure 3.18. The time is gathered from the CDA document’s effective time.

- As it is seen in the mapping of NHDD concepts to Transmission Schemas and CDA (Figure 3.14), for the data elements, wrapper second-level CDA sections are created in the Transmission Schemas. Within these sections, the values for the data elements

```

<author typeCode="AUT" contextControlCode="OP">
  <time value="20080402102643"/>
  <assignedAuthor classCode="ASSIGNED">
    <id root="2.16.840.1.113883.3.129.1.1.1" extension="12345678905"/>
  </assignedAuthor>
</author>

```

Figure 3.18: The Corresponding Conformant CDA Block for author

are given with instances of CDA Entry classes which can be seen in Figure 3.19 and are described in Section 2.1.2 - HL7 CDA. They are; act, encounter, observation, observationMedia, organizer, procedure, regionOfInterest, substanceAdministration and supply. Almost all of them are used in the NHIS messages. However, not surprisingly, substantial inconsistent changes are done on the original CDA Schema for these classes as well. It is not possible to list all these modifications here since renaming of the elements occurs for almost each attribute of entry classes. Instead, the most common modifications are presented in this item.

Figure 3.20 shows an example Discharge Diagnosis Section from the Examination Transmission Schema. The first major difference is; the CDA entry classes must be bound to sections with the “entry” participation as shown in Figure 3.19 but in the NHIS implementation “entry”s are replaced with “component”s. In the same manner, it an “entry” has “enrtyRelationship”, this is again modeled as “component” in the NHIS implementation. As the second major difference, the original name of the CDA entry classes are replaced with the name of the data elements in the NHDD. In our example, the “classCode” attribute of “dischargeDiagnosis” element reveals that this class is in fact an Observation. The corresponding conformant CDA block should look like as in Figure 3.21

14. The final major modification is the data type constraint applied on some XML elements which have abstract data type declarations originally. One example is the “value” element of the “observation” CDA Entry class. Its data type is “ANY”; the abstract HL7 data type. In the NHIS, according to the needs, the type of “value” is restricted to non-abstract types directly on the XSD level, which is against the CDA conformance. What should be done is specifying the required data type in the CDA document with the use of “xsi:type” attribute where “xsi” stands for XML Schema Instance [100]. An example use is already given in Figure 3.20 and Figure 3.21. No xsi:type declaration

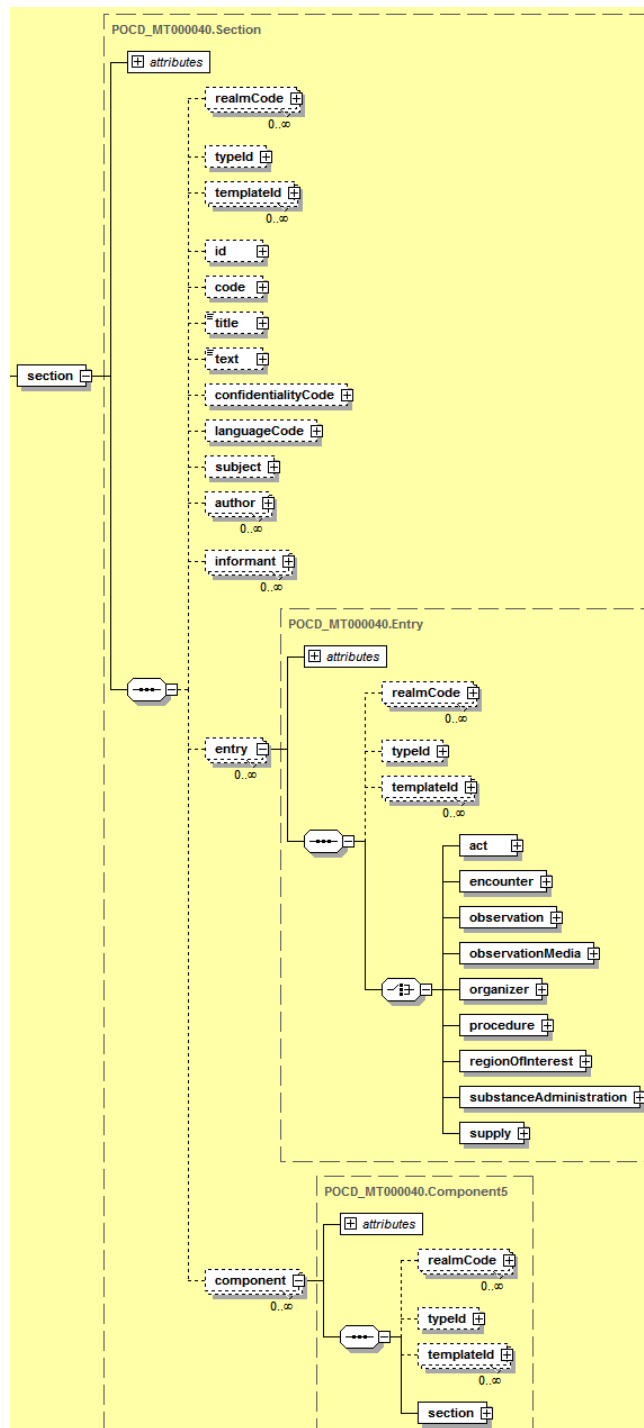


Figure 3.19: The Graphical View of the Schema of CDA Section

is available in the “value” element of dischargeDiagnosis. However, its type is “CV” (Coded Value) and it should be declared as shown in the conformant CDA block.

There is one more important conformance issue but this is not related with the schema of CDA, instead with the semantics. The section.text field of CDA is used to store narrative

```

<dischargeDiagnosisSection classCode="DOCSECT" moodCode="EVN">
  <id root="2.16.840.1.113883.3.129.2.1.5" extension="b66aa560-8b8e-4078-8236-d7edcdf5ce3a"/>
  <code code="TANI" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
    codeSystemVersion="1.0" displayName="Tanı Verisinin Olduğu Bölüm"/>
  <text mediaType="text/x-hl7-text+xml"/>
  <component typeCode="COMP" contextConductionInd="true">
    <dischargeDiagnosis moodCode="EVN" classCode="OBS">
      <code code="SEVKTANISI" codeSystem="2.16.840.1.113883.3.129.2.2.6" codeSystemName="Tanı Tipi"
        codeSystemVersion="1.0" displayName="Sevk Tanısı"/>
      <value code="K25" codeSystem="2.16.840.1.113883.6.3" codeSystemName="ICD-10"
        codeSystemVersion="1.0" displayName="Mide ülseri"/>
    </dischargeDiagnosis>
  </component>
</dischargeDiagnosisSection>

```

Figure 3.20: An Example dischargeDiagnosisSection from the NHIS

```

<section classCode="DOCSECT" moodCode="EVN">
  <id root="2.16.840.1.113883.3.129.2.1.5" extension="b66aa560-8b8e-4078-8236-d7edcdf5ce3a"/>
  <code code="TANI" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
    codeSystemVersion="1.0" displayName="Tanı Verisinin Olduğu Bölüm"/>
  <text mediaType="text/x-hl7-text+xml"/>
  <entry typeCode="COMP" contextConductionInd="true">
    <observation classCode="OBS" moodCode="EVN">
      <code code="SEVKTANISI" codeSystem="2.16.840.1.113883.3.129.2.2.6" codeSystemName="Tanı Tipi"
        codeSystemVersion="1.0" displayName="Sevk Tanısı"/>
      <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="CV" code="K25"
        codeSystem="2.16.840.1.113883.6.3" codeSystemName="ICD-10" codeSystemVersion="1.0"
        displayName="Mide ülseri"/>
    </observation>
  </entry>
</section>

```

Figure 3.21: The Corresponding Conformant CDA section for dischargeDiagnosisSection

to be rendered and thus called Narrative Block. CDA conformance rules state that if the CDA Body is structured (which is true in our case), the Narrative Block must be rendered; thus should not be empty. A section may only have no narrative content in the case where the entries represent information that is not part of the clinical content of the document. However, in the NHIS case, CDA is used for exchanging clinical data in the forms of data elements which are coded as much as possible as defined in the Health Coding Reference Server (HCRS). For this reason, although they exist within the messages, most of the time the narrative content of the sections are not filled with data. An example procedure section from a real Examination Transmission Schema is given in Figure 3.22.

```

<procedureSection classCode="DOCSECT" moodCode="EVN">
  <id root="2.16.840.1.113883.3.129.2.1.5" extension="cf813e90-e027-4441-9e87-c287bdc5b370"/>
  <code code="MUDAHALE" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
    codeSystemVersion="1.0" displayName="Müdahale Verisinin Olduğu Bölüm"/>
  <text/>
  <component typeCode="COMP" contextConductionInd="true">
    <procedure moodCode="EVN" classCode="PROC">
      <code code="700490" codeSystem="2.16.840.1.113883.3.129.1.2.2" codeSystemName="SUT"
        codeSystemVersion="1.0" displayName="Elektrokardiyogram, evde çekim"/>
    </procedure>
  </component>
</procedureSection>

```

Figure 3.22: An Example procedureSection from the NHIS

As it is clear, the “text” element of the section is empty. There is not a restriction in Turkey stating that “text” elements should never be filled in. But, since the mandatory data is sent in the “procedure” element as the coded value meaning “Elektrokardiyogram, evde çekim” (Electrocardiogram, home setting) in SUT: Sağlık Uygulama Talimatı (Health Application Instruction) code system of HCRS, the clients feel no obligation to fill in the “text” field. In these cases, the Narrative Block of the section can be compiled from its Entry classes. In this example, the display name of the procedure code can be copied to the text field. The corresponding conformant CDA section is presented in Figure 3.23.

```

<section classCode="DOCSECT" moodCode="EVN">
  <id root="2.16.840.1.113883.3.129.2.1.5" extension="cf813e90-e027-4441-9e87-c287bdc5b370"/>
  <code code="MUDAHALE" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
    codeSystemVersion="1.0" displayName="Müdahale Verisinin Olduğu Bölüm"/>
  <text>Elektrokardiyogram, evde çekim</text>
  <entry typeCode="COMP" contextConductionInd="true">
    <procedure moodCode="EVN" classCode="PROC">
      <code code="700490" codeSystem="2.16.840.1.113883.3.129.1.2.2" codeSystemName="SUT"
        codeSystemVersion="1.0" displayName="Elektrokardiyogram, evde çekim"/>
    </procedure>
  </entry>
</section>

```

Figure 3.23: The Corresponding Conformant CDA section for procedureSection

These are all the major incompatible cases of NHIS Transmission Schemas with the HL7 CDA R2. It is not possible to list every individual inconsistency of each Transmission Schema

in this thesis due to space limitations. This is not a problem because the described cases cover all varieties of incompatible modifications. The rest are just different examples of these generic cases.

3.2 The Developments for Enabling CDA Conformance of NHIS Transmission Schemas

The inconsistencies of the NHIS Transmission Schemas with the HL7 CDA R2 have been listed in the previous section. In order to get rid of these inconsistencies, an Adaptor for automatically transforming NHIS Transmission Schema instances to HL7 CDA R2 conformant EHRs has been developed.

The implementation is based on Extensible Stylesheet Language Transformations (XSLT) [103]. XSLT is an XML-based language used for the transformation of XML documents into other XML or “human-readable” documents such as HTML or plain text. A transformation expressed in XSLT describes rules for transforming a source tree into a result tree. During this process, the source tree is never edited. The details of XSLT can be found in Section 2.5.

In order to increase re-usability, the XSLT files have been developed per Minimum Health Data Set. Then Transmission Schema XSLT files have been built by including the relevant XSLTs of MHDSs. For example, Examination Transmission Schema consists of:

- Newborn Registration Data Set | Citizen Foreigner Registration Data Set
- Patient Admission Data Set
- Examination Data Set
- Patient Discharge Data Set
- Test Result Data Set (optional)
- Prescription Data Set (optional)

Very similarly, Diabetes Transmission Schema contains all of these MHDSs and an additional Diabetes MHDS. There are many similar cases for other Transmission Schemas as well. The development of MHDS based XSLT library increased the manageability and re-usability of the system.

In this section, some example XSLT rules are presented for a few selected items from the list presented in Section 3.1. With the same reason as in Section 3.1, it is not possible to present all XSLT rules in this section. The examples will again be given upon the Examination Transmission Schema and the complete XSLT file for Examination TS is available in our Web page [58]. Then some information about the implementation details will be presented.

3.2.1 Example XSLT Rules

The first example XSLT rule is for item 3 of the inconsistency list. This item is about the incompatible recordTarget content. An example invalid recordTarget for Newborn Registration is presented in Figure 3.4. The inconsistencies include both renaming of the elements and generation of new elements. The XSLT rule developed for transforming Newborn Registration recordTarget to CDA conformant recordTarget is presented with comments in Figure 3.24.

The first line declares that this rule will only be active when it matches a “babyPatientRole”. In the positive case, it generates “recordTarget” and “patientRole” elements. Then it generates an “id” element for the “patientRole” because it is mandatory but did not exist in the “babyPatientRole”. Then the “patient” element is created which is a replacement of the incompatible “patientBaby” element. The “birthTime” and “administrativeGenderCode” attributes are copied but “multipleBirthOrderNumber” is skipped since there is no such attribute of the “patient” Entity class. This value is moved to the “id” of the “patientRole”. Finally, the transformation of the “guardian” block is done. The CDA conformant output of the example input of Figure 3.4 can be found in Figure 3.5.

The second example XSLT rule is again related with an inconsistency in the CDA Header; item 6 in the inconsistency list. This item is about the “informationRecipient” attribute of CDA Header. Three renaming were done on the original attribute and they are explained in item 6. A related invalid “primaryInformationRecipient” example is shown in Figure 3.8. The XSLT code block for transforming “primaryInformationRecipient” into conformant “informationRecipient” is shown in Figure 3.25.

Unlike the first example, this rule does not initially check whether such an element named “primaryInformationRecipient” exists or not because it is mandatory. First an “informationRecipient” element is created and the XML attributes of “primaryInformationRecipient” are copied with the XSL statement in the second line. The same are done for the child and descendant elements of the “primaryInformationRecipient”, again with necessary renaming. As a result, the XSLT outputs the CDA conformant XML in Figure 3.26 for the input in

```

<xsl:if test="hl7:babyPatientRole">
  <xsl:element name="recordTarget">
    <xsl:element name="patientRole">
      <!-- id is an mandatory element of patientRole, so I had to generate it for baby patient -->
      <!-- id.extension is in the format motherID.birthtime.multiplebirthorder, the root is the
      same with the patient's -->
      <xsl:element name="id">
        <xsl:attribute name="root">
<xsl:value-of select="//hl7:guardianMother/hl7:id/@root"/></xsl:attribute>
        <xsl:attribute name="extension">
          <xsl:value-of select="//hl7:guardianMother/hl7:id/@extension"/>.
          <xsl:value-of select="//hl7:patientBaby/hl7:birthTime/@value"/>.
          <xsl:value-of select="//hl7:patientBaby/hl7:multipleBirthOrderNumber/@value"/>
        </xsl:attribute>
      </xsl:element>
      <xsl:for-each select="//hl7:babyPatientRole/hl7:addr">
        <xsl:copy-of select="." copy-namespaces="no"/>
      </xsl:for-each>
      <xsl:element name="patient">
        <xsl:copy-of select="//hl7:patientBaby/hl7:administrativeGenderCode" copy-namespaces="no"/>
        <xsl:copy-of select="//hl7:patientBaby/hl7:birthTime" copy-namespaces="no"/>
        <xsl:element name="guardian">
          <xsl:copy-of select="//hl7:patientBaby/hl7:guardian/@*" copy-namespaces="no"/>
          <!-- I had to move id of the guardianMother to here -->
          <xsl:copy-of select="//hl7:patientBaby//hl7:guardianMother/hl7:id" copy-namespaces="no"/>
          <xsl:copy-of select="//hl7:patientBaby/hl7:guardian/hl7:code" copy-namespaces="no"/>
          <!-- I had to add this empty guardianPerson element, it is mandatory. -->
          <xsl:element name="guardianPerson"/>
        </xsl:element>
      </xsl:element>
    </xsl:element>
  </xsl:element>
</xsl:if>

```

Figure 3.24: XSLT Rule for Newborn Registration recordTarget

Figure 3.8.

The final example XSLT rule covers four items from the inconsistency list; items 10, 11, 13 and 14. The XSLT block presented in Figure 3.27 is an excerpt from the Discharge Data Set XSLT file.

The XSLT block is composed of three templates. The first template tries to match “dischargeDataset” and when it is successful, it instantly opens a new “section” element. This is related with item 10 in the inconsistency list; the “section” elements cannot be renamed. Then for each “component1” element, the first template calls the second template called

```

<xsl:element name="informationRecipient">
  <xsl:copy-of select="hl7:primaryInformationRecipient/@*" copy-namespaces="no"/>
  <xsl:element name="intendedRecipient">
    <xsl:copy-of select="hl7:primaryInformationRecipient/*/@*" copy-namespaces="no"/>
    <xsl:element name="receivedOrganization">
      <xsl:copy-of select="hl7:primaryInformationRecipient/*/hl7:representedMinistryOfHealth/@*
|hl7:primaryInformationRecipient/*/hl7:representedMinistryOfHealth/*" copy-namespaces="no"/>
    </xsl:element>
  </xsl:element>
</xsl:element>

```

Figure 3.25: XSLT Rule for primaryInformationRecipient

```

<informationRecipient typeCode="PRCP">
  <intendedRecipient classCode="ASSIGNED">
    <receivedOrganization classCode="ORG" determinerCode="INSTANCE">
      <id root="2.16.840.1.113883.3.129.1.1.6" extension="5881"/>
    </receivedOrganization>
  </intendedRecipient>
</informationRecipient>

```

Figure 3.26: The Conformant Output for primaryInformationRecipient

“dischargeDataset-component1” that is created for handling the “dischargeDiagnosisSection” element. This template is related with item 11; it renames “component1” back to “component” and “dischargeDiagnosisSection” to “section”. Then it copies the basic fields of the section and finally calls the third template for each “component” of the “dischargeDiagnosisSection”. This last template is related with items 13 and 14; it first renames the “component” to “entry” and “dischargeDiagnosis” to “observation” Entry class. While copying the “value” field of “dischargeDiagnosis” in which ICD-10 codes are stored, the template adds the mandatory xsi type declaration as described in item 14. The example input and output is available in Figure 3.20 and Figure 3.21, respectively.

The CDA conformant transformation of the Examination Transmission Schema example presented in Appendix A, realized by applying the XSLT files provided in our web page, is available in Appendix B and in our web page [53]. XSLT files cannot be provided in the Appendix because they are quite large.

3.2.2 Implementation Details

The major work on the implementation side was the development of XSL Transformations. XSLT version 2.0 [104] was preferred due to its enhanced XML-tree copying functionalities,

```

<xsl:template match="hl7:dischargeDataset">
  <xsl:element name="section">
    ...
    <xsl:for-each select="./hl7:component1">
      <xsl:call-template name="dischargeDataset-component1"/>
    </xsl:for-each>
    ...
  </xsl:element>
</xsl:template>
<xsl:template name="dischargeDataset-component1">
  <xsl:element name="component">
    <xsl:copy-of select="@*" copy-namespaces="no"/>
    <xsl:element name="section">
      <xsl:copy-of select="hl7:dischargeDiagnosisSection/@*" copy-namespaces="no"/>
      <xsl:copy-of select="hl7:dischargeDiagnosisSection/hl7:id" copy-namespaces="no"/>
      <xsl:copy-of select="hl7:dischargeDiagnosisSection/hl7:code" copy-namespaces="no"/>
      <xsl:copy-of select="hl7:dischargeDiagnosisSection/hl7:text" copy-namespaces="no"/>
      <xsl:for-each select="hl7:dischargeDiagnosisSection/hl7:component">
        <xsl:call-template name="dischargeDataset-component1-component"/>
      </xsl:for-each>
    </xsl:element>
  </xsl:element>
</xsl:template>
<xsl:template name="dischargeDataset-component1-component">
  <xsl:element name="entry">
    <xsl:copy-of select="@*" copy-namespaces="no"/>
    <xsl:element name="observation">
      <xsl:attribute name="classCode">OBS</xsl:attribute>
      <xsl:attribute name="moodCode">EVN</xsl:attribute>
      <xsl:copy-of select="hl7:dischargeDiagnosis/hl7:code" copy-namespaces="no"/>
      <xsl:element name="value">
        <xsl:attribute name="xsi:type">CV</xsl:attribute>
        <xsl:copy-of select="hl7:dischargeDiagnosis/hl7:value/@*" copy-namespaces="no"/>
      </xsl:element>
    </xsl:element>
  </xsl:element>
</xsl:template>

```

Figure 3.27: An Excerpt from the Discharge Data Set XSLT File

such as discarding the unnecessary namespace declarations. During the development process, the generated XSLT files have been tested with various XSLT processors such as Altova XMLSpy [97] and various versions of Microsoft MSXML [52].

The real implementation took place on the Java Platform, Standard Edition 6. Open source Saxon XSLT processor [72] is preferred as the built-in XSLT processor of the Adaptor,

i.e. NHIS2CDA Transformer. The latest version of Saxon is version 9.1.0.1. This is a complete and conformant implementation of the XSLT 2.0, XQuery 1.0, and XPath 2.0. Saxon is very stable and widely used by the developers interested in XML technologies.

The implementation is designed as a service rather than an attractive GUI in fact. It supports two methods; the first one is getting any NHIS Transmission Schema instance belonging to any of the 25 HL7 Web Services in the string representation and returning the HL7 CDA R2 conformant EHR correspondence again in the string representation. The second method does the same functionality but this time accepts and outputs files. The input may be just the EHR content of Turkey's HL7 v3 messages or the complete HL7 v3 message (containing both the Transmission and Control Act Wrappers) or the complete SOAP Envelope element that are accepted by the HL7 v3 Web Services; the Adaptor is able to successfully locate the EHR content of the input in every case. Furthermore, the Adaptor does not discriminate the insert and update messages of the NHIS. Although they have slightly different schemas, the XSLT rules are designed to handle both cases at once without requesting extra information from the user.

Although the main purpose was to develop the NHIS2CDA Transformer as a service, a simple GUI has also been developed. Indeed, the GUI is multi-purpose; only the NHIS2CDA functionality is covered in this chapter. The rest are covered in the upcoming chapters. For this reason, the application has a more generic name; Transformation Environment for HL7 Messages of NHIS, Turkey.

Figure 3.28 presents the start-up screen of the Transformation Environment. A new project can be started via the File -> New Project menu item. As it seen in Figure 3.29, the application has a tabbed workspace so that each project has a dedicated tab. Each tab has an "Operations" panel at the top and two text areas in the middle, the one on the left as the input text area and the right as the output text area. The "Operations" panel at the top contains six buttons for six different functionalities. "Open input" opens the selected input file in the input text area and "Save output" saves the content of the output text area to the selected file. "Close" removes the tab from the workspace.

Each of the rest three buttons in the "Operations" panel is related with each of the three chapters of this thesis. "NHIS2CDA" button is of interest to this chapter while the others are related with the two following chapters and thus will not be mentioned here.

When the "NHIS2DA" button is pressed, the Transformation Environment reads the input text area, calls the NHIS2CDA Transformer with this input. The transformer applies XSL Transformation on the input with the previously developed XSLT files. After the trans-

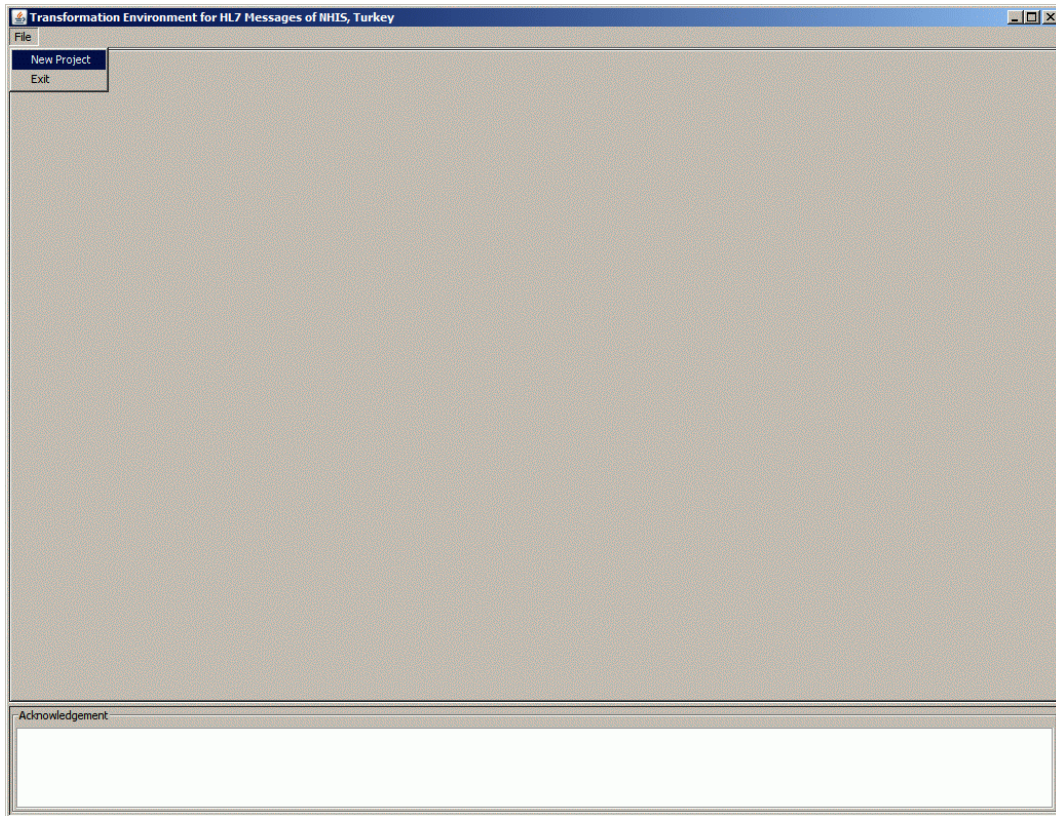


Figure 3.28: The Start-up Screen of the Transformation Environment

formation, the application runs the XSD Validator on the output and checks its conformance against original CDA Schema. The result of the validation process is printed in the “Acknowledgement” panel at the bottom of the workspace as it can be seen in Figure 3.30. If the validation fails, the details of the errors are presented to the user as well. Apache Xerces XML library [94] is used for XSD validation. Finally the output gathered from the transformer is printed on the output text area as in Figure 3.30.

The developed XSLT files and the implementation will be provided to the Ministry of Health. The transformation operation will be deployed as a Web Service that will be acting as a proxy between the Ministry of Health and the cross-border countries capable of healthcare data exchange, hopefully within the next few years. Moreover, MoH is planning to archive the EHRs received from the healthcare organizations in native CDA format. For this purpose, the transformer will be deployed as a front-end to the centralized database.

Another use of the Adaptor may be on the NHIS clients’ side; the clients can store the generated Transmission Schemas in native CDA format so that when a change has been done on the Web Service schemas, the previous records will still be accessible. The clients may

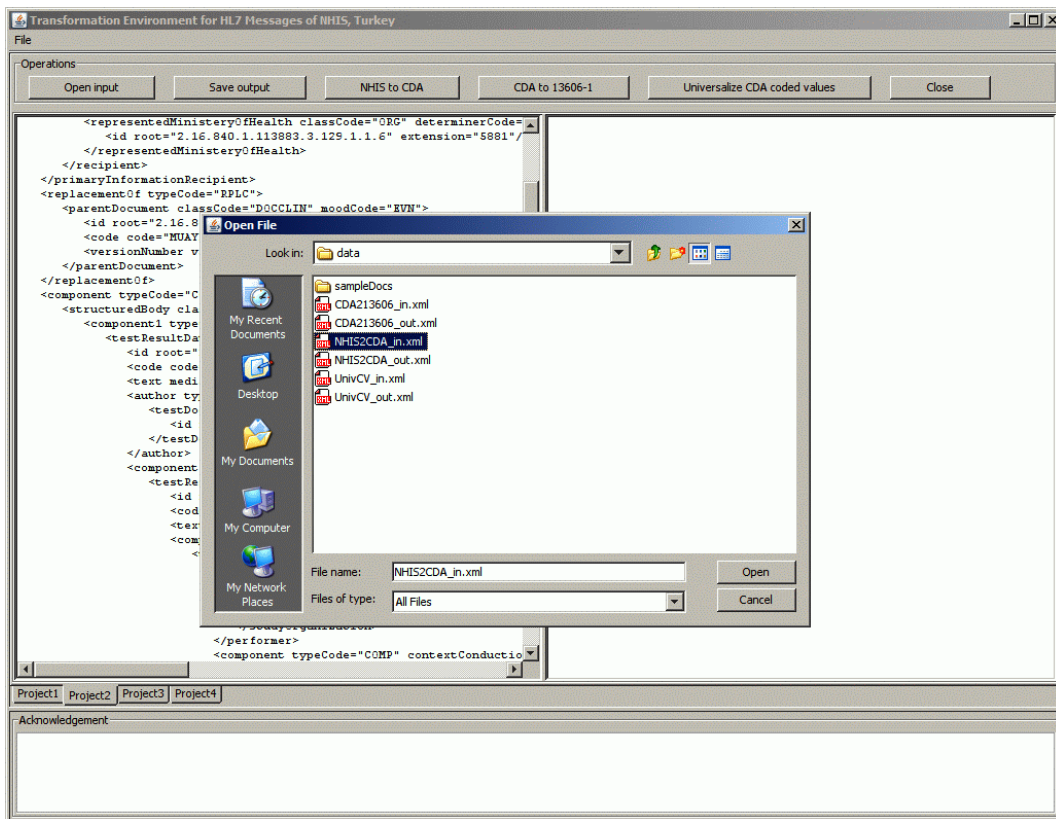


Figure 3.29: Opening New Projects and Input Files

also directly use the Web Services deployed at the Ministry of Health for this purpose. It is expected that annual updates/changes will be done on the National Health Data Dictionary which will affect the Transmission Schemas directly.

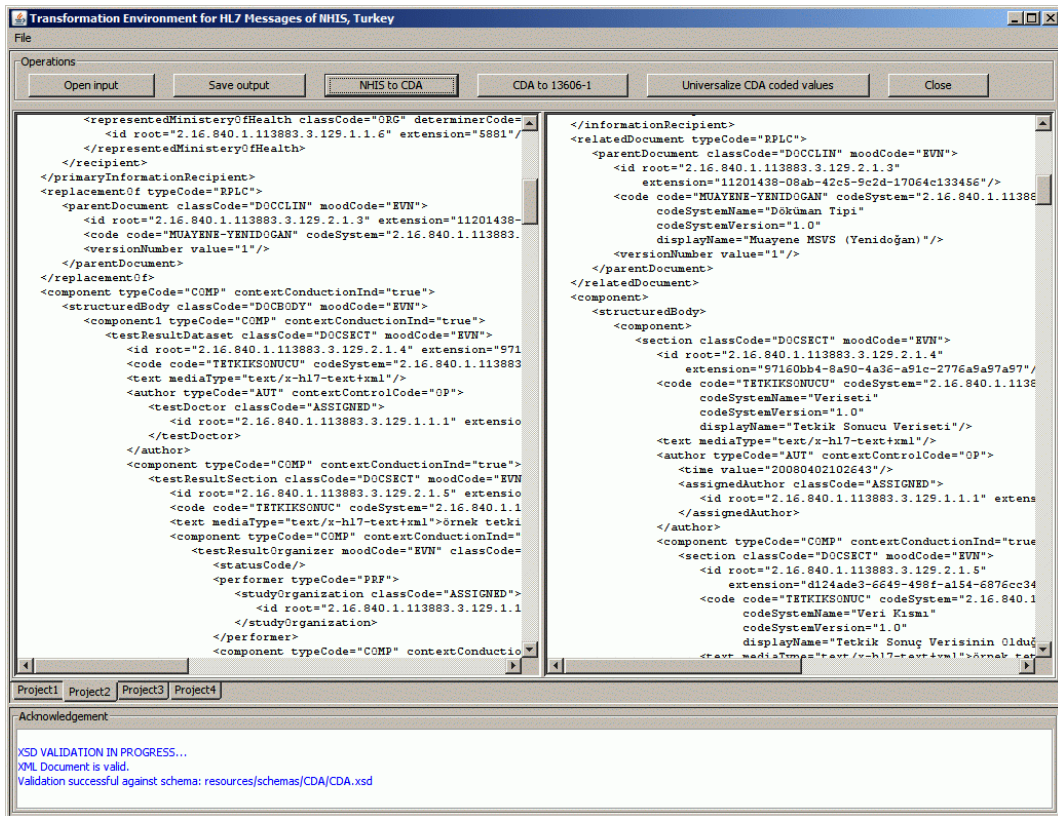


Figure 3.30: NHIS2CDA Transformer in Action

CHAPTER 4

AUTOMATIC MAPPING OF LOCAL CODED TERMS IN EHR DOCUMENTS TO INTERNATIONAL COUNTERPARTS

When the conformance of locally developed Electronic Healthcare Records to international standards such as HL7 CDA or CEN EN 13606-1 is realized (example cases for Turkey are given in Chapters 3 and 5), the structure level semantics of the EHR instances become understandable by everyone (system, organization, etc.) who is able to process these standards. This is truly an initial important step on the way to achieving interoperability. With the help of structure level interoperability, at least the EHR document can be processed on the receiver side. In the case of CDA for instance, the receiver knows that she can find the patient demographics information under the recordTarget element. Or, when she sees instances of CDA entry classes like “act”, “observation”, “procedure”, etc., she will know what these classes are used for.

However, there are still things to be done at the content level semantics of the EHR documents in order to converge to complete interoperability among the actors with different settings. Consider the case of EHR exchange between two countries speaking different languages; although they conform to the same EHR standard structure, it is not possible for them to extract the written and/or locally coded medical content. Similar problem may even appear between actors speaking the same language because of the local coding systems that are used within the documents. It is necessary to translate these local elements to their international counterparts.

For this purpose, as a part of our work, a Terminology Server based modular architecture enabling automatic mapping of the local coded terms that exist in the EHR documents to

international counterparts has been developed. As the Terminology Server, the Unified Medical Language System (UMLS) Knowledge Source Server (KSS) [84] has been used. UMLS provides secure Web Services for querying the KSS for finding clinical concepts, retrieving concept details, mapping clinical terms, etc. These Web Services have been used for automatic translation of the local coded terms to international counterparts. The details of the UMLS Web Services are presented in the following sub-section.

As the case study, again the National Health Information System of Turkey is chosen. The NHIS messages were transformed into HL7 CDA R2 conformant EHR documents as a result of the work described in the previous chapter. Now, the local coded terms that exist in these CDA documents are replaced with international counterparts if possible. As the universal language, English is chosen for now but the system can be extended for other languages. There are very specific and detailed coding systems in Turkey, so sometimes it is impossible to find a code from a universal coding system in the UMLS KSS. Such a case is called worst case and as a solution, the coded value is replaced with its direct English meaning.

For instance, now it is possible to automatically replace code “EKTANI” meaning “Ek Tanı” (secondary diagnosis) from the local “Tanı Tipi” (diagnosis type) code system with code “85097005” meaning “Secondary diagnosis” from the universal “SNOMED CT” [75] code system automatically.

In this chapter, we first provide information about the UMLS Web Services and then present the developments.

4.1 UMLS Knowledge Source Server Web Service API

The Unified Medical Language System (UMLS) Knowledge Sources and related lexical programs, developed at the U.S. National Library of Medicine (NLM), provide access to the UMLS. The UMLS Knowledge Source Server (UMLSKS) is the set of machines, programs and Application Programmer Interfaces (APIs), written in Java, located and maintained by staff at the NLM that allow access to the UMLSKS services [82].

For a long time, UMLSKS is providing a Java Messaging Service (JMS) based API but it has some disadvantages like the Java dependency and the necessity of registering each and every IP from which the API methods are called. Moreover, some methods are always resulting with exceptions with any kind of input. Currently, UMLSKS is also sharing a secure Web Service API with functions for retrieving Metathesaurus, Semantic Network,

and SPECIALIST Lexicon data from the UMLSKS.

Independent of the Web Service API, in order to benefit from any UMLS service including the Web interface, the users need to register. When the registration is confirmed, a developer certificate is emailed to the user which is necessary for accessing the Web Services. This certificate enables client programs to authenticate users and to obtain single-use tickets for embedding in Web Service API calls. There are two Web Services provided by the UMLSKS; the first one is the Authentication Web Service (WSDL available at: <http://-umlsks.nlm.nih.gov/authorization/services/AuthorizationPort?wsdl>) and the second is the UMLSKS Web Service (WSDL available at: <http://umlsks.nlm.nih.gov/UMLSKS/services-/UMLSKSService?wsdl>). They both support Simple Object Access Protocol (SOAP) 1.2 [76].

The basic steps for obtaining UMLS data from the UMLSKS Web Services are as follows:

1. Establish a connection to the Authentication web service.
2. Obtain a proxy granting ticket (PGT) from the Authentication web service.
3. Using the PGT, obtain single-use tickets from the Authentication web service to embed in each UMLSKS web service call.
4. Build the call argument(s) to the UMLSKS
5. Request the data from the UMLSKS web service.

If Java is used as the development platform, UMLS suggests using the Apache Axis Web Service library version 1.4 [3] which is also preferred in our implementation.

4.1.1 Authentication Web Service

The following code snippet shows how to establish a connection to the Authentication Web Service and request tickets:

A proxy granting ticket is generated by the Authentication Web Service using the username and password combination of a valid user of the UMLSKS. The developer certificate is also requested for Secure Sockets Layer (SSL) setup. Once a proxy granting ticket is generated, the client application may request generation of single-use proxy tickets which are necessary for the API calls to the UMLSKS Web Service. Each UMLSKS Web Service call must have a single-use proxy ticket embedded in it or it will generate an exception.

```

// Properties for SSL Security Provider
String protocolProp = "java.protocol.handler.pkgs";
String sunSSLProtocol = "com.sun.net.ssl.internal.www.protocol";
String sslStoreProp = "javax.net.ssl.trustStore";

// Enable SSL communication
System.setProperty(protocolProp, sunSSLProtocol);
Security.addProvider(new com.sun.net.ssl.internal.ssl.Provider());
System.setProperty(sslStoreProp, certPath);

// Locate the authentication web service
URL authURL = new URL(host + authExtension);
AuthorizationPortSoapBindingStub authStub = new AuthorizationPortSoapBindingStub(authURL, null);

// Obtain a proxy granting ticket
String pgt = authStub.getProxyGrantTicket(username, password);

// Obtain a single-use proxy ticket
String proxyTicket = authStub.getProxyTicket(pgt, host);

```

Figure 4.1: Establish a Connection to the Authentication Web Service

4.1.2 UMLSKS Web Service

The UMLSKS Web Service provides access to data through the 49 operations defined in its WSDL. More information about some of the operations will be provided in the following section. The below code snippet shows an example call for find a Concept Unique Identifier (CUI) using an exact matching of an input string. `ConceptIdExactRequest` is generated for the `findCUIByExact` call for this purpose. All the operations have a similar structure requesting the single-use proxy ticket as a part of the request. It is also possible to state the release and language of the UMLS.

4.1.3 UMLSKS Web Service Operations

As mentioned earlier, UMLSKS Web Service API provides 49 operations for retrieving Metathesaurus, Semantic Network, and SPECIALIST Lexicon data. These operations have perfect documentation at the UMLSKS Developer's Guide [83]. Here, some generic information about the operations is presented.

The operations are grouped under five categories:

- *General Operations* provide UMLSKS specific information such as returning the current UMLS release.


```

// Locate the UMLSKS web service
URL ksURL = new URL(host + ksExtension);
UMLSKSServiceSoapBindingStub ksStub = new UMLSKSServiceSoapBindingStub(ksURL, null);

// Build the request object
ConceptIdExactRequest request = new ConceptIdExactRequest();
request.setCasTicket(proxyTicket);
request.setRelease(release);
request.setLanguage(lang);
request.setSearchString(searchString);

// Execute the operation
ConceptIdGroup group = ksStub.findCUIByExact(request);

// Print the results
Object[] contents = group.getContents();
for (int i = 0; i < contents.length; i++) {
    ConceptId cid = (ConceptId)contents[i];
    System.out.println("CUI: " + cid.getCUI());
}

```

Figure 4.2: Obtaining UMLS Data from UMLSKS Web Service

- *Metathesaurus Operations* provide access to the resources of the Metathesaurus, such as locating a Concept Unique Identifier (CUI), retrieving concept details, retrieving mappings between source vocabularies.
- *Semantic Network Operations* provide access to the resources of the Semantic Network, such as locating semantic network nodes, retrieving semantic relation properties and ancestors.
- *Lexical Operations* provide access to the resources of the SPECIALIST Lexicon as well as other useful lexical operations, such as retrieving lexical records, returning spelling suggestions.
- *Raw Record Operations* provide access to the raw data of the three Knowledge Sources. SQL is supported in these operations.

As an example specific Metathesaurus operation, “Get Concept Properties” locates details about a concept based on the flag settings given as input. The concept details include these elements: terminology, definition(s), semantic type(s), hierarchical source vocabulary contexts(s), relations with other concepts, cooccurrences in NLM literature, attributes attached to a concept and concept’s relationships.

As another Metathesaurus operation, which proved to be very useful in our work, “Find Lexical Unique Identifier” returns the set of string lexical details for those strings in the Metathesaurus whose normalizations match the normalizations of the input term. The results may be further refined by specifying a set of sources to search and a language to match. With this operation, it is possible to search the string “asthma” in UMLS release 2008AA for English language and get the list of terms presented in Table 4.1 from various coding systems.

Table 4.1: The Result of the findLUI Operation with Search String “asthma”

Code System	Code	Display Name
CCPSS	1017373	ASTHMA
COSTAR	80	ASTHMA
CST	ASTHMA	ASTHMA
DXP	U000273	ASTHMA
OMIM	605010.0004	ASTHMA
SNMI	D2-51000	ASTHMA
WHO	1367	ASTHMA
CCS	128	Asthma
CCS	8.3	Asthma
ICD10	J45	Asthma
ICD10AM	J45	Asthma
ICD9CM	493	Asthma
ICPC	R96	Asthma
ICPC2EENG	R96	Asthma
ICPC2P	R96001	Asthma
LCH	U000405	Asthma
LOINC	MTHU020815	Asthma
MDR	10003553	Asthma
MEDLINEPLUS	T2	Asthma
MSH	D001249	Asthma
MTH	NOCODE	Asthma
NCI	C28397	Asthma
NDFRT	C1098	Asthma
OMIM	MTHU003537	Asthma
PSY	4190	Asthma
RCD	H33	Asthma
SNM	D-4790	Asthma
SNOMEDCT	195967001	Asthma
MDR	10003560	Asthma NOS
SNOMEDCT	266365004	Asthma NOS
SNMI	D2-51000	Asthma, NOS
MSH	D001249	Asthmas
AOD	5951	asthma
BI	BI00009	asthma
CSP	1525-2157	asthma
ICPC2ICD10ENG	MTHU008836	asthma
ICPC2P	R96001	asthma

4.2 The Developments for the Architecture Enabling Automatic Coded Value Translation in CDA Documents

Based on the UMLSKS Web Services, we have developed and extensible architecture for automatic translation of local coded terms that exist within EHR documents to their universal equivalents. As the case study, again National Health Information System of Turkey is chosen. We have achieved the conformance of NHIS Transmission Schemas to HL7 CDA

R2 standard in the previous chapter. This was a critical step for achieving interoperability. Now, in order to enhance content level interoperability, the locally defined coded values will be replaced with codes from international code systems as much as possible. There are 177 code systems in the Health Coding Reference Server [74] and only 4 of them (ICD10, ICPC2, ICD10-ICPC2 Mapping and ATC) are international code systems. So there are tons of locally defined coded values, which makes Turkey a perfect fit for our system. Our methodology is applicable for all locally defined EHRs as long as they involve coded values.

In this section, the developments in this respect are explained. First, we describe how to easily locate and gather the coded values from an EHR document. Then we explain the internal database that is built for storing the original and translated coded values and finally conclude with the translation methodology.

4.2.1 Locating the Coded Values

We have the CDA documents of NHIS Turkey as the input. So, our input format is XML. As explained in the Enabling Technologies and Standards section, the XML Path Language (XPath) [98] provides an efficient mechanism to address parts of an XML document and it is the de facto standard for this purpose.

A very simple XPath expression is sufficient to retrieve all the elements of a CDA document with HL7 data type CodedValue (CV):

```
//hl7:confidentialityCode//hl7:administrativeGenderCode//hl7:raceCode//hl7:code  
|//hl7:value[@xsi:type='CV']
```

Figure 4.3: XPath Expression to Locate All the CodedValues

This expression is enough to capture all the coded elements of a CDA document that are used in Turkey. Five individual expressions are combined with OR operation. The first one locates the confidentiality code of the document, the second one locates gender of the record target, the third one locates the race code of the record target, the fourth one is the most generic one since “code” element exists in all the classed that are derived from the “Act” class in the RIM and finally the last one captures the “value” attributes of the “Observation” class, which are used to provide some diagnosis codes in Turkey with specifying their data types as ‘CV’.

The CDA conformant output of an example Examination Transmission Schema is present in Appendix B. Figure 4.4 presents some of the results when XPath expression in Figure 4.3 is applied on the example CDA given in Appendix B. The “codeSystemVersion” attribute is not shown in this figure in order to save space.

```

<code code="MUAYENE-YENIDOGAN" codeSystem="2.16.840.1.113883.3.129.2.2.1"
  codeSystemName="Döküman Tipi" displayName="Muayene MSVS (Yenidoğan)">
<confidentialityCode code="1" codeSystem="2.16.840.1.113883.3.129.1.2.77"
  codeSystemName="Gizlilik" displayName="Normal">
<administrativeGenderCode code="1" codeSystem="2.16.840.1.113883.3.129.1.2.21" codeSystemName="Cinsiyet"
  displayName="Erkek">
<code code="TETKIKSONUCU" codeSystem="2.16.840.1.113883.3.129.2.2.2" codeSystemName="Veriseti"
  displayName="Tetkik Sonucu Veriseti">
<code code="700490" codeSystem="2.16.840.1.113883.3.129.1.2.2" codeSystemName="SUT"
  displayName="Elektrokardiyogram, evde çekim">
<code code="CIKIS" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
  displayName="Çıkış Verisinin Olduğu Bölüm">
<code code="4" codeSystem="2.16.840.1.113883.3.129.1.2.9" codeSystemName="Çıkış Şekli"
  displayName="Kurum dışına sevk">
<code code="TETKIK" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
  displayName="Tetkik Verisinin Olduğu Bölüm">
<code code="P608270" codeSystem="2.16.840.1.113883.3.129.1.2.2" codeSystemName="SUT"
  displayName="Bronkobiliyer fistül onarımı">
<code code="ANATANI" codeSystem="2.16.840.1.113883.3.129.2.2.6" codeSystemName="Tanı Tipi"
  displayName="Ana Tanı">
<value xsi:type="CV" code="K55" codeSystem="2.16.840.1.113883.6.3" codeSystemName="ICD-10"
  displayName="BARSAGIN VASKULER BOZUKLUKLARI">
<code code="ILAC" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
  displayName="İlaç Verisinin Olduğu Bölüm">
<code code="8699546011209" codeSystem="2.16.840.1.113883.3.129.1.2.3" codeSystemName="İlaçlar"
  displayName="ASPIRIN FORT TABLET 20 TB">
<code code="1" codeSystem="2.16.840.1.113883.3.129.1.2.48" codeSystemName="Reçete Türü"
  displayName="Normal">
<value xsi:type="CV" code="K25" codeSystem="2.16.840.1.113883.6.3" codeSystemName="ICD-10"
  displayName="Mide ülseri">

```

Figure 4.4: An Excerpt from the Output of the XPath Expression on the CDA Instance

In our implementation Saxon is used as the XPath evaluator.

4.2.2 Building the Local Database

As seen in Figure 4.4, there are many instances of local codes in a single EHR document and this is not the complete list in fact. Most local codes used in NHIS, Turkey are defined under the Health Coding Reference Server. There are 177 tables (i.e. code systems) in the HCRS and 173 of them are defined locally. These code systems are defined during the development of the National Health Data Dictionary. Some examples are:

- Fetus heart beat
- Patient discharge method
- Marital status
- Protein status in the urine
- Medication usage method
- Anemia treatment method
- Injury history
- Prescription type
- Clinics
- Addresses
- Vaccines
- Proficiencies

The complete list is available at http://sbu.saglik.gov.tr/SKRS2_Listesi/. It is obvious that there are very distinct and Turkey-specific code systems as well. The values of the “Fetus heart beat” code system are presented in Table 4.2.

Table 4.2: Fetus heart beat code system

Code	Name (TR)	Name (EN)
1	Kalp sesi yok	No heart beat
2	120 /dk dan az	Less than 120 /min
3	120-160 arası	Between 120 and 160 /min
4	160 /dk dan fazla	More than 160 /min

Further local code systems have been defined during the development of the Transmission Schemas based on CDA. These are for coding the ClinicalDocument (which represents the Transmission Schema), first-level sections (represent the Minimum Health Data Sets), second-level sections (represent the wrappers for data elements in the NHDD, called the data sections) and one more for classifying the diagnosis types. The values of the “Diagnosis type” code system are presented in Table 4.3.

Table 4.3: Diagnosis type code system

Code	Name (TR)	Name (EN)
ANATANI	Ana Tanı	Main diagnosis
EKTANI	Ek Tanı	Secondary diagnosis
SEVKTANISI	Sevk Tanısı	Referral diagnosis
CIKISTANISI	Çıkış Tanısı	Discharge diagnosis
KOMPLIKASYONTANISI	Komplikasyon Tanısı	Complication diagnosis

We are using the “Find Lexical Unique Identifier” operation of UMLS SKS Web Service for finding all the available coded terms for a search string. Our target language is English so, the search strings should be in English language. However, as seen in the example messages in Appendices and also in Figure 4.4, the `displayName` attributes of the coded elements are always in Turkish, naturally. We have to know the English translation of these display names for querying the UMLS SKS. An internal database is maintained both for this purpose and also for storing the results coming from the UMLS (or user-defined translations) so that the number of Web Service calls is reduced. Two classes that handle the management of coded values are designed (see the class diagram in Figure 4.5) and their instances are stored in the database as persistent objects.

The “CodedValue” class has five attributes:

- *code* holds the locally defined code
- *trDisplayName* holds the real display name in Turkish
- *enDisplayName* holds the English translation of the `trDisplayName`
- *prefCodeSystemName* holds the name of a universal code system which may be preferred by the users during the mapping process
- *variant* is an array of “UniversalCodedValue” class for holding the universal counterparts of the coded term in this “CodedValue”.

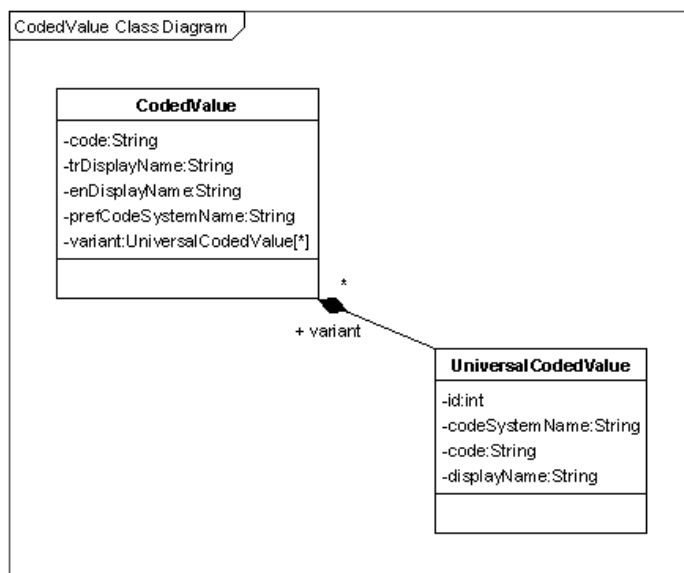


Figure 4.5: CodedValue Class Diagram for Persistent Objects

The “UniversalCodedValue” class has four attributes:

- *id* is an identifier generated automatically by the database
- *codeSystemName* holds the name of the universal code system
- *code* holds the code in the universal code system
- *displayName* holds the display name of the code in this class. Normally, it is expected to be the same with its parent “CodedValue” class’ *enDisplayName* but there may be some typographic differences which are handled by the UMLS.

The real instances of these classes for the local “Diagnosis type” code system that is presented in Table 4.3 are given in a tabular format in Table 4.4 and Table 4.5 (this representation is not the original database schema). The first table holds the elements of the original diagnosis type code system with three additional attributes as defined in the “Coded-Value” class. The only manual step of the complete system is entering the “enDisplayName”s of the local codes. If this attribute is null, nothing can be done regarding the mapping. It is not mandatory to provide a preferred code system, in our example two local codes have it; EKTANI and SEVKTANISI. If a value from this preferred code system occurs in the UMLS response, then it will be chosen. If it does not exist in the response, we have a priority list of universal code systems and selection is done on the basis of this list: 1) SNOMEDCT,

2) ICD10 and 3) LOINC which are the most widely used code systems. If none of them appears in the UMLS response, than the selection is done randomly. In any case, all the universal codes that are retrieved from the UMLSKS findLUI operation are stored in the database as they are; selection is done during the mapping phase. As another option, the user can oblige just English translation of the display name instead of mapping by inserting “ENGTRANSLATION” in the preferred code system field.

Table 4.4: “CodedValue” Instances for “Diagnosis Type” Code System

code	trDisplayName	enDisplayName	prefCodeSystemName	variant
ANATANI	Ana Tanı	Main diagnosis	NULL	{1, 2}
EKTANI	Ek Tanı	Secondary diagnosis	LOINC	{3, 4, 5, 6}
SEVKTANISI	Sevk Tanısı	Referral diagnosis	SNOMEDCT	{7}
CIKISTANISI	Çıkış Tanısı	Discharge diagnosis	NULL	{8}
KOMPLIKASYONTANISI	Komplikasyon Tanısı	Complication diagnosis	NULL	NULL

Table 4.5: Universal Variants for the “Diagnosis Type” Code System

id	codeSystemName	code	displayName
1	SNMI	G-1007	Main diagnosis
2	SNOMEDCT	8319008	Main diagnosis
3	LOINC	MTHU021495	Secondary diagnosis
4	MTH	U000281	Secondary diagnosis
5	SNMI	G-1008	Secondary diagnosis
6	SNOMEDCT	85097005	Secondary diagnosis
7	SNOMEDCT	406523004	Referral diagnosis
8	HL7V3.0	DISDX	discharge diagnosis

The “id”s of the universal counterparts of the coded values in Table 4.4 are given in the last column and the details of these counterparts are presented in Table 4.5. These values are retrieved from the UMLSKS. “Main diagnosis” has two, “secondary diagnosis” has four, “referral diagnosis” and “discharge diagnosis” have one and “complication diagnosis” has zero counterparts from various universal code systems. The results show that SNOMEDCT is the most dominant one.

Hibernate 3.2 is used for the management of persistent objects in the database. Hibernate is a powerful, high performance object/relational persistence and query service [33]. It is one of the most widely used Object-Relational Mapping (ORM) tools. The most obvious

advantage of using Hibernate in our work was being able to map one Java class to any number of database tables. This functionality is enabled by the “entity-name” property of Hibernate mappings. In our case, we need to keep a separate table for each of the locally defined code systems, which makes around 180. However, the schemas of all the tables are the same and we abstain from creating a new class for each table (which is very inefficient). Now, with only one Java class, we can reach any table by specifying its entity-name in the queries.

4.2.3 Translation Methodology

We have developed an algorithm for automatic translation of the locally defined coded elements. As it is seen in the previous section, translation may not be straightforward; for instance the UMLS can return zero coded terms for the search string. What needs to be done in these cases is determined by our algorithm. A pseudo-code like representation of the algorithm is presented below.

```
void translateNode(Node node) {

    // The object for accessing and querying the local database
    CodedValueRepository cvRep;

    // The object for accessing and querying the UMLS
    UMLSCClient umlsks;

    // If the codeSystem attribute is null, there is nothing we can do because we do not know
    the table
    if(node.codeSystem == null)
        return;

    // If the codeSystem does not exist in the local database yet, again there is nothing to do
    if(cvRep.checkTable(node.codeSystem) == null)
        return;

    // We are sure that we have a table in our db for the local codeSystem. Now, we are querying
    whether the specific ‘code’ appears in this table
    CodedValue cv = cvRep.retrieve(node.codeSystem, node.code);

    // If we don’t have this code in the table, then it means we do not know its English
    translation. So, return
    if(cv == null)
        return;

    // If both an English translation and any variant is missing, then nothing to do
    if(cv.enDisplayName == null && cv.variants == null)
```

```

        return;

// If ENGTRANSLATION is directly obliged in the db, then set the node.displayName
to cv.enDisplayName
if(cv.prefCodeSystemName != null && cv.prefCodeSystemName.equals(ENGTRANSLATION)) {
    node.displayName = cv.enDisplayName;
}

// Else, we can look at the variants now
else {

    // If the variants are null, do the UMLS query
    if(cv.variants == null) {
        cv.variants = umlsks.getClinicalTerms(cv.enDisplayName);

        // Since a UMLS query has been made, update the database with the retrieved values
        cvRep.update(node.codeSystem, cv);
    }

    // The variants were null, we queried the UMLS just a second ago but still they
    may be null if the UMLS response was empty. In this case, the only option is to do
    the English translation
    if(cv.variants() == null) {
        node.displayName = cv.enDisplayName;;
    }

    // Else, it means that we have at least one variant. We will definitely do a
    universalization in this block
    else {

        // We have to make a selection from the available coded terms from the universal
        code systems, which are currently stored in the cv.variants. The selection
        criteria is simple; if the variants contain a coded term from ‘prefCodeSystemName’,
        select it. Else, if variants contain a coded term from SNOMEDCT, select it. Else,
        if variants contain a coded term from ICD10, select it. Else, if variants contain
        a coded term from LOINC, select it. Else, select one of the variants randomly.
        int selected = doSelection(cv.variants);

        // After the selection is complete, update the attributes of the input CDA node
        with the selected universal coded term
        updateNode(node, selected);
    }
}
}
}

```

The details of the algorithm are presented with comments. Basically, we have a CDA node of type “CV” to be translated. We check whether its codeSystem attribute exists in the

document and it is non-empty, then we control our database whether we have already created a table for its elements. In the negative case, we abort the algorithm. As an important node related with the usage of code systems, HL7 and consequently HL7 CDA mandate that whenever a coded term or instance identifier appears in a message or document, the related code system should be provided with its unique Object Identifier (OID). These unique identifiers have a format like “2.16.840.1.113883.3.129.1.2.1” and they have to be registered to HL7. The OIDs defined in NHIS messages are already registered to HL7.

Continuing with our algorithm, we retrieve our local record for the input CDA node from the database. If its English translation is not provided and it does not already have universal variants, again we stop the algorithm since we can not query the UMLS SKS without the search string. Else, we query the UMLS SKS. If the response is empty, we just do the direct English translation on the input node. If it is not empty, we check whether coded terms from our preferred code systems exist in the response. In the positive case, one of them is selected according to their priorities and the input node is updated with the selected universal coded term.

4.2.4 Implementation Details

Most of the implementation efforts have been explained in the previous three sections in fact, so this section will be a consolidation upon those sections. As in the case of NHIS2CDA Transformer, the Universalizer is also designed as a service rather than an attractive GUI since it is planned to be deployed at the Ministry of Health servers and made available for public use through Web Services that will be acting as a proxy between the Ministry of Health and the cross-border countries capable of healthcare data exchange.

The implementation took place on the Java 6 platform again. Axis library is used for accessing UMLS SKS Web Services. Saxon is preferred as the XPath evaluator. The implementation accepts CDA conformant transformations of the NHIS Transmission Schema instances either as a string or file (it can also work on original Transmission Schema instances but this does not make sense for interoperability) and returns the document with its local coded terms universalized, as much as possible (to the degree that UMLS provides translations in fact).

The main components of the implementation are UMLSCClient and CodedValueTranslator which makes use of the XPathEvaluator. UMLSCClient is configurable based on the UMLS version, language and timeout of the Web Service call. In order to prevent any structural errors on the document that may appear during the translation, the output is validated

against the CDA schema as the final step.

Although the main purpose was to develop the Universalizer as a service, a simple GUI as a part of the Transformation Environment for HL7 Messages of NHIS, Turkey has been developed.

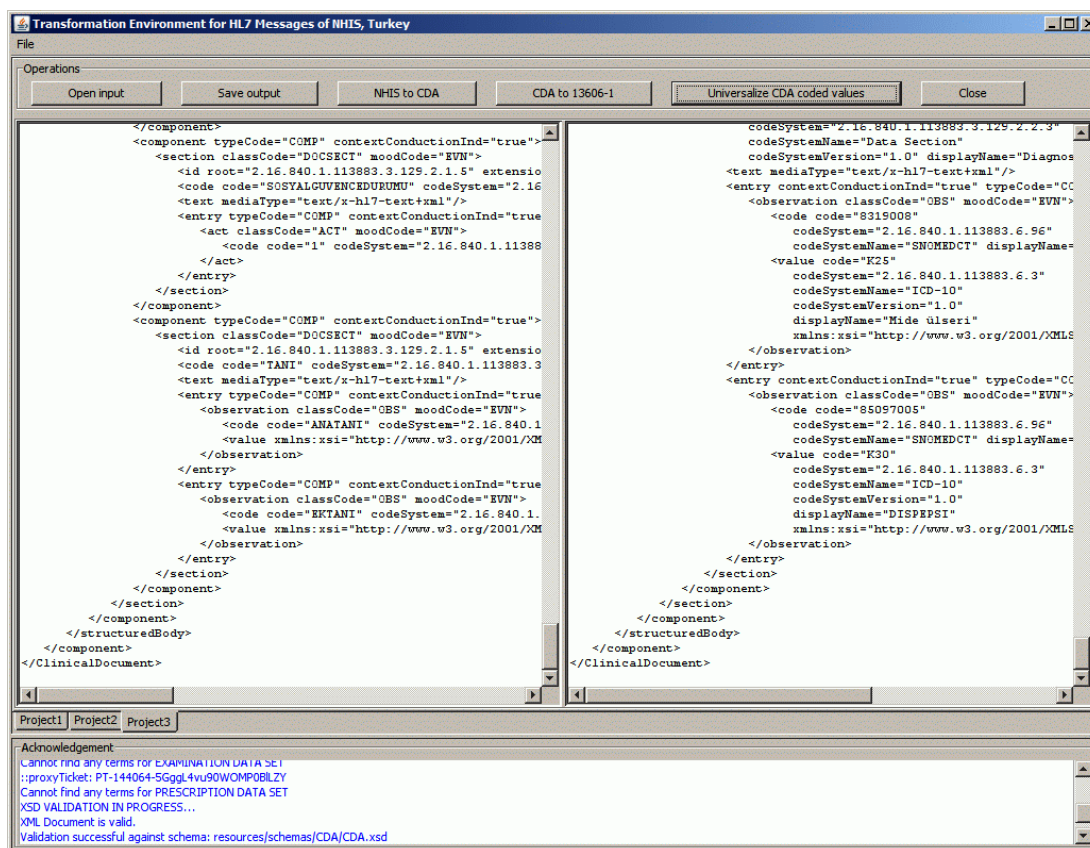


Figure 4.6: CDA CodedValue Universalizer in Action

Figure 4.6 shows the CDA CodedValue Universalizer in action, as a built-in module of the Transformation Environment. As usual, it is possible to create as many projects as needed. The left-hand side is the input text area. When the “Universalize CDA coded values” button is pressed, the Universalizer reads the input, locates all the coded values in it with the help of XPathEvaluator, and for each of them it calls the translateNode algorithm that is explained in the previous section. When all the nodes are completed, it calls the XSD Validator on the output and finally prints the result on the output text area. All these steps are acknowledged to the user through the Acknowledgement panel at the bottom.

Figure 4.4 presented some locally defined coded values. Now, the corresponding trans-

lations for some of these coded values are provided in Figure 4.7. All kinds of translation possibilities are available in the figure. Sometimes complete universalization is possible as in the case of `confidentialityCode` or `administrativeGenderCode`. But sometimes just an English translation is done as in the case of “Test result data set” since a universal code cannot be found for this string. The complete output of the Universalizer for the Examination example is accessible in our web page [56] but it cannot be provided in the Appendix due to space limitations.

We are very pleased to benefit from the UMLS^{SKS} Web Services. They have perfect documentation and high availability in terms of uptime. We have been testing the system for about a month and just for one day UMLS Release 2008AA was inaccessible while the other releases were still accessible. “Find Lexical Unique Identifier” operation is also able to restrict the searching to be done only on the requested code systems. So, if a cross-border country only accepts SNOMEDCT for instance, this can easily be achieved.

Independent of the Web Services, we have made some observations about the Knowledge Source Server of UMLS. For some of its concepts, it behaves very intelligently. For instance, when we make a query with “contagious disease”, it looks for “communicable disease” as well. But when “baby” is queried, “infant” cannot be deduced. As another example, SNOMEDCT code “8319008” is obtained for search strings “main diagnosis” and “principal diagnosis”. But search for “primary diagnosis” returns null while “secondary diagnosis” returns four coded terms. We believe that these mappings will be enhanced with wide usage and contribution from the field.

```

<code code="MUAYENE-YENIDOGAN" codeSystem="2.16.840.1.113883.3.129.2.2.1"
  codeSystemName="Döküman Tipi" displayName="Muayene MSVS (Yenidoğan)">

with

<code code="302199004" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMEDCT"
  displayName="Examination"/>

<administrativeGenderCode code="1" codeSystem="2.16.840.1.113883.3.129.1.2.21"
  codeSystemName="Cinsiyet" displayName="Erkek">

with

<administrativeGenderCode code="M" codeSystem="2.16.840.1.113883.5.1"
  codeSystemName="AdministrativeGender" displayName="Male"/>

<code code="TETKIKSONUCU" codeSystem="2.16.840.1.113883.3.129.2.2.2" codeSystemName="Veriseti"
  displayName="Tetkik Sonucu Veriseti">

with

<code code="TETKIKSONUCU" codeSystem="2.16.840.1.113883.3.129.2.2.2" codeSystemName="Data Set"
  codeSystemVersion="1.0" displayName="Test result data set"/>

<code code="ANATANI" codeSystem="2.16.840.1.113883.3.129.2.2.6" codeSystemName="Tanı Tipi"
  displayName="Ana Tanı">

with

<code code="8319008" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMEDCT"
  displayName="Main diagnosis"/>

<code code="ILAC" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
  displayName="İlaç Verisinin Olduğu Bölüm">

with

<code code="10160-0" codeSystem="2.16.840.1.113883.6.1" codeSystemName="LOINC"
  displayName="Medications"/>

```

Figure 4.7: Some Example Mapped Coded Values together with the Corresponding Local Coded Values

CHAPTER 5

MAPPING HL7 CDA R2 TO CEN EN 13606

In the previous chapters, we have explained how national/regional Electronic Healthcare Records can be transformed into international standards for enabling structure level interoperability as the initial step, and then developed a methodology based on UMLS Terminology Server for automatically generalizing the locally defined coded values within these EHRs for enhancing the content level interoperability. In both cases, the National Health Information System of Turkey is the subject of our approaches and conformant HL7 CDA R2 documents are produced as the result.

Although not popular as CDA, CEN EN 13606 Electronic Health Record Communication (EHRcom) is the European Standard for EHRs. CEN/TC 251 [77] is the technical committee on Health Informatics of the European Committee for Standardization (CEN) [10]. EHRcom results from the revision of the European pre-standard approved in 1999 and published in 2000. Its goal is to define a rigorous and stable information architecture for communicating part or all of the EHR of a single subject of care [11]. Detailed information about EHRcom is presented in Section 2.2 under Enabling Technologies and Standards chapter.

CEN EN 13606 consists of five parts. Parts 1 to 4 are published standards and Part 5 is at the final stages of the balloting phase:

1. *Reference Model* specifies the information architecture of the EHR data exchanged between systems and services.
2. *Archetypes Interchange Specification* specifies the Archetype Model and the language which are used to constrain the data.
3. *Reference Archetypes and Term Lists* presents some reference archetypes and specifies

the code lists which are used with the standard.

4. *Security* specifies the privileges and regulations necessary to access data.
5. *Interface Specification* defines a set of interfaces by which the artifacts defined in 13606 Parts 1, 2 and 4 may be requested and provided.

In order to enrich the interoperability support of our work, we have developed another XSLT based Adaptor for automatically creating CEN EN 13606 conformant EHR instances from HL7 CDA R2 conformant EHRs. Again, NHIS of Turkey is chosen as the subject our work. As a result, one of the first implementations of the CEN EN 13606-1 has been realized. More critically, our literature survey shows that Turkey is the first country to support cross-border clinical data exchange with more than one international standard at the same time, that is both as HL7 v3 CDA and CEN EN 13606.

In this chapter, first we discuss our efforts for developing an XML Schema Definition for CEN EN 13606-1. Unlike HL7, CEN does not provide any computer processable structure definition for this standard. After we have a schema to conform, we explain the mappings between CDA and 13606-1 elements. Some example XSLT rules together with sample inputs and outputs are provided as well. Then we discuss the implementation details and finally provide our observations gained through this challenging process, which we believe are very valuable since this is one of the first implementations of the recently approved European standard.

5.1 Developing the XML Schema Definition for CEN EN 13606 Reference Model

Unlike HL7, CEN/TC251 does not provide an Implementation Technology Specification or schemata for its Reference Model and thus the EHR documents. EN 13606-1 just models the Reference Model as a UML diagram and it is clearly stated in the normative content of the standard specification that a system for communication of EHR information is conformant with this standard if all information that is exchanged which is within the scope of this standard can be expressed in a form where there is a direct correspondence between the communicated data structure and the information model of an EHR_EXTRACT defined using UML [11]. In one sentence, for conformance all you need to do is be compatible with the UML class diagram of the reference model. This is a very problematic disadvantage

of the standard but further discussion of this issue will be presented in the Observations section. The UML class diagram of the reference model is given in Figure 5.1 [11].

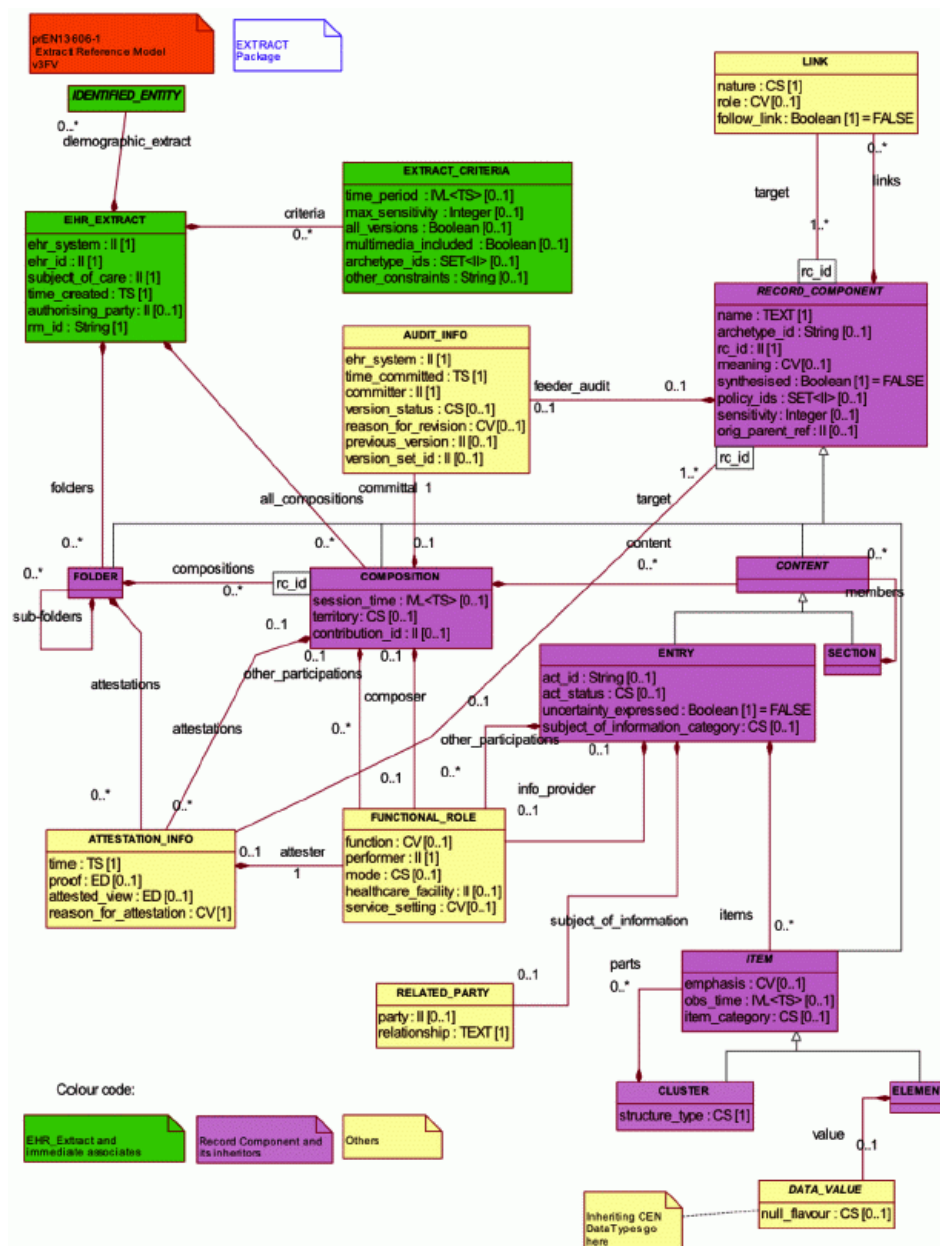


Figure 5.1: The UML Class Diagram of CEN EN 13606-1 Reference Model

We have realized that we have to develop the XML Schema Definitions (XSD) of the reference model by ourselves. But before starting, we have made a literature survey for looking whether anybody initiated such a study beforehand. We discovered that a project named LinkEHR that intends to develop a tool which allows transformation and standard-

ization of clinical data using a dual model approach for the EHR architecture; in particular the European Standard CEN/TC251 EN 13606 and OpenEHR initiated XSDs for EN 13606 reference model [46].

Thus, instead of starting from scratch, we have built our work on top of their XSDs. However, LinkEHR XSDs are not a complete coverage of the EN 13606 reference model. They implement quite a recent version of the reference model but many classes are missing in the definitions and for the existing classes some inconsistencies are present. For instance, the complete DEMOGRAPHICS package that contains 12 classes is missing. As an inconsistency example, the associations of the FOLDER class are modeled with cardinality one; however they should be unbounded according to the reference model. There are also some wrong definitions in the data types schema. All these missing classes and inconsistent definitions have been identified and the EN 13606-1 XSD has been developed as the necessary initial step of our development. Still, our XSDs are not fully compatible with the reference model by intention since there are many inefficient modeling issues in the original reference model. In order to overcome these issues, some mandatory attributes have been made optional. These issues will be discussed in the Observations section.

Due to space limitations the developed XSDs cannot be presented in the Appendix but they are accessible in our Web page [27]. High level graphical view of the EN 13606 reference model that has the EHR_EXTRACT as the root class by default is presented in Figure 5.2.

5.2 Mapping Process

In order to generate CEN EN 13606-1 conformant instances from HL7 CDA R2 instances, first we have to define a mapping between the building blocks of CDA and 13606-1.

The main hierarchy components (classes) of the EHR Extract Reference Model are given in Table 5.1 [11] together with some examples.

The EHR_EXTRACT is used to represent part or all of the health record information of a single subject of care extracted from an EHR provider system for the purposes of communication. HL7 CDA does not have such a concept, like dumping all the records of an EHR system. FOLDER enables high-level logical grouping and organization of the COMPOSITIONs. COMPOSITION corresponds to a single clinical session of record interaction by definition, and Kalra and Lloyd [20] state that it corresponds to an HL7 CDA document. However, we think that it is not possible to make such a strict restriction without knowing the application of CDA in a specific setting, as in the case of NHIS, Turkey. Table 5.1 gives

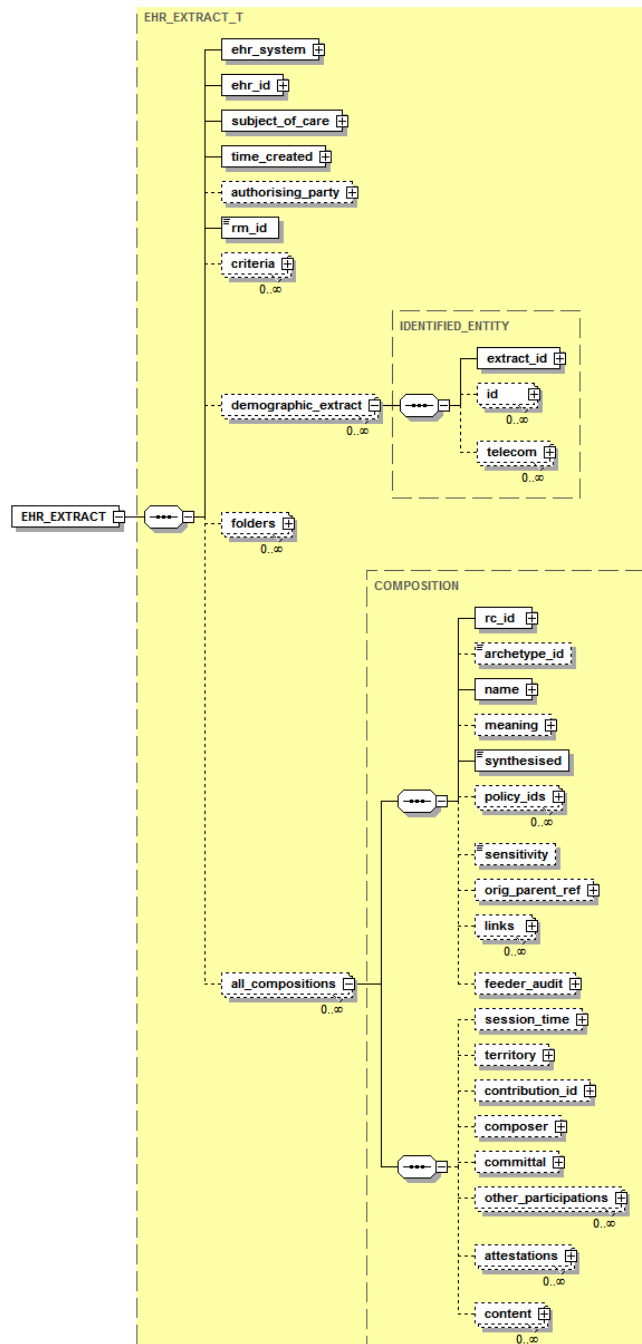


Figure 5.2: High Level View of the EN 13606 Reference Model Schema

examples for the reference model components. Examples to COMPOSITION are progress note, laboratory test result, referral letter, discharge summary, diabetes review, etc. all of which correspond to Minimum Health Data Sets of the National Health Data Dictionary and are modeled as the first-level sections of the CDA, rather than the CDA itself. Both for this reason and for better organizing the EHR instance, we map the ClinicalDocument to FOLDER and first-level CDA sections to COMPOSITION.

Table 5.1: Main hierarchy components of the EHR Extract Reference Model

EHR HIERARCHY COMPONENT	DESCRIPTION	EXAMPLES
EHR_EXTRACT	The top-level container of part or all of the EHR of a single subject of care, for communication between an EHR Provider system and an EHR Recipient.	(Not applicable)
FOLDER	The high level organisation within an EHR, dividing it into compartments relating to care provided for a single condition, by a clinical team or institution, or over a fixed time period such as an episode of care.	Diabetes care, Schizophrenia, Cholecystectomy, Paediatrics, St Mungo's Hospital, GP Folder, Episodes 2000-2001, Italy.
COMPOSITION	The set of information committed to one EHR by one agent, as a result of a single clinical encounter or record documentation session.	Progress note, Laboratory test result form, Radiology report, Referral letter, Clinic visit, Clinic letter, Discharge summary, Functional health assessment, Diabetes review.
SECTION	EHR data within a COMPOSITION that belongs under one clinical heading, usually reflecting the flow of information gathering during a clinical encounter, or structured for the benefit of future human readership.	Reason for encounter, Past history, Family History, Allergy information, Subjective symptoms, Objective findings, Analysis, Plan, Treatment, Diet, Posture, Abdominal examination, Retinal examination.
ENTRY	The information recorded in an EHR as a result of one clinical action, one observation, one clinical interpretation, or an intention. This is also known as a clinical statement.	A symptom, an observation, one test result, a prescribed drug, an allergy reaction, a diagnosis, a differential diagnosis, a differential white cell count, blood pressure measurement.
CLUSTER	The means of organising nested multi-part data structures such as time series, and to represent the columns of a table.	Audiogram results, electro-encephalogram interpretation, weighted differential diagnoses.
ELEMENT	The leaf node of the EHR hierarchy, containing a single data value.	Systolic blood pressure, heart rate, drug name, symptom, body weight.

SECTION represents the clinical data within a COMPOSITION such as abdominal examination, reason for encounter, etc. SECTION corresponds to the data element wrapper sections of the NHIS Transmission Schemas which are modeled as second-level sections in the CDA. ENTRY is known as the clinical statement class of reference model. Examples include observations, evaluations, and a prescribed drug. ENTRIES directly map to the CDA Entry classes (act, observation, substanceAdministration, etc.) which are also known as clinical statements. Part 3 - Reference Archetypes and Term Lists [12] of EHRcom provides some tables mapping the HL7 entry classes (Act, Observation, Procedure, SubstanceAdministration, Supply and Encounter) to CEN 13606 classes. On the CEN 13606 side, HL7 entry classes are mapped to a combination of ENRTY, CLUSTER and ELEMENT classes. However, these mappings are not a complete coverage of the CDA entry classes; Consumable and Organizer are missing. In our implementation, during the transformation of CDA entry classes to CEN 13606 we stuck to those mapping definitions in 13606-3 as much as possible and we made extensions for the missing parts.

ELEMENT is the leaf node of the EHR hierarchy containing a single value. ELEMENTs can optionally be grouped within a CLUSTER to form multi-part data structures. These two classes correspond to attributes and elements of the CDA Entry classes most of the time.

The relationships between the artifacts of NHDD, the Transmission Schemas and HL7 CDA R2 were provided beforehand. Now, in Figure 5.3 the mapping between Transmission Schemas, HL7 CDA R2 and CEN EN 13606-1 are presented.

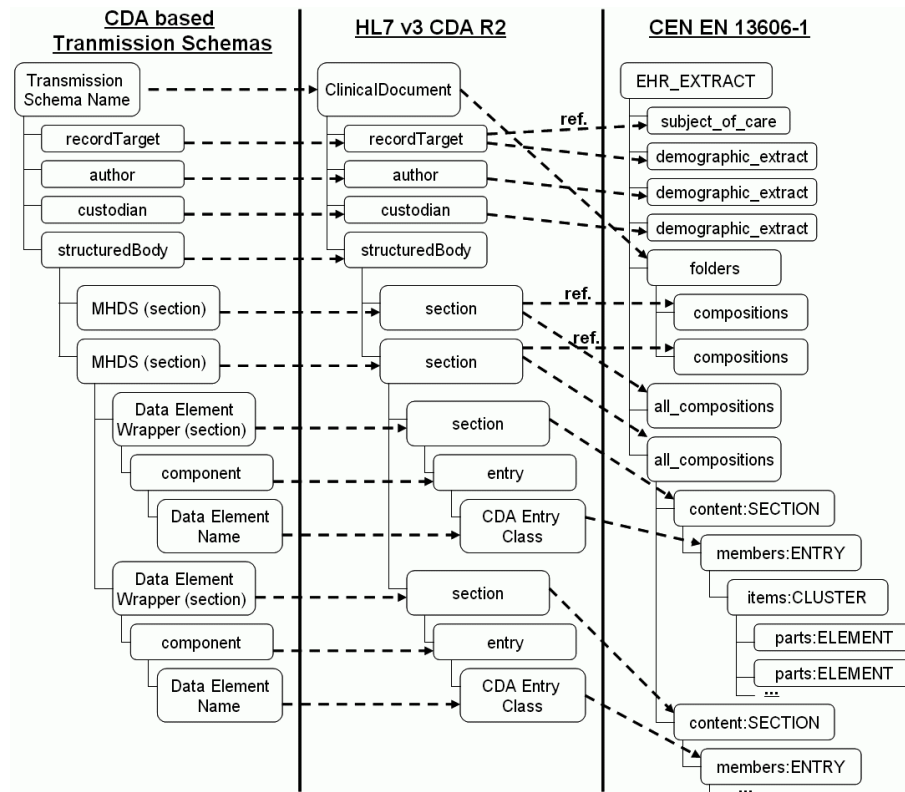


Figure 5.3: Mapping CDA (generated from Transmission Schemas) to EN 13606-1

Once this mapping is defined, we have developed the necessary XSLT rules for automatic transformation of CDA instances to valid EHRcom instances. Some example transformations and related XSLT rules are presented in the following sub-section.

5.2.1 Example Mappings and XSLT Rules

As the first example, we transform the recordTarget of a CDA instance to “demographic_extract” of EN 13606 reference model. “demographic_extract” element is used to store information about any participant of the EHR, including the patient, author or healthcare organization.

Since recordTarget holds information about the subject of care in CDA, we specify the type of “demographic_extract” to “SUBJECT_OF_CARE_PERSON_IDENTIFICATION”.

An example input recordTarget is given in Figure 5.4 EN 13606 reference model conformant “demographic_extract” instance for this recordTarget should look like the XML block presented in Figure 5.5.

```
<recordTarget contextControlCode="OP" typeCode="RCT">
  <patientRole classCode="PAT">
    <id extension="12345678905" root="2.16.840.1.113883.3.129.1.1.1"/>
    <patient classCode="PSN" determinerCode="INSTANCE">
      <name>
        <family>Yüksel</family>
        <given>Mustafa</given>
      </name>
      <administrativeGenderCode code="M" codeSystem="2.16.840.1.113883.5.1"
codeSystemName="AdministrativeGender" displayName="Male"/>
      <birthTime value="19840101"/>
      <raceCode code="TR" codeSystem="2.16.840.1.113883.3.129.1.2.52" codeSystemName="Uyruk"
codeSystemVersion="1.0" displayName="Türkiye"/>
    </patient>
  </patientRole>
</recordTarget>
```

Figure 5.4: An Example recordTarget from CDA

As it is seen in the figures, since the base reference information models of HL7 CDA and CEN EN 13606 are substantially divergent, the XML blocks for representing the same information are very different. This fact increases the transformation efforts as well. The XSLT rules for transforming Transmission Schemas to CDA were not very complex but this time they are rather complicated and long. The XSLT rule for transforming recordTarget of CDA into demographic_extract is given in Figure 5.6 and Figure 5.7.

As the second example, we transform a CDA entry class, namely observation, to EN 13606 by using the mapping definitions from CEN 13606-3 - Reference Archetypes and Term Lists. An example observation instance is presented in Figure 5.8. This is a very compact example; it has four elements two of which are simple attributes. “classCode” declares that this is an Observation class. “moodCode” value “EVN” states that this observation is an event, already completed. Then the instance has two elements, “code” for annotating the meaning of the observation, which is “Main diagnosis” from SNOMEDCT and “value” for

```

<demographic_extract xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:type="SUBJECT_OF_CARE_PERSON_IDENTIFICATION">
  <extract_id>
    <extension>12345678905</extension>
    <assigningAuthorityName>NVI, Turkey</assigningAuthorityName>
    <validTime/>
    <root>
      <oid>2.16.840.1.113883.3.129.1.1.1</oid>
    </root>
  </extract_id>
  <name>
    <use>
      <codeValue>Patient Name</codeValue>
    </use>
    <validTime/>
    <namePart>
      <entityPartName>Yüksel</entityPartName>
      <namePartType>
        <codeValue>FAM</codeValue>
      </namePartType>
      <namePartQualifier>
        <codeValue>CL</codeValue>
      </namePartQualifier>
    </namePart>
    <namePart>
      <!-- Mustafa similar to above namePart -->
    </namePart>
  </name>
  <administrativeGenderCode>
    <codingScheme>
      <oid>2.16.840.1.113883.5.1</oid>
    </codingScheme>
    <codingSchemeName>AdministrativeGender</codingSchemeName>
    <codingSchemeVersion/>
    <codeValue>M</codeValue>
  </administrativeGenderCode>
  <birthTime>
    <time>19840101</time>
  </birthTime>
</demographic_extract>

```

Figure 5.5: The Corresponding Conformant EN 13606-1 demographic_extract for record-Target

presenting this diagnosis as a coded value from ICD-10 meaning malaria.

CEN 13606-3 models this CDA entry class as an ENTRY in the 13606 reference model. This ENTRY involves one CLUSTER composed of ELEMENTs each of which are used for

```

<xsl:element name="demographic_extract">
  <xsl:attribute name="xsi:type">SUBJECT_OF_CARE_PERSON_IDENTIFICATION</xsl:attribute>
  <xsl:element name="extract_id">
    <xsl:element name="extension">
      <xsl:value-of select="hl7:recordTarget/hl7:patientRole/hl7:id/@extension"/>
    </xsl:element>
    <xsl:element name="assigningAuthorityName">NVI, Turkey</xsl:element>
    <xsl:element name="validTime"/>
  <xsl:element name="root">
    <xsl:element name="oid">
      <xsl:value-of select="hl7:recordTarget/hl7:patientRole/hl7:id/@root"/>
    </xsl:element>
  </xsl:element>
</xsl:element>
<xsl:for-each select="hl7:recordTarget/hl7:patientRole/hl7:telecom">
  <xsl:call-template name="transTelecom"/>
</xsl:for-each>
<xsl:if test="hl7:recordTarget//hl7:patient/hl7:name">
  <xsl:element name="name">
    <xsl:element name="use">
      <xsl:element name="codeValue">Patient Name</xsl:element>
    </xsl:element>
    <xsl:element name="validTime"/>
    <!-- family -->
    <xsl:element name="namePart">
      <xsl:element name="entityPartName">
        <xsl:value-of select="hl7:recordTarget//hl7:patient/hl7:name/hl7:family"/>
      </xsl:element>
      <xsl:element name="namePartType">
        <xsl:element name="codeValue">FAM</xsl:element>
      </xsl:element>
      <xsl:element name="namePartQualifier">
        <xsl:element name="codeValue">CL</xsl:element>
      </xsl:element>
    </xsl:element>
    <!-- given -->
    <xsl:element name="namePart">
      <xsl:element name="entityPartName">
        <xsl:value-of select="hl7:recordTarget//hl7:patient/hl7:name/hl7:given"/>
      </xsl:element>
      <!-- Similar to the above but "GIV" and "CL" are inserted -->
    </xsl:element>
  </xsl:element>
</xsl:if>

```

Figure 5.6: XSLT Rule for Transforming recordTarget to demographic_extract (Part 1)


```

<xsl:for-each select="hl7:recordTarget/hl7:patientRole/hl7:addr">
  <xsl:call-template name="transAddr"/>
</xsl:for-each>
<xsl:element name="administrativeGenderCode">
  <xsl:call-template name="CV2CS">
    <xsl:with-param name="inp1"
select="hl7:recordTarget//hl7:patient/hl7:administrativeGenderCode"/>
  </xsl:call-template>
</xsl:element>
<xsl:element name="birthTime">
  <xsl:element name="time">
    <xsl:value-of select="hl7:recordTarget//hl7:patient/hl7:birthTime/@value"/>
  </xsl:element>
</xsl:element>
</xsl:element>

```

Figure 5.7: XSLT Rule for Transforming recordTarget to demographic_extract (Part 2)

```

<observation classCode="OBS" moodCode="EVN">
  <code code="8319008" codeSystem="2.16.840.1.113883.6.96" codeSystemName="SNOMEDCT"
  displayName="Main diagnosis"/>
  <value code="B54" codeSystem="2.16.840.1.113883.6.3" codeSystemName="ICD-10"
  codeSystemVersion="1.0" displayName="Malaria" xsi:type="CV"/>
</observation>

```

Figure 5.8: An Example observation from CDA

modeling one property (attribute/element) of the CDA entry class. Our instance has four properties, 13606-3 skips the “classCode” attribute because they say that it has a fixed value which is completely wrong. In the CDA entry classes, both the “classCode” and “moodCode” attributes are mandatory since they do not have fixed values.

Continuing with the example, we have three properties to map to the ELEMENTs. A small part of the EN 13606 conformant counterpart of our observation instance is presented in Figure 5.9.

As it is truly obvious, the result is huge although we just display one ELEMENT instead of three. 4 lines of observation instance in the CDA is represented in EHRcom with 177 lines. Expressivity of EHRcom is too complex and it contains a lot of redundant mandatory elements even for the leaf nodes which should just present the values. It is impossible for an EHR system to have a globally unique identifier (rc_id in EHRcom) for leaf nodes for instance, but EHRcom expects this value for all classes since the abstract RECORD_COMPONENT class, which all the other classes are derived from, has it as mandatory element. Moreover,

```

<parts xsi:type="ELEMENT">
  <rc_id>
    <extension>NA</extension>
    <assigningAuthorityName>NHIS, Turkey</assigningAuthorityName>
    <validTime/>
    <root>
      <oid>NA</oid>
    </root>
  </rc_id>
  <name>
    <originalText>Act code</originalText>
  </name>
  <meaning>
    <codingScheme>
      <oid>987654321.1</oid>
    </codingScheme>
    <codingSchemeName>CEN/TC251/EN13606-3:RECORD_COMPONENT_MEANING</codingSchemeName>
    <codingSchemeVersion>1.0</codingSchemeVersion>
    <codeValue>HL7-04</codeValue>
    <displayName>Act_code</displayName>
  </meaning>
  <synthesised>>false</synthesised>
  <item_category>
    <codingScheme>
      <oid>987654321.3</oid>
    </codingScheme>
    <codingSchemeName>CEN/TC251/EN13606-3:ITEM_CATEGORY</codingSchemeName>
    <codingSchemeVersion>1.0</codingSchemeVersion>
    <codeValue>IC01</codeValue>
  </item_category>
  <value xsi:type="CV">
    <codingScheme>
      <oid>2.16.840.1.113883.6.96</oid>
    </codingScheme>
    <codingSchemeName>SNOMEDCT</codingSchemeName>
    <codingSchemeVersion/>
    <codeValue>8319008</codeValue>
    <displayName>Main diagnosis</displayName>
  </value>
</parts>

```

Figure 5.9: A Small part of the Corresponding Conformant EN 13606 ENTRY for observation

this id has to come from the EHR system itself. For this reason, we are not allowed to generate it. So, we have provided “NA” values for such elements. Such disadvantages of EHRcom are discussed in the Observations section.

We cannot present the XSLT rule for the above example in order to save space. The complete XSLT file and CEN 13606 conformant output of our Examination CDA are available in our Web page [58, 54].

5.3 Implementation Details

No additional component has been implemented for the purpose of CDA to CEN 13606 transformations; all the components were already available from the NHIS to CDA transformation module. Just the XSLT rules have been developed.

Similarly with the work done in the previous chapters, this module is also designed as service rather than a GUI. The service accepts conformant CDA instances as a file or string and returns the corresponding CEN 13606 conformant transformation. This service will also be deployed as a Web Service to the Ministry of Health servers for enabling cross-border clinical data exchange with the European standard. As a result of our work, Turkey is the first country to support EHR exchange with more than one international standard at the same time.

A graphical user interface is also developed as a part of the Transformation Environment for HL7 Messages of NHIS, Turkey interface. A screenshot of the interface is presented in Figure 5.10. The “CDA to 13606-1” button initiates the transformation process. The output is subject to XSD validation as in the previous cases. The user is acknowledged through the panel at the bottom.

5.4 Observations

We believe that we have gained valuable experience by implementing the recently published European Standard CEN/ISO 13606 (part 5 is still not published by the way). As one of the first implementers of the standard, we faced with many challenges. In this section, a list of these challenges and our opinions for some aspects of the standard are presented:

- As the major disadvantage, CEN 13606 does not provide a computer processable schema for its reference model. For being implementation technology independent, the UML class diagrams of the reference model are provided, which is perfectly fine. But, in order to enable uniqueness among the implementations of the standard, it should definitely be supported with Implementation Technology Specifications as in the case of HL7. XML is the dominant standard today, so XML ITS will suffice for

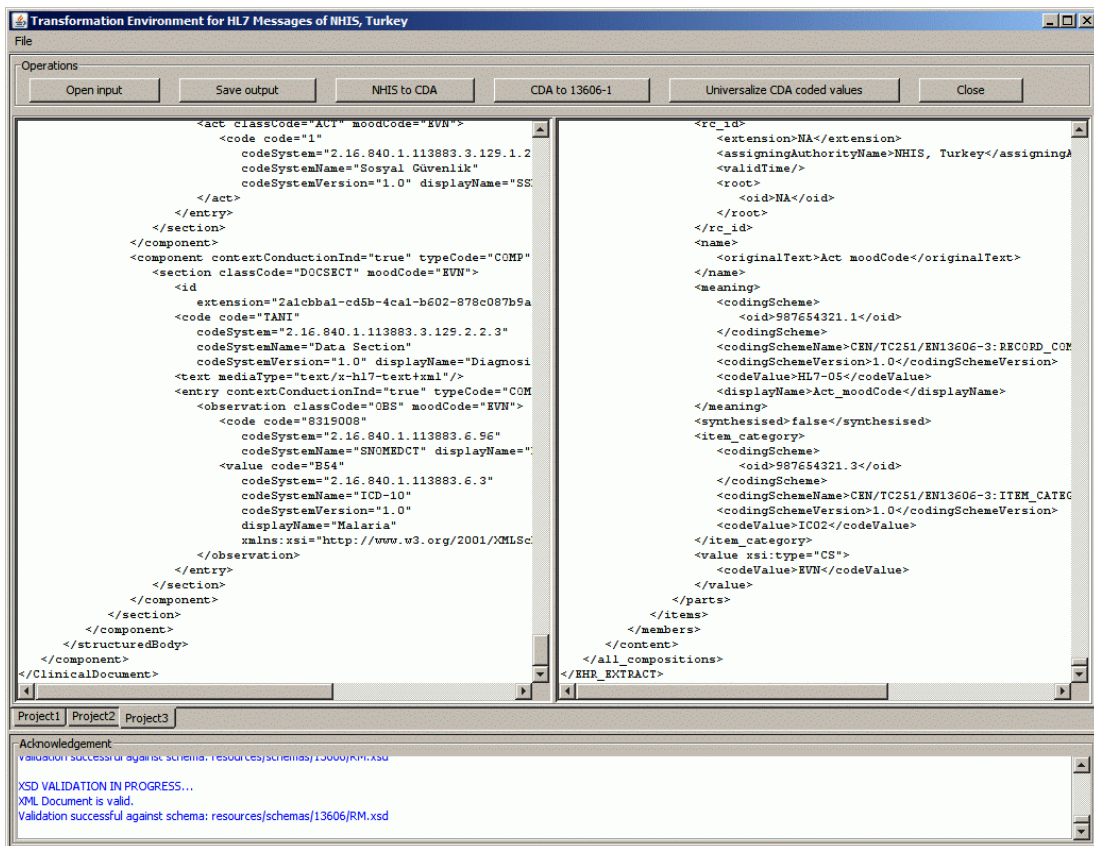


Figure 5.10: CDA to CEN 13606 Transformer in Action

now. Hopefully the non-complete part of 13606, namely Part 5: Interface specification, fills in this lack in the near future.

- In relationship with the above item, the conformance of CEN 13606 cannot be tested since there is not a common schema for it. As a direct result, it is not possible for two different implementers of the standard to interoperate in a plug'n play manner. In the normative sections of the standard, it is clearly stated that an EHR system is conformant with the standard as long as there is a direct correspondence between its communicated data structure and the information model of the EHR_EXTRACT defined using UML. European-level interoperability is expected from this standard; with the current status this is not achievable.
- The expressivity of the reference model is too complicated and there are many redundant but mandatory elements. As an example, 25 KBs CDA document is transformed into 174 KBs EHRcom document. The situation is much worse when the number of lines is compared. A single "moodCode" attribute of a CDA entry class can be

represented with an ELEMENT instance of 34 lines at minimum. This affects the readability of the EHR document as well.

- Although there are more redundant mandatory attributes, the most problematic one is the “rc_id” which is used to present the globally unique identifier of the classes that are derived from the abstract RECORD_COMPONENT class. Almost all of the classes are derived from RECORD_COMPONENT. It is completely normal to request unique identifiers for high level components such as FOLDER, COMPOSITION or SECTION. But it is nonsense to request these from leaf-level components that are used for presenting values. For instance, a “moodCode” or a coded term from ICD-10 cannot have an rc_id.
- The same problem is observed in some of the data types as well. For instance, the “TEXT” data type mandates the use of “charset” and “language” attributes for each “TEXT” instance. This is incredible; an EHR document may involve hundreds of TEXT elements and for each of them character set and language information have to be provided. Most probably, this will multiply the size of the document by three or four. Instead, the character set and language information can be notified at the root or section levels. Similarly, instance identifier (II) data type always requires “validTime” attribute. This should be optional so that it is used only if there is a limitation of the validity time of an instance identifier. The rest of the data types are not very different, almost all of their attributes are mandatory.
- Related with the expressivity and readability issues, it is obvious that CEN 13606 needs some more classes designed for specific purposes; the most required ones being entry classes as in the case of HL7. The ENTRY - CLUSTER - ELEMENT combinations can handle any kind of data (especially when they are modeled with archetypes) but they grow to enormous sizes and readability disappears. There is an ongoing work for modeling EHRcom with the HL7 Reference Information Model [14]. As the final progress, D-MIM and R-MIM of EHR_EXTRACT reference model was developed. If this study is concluded with a well-defined schema for 13606, then most of the challenges explained in this section will melt away.
- It is a must to use ISO Object Identifiers (OID) for identifying code systems. However, it is impossible to find the OID values for the code systems that are defined by CEN/TC251 itself. In our implementation, we had to make up these values. Further-

more, there is not a standardization of the code system names as well.

CHAPTER 6

RELATED WORK

Providing the interoperability of heterogeneous healthcare information systems has always been an active research area.

There are several projects granted for proposing solutions to the eHealth interoperability problem. One example is the RIDE project (A Roadmap for Interoperability of eHealth Systems in Support of COM 356 with Special Emphasis on Semantic Interoperability) [66] that is supported by the European Commission. In the second version of its roadmap [69], RIDE has proposed an architecture entitled the European Healthcare Network which aims to realize integration of different medical systems in a configurable network of interconnected organizations so that the proposed European Healthcare Network is flexible by design, allowing smooth involvement of the legacy systems within the network. Our work is in collaboration with the “Adaptor” model of RIDE European Healthcare Network.

In addition to supporting such interoperability projects, the European Commission publishes recommendations for cross-border eHealth interoperability [63, 64]. The renewed deadline for establishing EU-wide EHR exchange is 2015. The abovementioned RIDE European Healthcare Network is also recognized by the eHealth Interoperability Recommendation of the European Commission.

As an example more concrete solution, [5] describes how to mediate between HL7 Version 2 and HL7 Version 3 messages based on semantics. [60] describes how two different EHR standards (namely HL7 CDA and CEN EN 13606 as in our case) derived from the same Reference Information Model can be mapped to each other by using archetypes, Refined Message Information Model derivations and semantic tools. In our work, first of all, the EHR standards are not derived from the same RIM, they are kept as original. Secondly, this work addresses the mapping of only clinical statements while our work is for mapping the entire reference model of the standards.

There is an ongoing standardization effort for the harmonization of data types for information exchange as a result of valuable combination of requirements from HL7 Version 3, ISO 11404, CEN 13606 and past ISO work on healthcare data types and joint efforts of ISO/TC 215, HL7 and Connecting for Health [25]. Achieving uniqueness at the data type level is very important for enabling interoperability between different healthcare standards. With this effort, for instance it will be possible to map HL7 CDA instances to CEN EN 13606 instances more easily. A detailed survey and analysis of HL7 CDA, CEN 13606 and many more EHR standards is done in [28].

There is an ongoing work jointly by HL7 and CEN experts for modeling EHRcom with the HL7 Reference Information Model [14]. As the final progress, D-MIM and R-MIM of EHR_EXTRACT reference model was developed. If this study is concluded with solid results, achieving interoperability of CEN EHRcom and HL7 CDA will be an easy problem. But unfortunately, no recent progress is observed.

LinkEHR project [46] supported by the Spanish Ministry of Education and Science intends to develop a tool which allows transformation and standardization of clinical data using a dual model approach for the EHR architecture; in particular the European Standard CEN/TC251 EN 13606 and OpenEHR [61].

The most similar effort to a part of our work was announced in Belgium for converting instances of the locally developed “Kind Messaging for the Electronic Healthcare Records” (Kmehr) [44] standard to HL7 CDA using XSLT [45]. However, the information consists of just a single sentence and no recent development is observed.

An important aspect of semantic interoperability in the healthcare domain is the terminologies. Unified Medical Language System (UMLS) [81] explained in the Enabling Technologies and Standards section is a widely used Terminology Server which is also a critical component of our work. Similar work continues with the joint efforts of HL7 and ISO for defining and standardizing Common Terminology Services (CTS) [17] that defines an Application Programming Interface (API) that can be used when accessing and exchanging terminological content. The coverage and description of the API is satisfactory. Moreover, Web Service Description Language (WSDL) bindings of the APIs are provided as normative content. The importance of CTS will become more obvious with the first examples of truly automatic cross-border clinical data exchange, hopefully within the next few years. However, there are still some points that CTS has to improve itself. Currently CTS Mapping API provides an interface for mapping from local concepts to the standardized codes or between different standardized code sets but the mapping model is quite limited for now. In any case,

CTS defines just the interfaces. On the other hand, what we achieved based on UMLS is a real implementation that can also be updated to implement CTS interfaces.

CHAPTER 7

CONCLUSION AND FUTURE WORK

Today, the application of information and communication technologies to health domain is on the agenda of many countries and most of them have already accomplished some developments for establishing a complete national/regional electronic healthcare infrastructure. The main aim of these efforts is to make Electronic Healthcare Records of a patient accessible anywhere at any time to all authorized users. This is even valid in the cross-border case. After experiencing the benefits of local healthcare infrastructures, such as effective and efficient patient care by facilitating the retrieval and processing of clinical information about a patient from different sites, initiatives encouraging higher-level clinical data exchange boosted their activities. A closer example is the European Commission publishing eHealth interoperability recommendations to the EU Member States and looking forward to realizing freely traveling European citizen with her distributed EHRs scenario. Another example is the USA trying to share EHRs among the Regional Health Information Organizations through National Health Information Network (NHIN).

Interoperable cross-border clinical data exchange is an ambitious goal and naturally it has some challenges. There are plenty of standards and specifications in the healthcare informatics domain for different needs at different levels, i.e. messaging standards, EHR standards, terminology coding systems and so on. This variety increases when the locally developed standards and coding systems by the nations or states appear on the scene. We can confidently say that each country has its own EHR standard and in the case of a multi-party EHR exchange, it is not logical to realize all possible combinations of mappings among these standards. What needs to be done is obvious as already reported to the European Commission with RIDE Roadmaps II and III [69] Each participant in the network shall continue to maintain its local infrastructure and standards as they are. Only in the case of cross-border data exchange, each participant is responsible for transforming its local EHR

instances to one (or more) commonly agreed format which should be chosen among the widely accepted international standards, such as HL7 CDA and CEN EHRcom. This way, the legacy systems of the participants will continue working as they used to do. On top of these legacy systems, there is a need for the development of “Adaptors” that will facilitate the transformation process.

Especially in a multilingual network, as in the case of Europe, it is definite that the mapping of the local EHR content to the commonly agreed format, which we call structure level conformance, will not suffice. It is undeniable that locally developed coded terms exist in the local EHRs and in order to achieve true semantic interoperability, those terms shall be translated to a code system or language that the other participants in the network can understand. This necessitates a Terminology Server based automatic translation environment for the coded terms.

In accordance with the above mentioned methodology for cross-border interoperable EHR exchange, in this thesis we have made significant contribution. We were lucky enough to apply all of our approaches to the National Health Information System of Turkey. Our achievements can be summarized as follows:

- Turkey adapted HL7 CDA R2 as the EHR standard but unfortunately, in order to meet all national requirements directly at the schema level, many changes have been done on the original CDA Schema that broke the conformity of local “Transmission Schemas” to CDA. In terms of our methodology, here the “Transmission Schemas” are the locally developed EHRs and CDA is the commonly agreed format that we need to conform. For this purpose, we have developed an XSL Transformations (XSLT) based Adaptor to automatically generate CDA conformant instances from the “Transmission Schema” instances.

The Adaptor will be deployed as a Web Service to the Ministry of Health servers that will be acting as a proxy between the Ministry of Health and the cross-border countries capable of healthcare data exchange. Moreover, MoH is planning to archive the EHRs received from the healthcare organizations in native CDA format. For this purpose, the transformer will be deployed as a front-end to the centralized database.

Another use of the Adaptor may be on the NHIS clients’ side; the clients can store the generated Transmission Schemas in native CDA format so that when a change has been done on the Web Service schemas, the previous records will still be accessible. It is expected that annual updates/changes will be done on the National Health Data

Dictionary which will affect the Transmission Schemas directly.

- Structure level interoperability achieved in the previous bullet enables the processability of the EHR instance by anyone that conforms to the related standard. However, there are still things to be done at the semantic level of the EHR documents in order to converge to complete interoperability among the actors with different settings. For this purpose, Unified Medical Language Systems (UMLS) based modular architecture enabling automatic universalization of the local coded terms that exist in the EHR documents has been developed. UMLS Knowledge Source Server (KSS) Web Services are used for searching for international counterparts to the local coded terms that exist in the EHR documents.

There are very specific and detailed coding systems in Turkey that are shared through Health Coding Reference Server. HCRS is really a challenging test suite for our universalizer and UMLS Terminology Server. We have listed the strong and weak points of the UMLS based on our experience. Furthermore, our universalizer is suitable with the Common Terminology Server (CTS) approach of HL7.

Again, this module will be deployed as a Web Service to the Ministry of Health servers for public use.

- Although not popular as CDA, CEN EN 13606 Electronic Health Record Communication (EHRcom) is the European Standard for EHRs. In order to enrich the interoperability support of our work, we have developed another XSLT based Adaptor for automatically creating CEN EN 13606 conformant EHR instances from HL7 CDA R2 conformant EHRs. As a result, one of the first implementations of the CEN EN 13606-1 has been realized.

This Adaptor will be made public through a Web Service deployed at the Ministry of Health, as well.

- Our work proved that when a local implementation is already based on international standards as much as possible, the mapping efforts decrease enormously. Thus, development of a proprietary format instead of adapting a standard decreases interoperability. Mapping but CDA based (not conformant) Transmission Schemas to the original CDA was way easier than mapping them to CEN EHRcom.
- Our literature survey shows that, with the completion of our work Turkey is the first country to support cross-border EHR exchange with more than one international stan-

dard (HL7 CDA and CEN EHRcom) at the same time while majority of the other countries are unable to support one international standard.

- We have gained valuable experience by implementing the recently published European Standard CEN/ISO 13606. As one of the first implementers of the standard, we faced with many challenges. We presented the weak points of CEN 13606 as a very detailed list which includes items regarding the reference model, data types, expressivity issues, lack of schemata, and so on. We believe that CEN 13606 will continue to improve with the feedbacks coming from real implementations in the industry, as in our case.

As the future work, we plan to implement the Mapping interface of the HL7 Common Terminology Services. Moreover, although we tried our best to provide a generic mapping, the current transformation of CDA to EHRcom is rather NHIS, Turkey centric since we had instances from NHIS. In the future this can be overcome when we have a collection of CDA instances from different settings.

REFERENCES

- [1] The Family Medicine Information System (FMIS). http://www.sagliknet.saglik.gov.tr/portal_pages/notlogin/vatandas/vatandas_ailehekimiprototipi.htm, last visited on August 2008.
- [2] American National Standards Institute (ANSI). <http://www.ansi.org/>, last visited on August 2008.
- [3] Apache Axis Web Service Library for Java. <http://ws.apache.org/axis/>, last visited on August 2008.
- [4] S. K. Bechhofer, C. A. Goble, A. L. Rector, and W. D. Solomon. Terminologies and terminology servers for information environments. In *Proceedings of the 8th International Workshop on Software Technology and Engineering Practice*, pages 484–497, 1997.
- [5] V. Bicer, G. Laleci, A. Dogac, and Y. Kabak. Artemis message exchange framework: semantic interoperability of exchanged messages in the healthcare domain. *ACM Sigmod Record*, 34(3):71–76, 2005.
- [6] Clinical Document Architecture, Release 2. <http://www.hl7.org/v3ballot/html/infrastructure/cda/cda.htm>, last visited on August 2008.
- [7] HL7 CDA Conformance. http://www.hl7.org/v3ballot/html/infrastructure/cda/cda.htm#CDA_Conformance, last visited on August 2008.
- [8] HL7 Clinical Document Architecture (CDA) Refined Message Information Model (R-MIM). http://www.hl7.org/v3ballot/html/infrastructure/cda/cda.htm#HL7_CDA_RMIM, last visited on August 2008.
- [9] Clinical Document Architecture (CDA) Schema. <http://www.hl7.org/v3ballot/html/infrastructure/cda/CDA.xsd>, last visited on August 2008.

- [10] European Committee for Standardization (CEN). <http://www.cen.eu/>, last visited on August 2008.
- [11] CEN EN 13606-1. Health informatics – Electronic health record communication – Part 1: Reference model. Technical Report EN 13606-1, European Committee for Standardization, Brussels, Belgium, 2007.
- [12] CEN EN 13606-3. Health informatics – Electronic healthcare record communication – Part 3: Reference Archetypes and Term Lists. Technical Report EN 13606-3, European Committee for Standardization, Brussels, Belgium, 2008.
- [13] CEN ENV 13606. Medical informatics – Electronic health record communication. European prestandard env 13606, European Committee for Standardization, Brussels, Belgium.
- [14] HL7 Implementation Guide for 13606. <http://www.hl7.org.uk/repository/uploads-/402/1/HL7%20Implementation%20Guide%20for%2013606%20scopev0r6.doc>, last visited on August 2008.
- [15] HL7 Version 3 Standard: Clinical Statement Pattern, Release 1. <http://www.hl7.org-/v3ballot/html/domains/uvcs/uvcs.htm>, last visited on August 2008.
- [16] Cascading Style Sheets (CSS). <http://www.w3.org/Style/CSS/>, last visited on August 2008.
- [17] HL7 Common Terminology Services (CTS), Release 1. <http://www.hl7.org/v3ballot-/html/infrastructure/cts/cts.htm>, last visited on August 2008.
- [18] HL7 Data Types - Abstract Specification. <http://www.hl7.org/v3ballot/html-/infrastructure/datatypes/datatypes.htm>, last visited on August 2008.
- [19] Doctor Data Bank. <http://sbu.saglik.gov.tr/drBank/>, last visited on August 2008.
- [20] Dipak Kalra and David Lloyd. EHR Context Examples. Technical report. http://-www.chime.ucl.ac.uk/~rmhidsl/EHRcomDownloads/EHR-Context_Examplesv4.ppt, last visited on August 2008.
- [21] Description Logics. <http://dl.kr.org/>, last visited on August 2008.
- [22] HL7 Domain Message Information Model (D-MIM). <http://www.hl7.org/v3ballot-/html/help/v3guide/v3guide.htm#v3dmim>, last visited on August 2008.

- [23] Dossier Médical Personnel (DMP). <http://www.d-m-p.org/>, last visited on August 2008.
- [24] R. H. Dolin, L. Alschuler, S. Boyer, C. Beebe, F. M. Behlen, P. V. Biron, and A. Shabo. HL7 Clinical Document Architecture, Release 2. *Journal of the American Medical Informatics Association*, 13(1):30–39, 2006.
- [25] DRAFT prEN ISO 21090. Health Informatics – Harmonized data types for information interchange. Technical report.
- [26] HL7 Version 3 Standard: Transport Specification - ebXML, Release 2. <http://www.hl7.org/v3ballot/html/infrastructure/transport/transport-ebxml.htm>, last visited on August 2008.
- [27] CEN EN 13606 Reference Model XML Schema. <http://www.srdc.metu.edu.tr/~mustafa/thesis/schemas/13606%20-%20/>, last visited on August 2008.
- [28] M. Eichelberg, T. Aden, J. Riesmeier, A. Dogac, and G. Laleci. A survey and analysis of Electronic Healthcare Record standards. *ACM Computing Surveys*, 37(4):277–315, 2005.
- [29] e-Transformation in Health. Department of Information Processing, Ministry of Health, Turkey. http://www.saglik.gov.tr/EN/Tempdosyalar/533__e-transformationinhealth_07.pdf, last visited on August 2008.
- [30] Ulusal Sağlık Bilgi Sistemi/Sağlık-NET Entegrasyonu ile İlgili Genelge 2008/18. <http://www.saglik.gov.tr/TR/MevzuatGoster.aspx?F6E10F8892433CFF1A9547B61DAFFE2A56515916B329A1F1>, last visited on August 2008.
- [31] Carole A. Goble, Peter Crowther, and Danny Solomon. A medical terminology server. In *Database and Expert Systems Applications*, pages 661–670, 1994.
- [32] Canada Health Infoway. <http://www.w3.org/TR/xslt20>, last visited on August 2008.
- [33] Hibernate, Relational Persistence for Java and .NET. <http://www.hibernate.org/>, last visited on August 2008.
- [34] Health Level Seven (HL7). <http://www.hl7.org/>, last visited on August 2008.
- [35] HL7 Version 3 Design and Schema Tools. <http://www.hl7.org.au/HL7-V3-Resrcs.htm>, last visited on August 2008.

- [36] HL7 Hierarchical Message Description (HMD). <http://www.hl7.org/v3ballot/html/help/v3guide/v3guide.htm#v3ghmd>, last visited on August 2008.
- [37] Kose I., Akpınar N., Gurel M., Arslan Y., Ozer H., Yurt N., Kabak Y., Yuksel M., and Dogac A. Turkey's National Health Information System (NHIS). In *eChallenges Conference*, October 2008.
- [38] I. Iakovidis. Towards personal health records: Current situation, obstacles and trends in implementation of Electronic Healthcare Records in Europe . *International Journal of Medical Informatics*, 52(1-3):105–115, 1998.
- [39] International Statistical Classification of Diseases and Related Health Problems, 10th Revision (ICD-10), Second Edition. Technical report, World Health Organization, Geneva, Switzerland, 2005. <http://www.who.int/whosis/icd10/>, last visited on August 2008.
- [40] Saglik-Net Business Rules. Ministry of Health. http://www.sagliknet.saglik.gov.tr/portal_pages/notlogin/bilisimciler/docs/SaglikNET_is_Kurallari_BRMS.pdf, last visited on August 2008.
- [41] International Organization for Standardization (ISO). <http://www.iso.org/>, last visited on August 2008.
- [42] ISO/IEC 11179-4. [49] Information technology – Metadata registries (MDR) – Part 4: Formulation of data definitions. Technical Report ISO/IEC 11179-4, ISO.
- [43] Ministry of Health, Turkey HL7 Messages Integration Guide. http://www.sagliknet.saglik.gov.tr/portal_pages/notlogin/bilisimciler/docs/HL7_Mesajlari_Entegrasyon_Kilavuzu.rar, last visited on August 2008.
- [44] Kind Messaging for the Electronic Healthcare Records (Kmehr). <http://www.health.fgov.be/telematics/kmehr/one/>, last visited on August 2008.
- [45] Transformation of Kmehr to HL7 CDA. <http://www.chu-charleroi.be/kmehr/html/introductionXSLT.htm>, last visited on August 2008.
- [46] CEN EN 13606 Reference Model XML Schema by LinKEHR Project. <http://pangea.upv.es/linkehr/wp-content/uploads/2008/05/en13606schema.zip>, last visited on August 2008.

- [47] Logical Observation Identifiers Names and Codes (LOINC). <http://www.regenstrief.org/loinc/>, last visited on August 2008.
- [48] HL7 Message Development Framework, Version 3.3. <http://www.hl7.org/Library/MDF99/MDF99doc.zip>, last visited on August 2008.
- [49] MERNIS, Central Demographics Management System. http://www.nvi.gov.tr/Hakkimizda/Projeler,Mernis_Genel.html?pageindex=4, last visited on August 2008.
- [50] Medical Subject Heading Home provided by National Library of Medicine, National Institutes of Health (U.S.). <http://www.nlm.nih.gov/mesh/>, last visited on August 2008.
- [51] Transport Specification: Minimal Lower Layer Message Transport Protocol (MLLP), Release 2. <http://www.hl7.org/v3ballot/html/infrastructure/transport/transport-mllp.htm>, last visited on August 2008.
- [52] Microsoft XML Core Services (MSXML). <http://msdn.microsoft.com/en-us/xml/bb291077.aspx>, last visited on August 2008.
- [53] CDA Conformant Transformation of the Examination Transmission Schema. <http://www.srdc.metu.edu.tr/~mustafa/thesis/ExaminationCDA.xml>, last visited on August 2008.
- [54] CEN 13606 Conformant Transformation of the Examination CDA. <http://www.srdc.metu.edu.tr/~mustafa/thesis/ExaminationCEN13606.xml>, last visited on August 2008.
- [55] Example Examination Transmission Schema instance. <http://www.srdc.metu.edu.tr/~mustafa/thesis/ExaminationNHIS.xml>, last visited on August 2008.
- [56] CDA Conformant Examination Transmission Schema with its Coded Values Universalized. <http://www.srdc.metu.edu.tr/~mustafa/thesis/ExaminationCDAUniv.xml>, last visited on August 2008.
- [57] National Health Information Network (NHIN). <http://www.hhs.gov/healthit/healthnetwork/background/>, last visited on August 2008.
- [58] XSLT Files for NHIS to CDA and CDA to CEN 13606 Transformations. <http://www.srdc.metu.edu.tr/~mustafa/thesis/xslt/>, last visited on August 2008.

- [59] NHS UK. <http://www.nhs.uk/>, last visited on August 2008.
- [60] Kilic O. and Dogac A. Achieving Clinical Statement Interoperability using R-MIM and Archetype-based Semantic Transformations. *IEEE Transactions on Information Technology in Biomedicine*, to appear.
- [61] openEHR. <http://www.openehr.org/>, last visited on August 2008.
- [62] READ Codes. <http://www.connectingforhealth.nhs.uk/systemsandservices/data/readcodes/introduction>, last visited on August 2008.
- [63] The European Commission. Draft Recommendation on eHealth Interoperability. http://ec.europa.eu/information_society/newsroom/cf/document.cfm?action=display&doc_id=369, last visited on August 2008.
- [64] The European Commission. Commission Recommendation on Cross-Border Interoperability of Electronic Health Record Systems. <http://www.w3.org/TR/xslt20>, last visited on August 2008.
- [65] HL7 Refinement, Constraint and Localization, Release 2. <http://www.hl7.org/v3ballot/html/infrastructure/conformance/conformance.htm>, last visited on August 2008.
- [66] RIDE Project. <http://www.srdc.metu.edu.tr/webpage/projects/ride/>, last visited on August 2008.
- [67] RIDE Deliverable 2.1.1 - European Best Practices in Providing Semantic Interoperability in eHealth Domain. http://www.srdc.metu.edu.tr/webpage/projects/ride/deliverables/RIDE_D2.1.1_v2.0.doc, last visited on August 2008.
- [68] RIDE Deliverable 2.1.4 - European Good Practices. http://www.srdc.metu.edu.tr/webpage/projects/ride/deliverables/RIDE_D2.1.4_EuropGoodPractices_v1.2.doc, last visited on August 2008.
- [69] RIDE Deliverable 4.4.1 - RIDE Roadmap II. http://www.srdc.metu.edu.tr/webpage/projects/ride/deliverables/D4.4.1-RIDERoadmapII_v1.2.doc, last visited on August 2008.
- [70] HL7 Reference Information Model. <http://www.hl7.org/v3ballot/html/infrastructure/rim/rim.htm>, last visited on August 2008.

- [71] HL7 Refined Message Information Model (R-MIM). <http://www.hl7.org/v3ballot/html/help/v3guide/v3guide.htm#v3grmim>, last visited on August 2008.
- [72] Saxon, The XSLT and XQuery Processor. <http://saxon.sourceforge.net/>, last visited on August 2008.
- [73] S . Schulz and U. Hahn. Medical knowledge reengineering converting major portions of the UMLS into a terminological knowledge base . *International Journal of Medical Informatics*,, 64:207–221, 2001.
- [74] The Health Coding Reference Server 2. <http://sbu.saglik.gov.tr/SKRS2%5FListesini/>, last visited on August 2008.
- [75] The Systematized Nomenclature of Medicine (SNOMED). <http://www.ihtsdo.org/snomed-ct/>, last visited on August 2008.
- [76] Simple Object Access Protocol (SOAP). <http://www.w3.org/TR/soap/>, last visited on August 2008.
- [77] European Committee for Standardization - Technical Committee on Health Informatics. <http://www.cen251.org/>, last visited on August 2008.
- [78] Transformation in Health. Ministry of Health, Turkey. http://www.saglik.gov.tr/TR-/dosyagoster.aspx?DIL=1&BELGEANAH=9486&DOSYASIM=donusum_eng_2.zip, last visited on August 2008.
- [79] HL7 Version 3 Standard: Transport Specifications Overview. <http://www.hl7.org/v3ballot2008may/html/infrastructure/transport%20-%20transport-intro.htm>, last visited on August 2008.
- [80] The Scientific and Technological Research Council of Turkey (TUBITAK). http://www.nvi.gov.tr/Hakkimizda/Projeler,Mernis_Genel.html?pageindex=4, last visited on August 2008.
- [81] Unified Medical Language System (UMLS). <http://www.nlm.nih.gov/research/umls/>, last visited on August 2008.
- [82] UMLS Knowledge Source Server Developer’s Guide. <http://kswebp1.nlm.nih.gov/DocPortlet/html/dGuide/Guide.htm>, last visited on August 2008.

- [83] UMLSKS Developer's Guide, Web Service Operations. <http://kswebp1.nlm.nih.gov/DocPortlet/html/dGuide/webservices.html>, last visited on August 2008.
- [84] Unified Medical Language System (UMLS) Knowledge Source Server (KSS). <http://umlsks.nlm.nih.gov/>, last visited on August 2008.
- [85] United States National Library of Medicine. <http://www.nlm.nih.gov/>, last visited on August 2008.
- [86] The National Health Data Dictionary (NHDD) of Turkey. http://www.sagliknet.saglik.gov.tr/portal_pages/notlogin/bilisimciler/docs/usvs_sozluk_1.1.rar, last visited on August 2008.
- [87] Generation and registration of Universally Unique Identifiers (UUIDs) and their use as ASN.1 Object Identifier components. Open Systems Interconnection, International Telecommunication Union. <http://www.itu.int/ITU-T/studygroups/com17-oid/X.667-E.pdf>, last visited on August 2008.
- [88] HL7 Version 3 Guide. <http://www.hl7.org/v3ballot/html/help/v3guide/v3guide.htm>, last visited on August 2008.
- [89] HL7 Vocabulary. <http://www.hl7.org/v3ballot/html/infrastructure/vocabulary/vocabulary.htm>, last visited on August 2008.
- [90] The World Wide Web Consortium (W3C). <http://www.w3.org/>, last visited on August 2008.
- [91] World Health Organization. <http://www.who.int/en/>, last visited on August 2008.
- [92] <http://www.hl7.org/v3ballot/html/infrastructure/transport/transport-wsprofiles.htm>. <http://www.w3.org/XML/Schema>, last visited on August 2008.
- [93] Web Services Security: SOAP Message Security 1.1 (WS-Security 2004). OASIS Standard Specification. <http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf>, last visited on August 2008.
- [94] Apache Xerces Java XML Parser. <http://xerces.apache.org/xerces-j/>, last visited on August 2008.
- [95] The Extensible Markup Language (XML). <http://www.w3.org/XML/>, last visited on August 2008.

- [96] HL7 Version 3 Standard: XML Implementable Technology Specification for V3 Structures, R1. <http://www.hl7.org/v3ballot/html/infrastructure/itsxml/messaging-its-xml.htm>, last visited on August 2008.
- [97] Altova XMLSpy XML Editor. http://www.altova.com/products/xmlspy/xml_editor.html, last visited on August 2008.
- [98] The XML Path Language (XPath). <http://www.w3.org/TR/xpath>, last visited on August 2008.
- [99] XML Schema Definition. <http://www.w3.org/XML/Schema>, last visited on August 2008.
- [100] XML Schema Instance. <http://www.w3.org/TR/xmlschema-1/>, last visited on August 2008.
- [101] The Extensible Stylesheet Language (XSL) Family. <http://www.w3.org/Style/XSL/>, last visited on August 2008.
- [102] XSL Formatting Objects (XSL-FO). <http://www.w3.org/TR/xsl>, last visited on August 2008.
- [103] XSL Transformations (XSLT). <http://www.w3.org/TR/xslt>, last visited on August 2008.
- [104] XSL Transformations (XSLT) 2.0. <http://www.w3.org/TR/xslt20>, last visited on August 2008.
- [105] Kabak Y., Dogac A., Kose I., Akpinar N., Gurel M., Arslan Y., Ozer H., Yurt N., Ozcam A., Kirici S., Yuksel M., and Sabur E. The Use of HL7 CDA in the National Health Information System (NHIS) of Turkey. In *9th International HL7 Interoperability Conference (IHIC)*, October 2008.
- [106] H. Zimmermann. OSI Reference Model - The ISO Model of Architecture for Open Systems Interconnection . *IEEE Transactions on Communications*, COM-28:425–432, 1980.

APPENDIX A

AN EXAMPLE EXAMINATION TRANSMISSION SCHEMA INSTANCE

```
<?xml version="1.0" encoding="utf-8"?>
<MCCI_IN000001TR01 xmlns="urn:hl7-org:v3">
  <id root="2.16.840.1.113883.3.129.2.1.2" extension="5b50d046-b567-4fba-a365-60c220c9fae1"/>
  <creationTime value="20080502102643"/>
  <responseModeCode code="Q"/>
  <interactionId root="2.16.840.1.113883.3.129.2.1.1" extension="MCCI_IN000001TR01"/>
  <processingCode code="P"/>
  <processingModeCode code="T"/>
  <acceptAckCode code="AL"/>
  <receiver typeCode="RCV">
    <device classCode="DEV" determinerCode="INSTANCE">
      <id root="2.16.840.1.113883.3.129.1.1.5" extension="USBS"/>
    </device>
  </receiver>
  <sender typeCode="SND">
    <device classCode="DEV" determinerCode="INSTANCE">
      <id root="2.16.840.1.113883.3.129.1.1.5" extension="SRDC-Deneme"/>
    </device>
  </sender>
  <controlActEvent classCode="CACT" moodCode="EVN">
    <subject typeCode="SUBJ">
      <examination classCode="DOCCLIN" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.3" extension="0006b9bd-b282-0000-8b05-eeb9f3d42015"/>
        <code code="MUAYENE" codeSystem="2.16.840.1.113883.3.129.2.2.1" codeSystemName="Döküman Tipi"
          codeSystemVersion="1.0" displayName="Muayene MSVS (Vatandaş/Yabancı)"/>
        <effectiveTime value="20080502102643"/>
        <confidentialityCode code="1" codeSystem="2.16.840.1.113883.3.129.1.2.77"
          codeSystemName="Gizlilik" codeSystemVersion="1.0" displayName="Normal"/>
        <languageCode code="tr-TR"/>
        <versionNumber value="1"/>
        <recordTarget typeCode="RCT" contextControlCode="OP">
          <patientRole classCode="PAT">
```

```

<id root="2.16.840.1.113883.3.129.1.1.1" extension="12345678905"/>
<patient determinerCode="INSTANCE" classCode="PSN">
  <name>
    <family>Yüksel</family>
    <given>Mustafa</given>
  </name>
  <administrativeGenderCode code="1" codeSystem="2.16.840.1.113883.3.129.1.2.21"
    codeSystemName="Cinsiyet" codeSystemVersion="1.0" displayName="Erkek"/>
  <birthTime value="19840101"/>
  <raceCode code="TR" codeSystem="2.16.840.1.113883.3.129.1.2.52" codeSystemName="Uyruk"
    codeSystemVersion="1.0" displayName="Türkiye"/>
</patient>
</patientRole>
</recordTarget>
<author typeCode="AUT" contextControlCode="OP">
  <time value="20180402102643"/>
  <assignedAuthor classCode="ASSIGNED">
    <id root="2.16.840.1.113883.3.129.1.1.1" extension="12345678905"/>
  </assignedAuthor>
</author>
<custodian typeCode="CST">
  <assignedCustodian classCode="ASSIGNED">
    <representedHealthcareOrganization classCode="ENT" determinerCode="INSTANCE">
      <id root="2.16.840.1.113883.3.129.1.1.6" extension="4553"/>
    </representedHealthcareOrganization>
  </assignedCustodian>
</custodian>
<primaryInformationRecipient typeCode="PRCP">
  <recipient classCode="ASSIGNED">
    <representedMinistryOfHealth classCode="ORG" determinerCode="INSTANCE">
      <id root="2.16.840.1.113883.3.129.1.1.6" extension="5881"/>
    </representedMinistryOfHealth>
  </recipient>
</primaryInformationRecipient>
<component typeCode="COMP" contextConductionInd="true">
  <structuredBody classCode="DOCBODY" moodCode="EVN">
    <component1 typeCode="COMP" contextConductionInd="true">
      <testResultDataset classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.4"
          extension="97160bb4-8a90-4a36-a91c-2776a9a97a97"/>
        <code code="TETKIKSONUCU" codeSystem="2.16.840.1.113883.3.129.2.2.2"
          codeSystemName="Veriseti" codeSystemVersion="1.0" displayName="Tetkik Sonucu Veriseti"/>
        <text mediaType="text/x-h17-text+xml"/>
        <author typeCode="AUT" contextControlCode="OP">
          <testDoctor classCode="ASSIGNED">
            <id root="2.16.840.1.113883.3.129.1.1.1" extension="12345678905"/>
          </testDoctor>
        </author>
      </testResultDataset>
    </component1>
  </structuredBody>
</component>

```



```

<component typeCode="COMP" contextConductionInd="true">
  <testResultSection classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="d124ade3-6649-498f-a154-6876cc3441b6"/>
    <code code="TETKIKSONUC" codeSystem="2.16.840.1.113883.3.129.2.2.3"
      codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
      displayName="Tetkik Sonuç Verisinin Olduğu Bölüm"/>
    <text mediaType="text/x-hl7-text+xml">Örnek tetkik sonuç dokümanı</text>
    <component typeCode="COMP" contextConductionInd="true">
      <testResultOrganizer moodCode="EVN" classCode="CLUSTER">
        <statusCode code="completed"/>
        <performer typeCode="PRF">
          <studyOrganization classCode="ASSIGNED">
            <id root="2.16.840.1.113883.3.129.1.1.6" extension="3199"/>
          </studyOrganization>
        </performer>
        <component typeCode="COMP" contextConductionInd="true">
          <testResult moodCode="EVN" classCode="OBS">
            <code code="700490" codeSystem="2.16.840.1.113883.3.129.1.2.2"
              codeSystemName="SUT" codeSystemVersion="1.0"
              displayName="Elektrokardiyogram, evde çekim"/>
            <activityTime value="20080401"/>
          </testResult>
        </component>
        <component typeCode="COMP" contextConductionInd="true">
          <testResult moodCode="EVN" classCode="OBS">
            <code code="901620" codeSystem="2.16.840.1.113883.3.129.1.2.2"
              codeSystemName="SUT" codeSystemVersion="1.0"
              displayName="Tam Kan (Hemogram)/>
            <activityTime value="20080701"/>
          </testResult>
        </component>
      </testResultOrganizer>
    </component>
  </testResultSection>
</component>
</testResultDataset>
</component1>
<component2 typeCode="COMP" contextConductionInd="true">
  <dischargeDataset classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.4"
      extension="28661575-af59-45be-bdf5-3590385801df"/>
    <code code="CIKIS" codeSystem="2.16.840.1.113883.3.129.2.2.2"
      codeSystemName="Veriseti" codeSystemVersion="1.0" displayName="Çıkış Veriseti"/>
    <text mediaType="text/x-hl7-text+xml"/>
    <component2 typeCode="COMP" contextConductionInd="true">
      <dischargeSection classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.5"

```

```

        extension="ac4c115f-e1cb-4546-a2e5-db37ea55234b"/>
<code code="CIKIS" codeSystem="2.16.840.1.113883.3.129.2.2.3"
    codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
    displayName="Çıkış Verisinin Olduğu Bölüm"/>
<text mediaType="text/x-hl7-text+xml"/>
<component typeCode="COMP" contextConductionInd="true">
    <discharge classCode="ACT" moodCode="EVN">
        <code code="1" codeSystem="2.16.840.1.113883.3.129.1.2.9"
            codeSystemName="Çıkış Şekli" codeSystemVersion="1.0"
            displayName="Tedavi önerileriyle çıkış"/>
        <effectiveTime value="200804021021"/>
    </discharge>
</component>
</dischargeSection>
</component2>
</dischargeDataset>
</component2>
<component3 typeCode="COMP" contextConductionInd="true">
    <examinationDataset classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.4"
            extension="4e7e0004-8e9e-44d2-9a9e-099d071e646a"/>
        <code code="MUAYENE" codeSystem="2.16.840.1.113883.3.129.2.2.2"
            codeSystemName="Veriseti" codeSystemVersion="1.0" displayName="Muayene Veriseti"/>
        <text>Hastanın tezle uğraşırken geçirdiği hastalıkları özetler</text>
        <author typeCode="AUT" contextControlCode="OP">
            <doctor classCode="ASSIGNED">
                <id root="2.16.840.1.113883.3.129.1.1.1" extension="12345678905"/>
            </doctor>
        </author>
    </examinationDataset>
</component3>
<component1 typeCode="COMP" contextConductionInd="true">
    <examProtocolNoSection classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.5"
            extension="de380a98-d31f-4a33-a878-7366da629737"/>
        <code code="PROTOKOLNO" codeSystem="2.16.840.1.113883.3.129.2.2.3"
            codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
            displayName="Protokol No Bilgisinin Olduğu Bölüm"/>
        <text mediaType="text/x-hl7-text+xml"/>
        <component typeCode="COMP" contextConductionInd="true">
            <examProtocolNo moodCode="EVN" classCode="ACT">
                <id root="2.16.840.1.113883.3.129.1.1.4" extension="111"/>
            </examProtocolNo>
        </component>
    </examProtocolNoSection>
</component1>
<component2 typeCode="COMP" contextConductionInd="true">
    <reportSection classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.5"
            extension="cba90686-fa28-4a0e-9ee6-342d7bcb3302"/>
    </reportSection>
</component2>

```

```

<code code="RAPOR" codeSystem="2.16.840.1.113883.3.129.2.2.3"
  codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
  displayName="Rapor Verisinin Olduğu Bölüm"/>
<text mediaType="text/x-hl7-text+xml"/>
<component typeCode="COMP" contextConductionInd="true">
  <report moodCode="EVN" classCode="OBS">
    <code code="4" codeSystem="2.16.840.1.113883.3.129.1.2.8"
      codeSystemName="Rapor Türü" codeSystemVersion="1.0" displayName="Hastalık"/>
    <effectiveTime value="20080401"/>
    <value>rapor dokümanı</value>
  </report>
</component>
</reportSection>
</component2>
<component3 typeCode="COMP" contextConductionInd="true">
  <investigationSection classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="59c0024f-d799-498d-8b00-e849136ad426"/>
    <code code="BULGU" codeSystem="2.16.840.1.113883.3.129.2.2.3"
      codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
      displayName="Bulgu Verisinin Olduğu Bölüm"/>
    <text>Net bir problemle karşılaşılmadı</text>
  </investigationSection>
</component3>
<component4 typeCode="COMP" contextConductionInd="true">
  <historySection classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="e3c97aa2-ba47-40d4-a068-881a6c51417a"/>
    <code code="HIKAYE" codeSystem="2.16.840.1.113883.3.129.2.2.3"
      codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
      displayName="Hikaye Verisinin Olduğu Bölüm"/>
    <text>Hastanın tezle uğraşırken geçirdiği hastalıkları özetler</text>
  </historySection>
</component4>
<component5 typeCode="COMP" contextConductionInd="true">
  <complaintSection classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="6bde0f05-f303-4109-88ac-c22a23a67094"/>
    <code code="SIKAYET" codeSystem="2.16.840.1.113883.3.129.2.2.3"
      codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
      displayName="Şikayet Verisinin Olduğu Bölüm"/>
    <text>Bitkin</text>
  </complaintSection>
</component5>
<component6 typeCode="COMP" contextConductionInd="true">
  <testSection classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="e59daa46-6c71-4eea-852e-406e5a440f13"/>

```

```

<code code="TETKIK" codeSystem="2.16.840.1.113883.3.129.2.2.3"
  codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
  displayName="Tetkik Verisinin Olduğu Bölüm"/>
<text mediaType="text/x-hl7-text+xml"/>
<component typeCode="COMP" contextConductionInd="true">
  <test classCode="OBS" moodCode="EVN">
    <code code="901620" codeSystem="2.16.840.1.113883.3.129.1.2.2"
      codeSystemName="SUT" codeSystemVersion="1.0" displayName="Tam Kan (Hemogram)"/>
    <performer typeCode="PRF">
      <performerOrganization classCode="ASSIGNED">
        <id root="2.16.840.1.113883.3.129.1.1.6" extension="4553"/>
      </performerOrganization>
    </performer>
  </test>
</component>
<component typeCode="COMP" contextConductionInd="true">
  <test classCode="OBS" moodCode="EVN">
    <code code="701540" codeSystem="2.16.840.1.113883.3.129.1.2.2"
      codeSystemName="SUT" codeSystemVersion="1.0"
      displayName="Üzofagoskopi, gastroskopi, duodenoskopi (biri veya hepsi)"/>
    <performer typeCode="PRF">
      <performerOrganization classCode="ASSIGNED">
        <id root="2.16.840.1.113883.3.129.1.1.6" extension="4553"/>
      </performerOrganization>
    </performer>
  </test>
</component>
</testSection>
</component6>
<component7 typeCode="COMP" contextConductionInd="true">
  <procedureSection classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="cf813e90-e027-4441-9e87-c287bdc5b370"/>
    <code code="MUDAHALE" codeSystem="2.16.840.1.113883.3.129.2.2.3"
      codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
      displayName="Müdehale Verisinin Olduğu Bölüm"/>
    <text mediaType="text/x-hl7-text+xml"/>
    <component typeCode="COMP" contextConductionInd="true">
      <procedure moodCode="EVN" classCode="PROC">
        <code code="520030" codeSystem="2.16.840.1.113883.3.129.1.2.2"
          codeSystemName="SUT" codeSystemVersion="1.0"
          displayName="NORMAL POLIKLINİK MUAYENE ÜCRETLERİ"/>
      </procedure>
    </component>
  </procedureSection>
</component7>
<component8 typeCode="COMP" contextConductionInd="true">
  <diagnosisSection classCode="DOCSECT" moodCode="EVN">

```

```

<id root="2.16.840.1.113883.3.129.2.1.5"
  extension="6c4b1e87-c4f4-42b0-aaeb-05e23a77ed5a"/>
<code code="TANI" codeSystem="2.16.840.1.113883.3.129.2.2.3"
  codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
  displayName="Tanı Verisinin Olduğu Bölüm"/>
<text mediaType="text/x-hl7-text+xml"/>
<component typeCode="COMP" contextConductionInd="true">
  <diagnosis moodCode="EVN" classCode="OBS">
    <code code="ANATANI" codeSystem="2.16.840.1.113883.3.129.2.2.6"
      codeSystemName="Tanı Tipi" codeSystemVersion="1.0" displayName="Ana Tanı"/>
    <value code="K29.8" codeSystem="2.16.840.1.113883.6.3" codeSystemName="ICD-10"
      codeSystemVersion="1.0" displayName="Duodenit"/>
  </diagnosis>
</component>
<component typeCode="COMP" contextConductionInd="true">
  <diagnosis moodCode="EVN" classCode="OBS">
    <code code="EKTANI" codeSystem="2.16.840.1.113883.3.129.2.2.6"
      codeSystemName="Tanı Tipi" codeSystemVersion="1.0" displayName="Ek Tanı"/>
    <value code="G43" codeSystem="2.16.840.1.113883.6.3" codeSystemName="ICD-10"
      codeSystemVersion="1.0" displayName="Migren"/>
  </diagnosis>
</component>
</diagnosisSection>
</component8>
<component9 typeCode="COMP" contextConductionInd="true">
  <examinationSection classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="212802ba-7c64-47d5-8e52-b02a6ea44531"/>
    <code code="MUAYENE" codeSystem="2.16.840.1.113883.3.129.2.2.3"
      codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
      displayName="Muayene Verisinin Olduğu Bölüm"/>
    <text mediaType="text/x-hl7-text+xml"/>
    <component typeCode="COMP" contextConductionInd="true">
      <encounter classCode="ENC" moodCode="EVN">
        <effectiveTime>
          <low value="200803251021"/>
          <high value="200804011021"/>
        </effectiveTime>
        <location typeCode="LOC">
          <clinic classCode="SDLLOC">
            <code code="22" codeSystem="2.16.840.1.113883.3.129.1.2.1"
              codeSystemName="Klinikler" codeSystemVersion="1.0"
              displayName="Gastroenteroloji"/>
          </clinic>
        </location>
      </encounter>
    </component>
  </examinationSection>

```

```

        </component9>
    </examinationDataset>
</component3>
<component4 typeCode="COMP" contextConductionInd="true">
    <receptionDataset classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.4"
            extension="dd425f3b-a09c-434a-a648-9d67d3775adf"/>
        <code code="KABUL" codeSystem="2.16.840.1.113883.3.129.2.2"
            codeSystemName="Veriseti" codeSystemVersion="1.0" displayName="Kabul Veriseti"/>
        <text mediaType="text/x-hl7-text+xml"/>
    <component1 typeCode="COMP" contextConductionInd="true">
        <socialSecurityFollowNumberSection classCode="DOCSECT" moodCode="EVN">
            <id root="2.16.840.1.113883.3.129.2.1.5"
                extension="c25dd89c-6850-49eb-bf10-c0df3e92e5bf"/>
            <code code="SGKTAKIPNO" codeSystem="2.16.840.1.113883.3.129.2.2.3"
                codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
                displayName="SGK Takip No Bilgisinin Olduğu Bölüm"/>
            <text mediaType="text/x-hl7-text+xml">123</text>
        </socialSecurityFollowNumberSection>
    </component1>
    <component2 typeCode="COMP" contextConductionInd="true">
        <referralSection classCode="DOCSECT" moodCode="EVN">
            <id root="2.16.840.1.113883.3.129.2.1.5"
                extension="54083c07-d5f3-4a85-9fb9-8fcacb253a0b"/>
            <code code="SEVK" codeSystem="2.16.840.1.113883.3.129.2.2.3"
                codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
                displayName="Sevk Verisinin Olduğu Bölüm"/>
            <text mediaType="text/x-hl7-text+xml"/>
        <component typeCode="COMP" contextConductionInd="true">
            <referral moodCode="EVN" classCode="ACT">
                <effectiveTime value="20080313"/>
                <referrer typeCode="REF" contextControlCode="OP">
                    <referralFromClinic classCode="SDL0C">
                        <code code="22" codeSystem="2.16.840.1.113883.3.129.1.2.1"
                            codeSystemName="Klinikler" codeSystemVersion="1.0"
                            displayName="Gastroenteroloji"/>
                    </referralFromClinic>
                </referrer>
            <component typeCode="COMP">
                <referralDiagnosis moodCode="EVN" classCode="OBS">
                    <code code="SEVKTANISI" codeSystem="2.16.840.1.113883.3.129.2.2.6"
                        codeSystemName="Tanı Tipi" codeSystemVersion="1.0"
                        displayName="Sevk Tanısı"/>
                    <value code="K55" codeSystem="2.16.840.1.113883.6.3"
                        codeSystemName="ICD-10" codeSystemVersion="1.0"
                        displayName="BARSAĞIN VASKÜLER BOZUKLUKLARI"/>
                </referralDiagnosis>
            </component>
        </component>
    </component2>
</component4>

```

```

        </referral>
    </component>
</referralSection>
</component2>
<component3 typeCode="COMP" contextConductionInd="true">
    <registrationSection classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.5"
            extension="41f0e5f8-e66a-4af7-bcb9-0dc9a93ba720"/>
        <code code="KABUL" codeSystem="2.16.840.1.113883.3.129.2.2.3"
            codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
            displayName="Kabul Verisinin Olduğu Bölüm"/>
        <text mediaType="text/x-hl7-text+xml"/>
        <component typeCode="COMP" contextConductionInd="true">
            <registration moodCode="EVN" classCode="ACT">
                <code code="1" codeSystem="2.16.840.1.113883.3.129.1.2.7"
                    codeSystemName="Kabul Şekli" codeSystemVersion="1.0" displayName="Diğer"/>
                <effectiveTime value="200803231021"/>
            </registration>
        </component>
    </registrationSection>
</component3>
<component4 typeCode="COMP" contextConductionInd="true">
    <caseTypeSection classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.5"
            extension="537a52e1-36f1-4e04-a4bd-a71056471913"/>
        <code code="VAKATURU" codeSystem="2.16.840.1.113883.3.129.2.2.3"
            codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
            displayName="Vaka Türü Bilgisinin Olduğu Bölüm"/>
        <text mediaType="text/x-hl7-text+xml"/>
        <component typeCode="COMP" contextConductionInd="true">
            <caseType moodCode="EVN" classCode="ACT">
                <code code="1" codeSystem="2.16.840.1.113883.3.129.1.2.53"
                    codeSystemName="Vaka Türü" codeSystemVersion="1.0" displayName="Normal"/>
            </caseType>
        </component>
    </caseTypeSection>
</component4>
</receptionDataset>
</component4>
<component5 typeCode="COMP" contextConductionInd="true">
    <prescriptionDataset classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.4"
            extension="388c478d-c649-4273-803f-51b3c383f6ac"/>
        <code code="RECETE" codeSystem="2.16.840.1.113883.3.129.2.2.2"
            codeSystemName="Veriseti" codeSystemVersion="1.0" displayName="Reçete Veriseti"/>
        <text mediaType="text/x-hl7-text+xml"/>
        <author typeCode="AUT" contextControlCode="OP">
            <assignedDoctor classCode="ASSIGNED">

```

```

        <id root="2.16.840.1.113883.3.129.1.1.1" extension="12345678905"/>
        <id root="2.16.840.1.113883.3.129.1.1.2" extension="12345678906"/>
    </assignedDoctor>
</author>
<component1 typeCode="COMP" contextConductionInd="true">
    <medicationSection classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.5"
            extension="4ec898be-2c17-4637-b383-9139be24b305"/>
        <code code="ILAC" codeSystem="2.16.840.1.113883.3.129.2.2.3"
            codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
            displayName="İlaç Verisinin Olduğu Bölüm"/>
        <text mediaType="text/x-hl7-text+xml"/>
        <component typeCode="COMP" contextConductionInd="true">
            <substanceAdministration classCode="SBADM" moodCode="RQO">
                <effectiveTime>
                    <period value="1" unit="d"/>
                </effectiveTime>
                <routeCode code="1" codeSystem="2.16.840.1.113883.3.129.1.2.47"
                    codeSystemName="İlaç Kullanım Şekli" codeSystemVersion="1.0"
                    displayName="Ağızdan (Oral)"/>
                <doseQuantity value="1"/>
                <consumable typeCode="CSM">
                    <manufacturedProduct classCode="MANU">
                        <manufacturedMaterial determinerCode="KIND" classCode="MMAT">
                            <code code="8699536160085" codeSystem="2.16.840.1.113883.3.129.1.2.3"
                                codeSystemName="İlaçlar" codeSystemVersion="1.0"
                                displayName="LANSOR MIKROPELLET KAPSÜL 30 MG 28 CAP"/>
                        </manufacturedMaterial>
                    </manufacturedProduct>
                </consumable>
                <component typeCode="COMP" contextConductionInd="true">
                    <supply moodCode="INT" classCode="SPLY">
                        <quantity value="2"/>
                    </supply>
                </component>
            </substanceAdministration>
        </component>
    </medicationSection>
</component1>
<component2 typeCode="COMP" contextConductionInd="true">
    <prescriptionHeaderSection classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.5"
            extension="d7cb1191-033d-4730-9236-d3782ddf0b60"/>
        <code code="RECETEBASLIK" codeSystem="2.16.840.1.113883.3.129.2.2.3"
            codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
            displayName="Reçete Başlık Bilgisinin Olduğu Bölüm"/>
        <text mediaType="text/x-hl7-text+xml"/>
        <component typeCode="COMP" contextConductionInd="true">

```



```

<prescriptionHeader classCode="ACT" moodCode="EVN">
  <code code="1" codeSystem="2.16.840.1.113883.3.129.1.2.48"
    codeSystemName="Reçete Türü" codeSystemVersion="1.0" displayName="Normal"/>
  <effectiveTime value="20080401"/>
  <component1 typeCode="COMP" contextConductionInd="true">
    <protocolNo moodCode="EVN" classCode="ACT">
      <id root="2.16.840.1.113883.3.129.1.1.4" extension="111"/>
    </protocolNo>
  </component1>
  <component2 typeCode="COMP" contextConductionInd="true">
    <sGKRationBook moodCode="EVN" classCode="ACT">
      <id root="2.16.840.1.113883.3.129.1.1.3" extension="12321123"/>
    </sGKRationBook>
  </component2>
</prescriptionHeader>
</component>
</prescriptionHeaderSection>
</component2>
<component3 typeCode="COMP" contextConductionInd="true">
  <guarantorSection classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="a0a7a2ea-453a-4b52-b741-d39ff1f968b0"/>
    <code code="SOSYALGUVENCEDURUMU" codeSystem="2.16.840.1.113883.3.129.2.2.3"
      codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
      displayName="Sosyal Güvence Durumu Bilgilerinin Olduğu Bölüm"/>
    <text mediaType="text/x-hl7-text+xml"/>
    <component typeCode="COMP" contextConductionInd="true">
      <guarantor moodCode="EVN" classCode="ACT">
        <code code="1" codeSystem="2.16.840.1.113883.3.129.1.2.11"
          codeSystemName="Sosyal Güvenlik" codeSystemVersion="1.0" displayName="SSK"/>
      </guarantor>
    </component>
  </guarantorSection>
</component3>
<component4 typeCode="COMP" contextConductionInd="true">
  <presDiagnosisSection classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="2a1cbba1-cd5b-4ca1-b602-878c087b9a49"/>
    <code code="TANI" codeSystem="2.16.840.1.113883.3.129.2.2.3"
      codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
      displayName="Tanı Verisinin Olduğu Bölüm"/>
    <text mediaType="text/x-hl7-text+xml"/>
    <component typeCode="COMP" contextConductionInd="true">
      <presDiagnosis moodCode="EVN" classCode="OBS">
        <code code="ANATANI" codeSystem="2.16.840.1.113883.3.129.2.2.6"
          codeSystemName="Tanı Tipi" codeSystemVersion="1.0" displayName="Ana Tanı"/>
        <value code="K29.8" codeSystem="2.16.840.1.113883.6.3"
          codeSystemName="ICD-10" codeSystemVersion="1.0" displayName="Duodenit"/>
      </presDiagnosis>
    </component>
  </presDiagnosisSection>
</component4>

```

```
        </presDiagnosis>
      </component>
    </presDiagnosisSection>
  </component4>
</prescriptionDataset>
</component5>
</structuredBody>
</component>
</examination>
</subject>
</controlActEvent>
</MCCI_IN000001TR01>
```

APPENDIX B

THE CDA CONFORMANT EXAMINATION TRANSMISSION SCHEMA INSTANCE

```
<?xml version="1.0" encoding="UTF-8"?>
<ClinicalDocument xmlns="urn:hl7-org:v3" classCode="DOCCLIN" moodCode="EVN">
  <typeId root="2.16.840.1.113883.1.3" extension="POCD_HD000040"/>
  <id root="2.16.840.1.113883.3.129.2.1.3" extension="0006b9bd-b282-0000-8b05-eeb9f3d42015"/>
  <code code="MUAYENE" codeSystem="2.16.840.1.113883.3.129.2.2.1" codeSystemName="Döküman Tipi"
    codeSystemVersion="1.0" displayName="Muayene MSVS (Vatandaş/Yabancı)"/>
  <effectiveTime value="20080502102643"/>
  <confidentialityCode code="1" codeSystem="2.16.840.1.113883.3.129.1.2.77" codeSystemName="Gizlilik"
    codeSystemVersion="1.0" displayName="Normal"/>
  <languageCode code="tr-TR"/>
  <versionNumber value="1"/>
  <recordTarget typeCode="RCT" contextControlCode="OP">
    <patientRole classCode="PAT">
      <id root="2.16.840.1.113883.3.129.1.1.1" extension="12345678905"/>
      <patient determinerCode="INSTANCE" classCode="PSN">
        <name>
          <family>Yüksel</family>
          <given>Mustafa</given>
        </name>
        <administrativeGenderCode code="1" codeSystem="2.16.840.1.113883.3.129.1.2.21"
          codeSystemName="Cinsiyet" codeSystemVersion="1.0" displayName="Erkek"/>
        <birthTime value="19840101"/>
        <raceCode code="TR" codeSystem="2.16.840.1.113883.3.129.1.2.52" codeSystemName="Uyruk"
          codeSystemVersion="1.0" displayName="Türkiye"/>
      </patient>
    </patientRole>
  </recordTarget>
  <author typeCode="AUT" contextControlCode="OP">
    <time value="20180402102643"/>
  </author>
</ClinicalDocument>
```

```

    <assignedAuthor classCode="ASSIGNED">
      <id root="2.16.840.1.113883.3.129.1.1.1" extension="12345678905"/>
    </assignedAuthor>
  </author>
  <custodian typeCode="CST">
    <assignedCustodian classCode="ASSIGNED">
      <representedCustodianOrganization>
        <id root="2.16.840.1.113883.3.129.1.1.6" extension="4553"/>
      </representedCustodianOrganization>
    </assignedCustodian>
  </custodian>
  <informationRecipient typeCode="PRCP">
    <intendedRecipient classCode="ASSIGNED">
      <receivedOrganization classCode="ORG" determinerCode="INSTANCE">
        <id root="2.16.840.1.113883.3.129.1.1.6" extension="5881"/>
      </receivedOrganization>
    </intendedRecipient>
  </informationRecipient>
  <component>
    <structuredBody>
      <component>
        <section classCode="DOCSECT" moodCode="EVN">
          <id root="2.16.840.1.113883.3.129.2.1.4" extension="97160bb4-8a90-4a36-a91c-2776a9a97a97"/>
          <code code="TETKIKSONUCU" codeSystem="2.16.840.1.113883.3.129.2.2.2"
            codeSystemName="Veriseti" codeSystemVersion="1.0" displayName="Tetkik Sonucu Veriseti"/>
          <text mediaType="text/x-hl7-text+xml"/>
          <author typeCode="AUT" contextControlCode="OP">
            <time value="20080502102643"/>
            <assignedAuthor classCode="ASSIGNED">
              <id root="2.16.840.1.113883.3.129.1.1.1" extension="12345678905"/>
            </assignedAuthor>
          </author>
        </section>
        <component typeCode="COMP" contextConductionInd="true">
          <section classCode="DOCSECT" moodCode="EVN">
            <id root="2.16.840.1.113883.3.129.2.1.5"
              extension="d124ade3-6649-498f-a154-6876cc3441b6"/>
            <code code="TETKIKSONUC" codeSystem="2.16.840.1.113883.3.129.2.2.3"
              codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
              displayName="Tetkik Sonuç Verisinin Olduğu Bölüm"/>
            <text mediaType="text/x-hl7-text+xml">Örnek tetkik sonuç dokümanı</text>
          </section>
          <entry>
            <organizer classCode="CLUSTER" moodCode="EVN">
              <statusCode code="completed"/>
              <performer>
                <assignedEntity classCode="ASSIGNED">
                  <id root="2.16.840.1.113883.3.129.1.1.6" extension="3199"/>
                </assignedEntity>
              </performer>
            </organizer>
          </entry>
        </component>
      </component>
    </structuredBody>
  </component>

```

```

<component typeCode="COMP" contextConductionInd="true">
  <observation moodCode="EVN" classCode="OBS">
    <code code="700490" codeSystem="2.16.840.1.113883.3.129.1.2.2"
      codeSystemName="SUT" codeSystemVersion="1.0"
      displayName="Elektrokardiyogram, evde çekim"/>
    <effectiveTime value="20080401"/>
  </observation>
</component>
<component typeCode="COMP" contextConductionInd="true">
  <observation moodCode="EVN" classCode="OBS">
    <code code="901620" codeSystem="2.16.840.1.113883.3.129.1.2.2"
      codeSystemName="SUT" codeSystemVersion="1.0"
      displayName="Tam Kan (Hemogram)"/>
    <effectiveTime value="20080701"/>
  </observation>
</component>
</organizer>
</entry>
</section>
</component>
</section>
</component>
<component>
  <section classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.4" extension="28661575-af59-45be-bdf5-3590385801df"/>
    <code code="CIKIS" codeSystem="2.16.840.1.113883.3.129.2.2.2" codeSystemName="Veriseti"
      codeSystemVersion="1.0" displayName="Çıkış Veriseti"/>
    <text mediaType="text/x-hl7-text+xml"/>
    <component typeCode="COMP" contextConductionInd="true">
      <section classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.5"
          extension="ac4c115f-e1cb-4546-a2e5-db37ea55234b"/>
        <code code="CIKIS" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
          codeSystemVersion="1.0" displayName="Çıkış Verisinin Olduğu Bölüm"/>
        <text mediaType="text/x-hl7-text+xml"/>
        <entry typeCode="COMP" contextConductionInd="true">
          <act classCode="ACT" moodCode="EVN">
            <code code="1" codeSystem="2.16.840.1.113883.3.129.1.2.9" codeSystemName="Çıkış Şekli"
              codeSystemVersion="1.0" displayName="Tedavi önerileriyle çıkış"/>
            <effectiveTime value="200804021021"/>
            <participant typeCode="DST">
              <participantRole/>
            </participant>
          </act>
        </entry>
      </section>
    </component>
  </section>
</component>

```

```

</component>
<component>
  <section classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.4" extension="4e7e0004-8e9e-44d2-9a9e-099d071e646a"/>
    <code code="MUAYENE" codeSystem="2.16.840.1.113883.3.129.2.2.2"
      codeSystemName="Veriseti" codeSystemVersion="1.0" displayName="Muayene Veriseti"/>
    <text>Hastanın tezle uğraşırken geçirdiği hastalıkları özetler</text>
    <author typeCode="AUT" contextControlCode="OP">
      <time value="20080502102643"/>
      <assignedAuthor classCode="ASSIGNED">
        <id root="2.16.840.1.113883.3.129.1.1.1" extension="12345678905"/>
      </assignedAuthor>
    </author>
  </section>
  <component typeCode="COMP" contextConductionInd="true">
    <section classCode="DOCSECT" moodCode="EVN">
      <id root="2.16.840.1.113883.3.129.2.1.5"
        extension="de380a98-d31f-4a33-a878-7366da629737"/>
      <code code="PROTOKOLNO" codeSystem="2.16.840.1.113883.3.129.2.2.3"
        codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
        displayName="Protokol No Bilgisinin Olduğu Bölüm"/>
      <text mediaType="text/x-hl7-text+xml"/>
      <entry typeCode="COMP" contextConductionInd="true">
        <act classCode="ACT" moodCode="EVN">
          <id root="2.16.840.1.113883.3.129.1.1.4" extension="111"/>
          <code code="PROTOKOLNO"/>
        </act>
      </entry>
    </section>
  </component>
  <component typeCode="COMP" contextConductionInd="true">
    <section classCode="DOCSECT" moodCode="EVN">
      <id root="2.16.840.1.113883.3.129.2.1.5"
        extension="cba90686-fa28-4a0e-9ee6-342d7bcb3302"/>
      <code code="RAPOR" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
        codeSystemVersion="1.0" displayName="Rapor Verisinin Olduğu Bölüm"/>
      <text mediaType="text/x-hl7-text+xml"/>
      <entry typeCode="COMP" contextConductionInd="true">
        <observation classCode="OBS" moodCode="EVN">
          <code code="4" codeSystem="2.16.840.1.113883.3.129.1.2.8" codeSystemName="Rapor Türü"
            codeSystemVersion="1.0" displayName="Hastalık"/>
          <effectiveTime value="20080401"/>
          <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="ST">
            rapor dokümanı</value>
        </observation>
      </entry>
    </section>
  </component>
  <component typeCode="COMP" contextConductionInd="true">

```

```

<section classCode="DOCSECT" moodCode="EVN">
  <id root="2.16.840.1.113883.3.129.2.1.5"
    extension="59c0024f-d799-498d-8b00-e849136ad426"/>
  <code code="BULGU" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
    codeSystemVersion="1.0" displayName="Bulgu Verisinin Olduğu Bölüm"/>
  <text>Net bir problemle karşılaşılmadı</text>
</section>
</component>
<component typeCode="COMP" contextConductionInd="true">
  <section classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="e3c97aa2-ba47-40d4-a068-881a6c51417a"/>
    <code code="HIKAYE" codeSystem="2.16.840.1.113883.3.129.2.2.3"
      codeSystemName="Veri Kısmı"
      codeSystemVersion="1.0" displayName="Hikaye Verisinin Olduğu Bölüm"/>
    <text>Hastanın tezle uğraşırken geçirdiği hastalıkları özetler</text>
  </section>
</component>
<component typeCode="COMP" contextConductionInd="true">
  <section classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="6bde0f05-f303-4109-88ac-c22a23a67094"/>
    <code code="SIKAYET" codeSystem="2.16.840.1.113883.3.129.2.2.3"
      codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
      displayName="Şikayet Verisinin Olduğu Bölüm"/>
    <text>Bitkin</text>
  </section>
</component>
<component typeCode="COMP" contextConductionInd="true">
  <section classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="e59daa46-6c71-4eea-852e-406e5a440f13"/>
    <code code="TETKIK" codeSystem="2.16.840.1.113883.3.129.2.2.3"
      codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
      displayName="Tetkik Verisinin Olduğu Bölüm"/>
    <text mediaType="text/x-hl7-text+xml"/>
    <entry typeCode="COMP" contextConductionInd="true">
      <observation classCode="OBS" moodCode="EVN">
        <code code="901620" codeSystem="2.16.840.1.113883.3.129.1.2.2"
          codeSystemName="SUT" codeSystemVersion="1.0" displayName="Tam Kan (Hemogram)/>
        <performer typeCode="PRF">
          <assignedEntity classCode="ASSIGNED">
            <id root="2.16.840.1.113883.3.129.1.1.6" extension="4553"/>
          </assignedEntity>
        </performer>
      </observation>
    </entry>
    <entry typeCode="COMP" contextConductionInd="true">

```

```

<observation classCode="OBS" moodCode="EVN">
  <code code="701540" codeSystem="2.16.840.1.113883.3.129.1.2.2" codeSystemName="SUT"
    codeSystemVersion="1.0"
    displayName="Üzofagoskopi, gastroskopi, duodenoskopi (biri veya hepsi)"/>
  <performer typeCode="PRF">
    <assignedEntity classCode="ASSIGNED">
      <id root="2.16.840.1.113883.3.129.1.1.6" extension="4553"/>
    </assignedEntity>
  </performer>
</observation>
</entry>
</section>
</component>
<component typeCode="COMP" contextConductionInd="true">
  <section classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="cf813e90-e027-4441-9e87-c287bdc5b370"/>
    <code code="MUDAHALE" codeSystem="2.16.840.1.113883.3.129.2.2.3"
      codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
      displayName="Müdehale Verisinin Olduğu Bölüm"/>
    <text mediaType="text/x-hl7-text+xml"/>
    <entry typeCode="COMP" contextConductionInd="true">
      <procedure classCode="PROC" moodCode="EVN">
        <code code="520030" codeSystem="2.16.840.1.113883.3.129.1.2.2"
          codeSystemName="SUT" codeSystemVersion="1.0"
          displayName="NORMAL POLIKLINIK MUAYENE ÜCRETLERI"/>
      </procedure>
    </entry>
  </section>
</component>
<component typeCode="COMP" contextConductionInd="true">
  <section classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="6c4b1e87-c4f4-42b0-aaeb-05e23a77ed5a"/>
    <code code="TANI" codeSystem="2.16.840.1.113883.3.129.2.2.3"
      codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
      displayName="Tanı Verisinin Olduğu Bölüm"/>
    <text mediaType="text/x-hl7-text+xml"/>
    <entry typeCode="COMP" contextConductionInd="true">
      <observation classCode="OBS" moodCode="EVN">
        <code code="ANATANI" codeSystem="2.16.840.1.113883.3.129.2.2.6"
          codeSystemName="Tanı Tipi" codeSystemVersion="1.0" displayName="Ana Tanı"/>
        <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="CV"
          code="K29.8" codeSystem="2.16.840.1.113883.6.3" codeSystemName="ICD-10"
          codeSystemVersion="1.0" displayName="Duodenit"/>
      </observation>
    </entry>
  <entry typeCode="COMP" contextConductionInd="true">

```



```

    <observation classCode="OBS" moodCode="EVN">
      <code code="EKTANI" codeSystem="2.16.840.1.113883.3.129.2.2.6"
        codeSystemName="Tanı Tipi" codeSystemVersion="1.0" displayName="Ek Tanı"/>
      <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="CV"
        code="G43" codeSystem="2.16.840.1.113883.6.3" codeSystemName="ICD-10"
        codeSystemVersion="1.0" displayName="Migren"/>
    </observation>
  </entry>
</section>
</component>
<component typeCode="COMP" contextConductionInd="true">
  <section classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="212802ba-7c64-47d5-8e52-b02a6ea44531"/>
    <code code="MUAYENE" codeSystem="2.16.840.1.113883.3.129.2.2.3"
      codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
      displayName="Muayene Verisinin Olduğu Bölüm"/>
    <text mediaType="text/x-hl7-text+xml"/>
    <entry typeCode="COMP" contextConductionInd="true">
      <encounter classCode="ENC" moodCode="EVN">
        <effectiveTime>
          <low value="200803251021"/>
          <high value="200804011021"/>
        </effectiveTime>
        <participant typeCode="LOC">
          <participantRole classCode="SDLLOC">
            <code code="22" codeSystem="2.16.840.1.113883.3.129.1.2.1"
              codeSystemName="Klinikler" codeSystemVersion="1.0"
              displayName="Gastroenteroloji"/>
          </participantRole>
        </participant>
      </encounter>
    </entry>
  </section>
</component>
</section>
</component>
<component>
  <section classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.4"
      extension="dd425f3b-a09c-434a-a648-9d67d3775adf"/>
    <code code="KABUL" codeSystem="2.16.840.1.113883.3.129.2.2.2" codeSystemName="Veriseti"
      codeSystemVersion="1.0" displayName="Kabul Veriseti"/>
    <text mediaType="text/x-hl7-text+xml"/>
    <component typeCode="COMP" contextConductionInd="true">
      <section classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.5"
          extension="c25dd89c-6850-49eb-bf10-c0df3e92e5bf"/>

```

```

<code code="SGKTAKIPNO" codeSystem="2.16.840.1.113883.3.129.2.2.3"
  codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
  displayName="SGK Takip No Bilgisinin Olduğu Bölüm"/>
<text mediaType="text/x-hl7-text+xml">123</text>
</section>
</component>
<component typeCode="COMP" contextConductionInd="true">
  <section classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="54083c07-d5f3-4a85-9fb9-8fcacb253a0b"/>
    <code code="SEVK" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
      codeSystemVersion="1.0" displayName="Sevk Verisinin Olduğu Bölüm"/>
    <text mediaType="text/x-hl7-text+xml"/>
    <entry typeCode="COMP" contextConductionInd="true">
      <act classCode="ACT" moodCode="EVN">
        <code code="SEVK"/>
        <effectiveTime value="20080313"/>
        <participant typeCode="REF" contextControlCode="OP">
          <participantRole classCode="SDLOC">
            <code code="22" codeSystem="2.16.840.1.113883.3.129.1.2.1"
              codeSystemName="Klinikler" codeSystemVersion="1.0"
              displayName="Gastroenteroloji"/>
          </participantRole>
        </participant>
        <entryRelationship typeCode="COMP">
          <observation classCode="OBS" moodCode="EVN">
            <code code="SEVKTANISI" codeSystem="2.16.840.1.113883.3.129.2.2.6"
              codeSystemName="Tanı Tipi" codeSystemVersion="1.0" displayName="Sevk Tanısı"/>
            <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="CV"
              code="K55" codeSystem="2.16.840.1.113883.6.3" codeSystemName="ICD-10"
              codeSystemVersion="1.0" displayName="BARSAĞIN VASKÜLER BOZUKLUKLARI"/>
          </observation>
        </entryRelationship>
      </act>
    </entry>
  </section>
</component>
<component typeCode="COMP" contextConductionInd="true">
  <section classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="41f0e5f8-e66a-4af7-bcb9-0dc9a93ba720"/>
    <code code="KABUL" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
      codeSystemVersion="1.0" displayName="Kabul Verisinin Olduğu Bölüm"/>
    <text mediaType="text/x-hl7-text+xml"/>
    <entry typeCode="COMP" contextConductionInd="true">
      <act classCode="ACT" moodCode="EVN">
        <code code="1" codeSystem="2.16.840.1.113883.3.129.1.2.7" codeSystemName="Kabul Şekli"
          codeSystemVersion="1.0" displayName="Diğer"/>
      </act>
    </entry>
  </section>
</component>

```

```

        <effectiveTime value="200803231021"/>
    </act>
</entry>
</section>
</component>
<component typeCode="COMP" contextConductionInd="true">
    <section classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.5"
            extension="537a52e1-36f1-4e04-a4bd-a71056471913"/>
        <code code="VAKATURU" codeSystem="2.16.840.1.113883.3.129.2.2.3"
            codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
            displayName="Vaka Türü Bilgisinin Olduğu Bölüm"/>
        <text mediaType="text/x-hl7-text+xml"/>
        <entry typeCode="COMP" contextConductionInd="true">
            <act classCode="ACT" moodCode="EVN">
                <code code="1" codeSystem="2.16.840.1.113883.3.129.1.2.53" codeSystemName="Vaka Türü"
                    codeSystemVersion="1.0" displayName="Normal"/>
            </act>
        </entry>
    </section>
</component>
</section>
</component>
<component>
    <section classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.4" extension="388c478d-c649-4273-803f-51b3c383f6ac"/>
        <code code="RECETE" codeSystem="2.16.840.1.113883.3.129.2.2.2" codeSystemName="Veriseti"
            codeSystemVersion="1.0" displayName="Reçete Veriseti"/>
        <text mediaType="text/x-hl7-text+xml"/>
        <author typeCode="AUT" contextControlCode="OP">
            <time value="20080502102643"/>
            <assignedAuthor classCode="ASSIGNED">
                <id root="2.16.840.1.113883.3.129.1.1.1" extension="12345678905"/>
                <id root="2.16.840.1.113883.3.129.1.1.2" extension="12345678906"/>
            </assignedAuthor>
        </author>
        <component typeCode="COMP" contextConductionInd="true">
            <section classCode="DOCSECT" moodCode="EVN">
                <id root="2.16.840.1.113883.3.129.2.1.5"
                    extension="4ec898be-2c17-4637-b383-9139be24b305"/>
                <code code="ILAC" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
                    codeSystemVersion="1.0" displayName="İlaç Verisinin Olduğu Bölüm"/>
                <text mediaType="text/x-hl7-text+xml"/>
                <entry typeCode="COMP" contextConductionInd="true">
                    <substanceAdministration classCode="SBADM" moodCode="RQO">
                        <effectiveTime xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
                            xsi:type="PIVL_TS">
                            <period value="1" unit="d"/>
                        </substanceAdministration>
                    </entry>
                </section>
            </component>
        </author>
    </section>
</component>

```

```

</effectiveTime>
<routeCode code="1" codeSystem="2.16.840.1.113883.3.129.1.2.47"
  codeSystemName="İlaç Kullanım Şekli" codeSystemVersion="1.0"
  displayName="Ağızdan (Oral)"/>
<doseQuantity value="1"/>
<consumable typeCode="CSM">
  <manufacturedProduct classCode="MANU">
    <manufacturedMaterial determinerCode="KIND" classCode="MMAT">
      <code code="8699536160085" codeSystem="2.16.840.1.113883.3.129.1.2.3"
        codeSystemName="İlaçlar" codeSystemVersion="1.0"
        displayName="LANSOR MIKROPELLET KAPSÜL 30 MG 28 CAP"/>
    </manufacturedMaterial>
  </manufacturedProduct>
</consumable>
<entryRelationship typeCode="COMP" contextConductionInd="true">
  <supply moodCode="INT" classCode="SPLY">
    <quantity value="2"/>
  </supply>
</entryRelationship>
</substanceAdministration>
</entry>
</section>
</component>
<component typeCode="COMP" contextConductionInd="true">
  <section classCode="DOCSECT" moodCode="EVN">
    <id root="2.16.840.1.113883.3.129.2.1.5"
      extension="d7cb1191-033d-4730-9236-d3782ddf0b60"/>
    <code code="RECETEBASLIK" codeSystem="2.16.840.1.113883.3.129.2.2.3"
      codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
      displayName="Reçete Başlık Bilgisinin Olduğu Bölüm"/>
    <text mediaType="text/x-hl7-text+xml"/>
    <entry typeCode="COMP" contextConductionInd="true">
      <act classCode="ACT" moodCode="EVN">
        <code code="1" codeSystem="2.16.840.1.113883.3.129.1.2.48"
          codeSystemName="Reçete Türü" codeSystemVersion="1.0" displayName="Normal"/>
        <effectiveTime value="20080401"/>
        <entryRelationship typeCode="COMP">
          <act classCode="ACT" moodCode="EVN">
            <id root="2.16.840.1.113883.3.129.1.1.4" extension="111"/>
            <code code="protocolNo"/>
          </act>
        </entryRelationship>
        <entryRelationship typeCode="COMP">
          <act classCode="ACT" moodCode="EVN">
            <id root="2.16.840.1.113883.3.129.1.1.3" extension="12321123"/>
            <code code="sGKRationBook"/>
          </act>
        </entryRelationship>
      </act>
    </entry>
  </section>
</component>

```

```

        </act>
    </entry>
</section>
</component>
<component typeCode="COMP" contextConductionInd="true">
    <section classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.5"
            extension="a0a7a2ea-453a-4b52-b741-d39ff1f968b0"/>
        <code code="SOSYALGUVENCEDURUMU" codeSystem="2.16.840.1.113883.3.129.2.2.3"
            codeSystemName="Veri Kısmı" codeSystemVersion="1.0"
            displayName="Sosyal Güvence Durumu Bilgilerinin Olduğu Bölüm"/>
        <text mediaType="text/x-hl7-text+xml"/>
        <entry typeCode="COMP" contextConductionInd="true">
            <act classCode="ACT" moodCode="EVN">
                <code code="1" codeSystem="2.16.840.1.113883.3.129.1.2.11"
                    codeSystemName="Sosyal Güvenlik" codeSystemVersion="1.0" displayName="SSK"/>
            </act>
        </entry>
    </section>
</component>
<component typeCode="COMP" contextConductionInd="true">
    <section classCode="DOCSECT" moodCode="EVN">
        <id root="2.16.840.1.113883.3.129.2.1.5"
            extension="2a1cbba1-cd5b-4ca1-b602-878c087b9a49"/>
        <code code="TANI" codeSystem="2.16.840.1.113883.3.129.2.2.3" codeSystemName="Veri Kısmı"
            codeSystemVersion="1.0" displayName="Tanı Verisinin Olduğu Bölüm"/>
        <text mediaType="text/x-hl7-text+xml"/>
        <entry typeCode="COMP" contextConductionInd="true">
            <observation classCode="OBS" moodCode="EVN">
                <code code="ANATANI" codeSystem="2.16.840.1.113883.3.129.2.2.6"
                    codeSystemName="Tanı Tipi" codeSystemVersion="1.0" displayName="Ana Tanı"/>
                <value xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:type="CV"
                    code="K29.8" codeSystem="2.16.840.1.113883.6.3" codeSystemName="ICD-10"
                    codeSystemVersion="1.0" displayName="Duodenit"/>
            </observation>
        </entry>
    </section>
</component>
</section>
</structuredBody>
</component>
</ClinicalDocument>

```